RGB624/RGB624DB

170/220 MHz Digital Video Palette DAC

Product Description

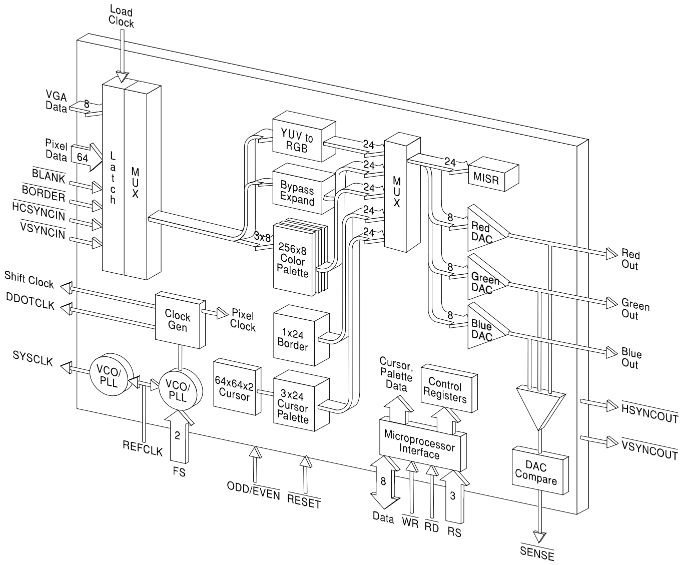
The RGB624™ and RGB624DB™ High Performance Digital Video Palette DACs from IBM bring photorealistic color to the mainstream market. With their custom, real-time colorspace conversion circuitry, the RGB624 and RGB624DB enable the concurrent display of business graphics, true-color digital video and photorealistic natural images. The RGB624DB's double-buffered pixel modes enhance the display of 3D and digital video images.

The full capabilities of the RGB624 and RGB624DB are realized with as little as half of the graphics memory previously required. Astonishingly vivid natural images can be obtained at 1024x768 resolution with only 2-Megabytes and up to 1600x1280 resolution with only 4-Megabytes.

The RGB624 and RGB624DB bring to the mainstream market features formerly found only in graphics workstations, including fully programmable YUV-to-RGB conversion and simultaneous multiple pixel formats.

The RGB624 and RGB624DB provide vibrant color with unsurpassed image stability and bright, flicker-free display even on large, high-resolution monitors.

Functional Block Diagram



Product Highlights

- ❑ 170, 220 MHz operation
- ❑ RGB524 and RGB526/RGB526DB pin and register compatible
- ❑ Fully programmable YUV-RGB conversion
- ❑ Mixed YUV and RGB pixels
- ❑ 64/32-bit wide pixel data bus
- ❑ Double-buffered pixels for 3D and digital video
- ❑ Keyed overlays
- ❑ Fine-grained PLL programming optimizes display
- ❑ Display modes up to 1600x1280
- ❑ Large Screen ISO-compliant refresh rates
- ❑ Packed 24-bit pixels
- ❑ 24-bit pixels with 8-bit overlay
- ❑ 4/8/16/24/32-bits per pixel
- ❑ Direct color (YUV or RGB)
- ❑ Gamma correction
- ❑ 256-shade gray scale
- ❑ Three 256x8 color palette RAMs
- ❑ Anti-sparkle Circuitry
- ❑ Per-pixel palette bypass control
- ❑ Palette paging
- ❑ Triple monotonic 8-bit DACs
- ❑ Two on-chip clock generators
- ❑ 64x64/32x32 translucent hardware cursor
- ❑ 100 MHz 8-bit VGA Data Input
- ❑ On-chip diagnostic functions
- ❑ Low-power 3.3V operation
- ❑ 5V-tolerant inputs
- ❑ Power-down modes
- ❑ 144 pin QFP package
- ❑ 0.8 µ CMOS

Applications

- ❑ Digital Video and Multimedia
- ❑ Graphical user interfaces
- ❑ Business graphics
- ❑ 3D Games and virtual reality
- ❑ CAD/CAM
- ❑ Medical Imaging
- ❑ Scientific Visualization

Table of Contents

List of Tables

List of Figures

1.0 Microprocessor Access

As seen on the microprocessor bus there are eight I/O addresses, selected by RS[2:0]. Two indirect schemes are used to access all of the internal registers and arrays through these eight primary I/O addresses.

The first scheme is standard VGA, and operates when RS[2] = 0. Of the four I/O addresses then available with RS[1:0], only one address directly accesses a register, the Pixel Mask. The other three addresses are used to indirectly access the three 256x8 palettes.

The second scheme is an indexed scheme and is used to access all of the remaining registers including the cursor array. This scheme operates when RS[2] =1. Of the four I/O addresses then available using RS[1:0], two are used to load an index register (Low and High). The third address is used to write or read the register or array position pointed to by the index register. The fourth address is used to directly access a register which controls whether the index register automatically increments following an indexed register access.

The eight I/O addresses selected by RS[2:0] are listed in Table 1 below:

Table 1. I/O Addresses

RS[2:0]	Register
000	Palette Address (Write Mode)
001	Palette Data
010	Pixel Mask
011	Palette Address (Read Mode)
100	Index Low
101	Index High
110	Index Data (Indexed Registers)
111	Index Control

1.1 VGA Access

1.1.1 Palette

Internally the three 256x8 palettes are accessed by the microprocessor as a single 256x24 palette, with all 24 bits written or read in one operation.

A single Palette Address register points to 1 of 256 locations for writing or reading the 24 bits. Two different Register Select addresses are used to access the Palette Address register.

A write to RS[2:0] = 000 (Palette Address Write Mode) initializes the palette logic for write operations. Subsequent writes to Palette Data (RS[2:0] = 001) will load internal palette color registers and cause these register contents to be written into the palettes.

A write to RS[2:0] = 011 (Palette Address Read Mode) initializes the palette logic for read operations. Data from the palettes will be loaded into internal palette color registers. Subsequent reads from Palette Data (RS[2:0] = 001) will read these palette color registers.

Every three accesses of Palette Data (RS[2:0] = 001) will cause the Palette Address register to be incremented. An increment past 0xff will “wrap around” to 0x00.

A read from either Palette Address (Write Mode) or Palette Address (Read Mode) will read the Palette Address register. The same register is used for writing and reading, thus, changing modes destroys the contents of the previous mode’s palette address. For example, if some reads are performed and then Palette Address (Write mode) is written, the read address will be lost and a read of either Palette Address (Write Mode) or Palette Address (Read Mode) will produce the same result: the address that was written into Palette Address (Write Mode).

1.1.2 Palette Write

Palette writes must be initialized by writing the Palette Address (Write Mode) register. This provides a starting address for writes and initializes the internal circuitry for palette write operations.

Palette writes are then performed by writing to Palette Data in a red, green, blue... sequence. These writes will load internal palette data registers in sequence. Immediately following every third write, an internal write will be triggered to the palette of the 24 bits contained in the internal palette data registers, at the address contained in the Palette Address register.

Immediately following the internal palette write triggered by the third write to Palette Data, the Palette Address register will be incremented. Thus, continuous writes to Palette Data will load the palette, stepping through the palette addresses in ascending order.

1.1.3 Palette Read

Palette reads must be initialized by writing the Palette Address (Read Mode) register. This provides a starting address for reads and initializes the internal circuitry for palette read operations.

Immediately following the writing of Palette Address (Read Mode), a read of the palette will be performed at the address just written. Internal palette data registers are loaded with the read data, and the Palette Address register is incremented.

Palette reads are then performed by reading from Palette Data. Red, green, blue... data from the preloaded internal registers will be presented in sequence. Immediately following every third read, an internal read of the palette to the 24 bits contained in the internal registers will be performed at the address contained in the Palette Address register.

Immediately following the internal palette read triggered by the third read of Palette Data, the Palette Address register will be incremented. Thus, continuous reads of Palette Data will read the palette, stepping through the palette addresses in ascending order.

1.1.4 6/8 Bit Palette Access

The original VGA had 6-bit DACs and 6-bit palette entries, and the low order 6 bits from/to the microprocessor port were written/read into the palette.

For RGB624/RGB624DB, the DACs and palette entries are 8 bits. For non-VGA emulation all 8 bits are used. To emulate 6-bit VGA operation the upper 6 bits of the palette hold the VGA 6-bit color and the two low order bits are set to 00. The COL RES bit (color resolution) of the Miscellaneous Control 2 register determines if the access is 6-bit or 8-bit.

The reset condition is to emulate VGA using the 6 low order microprocessor data bits. COL RES is set to 6 bits. In this mode, for writing, microprocessor bits [7:6] are discarded, bits [5:0] are shifted to bits [7:2], and bits [1:0] are set to 00 before being written into the internal data registers. For reading, the internal data register bits [7:2] are shifted to bits [5:0], and bits [7:6] are set to 00 before being presented on the microprocessor data signals.

If COL RES is set to 8 bits then all 8 bits from/to the microprocessor will be written to and read from the color palette registers.

Note that the 6-to-8 bit translation is only done between the microprocessor port and the internal data registers. Internally, on writes, all 8 bits of the internal registers are written to the palette, and on reads, the internal registers retain all 8 bits read from the palette. Thus, if the palette is loaded with 8-bit values with COL RES set to 8 bits, and then the palette is read with COL RES set to 6 bits, the internal palette color registers will still be loaded with the 8 bits that were written into the palette. But the data read on the microprocessor data lines will be 6 bits.

1.1.5 Palette Clocking

Palette accesses are synchronized internally with the pixel clock. On writes, the pixel values of the previous cycle are held and displayed during the write cycle. Both of these features minimize disturbance of displayed pixels when the palette is accessed (anti-sparkle).

The pixel clock (as selected by the PCLK SEL bits in Miscellaneous Control 2) must be running for palette access to be valid.

The timings for the microprocessor signals are specified in units of pixel clocks. These specifications are derived from the requirement for the pixel clock to be running for palette access, as well as to allow time for the Palette Accesses and Palette Address increments to occur internally following a palette access.

1.1.6 Palette Access Status

The original VGA logic had an override for read accesses of the Palette Address (Read Mode) register. Instead of reading the Palette Address register, a value was returned that indicates the status of the last palette access, write or read.

The reset condition of RGB624/RGB624DB is to return the address value for a read of Palette Address (Read Mode). The VGA logic may be emulated by setting the RADR RFMT bit in Miscellaneous Control 1. This causes the status of the last palette access to be returned.

The value of the status returned is 0x00 if the last write to Palette Address was Write Mode, and 0x03 if the last write to Palette Address was Read Mode.

1.1.7 Pixel Mask

The pixel mask is an 8-bit register addressed with RS[2:0] = 010. It can be accessed at any time without disturbing a palette write or read sequence.

Accesses to the pixel mask are asynchronous to the pixel clock. Temporary color disturbances can be expected if the mask is changed while displaying pixels through the palette.

1.2 Indexed Access

The cursor array and a number of control registers are addressed with an internal 11-bit index register. The microprocessor accesses this as Index High (RS[2:0] = 101) and Index Low (RS[2:0] = 100).

A write or read to Index Data (RS[2:0] = 110) actually writes or reads the register/cursor array location addressed by the Index register.

Following a write or read of Index Data, the index register will increment if the INDX CNTL bit is set. The Index Control register (RS[2:0] = 111) contains this bit. To allow for future expansion, wraparound from 0x07ff to 0x0000 is **not** supported.

In general, access of Index Low, Index High, Index Control, or any of the Indexed registers is independent of the palette access and will not disturb a palette write or read sequence. However, as described above the PADR RFMT bit in Miscellaneous Control 1, the COL RES bit in Miscellaneous Control 2, and the 6BIT ACC bit in Palette Control all affect palette access.

Also, as described above, the pixel clock must be running for valid access of the palette, and the pixel clock is affected by a number of indexed registers.

1.2.1 Cursor Array

In general, the indexed registers may be written or read at any time, using the address held in Index High and Index Low. This address may be set by writing to Index High or Index Low, or the value may result from the auto-increment action of a previous access.

However, as described in section 10.2.3, "Cursor Array Reads" on page 31, to access the cursor array a write to Index High or Index Low must be performed first. That is, the cursor array cannot be accessed by auto-increment from address 0x00ff to 0x0100.

Also, as with the palette, the pixel clock must be running to access the cursor array.

2.0 Clocking

2.1 Clock Generators

There are two on-board clock generators: pixel clock and system clock (SYSCLK). Each clock generator uses a separate programmable phase locked loop (PLL).

The pixel clock generator provides the fundamental "dot" timings; it serves generally as the clock both for internal chip clocking and for on-card CRT timings.

The system clock generator is provided for the convenience of the graphics subsystem design. No internal use is made of this clock; the clock generator simply drives the SYSCLK output of the chip.

2.2 PLL Input

2.2.1 REFCLK

The REFCLK input is a reference clock that the PLLs use in conjunction with programming registers to produce a wide variety of frequencies.

In general, REFCLK can be any frequency from 1 MHz through 100 MHz. However, as discussed below, the two clock generators each come up at "start up" frequencies which are dependant on the REFCLK frequency. This can govern the value chosen for REFCLK, depending on the application requirements at "reset" time.

Following a reset, the PLL driving the SYSCLK output is enabled with the start-up frequency:

$$SYSCLK\ frequency = (33/16) \times REFCLK\ frequency$$

The pixel clock PLL has four start up frequencies:

$$F0\ frequency = (7/4) \times REFCLK\ frequency$$

$$F1\ frequency = (79/40) \times REFCLK\ frequency$$

$$F2\ frequency = Undefined$$

$$F3\ frequency = Undefined$$

The desired frequency is selected externally using the FS[1:0] input pins.

The start-up values are chosen for use with REFCLK = 14.31818 MHz (a common graphics adapter frequency):

$$SYSCLK\ frequency = (33/16) \times 14.31818 = 29.53\ MHz$$

$$F0\ frequency = (7/4) \times 14.31818 = 25.057\ MHz$$

$$F1\ frequency = (79/40) \times 14.31818 = 28.278\ MHz$$

This causes the SYSCLK start up frequency to be approximately 30 MHz and the pixel clock frequencies to be approximately the standard VGA frequencies 25.175 MHz and 28.322 MHz.

With a 14.31818 MHz REFCLK the start-up pixel clock frequencies selected by FS[1:0] are:

Table 2. Start-up Pixel Clock Frequency

FS[1:0]	Frequency Selected	Value
00	F0	25.057 MHz
01	F1	28.278 MHz
10	F2	Undefined
11	F3	Undefined

2.3 SYSCLK PLL Output

The system clock PLL drives the SYSCLK output. This output can be 3-stated with Bit 6, SYSC DSAB, in the System Clock Control register.

The supported frequency range for SYSCLK is 8.125 MHz to 100 MHz.

2.4 Pixel PLL Outputs

The pixel PLL is used internally as the pixel clock. The maximum allowed generated frequency is 170/220 MHz, dependent on the product version.

The pixel PLL Output is not available directly. However, two divided versions are provided as output signals:

- ❑ SCLK
- ❑ DDOTCLK

2.4.1 SCLK

SCLK (Serial Clock) is intended for clocking of the serial outputs of the VRAMs to the pixel port inputs. As such, the divide factor is a function of the VRAM pixel port width (64 or 32 bits), and the number of pixels contained in an access. For example, with a VRAM width of 64 and operating at 16 bits-per-pixel, there will be 64/16 = 4 pixels brought in with each VRAM access, and SCLK will operate at 1/4 the frequency of the pixel PLL output.

If the VGA port is selected SCLK will simply be the output of the pixel PLL. Table 3, "SCLK Frequencies," is a table of all the SCLK frequencies that are produced.

Table 3. SCLK Frequencies

BPP	VRAM=32	VRAM=64
4	÷ 8	÷ 16
8	÷ 4	÷ 8
15/16	÷ 2	÷ 4
15/16 DB	÷ 1	÷ 2
32	÷ 1	÷ 2
24 Packed	Invalid	÷ (8/3)
VGA	÷ 1	

“24 Packed” is a special case. It is only valid with a VRAM width of 64, and it produces 3 SCLKs for every 8 internal pixel clocks as shown in Figure 1.

“15/16 DB” (double buffer) is also a special case, and is only used on the RGB624DB. When 16 BPP mode is used, and double buffer operation is enabled, 32 bits will be fetched for each pixel. SCLK will run at the same frequency as for 32 BPP mode.

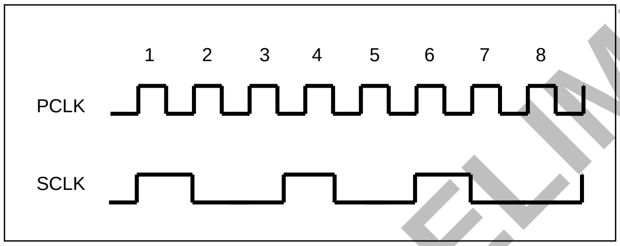


Figure 1. SCLK for 24 BPP Packed

The SCLK output can be inverted with the SCLK INVT bit of the Miscellaneous Clock Control register. This allows either polarity to be used, as desired, to aid in meeting critical timings at the card level.

The SCLK DSAB bit of the Miscellaneous Clock Control register can be used to 3-state the SCLK output if desired.

2.4.2 DDOTCLK

DDOTCLK (Divided Dot Clock) is simply the pixel PLL output divided by 1, 2, 4, 8 or 16 as determined by the DDOT DIV bits of the Miscellaneous Clock Control register.

Note that the maximum supported output frequency of DDOTCLK is 100 MHz, so some values of the DDOT DIV bits will become illegal when the pixel PLL is programmed to operate beyond this frequency.

When a pixel format of 24 BPP Packed is selected, the SCLK output may be driven on DDOTCLK instead of the divided pixel PLL output, under control of the B24P DDOT bit of the Miscellaneous Clock Control register.

DDOTCLK is similar to the SYSCLK output in that it is provided for general card use and is not used internally. However, its frequency is slaved to the pixel clock, whereas SYSCLK is independent. The DDOTCLK may be 3-stated with the DDOT DSAB bit of the Miscellaneous Clock Control register if desired.

Note: Although DDOTCLK and SCLK are both derived from the pixel clock. there is otherwise no particular relationship between the two clocks. In particular, the two clocks do not have any associated phase relationship.

2.5 Additional Clocks

2.5.1 Load Clock

The LCLK input (Load Clock) is used to latch up all incoming pixel data and video controls. The maximum frequency of this input is 100 MHz.

2.5.2 Pixel Clock (Dot Clock)

The pixel clock, or dot clock, is the internal clock used to clock pixel data up through the DACs. It is also required to be running to access the palette and the cursor array. The maximum frequency of this clock is 170/220 MHz (depending on the product rated speed).

There are several sources of the pixel clock, as selected by the PCLK SEL bits in the Miscellaneous Control 2 register:

LCLK input This is the reset default. It is intended to be used when the VGA port is selected as the pixel source.

Pixel PLL output This is intended to be used when the VRAM pixel port is selected as the pixel source. It provides the highest pixel clock operation.

REFCLK input This is intended for laboratory bringup.

When LCLK is selected as the pixel clock all internal pixel operations are synchronous with LCLK. If the pixel clock is sourced by the pixel PLL output or REFCLK, then the incoming pixels and video controls are expected to be derived from SCLK. After latching the signals with LCLK, the signals are clocked with an internal SCLK, and then clocked with the internal pixel clock.

2.6 PLL Operation Compatibility with RGB51x/RGB52x

Although the RGB624/RGB624DB is generally upward compatible with the RGB51x and RGB52x products the method for programming the two PLLs as described in the following sections is not the same. The RGB51x and RGB52x products have restrictions on the programming values which are removed with the RGB6xx products.

For software compatibility with the RGB51x and RGB52x the “restricted” programming mode used by those products is retained. This programming method is described in Appendix D.0 “PLL Compatibility Programming” on page 91.

The programming mode for the SYSCLK PLL is determined by the PROG MODE bit of the System Clock Control register (index 0x0008). This bit must be set to ‘1’ to use the method described in the following sections. When set to ‘0’ the compatibility mode described in Appendix D.0 is used.

The programming mode for the pixel PLL is determined by the EXT/INT bits of the Pixel PLL Control 1 register (index 0x0010). These bits when set to 100 or 101 will use the method described in the following sections. When set to 000, 001, 010, or 011 the compatibility mode described in Appendix D.0 is used.

2.7 PLL Operation and Programming

The two PLLs are generally identical in their operation and programming. A simplified diagram of the PLL is shown in Figure 2. The PLL takes the incoming reference clock, REFCLK, and generates the CLOCKOUT output. The frequency of CLOCKOUT is determined by three programming values contained in registers, M, N and P. A fourth value, C, is required to set the operating points of the analog circuits.

The heart of the PLL is the VCO (voltage controlled oscillator). The VCO can operate over the range of 65 MHz to 170 MHz/220 MHz (depending on the product speed chosen).

The VCO voltage input value is produced by comparing a divided version of the VCO output with a reference frequency. The value M sets the divide value for the VCO output. Values for M can be 2 through 127 (0 and 1 are illegal). ‘1’ is added to M, to produce a divide value of 3 through 128.

The internal reference frequency (f_{INTREF}) is produced by dividing the incoming REFCLK, with the divide value set by N. N can range from 0 through 63, and ‘1’ is added to this value to produce a divider value of 1 through 64.

The divided VCO output frequency is compared to the internal reference f_{INTREF} by a phase comparator. The phase comparator drives a charge pump which drives a filter connected to the VCO. A capacitor in the filter develops the voltage supplied to the VCO. The voltage goes up or down depending on whether the phase comparator is driving the charge pump current up or down.

When the divided VCO frequency is equal to f_{INTREF} no pump current is produced, the voltage to the VCO stays constant, and the VCO frequency stays constant. If the VCO tends to drift in frequency the phase comparator will detect the difference and will drive filter voltage, via the charge pump, in the appropriate direction. This adjusts the VCO frequency such that the frequency difference at the input to the phase comparator again becomes zero. With a constant f_{INTREF} the VCO frequency will be “locked” to the internal reference.

The internal reference frequency is:

(1) $$f_{INTREF} = \frac{f_{REFCLK}}{N+1}$$

The VCO frequency is:

(2) $$_{VCO} = f_{INTREF} \times (M+1) = f_{REFCLK} \times \frac{M+1}{N+1}$$

The output of the VCO is divided down by 1, 2, 4, 6 or 8, depending on the P programming value, to produce the final programmed frequency. For example, a clock frequency of 28 MHz is produced by programming the VCO to oscillate at 4 x 28 = 112 MHz, and then setting P to divide the VCO output by 4.

Values for P are 0 through 4 for the Pixel PLL and 1 through 4 for the SYSCLK PLL. When P is 0 the divide value is 1; when P = 1, 2, 3 or 4 the divide factor is 2 x P.

The generated CLOCK OUT frequency is:

(3a) $$_{CLOCKOUT} = f_{REFCLK} \times \frac{M+1}{N+1};\ P = 0$$

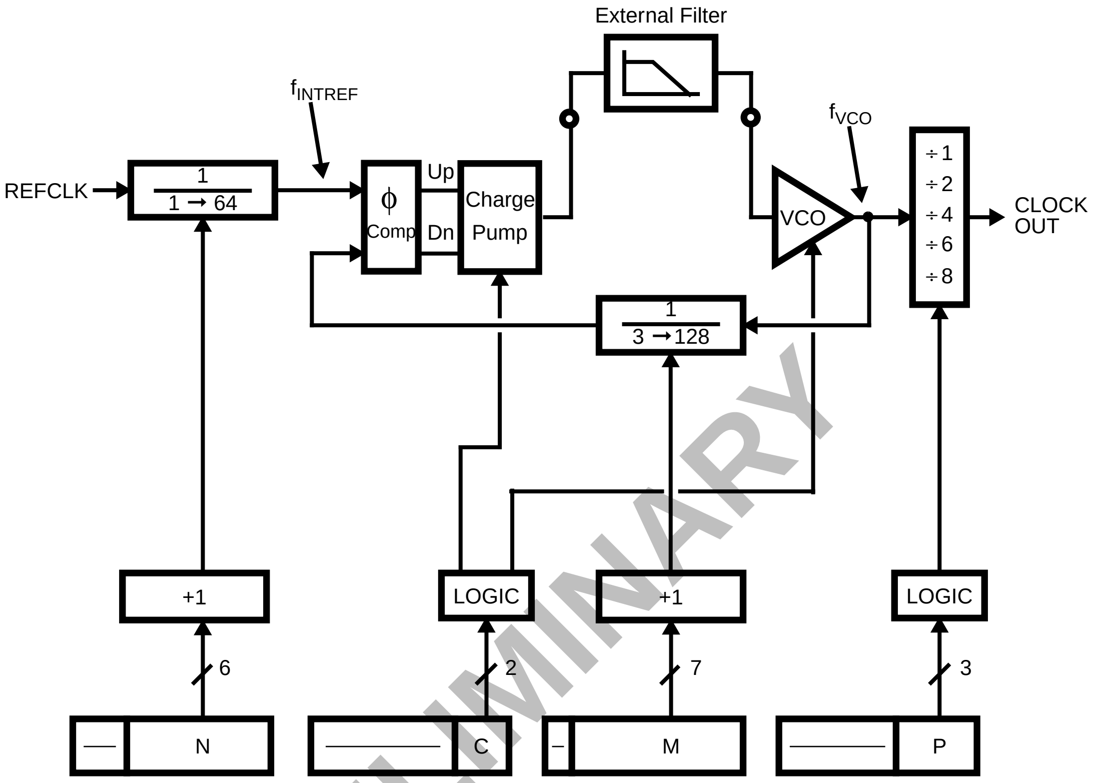
(3b) $$_{CLOCKOUT} = f_{REFCLK} \times \frac{M+1}{(N+1) \times 2P};\ P = 1, 2, 3, 4$$

Internal to the PLL the charge pump bias current and the VCO gain must be adjusted depending on the frequency of operation. These adjustments are controlled by the C programming bits:

(4a) $$\text{et}\ C = 1;\ \text{when}\ 65\ \text{MHz} \leq f_{VCO} \leq 128\ \text{MHz}$$

(4b) $$\text{et}\ C = 2;\ \text{when}\ f_{VCO} > 128\ \text{MHz}$$

Other values of C are reserved.



Input Values	$1\ \text{MHz} \le f_{REFCLK} \le 100\ \text{MHz}$ $2 \le M \le 127$ $0 \le N \le 63$ $0 \le P \le 4$
Output Frequency	$65\text{MHz} \le \left(f_{CLOCKOUT} = f_{REFCLK} \times \frac{M+1}{N+1};\ P = 0\right) \le 170\ \text{MHz} / 220\ \text{MHz}$ 2.5,16.25 ,).83, 8.125 MHz $\le \left(f_{CLOCKOUT} = f_{REFCLK} \times \frac{M+1}{(N+1) \times 2P};\ P = 1, 2, 3, 4\right) \le 170\ \text{MHz} / 220$ MH
Requirements for Correct Operation	L MHz $\le \left(f_{INTREF} = \frac{f_{REFCLK}}{N+1}\right)$ $5\ \text{MHz} \le \left(f_{VCO} = f_{REFCLK} \times \frac{M+1}{N+1} = f_{INTREF} \times (M+1)\right) \le 170\ \text{MHz} / 220$ MH or the 2 PLLs, $f_{higher} \ne (n \times f_{lower}) \pm 3\ \text{MHz};\ n = 1,2,3.$
Additional input(s) to set operating point	et $C = 1$; when $65\ \text{MHz} \le f_{VCO} \le 128\ \text{MHz}$ set $C = 2$; when $f_{VCO} > 128\ \text{MHz}$

Figure 2. PLL Simplified Diagram

2.7.1 Additional Constraints

2.7.1.1 Internal Reference

To avoid excessive jitter the internal reference frequency f_{INTREF} must not be less than 1 MHz.

2.7.1.2 VCO Frequency Range

The minimum VCO frequency is 65 MHz. The maximum frequency is the maximum speed rating of the product: 170 MHz / 220 MHz. (E.g., if it is desired to have a Dot Clock of 150 MHz it is not permitted to program the PLL to run at 300 MHz with a P value of 1 to divide down to 150 MHz.)

2.7.1.3 PLL Interaction

The Pixel PLL and the SYSCLK PLL will interfere with each other (they will modulate the output frequency of the other PLL) if the higher frequency falls within 3 MHz of an integer multiple of the lower frequency. That is, if f_{higher} is the higher of the two frequencies and f_{lower} is the lower frequency, then the following equation must be satisfied:

$$(5a)\quad f_{higher} \neq (n \times f_{lower}) \pm 3\text{ MHz};\ n = 1,2,3\ldots$$

2.7.2 Programming Summary

1. Select M, N and P values that produce the desired frequency using equation (3a) or (3b). Note that for the SYSCLK PLL a value of ‘0’ for P is not permitted, so equation (3a) is not valid for this PLL.
2. Verify that f_{INTREF} is not less than 1 MHz, using equation (1).
3. Verify that f_{VCO} is not less than 65 MHz and not greater than 170 MHz / 220 MHz, using equation (2).
4. Verify that the higher frequency of the 2 PLLs, plus or minus 3 MHz, is not an integral multiple of the lower frequency PLL (equation (5a)).
5. If all of the above conditions are met then M, N and P are valid values. Select the appropriate C value using equation (4a) or (4b).

2.7.3 Glitching on Frequency Change

When the operating frequency of either PLL is changed by changing one of the programming register values, the transition from the original frequency to the new frequency can either occur smoothly or can glitch, depending upon the following:

1. If the P bits are not changed, then changing the M and N bits will not cause a glitch.
2. If the P value is changed then the PLL output can glitch. For the Pixel PLL there is no protection against this situation.

   For the SYSCLK PLL there is additional logic (not shown) which causes the “2,4,6,8” VCO divider circuit to update synchronously with the VCO output when the P value is changed, such that changing P will not cause SYSCLK to glitch.
3. Caution should be used when changing the C value. Although the output will not “glitch”, the frequency can change rapidly to intermediate values before settling to the new value.

The Pixel PLL has selectable banks of programming values (described below). Note that switching between banks has the same glitching considerations as changing single M, N, P or C values.

2.8 Programming Registers

2.8.1 SYSCLK PLL

When the PROG MODE bit of the System Clock Control register (index 0x0008) is set to ‘1’ the SYSCLK N, M, P, and C registers (indices 0x0015, 0x0016, 0x0017, 0x0018 respectively) are used to provide the SYSCLK PLL programming values.

When the PROG MODE bit is set to ‘0’ the registers at 0x0015 and 0x0016 are used for compatibility mode programming as described in Appendix D.0, and the registers at 0x0017 and 0x0018 are ignored.

2.8.2 Pixel PLL

For the Pixel PLL four banks of registers, each containing a set of M, N, P and C values are provided. The selection source of these banks, external or internal, is controlled by the EXT/INT bits of the PLL Control 1 register (index 0x0010).

When EXT/INT = ‘100’ the selection is external, with the values on the FS[1:0] pins used to select the register bank.

When EXT/INT = ‘101’ the selection is internal, with the value of the Pixel PLL Control 2 register used to select the register bank.

Table 4. Pixel PLL Bank Selection

FS[1:0] or PLL Control 2 [1:0]	Programming Registers	Register Indices
00	M0,N0,P0,C0	0x0020 - 0x0023
01	M1,N1,P1,C1	0x0024 - 0x0027
10	M2,N2,P2,C2	0x0028 - 0x0031
11	M3,N3,P3,C3	0x0032 - 0x0035

When EXT/INT = ‘000’ through ‘011’ the registers at 0x0020 through 0x0035 are used for compatibility mode programming as described in Appendix D.0. A register at 0x0014, the Fixed PLL Reference Divider, is used for compatibility mode programming but is ignored when EXT/INT = ‘100’ or ‘101’.

2.8.3 Diagnostic Readback

The read-only registers Pixel M Input (index 0x008e), Pixel N Input (index 0x008f), Pixel P Input (index 0x008c) and Pixel C Input (index 0x008d) contain the programming values actually used by the pixel PLL. These registers can be used to verify that the desired programming registers are the ones actually selected.

When compatibility mode programming is used the registers at 0x008e and 0x008f contain the Pixel PLL programming values as described in Appendix D.0, and the values in the registers at 0x008c and 0x008d are undefined.

2.9 PLL Disable

Following a reset the PLLs are “enabled” and will change frequencies with changes to the programming values. Each PLL can be separately “disabled”. In the disabled state the PLL does not respond to the M and N programming values. This causes the internal VCO to oscillate somewhere in the range of 20 KHz to 1 MHz.

Changing the C value may change the VCO frequency but the frequency will stay within the 20 KHz to 1 MHz range.

The VCO output continues to be affected by the P value, so the actual PLL output will be (20 KHz through 1 MHz) divided by 1, 2, 4, 6 or 8, for the Pixel PLL, and (20 KHz through 1 MHz) divided by 2, 4, 6 or 8, for the SYSCLK PLL.

The Pixel PLL is disabled by setting the PPLL ENAB in the Miscellaneous Clock Control register to 0. The SYSCLK PLL is disabled by setting the SPPL ENAB bit in the System Clock Control register to 0.

3.0 Modes of Operation

Pixel data can come from the VGA port or the VRAM pixel port, as selected by the PORT SEL bit of the Miscellaneous Control 2 register.

If the VRAM pixel port is selected, the pixel format can be 4 BPP (bits per pixel), 8 BPP, 15/16 BPP, 24 BPP Packed, or 32 BPP, selected by the Format bits of the Pixel Format register. Tables 6 through 10 show how the input bits are selected as a function of Pixel Format.

VGA data and 4 BPP data are always used to indirectly generate 24 bits of color by indexing into the 256 entry palettes. The Pixel Mask register is used to selectively mask off the index bits as desired.

8 BPP, 15/16 BPP, 24 BPP Packed, and 32 BPP from the VRAM pixel port can either be indirect (through the palettes) or direct (bypassing the palettes).

15/16 BPP, 24 BPP Packed, and 32 BPP can also be directed through a color space converter to convert incoming YUV format (video luminance/chrominance) to 24-bit RGB. Converted YUV pixels are always direct (bypassing the palette).

Each of these formats has an associated control register with bits to select indirect or direct color. Additionally 15/16 BPP and 32 BPP formats allow a bit within the incoming data to dynamically select indirect or direct color. There is also a mode for 15/16 BPP which allows a bit within the incoming data to select between RGB and YUV pixel formats.

As with VGA and 4 BPP, the Pixel Mask is used to mask off palette address bits with indirect color access for 8, 15/16, 24 Packed, and 32 BPP.

3.1 YUV Formats

With YUV encoding the Y value represents the luminance of the pixel (or sum of the RGB values), U represents the difference between the B component value and the Y value, and V represents the difference between the R component value and the Y value.

The 24 BPP Packed and 32 BPP YUV formats use 8 bits each for Y, U and V. These 24 bits are sent directly to the color space converter. The full 24-bit YUV format is also referred to as 4:4:4 format (from CCIR Recommendation 601.)

Additional details of the color space conversion are given in Section 7.0 on page 26.

The 16 BPP YUV format is “sub-sampled”. Each pixel contains an 8-bit Y value. The remaining 8 bits are either a U value or V value, with U and V alternating between pixels. Internally the 16 bits are expanded to 24 bits for presentation to the color space converter. This is done by using the same U value for every two pixels, and the same V value for every two pixels.

The sub-sampled format is also known as 4:2:2 format. It is discussed more fully in section 8.0, "Sub-sampled Chrominance Issues" on page 27.

A variation of 16-bit subsampled YUV is used with the 16 BPP “Tagged RGB/YUV” format. The high order bit is a tag that determines if the remaining 15 bits are RGB or YUV. If RGB, the format is 555 RGB, the same as 15-bit RGB. If YUV, the format is 15-bit subsampled YUV. The Y value can be 8 bits with 7 bit alternating U and V values, or the Y value can be 7 bits with 8 bit alternating U and V values.

3.2 Overlays and Key Operation

A pixel format is supported in which two pixels, overlay and underlay, are packed together and one or the other pixel is selected by “key” matching.

Two 8-bit registers, the “Key” and the “Key Mask” are provided. In general, the contents of the Key register are considered to have a value that is “transparent”, and is compared to the overlay value on a pixel by pixel basis. If the values do not match (the overlay is not transparent) then the overlay pixel is displayed. If the values match (the overlay is transparent) then the underlay pixel is displayed (it “shows through”).

The Key Mask register allows bits in the Mask register to be “don’t cared”. Bits that are ‘1’ in the Key Mask register will cause corresponding bits in the Mask register to be used in the compare operation with the overlay pixel. Bits that are ‘0’ will cause the corresponding bits in the Key register to be ignored. For example, if the bits in the Key Mask register are 11110000 then only the four high order bits of the overlay layer have to match the four high order bits of the Key register to switch to the underlay.

The key matching operation is enabled by setting the KEY ENAB bit of the Key Control operation. If the pixel format supports overlay/underlay then key matching will take place; otherwise the KEY ENAB bit has no effect.

The RGB624/RGB624DB supports key matching for the 16 BPP format and the 32 BPP format.

There are several 16 BPP formats; key matching can be used when the format is designated as 4:2:2 YUV. The Key value is considered to be a “Luma Key”. Overlay pixels use the entire 16-bit, 4:2:2 YUV value. Underlay pixels use just the low 8 bits or high 8 bits as indirect RGB.

With 32 BPP the upper 8 bits can be designated as an 8-bit RGB overlay, and the Key value is considered to be a “Chroma Key”. The remaining 24 bits are the underlay and can be either 24-bit YUV or 24-bit RGB.

3.3 RGB624DB Double Buffer Operation

The RGB624DB has a “double buffer” mode of operation. The mode is valid only when the pixel format is set for 15/16 BPP (RGB only, not YUV), and is described further in Section 5.7 on page 16.

4.0 VGA Port

VGA uses 8 bits per pixel. Bit order is high-to-low; the MSB is ‘7’ and the LSB is ‘0’. When the VGA port is selected only indirect mode is used. The 8 bits are masked with the Pixel Mask register and presented to the red, green, and blue palettes as indices into the 256 entries of each palette. The masked data is used as the same index into each of the three color palettes.

5.0 VRAM Pixel Formats

5.1 Bit Ordering

Bit order is high-to-low. For 8 BPP, the MSB is ‘7’ and the LSB is ‘0’; for 16 BPP the MSB is ‘15’ and the LSB is ‘0’, and so on.

When the VRAM pixel port is selected the default condition is to access the pixels from low to high. For each LCLK, the first pixel used is at the end with bit PIX[00], and the last pixel used is at the end with bit PIX[63] (bit PIX[31] for VRAM width = 32). For example, for 8 BPP, the first pixel is PIX[07:00], the second pixel is PIX[15:08], and so on.

4 BPP is a special case. Within a byte, the default condition is to select first the high nibble (e.g., PIX[07:04]), then the low nibble (PIX[03:00]). The SWAP NIB bit of the Miscellaneous Control 3 register may be used to swap the order the two nibbles are used. This swap is applied to every byte that is read in, and is only active, when set, for 4 BPP.

For 15/16 BPP, 24 BPP, and 32 BPP RGB formats, the Red and Blue components can be swapped with the SWAP RB bit of the Miscellaneous Control 3 register.

For YUV formats the incoming U and V order can be swapped using the SWAP UV bit of the Miscellaneous Control 3 register.

Independent of the VRAM pixel format, the two bytes within each pair of incoming bytes may be altered with the SWAP BYTE bits of the Miscellaneous Control 3 register. When these bits are set, PIX[15:08] are used as PIX[07:00] and PIX[07:00] are used as PIX[15:08], PIX[31:24] are used as PIX[23:16] and PIX[23:16] are used as PIX[31:24], and so on.

5.2 Pixel Format Tables

Tables 6 through 10 show the bit assignments of the pixel data port for each supported pixel format. Prefixes A - P identify individual pixels, and numbers 0 - 7 identify the bit within the pixel.

The pixel formats given in Table 6 are exactly compatible with the RGB51x and RGB52x products. For 4 bit pixels, this information is the data seen by the three color palettes. For 8 bit pixels, it is the data seen by the three color palettes in indirect color mode, and it is the data seen by the three DACs in direct color mode. The suffixes (blu, grn, red) identify the data seen by each of the color palettes (indirect mode) or each of the DACs (direct mode) for 16, 24, and 32 bit RGB pixels.

The YUV and overlay formats are new with the RGB6xx products; these new formats are given in Tables 7 and 8.

Tables 9 and 10 are the same as tables 7 and 8 except they shows the effect of setting the SWAP UV bit.

Table 11 shows the bit assignments for the double buffer formats that are unique to the RGB624DB.

The effects of setting the SWAP BYTE bits of the Miscellaneous Control 3 register are not shown in Tables 6 through 10.

5.3 Register Control Bits

As mentioned previously the Pixel Format register is used to select the expected pixel size within the frame buffer (4, 8, 16, 24 packed, or 32 bits per pixel). Associated with each format is an individual control register (4 BPP Control, 8 BPP Control, etc.)

For the larger pixel sizes control bits in these registers may select between RGB and YUV. Otherwise, these control bits tend to be for use only with the RGB formats. Some control bits are replicated. E.g. the 16 BPP Control register has dynamic bypass control bits and the 32 BPP Control register has dynamic bypass control bits. An exception to this is the SWAP RB bit in the Miscellaneous Control 3 register; it globally affects the red/blue order for the 15/16, 24 packed, and 32 bit RGB formats.

The controls for the YUV formats tend to be in other registers that act globally on all YUV formats. These are the four constants registers for controlling the YUV to RGB conversion, and the SWAP UV bit in the Miscellaneous Control 3 register for swapping U and V in the incoming pixels.

5.4 4 BPP

The 4 BPP format is the same as used on the RGB51x and RGB52x products.

With 4 BPP format 8 pixels (32 bit VRAM width) or 16 pixels (64 bit VRAM width) are obtained for each pixel port data access. As noted above the default access of the two pixels within each byte are high-to-low:

PIX[7:4] = pixel one

PIX[3:0] = pixel two, but this can be reversed with the SWAP NIB bit of the Miscellaneous Control 3 register.

4 BPP is only used in indirect color mode. The 4 bits are masked with the 4 low order bits [3:0] of the Pixel Mask. The resultant masked 4 bits are then used to index into each of the red, green, and blue palettes.

With 4 BPP the 256 entry palettes are divided into 16 partitions of 16 entries per partition. The upper 4 bits of the Pixel Mask register are ignored. The PARTITION bits of the Palette Control register are used as the upper 4 bits of the palette address to select the desired partition. The 4 masked pixel bits are used to index to 1-of-16 entries within the selected partition.

5.5 8 BPP

The 8 BPP format is the same as used on the RGB51x and RGB52x products.

With 8 BPP format 4 pixels (32 bit VRAM width) or 8 pixels (64 bit VRAM width) are obtained for each pixel port data access.

8 BPP can be indirect or direct, under control of the B8 DCOL bit of the 8 BPP Control register. If indirect, the 8 bits are masked with the Pixel Mask register and presented to the red, green, and blue palettes as indices into the 256 entries of each palette.

If direct, the 8 bits are presented to the red, green, and blue DACs. Note that since the red, green, and blue colors are identical the displayed image will be monochrome.

5.6 16 BPP

With 15 BPP or 16 BPP format 2 pixels (32-bit VRAM width) or 4 pixels (64-bit VRAM width) are obtained for each pixel port data access. The 15 or 16 bits are expanded to 24 bits, under control of the 16 BPP Control register.

A variety of formats and options are supported with 16 BPP. Two bits in the 16 BPP Control register, B16 MODE, select one of four pixel formats:

00	15/16-bit RGB
01	Tagged RGB/YUV 8 + 7
10	16-bit YUV
11	Tagged RGB/YUV 7 + 8

The definition of the remaining bits in the 16 BPP Control register depend on the value of the B16 MODE bits. (E.g., when B16 MODE = 00, bit 5 of the 16 BPP Control register is defined as the “BY16 POL” bit, for use with 16 BPP dynamic bypass. But if B16 MODE = 01, then bit 5 is defined as “TAG POL”, for use with the tagged RGB/YUV format.)

5.6.1 15/16-bit RGB

This format is selected with B16 MODE = 00, and is the same as used on the RGB51x and RGB52x products. The 16 BPP Control register provides these options with this format:

1. The incoming pixel can be 15 bits (555 format) or 16 bits (565 format).
2. The color path can be indirect (through the palettes) or direct (bypassing the palettes). Also, with 555 format, the 16th bit can be used to dynamically switch on a pixel-by-pixel basis between indirect and direct color.
3. If indirect color is selected, the addressing of the palettes can be “sparse” (pixel bits used as high order palette address bits) or “contiguous” (pixel bits used as low order palette address bits).
4. If indirect color with contiguous addressing is selected, the palettes can be divided into partitions. The PARTITION bits of the Palette Control register are used to select the partition by filling in the upper palette address bits. With 555 format 8 partitions are available; with 565 format there are 4 partitions.
5. If direct color is used the pixel bits are sent to the DAC high order bits. The low order bits can be zero filled, or the low order bits can be filled with the high order bits of the pixel data. (See description of ZIB/LIN bit below.)

If dynamic bypass is selected the following conditions will apply:

1. The format will be forced to 15 bit (555), with the unused 16th bit now used to control indirect/direct color selection.
2. The indirect color path will be forced to use sparse addressing of the palettes. Partitions cannot be used.
3. The direct color path will force the low order bits to the DACs to be zero filled (ZIB). LIN format cannot be used.
4. The Pixel Mask will mask the pixel data regardless of whether or not the palette is bypassed.

5.6.1.1 555/565 Formats

The 555/565 bit determines if the pixel is 15 bits (5:5:5 format) or 16 bits (5:6:5 format). The format designator, 5:5:5 or 5:6:5, refers to the bit allocations, high-to-low, for red:green:blue.

With 15 BPP the high order bit of each two bytes (PIX[15], PIX[31], PIX[47], PIX[63]) is discarded unless dynamic bypass is specified (B16 DCOL bits = 01). With dynamic bypass, this bit is used for indirect/direct color selection.

As noted above setting the mode to dynamic bypass will force the format to 555 regardless of the setting of the 555/565 bit.

5.6.1.2 Color Path Selection

The B16 DCOL bits are used to select one of:

1. Indirect color always (00).
2. Direct color always (11).
3. Dynamic selection of indirect or direct color (01).

The expansion to 24 bits varies depending on whether the color path is indirect or direct.

Indirect Color: The palette addressing can be sparse or contiguous and is controlled by the SPR/CNT bit. With sparse addressing the pixels will address 32 locations each for the red and blue palettes, and 32 locations for green in 555 format or 64 locations for green in 565 format. With the lower address bits set to zeroes the locations accessed will be “scattered” through the palettes, with the intermediate locations unused.

With contiguous addressing the PARTITION bits of the Palette Control register are used for the high order palette address bits, and the access within each palette is contiguous. For 555 format there are 8 partitions and 32 entries within each partition. For 565 format there are 4 partitions. All 64 entries in the green palette are addressed. Only the lower 32 entries of the red and blue palettes are used; the high 32 entries are not used.

For sparse addressing the low order bits are dependent on the ZIB/LIN bit. This bit *must* be set to 0 (ZIB). This will force the low order bits to zeros. If ZIB/LIN is 1 (LIN) then the values of the low order bits presented to the palettes are undefined.

For sparse addressing the low order Pixel Mask bits have no effect.

For contiguous addressing the high order bits are always supplied by the PARTITION bits and the high order Pixel Mask bits have no effect.

As noted above for dynamic bypass mode the format is forced to 555 mode and addressing is forced to be sparse regardless of the setting of the SPR/CNT bit.

Direct Color: To expand the 5 or 6 bits of color from the pixel data to 8 bits, the ZIB/LIN bit of the 16 BPP Control register specifies the generation of the low order 3 or 2 bits. If ZIB (Zero Intensity Black), the low order bits are made 0. If LIN (Linear), the low order bits are made equal to the high order bits. This causes the 5 or 6 bits to expand to 8 bits in a linear fashion, with both zero scale and full scale values used. With Zero Intensity Black, full scale cannot be achieved.

As noted above for dynamic bypass mode the format is forced to 555 mode and the low order fill is forced as ZIB, regardless of the setting of the ZIB/LIN bit.

5.6.1.3 Dynamic Bypass

As described above the selection of “dynamic bypass” mode forces the 555 format and uses the high order bit of the incoming 16-bit pixels as a control bit to select, on a pixel-by-pixel basis, the indirect (color lookup) or direct (lookup bypass) path.

The meaning of this bit depends on the BY16 bit in the 16 BPP Control register. When BY16 = 0 the incoming control bit forces the bypass. That is, when the control bit is 1 the palette is bypassed (direct color), and when 0 the palette is not bypassed (indirect color).

When BY16 = 1 the meaning of the incoming control bit is reversed; it now forces a lookup. That is, when the control bit is 1 the palette is used (color lookup), but when 0 the palette is bypassed (direct color).

5.6.2 Tagged RGB/YUV

This format is selected with B16 MODE = 01 or B16 MODE = 11. The high order bit of the incoming pixel data (PIX[15], PIX[31], PIX[47], PIX[63]) is used as a tag to interpret the remaining 15 bits as either an RGB or a YUV pixel.

If RGB, the pixel is 555 RGB with similar options as for the 16 BPP 555 format.

If YUV, the format is sub-sampled, with 4 variations available for the 15-bit YUV pixel. B16 MODE determines if the format is 8 high bits followed by 7 low bits (B16 MODE = 01) or the format is 7 high bits followed by 8 low bits (B16 MODE = 11).

For both B16 MODE = 01 and B16 MODE = 11, the SWAP YC bit determines which bits, high or low, are Y (luminance) or C (Chrominance, i.e. alternating U and V values).

The assignment of incoming pixel data PIX[63:00] to the Y, U and V values for the various settings is given in Tables 7 and 9. The following table summarizes the pixel values for the first pixel. (PIX[15] is the tag, PIX[14:00] are the YUV data):

Table 5. YUV Settings for Tagged RGB/YUV

B16 MODE	SWAP YC	High Bits	Low Bits
01	0	PIX[14:07] = Y	PIX[06:00] = U/V
01	1	PIX[14:07] = U/V	PIX[06:00] = Y
11	0	PIX[14:08] = Y	PIX[07:00] = U/V
11	1	PIX[14:08] = U/V	PIX[07:00] = Y

For the incoming values of Y or U and V that 7 bits, internally these values have a ‘0’ appended as the rightmost (least significant) bit before presentation to the color space converter, to pad the values to 8 bits.

Compared to the 16-bit YUV format, the 15-bit YUV format for B16 MODE = 01 is as if the entire 16-bit value is shifted to the right by 1 bit, with the rightmost bit discarded (and used as ‘0’ internally). For B16 MODE = 11, it is as if the high byte of the 16-bit YUV value is shifted right by 1 bit with the rightmost bit discarded (and used as ‘0’ internally), with the low byte preserved as is.

5.6.2.1 Tag Usage

The TAG POL bit (tag polarity) sets the meaning of the incoming tag bit. When TAG POL is 0, a tag of ‘0’ indicates the pixel is RGB and a tag of ‘1’ indicates the pixel is YUV.

When TAG POL is 1, a tag of ‘0’ indicates the pixel is YUV and a tag of ‘1’ indicates the pixel is RGB.

5.6.2.2 RGB Color Path Selection

When the tag bit selects the RGB format the color path can be indirect or direct, with similar options as for 16 BPP RGB. However, dynamic bypass is not available.

As with 16 BPP RGB, the B16 DCOL bits are used to select the color path:

1. Indirect color (00).
2. Direct color (11).

The expansion to 24 bits varies depending on whether the color path is indirect or direct.

Indirect Color: The palette addressing can be sparse or contiguous and is controlled by the SPR/CNT bit. With sparse addressing the pixels will address 32 locations each for the red, green and blue palettes. With the lower address bits set to zeroes the locations accessed will be "scattered" through the palettes, with the intermediate locations unused.

With contiguous addressing the PARTITION bits of the Palette Control register are used for the high order palette address bits, and the access within each palette is contiguous. There are 8 partitions and 32 entries within each partition.

For sparse addressing the low order bits are dependent on the ZIB/LIN bit. This bit *must* be set to 0 (ZIB). This will force the low order bits to zeros. If ZIB/LIN is 1 (LIN) then the values of the low order bits presented to the palettes are undefined.

For sparse addressing the low order Pixel Mask bits have no effect.

For contiguous addressing the high order bits are always supplied by the PARTITION bits and the high order Pixel Mask bits have no effect.

Direct Color: To expand the 5 bits of color from the pixel data to 8 bits, the ZIB/LIN bit of the 16 BPP Control register specifies the generation of the low order 3 bits. If ZIB (Zero Intensity Black), the low order bits are made 0. If LIN (Linear), the low order bits are made equal to the high order bits. This causes the 5 bits to expand to 8 bits in a linear fashion, with both zero scale and full scale values used. With Zero Intensity Black, full scale cannot be achieved.

5.6.3 16-bit YUV

This format is selected with B16 MODE = 10. The pixels are YUV with sub-sampled chrominance. The SWAP YC bit determines the luminance/chrominance order. When SWAP YC = 0 the high order 8 bits contain the Y value and the low order 8 bits contain the alternating U and V values. When SWAP YC = 1 the high order 8 bits contain the alternating U and V values and the low order 8 bits contain the Y value.

If key matching is enabled (KEY ENAB = 1 in the Key Control register), then the 16-bit pixel has two formats:

1. Overlay - all 16 bits are 4:2:2 YUV, as described above.
2. Underlay - the low or high 8 bits (whichever is not the Y value) are used as 8-bit RGB (the same as 8 BPP). In tables 7 and 9, these pixels are designated as CLUT8 (Color Lookup Table, 8 bits).

When key matching is enabled 255 Y values are available for display of the overlay. The remaining 256th value is matched against the contents of the (masked) Key register. When there is a match the pixel selection logic discards the Y value and uses the remaining 8 bits as an 8-bit RGB underlay pixel.

When key matching is enabled the 8 BPP underlay pixel may be used as indirect (color lookup table) or direct color (the 8 bits are presented directly to the red, green and blue DACs, which displays as monochrome), using the B8 DCOL bit. If key matching is not enabled, the B8 DCOL bit has no effect.

5.7 16 BPP Double Buffered (RGB624DB Only)

For the RGB624DB only, a double buffer mode of operation can be set for use with 15-bit RGB pixels. Double buffer operation is controlled by two register bits:

- ❑ The DBUF MODE (double buffer mode enable) bit. This is bit 0 in the Miscellaneous Control 3 register (index 0x0072).
- ❑ The SUPP BUFB (suppress buffer B) bit. This is bit 0 of the Key Control/DB Operation register (index 0x0078).

When the DBUF MODE bit is set, the VRAM inputs are treated as coming from two frame buffers, A and B.

Instead of processing 16 bits for each pixel, 32 bits from the VRAM inputs are used for each pixel. Each group of 32 input bits is considered to be:

- ❑ 15-bit buffer A pixel data (PIX[14:0], PIX[46:32])
- ❑ An unused bit (PIX[15], PIX[47])
- ❑ 15-bit buffer B pixel data (PIX[30:16], PIX[62:48])
- ❑ A Buffer Select bit (PIX[31], PIX[63])

The Buffer Select bit selects either buffer A data (Buffer Select bit = ‘0’) or buffer B data (Buffer Select bit = ‘1’) for display. Unselected buffer data is ignored. Thus, the A/B buffer selection is made on a pixel-by-pixel basis.

When the SUPP BUFB bit is set, the Buffer Select bit is ignored, and only buffer A data is displayed. The SUPP BUFB bit can be used for A/B buffer selection on a per-frame basis by setting the Buffer Select bits to ‘1’s for all pixels in any double buffered region [window] of the screen. For these pixels, if SUPP BUFB is set, then buffer A is selected, and if SUPP BUFB is reset, then buffer B is selected.

Note that bit 0 of the Key Control/DB Operation register has a dual usage. When 16 BPP double buffer operation is enabled bit 0 of the Key Control register is used as the SUPP BUFB bit as described above. For all other modes of operation this bit acts as the KEY ENAB (key enable) bit, for key matching operation.

Several restrictions apply to double buffer operation:

1. When the VRAM port is selected, the DBUF MODE bit should only be set when the Pixel Format register (index 0x000a) is set to 15/16 BPP (Format bits = ‘100’.)
2. In the 16 Bit Pixel Control register (index 0x000c) the B16 MODE bits (bits 4 and 3) must be set to ‘00’ for RGB operation only.
3. Also in the 16 Bit Pixel Control register the 555/565 bit (bit 1) must be set to ‘0’ for 555 operation.
4. Also in the 16 Bit Pixel Control register, the B16 DCOL (direct color control) bits (bits 7 and 6) must be set to either ‘00’ (Indirect Color) or ‘11’ (Direct Color).

If any of these four conditions is violated the resultant operation of the product is indeterminate.

As implied above, the 15 bits selected for display, either A or B, are interpreted as 555 format. As with normal 16 BPP, 555 format, the color path can be indirect (through the palettes) or direct, as set by the B16 DCOL bits. However, the “dynamic bypass” mode (B16 DCOL = ‘01’) is not available. Also, operation with YUV pixels is not available.

The ZIB/LIN (bit fill selection) and SPR/CNT (palette addressing) control bits operate on the double buffer pixels the same as normal 555 pixels.

The actions of the B16 DCOL, ZIB/LIN and SPR/CNT control bits apply to all pixel data, whether from buffer A or buffer B.

As noted in section 2.4.1, when double buffer operation is enabled the outgoing SCLK is generated with the same timings as for 32 BPP.

5.8 24 BPP

24 BPP Packed can only be selected when the VRAM width is 64 bits. If 24 BPP Packed format is selected with the Pixel Format register, but the VRAM SIZE bit in the Miscellaneous Control 1 register is set for 32 bits, then the product operation is undefined.

With 24 BPP Packed format the pixels can be either all RGB or all YUV. Each 64-bit pixel port data access contains 2+2/3 pixels. Every 3 consecutive pixel port data accesses (3 × 8 = 24 bytes) contains 8 pixels of 3 bytes each. The assignment of the bytes for each of the three accesses is shown in Figure 3 on page 17 for RGB pixels, and in Figure 4 for YUV pixels.

5.8.1 24-bit RGB

RGB pixels are selected when the RGB/YUV bit is set to 0 in the 24 Bit Packed Pixel Control Register. This format is the same as used on the RGB51x and RGB52x products. Each byte contains 8 bits of red, green, or blue color. Color access can be indirect or direct, and is selected with the B24P DCOL bit of the 24 BPP Packed Control register.

For indirect color, the 8 bits of red, green, and blue are each masked by the Pixel mask, and then presented to the red, green, and blue palettes as indices into the 256 entries of each palette.

For direct color, the 8 bits of red, green, and blue are presented to the DACs.

5.8.2 24-bit YUV

YUV pixels are selected when the RGB/YUV bit is set to 1. There are no other options that can be selected with the 24 Bit Packed Control register.

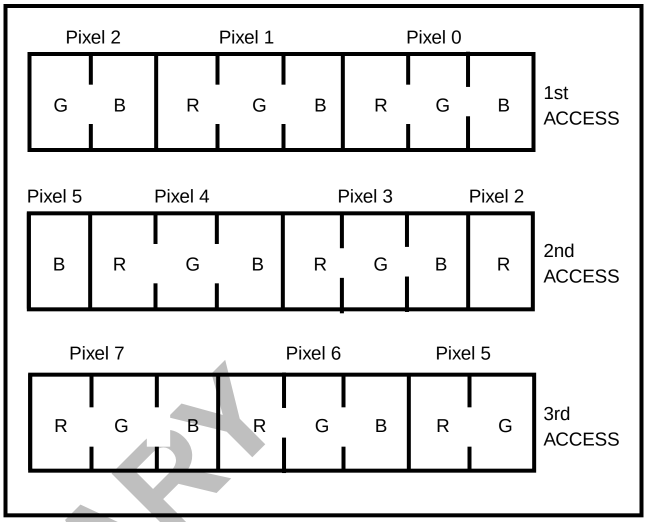


Figure 3. RGB - 24 BPP Packed Pixel Input from VRAM

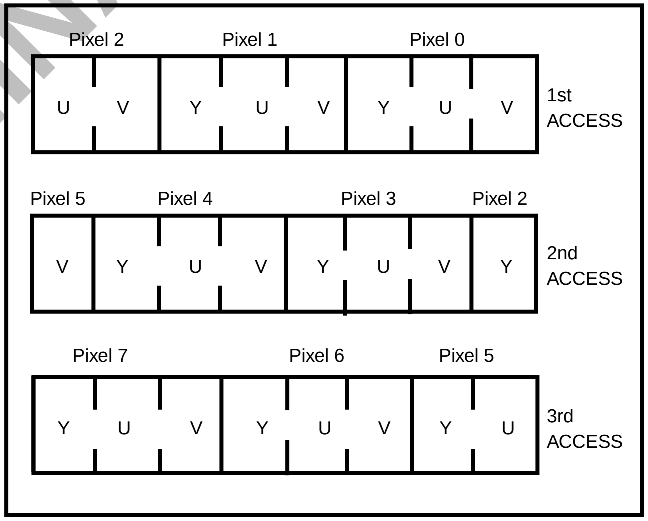


Figure 4. YUV - 24 BPP Packed Pixel Input from VRAM

5.9 32 BPP

With 32 BPP format 1 pixel (32-bit VRAM width) or 2 pixels (64-bit VRAM width) are obtained for each pixel port data access. Within the 32 bits there are four formats:

1. High order byte unused, three low order bytes are RGB.
2. High order byte unused, three low order bytes are YUV.
3. High order byte is 8-bit RGB, three low order bytes are RGB.
4. High order byte is 8-bit RGB, three low order bytes are YUV.

Options 1 and 2 are when key matching is off (KEY ENAB = 0 in the Key Control register); options 3 and 4 are when key matching is on.

The RGB/YUV bit in the 32 BPP Control register determines if the three low order bytes will be used as RGB (options 1 and 3) or YUV (options 2 and 4). When RGB is selected the three bytes are 8 bits each of red, green and blue. When YUV is selected the three bytes are 8 bits each of Y, U and V.

5.9.1 Underlay Options

If YUV format is specified for the underlay (three low order bytes) there are no other options that can be specified with the 32 BPP Control register.

If RGB is selected, the color path can be selected with the B32 DCOL bits:

1. Indirect color (00).
2. Direct color (11).

In addition, if key matching is *not* selected, then a dynamic selection of indirect color or direct color on a pixel-by-pixel basis can be specified by setting the B32 DCOL bits to 01.

With dynamic selection (dynamic bypass), the “25th” bit is used as the indirect/direct control bit (PIX[24], PIX[56]) and the remaining bits of the high order byte are unused. (PIX[31:25] and PIX[63:57].) The pixel data in this mode is masked by the Pixel Mask regardless of whether or not the palette is bypassed.

For indirect color, the 8 bits of red, green, and blue are each masked by the Pixel mask, and then presented to the red, green, and blue palettes as indices into the 256 entries of each palette.

For direct color, the 8 bits of red, green, and blue are presented to the DACs.

5.9.1.1 Dynamic Bypass

As described above the selection of “dynamic bypass” mode uses the “25th” bit of the incoming 32-bit pixels as a control bit to select, on a pixel-by-pixel basis, the indirect (color lookup) or direct (lookup bypass) path.

The meaning of this bit depends on the BY32 bit in the 32 BPP Control register. When BY32 = 0 the incoming control bit forces the bypass. That is, when the control bit is 1 the palette is bypassed (direct color), and when 0 the palette is not bypassed (indirect color).

When BY32 = 1 the meaning of the incoming control bit is reversed; it now forces a lookup. That is, when the control bit is 1 the palette is used (color lookup), but when 0 the palette is bypassed (direct color).

5.9.2 Overlay Options

When key matching is enabled the 8 high order bits are used as the overlay pixel.

If the incoming 8-bit pixel value matches the (masked) value in the Key register the pixel selection logic will switch to the underlay layer, the high order byte will be discarded, and the three low order bytes will be used as the displayed pixel using the options described above.

If the value does not match the masked key, the high order byte will be displayed as the overlay pixel, as one of 255 values. The pixel path can be indirect or direct, under control of the B8 DCOL bit in the 32 BPP Control register.

If indirect, the 8 bits are masked with the Pixel Mask register and presented to the red, green, and blue palettes as indices into the 256 entries of each palette.

If direct, the 8 bits are presented to the red, green, and blue DACs. Note that since the red, green, and blue colors are identical the displayed image will be monochrome.

Also note that when the overlay path is indirect and the underlay layer is RGB, the underlay probably should not use indirect color since both pixel types, 8-bit overlay and 24-bit underlay, would be trying to share the same lookup table.

6.0 6 Bit Linear Palette Output

The 6BIT LIN (6 bit linear) bit of the Palette Control register affects the format of RGB color data read from the palettes and presented to the DACs in indirect color mode. It only has effect when the color resolution is set to 6 bits with the COL RES bit of the Miscellaneous Control 2 register and DCOL CNTL is set to indirect color.

If the palettes contain data with the two low order bits set to 00 (which will be the case when the palettes are loaded with COL RES set to 6 bits), then without special processing the data values presented to the DACs will range from 0x00 through 0xfd. The maximum output of the DACs will be approximately 1.5% less than full scale (0xff). This will occur when 6BIT LIN is set to 1.

When 6BIT LIN is set to 0 (the default), then the outputs of the palettes will be modified to allow the DACs to reach full scale output. The modification consists of discarding the two low order bits from the palettes, and substituting the two high order bits for the two low order bits presented to the DACs. (i.e., the palette bits presented to a DAC will be bits 7 6 5 4 3 2 7 6).

With this bit substitution there will be a “linear” mapping of the palette data range (0x00 – 0xfd) to the DAC data range (0x00 – 0xff), and the DACs will operate over their full range.

If COL RES = 1 (8-bit color resolution) the palette outputs are presented to the DACs unchanged, and 6BIT LIN has no effect. The DACs will operate over the 8-bit range from completely off to full scale on.

Palette linear output is intended for emulation of the VGA 6-bit DACs in which the palette is loaded with 6-bit colors in the 6 high-order bits by setting COL RES to 6-bits. However, regardless of how the palette was loaded or what the pixel format is (VGA, 4, 8, 15/16, 24, 32 BPP), if enabled (DCOL = indirect, COL RES = 6 bit, 6BIT LIN = 0) the palette outputs will be affected as discussed above.

In summary, with the default conditions for VGA mode (indirect color, 6-bit color resolution, 6BIT LIN = 0), there will be a linear mapping of the 6-bit VGA palette data to the DACs, and the DACs will operate over their full range. The mapping can be turned off by setting 6BIT LIN to 1, in which case the 8 bits from the palettes are presented to the DACs unmodified. With 00 in the two low order bits of the palettes the DACs will not reach full scale output.

With 8-bit color resolution (indirect color), or with direct color, the setting of 6BIT LIN has no effect.

Table 6. Pixel Format Table - RGB51x, RGB52x Formats

Pixel Port Bit	4 BPP[1]		8 BPP	15/16 BPP[2,3]				24 BPP Packed			32 BPP
	SWAP NIB=0	SWAP NIB=1		555 Sparse or Direct Color	555 CONTIG	565 Sparse or Direct Color	565 Contig	1st Access	2nd Access	3rd Access	
0	B0	A0	A0	A3BLU	A0BLU	A3BLU	A0BLU	A0BLU	C0RED	F0GRN	A0BLU
1	B1	A1	A1	A4BLU	A1BLU	A4BLU	A1BLU	A1BLU	C1RED	F1GRN	A1BLU
2	B2	A2	A2	A5BLU	A2BLU	A5BLU	A2BLU	A2BLU	C2RED	F2GRN	A2BLU
3	B3	A3	A3	A6BLU	A3BLU	A6BLU	A3BLU	A3BLU	C3RED	F3GRN	A3BLU
4	A0	B0	A4	A7BLU	A4BLU	A7BLU	A4BLU	A4BLU	C4RED	F4GRN	A4BLU
5	A1	B1	A5	A3GRN	A0GRN	A2GRN	A0GRN	A5BLU	C5RED	F5GRN	A5BLU
6	A2	B2	A6	A4GRN	A1GRN	A3GRN	A1GRN	A6BLU	C6RED	F6GRN	A6BLU
7	A3	B3	A7	A5GRN	A2GRN	A4GRN	A2GRN	A7BLU	C7RED	F7GRN	A7BLU
8	D0	C0	B0	A6GRN	A3GRN	A5GRN	A3GRN	A0GRN	D0BLU	F0RED	A0GRN
9	D1	C1	B1	A7GRN	A4GRN	A6GRN	A4GRN	A1GRN	D1BLU	F1RED	A1GRN
10	D2	C2	B2	A3RED	A0RED	A7GRN	A5GRN	A2GRN	D2BLU	F2RED	A2GRN
11	D3	C3	B3	A4RED	A1RED	A3RED	A0RED	A3GRN	D3BLU	F3RED	A3GRN
12	C0	D0	B4	A5RED	A2RED	A4RED	A1RED	A4GRN	D4BLU	F4RED	A4GRN
13	C1	D1	B5	A6RED	A3RED	A5RED	A2RED	A5GRN	D5BLU	F5RED	A5GRN
14	C2	D2	B6	A7RED	A4RED	A6RED	A3RED	A6GRN	D6BLU	F6RED	A6GRN
15	C3	D3	B7	(NOTE 4)	UNUSED	A7RED	A4RED	A7GRN	D7BLU	F7RED	A7GRN
16	F0	E0	C0	B3BLU	B0BLU	B3BLU	B0BLU	A0RED	D0GRN	G0BLU	A0RED
17	F1	E1	C1	B4BLU	B1BLU	B4BLU	B1BLU	A1RED	D1GRN	G1BLU	A1RED
18	F2	E2	C2	B5BLU	B2BLU	B5BLU	B2BLU	A2RED	D2GRN	G2BLU	A2RED
19	F3	E3	C3	B6BLU	B3BLU	B6BLU	B3BLU	A3RED	D3GRN	G3BLU	A3RED
20	E0	F0	C4	B7BLU	B4BLU	B7BLU	B4BLU	A4RED	D4GRN	G4BLU	A4RED
21	E1	F1	C5	B3GRN	B0GRN	B2GRN	B0GRN	A5RED	D5GRN	G5BLU	A5RED
22	E2	F2	C6	B4GRN	B1GRN	B3GRN	B1GRN	A6RED	D6GRN	G6BLU	A6RED
23	E3	F3	C7	B5GRN	B2GRN	B4GRN	B2GRN	A7RED	D7GRN	G7BLU	A7RED
24	H0	G0	D0	B6GRN	B3GRN	B5GRN	B3GRN	B0BLU	D0RED	G0GRN	(NOTE 4)
25	H1	G1	D1	B7GRN	B4GRN	B6GRN	B4GRN	B1BLU	D1RED	G1GRN	UNUSED
26	H2	G2	D2	B3RED	B0RED	B7GRN	B5GRN	B2BLU	D2RED	G2GRN	UNUSED
27	H3	G3	D3	B4RED	B1RED	B3RED	B0RED	B3BLU	D3RED	G3GRN	UNUSED
28	G0	H0	D4	B5RED	B2RED	B4RED	B1RED	B4BLU	D4RED	G4GRN	UNUSED
29	G1	H1	D5	B6RED	B3RED	B5RED	B2RED	B5BLU	D5RED	G5GRN	UNUSED
30	G2	H2	D6	B7RED	B4RED	B6RED	B3RED	B6BLU	D6RED	G6GRN	UNUSED
31	G3	H3	D7	(NOTE 4)	UNUSED	B7RED	B4RED	B7BLU	D7RED	G7GRN	UNUSED
32	J0	I0	E0	C3BLU	C0RED	C3BLU	C0BLU	B0GRN	E0BLU	G0RED	B0BLU
33	J1	I1	E1	C4BLU	C1BLU	C4BLU	C1BLU	B1GRN	E1BLU	G1RED	B1BLU
34	J2	I2	E2	C5BLU	C2BLU	C5BLU	C2BLU	B2GRN	E2BLU	G2RED	B2BLU
35	J3	I3	E3	C6BLU	C3BLU	C6BLU	C3BLU	B3GRN	E3BLU	G3RED	B3BLU
36	I0	J0	E4	C7BLU	C4BLU	C7BLU	C4BLU	B4GRN	E4BLU	G4RED	B4BLU
37	I1	J1	E5	C3GRN	C0GRN	C2GRN	C0GRN	B5GRN	E5BLU	G5RED	B5BLU
38	I2	J2	E6	C4GRN	C1GRN	C3GRN	C1GRN	B6GRN	E6BLU	G6RED	B6BLU
39	I3	J3	E7	C5GRN	C2GRN	C4GRN	C2GRN	B7GRN	E7BLU	G7RED	B7BLU
40	L0	K0	F0	C6GRN	C3GRN	C5GRN	C3GRN	B0RED	E0GRN	H0BLU	B0GRN
41	L1	K1	F1	C7GRN	C4GRN	C6GRN	C4GRN	B1RED	E1GRN	H1BLU	B1GRN
42	L2	K2	F2	C3RED	C0RED	C7GRN	C5GRN	B2RED	E2GRN	H2BLU	B2GRN
43	L3	K3	F3	C4RED	C1RED	C3RED	C0RED	B3RED	E3GRN	H3BLU	B3GRN
44	K0	L0	F4	C5RED	C2RED	C4RED	C1RED	B4RED	E4GRN	H4BLU	B4GRN
45	K1	L1	F5	C6RED	C3RED	C5RED	C2RED	B5RED	E5GRN	H5BLU	B5GRN
46	K2	L2	F6	C7RED	C4RED	C6RED	C3RED	B6RED	E6GRN	H6BLU	B6GRN
47	K3	L3	F7	(NOTE 4)	UNUSED	C7RED	C4RED	B7RED	E7GRN	H7BLU	B7GRN
48	N0	M0	G0	D3BLU	D0BLU	D3BLU	D0BLU	C0BLU	E0RED	H0GRN	B0RED
49	N1	M1	G1	D4BLU	D1BLU	D4BLU	D1BLU	C1BLU	E1RED	H1GRN	B1RED
50	N2	M2	G2	D5BLU	D2BLU	D5BLU	D2BLU	C2BLU	E2RED	H2GRN	B2RED
51	N3	M3	G3	D6BLU	D3BLU	D6BLU	D3BLU	C3BLU	E3RED	H3GRN	B3RED
52	M0	N0	G4	D7BLU	D4BLU	D7BLU	D4BLU	C4BLU	E4RED	H4GRN	B4RED
53	M1	N1	G5	D3GRN	D0GRN	D2GRN	D0GRN	C5BLU	E5RED	H5GRN	B5RED
54	M2	N2	G6	D4GRN	D1GRN	D3GRN	D1GRN	C6BLU	E6RED	H6GRN	B6RED
55	M3	N3	G7	D5GRN	D2GRN	D4GRN	D2GRN	C7BLU	E7RED	H7GRN	B7RED
56	P0	O0	H0	D6GRN	D3GRN	D5GRN	D3GRN	C0GRN	F0BLU	H0RED	(NOTE 4)
57	P1	O1	H1	D7GRN	D4GRN	D6GRN	D4GRN	C1GRN	F1BLU	H1RED	UNUSED
58	P2	O2	H2	D3RED	D0RED	D7GRN	D5GRN	C2GRN	F2BLU	H2RED	UNUSED
59	P3	O3	H3	D4RED	D1RED	D3RED	D0RED	C3GRN	F3BLU	H3RED	UNUSED
60	O0	P0	H4	D5RED	D2RED	D4RED	D1RED	C4GRN	F4BLU	H4RED	UNUSED
61	O1	P1	H5	D6RED	D3RED	D5RED	D2RED	C5GRN	F5BLU	H5RED	UNUSED
62	O2	P2	H6	D7RED	D4RED	D6RED	D3RED	C6GRN	F6BLU	H6RED	UNUSED
63	O3	P3	H7	(NOTE 4)	UNUSED	D7RED	D4RED	C7GRN	F7BLU	H7RED	UNUSED

Note 1: In 4 BPP mode the 4 most significant bits of each pixel come from the partition bits of the palette control register.

Note 2: For 15/16 BPP Direct Color the low order bits for each color component are determined by the ZIB/LIN Bit of the 16 BPP Control register. For 15/16 BPP sparse format (indirect color), the ZIB/LIN bit must be set to ZIB, and the low order bits for each color component will be zeroes.

Note 3: In CONTIGUOUS format for 15/16 BPP the most significant bits of each pixel come from the partition bits of the palette control register.

Note 4: These bits are used for DYNAMIC BYPASS when that mode is enabled, otherwise they are unused.

Table 7. Pixel Format Table - 15/16 BPP YUV and Overlays

Pixel Port Bit	Tagged[2,3]						YUV		YUV with Luma Key[4]			
	RGB 555		YUV				SWAP YC = 0	SWAP YC = 1	SWAP YC = 0		SWAP YC = 1	
	SPR	CNT	B16 MODE = 01		B16 MODE = 11				No Match	Match	No Match	Match
			SWAP YC = 0	SWAP YC = 1	SWAP YC = 0	SWAP YC = 1						
0	A3BLU	A0BLU	A1U	A1Y	A0U	A0Y	A0U	A0Y	A0U	A0CLUT8	A0Y	A0KEY
1	A4BLU	A1BLU	A2U	A2Y	A1U	A1Y	A1U	A1Y	A1U	A1CLUT8	A1Y	A1KEY
2	A5BLU	A2BLU	A3U	A3Y	A2U	A2Y	A2U	A2Y	A2U	A2CLUT8	A2Y	A2KEY
3	A6BLU	A3BLU	A4U	A4Y	A3U	A3Y	A3U	A3Y	A3U	A3CLUT8	A3Y	A3KEY
4	A7BLU	A4BLU	A5U	A5Y	A4U	A4Y	A4U	A4Y	A4U	A4CLUT8	A4Y	A4KEY
5	A3GRN	A0GRN	A6U	A6Y	A5U	A5Y	A5U	A5Y	A5U	A5CLUT8	A5Y	A5KEY
6	A4GRN	A1GRN	A7U	A7Y	A6U	A6Y	A6U	A6Y	A6U	A6CLUT8	A6Y	A6KEY
7	A5GRN	A2GRN	A0Y	A0U	A7U	A7Y	A7U	A7Y	A7U	A7CLUT8	A7Y	A7KEY
8	A6GRN	A3GRN	A1Y	A1U	A1Y	A1U	A0Y	A0U	A0Y	A0KEY	A0U	A0CLUT8
9	A7GRN	A4GRN	A2Y	A2U	A2Y	A2U	A1Y	A1U	A1Y	A1KEY	A1U	A1CLUT8
10	A3RED	A0RED	A3Y	A3U	A3Y	A3U	A2Y	A2U	A2Y	A2KEY	A2U	A2CLUT8
11	A4RED	A1RED	A4Y	A4U	A4Y	A4U	A3Y	A3U	A3Y	A3KEY	A3U	A3CLUT8
12	A5RED	A2RED	A5Y	A5U	A5Y	A5U	A4Y	A4U	A4Y	A4KEY	A4U	A4CLUT8
13	A6RED	A3RED	A6Y	A6U	A6Y	A6U	A5Y	A5U	A5Y	A5KEY	A5U	A5CLUT8
14	A7RED	A4RED	A7Y	A7U	A7Y	A7U	A6Y	A6U	A6Y	A6KEY	A6U	A6CLUT8
15	ATAGNOT		ATAG				A7Y	A7U	A7Y	A7KEY	A7U	A7CLUT8
16	B3BLU	B0BLU	B1V	B1Y	B0V	B0Y	B0V	B0Y	B0V	B0CLUT8	B0Y	B0KEY
17	B4BLU	B1BLU	B2V	B2Y	B1V	B1Y	B1V	B1Y	B1V	B1CLUT8	B1Y	B1KEY
18	B5BLU	B2BLU	B3V	B3Y	B2V	B2Y	B2V	B2Y	B2V	B2CLUT8	B2Y	B2KEY
19	B6BLU	B3BLU	B4V	B4Y	B3V	B3Y	B3V	B3Y	B3V	B3CLUT8	B3Y	B3KEY
20	B7BLU	B4BLU	B5V	B5Y	B4V	B4Y	B4V	B4Y	B4V	B4CLUT8	B4Y	B4KEY
21	B3GRN	B0GRN	B6V	B6Y	B5V	B5Y	B5V	B5Y	B5V	B5CLUT8	B5Y	B5KEY
22	B4GRN	B1GRN	B7V	B7Y	B6V	B6Y	B6V	B6Y	B6V	B6CLUT8	B6Y	B6KEY
23	B5GRN	B2GRN	B0Y	B0V	B7V	B7Y	B7V	B7Y	B7V	B7CLUT8	B7Y	B7KEY
24	B6GRN	B3GRN	B1Y	B1V	B1Y	B1V	B0Y	B0V	B0Y	B0KEY	B0V	B0CLUT8
25	B7GRN	B4GRN	B2Y	B2V	B2Y	B2V	B1Y	B1V	B1Y	B1KEY	B1V	B1CLUT8
26	B3RED	B0RED	B3Y	B3V	B3Y	B3V	B2Y	B2V	B2Y	B2KEY	B2V	B2CLUT8
27	B4RED	B1RED	B4Y	B4V	B4Y	B4V	B3Y	B3V	B3Y	B3KEY	B3V	B3CLUT8
28	B5RED	B2RED	B5Y	B5V	B5Y	B5V	B4Y	B4V	B4Y	B4KEY	B4V	B4CLUT8
29	B6RED	B3RED	B6Y	B6V	B6Y	B6V	B5Y	B5V	B5Y	B5KEY	B5V	B5CLUT8
30	B7RED	B4RED	B7Y	B7V	B7Y	B7V	B6Y	B6V	B6Y	B6KEY	B6V	B6CLUT8
31	BTAGNOT		BTAG				B7Y	B7V	B7Y	B7KEY	B7V	B7CLUT8
32	C3BLU	C0RED	C1U	C1Y	C0U	C0Y	C0U	C0Y	C0U	C0CLUT8	C0Y	C0KEY
33	C4BLU	C1BLU	C2U	C2Y	C1U	C1Y	C1U	C1Y	C1U	C1CLUT8	C1Y	C1KEY
34	C5BLU	C2BLU	C3U	C3Y	C2U	C2Y	C2U	C2Y	C2U	C2CLUT8	C2Y	C2KEY
35	C6BLU	C3BLU	C4U	C4Y	C3U	C3Y	C3U	C3Y	C3U	C3CLUT8	C3Y	C3KEY
36	C7BLU	C4BLU	C5U	C5Y	C4U	C4Y	C4U	C4Y	C4U	C4CLUT8	C4Y	C4KEY
37	C3GRN	C0GRN	C6U	C6Y	C5U	C5Y	C5U	C5Y	C5U	C5CLUT8	C5Y	C5KEY
38	C4GRN	C1GRN	C7U	C7Y	C6U	C6Y	C6U	C6Y	C6U	C6CLUT8	C6Y	C6KEY
39	C5GRN	C2GRN	C0Y	C0U	C7U	C7Y	C7U	C7Y	C7U	C7CLUT8	C7Y	C7KEY
40	C6GRN	C3GRN	C1Y	C1U	C1Y	C1U	C0Y	C0U	C0Y	C0KEY	C0U	C0CLUT8
41	C7GRN	C4GRN	C2Y	C2U	C2Y	C2U	C1Y	C1U	C1Y	C1KEY	C1U	C1CLUT8
42	C3RED	C0RED	C3Y	C3U	C3Y	C3U	C2Y	C2U	C2Y	C2KEY	C2U	C2CLUT8
43	C4RED	C1RED	C4Y	C4U	C4Y	C4U	C3Y	C3U	C3Y	C3KEY	C3U	C3CLUT8
44	C5RED	C2RED	C5Y	C5U	C5Y	C5U	C4Y	C4U	C4Y	C4KEY	C4U	C4CLUT8
45	C6RED	C3RED	C6Y	C6U	C6Y	C6U	C5Y	C5U	C5Y	C5KEY	C5U	C5CLUT8
46	C7RED	C4RED	C7Y	C7U	C7Y	C7U	C6Y	C6U	C6Y	C6KEY	C6U	C6CLUT8
47	CTAGNOT		CTAG				C7Y	C7U	C7Y	C7KEY	C7U	C7CLUT8
48	D3BLU	D0BLU	D1V	D1Y	D0V	D0Y	D0V	D0Y	D0V	D0CLUT8	D0Y	D0KEY
49	D4BLU	D1BLU	D2V	D2Y	D1V	D1Y	D1V	D1Y	D1V	D1CLUT8	D1Y	D1KEY
50	D5BLU	D2BLU	D3V	D3Y	D2V	D2Y	D2V	D2Y	D2V	D2CLUT8	D2Y	D2KEY
51	D6BLU	D3BLU	D4V	D4Y	D3V	D3Y	D3V	D3Y	D3V	D3CLUT8	D3Y	D3KEY
52	D7BLU	D4BLU	D5V	D5Y	D4V	D4Y	D4V	D4Y	D4V	D4CLUT8	D4Y	D4KEY
53	D3GRN	D0GRN	D6V	D6Y	D5V	D5Y	D5V	D5Y	D5V	D5CLUT8	D5Y	D5KEY
54	D4GRN	D1GRN	D7V	D7Y	D6V	D6Y	D6V	D6Y	D6V	D6CLUT8	D6Y	D6KEY
55	D5GRN	D2GRN	D0Y	D0V	D7V	D7Y	D7V	D7Y	D7V	D7CLUT8	D7Y	D7KEY
56	D6GRN	D3GRN	D1Y	D1V	D1Y	D1V	D0Y	D0V	D0Y	D0KEY	D0V	D0CLUT8
57	D7GRN	D4GRN	D2Y	D2V	D2Y	D2V	D1Y	D1V	D1Y	D1KEY	D1V	D1CLUT8
58	D3RED	D0RED	D3Y	D3V	D3Y	D3V	D2Y	D2V	D2Y	D2KEY	D2V	D2CLUT8
59	D4RED	D1RED	D4Y	D4V	D4Y	D4V	D3Y	D3V	D3Y	D3KEY	D3V	D3CLUT8
60	D5RED	D2RED	D5Y	D5V	D5Y	D5V	D4Y	D4V	D4Y	D4KEY	D4V	D4CLUT8
61	D6RED	D3RED	D6Y	D6V	D6Y	D6V	D5Y	D5V	D5Y	D5KEY	D5V	D5CLUT8
62	D7RED	D4RED	D7Y	D7V	D7Y	D7V	D6Y	D6V	D6Y	D6KEY	D6V	D6CLUT8
63	DTAGNOT		DTAG				D7Y	D7V	D7Y	D7KEY	D7V	D7CLUT8

Note 1: YUV formats are for U,V order (Miscellaneous Control 3 register SWAP UV bit = 0). All YUV pixels go directly to the DACs.

Note 2: For Tagged RGB/YUV the TAG POL bit of the 16 BPP Control register sets the meaning of the tag bit. (For TAG POL = 0, incoming TAG = 0 selects RGB, TAG = 1 selects YUV. For TAG POL = 1, incoming TAG = 0 selects YUV, TAG = 1 selects RGB.)

Note 3: For Tagged RGB/YUV, when the tag selects RGB the Indirect/Direct, Sparse/Contiguous, and ZIB/LIN selections are the same as untagged 555 RGB. However, dynamic bypass is not available.

Note 4: For 16 bit YUV with luma key enabled, if the Y value does not match the luma key YUV is used. When the Y position matches the key, the remaining byte is used as 8 BPP indirect color (CLUT8 = Color Lookup Table, 8 BPP). The bytes used, high or low, for Y and CLUT8 are determined by the SWAP YC bit.

Table 8. Pixel Format Table - 24 BPP Packed, 32 BPP YUV and Overlays

Pixel Port Bit	24 BPP Packed YUV[1]			32 BPP[1]		
	1st Access	2nd Access	3rd Access	YUV	RGB OVL8[2]	YUV OVL8[2]
0	A0V	C0Y	F0U	A0V	A0BLU	A0V
1	A1V	C1Y	F1U	A1V	A1BLU	A1V
2	A2V	C2Y	F2U	A2V	A2BLU	A2V
3	A3V	C3Y	F3U	A3V	A3BLU	A3V
4	A4V	C4Y	F4U	A4V	A4BLU	A4V
5	A5V	C5Y	F5U	A5V	A5BLU	A5V
6	A6V	C6Y	F6U	A6V	A6BLU	A6V
7	A7V	C7Y	F7U	A7V	A7BLU	A7V
8	A0U	D0V	F0Y	A0U	A0GRN	A0U
9	A1U	D1V	F1Y	A1U	A1GRN	A1U
10	A2U	D2V	F2Y	A2U	A2GRN	A2U
11	A3U	D3V	F3Y	A3U	A3GRN	A3U
12	A4U	D4V	F4Y	A4U	A4GRN	A4U
13	A5U	D5V	F5Y	A5U	A5GRN	A5U
14	A6U	D6V	F6Y	A6U	A6GRN	A6U
15	A7U	D7V	F7Y	A7U	A7GRN	A7U
16	A0Y	D0U	G0V	A0Y	A0RED	A0Y
17	A1Y	D1U	G1V	A1Y	A1RED	A1Y
18	A2Y	D2U	G2V	A2Y	A2RED	A2Y
19	A3Y	D3U	G3V	A3Y	A3RED	A3Y
20	A4Y	D4U	G4V	A4Y	A4RED	A4Y
21	A5Y	D5U	G5V	A5Y	A5RED	A5Y
22	A6Y	D6U	G6V	A6Y	A6RED	A6Y
23	A7Y	D7U	G7V	A7Y	A7RED	A7Y
24	B0V	D0Y	G0U	UNUSED	A0OVL8	A0OVL8
25	B1V	D1Y	G1U	UNUSED	A1OVL8	A1OVL8
26	B2V	D2Y	G2U	UNUSED	A2OVL8	A2OVL8
27	B3V	D3Y	G3U	UNUSED	A3OVL8	A3OVL8
28	B4V	D4Y	G4U	UNUSED	A4OVL8	A4OVL8
29	B5V	D5Y	G5U	UNUSED	A5OVL8	A5OVL8
30	B6V	D6Y	G6U	UNUSED	A6OVL8	A6OVL8
31	B7V	D7Y	G7U	UNUSED	A7OVL8	A7OVL8
32	B0U	E0V	G0Y	B0V	B0BLU	B0V
33	B1U	E1V	G1Y	B1V	B1BLU	B1V
34	B2U	E2V	G2Y	B2V	B2BLU	B2V
35	B3U	E3V	G3Y	B3V	B3BLU	B3V
36	B4U	E4V	G4Y	B4V	B4BLU	B4V
37	B5U	E5V	G5Y	B5V	B5BLU	B5V
38	B6U	E6V	G6Y	B6V	B6BLU	B6V
39	B7U	E7V	G7Y	B7V	B7BLU	B7V
40	B0Y	E0U	H0V	B0U	B0GRN	B0U
41	B1Y	E1U	H1V	B1U	B1GRN	B1U
42	B2Y	E2U	H2V	B2U	B2GRN	B2U
43	B3Y	E3U	H3V	B3U	B3GRN	B3U
44	B4Y	E4U	H4V	B4U	B4GRN	B4U
45	B5Y	E5U	H5V	B5U	B5GRN	B5U
46	B6Y	E6U	H6V	B6U	B6GRN	B6U
47	B7Y	E7U	H7V	B7U	B7GRN	B7U
48	C0V	E0Y	H0U	B0Y	B0RED	B0Y
49	C1V	E1Y	H1U	B1Y	B1RED	B1Y
50	C2V	E2Y	H2U	B2Y	B2RED	B2Y
51	C3V	E3Y	H3U	B3Y	B3RED	B3Y
52	C4V	E4Y	H4U	B4Y	B4RED	B4Y
53	C5V	E5Y	H5U	B5Y	B5RED	B5Y
54	C6V	E6Y	H6U	B6Y	B6RED	B6Y
55	C7V	E7Y	H7U	B7Y	B7RED	B7Y
56	C0U	F0V	H0Y	UNUSED	B0OVL8	B0OVL8
57	C1U	F1V	H1Y	UNUSED	B1OVL8	B1OVL8
58	C2U	F2V	H2Y	UNUSED	B2OVL8	B2OVL8
59	C3U	F3V	H3Y	UNUSED	B3OVL8	B3OVL8
60	C4U	F4V	H4Y	UNUSED	B4OVL8	B4OVL8
61	C5U	F5V	H5Y	UNUSED	B5OVL8	B5OVL8
62	C6U	F6V	H6Y	UNUSED	B6OVL8	B6OVL8
63	C7U	F7V	H7Y	UNUSED	B7OVL8	B7OVL8

Note 1: YUV formats are for U.V order (Miscellaneous Control 3 register SWAP UV bit = 0). All YUV pixels go directly to the DACs.

Note 2: For 32 BPP with chroma key enabled, if the upper byte does not match the chroma key it is used as an 8 BPP indirect color overlay (OVL8). If the chroma key is matched the three lower bytes are used as 24 bit RGB (indirect or direct) or 24 bit YUV. Dynamic bypass is not available when chroma keying is enabled.

Table 9. Pixel Format Table - 15/16 BPP YUV and Overlays, U/V Swapped

Pixel Port Bit	Tagged[2,3]						YUV		YUV with Luma Key[4]			
	RGB 555		YUV				SWAP YC = 0	SWAP YC = 1	SWAP YC = 0		SWAP YC = 1	
	SPR	CNT	B16 MODE = 01		B16 MODE = 11				No Match	Match	No Match	Match
			SWAP YC = 0	SWAP YC = 1	SWAP YC = 0	SWAP YC = 1						
0	A3BLU	A0BLU	A1V	A1Y	A0V	A0Y	A0V	A0Y	A0V	A0CLUT8	A0Y	A0KEY
1	A4BLU	A1BLU	A2V	A2Y	A1V	A1Y	A1V	A1Y	A1V	A1CLUT8	A1Y	A1KEY
2	A5BLU	A2BLU	A3V	A3Y	A2V	A2Y	A2V	A2Y	A2V	A2CLUT8	A2Y	A2KEY
3	A6BLU	A3BLU	A4V	A4Y	A3V	A3Y	A3V	A3Y	A3V	A3CLUT8	A3Y	A3KEY
4	A7BLU	A4BLU	A5V	A5Y	A4V	A4Y	A4V	A4Y	A4V	A4CLUT8	A4Y	A4KEY
5	A3GRN	A0GRN	A6V	A6Y	A5V	A5Y	A5V	A5Y	A5V	A5CLUT8	A5Y	A5KEY
6	A4GRN	A1GRN	A7V	A7Y	A6V	A6Y	A6V	A6Y	A6V	A6CLUT8	A6Y	A6KEY
7	A5GRN	A2GRN	A0Y	A0V	A7V	A7Y	A7V	A7Y	A7V	A7CLUT8	A7Y	A7KEY
8	A6GRN	A3GRN	A1Y	A1V	A1Y	A1V	A0Y	A0V	A0Y	A0KEY	A0V	A0CLUT8
9	A7GRN	A4GRN	A2Y	A2V	A2Y	A2V	A1Y	A1V	A1Y	A1KEY	A1V	A1CLUT8
10	A3RED	A0RED	A3Y	A3V	A3Y	A3V	A2Y	A2V	A2Y	A2KEY	A2V	A2CLUT8
11	A4RED	A1RED	A4Y	A4V	A4Y	A4V	A3Y	A3V	A3Y	A3KEY	A3V	A3CLUT8
12	A5RED	A2RED	A5Y	A5V	A5Y	A5V	A4Y	A4V	A4Y	A4KEY	A4V	A4CLUT8
13	A6RED	A3RED	A6Y	A6V	A6Y	A6V	A5Y	A5V	A5Y	A5KEY	A5V	A5CLUT8
14	A7RED	A4RED	A7Y	A7V	A7Y	A7V	A6Y	A6V	A6Y	A6KEY	A6V	A6CLUT8
15	ATAGNOT		ATAG				A7Y	A7V	A7Y	A7KEY	A7V	A7CLUT8
16	B3BLU	B0BLU	B1U	B1Y	B0U	B0Y	B0U	B0Y	B0U	B0CLUT8	B0Y	B0KEY
17	B4BLU	B1BLU	B2U	B2Y	B1U	B1Y	B1U	B1Y	B1U	B1CLUT8	B1Y	B1KEY
18	B5BLU	B2BLU	B3U	B3Y	B2U	B2Y	B2U	B2Y	B2U	B2CLUT8	B2Y	B2KEY
19	B6BLU	B3BLU	B4U	B4Y	B3U	B3Y	B3U	B3Y	B3U	B3CLUT8	B3Y	B3KEY
20	B7BLU	B4BLU	B5U	B5Y	B4U	B4Y	B4U	B4Y	B4U	B4CLUT8	B4Y	B4KEY
21	B3GRN	B0GRN	B6U	B6Y	B5U	B5Y	B5U	B5Y	B5U	B5CLUT8	B5Y	B5KEY
22	B4GRN	B1GRN	B7U	B7Y	B6U	B6Y	B6U	B6Y	B6U	B6CLUT8	B6Y	B6KEY
23	B5GRN	B2GRN	B0Y	B0U	B7U	B7Y	B7U	B7Y	B7U	B7CLUT8	B7Y	B7KEY
24	B6GRN	B3GRN	B1Y	B1U	B1Y	B1U	B0Y	B0U	B0Y	B0KEY	B0U	B0CLUT8
25	B7GRN	B4GRN	B2Y	B2U	B2Y	B2U	B1Y	B1U	B1Y	B1KEY	B1U	B1CLUT8
26	B3RED	B0RED	B3Y	B3U	B3Y	B3U	B2Y	B2U	B2Y	B2KEY	B2U	B2CLUT8
27	B4RED	B1RED	B4Y	B4U	B4Y	B4U	B3Y	B3U	B3Y	B3KEY	B3U	B3CLUT8
28	B5RED	B2RED	B5Y	B5U	B5Y	B5U	B4Y	B4U	B4Y	B4KEY	B4U	B4CLUT8
29	B6RED	B3RED	B6Y	B6U	B6Y	B6U	B5Y	B5U	B5Y	B5KEY	B5U	B5CLUT8
30	B7RED	B4RED	B7Y	B7U	B7Y	B7U	B6Y	B6U	B6Y	B6KEY	B6U	B6CLUT8
31	BTAGNOT		BTAG				B7Y	B7U	B7Y	B7KEY	B7U	B7CLUT8
32	C3BLU	C0RED	C1V	C1Y	C0V	C0Y	C0V	C0Y	C0V	C0CLUT8	C0Y	C0KEY
33	C4BLU	C1BLU	C2V	C2Y	C1V	C1Y	C1V	C1Y	C1V	C1CLUT8	C1Y	C1KEY
34	C5BLU	C2BLU	C3V	C3Y	C2V	C2Y	C2V	C2Y	C2V	C2CLUT8	C2Y	C2KEY
35	C6BLU	C3BLU	C4V	C4Y	C3V	C3Y	C3V	C3Y	C3V	C3CLUT8	C3Y	C3KEY
36	C7BLU	C4BLU	C5V	C5Y	C4V	C4Y	C4V	C4Y	C4V	C4CLUT8	C4Y	C4KEY
37	C3GRN	C0GRN	C6V	C6Y	C5V	C5Y	C5V	C5Y	C5V	C5CLUT8	C5Y	C5KEY
38	C4GRN	C1GRN	C7V	C7Y	C6V	C6Y	C6V	C6Y	C6V	C6CLUT8	C6Y	C6KEY
39	C5GRN	C2GRN	C0Y	C0V	C7V	C7Y	C7V	C7Y	C7V	C7CLUT8	C7Y	C7KEY
40	C6GRN	C3GRN	C1Y	C1V	C1Y	C1V	C0Y	C0V	C0Y	C0KEY	C0V	C0CLUT8
41	C7GRN	C4GRN	C2Y	C2V	C2Y	C2V	C1Y	C1V	C1Y	C1KEY	C1V	C1CLUT8
42	C3RED	C0RED	C3Y	C3V	C3Y	C3V	C2Y	C2V	C2Y	C2KEY	C2V	C2CLUT8
43	C4RED	C1RED	C4Y	C4V	C4Y	C4V	C3Y	C3V	C3Y	C3KEY	C3V	C3CLUT8
44	C5RED	C2RED	C5Y	C5V	C5Y	C5V	C4Y	C4V	C4Y	C4KEY	C4V	C4CLUT8
45	C6RED	C3RED	C6Y	C6V	C6Y	C6V	C5Y	C5V	C5Y	C5KEY	C5V	C5CLUT8
46	C7RED	C4RED	C7Y	C7V	C7Y	C7V	C6Y	C6V	C6Y	C6KEY	C6V	C6CLUT8
47	CTAGNOT		CTAG				C7Y	C7V	C7Y	C7KEY	C7V	C7CLUT8
48	D3BLU	D0BLU	D1U	D1Y	D0U	D0Y	D0U	D0Y	D0U	D0CLUT8	D0Y	D0KEY
49	D4BLU	D1BLU	D2U	D2Y	D1U	D1Y	D1U	D1Y	D1U	D1CLUT8	D1Y	D1KEY
50	D5BLU	D2BLU	D3U	D3Y	D2U	D2Y	D2U	D2Y	D2U	D2CLUT8	D2Y	D2KEY
51	D6BLU	D3BLU	D4U	D4Y	D3U	D3Y	D3U	D3Y	D3U	D3CLUT8	D3Y	D3KEY
52	D7BLU	D4BLU	D5U	D5Y	D4U	D4Y	D4U	D4Y	D4U	D4CLUT8	D4Y	D4KEY
53	D3GRN	D0GRN	D6U	D6Y	D5U	D5Y	D5U	D5Y	D5U	D5CLUT8	D5Y	D5KEY
54	D4GRN	D1GRN	D7U	D7Y	D6U	D6Y	D6U	D6Y	D6U	D6CLUT8	D6Y	D6KEY
55	D5GRN	D2GRN	D0Y	D0U	D7U	D7Y	D7U	D7Y	D7U	D7CLUT8	D7Y	D7KEY
56	D6GRN	D3GRN	D1Y	D1U	D1Y	D1U	D0Y	D0U	D0Y	D0KEY	D0U	D0CLUT8
57	D7GRN	D4GRN	D2Y	D2U	D2Y	D2U	D1Y	D1U	D1Y	D1KEY	D1U	D1CLUT8
58	D3RED	D0RED	D3Y	D3U	D3Y	D3U	D2Y	D2U	D2Y	D2KEY	D2U	D2CLUT8
59	D4RED	D1RED	D4Y	D4U	D4Y	D4U	D3Y	D3U	D3Y	D3KEY	D3U	D3CLUT8
60	D5RED	D2RED	D5Y	D5U	D5Y	D5U	D4Y	D4U	D4Y	D4KEY	D4U	D4CLUT8
61	D6RED	D3RED	D6Y	D6U	D6Y	D6U	D5Y	D5U	D5Y	D5KEY	D5U	D5CLUT8
62	D7RED	D4RED	D7Y	D7U	D7Y	D7U	D6Y	D6U	D6Y	D6KEY	D6U	D6CLUT8
63	DTAGNOT		DTAG				D7Y	D7U	D7Y	D7KEY	D7U	D7CLUT8

Note 1: YUV formats are for V,U order (Miscellaneous Control 3 register SWAP UV bit = 1). All YUV pixels go directly to the DACs.

Note 2: For Tagged RGB/YUV the TAG POL bit of the 16 BPP Control register sets the meaning of the tag bit. (For TAG POL = 0, incoming TAG = 0 selects RGB, TAG = 1 selects YUV. For TAG POL = 1, incoming TAG = 0 selects YUV, TAG = 1 selects RGB.)

Note 3: For Tagged RGB/YUV, when the tag selects RGB the Indirect/Direct, Sparse/Contiguous, and ZIB/LIN selections are the same as untagged 555 RGB. However, dynamic bypass is not available.

Note 4: For 16 bit YUV with luma key enabled, if the Y value does not match the luma key YUV is used. When the Y position matches the key, the remaining byte is used as 8 BPP indirect color (CLUT8 = Color Lookup Table, 8 BPP). The bytes used, high or low, for Y and CLUT8 are determined by the SWAP YC bit.

Table 10. Pixel Format Table - 24 BPP Packed, 32 BPP YUV and Overlays, U/V Swapped

Pixel Port Bit	24 BPP Packed YUV[1]			32 BPP[1]		
	1st Access	2nd Access	3rd Access	YUV	RGB OVL8[2]	YUV OVL8[2]
0	A0U	C0Y	F0V	A0U	A0BLU	A0U
1	A1U	C1Y	F1V	A1U	A1BLU	A1U
2	A2U	C2Y	F2V	A2U	A2BLU	A2U
3	A3U	C3Y	F3V	A3U	A3BLU	A3U
4	A4U	C4Y	F4V	A4U	A4BLU	A4U
5	A5U	C5Y	F5V	A5U	A5BLU	A5U
6	A6U	C6Y	F6V	A6U	A6BLU	A6U
7	A7U	C7Y	F7V	A7U	A7BLU	A7U
8	A0V	D0U	F0Y	A0V	A0GRN	A0V
9	A1V	D1U	F1Y	A1V	A1GRN	A1V
10	A2V	D2U	F2Y	A2V	A2GRN	A2V
11	A3V	D3U	F3Y	A3V	A3GRN	A3V
12	A4V	D4U	F4Y	A4V	A4GRN	A4V
13	A5V	D5U	F5Y	A5V	A5GRN	A5V
14	A6V	D6U	F6Y	A6V	A6GRN	A6V
15	A7V	D7U	F7Y	A7V	A7GRN	A7V
16	A0Y	D0V	G0U	A0Y	A0RED	A0Y
17	A1Y	D1V	G1U	A1Y	A1RED	A1Y
18	A2Y	D2V	G2U	A2Y	A2RED	A2Y
19	A3Y	D3V	G3U	A3Y	A3RED	A3Y
20	A4Y	D4V	G4U	A4Y	A4RED	A4Y
21	A5Y	D5V	G5U	A5Y	A5RED	A5Y
22	A6Y	D6V	G6U	A6Y	A6RED	A6Y
23	A7Y	D7V	G7U	A7Y	A7RED	A7Y
24	B0U	D0Y	G0V	UNUSED	A0OVL8	A0OVL8
25	B1U	D1Y	G1V	UNUSED	A1OVL8	A1OVL8
26	B2U	D2Y	G2V	UNUSED	A2OVL8	A2OVL8
27	B3U	D3Y	G3V	UNUSED	A3OVL8	A3OVL8
28	B4U	D4Y	G4V	UNUSED	A4OVL8	A4OVL8
29	B5U	D5Y	G5V	UNUSED	A5OVL8	A5OVL8
30	B6U	D6Y	G6V	UNUSED	A6OVL8	A6OVL8
31	B7U	D7Y	G7V	UNUSED	A7OVL8	A7OVL8
32	B0V	E0U	G0Y	B0U	B0BLU	B0U
33	B1V	E1U	G1Y	B1U	B1BLU	B1U
34	B2V	E2U	G2Y	B2U	B2BLU	B2U
35	B3V	E3U	G3Y	B3U	B3BLU	B3U
36	B4V	E4U	G4Y	B4U	B4BLU	B4U
37	B5V	E5U	G5Y	B5U	B5BLU	B5U
38	B6V	E6U	G6Y	B6U	B6BLU	B6U
39	B7V	E7U	G7Y	B7U	B7BLU	B7U
40	B0Y	E0V	H0U	B0V	B0GRN	B0V
41	B1Y	E1V	H1U	B1V	B1GRN	B1V
42	B2Y	E2V	H2U	B2V	B2GRN	B2V
43	B3Y	E3V	H3U	B3V	B3GRN	B3V
44	B4Y	E4V	H4U	B4V	B4GRN	B4V
45	B5Y	E5V	H5U	B5V	B5GRN	B5V
46	B6Y	E6V	H6U	B6V	B6GRN	B6V
47	B7Y	E7V	H7U	B7V	B7GRN	B7V
48	C0U	E0Y	H0V	B0Y	B0RED	B0Y
49	C1U	E1Y	H1V	B1Y	B1RED	B1Y
50	C2U	E2Y	H2V	B2Y	B2RED	B2Y
51	C3U	E3Y	H3V	B3Y	B3RED	B3Y
52	C4U	E4Y	H4V	B4Y	B4RED	B4Y
53	C5U	E5Y	H5V	B5Y	B5RED	B5Y
54	C6U	E6Y	H6V	B6Y	B6RED	B6Y
55	C7U	E7Y	H7V	B7Y	B7RED	B7Y
56	C0V	F0U	H0Y	UNUSED	B0OVL8	B0OVL8
57	C1V	F1U	H1Y	UNUSED	B1OVL8	B1OVL8
58	C2V	F2U	H2Y	UNUSED	B2OVL8	B2OVL8
59	C3V	F3U	H3Y	UNUSED	B3OVL8	B3OVL8
60	C4V	F4U	H4Y	UNUSED	B4OVL8	B4OVL8
61	C5V	F5U	H5Y	UNUSED	B5OVL8	B5OVL8
62	C6V	F6U	H6Y	UNUSED	B6OVL8	B6OVL8
63	C7V	F7U	H7Y	UNUSED	B7OVL8	B7OVL8

Note 1: YUV formats are for V.U order (Miscellaneous Control 3 register SWAP UV bit = 1). All YUV pixels go directly to the DACs.

Note 2: For 32 BPP with chroma key enabled, if the upper byte does not match the chroma key it is used as an 8 BPP indirect color overlay (OVL8). If the chroma key is matched the three lower bytes are used as 24 bit RGB (indirect or direct) or 24 bit YUV. Dynamic bypass is not available when chroma keying is enabled.

Table 11. Pixel Format Table - RGB624DB Double Buffer Operation

Pixel Port Bit				15/16 BPP^{2,3}								
				555 Sparse or Direct Color	555 CONTIG							
0				A - A3BLU	A - A0BLU							
1				A - A4BLU	A - A1BLU							
2				A - A5BLU	A - A2BLU							
3				A - A6BLU	A - A3BLU							
4				A - A7BLU	A - A4BLU							
5				A - A3GRN	A - A0GRN							
6				A - A4GRN	A - A1GRN							
7				A - A5GRN	A - A2GRN							
8				A - A6GRN	A - A3GRN							
9				A - A7GRN	A - A4GRN							
10				A - A3RED	A - A0RED							
11				A - A4RED	A - A1RED							
12				A - A5RED	A - A2RED							
13				A - A6RED	A - A3RED							
14				A - A7RED	A - A4RED							
15				UNUSED								
16				B - A3BLU	B - A0BLU							
17				B - A4BLU	B - A1BLU							
18				B - A5BLU	B - A2BLU							
19				B - A6BLU	B - A3BLU							
20				B - A7BLU	B - A4BLU							
21				B - A3GRN	B - A0GRN							
22				B - A4GRN	B - A1GRN							
23				B - A5GRN	B - A2GRN							
24				B - A6GRN	B - A3GRN							
25				B - A7GRN	B - A4GRN							
26				B - A3RED	B - A0RED							
27				B - A4RED	B - A1RED							
28				B - A5RED	B - A2RED							
29				B - A6RED	B - A3RED							
30				B - A7RED	B - A4RED							
31				A/B BUF SEL (NOTE 4)								
32				A - B3BLU	A - B0RED							
33				A - B4BLU	A - B1BLU							
34				A - B5BLU	A - B2BLU							
35				A - B6BLU	A - B3BLU							
36				A - B7BLU	A - B4BLU							
37				A - B3GRN	A - B0GRN							
38				A - B4GRN	A - B1GRN							
39				A - B5GRN	A - B2GRN							
40				A - B6GRN	A - B3GRN							
41				A - B7GRN	A - B4GRN							
42				A - B3RED	A - B0RED							
43				A - B4RED	A - B1RED							
44				A - B5RED	A - B2RED							
45				A - B6RED	A - B3RED							
46				A - B7RED	A - B4RED							
47				UNUSED								
48				B - B3BLU	B - B0BLU							
49				B - B4BLU	B - B1BLU							
50				B - B5BLU	B - B2BLU							
51				B - B6BLU	B - B3BLU							
52				B - B7BLU	B - B4BLU							
53				B - B3GRN	B - B0GRN							
54				B - B4GRN	B - B1GRN							
55				B - B5GRN	B - B2GRN							
56				B - B6GRN	B - B3GRN							
57				B - B7GRN	B - B4GRN							
58				B - B3RED	B - B0RED							
59				B - B4RED	B - B1RED							
60				B - B5RED	B - B2RED							
61				B - B6RED	B - B3RED							
62				B - B7RED	B - B4RED							
63				A/B BUF SEL (NOTE 4)								

Note 1: The double buffer formats in this table are only for the RGB624DB, and only when the DBUF MODE bit is set (bit 0 of the Miscellaneous Control 3 register).

Note 2: For 15/16 BPP Direct Color the low order bits for each color component are determined by the ZIB/LIN Bit of the 16 BPP Control register. For 15/16 BPP sparse format (indirect color), the ZIB/LIN bit must be set to ZIB, and the low order bits for each color component will be zeroes.

Note 3: In CONTIGUOUS format for 15/16 BPP the most significant bits of each pixel come from the partition bits of the palette control register.

Note 4: When the SUPP BUFB bit is set (bit 0 of the Key Control/DB Operation register), the A/B BUFFER SELECT bits are ignored and Buffer A pixels are selected.

7.0 Color Space Conversion

The internal color space converter takes pixels with Y, U, and V values of 8 bits each and converts them to pixels of R, G, and B values of 8 bits each, using the following equations:

R = Y + K1(V - 128)

G = Y - K2(V - 128) - K3(U - 128)

B = Y + K4(U - 128)

7.1 Input Values

Y, the luminance value, ranges from 0.0 (0x00) to 1.0 (0xff). U and V, the chrominance values, are “offset 128” values, and represent the range -0.5 to +0.5 after 128 (0x80) is subtracted. (This makes an incoming value of 128 the “zero chrominance” value.)

For 16-bit YUV there are 256 values for Y, U and V. For the 15-bit YUV formats used with tagged RGB/YUV, whichever component (Y, or U/V) that has the 7-bit field will only have 128 values. As discussed in section 5.6.2, the 7-bit field is missing the low order bit, and a ‘0’ is substituted internally for this bit. Note that the most positive value for Y or U/V in this case is 0xfe.

7.1.1 Range Endpoints

Some encoding systems for video do not use the full scale range. E.g., it is common to restrict the range of Y values to 16 through 235. The unused values may be used for other purposes. In particular, the values 0 and 255 are used for timing information.

Some color space converters convert values less than 16 to 16, and values greater than 235 to 235. The RGB624/RGB624DB does *not* do this. All values of Y, U and V are presented unchanged to the color space converter as received from the frame buffer.

7.2 Coefficients

The constants K1 - K4 are 8 bit values are can be programmed by the user with four new 8 bit registers. The constants represent the range 0.0 to 1.9921875, using the following bit weighting:

Kn[7] = 1.0000000 (MSB)

Kn[6] = 0.5000000

Kn[5] = 0.2500000

Kn[4] = 0.1250000

Kn[3] = 0.0625000

Kn[2] = 0.0312500

Kn[1] = 0.0156250

Kn[0] = 0.0078125 (LSB)

(Kn is the register K1, K2, K3 or K4. [7], [6], [5]… is the bit position within the register.)

The constants must be stable during conversion, (i.e., the register values must not be changed.)

7.3 Output Values

The R, G and B outputs of the color space converter do not overflow or underflow. Excessive high end values “saturate” at 0xff (full on), and low end values saturate at 0x00 (full off).

7.4 Bit Precision

Internally, the multiplications are generated at full precision. The computed R, G, and B values are then rounded to 10 bits. Finally, the two least significant bits for each color output of the color space converter are dropped (the 10 bit results are truncated to 8 bits).

8.0 Sub-sampled Chrominance Issues

Several issues must be addressed when using sub-sampled chrominance formats:

1. Pair Alignment
2. Interpolation
3. Edge-effects

8.1 Pair alignment

The RGB624/RGB624DB requires that the pairs of YU/V pixels be aligned on even pixel position values. That is, starting at the left end of a line, pixels 0 and 1 can be a pair of YU/V pixels (Y0U0, Y1V0) and pixels 2 and 3 can be pair of YU/V pixels (Y2U2,Y3V2), but pixels 1 and 2 cannot be a pair of YU/V pixels.

When the YU/V data is intermixed with RGB data, it is preferable that the RGB pixels also start on even pixel position values and end on odd pixel position values. This will avoid “edge effects”. (See below.)

8.1.1 U,V ordering

For a pair of YU/V samples, the default condition is to have the first pixel contain U and the second pixel contain V:

[Y0,U0] [Y1,V0] [Y2,U2] [Y3,V2] …

This order can be reversed by setting the UV SWAP bit in the Miscellaneous Control 3 register:

[Y0,V0] [Y1,U0] [Y2,V2] [Y3,U2] …

8.2 Interpolation

For the YUV formats using sub-sampled chrominance, the missing chrominance values will be generated by simply duplicating the existing value from the sample pair. Thus, for the incoming samples:

[Y0,U0] [Y1,V0] [Y2,U2] [Y3,V2] …

the pixels presented to the color space converter will be:

[Y0,U0,V0] [Y1,U0,V0] [Y2,U2,V2] [Y3,U2,V2] …

8.3 Edge Effect Handling

With mixed YU/V and RGB pixels, the requirement that the YU/V pixels occur in pairs also, in effect, requires that the RGB pixels occur in pairs and that the pairs be aligned on even pixel boundaries. If this condition is violated (there is a pair of pixels with one pixel tagged as RGB and the other tagged as YUV) then the YUV pixel will be missing either the U or the V value.

As an example of this situation, consider a GUI environment in which a rectangular window of subsampled YUV pixels is displayed against a background of RGB pixels. With the default U,V ordering, if the YUV window starts on an odd pixel boundary, then the first pixels of that window (the left edge) will be missing their U values. If the right edge ends on an even boundary then those pixels will be missing their V values. Whatever values are used for the missing U and V values can distort the colors of the left and right edges, hence the phrase “edge effects”.

The RGB624/RGB624DB handles this situation as follows. If a pixel has a missing chrominance value, either U or V, then the missing value is forced to the “zero chrominance” value. As discussed previously, this is decimal 128 (hex 0x80).

In addition, the V or U value that was received for that pixel is discarded, and the “zero chrominance” value is substituted. That is, the pixel presented to the color space converter is [Y,128,128].

This has the effect of causing all “edge effect” pixels to display in shades of grey.

9.0 Controls

9.1 Blank and Border Control

The BLANK and BORDER/OE signals control the way in which data is presented to the DACs. These control signals are used to determine when pixel data is valid, when the border color is to be displayed, where the cursor should be located on the screen, and how the MISR will accumulate its signature.

9.2 Blanking Control

BLANK is latched by the rising edge of LCLK. When BLANK is active (low), the data presented to the DACs is forced to zeroes. When BLANK is inactive (high), the pixel data or VGA data is considered valid (unless BORDER is active), and the data is presented to the DACs as determined by the current mode of operation. Cursor data will override pixel data when the cursor is to be displayed.

9.3 Vertical Blanking

When BLANK is active (low) an internal counter is used to determine whether or not the current blanking interval is vertical blanking. If the counter reaches its maximum count of 2048 pixels, an internal signal will become active to indicate that the end of the current frame has been reached. This internal signal will remain active until BLANK becomes inactive (high). This vertical blanking detection is used by the cursor logic to position the cursor (if enabled) in the following frame. It is also used by the MISR (if enabled) to control the accumulation of a signature for one complete frame of pixel data.

9.4 Border Control

BORDER/OE is a shared function input. It can indicate either "Border" time, for displaying a border, or "Odd/Even" for use with interlace mode. The usage of this pin is determined by the BRDR/INTL of the Miscellaneous Control 2 register. When used as a border control interlace mode is not supported with display of the hardware cursor.

When used as BORDER, the input is latched by the rising edge of LCLK. When BLANK is active (low), BORDER must also be active (low). When BLANK is inactive (high), the state of BORDER will determine whether or not the color in the Border Color registers is displayed. If BORDER is active (low), the border color is displayed, and if BORDER is inactive (high), the pixel data or cursor data is displayed. For cursor positioning, the active display area is considered valid when BORDER and BLANK are both inactive (high). The MISR signature is accumulated when BLANK alone is inactive (high), thus the border area is included in the MISR accumulation. If no border is required, the BORDER input should be tied to BLANK.

The intent of the BORDER signal is to create a "picture frame" around the active display area. BORDER can remain active (low) for entire scan lines at the top and bottom of the active display area, or it can be active at the beginning and end of each scan line to create this effect. Other changes in the BORDER signal within the active display area are not allowed.

If the BRDR/INTL bit in Miscellaneous Control 2 is set to "INTL" operation, no border will be displayed.

9.5 Sync Control

Three sync signals are brought into the device on two pins, HCSYNCIN and VSYNCIN.

Five registers control what is done with these signals:

- ❑ Sync Control (index 0x0003)
- ❑ Horizontal Sync Position (index 0x0004)
- ❑ DAC Operation (index 0x0006)
- ❑ Power Management (index 0x0005)
- ❑ Miscellaneous Control 1 (index 0x0070)

Horizontal sync on HCSYNCIN is processed and sent out on HSYNCOUT. Vertical sync on VSYNCIN is processed and sent out on VSYNCOUT.

The intent of processing horizontal sync is to delay it to match the delay seen by the pixel data from the inputs (VGA[7:0] or PIX[63:0]) to the DAC outputs. In addition, the signal may be inverted, forced low or high, or 3-stated.

A mismatch between pixel delay and horizontal sync delay can cause a visible effect, that is, the display may not be centered horizontally on the screen. The vertical display timings are generally such that mismatches are not visible. Vertical sync is brought in on VSYNCIN and sent out on VSYNCOUT to provide the same invert, force low or high, and 3-state controls as provided for horizontal sync.

Composite sync on HCSYNCIN may be injected onto the Green DAC output for composite-sync-on-green. This function is enabled by setting the SOG bit of the DAC Operation register. If this bit is off HCSYNCIN generally is not used for composite sync.

However, in lieu of providing a composite sync externally, if horizontal sync is provided on the HCSYNCIN and vertical sync is provided on the VSYNCIN, a “synthetic” composite sync may be generated internally by exclusive ORing these two signals. This is done by setting the XOR SYNC bit of the Miscellaneous Control 1 register.

If XOR SYNC is set and SOG is set the internally XORed sync signal will be injected onto the Green DAC output. If XOR SYNC is set but SOG is not set, then the synthetic composite sync will be presented on the HSYNCOUT output, but will not be injected on the Green DAC output.

Composite sync, whether it is the signal presented on HCSYNCIN or the synthetic internal composite sync, is delayed internally to match the pixel pipeline delay.

Since external horizontal and composite sync are shared on the same pin, only one of them should be enabled at a given time. For example, if the signal on HCSYNCIN is horizontal sync and XOR SYNC is not being used, then the SOG bit on the DAC control register should be off. Or, if SOG is on to inject composite sync on HCSYNCIN onto the green DAC output, some decision must be made on how to handle the HSYNCOUT output (force low, high, 3-state, or leave unconnected).

9.6 Clocking and Pipeline Delay

9.6.1 Horizontal Sync

The clocking and delay of HCSYNCIN to HSYNCOUT depends on the DLY CNTL bit of the Sync Control register. If this bit is set to 1, HCSYNCIN is passed directly to HSYNCOUT without latching and without pipeline delay matching.

If DLY CNTL is set to 0 and SOG is off (no composite sync), then HCSYNCIN is latched on the rising edge of LCLK and delayed internally to match the pixel pipeline delay before being sent out on HSYNCOUT. Also, additional delay may be added with the Horizontal Position register (see section below).

If SOG is on (composite sync) then DLY CNTL has no effect and HCSYNCIN is passed directly to HSYNCOUT without latching and without pipeline delay matching. (This is not a typical use for this input, it is just a by-product of sharing horizontal sync with composite sync.)

9.6.2 Vertical Sync

VSYNCIN is passed directly to VSYNCOUT without latching and without pipeline delay matching.

9.6.3 Composite Sync

The HCSYNCIN input is always latched on the rising edge of LCLK for use as composite sync. When enabled with the SOG bit, it is delayed internally to match the pipeline delay of the pixel data, and then is injected onto the Green DAC output. As with horizontal sync, additional delay can be added with the Horizontal Sync Position register.

9.6.4 Horizontal Position Control

Additional delay of 0 to 15 pixel clock periods may be added to the horizontal sync and composite sync signals with the Horizontal Sync Position register.

The intent of this additional delay is to provide a “fine tune” control of the horizontal screen position. Typically the incoming sync signals can only be adjusted in multiples of the pixel clock. The additional delay added with the Horizontal Position Control register adjusts the screen position with pixel increments.

The Horizontal Position register can be used on horizontal sync when DLY CNTL is set to 0 and SOG is off. The register can also be used with composite sync when SOG is on.

9.7 Additional Sync Control

The polarity of the received HCSYNCIN input may be inverted before it is applied to the green DAC using the CSYN INVT bit of the Sync Control register.

The polarity may be inverted between HCSYNCIN and HSYNCOUT using the HSYN INVT bit, and the polarity may be inverted between VSYNCIN and VSYNCOUT using the VSYN INVT bit.

The HSYNCOUT and VSYNCOUT signals may be individually forced low, forced high, or forced to high impedance using the HSYN CNTL and VSYN CNTL bits of the Sync Control register.

As discussed in 12.3 “Clocking Power” on page 35, the clocks to the sync delay circuits can be shut off with the SYNC PWR bit of the Power Management register.

9.8 24 Bit Packed Pixel Control

The 24 bit packed pixel format requires special consideration. In this mode the pixel data at the beginning of a line must be aligned on an eight pixel boundary as shown in Figure 3, “RGB - 24 BPP Packed Pixel Input from VRAM” on page 17. These eight pixels correspond to three 64-bit pixel port loads or three SCLK cycles. In order to keep pixel data and control signals properly aligned, all control signals ($\overline{\text{BLANK}}$, $\overline{\text{BORDER/OE}}$ and $\overline{\text{HCSYNCIN}}$) are required to change in increments of eight pixels (3 SCLKS). When either $\overline{\text{BLANK}}$ or $\overline{\text{BORDER/OE}}$ changes to indicate the beginning of an active display line, it is assumed that the pixel data which begins that line is aligned on the proper eight pixel boundary.

10.0 Cursor Operation

The cursor is a 32x32 or 64x64 pixel pattern that is overlaid on the display pixels just before presentation to the DACs. The cursor size, 32x32 or 64x64 is set with the CURS SIZE bit of the Cursor Control register.

Pixel columns are numbered left to right starting with 0. Pixel rows are numbered top to bottom starting with 0.

10.1 Cursor Enable

The cursor is enabled when the CURSOR MODE bits of the Cursor Control register are not 00. When enabled, the cursor will display if it has not been moved off-screen. If disabled (CURSOR MODE = 00), the cursor will not be displayed.

The cursor may be used with either pixel port (VGA or PIX), with any of the pixel formats (VGA, 4, 8, 15/16, 24, 32 BPP), and with indirect or direct color.

10.2 Cursor Array

The cursor image is stored in the Cursor Array. The array is organized 1024x8 (1024 bytes). It is accessed as Indexed Data using index addresses 0x0100 through 0x04ff.

Each pixel of the cursor uses 2 bits, thus 4 cursor pixels are stored in each byte of the array. The entire array is used to contain the 64x64 cursor image (4 pixels/byte × 1024 bytes = 4096 pixels = 64x64).

For the 32x32 cursor only 256 bytes are required (4 pixels/byte × 256 bytes = 1024 pixels = 32x32.) The cursor array is divided into four contiguous slots to allow the storage of four cursor images. The SMLC SLOT bits of the Cursor Control register are used to select one of the four slots for display. The SMLC SLOT bits have no effect when the cursor size is 64x64.

Storage of the cursor within the array starts with the top row. For the 64x64 cursor the first 16 bytes hold row 0, the next 16 bytes hold row 1, and so on, starting with the first byte in the array at index address 0x0100.

For the 32x32 cursor the first 8 bytes hold row 0, the next 8 bytes hold row 1, and so on, starting with the first byte in a slot (index addresses 0x0100, 0x0200, 0x0300 or 0x0400).

Within a row the pixels are stored left to right in groups of four. The first byte holds pixels 0, 1, 2, 3, the next byte holds pixels 4, 5, 6, 7, and so on.

Within a byte the four pixels may be stored right to left or left to right, depending on the PIX ORDR bit of the Cursor Control register. If PIX ORDR = 0 the pixels are stored right to left (3, 2, 1, 0); if PIX ORDR = 1 the pixels are stored left to right (0, 1, 2, 3).

10.2.1 Cursor Array Access

Cursor Array writes and reads are synchronized with the internal pixel clock, so the pixel clock must be running for microprocessor accesses to be valid. If this condition is met, the cursor array may be written or read at any time.

Microprocessor read accesses of the cursor array may disturb the cursor image if it is being displayed at that time. However, no more than one cursor pixel will be disturbed per cursor read access. Microprocessor write accesses of the cursor array will not disturb the cursor.

10.2.2 Cursor Array Writes

A write to the cursor array is accomplished by writing the Index High and Index Low registers with an index address for the array (0x0100 – 0x04ff), followed by a write of the desired data to Index Data. If auto-increment is turned on, the entire array may be written sequentially by repeated writes to Index Data.

10.2.3 Cursor Array Reads

To meet the bus timings for reads, the cursor array read data is pre-fetched. A pre-fetch is triggered by writing the Index High or Index Low register such that the resulting index address is for an entry in the array (0x0100 -- 0x04ff). At the end of the write cycle the cursor array will be read at the address held in the index address registers, and the read data will be held in an internal register. A subsequent read of Index Data will read this pre-fetched data. At the end of the read another pre-fetch will be triggered. If auto-increment is turned on, this pre-fetch will be for the next address in the array. Thus, the entire array can be read by repeated reads from Index Data.

The pre-fetching of cursor array data will stop if

1. The index register auto-increments beyond 0x04ff

   OR

2. A write is done to Index Data

10.3 Cursor Modes

Each pixel of the cursor is specified with 2 bits. There are two fundamental ways to interpret these bits: Standard Modes and Advanced Mode.

The ACA ENAB (Advanced Cursor Attribute Enable) bit in the Advanced Cursor Control register selects either the Standard Cursor (ACA ENAB = ‘0’) or the Advanced Cursor (ACA ENAB = ‘1’).

For either cursor type, there are three cursor colors that may be displayed. The colors are stored in the Cursor Color 1 Red, Green, Blue, Cursor Color 2 Red, Green, Blue, and Cursor Color 3 Red, Green, Blue registers. Each red, green, and blue register is 8 bits, yielding a full 24-bit color for each of the three cursor colors. The cursor color is always 24 bits, and is not affected by the COL RES or 6BIT LIN control bits, or any of the pixel formats (VGA, 4, 8, 15/16, 24, 32 BPP).

The Advanced Cursor has an additional color, Color 0, which is always Black. That is, all 24 bits are ‘0’s.

Both cursor types can generate three types of displayed pixels:

1. Transparent - the underlying pixel (the normally displayed pixel) is displayed. This pixel will either be a palette output or a formatted VRAM pixel, depending on whether the pixel format is VGA, indirect color, direct color or converted YUV.
2. Solid Color - One of the three cursor colors is displayed (Cursor Color 1, 2 or 3). The Advanced Cursor can also display Cursor Color 0.
3. Highlighted - a bitwise complement of the underlying display pixel is displayed. The intent is to highlight the cursor by “reversing” the color of the background pixels.

The Advanced Cursor has an additional display type:

4. Translucent - the underlying pixel is “mixed” with one of the Cursor Colors (0, 1, 2 or 3), such that a “dimmed” version of the underlying pixel appears to “shine through” a translucent version of the Cursor Color.

   This is achieved by shifting each red, green and blue component of the underlying pixel right by 1 bit (dimming the pixel by dividing by 2). The cursor color is mixed in by setting the now vacant high order bit of each component of the display pixel with the high order bit of the corresponding Cursor Color component.

10.3.1 Standard Cursor

The Standard Cursor modes are used when the cursor is ON (CURSOR MODE not equal to ‘00’), and the Advanced Attributes are OFF (ACA ENAB = ‘0’).

The Standard Cursor has three modes of display, specified with the remaining bit values of CURSOR MODE (CURSOR MODE = ‘01’, ‘10’ or ‘11’). The two cursor pixel bits can select one of three colors, one of two colors plus highlighting, or just two colors (where the selected cursor colors are solid colors):

Table 12. Standard Cursor Modes

CURSOR PIXEL	DISPLAY PIXEL		
	Mode 0 CURSOR MODE = 01	Mode 1 CURSOR MODE = 10	Mode 2 CURSOR MODE = 11
00	Transparent	Cursor Color 1	Transparent
01	Cursor Color 1	Cursor Color 2	Transparent
10	Cursor Color 2	Transparent	Cursor Color 1
11	Cursor Color 3	Highlighted	Cursor Color 2

10.3.2 Advanced Cursor

The Advanced Cursor mode is used when the cursor is ON (CURSOR MODE not equal to ‘00’), and the Advanced Attributes are ON (ACA ENAB = ‘1’).

For the Advanced Cursor, the two cursor pixel bits simply select one of four colors:

Table 13. Advanced Cursor Colors

CURSOR PIXEL	CURSOR COLOR
00	Color 0
01	Color 1
10	Color 2
11	Color 3

Associated with each color is a two bit Attribute. The Attribute specifies the display of the cursor pixel:

Table 14. Advanced Cursor Attributes

ATTRIBUTE	DISPLAY PIXEL
00	Transparent
01	Solid color
10	Translucent
11	Highlighted

The attribute values for each color are set in the Advanced Cursor Attribute register. This register contains a two bit attribute value for each of the four colors.

Example: Suppose a given cursor pixel value read from the cursor array is ‘01’. From Table 13, Cursor Color 1 will be selected.

Bits 3 and 2 of the Advanced Cursor Attribute register contain the attribute value for Color 1. Suppose these bits are ‘01’. This specifies a solid color, so the values in the Cursor Color 1 Red, Green and Blue registers will be displayed.

If Bits 3 and 2 of the Advanced Cursor Attribute register were ‘10’, a translucent pixel would be selected, and the values in the Cursor Color 1 Red, Green and Blue registers would be mixed in with the underlying pixels for display.

10.4 Cursor Hot Spot

The hot spot is the point within the cursor that is used to locate the cursor's position on the screen. Any pixel within the cursor may be identified as the hot spot.

The Cursor Hot Spot X and Cursor Hot Spot Y registers hold the unsigned cursor pixel X (column) and Y (row) coordinates for the hot spot. The range for the X and Y values is 0 to 31 for the 32x32 cursor and 0 to 63 for the 64x64 cursor.

10.5 Cursor Position

The Cursor X Low, Cursor X High, Cursor Y Low, and Cursor Y High registers specify the position of the cursor (the cursor hot spot) on the screen.

The X and Y positions are specified as *signed* numbers in two's complement format. The High and Low pairs yield 16-bit position registers, of which 12 bits plus a sign bit are used.

The hardware automatically extends the sign bit into the unused bit positions of the position registers. The valid X and Y ranges are -4096 to +4095.

The X and Y screen coordinates are for non-border display pixels. (0,0) is the upper left corner pixel of the screen that is not in the border area. The X value increases positively left-to-right, and the Y value increases positively top-to-bottom. Negative X values are to the left of the non-border display area and negative Y values are above the top of the non-border display area.

The cursor is clipped by the edges of the screen if there is no border, or by the border if a border is used. For example, if the hot spot is (0,0) the full cursor will be displayed in the upper left corner if the X position is +0 and the Y position is +0. If the X value is changed to -1 (0xffff) only columns 1 through 31 of the cursor will be displayed. If the X value is -31 (0xffe1) only column 31 of the cursor will be displayed. If the X value is more negative than -31 the cursor will not be displayed.

10.6 Interlace

The selection of cursor rows for display is changed if interlace mode is specified. This is controlled with the INTL MODE bit of the Miscellaneous Control 2 register.

In non-interlaced mode, the cursor rows are displayed sequentially, starting with the first non-clipped row to be displayed based on the Y position and Y Hot Spot register contents.

When interlaced mode is specified, the ODD/$\overline{\text{EVEN}}$ signal (actually $\overline{\text{BORDER}}$/O$\overline{\text{E}}$, see below) is used to determine if odd or even scan lines are being displayed. If ODD/$\overline{\text{EVEN}}$ is low (even field), the first non-clipped cursor row that falls on an even scan line is displayed. Similarly if ODD/$\overline{\text{EVEN}}$ is high (odd field), the first non-clipped cursor row that falls on an odd scan line is displayed. In either case, if the first cursor line displayed is an even-numbered cursor row (as determined by Y Position and Y Hot Spot) then successive even-numbered cursor rows will be displayed during that field. If the first cursor line displayed is an odd-numbered cursor row then successive odd-numbered cursor rows will be displayed during that field.

ODD/$\overline{\text{EVEN}}$ should only change during vertical blanking time for proper cursor display.

The ODD/$\overline{\text{EVEN}}$ input is actually the shared input pin $\overline{\text{BORDER}}$/O$\overline{\text{E}}$, where the usage of the pin is determined by the BRDR/INTL of the Miscellaneous Control 2 register. BRDR/INTL must be set to INTL for using the cursor with interlaced mode. If BRDR/INT is set as BRDR then the INTL MODE bit will be ignored (non-interlaced mode will be used).

10.7 Cursor Update and Display

10.7.1 Position

Writing any of the Cursor X Low, Cursor X High, or Cursor Y Low registers will not affect the position of the cursor on the screen. When the Cursor Y High register is written, the X and Y positions are captured in a second set of registers.

When vertical blanking is detected (see 9.3 "Vertical Blanking" on page 28), the "captured" X and Y positions are sampled. The sampled position is saved until it is re-sampled on the next vertical blanking time. Between vertical blanking times the sampled position is used, along with the Cursor Hot Spot, to calculate which pixels of the cursor are used and where they are displayed on the screen.

When INTL MODE is set the ODD/$\overline{\text{EVEN}}$ signal is examined at the end of vertical blanking to determine if only even or only odd rows will be displayed.

10.7.2 Controls

When vertical blanking is detected the Cursor Control register is sampled along with the X and Y position registers. This allows the cursor to be toggled on and off on a frame-by-frame basis with the CURSOR MODE bits, and if the cursor is 32x32, it allows toggling among the four slots on a frame-by-frame basis using the SMLC SLOT bits.

Note that in interlace mode the sampling is on a field-by-field basis. Also, since the PIX ORDER and CURS SIZE bits are also sampled these functions will only change when vertical blanking is detected.

10.7.3 Other

Changes to the Cursor Color registers and the Cursor X and Y Hot Spot registers are propagated to the cursor logic as soon as they are made, so if they are updated while the cursor is being displayed the cursor image will be disturbed.

Changes to the cursor array are also propagated to the cursor logic as soon as they are made. Also, as noted above, microprocessor read accesses of the cursor array may interfere with the cursor display logic. For example, with a 32x32 cursor being displayed from slot 0, microprocessor read accesses to slot 1 may cause the display of the slot 0 cursor to be disturbed.

It is recommended that Cursor Array Reads and changes to the Cursor Color registers and the Cursor X and Y Hot Spot registers be made only when the cursor is disabled, off screen, or during vertical blanking time.

11.0 DAC Control

Several miscellaneous features of the DACs are controlled by the DAC Operation register.

11.1 SOG - Composite Sync-On-Green

When the SOG bit is set, composite sync will be merged with the pixel data on the green DAC.

The source of the composite sync can be the signal on the $\overline{\text{HCSYNCIN}}$ input, or an internally generated (synthetic) sync signal formed by the XOR of the signals on the $\overline{\text{HCSYNCIN}}$ and $\overline{\text{VSYNCIN}}$ inputs. The incoming $\overline{\text{HCSYNCIN}}$ signal may be inverted and/or delayed before presentation at the DAC. See sections 9.5 through 9.7 for more details.

11.2 BRB - Blank Red and Blue DACs

When this is set the red and blue DACs are set to the blanking level. This is intended for use when a monochrome display is driven by the green DAC.

11.3 DSR - DAC Slew Rate

This bit affects the rise and fall times of the DAC analog outputs (slew rate). The default value (on) uses a “slow” rate, typically 5 ns. When the bit is set to “off”, the slew rate will be “fast”, typically 2 ns. The rise and fall times are measured using the 10% point and 90% point.

The faster slew rate will yield the sharpest pixels if the monitor can support that rate. For some monitors it may be desirable to set the DACs to the slower slew rate.

Note: The default value of “on” is new to the RGB6xx and RGB526/RGB526DB products. For most of the RGB51x and RGB52x products the default value is “off”.

11.4 DPE - DAC Blanking Pedestal Enable

When off, the DAC pedestal is disabled (blanking level = 0 IRE). When on, the pedestal is enabled (7.5 IRE).

12.0 Power Management

The following registers are used to control power dissipation:

- Power Management (index 0x0005)
- Miscellaneous Clock Control (index 0x0002)
- Sync Control (index 0x0003)
- Miscellaneous Control 1 (index 0x0070)

12.1 DAC Power

The analog portion of the DACs can be shut down with the DAC PWR bit of the Power Management register. A small amount of current (approximately 100 µA) will continue to be drawn through the VREFIN input. This can be eliminated if the voltage on VREFIN is reduced to 0 V.

12.2 Driver Power

The power dissipated by the logic output signals can be reduced by 3-stating the drivers. This is done for the SCLK driver by setting the SCLK DSAB bit of the Miscellaneous Clock Control register. The DDOTCLK driver can also be 3-stated by setting the DDOT DSAB bit of the Miscellaneous Clock Control register. $\overline{\text{HSYNCOUT}}$ and $\overline{\text{VSYNCOUT}}$ are 3-stated by setting HSYN CNTL and VSYN CNTL bits of the Sync Control register. The $\overline{\text{SENSE}}$ output is 3-stated by setting the SENS DSAB bit of the Miscellaneous Control 1 register.

The remaining drivers are the microprocessor D[7:0] signals. These are normally 3-stated and will not dissipate power unless a microprocessor read is performed.

12.3 Clocking Power

Most of the digital logic power dissipation occurs as a result of clocking. The ICLK PWR, SCLK PWR, DDOT PWR, and SYNC PWR bits of the Power Management register are used to inhibit the digital logic clocking.

The ICLK PWR bit, when set, inhibits all internal clocking except for the following:

- The PLL.
- The palette arrays and the cursor array control logic. The clocks to the internal logic are left running because this is required for microprocessor access.
- SCLK and DDOTCLK - The circuitry that generates these clocks is left running in case external components need to run off these clocks.
- The horizontal and vertical sync delay circuits. These circuits are left running to allow sync signals to propagate to the display monitor.

When the ICLK PWR bit is set the DAC outputs will remain stuck at whatever was last clocked into the DACs, unless the DACs are shut down with DAC PWR.

The SCLK PWR bit may be set to disable the clocking to the SCLK generator. The resultant static SCLK output may be left at either the low or high state. As noted above, the SCLK output may be 3-stated with the SCLK DSAB bit of the Miscellaneous Clock Control register.

The DDOT PWR bit may be set to disable the clocking to the DDOTCLK generator. The resultant static DDOTCLK output may be left at either the low or high state. As noted above, the DDOTCLK output may be 3-stated with the DDOT DSAB bit of the Miscellaneous Clock Control register.

The SYNC PWR bit may be set to disable the clocking to the horizontal and vertical sync circuits. These outputs may be left at either the low or high state. (But note that the outputs can be forced high or low or 3-stated with the HSYN CNTL and VSYN CNTL bits of the Sync Control register.)

The starting and stopping of clocks with the SCLK PWR, DDOT PWR, and SYNC PWR bits is asynchronous. Thus, “chopped” pulses may be produced on the SCLK, DDOTCLK, $\overline{\text{HSYNCOUT}}$ and $\overline{\text{VSYNCOUT}}$ outputs when these bits are changed.

Similarly, changing the ICLK PWR bit can disturb the stopping and starting of the internal clocks such that the display is disturbed for a frame. It is recommended that the DACs be blanked with the BLANK CNTL bit of the Miscellaneous Control 2 register before shutting off the clocks, and that a frame be run by after turning on the clocks before the DACs are unblanked again with BLANK CNTL.

12.4 PLL Power

Each PLL uses approximately 3 mW of power. They can be shut off with the PPLL ENAB bit of the Miscellaneous Clock Control register (Pixel PLL) and the SPLL ENAB bit of the System Clock Control register (SYSCLK PLL). This, in conjunction with turning off the DACs and 3-stating the drivers, produces the lowest power consumption.

Note that in general the Pixel PLL drives SCLK, and the incoming LCLK is generally derived externally from SCLK. If the Pixel PLL is disabled, SCLK will stop running regardless of the setting of SCLK PWR, DDOTCLK will stop running regardless of the setting of DDOT PWR, internal clocking will stop regardless of the setting of ICLK PWR, sync signal clocking will stop regardless of the setting of SYNC PWR, and external circuitry running off SCLK and/or DDOTCLK will stop running.

Also, if the REFCLK input is used as the pixel clock instead of the pixel PLL, the same effect can be achieved by stopping REFCLK.

13.0 Diagnostic Support

13.1 Data Masks

The Pixel data inputs may be masked by the VRAM Mask registers to diagnose frame buffer problems. Each active bit in the VRAM Mask register controls four bits of the pixel port. Masked bytes introduce zeroes into the data path.

13.2 MISR

The MISR (Multiple Input Signature Register) consists of three registers, MISR Red, MISR Green and MISR Blue. Together the three registers form a 24-bit shift register with feedback to accumulate a signature of the data presented to the DACs during one screen frame. In interlace mode accumulation of a frame starts with an ODD field followed by an EVEN field.

Signature accumulation is controlled by the MISR CNTL bit in the Miscellaneous Control 1 register, and detection of vertical blanking. The MISR DONE bit in the MISR Status register indicates when a signature has been accumulated.

Signature accumulation uses the following sequence:

1. Write the Miscellaneous Control 1 register to change the MISR CNTL bit from ‘0’ to ‘1’.
2. The next detection of vertical blanking (as defined in section 9.3) will “arm” the MISR circuits.
3. The start of the next frame (in non-interlace mode when the $\overline{\mathrm{BLANK}}$ input goes from low to high; in interlace mode when the $\overline{\mathrm{BLANK}}$ input goes from low to high and the $\overline{\mathrm{BORDER}}/\overline{\mathrm{OE}}$ input is high to indicate an ODD field) resets the three MISR registers, resets the MISR DONE bit, and starts the accumulation of the signature.
4. A signature is accumulated for all pixel data presented to the DACs in a frame. Note that this includes border pixels and cursor pixels if they are displayed.
5. Signature accumulation stops with the next detection of vertical blanking, for non-interlace mode, or the detection of vertical blanking at the end of an EVEN field in interlace mode. At this point the three MISR registers are frozen and the MISR DONE bit changes from ‘0’ to ‘1’.

The accumulated signature can now be read from the MISR Red, MISR Green and MISR Blue registers.

When the MISR CNTL bit is changed from ‘0’ to ‘1’ it must stay at ‘1’ throughout the signature accumulation, until the MISR DONE bit changes to ‘1’. At this point, leaving the MISR CNTL bit at ‘1’, or changing it back to ‘0’, will perform no action. The MISR DONE bit will remain at ‘1’ and the three MISR registers will retain their contents.

To accumulate another signature the MISR CNTL bit must be reset to ‘0’ and then set to ‘1’. A new signature will be accumulated using the sequence listed above.

13.3 DAC Comparators

Each DAC output is connected to a comparator. Both latched and unlatched copies of the comparator outputs can be read from the DAC Sense register. The logical AND of either the latched or unlatched comparator bits is presented on the $\overline{\text{SENSE}}$ output. The reference inputs of the comparators are connected to the chip CVREF pin. With the internally applied reference voltage of 0.35 V, the corresponding Sense bit will be ‘1’b when the DAC output is 0 to 0.28 V or ‘0’b when it is 0.42 V to 0.70 V. These values apply when the DAC is doubly terminated in 75 Ω, RREF=698 Ω, and no sync or blank is present.

14.0 Internal Register - Summary

Table 15 contains a summary of the internal registers. Detailed descriptions are given in Section 15.0.

For correct operation, and to preserve upward compatibility, the registers listed as “Reserved” must not be written to. Within a register, individual bits listed as “Reserved” must be set to ‘0’s.

Table 15. Internal Register Summary

RS[2:0]	Index	R/W	Reset Value	Register Name
000	-	✓	U	Palette Address (Write Mode)
001	-	✓	U	Palette Data
010	-	✓	U	Pixel Mask
011	-	✓	U	Palette Address (Read Mode)
100	-	✓	U	Index Low
101	-	✓	U	Index High
110	-	✓	U	Index Data (Indexed Registers)
111	-	✓	U	Index Control
110	0x0000	RO	Rev	Revision Level
110	0x0001	RO	0x30	ID
110	0x0002	✓	0x01	Miscellaneous Clock Control
110	0x0003	✓	0x00	Sync Control
110	0x0004	✓	0x00	Horizontal Sync Position
110	0x0005	✓	0x00	Power Management
110	0x0006	✓	0x02	DAC Operation
110	0x0007	✓	0x00	Palette Control
110	0x0008	✓	0x01	System Clock Control
110	0x0009	-	-	(Reserved)
110	0x000a	✓	U	Pixel Format
110	0x000b	✓	U	8 BPP Control
110	0x000c	✓	U	16 BPP Control
110	0x000d	✓	U	24 BPP Packed Control
110	0x000e	✓	U	32 BPP Control
110	0x000f	-	-	(Reserved)
110	0x0010	✓	0x00	Pixel PLL Control 1
110	0x0011	✓	0x00	Pixel PLL Control 2
110	0x0012 - 0x0013	-	-	(Reserved)
110	0x0014	✓	0x05	(Fixed Pixel PLL Reference Divider)
110	0x0015	✓	0x08	SYSCLK N (System PLL Reference Divider)
110	0x0016	✓	0x41	SYSCLK M (System PLL VCO Divider)
110	0x0017	✓	U	SYSCLK P
110	0x0018	✓	U	SYSCLK C
110	0x0019 - 0x001f	-	-	(Reserved)
110	0x0020	✓	0x05	Pixel M0 (F0/M0)
110	0x0021	✓	0x0e	Pixel N0 (F1/N0)
110	0x0022	✓	0x00	Pixel P0 (F2/M1)
110	0x0023	✓	0x00	Pixel C0 (F3/N1)
110	0x0024	✓	0x00	Pixel M1 (F4/M2)
110	0x0025	✓	0x00	Pixel N1 (F5/N2)
110	0x0026	✓	0x00	Pixel P1 (F6/M3)
110	0x0027	✓	0x00	Pixel C1 (F7/N3)
110	0x0028	✓	0x00	Pixel M2 (F8/M4)
110	0x0029	✓	0x00	Pixel N2 (F9/N4)
110	0x002a	✓	0x00	Pixel P2 (F10/M5)
110	0x002b	✓	0x00	Pixel C2 (F11/N5)
110	0x002c	✓	0x00	Pixel M3 (F12/M6)
110	0x002d	✓	0x00	Pixel N3 (F13/N6)
110	0x002e	✓	0x00	Pixel P3 (F14/M7)
110	0x002f	✓	0x00	Pixel C3 (F15/N7)
110	0x0030	✓	0x00	Cursor Control
110	0x0031	✓	U	Cursor X Low
110	0x0032	✓	U	Cursor X High
110	0x0033	✓	U	Cursor Y Low
110	0x0034	✓	U	Cursor Y High
110	0x0035	✓	U	Cursor Hot Spot X
110	0x0036	✓	U	Cursor Hot Spot Y
110	0x0037	✓	0x00	Advanced Cursor Control
110	0x0038	✓	U	Advanced Cursor Attribute
110	0x0039 - 0x003f	-	-	(Reserved)
110	0x0040	✓	U	Cursor Color 1 Red
110	0x0041	✓	U	Cursor Color 1 Green
110	0x0042	✓	U	Cursor Color 1 Blue

Table 15. Internal Register Summary (Continued)

RS[2:0]	Index	R/W	Reset Value	Register Name
110	0x0043	✓	U	Cursor Color 2 Red
110	0x0044	✓	U	Cursor Color 2 Green
110	0x0045	✓	U	Cursor Color 2 Blue
110	0x0046	✓	U	Cursor Color 3 Red
110	0x0047	✓	U	Cursor Color 3 Green
110	0x0048	✓	U	Cursor Color 3 Blue
110	0x0049 - 0x005f	-	-	(Reserved)
110	0x0060	✓	U	Border Color Red
110	0x0061	✓	U	Border Color Green
110	0x0062	✓	U	Border Color Blue
110	0x0063 - 0x0067	-	-	(Reserved)
110	0x0068	✓	U	Key
110	0x0069 - 0x006b	-	-	(Reserved)
110	0x006c	✓	U	Key Mask
110	0x006d - 0x006f	-	-	(Reserved)
110	0x0070	✓	0x00	Miscellaneous Control 1
110	0x0071	✓	0x00	Miscellaneous Control 2
110	0x0072	✓	0x00	Miscellaneous Control 3
110	0x0073 - 0x0077	-	-	(Reserved)
110	0x0078	✓	0x00	Key Control/DB Operation
110	0x0079 - 0x0081	-	-	(Reserved)
110	0x0082	RO	U	DAC Sense
110	0x0083	RO	0x00	MISR Status
110	0x0084	RO	U	MISR Red
110	0x0085	-	-	(Reserved)
110	0x0086	RO	U	MISR Green
110	0x0087	-	-	(Reserved)
110	0x0088	RO	U	MISR Blue
110	0x0089 - 0x008b	-	-	(Reserved)
110	0x008c	RO	U	Pixel P Input
110	0x008d	RO	U	Pixel C Input
110	0x008e	RO	0x05 (FS[1:0] = 00) 0x0e (FS[1:0] = 01) 0x00 (FS[1:0] = 10 or 11)	Pixel M Input (Pixel PLL VCO Divider Input)
110	0x008f	RO	0x05	Pixel N Input (Pixel PLL Reference Divider Input)
110	0x0090	✓	U	VRAM Mask Low
110	0x0091	✓	U	VRAM Mask High
110	0x0092 - 0x009f	-	-	(Reserved)
110	0x00a0	✓	U	K1 - YUV:RGB
110	0x00a1	✓	U	K2 - YUV:RGB
110	0x00a2	✓	U	K3 - YUV:RGB
110	0x00a3	✓	U	K4 - YUV:RGB
110	0x00a4 - 0x00ff	-	-	(Reserved)
110	0x0100 - 0x04ff	✓	U	Cursor Array
110	0x0500 0x07ff	-	-	(Reserved)

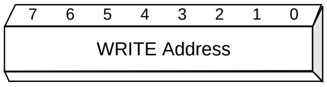
RO=Read Only, U=Undefined, Rev=Revision Level

15.0 Register Descriptions

15.1 Direct Access Registers

The direct access registers are addressed using RS[2:0] inputs.

Palette Address (Write Mode)



RS[2:0]: 000

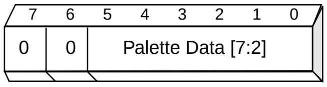
Access: Read/Write

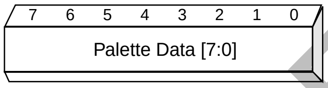
Power on Value: Undefined

Bits 7 - 0 WRITE Address - Palette address in write mode.

Operation of this register is discussed in 1.0 “Microprocessor Access” on page 1.

Palette Data





RS[2:0]: 001

Access: Read/Write

Power on Value: Undefined

The format of the palette data depends on the color resolution, 6 or 8 bit.

6 bit color resolution
Miscellaneous Control 2 COL RES = 0

Bits 7 - 6 00

Bits 5 - 0 6 bit palette data

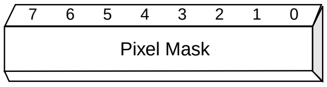
On WRITEs bits 7:6 from the microprocessor are discarded, bits 5:0 are written to bits 7:2 internally, and internal bits 1:0 are set to ‘00’. On reads internal bits 7:2 are read as bits 5:0, an bits 7:6 are returned as ‘00’.

8 bit color resolution
Miscellaneous Control 2 COL RES = 1

Bits 7-0 8 bit palette data. Bits 7:0 are written/read internally as bits 7:0

Operation of this register is discussed in 1.0 “Microprocessor Access” on page 1.

Pixel Mask



RS[2:0]: 010

Access: Read/Write

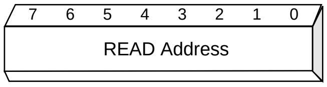
Power on Value: Undefined

Bits 7 - 0 Pixel Mask

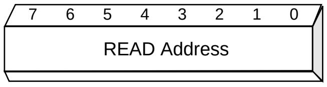
In indirect color modes this register masks the pixel values used to index into the palettes. Each bit is ANDed with its corresponding pixel bit. A value of 0xff is required to pass the pixel values to the palettes unchanged

The same mask is applied to each of the red, green, and blue pixel addresses into the palettes.

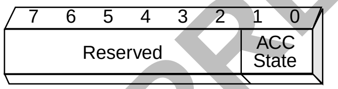
Palette Address (Read Mode) / Palette Access State



RS[2:0]: 011
Access: Write
Power on Value: -
Bits 7 - 0 READ Address - Palette address in read mode.



RS[2:0]: 011
Access: Read
Power on Value: Undefined
PADR RFMT: 0
Bits 7 - 0 READ Address - Palette address in read mode.

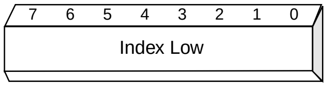


RS[2:0]: 011
Access: Read
Power on Value: Undefined
PADR RFMT: 1
Bits 7 - 2 Reserved
Bits 1 - 0 ACC STATE - Palette Access State. Reports which mode was used on last write of Palette Address Register.

00 Write Mode
11 Read Mode

Note that the palette address to be read is written into this register, but the contents that are read depends on the PADR RFMT bit in the Miscellaneous Control 1 register. Operation of these registers is discussed in 1.0 "Microprocessor Access" on page 1.

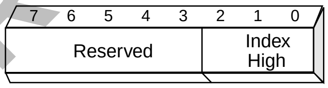
Index Low



RS[2:0]: 100
Access: Read/Write
Power on Value: Undefined
Bits 7 - 0 Index Low

This register, together with Index High, forms the internal index register. It selects the register that will be accessed when the Indexed Data register is written or read.

Index High

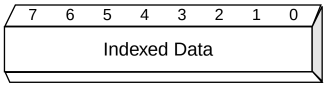


RS[2:0]: 101
Access: Read/Write
Power on Value: Undefined
Bits 7 - 3 Reserved
Bits 2 - 1 Index High

This register provides the high-order bits of the internal index register.

If auto-increment is turned on, the resulting index is not defined if an increment past the maximum index value occurs.

Indexed Data



RS[2:0]: 110
Access: Read/Write
Power on Value: Undefined
Bits 7 - 0 Indexed Data

A write or read to this register will write or read the register addressed by the internal index register (Index High and Index Low).

Following a write or read to Indexed Data, the index register will be incremented if auto-increment is turned on (INDX CNTL bit of the Index Control register).

Index Control

7	6	5	4	3	2	1	0
Reserved							INDX CNTL

RS[2:0]: 111

Access: Read/Write

Power on Value: Undefined

Bits 7 - 1 Reserved

Bit 0 INDX CNTL - Index Control. Controls auto-increment of the index register.

0 Off - no auto-increment.

1 On - the index register (Index High and Index Low) will increment by one following a write or read to Indexed Data.

15.2 Indexed Registers

The indexed registers are accessed by setting the desired address into the internal index register (Index High and Index Low) and writing or reading the Indexed Data register.

15.2.1 Miscellaneous Control

Miscellaneous Control 1

7	6	5	4	3	2	1	0
MISR CNTL	VMSK CNTL	PADR RFMT	SENS DSAB	SENS SEL	XOR SYNC	RSVD	VRAM SIZE

Index: 0x0070

Access: Read/Write

Power on Value: 0x00

Bit 7 MISR CNTL

0 Off. If the MISR is running, it will stop at the beginning of the next frame.

1 On. The MISR will start accumulating a signature at the start of the next frame (end of vertical blanking).

Bit 6 VMSK CNTL - VRAM Mask Control

0 No VRAM masking.

1 The VRAM inputs on the PIX[63:00] inputs will be masked under control of the VRAM Mask High and VRAM Mask Low registers.

This bit has no effect when the VGA port is selected.

Bit 5 PADR RFMT - Palette Address Register (Read Mode) Format. Specifies the contents returned from the Palette Address register, read mode (RS[2:0] = 011)

0 Return the eight bits of the read address

1 Return the palette access state in the two low order bits

Bit 4 SENS DSAB -$\overline{\text{SENSE}}$ Driver Disable

0 $\overline{\text{SENSE}}$ driver enabled

1 $\overline{\text{SENSE}}$ driver disabled (3-stated)

Bit 3 SENS SEL - Sense Select. Selects which bit of the DAC Sense register is presented on the $\overline{\text{SENSE}}$ driver.

0 Bit 3 - Unlatched Sense

1 Bit 7 - Latched Sense

Bit 2 XOR SYNC - Composite Sync Source. Selects source of composite sync.

0 Composite sync taken from $\overline{\text{HCSYNCIN}}$ input.

1 Composite sync generated internally from XOR of $\overline{\text{HCSYNCIN}}$ input and $\overline{\text{VSYNCIN}}$ input.

Bit 1 RSVD - Reserved

Bit 0 VRAM SIZE - VRAM interface width

0 32 bits. PIX[31:0] used, PIX[63:32] ignored.

1 64 bits. PIX[63:00] used.

(This bit has no effect when the VGA port is selected.)

Miscellaneous Control 2

7	6	5	4	3	2	1	0
PCLK SEL		INTL MODE	BLANK CNTL	RSVD	COL RES	BRDR/ INTL	PORT SEL

Index: 0x0071

Access: Read/Write

Power on Value: 0x00

Bits 7 - 6 PCLK SEL - Pixel Clock Select. Specifies the source of the internal pixel clock.

00 LCLK input

01 Internal PLL output

10 REFCLK input

11 Reserved

Note: A selection of 00 (LCLK input) for the pixel clock is required and only valid when PORT SEL = 0 (VGA data inputs), or 32 BPP is selected with a VRAM width of 32.

Bit 5 INTL MODE - Interlace Mode. Controls effect of $\overline{\text{BORDER}}$/$\overline{\text{OE}}$ input on cursor when this input is used as the ODD/$\overline{\text{EVEN}}$ interlace control.

0 Non-interlaced. The $\overline{\text{BORDER}}$/$\overline{\text{OE}}$ input is ignored.

1 Interlaced. If the cursor is turned on, the $\overline{\text{BORDER}}$/$\overline{\text{OE}}$ input will be used to select display of the odd or even cursor rows.

This bit has no effect when BRDR/INTL (bit 1) is set '0' ($\overline{\text{BORDER}}$/$\overline{\text{OE}}$ used as $\overline{\text{BORDER}}$ input).

Bit 4 BLANK CNTL - Blanking Control

0 Normal operation.

1 DACs are blanked. No pixel data is presented on the DACs, but all other operations remain normal, including the collection of a signature if the MISR is turned on.

Bit 3 RSVD - Reserved

Bit 2 COL RES - Color Resolution

0 6-bit

1 8-bit

With 6-bit color resolution only 6 bits of microprocessor data are loaded into the palettes. Microprocessor data bits D[5:0] are written to/read from palette bits [7:2]. Internally 00 is written to palette bits [1:0], and on reads D[7:6] are forced to 00.

Also with 6-bit color resolution the two low order bits presented from the palettes to the DACs are controlled by Palette Control bit 6BIT LIN.

With 8-bit color resolution all 8 bits from/to the microprocessor are written/read to the palette, and the 8 bits presented to the DACs are unmodified. The 6BIT LIN bit has no effect.

Bit 1 BRDR/INTL - Border/Interlace. Controls usage of $\overline{\text{BRDR}}/\overline{\text{OE}}$ input.

0 $\overline{\text{BORDER}}/\overline{\text{OE}}$ input used to indicate "$\overline{\text{BORDER}}$". (Interlace operation is not supported.)

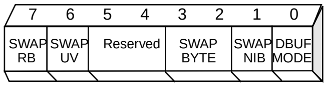
1 $\overline{\text{BORDER}}/\overline{\text{OE}}$ input used to indicate "ODD/$\overline{\text{EVEN}}$". (Border operation is not supported.)

Bit 0 PORT SEL - Port Select

0 VGA Data inputs.

1 VRAM pixel port inputs.

Miscellaneous Control 3



Index: 0x0072

Access: Read/Write

Power on Value: 0x00

Bit 7 SWAP RB - For RGB pixels, swap Red and Blue pixel components. In 16, 24, and 32 BPP, this bit causes the red and blue components of the pixels to be swapped. In indirect mode, the swapping takes place before the Palette.

0 Normal operation.

1 Swap Red and Blue components of the pixel.

This bit only has an effect in 16, 24,and 32 BPP on RGB pixels.

Bit 6 SWAP UV - For YUV pixels, swap U and V pixel components. For the 4:4:4 format the U and V values are swapped within the pixel. For the 4:2:2 format the U and V values are swapped between pixel pairs.

0 Order is Y,U,V.

1 Order is Y,V,U.

This bit only has an effect in 16, 24,and 32 BPP on YUV pixels.

Bits 5 - 4 Reserved

Bits 3 - 2 SWAP BYTE - Controls swapping of the incoming bytes on the PIX[63:00] inputs.

00 Normal operation.

01 Reserved

10 Reserved

11 For each pair of bytes on the PIX[63:00] inputs, swap the incoming bytes.

When byte swapping is enabled the swapping is in effect and is identical for all VRAM pixel formats (4, 8 16, 24 or 32 BPP).

Bit 1 SWAP NIB - Swap nibbles within bytes. Used with 4 BPP.

0 Use high nibble (e.g., PIX[07:04]) for first pixel, use low nibble (e.g., PIX[03:00]) for next pixel.

1 Use low nibble (e.g., PIX[03:00]) for first pixel, use high nibble (e.g., PIX[07:04]) for next pixel.

The same nibble order is applied to each of the incoming bytes. This bit has no effect if the pixel format is not 4 BPP.

Bit 0 DBUF MODE - Double Buffer Mode Enable. Used with 16 BPP.

This register has no effect when the VGA port is selected.

Miscellaneous Clock Control

7	6	5	4	3	2	1	0
DDOT DSAB	SCLK DSAB	B24P DDOT	SCLK INVT	DDOT DIV			PPLL ENAB

Index: 0x0002

Access: Read/Write

Power on Value: 0x01

Bit 7 DDOT DSAB - DDOTCLK driver disable

DDOTCLK driver enabled

0 DDOTCLK driver enabled

1 DDOTCLK driver disabled (3-stated)

Bit 6 SCLK DSAB - SCLK driver disable

0 SCLK driver enabled

1 SCLK driver disabled (3-stated)

Bit 5 B24P DDOT - Selects which clock is driven on DDOTCLK when 24 Bit Packed Pixel format is selected.

0 Use divided Pixel PLL output under control of DDOT DIV bits.

1 Output the same signal as SCLK.

When a format other than 24 BPP Packed is selected, the B24P DDOT bit has no effect and the divided Pixel PLL output is used.

Bit 4 SCLK INVT - Inverts the SCLK output.

Bits 3 - 1 DDOT DIV - DDOTCLK divide factor. Specifies the divide factor applied to the internal Pixel PLL output to produce the DDOTCLK output signal.

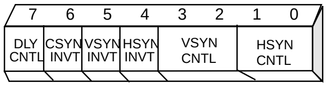
000	Pixel PLL out/1
001	Pixel PLL out/2
010	Pixel PLL out/4
011	Pixel PLL out/8
100	Pixel PLL out/16
101	Reserved
110	Reserved
111	Reserved

Bit 0 PPLL ENAB - Pixel PLL Enable

0 Pixel PLL programming disabled.

1 Pixel PLL programming enabled.

Sync Control



Index: 0x0003

Access: Read/Write

Power on Value: 0x00

Bit 7 DLY CNTL - Sync Delay Control. Specifies whether delay matching the pixel pipeline delay should be added to the horizontal sync signal.

0 Add matching delay

1 Do not add delay

This bit only has effect when the SOG bit of the DAC Operation register is off. If SOG is on (composite sync enabled) then matching pipeline delay is not added to horizontal sync.

Bit 6 CSYN INVT - Invert incoming $\overline{\text{HCSYNCIN}}$ when used as Composite Sync

0 Do not invert incoming $\overline{\text{HCSYNCIN}}$

1 Invert incoming $\overline{\text{HCSYNCIN}}$

Bit 5 VSYN INVT - Vertical Sync Invert

0 Do not invert incoming $\overline{\text{VSYNCIN}}$

1 Invert incoming $\overline{\text{VSYNCIN}}$

Bit 4 HSYN INVT - Invert incoming $\overline{\text{HCSYNCIN}}$ when used as Horizontal Sync

0 Do not invert incoming $\overline{\text{HCSYNCIN}}$

1 Invert incoming $\overline{\text{HCSYNCIN}}$

Bits 3 - 2 VSYN CNTL - Vertical Sync Output Control

00 Normal output

01 $\overline{\text{VSYNCOUT}}$ forced high

10 $\overline{\text{VSYNCOUT}}$ forced low

11 $\overline{\text{VSYNCOUT}}$ disabled (3-stated)

Bits 1 - 0 HSYN CNTL - Horizontal Sync Output Control

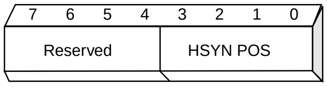
00 Normal output

01 $\overline{\text{HSYNCOUT}}$ forced high

10 $\overline{\text{HSYNCOUT}}$ forced low

11 $\overline{\text{HSYNCOUT}}$ disabled (3-stated)

Horizontal Sync Control



Index: 0x0004

Access: Read/Write

Power on Value: 0x00

Bits 7 - 4 Reserved

Bits 3 - 0 HSYN POS - Horizontal Sync Position. Specifies number of additional pixel delays to add to the horizontal sync signal and the composite sync signal.

0000	0 pixels
0001	1 pixel
0010	2 pixels
0011	3 pixels
0100	4 pixels
0101	5 pixels
0110	6 pixels
0111	7 pixels
1000	8 pixels
1001	9 pixels
1010	10 pixels
1011	11 pixels
1100	12 pixels
1101	13 pixels
1110	14 pixels
1111	15 pixels

If the SOG bit of the DAC Operation register is on, the additional pixel delays are added to composite sync. If SOG is off, then the delays are added to horizontal sync, under the control of the DLY CNTL bit of the Sync Control register. (The additional delay specified by the Horizontal Position register will only be added if DLY CNTL is set to 0.)

Power Management

7	6	5	4	3	2	1	0
Reserved			SCLK PWR	DDOT PWR	SYNC PWR	ICLK PWR	DAC PWR

Index: 0x0005

Access: Read/Write

Power on Value: 0x00

Bits 7 - 5 Reserved

Bit 4 SCLK PWR - SCLK Power Control

0 Normal Operation

1 Disable clocks to SCLK generator

Bit 3 DDOT PWR - DDOTCLK Power Control

0 Normal Operation

1 Disable clocks to DDOTCLK generator

Bit 2 SYNC PWR - Sync Power Control

0 Normal Operation

1 Disable clocks to horizontal and vertical sync circuits

Bit 1 ICLK PWR - Internal Clock Power Control

0 Normal Operation

1 Disable all internal clocks except those for the SCLK generator, DDOTCLK generator, and horizontal and vertical sync circuits.

A clock is left running to allow microprocessor access of the palette and cursor array, but otherwise the palette and cursor RAMs will not be clocked. The two RAMs will retain their contents.

Bit 0 DAC PWR - DAC Analog Power Control

0 Normal Operation

1 Disable analog power to the DACs

DAC Operation

7	6	5	4	3	2	1	0
Reserved				SOG	BRB	DSR	DPE

Index: 0x0006

Access: Read/Write

Power on Value: 0x02

Bits 7 - 4 Reserved

Bit 3 SOG - Composite Sync-On-Green

0 Sync is disabled on Green DAC.

1 Sync is enabled on Green DAC.

Bit 2 BRB - Blank Red and Blue DACs

0 Red and Blue DACs have normal function.

1 Red and Blue DACs are always blanked.

Bit 1 DSR - DAC Slew Rate

0 Fast - typically 2 ns

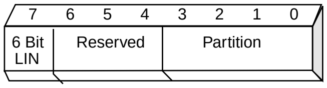
1 Slow - typically 5 ns

Bit 0 DPE - DAC blanking Pedestal Enable

0 Blanking pedestal disabled (0 IRE)

1 Blanking pedestal enabled (7.5 IRE)

Palette Control



Index: 0x0007

Access: Read/Write

Power on Value: 0x00

Bit 7 6BIT LIN - 6 Bit Linear Color

0 Apply linear palette output

1 Do not apply linear palette output

This bit only has effect with indirect color modes, and when Color Resolution is set to 6 bits (Miscellaneous Control 2 COL RES bit = 0). For the 8 bits of palette output for each color, the high order two bits 7 and 6 will be substituted for the two low order bits 1 and 0.

Bits 6 - 4 Reserved.

Bits 3 - 0 PALETTE PARTITION - Selects which partition to use within the palettes when the pixel format is either 4 BPP, 15 BPP indirect color, or 16 BPP indirect color.

With 4 BPP the palettes are divided into 16 partitions. Each partition contains 16 entries. Bits 3 - 0 select 1 of the 16 partitions.

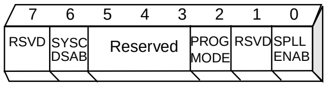
With 15 BPP (555) indirect color, the palettes are divided into 8 partitions. Each partition contains 32 entries. Bits 3 - 1 select 1 of the 8 partitions and bit 0 is not used.

With 16 BPP (565) indirect color, the palettes are divided into 4 partitions. Each partition contains 64 entries. All 64 entries of the Green palette are used in each partition. For the Red and Blue palettes only the first 32 entries of each partition are used. Bits 3 - 2 select 1 of the 4 partitions and bits 1 and 0 are not used.

The PARTITION bits have no effect when the pixel format is not 4 BPP, 15 BPP, or 16 BPP. Also, with 15 BPP and 16 BPP the PARTITION bits have no effect unless

1. Indirect color is chosen (16 BPP Control register bits B16 DCOL = 00), AND
2. Contiguous addressing is chosen (16 BPP Control register bit SPR/CNT = 1)

System Clock Control



Index: 0x0008

Access: Read/Write

Power on Value: 0x01

Bit 7 RSVD - Reserved

Bit 6 SYSC DSAB - SYSCLK driver disable

0 SYSCLK driver enabled

1 SYSCLK driver disabled (3-stated)

Bits 5 - 3 Reserved

Bit 2 PROG MODE - SYSCLK PLL programming mode

0 Compatibility mode - program with registers at 0x0015 and 0x0016 as described in Appendix D.0

1 Standard mode - program with N, M, P and C (registers at 0x0016, 0x0017, 0x0018, 0x0019) as described in Section 2.7

Bit 1 Reserved

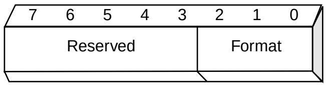
Bit 0 SPLL ENAB - System PLL enable

0 SYSCLK PLL programming disabled

1 SYSCLK PLL programming enabled

15.2.2 Pixel Representation

Pixel Format



Index: 0x000a

Access: Read/Write

Power on Value: Undefined

Bits 7 - 3 Reserved

Bits 2 - 0 Pixel Format

000 Reserved

001 Reserved

010 4 BPP

011 8 BPP

100 15/16 BPP

101 24 BPP Packed

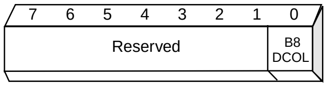
110 32 BPP

111 Reserved

This register has no effect when the VGA port is selected.

The 24 BPP Packed format requires the VRAM SIZE (Miscellaneous Control 1 register bit 0) to be set for 64 bits. If VRAM SIZE is set to 32 bits, the product operation will be undefined if the 24 BPP Packed format is selected.

8 Bit Pixel Control



Index: 0x000b

Access: Read/Write

Power on Value: Undefined

Bits 7 - 1 Reserved

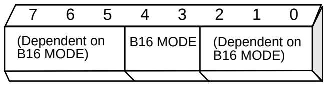
Bit 0 B8 DCOL - 8 BPP Direct Color Control

0 Indirect Color (through the palette).

1 Direct Color (palette bypass). Since the same 8-bit value will be applied to each of the Red, Green, and Blue DACs a monochrome image will be displayed.

This register only affects 8 BPP mode.

16 Bit Pixel Control



Index: 0x000c

Access: Read/Write

Power on Value: Undefined

Bits 4 - 3 B16 MODE

00 RGB - 15 or 16 bits of RGB (555 or 565).

01 Tagged RGB/YUV 8 + 7
The high order pixel bit is a tag to indicate that the remaining 15 pixel bits are RGB (555) or YUV (8 bits of Y and 7 bits of subsampled U/V, or 8 bits of subsampled U/V and 7 bits of Y).

10 YUV - 16 bit subsampled YUV (8 bits Y and 8 bits subsampled U/V).

11 Tagged RGB/YUV 7 + 8
The high order pixel bit is a tag to indicate that the remaining 15 pixel bits are RGB (555) or YUV (7 bits of Y and 8 bits of subsampled U/V, or 7 bits of subsampled U/V and 8 bits of Y).

This register only affects 15/16 BPP mode.

Note: For the RGB624DB the YUV pixel formats (B16 MODE = ‘01’, ‘10’ and ‘11’) are not a valid option when double buffer operation is enabled.

16 BPP - RGB

7	6	5	4	3	2	1	0
B16 DCOL		BY16 POL	00		ZIB/ LIN	555/ 565	SPR/ CNT

Bits 7 - 6 B16 DCOL - 16 BPP Direct Color Control

00 Indirect Color (always goes through the palette). Either the 555 or 565 format can be selected. The SPR/CNT bit determines if the access of the palettes is sparse or contiguous. If CNT (contiguous), then the PARTITION bits of the Palette Control register determine which partition of the palettes is used. If SPR (Sparse) the ZIB/LIN bit must be set to 0 (ZIB).

01 Dynamic Bypass. The high order bit of each 16-bit pixel (PIX[15], PIX[31], PIX[47], PIX[63].) is used to select on a pixel-by-pixel basis to either go through the palette (indirect color) or bypass the palette (direct color). When this mode is selected the following conditions apply:

1. The 555/565 bit has no effect. Internally, the pixel format is forced to 5 bits per color (555).
2. The SPR/CNT bit has no effect. Internally, sparse addressing (SPR) is forced for palette access.
3. The ZIB/LIN bit has no effect. Internally, the low order bits for each color are forced to '0's (ZIB) for both access of the palette (indirect color) and palette bypass (direct color).
4. The Pixel Mask is applied to the pixel data regardless of whether or not the palette is bypassed.

Note: For the RGB624DB Dynamic Bypass is not a valid option when double buffer operation is enabled.

10 Reserved

11 Direct Color (always bypasses the palette). Either the 555 or 565 format can be selected. The ZIB/LIN bit determines the expansion to 24 bits (low order bit fill). The SPR/CNT bit has no effect.

Bit 5 BY16 POL - Bypass control bit polarity. Determines the meaning of the dynamic bypass control bit (PIX[15], PIX[31], PIX[47], PIX[63]).

0 Control Bit Forces Bypass

Control Bit	***Pixel Path***
0	Through Palette (Indirect Color)
1	Bypass Palette (Direct Color)

1 Control Bit Forces Lookup

Control Bit	***Pixel Path***
0	Bypass Palette (Direct Color)
1	Through Palette (Indirect Color)

The BY16 POL bit has no effect unless the B16 DCOL bits are set to 01.

Bits 4 - 3 00

Bit 2 ZIB/LIN - Bit fill selection. For direct color this bit specifies how the low order bits of each color, R,G,B are filled when 555 or 565 formats are expanded to 24 bits.

0 ZIB - Zero Intensity Black. Low order bits are set to 0.

1 LIN - Linear. The low order bits are set to the values of the high order bits.

For indirect color if CNT (contiguous) addressing is selected, then ZIB/LIN has no effect. If SPR (sparse) addressing is selected then ZIB/LIN **must** be set to 0 (ZIB). The palette addressing is undefined if LIN bit fill is selected with sparse addressing.

Bit 1 555/565 - Selects 5 bits per color (555) or 5 red, 6 green, 5 blue (565).

0	555
1	565

Note: for the RGB624DB this bit must be set for 555 format, when double buffer operation is enabled.

Bit 0 SPR/CNT - Sparse/Contiguous. In indirect mode, selects whether index into palette is sparse or contiguous.

0 Sparse

1 Contiguous

16 BPP - Tagged RGB/YUV 8 + 7

7	6	5	4	3	2	1	0
RGB DCOL		TAG POL	01		ZIB/ LIN	SWAP YC	SPR/ CNT

Bits 7 - 6 RGB DCOL - RGB Direct Color Control. Determines path to DACs for pixels tagged as RGB.

00 Indirect Color (always goes through the palette). The SPR/CNT bit determines if the access of the palettes is sparse or contiguous. If CNT (contiguous), then the PARTITION bits of the Palette Control register determine which partition of the palettes is used. If SPR (Sparse) the ZIB/LIN bit must be set to 0 (ZIB).

01 Reserved

10 Reserved

11 Direct Color (always bypasses the palette). The ZIB/LIN bit determines the expansion to 24 bits (low order bit fill). The SPR/CNT bit has no effect.

Bits 7-6 have no effect on pixels tagged as YUV.

Bit 5 TAG POL - Tag bit polarity. Determines the meaning of the tag bit (PIX[15], PIX[31], PIX[47], PIX[63]).

0 Tag indicates YUV

Tag Bit	***Pixel***
0	RGB
1	YUV

1 Tag indicates RGB

Tag Bit	***Pixel***
0	YUV
1	RGB

Bits 4 - 3 01

Bit 2 RGB ZIB/LIN - Bit fill selection for pixels tagged as RGB. For direct color this bit specifies how the low order bits of each color, R,G,B are filled when 555 or 565 formats are expanded to 24 bits.

0 ZIB - Zero Intensity Black. Low order bits are set to 0.

1 LIN - Linear. The low order bits are set to the values of the high order bits.

For indirect color if CNT (contiguous) addressing is selected, then ZIB/LIN has no effect. If SPR (sparse) addressing is selected then ZIB/LIN **must** be set to 0 (ZIB). The palette addressing is undefined if LIN bit fill is selected with sparse addressing.

Bit 2 has no effect on pixels tagged as YUV.

Bit 1 SWAP YC - Swap Luminance/Chrominance. For the Y (luminance and subsampled U/V (chrominance) values, this bit determines which value is in the high 8 bits and low 7 bits.

0 High 8 bits = Y
Low 7 bits = U/V

1 High 8 bits = U/V
Low 7 bits = Y

Bit 1 has no effect on pixels tagged as RGB.

Bit 0 RGB SPR/CNT - Sparse/Contiguous addressing for RGB pixels. In indirect mode, this bit selects whether indexing into palette is sparse or contiguous.

0 Sparse

1 Contiguous

Bit 0 has no effect on pixels tagged as YUV.

16 BPP - YUV

7	6	5	4	3	2	1	0
RESERVED		B8 DCOL	10		RSVD	SWAP YC	RSVD

Bits 7 - 6 Reserved

Bit 5 B8 DCOL - When key matching is enabled this bit determines the path of the 8 bit overlay pixel to the DACs:

0 Indirect (color lookup through the palette).

1 Direct (the 8 bits are sent to each of the red, green and blue DACs to yield grey scale).

If key matching is not enabled then Bit 5 has no effect.

Bits 4 - 3 10

Bit 2 Reserved.

Bit 1 SWAP YC - Swap Luminance/Chrominance. For the Y (luminance and subsampled U/V (chrominance) values, this bit determines which value is in the high 8 bits and low 8 bits.

0 High 8 bits = Y
Low 8 bits = U/V

1 High 8 bits = U/V
Low 8 bits = Y

Bit 0 Reserved.

16 BPP - Tagged RGB/YUV 7 + 8

7	6	5	4	3	2	1	0
RGB DCOL		TAG POL	11		ZIB/ LIN	SWAP YC	SPR/ CNT

Bits 7 - 6 RGB DCOL - RGB Direct Color Control. Determines path to DACs for pixels tagged as RGB.

00 Indirect Color (always goes through the palette). The SPR/CNT bit determines if the access of the palettes is sparse or contiguous. If CNT (contiguous), then the PARTITION bits of the Palette Control register determine which partition of the palettes is used. If SPR (Sparse) the ZIB/LIN bit must be set to 0 (ZIB).

01 Reserved

10 Reserved

11 Direct Color (always bypasses the palette). The ZIB/LIN bit determines the expansion to 24 bits (low order bit fill). The SPR/CNT bit has no effect.

Bits 7-6 have no effect on pixels tagged as YUV.

Bit 5 TAG POL - Tag bit polarity. Determines the meaning of the tag bit (PIX[15], PIX[31], PIX[47], PIX[63]).

0 Tag indicates YUV

Tag Bit	***Pixel***
0	RGB
1	YUV

1 Tag indicates RGB

Tag Bit	***Pixel***
0	YUV
1	RGB

Bits 4 - 3 11

Bit 2 RGB ZIB/LIN - Bit fill selection for pixels tagged as RGB. For direct color this bit specifies how the low order bits of each color, R,G,B are filled when 555 or 565 formats are expanded to 24 bits.

0 ZIB - Zero Intensity Black. Low order bits are set to 0.

1 LIN - Linear. The low order bits are set to the values of the high order bits.

For indirect color if CNT (contiguous) addressing is selected, then ZIB/LIN has no effect. If SPR (sparse) addressing is selected then ZIB/LIN **must** be set to 0 (ZIB). The palette addressing is undefined if LIN bit fill is selected with sparse addressing.

Bit 2 has no effect on pixels tagged as YUV.

Bit 1 SWAP YC - Swap Luminance/Chrominance. For the Y (luminance and subsampled U/V (chrominance) values, this bit determines which value is in the high 7 bits and low 8 bits.

0 High 7 bits = Y
Low 8 bits = U/V

1 High 7 bits = U/V
Low 8 bits = Y

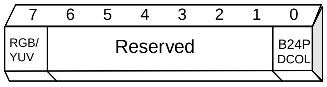
Bit 1 has no effect on pixels tagged as RGB.

Bit 0 RGB SPR/CNT - Sparse/Contiguous addressing for RGB pixels. In indirect mode, this bit selects whether indexing into palette is sparse or contiguous.

0 Sparse
1 Contiguous

Bit 0 has no effect on pixels tagged as YUV.

24 Bit Packed Pixel Control



Index: 0x000d

Access: Read/Write

Power on Value: Undefined

Bit 7 RGB/YUV

0 RGB Mode. The 24 pixel bits are treated as eight bits each of red, green and blue. Bit 0 determines if the palette is used or bypassed.

1 YUV Mode. The 24 pixel bits are treated as eight bits each of Y, U and V. These values are converted to red, green and blue and sent directly to the DACs. The palette is not used, and bit 0 has no effect.

Bits 6 - 1 Reserved

Bit 0 B24P DCOL - 24 BPP Packed Direct Color Control

0 Indirect Color (through the palette).

1 Direct Color (palette bypass).

This bit only has effect when bit 7 is 0 (RGB mode).

This register only affects 24 BPP Packed mode.

32 Bit Pixel Control

7	6	5	4	3	2	1	0
RGB/ YUV	B8 DCOL	Reserved			BY32 POL	B32 DCOL	

Index: 0x000e

Access: Read/Write

Power on Value: Undefined

Bit 7 RGB/YUV

0 RGB Mode. The 24 low order pixel bits are treated as eight bits each of red, green and blue. The bits are sent directly to the DACs or through the palette, under control of bits 1 and 0.

If key matching is enabled the 8 high order pixel bits are used as an overlay pixel, and the 24 low order bits are used as the underlay.

If key matching is not enabled then the dynamic bypass mode is available, under control of bits 2-0.

1 YUV Mode. The 24 low order pixel bits are treated as eight bits each of Y, U and V. These values are converted to red, green and blue and sent directly to the DACs.

Bits 2- 0 have no effect. If key matching is enabled the 8 high order pixel bits are used as an overlay pixel, and the YUV bits are used as the underlay.

Bit 6 B8 DCOL - When key matching is enabled this bit determines the path of the 8 bit overlay pixel to the DACs:

0 Indirect (color lookup through the palette).

1 Direct (the 8 bits are sent to each of the red, green and blue DACs to yield grey scale).

If key matching is not enabled then Bit 6 has no effect.

Bit 5 - 3 Reserved

Bit 2 BY32 POL - Bypass control bit polarity. Determines the meaning of the dynamic bypass control bit (PIX[24], PIX[56]).

0 Control Bit Forces Bypass

Control Bit	***Pixel Path***
0	Through Palette (Indirect Color)
1	Bypass Palette (Direct Color)

1 Control Bit Forces Lookup

Control Bit	***Pixel Path***
0	Bypass Palette (Direct Color)
1	Through Palette (Indirect Color)

The BY32 POL bit has no effect unless the RGB/YUV bit is set to RGB, key matching is not enabled, and the B32 DCOL bits are set to 01.

Bits 1 - 0 B32 DCOL - 32 BPP RGB Direct Color Control. Used to select path to DACs when RGB/YUV is set to RGB mode:

00 Indirect Color (always goes through the palette). 24 bits (8 bits each for Red, Green, Blue) are used to index into the palettes. The 8 high order bits (PIX[31:24], PIX[63:56]) are not used.

01 Dynamic Bypass (Key Match not enabled).

A control bit in the high order byte (PIX[24], PIX[56]) is used to select on a pixel-by-pixel basis to either go through the palette (indirect color) or bypass the palette (direct color). The remaining bits in the high order byte (PIX[31:25], PIX[63:57]) are not used. In this mode, the Pixel Mask is applied to the pixel data regardless of whether or not the palette is bypassed.

01 Reserved (Key Match enabled).

10 Reserved

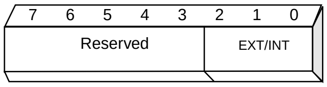
11 Direct Color (always bypasses the palette). 24 bits (8 bits each for Red, Green, Blue) are presented to the DACs. The 8 high order bits (PIX[31:24], PIX[63:56] are not used.

Bits 1-0 have no effect when RGB/YUV is set to YUV.

This register only affects 32 BPP mode.

15.2.3 Pixel Clock Frequency Selection

Pixel PLL Control 1



Index: 0x0010

Access: Read/Write

Power on Value: 0x00

Bits 7 - 3 Reserved

Bits 2 - 0 EXT/INT

Determines the programing mode (standard or compatibility), and the source and selection for the Pixel PLL programming registers.

000 Compatibility mode:
External FS[1:0] inputs

One of the F0 - F3 registers is selected with external signals FS[1:0]. The selected register provides the Pixel PLL VCO divider value. The Fixed Pixel PLL Reference Divider register is used to pre-scale the PLL reference clock.

001 Compatibility mode:
External FS[1:0] inputs (4 value M/N programming)

One of four pairs of registers M0/N0, M1/N1, M2/N2, M3/N3 are selected with external signals FS[1:0] to provide the VCO divider/reference divider inputs to the Pixel PLL.

The Fixed PLL Reference Divider register is not used.

010 Compatibility mode:
Pixel PLL Control 2 register bits [3:0] (16 value direct programming)

One of the F0 - F15 registers is selected with Pixel PLL Control 2 register bits [3:0]. The selected register provides the Pixel PLL VCO divider value. The Fixed Pixel PLL Reference Divider register is used to prescale the Pixel PLL reference clock.

011 Compatibility mode:
Pixel PLL Control 2 register bits [2:0] (8 value M/N direct programming)

One of eight pairs of registers M0/N0, M1/N1, M2/N2, M3/N3, M4/N4, M5/N5, M6/N6, M7/N7 is selected with Pixel PLL Control 2 register bits [2:0] to provide the VCO divider/reference divider inputs to the Pixel PLL.

The Fixed Pixel PLL Reference Divider register is not used. Pixel PLL Control 2 register bit 3 has no effect.

100 Standard mode:
External FS[1:0] inputs
One of four sets of registers M0/N0/P0/C0, M1/N1/P1/C1, M2/N2/P2/C2, M3/N3/P3/C3 is selected with external signals FS[1:0] to provide the M, N, P and C programming values to the Pixel PLL.

101 Standard mode:
Pixel PLL Control 2 register bits [1:0]

One of four sets of registers M0/N0/P0/C0, M1/N1/P1/C1, M2/N2/P2/C2, M3/N3/P3/C3 is selected with Pixel PLL Control 2 register bits [1:0] to provide the M, N, P and C programming values to the Pixel PLL.

Pixel PLL Control 2 register bits [3:2] have no effect.

110 Reserved

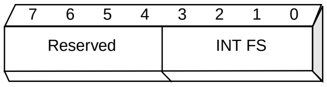
111 Reserved

The Pixel PLL frequency is programmed using several values. Multiple sets of these values may be stored in registers sets. The desired set of values can be selected using external signals (the FS[1:0] inputs) or by using the internal FS[3:0] bits of the PLL Control 2 register.

When the standard programming method is used then EXT/INT is set to 100 or 101 to use external selection or internal selection of frequency values.

When one of the two compatibility methods is used then EXT/INT is set to 000 or 001 for external selection, and 010 or 011 for internal selection.

Pixel PLL Control 2



Index: 0x0011

Access: Read/Write

Power on Value: 0x00

Bits 7 - 3 Reserved

Bits 3 - 0 INT FS - Internal Frequency Selection. Identifies which Pixel PLL programming registers to use when Pixel PLL Control 1 register bits EXT/INT specify internal frequency selection (EXT/INT = 010, 011 or 101).

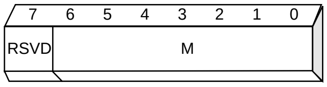
EXT/INT = 100	**EXT/INT = 101**	Standard Programming Method
External FS[1:0]	INT FS[1:0]	Selected Registers
00	00	M0,N0,P0,C0
01	01	M1,N1,P1,C1
10	10	M2,N2,P2,C2
11	11	M3,N3,P3,C3
Note: External FS[1:0] are the chip inputs FS[1:0]. INT FS[1:0] are Pixel PLL Control 2 register bits [1:0]. When EXT/INT = 100 Pixel PLL Control 2 register is not used. When EXT/INT = 101 Pixel PLL Control 2 register bits [3:2] are not used and have no effect.		

EXT/INT = 000		**EXT/INT = 001**	
External FS[1:0]	Selected Register	External FS[1:0]	Selected Registers
00	F0	00	M0, N0
01	F1	01	M1, N1
10	F2	10	M2, N2
11	F3	11	M3, N3
Note: External FS[1:0] are the chip inputs FS[1:0]. Pixel PLL Control 2 register is not used when Pixel PLL Control 1 register bits EXT/INT = 000 or 001.			

EXT/INT = 010		**EXT/INT = 011**	
INT FS[3:0]	Selected Register	INT FS[2:0]	Selected Registers
0000	F0	000	M0, N0
0001	F1	001	M1, N1
0010	F2	010	M2, N2
0011	F3	011	M3, N3
0100	F4	100	M4, N4
0101	F5	101	M5, N5
0110	F6	110	M6, N6
0111	F7	111	M7, N7
1000	F8	**Note**: When Pixel PLL Control 1 EXT/INT bits = 010 or 011, Pixel PLL Control 2 bits INT FS[3:0] select the frequency register(s). For EXT/INT = 011, INT FS[3] bit is not used and has no effect.	
1001	F9		
1010	F10		
1011	F11		
1100	F12		
1101	F13		
1110	F14		
1111	F15		

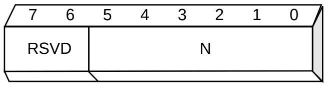
15.2.4 Pixel PLL Programming - Standard Mode

Pixel M0, Pixel M1, Pixel M2, Pixel M3



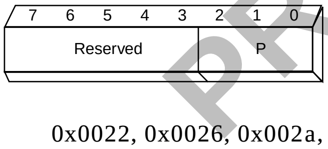
Index:	0x0020, 0x0024, 0x0028, 0x002c
Access:	Read/Write
Power on Value:	0x05 (index 0x0020)
	0x00 (others)
Bits 7 - 6	Reserved
Bits 5 - 0	M

Pixel N0, Pixel N1, Pixel N2, Pixel N3



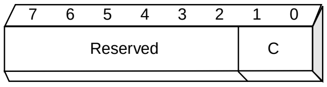
Index:	0x0021, 0x0025, 0x0029, 0x002d
Access:	Read/Write
Power on Value:	0x0e (index 0x0021)
	0x00 (others)
Bit 7	Reserved
Bits 6 - 0	N

Pixel P0, Pixel P1, Pixel P2, Pixel P3



Index:	0x0022, 0x0026, 0x002a, 0x002e
Access:	Read/Write
Power on Value:	0x00 (all)
Bits 7 - 3	Reserved
Bits 2 - 0	P

Pixel C0, Pixel C1, Pixel C2, Pixel C3



Index:	0x0023, 0x0027, 0x002b, 0x002f
Access:	Read/Write
Power on Value:	0x00 (all)
Bits 7 - 2	Reserved
Bits 1- 0	C

When the standard programming method is chosen for the Pixel PLL (Pixel PLL Control 1 register, EXT/INT bits = ‘100’ or ‘101’), the registers at 0x0020 - 0x002f provide the N, M, P and C programming values.

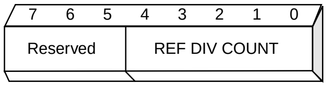
The valid ranges of these bits and their use are described in Sections 2.6 through 2.9.

There are four sets of M, N, P and C values. One of four sets is chosen under control of the PLL Control 1 and PLL Control 2 registers.

Note that, following a reset, the Pixel PLL is set to use compatibility programming using power on values appropriate for that mode (in registers at 0x0020 and 0x0021). Thus, when switched to standard programming, the power on values found in those registers are actually “left over” values that are not intended for use with the standard programming method.

15.2.5 Pixel PLL Programming - Compatibility Mode

Fixed Pixel PLL Reference Divider



Index: 0x0014

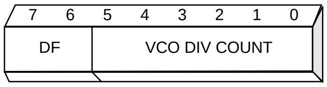
Access: Read/Write

Power on Value: 0x05

Bits 7 - 5 Reserved

Bits 4 - 0 REF DIV COUNT - Reference Divide Count

F0-F15: Pixel Frequency 0 to Frequency 15



Index: 0x0020 - 0x002f

Access: Read/Write

Power on Value: 0x05 (index 0x0020)
0x0e (index 0x0021)
0x00 (others)

Bits 7 - 6 DF - Desired Frequency

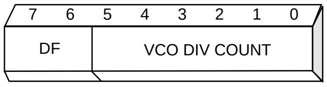
Bits 5 - 0 VCO DIV COUNT - VCO Divide Count

When the compatibility programming method is chosen for the Pixel PLL (Pixel PLL Control 1 register, EXT/INT bits = ‘000’ through ‘011’), the registers at 0x0014 and 0x0020 - 0x002f provide the “Reference Divider”, “VCO Divider” and DF programming values.programming values.

The above register diagram shows the format for the 16 pixel frequency registers F0 - F15. This format is selected when the EXT/INT bits (Pixel PLL Control 1 register, bits 2:0) = 000 or 010. The selected F0-F15 register provides the Pixel PLL with the DF value and the VCO divide count. All 16 frequency registers work with the same reference divide count, provided by the Fixed Pixel PLL Reference Divider register.

These 16 registers have a different format (M, N) (shown next), when EXT/INT = 001 or 011.

M0-M7, N0-N7



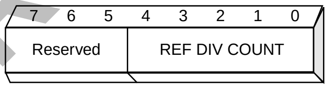
Index: 0x0020, 0x0022, 0x0024, 0x0026, 0x0028, 0x002A, 0x002C, 0x002E

Access: Read/Write

Power on Value: 0x05 (index 0x0020)
0x00 (others)

Bits 7 - 6 DF - Desired Frequency

Bits 5 - 0 VCO DIV COUNT - VCO Divide Count



Index: 0x0021, 0x0023, 0x0025, 0x0027, 0x0029, 0x002B, 0x002D, 0x002F

Access: Read/Write

Power on Value: 0x0e (index 0x0021)
0x00 (others)

Bits 7 - 5 Reserved

Bits 4 - 0 REF DIV COUNT - Reference Divide Count

The above diagrams show the formats for the 8 ‘M’ and 8 “N” pixel frequency registers. These formats are selected when the EXT/INT bits (Pixel PLL Control 1 Register, bits 2:0) = 001 or 011.

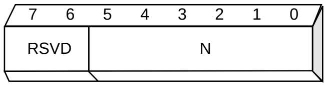
The 8 registers are grouped into four pairs, M0/N0, M1/N1, M2/N2, M3/N3. For a given pair, the “M” register provides the Pixel PLL with the DF value and the VCO divide count, and the “N” register provides the Pixel PLL with the reference divide count.

As described above these 16 registers have a different format (F) when EXT/INT = 000 or 010.

Further discussion of these registers and their use are given in Appendix D.0.

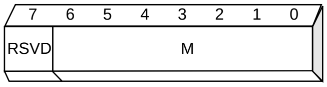
15.2.6 SYSCLK PLL Programming - Standard Mode

SYSCLK N



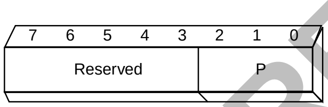
Index:	0x0015
Access:	Read/Write
Power on Value:	0x08
Bits 7 - 6	Reserved
Bits 5 - 0	N

SYSCLK M



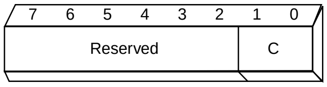
Index:	0x0016
Access:	Read/Write
Power on Value:	0x41
Bit 7	Reserved
Bits 6 - 0	M

SYSCLK P



Index:	0x0017
Access:	Read/Write
Power on Value:	Undefined
Bits 7 - 3	Reserved
Bits 2 - 0	P

SYSCLK C



Index:	0x0018
Access:	Read/Write
Power on Value:	Undefined
Bits 7 - 2	Reserved
Bits 1- 0	C

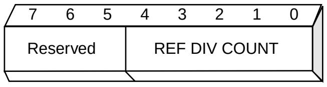
When the standard programming method is chosen for the SYSCLK PLL (System Clock Control register, PROG MODE bit = ‘1’), the registers at 0x0015 - 0x0018 provide the N, M, P and C programming values.

The valid ranges of these bits and their use are described in Sections 2.6 through 2.9.

Note that, following a reset, the SYSCLK PLL is set to use compatibility programming using power on values appropriate for that mode. Thus, when switched to standard programming, the power on values found in the SYSCLK N and SYSCLK M registers are actually “left over” values that are not intended for use with the standard programming method.

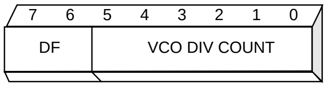
15.2.7 System Clock Frequency Selection - Compatibility Mode

System PLL Reference Divider



Index:	0x0015
Access:	Read/Write
Power on Value:	0x08
Bits 7 - 5	Reserved
Bits 4 - 0	REF DIV COUNT - Reference Divide Count

System PLL VCO Divider



Index:	0x0016
Access:	Read/Write
Power on Value:	0x41
Bits 7 - 6	DF - Desired Frequency
Bits 5 - 0	VCO DIV COUNT - VCO Divide Count

When the compatibility programming method is chosen for the SYSCLK PLL (System Clock Control register, PROG MODE bit = ‘0’), the registers at 0x0015 - 0x0016 provide the “Reference Divider”, “VCO Divider” and DF programming values.

The valid ranges of these bits and their use are described in Appendix D.0.

The registers at 0x0017 and 0x0018 are not used.

Compatibility programming is the default mode when the product is reset. The power on values will cause the PLL to produce a SYSCLK frequency of approximately 30 MHz (29.53 MHz) if the incoming REFCLK is the commonly available 14.31818 MHz frequency.

15.2.8 Cursor

Cursor Control

7	6	5	4	3	2	1	0
SMLC PART		PIX ORDR	LOC READ	UPDT CNTL	CURS SIZE	Cursor Mode	

Index: 0x0030

Access: Read/Write

Power on Value: 0x00

Bits 7 - 6 SMLC PART - Small Cursor Partition. Selects 1 of 4 partitions within the cursor array to use for the 32x32 cursor:

00	0x0100 - 0x01ff
01	0x0200 - 0x02ff
10	0x0300 - 0x03ff
11	0x0400 - 0x04ff

These bits have no effect when the cursor size is 64x64.

Bit 5 PIX ORDR - Pixel Order. Specifies ordering of pixel bits in the bytes of the cursor array.

0 Right-to-left

7	6	5	4	3	2	1	0
n+3		n+2		n+1		n+0	

1 Left-to-right

7	6	5	4	3	2	1	0
n+0		n+1		n+2		n+3	

Bit 4 LOC READ - Location Read-back Value. Specifies the value obtained by microprocessor reads of the Cursor X Low, Cursor X High, Cursor Y Low, and Cursor Y High registers.

0 Written Value - the value last written.

1 Actual Location - the location presently used for display. This will be different than the written value if a location register has been written but the location has not yet been updated. Following a cursor location update the "Written Value" and the "Actual Location" will be the same.

Bit 3 UPDT CNTL - Cursor Location Update Control. Controls when Cursor Location registers are sampled to change the cursor position.

0 Delayed - A write to the Cursor Y High register arms the circuitry for the update. The position is then updated (the cursor moves to the new location) when a vertical blanking period is detected.

1 Immediate - Move the cursor immediately following a write to any of the Cursor X Low, Cursor X High, Cursor Y Low, or Cursor Y High registers.

Bit 2 Cursor Size

0	32x32
1	64x64

Bits 1 - 0 Cursor Mode

Standard Cursor (ACA ENAB = '0' in Advanced Cursor Control register):

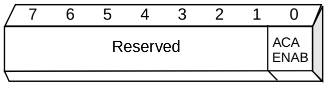
00	OFF
01	ON - Mode 0 (3 colors)
10	ON - Mode 1 (2 colors and highlighting)
11	ON - Mode 2 (2 colors)

Advanced Cursor (ACA ENAB = '1' in Advanced Cursor Control register):

00	OFF
01	ON
10	ON
11	ON

When the Advanced Cursor is enabled the cursor display modes are set in the Advanced Cursor Attribute register.

Advanced Cursor Control



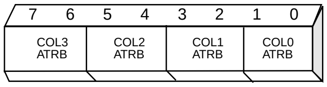
Index: 0x0037
Access: Read/Write
Power on Value: 0x00
Bits 7 - 1 Reserved
Bit 0 ACA ENAB - Advanced Cursor Attribute Enable. When the cursor is ON, determines how the cursor bits in the cursor array are interpreted.

0 Standard Cursor - use Cursor Mode bits in Cursor Control register.

1 Advanced Cursor - use Advanced Cursor Attribute register.

This register has no effect when the cursor is OFF (Cursor Mode = ‘00’).

Advanced Cursor Attribute



Index: 0x0038
Access: Read/Write
Power on Value: Undefined
Bits 7 - 6 COL3 ATRB - Cursor Color 3 Attribute
Bits 5 - 4 COL2 ATRB - Cursor Color 2 Attribute
Bits 3 - 2 COL1 ATRB - Cursor Color 1 Attribute
Bits 1 - 0 COL0 ATRB - Cursor Color 0 Attribute

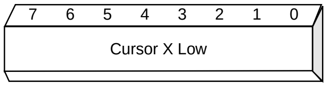
When the cursor is ON (Cursor Control register, Cursor Mode is not ‘00’), and the Advanced Cursor is enabled (Advanced Cursor Control register, ACA ENAB = ‘1’), then the attributes in this register control the interpretation of the cursor pixel bits in the cursor array. The interpretations specified by the Cursor Mode bits (‘01’, ‘10’, and ‘11’) are ignored.

For each of the four colors that can be selected by the cursor pixel bits (Color 3, Color 2, Color 1, Color 0), the COLx ATRB bits specify the display of that color as follows:

00	Transparent
01	Solid Color
10	Translucent
11	Highlighted

When the ACA ENAB bit is off this register has no effect.

Cursor X Low



Index: 0x0031

Access: Read/Write

Power on Value: Undefined

Bits 7 - 0 Cursor X Low. The low order bits of the cursor X (horizontal) position.

A write to this register will update the cursor position:

1. When both the Cursor Y High register is written and vertical blanking is detected, OR
2. Immediately

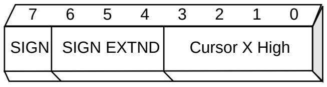
under control of the UPDT CNTL bit of the Cursor Control register.

The value read back:

1. Written Value, or
2. Actual Location

is controlled by the LOC READ bit of the Cursor Control register:

Cursor X High



Index: 0x0032

Access: Read/Write

Power on Value: Undefined

Bit 7 Sign

Bits 6 - 4 SIGN EXTND - Sign Extended. These bits are always the same as bit 7. On a write these bits are discarded and replaced with the value written to bit 7. On a read they will return the same value as bit 7.

Bits 3 - 0 Cursor X High. The high order bits of the cursor X (horizontal) position.

Cursor X High and Cursor X Low form a combined register that holds a signed cursor X position in two's complement form. The X position range is -4096 to +4095.

A write to this register will update the cursor position:

1. When both the Cursor Y High register is written and vertical blanking is detected, OR
2. Immediately

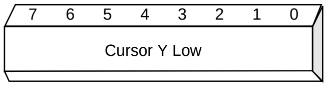
under control of the UPDT CNTL bit of the Cursor Control register

The value read back:

1. Written Value, or
2. Actual Location

is controlled by the LOC READ bit of the Cursor Control register.

Cursor Y Low



Index: 0x0033

Access: Read/Write

Power on Value: Undefined

Bits 7 - 0 Cursor Y Low. The low order bits of the cursor Y (vertical) position.

A write to this register will update the cursor position:

1. When both the Cursor Y High register is written and vertical blanking is detected, or
2. Immediately

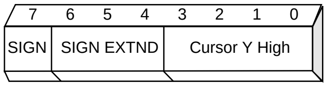
under control of the UPDT CNTL bit of the Cursor Control register.

The value read back:

1. Written Value, or
2. Actual Location

is controlled by the LOC READ bit of the Cursor Control register.

Cursor Y High



Index: 0x0034

Access: Read/Write

Power on Value: Undefined

Bit 7 Sign

Bits 6 - 4 SIGN EXTND - Sign Extended. These bits are always the same as bit 7. On a write these bits are discarded and replaced with the value written to bit 7. On a read they will return the same value as bit 7.

Bits 3 - 0 Cursor Y High. The high order bits of the cursor Y (vertical) position.

Cursor Y High and Cursor Y Low form a combined register that holds a signed cursor Y position in two's complement form. The Y position range is -4096 to +4095.

A write to this register will update the cursor position:

1. When vertical blanking is detected, or
2. Immediately

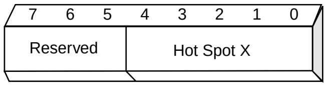
under control of the UPDT CNTL bit of the Cursor Control register.

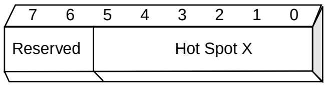
The value read back:

1. Written Value, or
2. Actual Location

is controlled by the LOC READ bit of the Cursor Control register.

Cursor Hot Spot X





Index: 0x0035

Access: Read/Write

Power on Value: Undefined

32x32 Cursor

Bits 7 - 5 Reserved

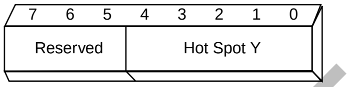
Bits 4 - 0 HOT SPOT X. Specifies which pixel in a cursor row is the X position pixel.

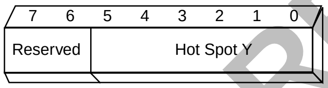
64x64 Cursor

Bits 7 - 6 Reserved

Bits 5 - 0 HOT SPOT X. Specifies which pixel in a cursor row is the X position pixel.

Cursor Hot Spot Y





Index: 0x0036

Access: Read/Write

Power on Value: Undefined

32x32 Cursor

Bits 7 - 5 Reserved

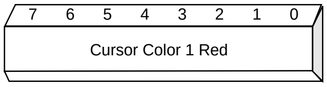
Bits 4 - 0 HOT SPOT Y. Specifies which pixel in a cursor column is the Y position pixel.

64x64 Cursor

Bits 7 - 6 Reserved

Bits 5 - 0 HOT SPOT Y. Specifies which pixel in a cursor column is the Y position pixel.

Cursor Color 1 Red



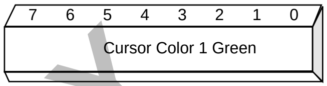
Index: 0x0040

Access: Read/Write

Power on Value: Undefined

Bits 7 - 0 Cursor Color 1 Red

Cursor Color 1 Green



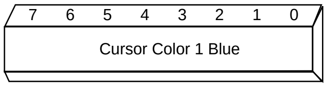
Index: 0x0041

Access: Read/Write

Power on Value: Undefined

Bits 7 - 0 Cursor Color 1 Green

Cursor Color 1 Blue



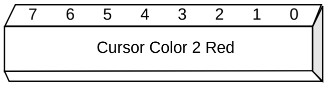
Index: 0x0042

Access: Read/Write

Power on Value: Undefined

Bits 7 - 0 Cursor Color 1 Blue

Cursor Color 2 Red



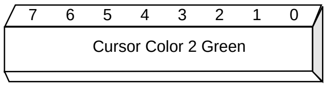
Index: 0x0043

Access: Read/Write

Power on Value: Undefined

Bits 7 - 0 Cursor Color 2 Red

Cursor Color 2 Green



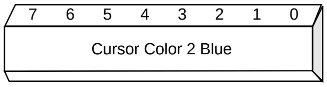
Index: 0x0044

Access: Read/Write

Power on Value: Undefined

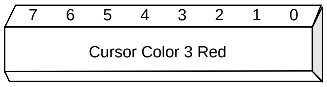
Bits 7 - 0 Cursor Color 2 Green

Cursor Color 2 Blue



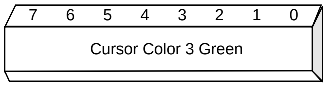
Index: 0x0045
Access: Read/Write
Power on Value: Undefined
Bits 7 - 0 Cursor Color 2 Blue

Cursor Color 3 Red



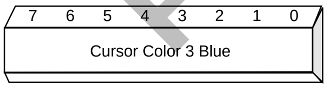
Index: 0x0046
Access: Read/Write
Power on Value: Undefined
Bits 7 - 0 Cursor Color 3 Red

Cursor Color 3 Green



Index: 0x0047
Access: Read/Write
Power on Value: Undefined
Bits 7 - 0 Cursor Color 3 Green

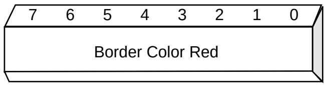
Cursor Color 3 Blue



Index: 0x0048
Access: Read/Write
Power on Value: Undefined
Bits 7 - 0 Cursor Color 3 Blue

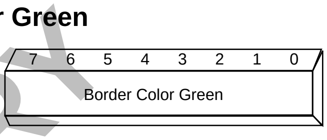
15.2.9 Border Color

Border Color Red



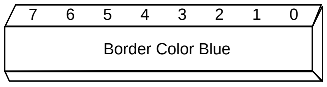
Index: 0x0060
Access: Read/Write
Power on Value: Undefined
Bits 7 - 0 Border Color Red

Border Color Green



Index: 0x0061
Access: Read/Write
Power on Value: Undefined
Bits 7 - 0 Border Color Green

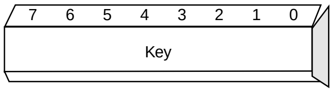
Border Color Blue



Index: 0x0062
Access: Read/Write
Power on Value: Undefined
Bits 7 - 0 Border Color Blue

15.2.10 Key Support

Key



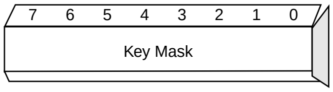
Index: 0x0068

Access: Read/Write

Power on Value: Undefined

Bits 7 - 0 Key

Key Mask



Index: 0x006c

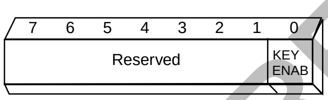
Access: Read/Write

Power on Value: Undefined

Bits 7 - 0 Key Mask

The Key and Key Mask registers are used for Chroma Key and Luma Key matching. The Key Mask is ANDed with the Key.

Key Control (RGB624)



Index: 0x0078

Access: Read/Write

Power on Value: 0x00

Bits 7 - 1 Reserved

Bit 0 KEY ENAB - Key Match Enable

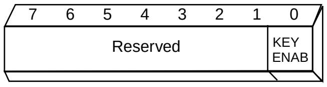
0 Key matching not enabled.

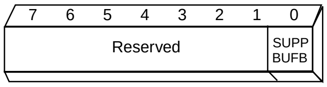
1 Chroma Key/Luma Key matching enabled.

This register has no effect for modes which do not support key operation, such as 24 BPP Packed, 16 BPP in straight RGB mode, or when the VGA port is selected.

For modes that do support key operation, such as 16 BPP in YUV mode, or 32 BPP, this register determines if key matching is in effect

Key Control/DB Operation (RGB624DB)





Index: 0x0078

Access: Read/Write

Power on Value: 0x00

Bits 7 - 1 Reserved

Bit 0 KEY ENAB - Key Match Enable

0 Key matching not enabled.

1 Chroma Key/Luma Key matching enabled.

Bit 0 SUPP BUFB - Suppress Buffer B

0 Normal double buffer operation.

1 Buffer A only; Buffer B suppressed.

For the RGB624DB the use of Bit 0 depends upon the mode of operation:

For 16 BPP YUV and 32 BPP, which support key matching, Bit 0 is used as the KEY ENAB bit to determine if key matching is enabled, the same as the RGB624.

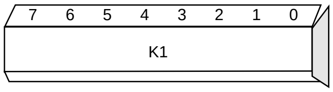
For 16 BPP RGB, which supports the double buffer operation, Bit 0 is used to suppress the display of the B buffer.

This register has no effect for modes which do not support key operation or double buffer operation.

.

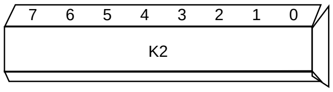
15.2.11 YUV:RGB Constants

K1



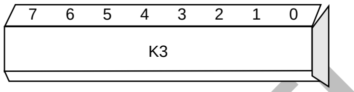
Index:	0x00a0
Access:	Read/Write
Power on Value:	Undefined
Bits 7 - 0	K1

K2



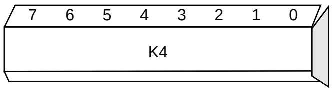
Index:	0x00a1
Access:	Read/Write
Power on Value:	Undefined
Bits 7 - 0	K2

K3



Index:	0x00a2
Access:	Read/Write
Power on Value:	Undefined
Bits 7 - 0	K3

K4



Index:	0x00a3
Access:	Read/Write
Power on Value:	Undefined
Bits 7 - 0	K4

These four registers hold the four constants used by the YUV to RGB color space converter.

15.2.12 Diagnostic Support

Revision Level

7	6	5	4	3	2	1	0
Revision Level							

Index: 0x0000
Access: Read Only
Power on Value: Revision Level
Bits 7 - 0 Product Revision Level

The value in this register indicates the revision level of the product.

ID

7	6	5	4	3	2	1	0
ID							

Index: 0x0001
Access: Read Only
Power on Value: 0x30
Bits 7 - 0 Product Identification Code

This register distinguishes among the various members of the IBM Microelectronics Palette DAC family. The value of 0x30 indicates that this product is one of several that have YUV to RGB conversion.

DAC Sense

7	6	5	4	3	2	1	0
LSNS	LBLU COMP	LGRN COMP	LRED COMP	SENS	BLUE COMP	GRN COMP	RED COMP

Index: 0x0082
Access: Read Only
Power on Value: Undefined
Bit 7 LSNS - Latched Sense
Bit 6 LBLU COMP - Latched Blue DAC Comparator Output
Bit 5 LGRN COMP - Latched Green DAC Comparator Output
Bit 4 LRED COMP - Latched Red DAC Comparator Output
Bit 3 SENS - Sense
Bit 2 BLU COMP - Blue DAC Comparator Output
Bit 1 GRN COMP - Green DAC Comparator Output
Bit 0 RED COMP - Red DAC Comparator Output

Bits 2,1,0 are the outputs of the three DAC reference comparators. The DAC output voltages are compared against the 0.35 V internal reference voltage (presented on COMPVREF). These bits are the “raw” outputs of the comparators.

Bits 6,5,4 are latched copies of bits 2,1,0. The latches are clocked during active line time (when $\overline{\text{BLANK}}$ [and $\overline{\text{BORDER}}$, when a border is used] are both high).

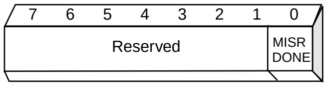
Bit 3 (Sense) represents the combined status of bits 2,1,0. If any of these bits is 0, bit 3 will be 0.

Bit 7 (Latched Sense) represents the combined status of bits 6,5,4. If any of these bits is 0, bit 7 will be 0.

Either bit 3 or bit 7 will be presented on the $\overline{\text{SENSE}}$ output, depending on the SENS SEL bit of the Miscellaneous Control 1 register.

If the selected bit is 0, $\overline{\text{SENSE}}$ will be low.
If the selected bit is 1, $\overline{\text{SENSE}}$ will be high.

MISR Status



Index: 0x0083

Access: Read Only

Power on Value: 0x00

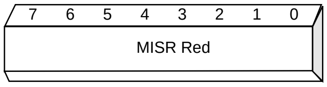
Bits 7 - 1 Reserved

Bit 0 MISR DONE - Indicates the status of the diagnostic signature accumulation in the MISR Red, MISR Green and MISR Blue registers

0 Not done - the MISR is not running or has not finished accumulating a signature.

1 Done - The signature accumulation has completed and the contents of the three MISR registers are valid.

MISR Red



Index: 0x0084

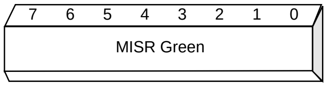
Access: Read Only

Power on Value: Undefined

Bits 7 - 0 Multiple Input Signature Register Red

This register along with MISR GREEN and MISR BLUE is used to accumulate a diagnostic signature on the values presented to the DACs. The input to the Red DAC is the parallel data input to this portion of the MISR.

MISR Green



Index: 0x0086

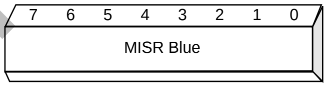
Access: Read Only

Power on Value: Undefined

Bits 7 - 0 Multiple Input Signature Register Green

This register along with MISR RED and MISR BLUE is used to accumulate a diagnostic signature on the values presented to the DACs. The input to the Green DAC is the parallel data input to this portion of the MISR.

MISR Blue



Index: 0x0088

Access: Read Only

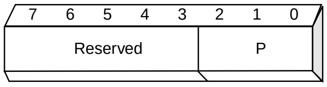
Power on Value: Undefined

Bits 7 - 0 Multiple Input Signature Register Blue

This register along with MISR RED and MISR GREEN is used to accumulate a diagnostic signature on the values presented to the DACs. The input to the Blue DAC is the parallel data input to this portion of the MISR.

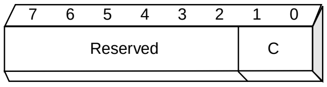
Note: The reset, accumulation, and hold function of the MISR, and the meaning of the MISR DONE bit, is controlled by the MISR CNTL bit of the Miscellaneous Control 1 register, and the $\overline{\text{BLANK}}$ input (and the $\overline{\text{BORDER}}/\overline{\text{OE}}$ input, in interlace mode). See 13.0 "Diagnostic Support" on page 36 for more information.

Pixel P Input



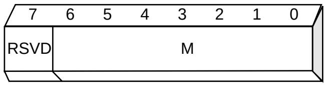
Index:	0x008c
Access:	Read Only
Power on Value:	Undefined
Bits 7 - 3	Reserved
Bits 2 - 0	P

Pixel C Input



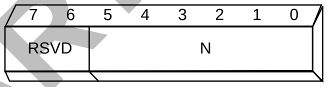
Index:	0x008d
Access:	Read Only
Power on Value:	Undefined
Bits 7 - 2	Reserved
Bits 1- 0	C

Pixel M Input



Index:	0x008e
Access:	Read Only
Power on Value:	0x05 (FS[1:0] = 00) 0x0e (FS[1:0] = 01) 0x00 (FS[1:0] = 10 or 11)
Bit 7	Reserved
Bits 6 - 0	M

Pixel N Input

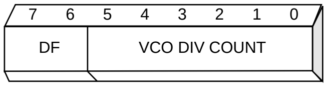


Index:	0x008f
Access:	Read Only
Power on Value:	0x05
Bits 7 - 6	Reserved
Bits 5 - 0	N

These four registers allow readback of the selected programming values presented to the Pixel PLL. When the standard programming method is used (Pixel PLL Control 1 register, EXT/INT bits = ‘100’ or ‘101’) the above representations are valid.

When compatibility programming is used (EXT/INT bits = ‘000’ through ‘011’) the contents of registers at 0x008c and 0x008d are undefined, and the representation of the registers at 0x008e and 0x008f are given in the next two register descriptions.

PLL VCO Divider Input



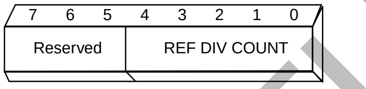
Index: 0x008e
Access: Read Only
Power on Value: 0x05 (FS[1:0] = 00)
0x0e (FS[1:0] = 01)
0x00 (FS[1:0] = 10 or 11)
Bits 7 - 6 DF - Desired Frequency
Bits 5 - 0 VCO DIV COUNT - VCO Divide Count

This register allows readback of the selected PLL VCO divider input. It is one of these registers:

- F0, F1, F2, F3, F4, F5, F6, F7, F8, F9, F10, F11, F12, F13, F14, F15
- M0, M1, M2, M3, M4, M5, M6, M7

as determined by the PLL Control 1 EXT/INT bits (2:0), the inputs FS[3:0], and PLL Control 2 INT FS bits (3:0).

PLL Reference Divider Input



Index: 0x008f
Access: Read Only
Power on Value: 0x05
Bits 7 - 5 Reserved
Bits 4 - 0 REF DIV COUNT - Reference Divide Count

This register allows readback of the input to the PLL reference divider.

- Fixed PLL Reference Divider
- N0, N1, N2, N3, N4, N5, N6, N7

as determined by the PLL Control 1 EXT/INT bits (2:0), the inputs FS[3:0], and PLL Control 2 INT FS bits (3:0).

The above register contents are valid when the compatibility programming mode is used (Pixel PLL Control 1 register, EXT/INT bits = ‘000’ through ‘011’).

VRAM Mask Low

7	6	5	4	3	2	1	0
MASK 31:28	MASK 27:24	MASK 23:20	MASK 19:16	MASK 15:12	MASK 11:08	MASK 07:04	MASK 03:00

Index:	0x0090
Access:	Read/Write
Power on Value:	Undefined
Bit 7	Mask VRAM PIX inputs 31:28
Bit 6	Mask VRAM PIX inputs 27:24
Bit 5	Mask VRAM PIX inputs 23:20
Bit 4	Mask VRAM PIX inputs 19:16
Bit 3	Mask VRAM PIX inputs 15:12
Bit 2	Mask VRAM PIX inputs 11:08
Bit 1	Mask VRAM PIX inputs 07:04
Bit 0	Mask VRAM PIX inputs 03:00

A value of 1 on any of the bits in this register masks (forces to 0) the corresponding received VRAM pixel port inputs. This register has no effect on the inputs unless enabled with the VMSK CNTL bit of the Miscellaneous Control 1 register.

VRAM Mask High

7	6	5	4	3	2	1	0
MASK 63:60	MASK 59:56	MASK 55:52	MASK 51:48	MASK 47:44	MASK 43:40	MASK 39:36	MASK 35:32

Index:	0x0091
Access:	Read/Write
Power on Value:	Undefined
Bit 7	Mask VRAM PIX inputs 63:60
Bit 6	Mask VRAM PIX inputs 59:56
Bit 5	Mask VRAM PIX inputs 55:52
Bit 4	Mask VRAM PIX inputs 51:48
Bit 3	Mask VRAM PIX inputs 47:44
Bit 2	Mask VRAM PIX inputs 43:40
Bit 1	Mask VRAM PIX inputs 39:36
Bit 0	Mask VRAM PIX inputs 35:32

A value of 1 on any of the bits in this register masks (forces to 0) the corresponding received VRAM pixel port inputs. This register has no effect on the inputs unless enabled with the VMSK CNTL bit of the Miscellaneous Control 1 register.

Note: The mask function is intended to be used with the MISR for diagnostics. See 13.0 “Diagnostic Support” on page 36 for more details.

16.0 Pin Descriptions

Table 16. Pin Descriptions

Signal	Type	Pin(s)	Description
Clocks and Clock Controls			
REFCLK	I	80	Reference Clock. A fixed frequency of 2 MHz to 100 MHz applied to this pin provides the reference clock for the programmable pixel and system clock PLLs. When the Direct Programming method is used the REFCLK frequency range is 4 MHz to 62 MHz on 2 MHz boundaries.
FS[1:0]	I	96,83	Frequency Select. These two inputs select one of four sets of registers containing the programming values for the pixel PLL.
DDOTCLK	O	116	Divided Dot Clock. The output of the pixel PLL, divided by 1, 2, 4, 8 or 16. The divide factor is under register control. In 24 BPP Packed pixel mode the SCLK signal can be selected for this output instead of the divided pixel PLL output, under register control. This output can be 3-stated under register control.
SCLK	O	113	Serial Clock. A divided version of the pixel PLL, where the divide ratio is determined by the required bandwidth of the incoming pixels. When the PIX port is selected, the SCLK frequency is a function of the VRAM width and the pixel format (bits per pixel). SCLK is equal to the pixel PLL output when the VGA port is selected. This output can be 3-stated under register control.
LCLK	I	109	Load Clock. Latches data from the PIX port, the VGA port, and the video control inputs.
SYSCLK	O	75	System Clock. The output of the SYSCLK programmable PLL. This output can be 3-stated under register control.
Video Data Inputs			
PIX[63:0]	I	10,9,8,7,67,66,65,64, 123,122,121,120.119,118, 117,140,139,138,137,136, 135,134,133,74,71,70,69, 68,6,5,4,3,2,1,144,143, 131,130,129,128,127,126, 125,124,56,55,54,53,52, 50,63,48,47,62,46,61,60, 59,58,45,44,57,49,43	Pixel data in from VRAMs. Pixel data in can be selected as 64 or 32 bits using the VRAM SIZE bit of the Miscellaneous Control 1 register. For 32 bit use inputs PIX[63:32] are not used. Latched on rising edge of LCLK.
VGA[7:0]	I	42,41,40,39,38,37,36,35	VGA data in. Latched on rising edge of LCLK.

Type: I = Input, O = Output, B = Bidirectional, C = Component

Table 16. Pin Descriptions (Continued)

Signal	Type	Pin(s)	Description
Video Control Inputs			
$\overline{\text{BLANK}}$	I	78	A low level indicates blanking time; a high level indicates active picture time (pixel data, cursor, or border displayed). Latched on rising edge of LCLK.
$\overline{\text{BORDER}}$/O$\overline{\text{E}}$	I	79	This is a shared input. It may be used either as a border indicator, or as an interlace control. Within this document, this input may be referred to as $\overline{\text{BORDER}}$ or ODD/$\overline{\text{EVEN}}$, depending on usage. When used as $\overline{\text{BORDER}}$: When $\overline{\text{BLANK}}$ is high (picture time), a low level on $\overline{\text{BORDER}}$ indicates the contents of the border registers should be displayed, and a high level indicates that pixel data or cursor should be displayed. When $\overline{\text{BLANK}}$ is low (blanking time) $\overline{\text{BORDER}}$ must be low. If no border is to be displayed $\overline{\text{BORDER}}$ should be tied to $\overline{\text{BLANK}}$. Latched on rising edge of LCLK. When used as O/$\overline{\text{E}}$ (ODD/$\overline{\text{EVEN}}$): Used in interlace mode to identify a field as odd or even; determines which row of cursor RAM to display if the cursor is enabled. In this usage the input should only change during vertical blanking.
$\overline{\text{HCSYNCIN}}$	I	77	This is a shared input. It may be used either as horizontal sync in or composite sync in. When used as Horizontal Sync In, a delayed copy of this signal is presented on $\overline{\text{HSYNCOUT}}$ to align the timing of horizontal sync to the pixel data at the DACs. The incoming polarity can be inverted under register control. Latched on rising edge of LCLK. When used as Composite Sync In, when enabled, this signal is presented on the Green DAC with the video data. The signal is delayed to match the delay of the pixel data. The incoming polarity can be inverted under register control. Latched on rising edge of LCLK
$\overline{\text{VSYNCIN}}$	I	76	Vertical Sync In. A copy of this signal is presented on $\overline{\text{VSYNCOUT}}$. The incoming polarity can be inverted under register control.
Video Control Outputs			
$\overline{\text{HSYNCOUT}}$	O	32	Horizontal Sync Out. This is a copy of $\overline{\text{HCSYNCIN}}$ (or inverted $\overline{\text{HCSYNCIN}}$), delayed by the same number of pixel clocks as seen by the pixel data at the input to the DACs. It can be forced to a high or low level or 3-stated under register control. The amount of delay may also be adjusted with the Horizontal Sync Position register.
$\overline{\text{VSYNCOUT}}$	O	12	Vertical Sync Out. This is a copy of $\overline{\text{VSYNCIN}}$ (or inverted $\overline{\text{VSYNCIN}}$). It can be forced to a high or low level or 3-stated under register control.
Type: I = Input, O = Output, B = Bidirectional, C = Component			

Table 16. Pin Descriptions (Continued)

Signal	Type	Pin(s)	Description
Microprocessor Interface			
$\overline{\text{WR}}$	I	103	Write strobe. Writes data into the register selected by RS[2:0] The leading edge samples RS[2:0]. The trailing edge clocks the data on D[7:0] into the selected register.
$\overline{\text{RD}}$	I	102	Read strobe. Drives the register contents selected by RS[2:0] onto D[7:0]. The leading edge samples RS[2:0]. When $\overline{\text{RD}}$ is low the D[7:0] drivers are enabled.
RS[2:0]	I	101,100,99	Register selects. Sampled on the leading edge of $\overline{\text{WR}}$ and $\overline{\text{RD}}$ and used to select one of the direct access registers. See Direct Access Registers for more details.
D[7:0]	B	112,111,110,108, 107,106,105,104	Bidirectional data bus for internal register write and read data. The drivers are enabled when $\overline{\text{RD}}$ is low, otherwise they are 3-stated.
Miscellaneous			
$\overline{\text{RESET}}$	I	81	Internal register and PLL reset. Resets bits of certain registers to a given state. (Generally set to VGA operation. See register descriptions for details.) Also initializes PLL circuits. A reset is **required** following power on to guarantee proper PLL operation.
$\overline{\text{VGAMODE}}$	O	98	Indicates the VGA port is the selected pixel port. This output is the inverted state of Miscellaneous Control 2 register bit 0 (PORT SEL).
$\overline{\text{SENSE}}$	O	11	DAC sense comparator output. Goes low when one or more of the three DAC outputs is above the comparator voltage reference. The three individual comparator outputs are also available as register bits. Either unlatched or latched comparator outputs may be selected for generating the $\overline{\text{SENSE}}$ output, under register control. This output can be 3-stated under register control.
Video Outputs			
RED	O	20	RED video out.
GREEN	O	28	GREEN video out.
BLUE	O	30	BLUE video out.

Type: I = Input, O = Output, B = Bidirectional, C = Component

Table 16. Pin Descriptions (Continued)

Signal	Typ e	Pin(s)	Description
DAC Support			
VREFIN	C	18	Voltage Reference In for the DACs. Connect 1.235 V to this pin and decouple it with a 10 nF ceramic capacitor to DACGND.
RREF	C	26	Resistor Reference. Connect a resistor from this pin to DACGND. This pin connects to an internal op amp which compares the voltage on this pin to that of VREFIN, and adjusts the current flowing out of the RREF pin such that the voltage developed across the reference resistor matches VREFIN. The value of the resistor determines the full scale output current of the DACs. A value of 698 Ω is recommended.
CVREF	C	15	Comparator Voltage Reference. An internal voltage divider between VREFIN and DACGND sets this pin to 0.35V. It is used internally by comparators to sense the values of the DAC outputs. Decouple this pin to DACGND with a 1nF ceramic capacitor.
GREF	C	14	DAC Gate Reference. Output of DAC op amp and input to gates of devices connecting DACVDD to the DAC current switches. Decouple this pin to DACVDD with a 1 nF ceramic capacitor.
PLL Support			
RCI0	C	92	Pixel PLL Loop filter. Connect to parallel 1.3 KΩ resistor and 680 pF capacitor. Return parallel resistor and capacitor to RCRET0 through 8.2 nF capacitor.
RCRET0	C	91	Pixel PLL Loop filter return. Connect as described above under RCI0.
RCI1	C	87	SYSCLK PLL Loop filter. Connect to parallel 1.3 KΩ resistor and 680 pF capacitor. Return parallel resistor and capacitor to RCRET1 through 8.2 nF capacitor.
RCRET1	C	86	SYSCLK PLL Loop filter return. Connect as described above under RCI1.
Type: I = Input, O = Output, B = Bidirectional, C = Component			

Table 16. Pin Descriptions (Continued)

Signal	Type	Pin(s)	Description
Manufacturing Test			
$\overline{\text{TESTMODE}}$	I	94	This input must be low for functional use. Leave it unconnected. An internal pulldown resistor causes this input to be at ground level.
$\overline{\text{DI1}}$	I	84	This input must be high for functional use.
$\overline{\text{DI2}}$	I	85	This input must be high for functional use.
$\overline{\text{RI}}$	I	95	This input must be high for functional use. An external 1 KΩ resistor to VDD is recommended.
Power and Ground			
VDD		34,73,115,142	Logic Power (3.3 V)
GND		17,33,51,72,82,97,114, 132,141	Logic Ground
DACVDD		13,19,21,24,27,29	DAC Power (3.3 V)
DACGND		16,22,23,25,31	DAC Ground
PLLVDD		90,93	PLL Power (3.3 V)
PLLGND		88,89	PLL Ground

Type: I = Input, O = Output, B = Bidirectional, C = Component

Table 17. Signal List by Pin Number

Pin	Signal	Description
001	PIX[30]	Pixel Data In
002	PIX[31]	Pixel Data In
003	PIX[32]	Pixel Data In
004	PIX[33]	Pixel Data In
005	PIX[34]	Pixel Data In
006	PIX[35]	Pixel Data In
007	PIX[60]	Pixel Data In
008	PIX[61]	Pixel Data In
009	PIX[62]	Pixel Data In
010	PIX[63]	Pixel Data In
011	$\overline{\mathrm{SENSE}}$	DAC Sense
012	$\overline{\mathrm{VSYNCOUT}}$	Vertical Sync Out
013	DACVDD	DAC Power (+3.3V)
014	GREF	DAC Gate Ref
015	CVREF	DAC Comp. VREF
016	DACGND	DAC Ground
017	GND	Logic Ground
018	VREFIN	DAC Voltage Ref
019	DACVDD	DAC Power (+3.3V)
020	RED	+ Red Output
021	DACVDD	DAC Power (+3.3V)
022	DACGND	DAC Ground
023	DACGND	DAC Ground
024	DACVDD	DAC Power (+3.3V)
025	DACGND	DAC Ground
026	RREF	DAC Resistor Ref
027	DACVDD	DAC Power (+3.3V)
028	GREEN	+ Green Output
029	DACVDD	DAC Power (+3.3V)
030	BLUE	+ Blue Output
031	DACGND	DAC Ground
032	$\overline{\mathrm{HSYNCOUT}}$	Horizontal Sync Out
033	GND	Logic Ground
034	VDD	Logic Power (+3.3 V)
035	VGA[0]	VGA Data In
036	VGA[1]	VGA Data In
037	VGA[2]	VGA Data In
038	VGA[3]	VGA Data In
039	VGA[4]	VGA Data In
040	VGA[5]	VGA Data In
041	VGA[6]	VGA Data In
042	VGA[7]	VGA Data In
043	PIX[0]	Pixel Data In
044	PIX[3]	Pixel Data In
045	PIX[4]	Pixel Data In
046	PIX[9]	Pixel Data In
047	PIX[11]	Pixel Data In
048	PIX[12]	Pixel Data In
049	PIX[1]	Pixel Data In
050	PIX[14]	Pixel Data In
051	GND	Logic Ground
052	PIX[15]	Pixel Data In
053	PIX[16]	Pixel Data In
054	PIX[17]	Pixel Data In
055	PIX[18]	Pixel Data In
056	PIX[19]	Pixel Data In
057	PIX[2]	Pixel Data In
058	PIX[5]	Pixel Data In
059	PIX[6]	Pixel Data In
060	PIX[7]	Pixel Data In
061	PIX[8]	Pixel Data In
062	PIX[10]	Pixel Data In
063	PIX[13]	Pixel Data In
064	PIX[56]	Pixel Data In
065	PIX[57]	Pixel Data In
066	PIX[58]	Pixel Data In
067	PIX[59]	Pixel Data In
068	PIX[36]	Pixel Data In
069	PIX[37]	Pixel Data In
070	PIX[38]	Pixel Data In
071	PIX[39]	Pixel Data In
072	GND	Logic Ground
073	VDD	Logic Power (+3.3 V)
074	PIX[40]	Pixel Data In
075	SYSCLK	System Clock Out
076	$\overline{\mathrm{VSYNCIN}}$	Vertical Sync In
077	$\overline{\mathrm{HCSYNCIN}}$	Hor/Comp Sync In
078	$\overline{\mathrm{BLANK}}$	Blank In
079	$\overline{\mathrm{BORDER}}/\overline{\mathrm{OE}}$	Border/Interlace In
080	REFCLK	PLL Ref. Clock In
081	$\overline{\mathrm{RESET}}$	Reset
082	GND	Logic Ground
083	FS[0]	Frequency Select
084	$\overline{\mathrm{DI1}}$	Driver Inhibit 1 (Test)
085	$\overline{\mathrm{DI2}}$	Driver Inhibit 2 (Test)
086	RCRET1	Sys Loop Filter Rtrn
087	RCI1	Sys Loop Filter
088	PLLGND	PLL Ground
089	PLLGND	PLL Ground
090	PLLVDD	PLL Power (+3.3V)
091	RCRET0	Pix Loop Filter Rtrn
092	RCI0	Pix Loop Filter
093	PLLVDD	PLL Power (+3.3V)
094	TESTMODE	Test Mode (Test)
095	$\overline{\mathrm{RI}}$	Receiver Inhibit (Test)
096	FS[1]	Frequency Select
097	GND	Logic Ground
098	VGAMODE	VGA Mode Out
099	RS[0]	Register Select [0]
100	RS[1]	Register Select [1]
101	RS[2]	Register Select [2]
102	$\overline{\mathrm{RD}}$	Microprocessor Read
103	$\overline{\mathrm{WR}}$	Microprocessor Write
104	D[0]	Microprocessor Data
105	D[1]	Microprocessor Data
106	D[2]	Microprocessor Data
107	D[3]	Microprocessor Data
108	D[4]	Microprocessor Data
109	LCLK	Load Clock In
110	D[5]	Microprocessor Data
111	D[6]	Microprocessor Data
112	D[7]	Microprocessor Data
113	SCLK	Serial Clock Out
114	GND	Logic Ground
115	VDD	Logic Power (+3.3 V)
116	DDOTCLK	Divided Dot Clock Out
117	PIX[49]	Pixel Data In
118	PIX[50]	Pixel Data In
119	PIX[51]	Pixel Data In
120	PIX[52]	Pixel Data In
121	PIX[53]	Pixel Data In
122	PIX[54]	Pixel Data In
123	PIX[55]	Pixel Data In
124	PIX[20]	Pixel Data In
125	PIX[21]	Pixel Data In
126	PIX[22]	Pixel Data In
127	PIX[23]	Pixel Data In
128	PIX[24]	Pixel Data In
129	PIX[25]	Pixel Data In
130	PIX[26]	Pixel Data In
131	PIX[27]	Pixel Data In
132	GND	Logic Ground
133	PIX[41]	Pixel Data In
134	PIX[42]	Pixel Data In
135	PIX[43]	Pixel Data In
136	PIX[44]	Pixel Data In
137	PIX[45]	Pixel Data In
138	PIX[46]	Pixel Data In
139	PIX[47]	Pixel Data In
140	PIX[48]	Pixel Data In
141	GND	Logic Ground
142	VDD	Logic Power (+3.3 V)
143	PIX[28]	Pixel Data In
144	PIX[29]	Pixel Data In

17.0 Electrical and Timing Specifications

Table 18. Recommended Operating Conditions

Parameter	Symbol	170 MHz		220 MHz		Units
		Min.	Max.	Min.	Max.	
Power Supply	VDD, DACVDD, PLLVDD	3.0	3.6	3.0	3.6	Volts
Case Temperature	T_C	0	100	0	100	˚C
DAC Output Load	R_L	37.5	50	37.5	50	Ω
Reference Voltage	V_{REF}	1.204	1.266	1.204	1.266	Volts

Table 19. Absolute Maximum Ratings

Parameter	Symbol	Min.	Max.	Units
Power Supply	VDD, DACVDD, PLLVDD	-0.5	3.8	Volts
Signal Pin Voltage		-0.5	5.5	Volts
DAC Output Short Circuit Duration	t_{SC}		∞	seconds
Case Temperature	T_C	0	145	˚C
Soldering Temperature (5 seconds, 0.25 in. from case)	T_{SOL}		260	˚C
Vapor Phase Soldering Temperature (1 minute)	$T_{V,SOL}$		220	˚C

Table 20. DC Characteristics

Parameter	Symbol	Min.	Typical	Max.	Units
DAC Outputs					
Resolution		8	8	8	Bits
Integral Linearity Error	ILE			3/4	LSB
Differential Linearity Error	DLE			3/4	LSB
Grey Scale Error				5	% Grey Scale
Monotonicity		Guaranteed	Guaranteed	Guaranteed	
Coding					Binary
CMOS Digital Inputs					
Input High Voltage (V_{DD} = 3.3 V)	V_{IH}	2.0		5.5	Volts
Input Low Voltage	V_{IL}	-0.5		0.8	Volts
Input High Current (V_{IH} = 2.4V)	I_{IH}			20	µA
Input Low Current (V_{IL} = 0.0V)	I_{IL}	-20			µA
Input Capacitance (f=1 MHz, V_{IN}=2.4V)	C_{IN}		4	8	pF
Digital Outputs					
Output High Voltage (I_{OH} = -12 mA)	V_{OH}	2.4			Volts
Output Low Voltage (I_{OL} = 6 mA)	V_{OL}			0.4	Volts
Hi-Z Current ($0 \leq V \leq 3.6$ V)	I_{OZ}	-20		20	µA
Analog Outputs					
DAC Inaccuracy				7	%
DAC-to-DAC Mismatch				5	%
Output Compliance	V_{OC}	-0.5		1.2	Volts

Table 21. AC Characteristics

Parameter	Symbol	Spec.	170 MHz	220 MHz	Units
RS[2:0] Setup	t_1	min	10	10	ns
RS[2:0] Hold	t_2	min	10	10	ns
$\overline{\text{RD}}$, $\overline{\text{WR}}$ Low	t_3	min	50	50	ns
$\overline{\text{RD}}$, $\overline{\text{WR}}$ High	t_4	min	6 × pclk	6 × pclk	ns
$\overline{\text{RD}}$ Low to Data Bus Driven	t_5	min	2	2	ns
$\overline{\text{RD}}$ Low to Data Bus Valid	t_6	max	40	40	ns
$\overline{\text{RD}}$ High to Data Bus 3-Stated	t_7	max	20	20	ns
Data Bus Hold from $\overline{\text{RD}}$ High	t_8	min	2	2	ns
Write Data Setup	t_9	min	10	10	ns
Write Data Hold	t_{10}	min	10	10	ns
LCLK, SCLK Low	t_{11}	min	4	4	ns
LCLK, SCLK High	t_{12}	min	4	4	ns
LCLK, SCLK Cycle	t_{13}				
16:1 MUX Mode		max	10.6	13.75	MHz
8:1 MUX Mode		max	21.25	27.5	MHz
4:1 MUX Mode		max	42.5	55	MHz
2:1 MUX Mode		max	85	100	MHz
1:1 MUX Mode		max	100	100	MHz
16:1 MUX Mode		min	94.12	72.7	ns
8:1 MUX Mode		min	47.06	36.4	ns
4:1 MUX Mode		min	23.53	18.2	ns
2:1 MUX Mode		min	11.77	10	ns
1:1 MUX Mode		min	10	10	ns
PIX[63:0] Setup	t_{14}	min	1	1	ns
PIX[63:0] Hold	t_{15}				
1:1 MUX Mode		min	4	4	ns
Not 1:1 MUX Mode		min	2	2	ns
VGA[7:0], $\overline{\text{BLANK}}$, $\overline{\text{BORDER/OE}}$ $\overline{\text{HCSYNCIN}}$, Setup(1)	t_{16}	min	3	3	ns
VGA[7:0], $\overline{\text{BLANK}}$, $\overline{\text{BORDER/OE}}$ $\overline{\text{HCSYNCIN}}$, Hold(1)	t_{17}	min	3	3	ns
SCLK to LCLK skew	t_{18}(4)	min	Any	Any	ns
(T=SCLK cycle time)		max	Any	Any	ns
Supply Current (1)		typ(2)	450	650	mA
		max(3)	716	890	mA

Notes:

1. Supply current is the total of I_{VDD}, I_{VDDDAC} and I_{VDDPLL}.
2. Typical power dissipation is for VDD, VDDDAC, VDDPLL = 3.3 V, TA = 20 ˚C, with typical pixel patterns such as displayed with graphical user interfaces, and
   170 MHz part running at 135 MHz (e.g., for 1280 x 1024 screen)
   220 MHz part running at 216 MHz (e.g., for 1600 x 1280 screen)
3. Maximum power dissipation is for VDD, VDDDAC, VDDPLL = 3.6 V, TA = 0 ˚C, with alternating full black/full white pixels running at the maximum specified frequency (170/220 MHz)
4. For the RGB51x and RGB52x products there is a specified relationship (min and max skew) between SCLK and LCLK. For the RGB624/RGB624DB this restriction is eliminated.

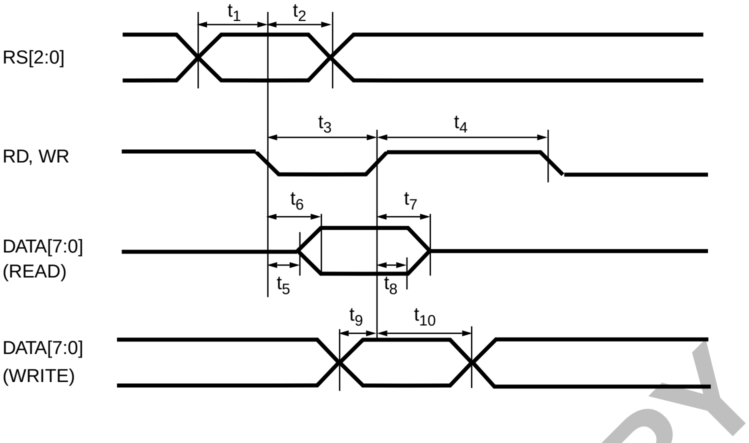


Figure 5. Microprocessor Interface Timing

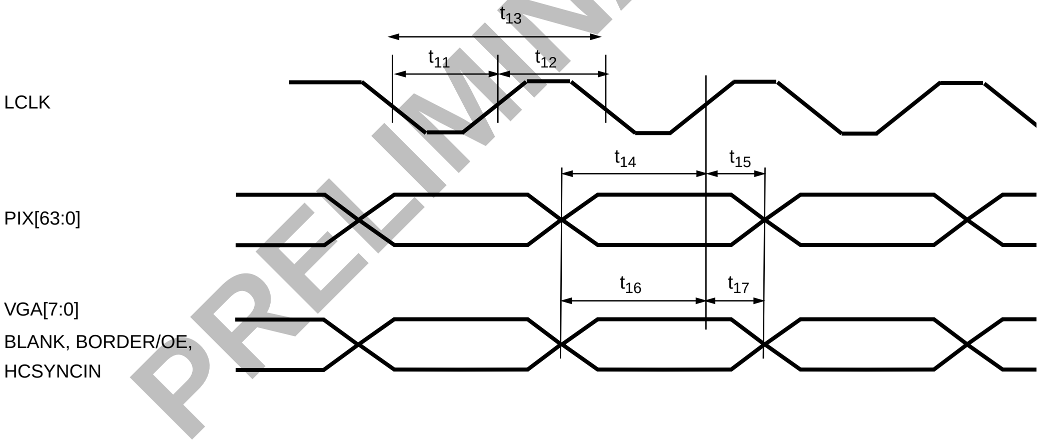


Figure 6. Pixel Data and Video Control Interface Timing

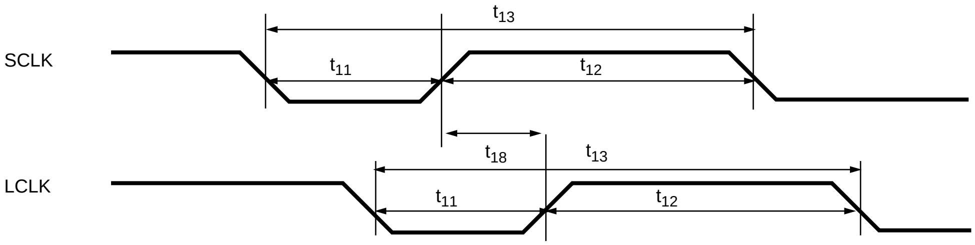


Figure 7. SCLK and LCLK Timing

18.0 Video Waveforms

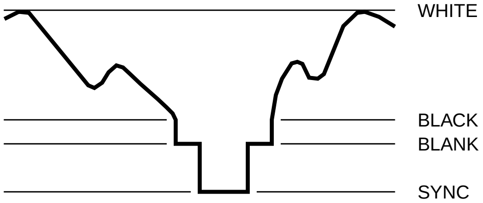


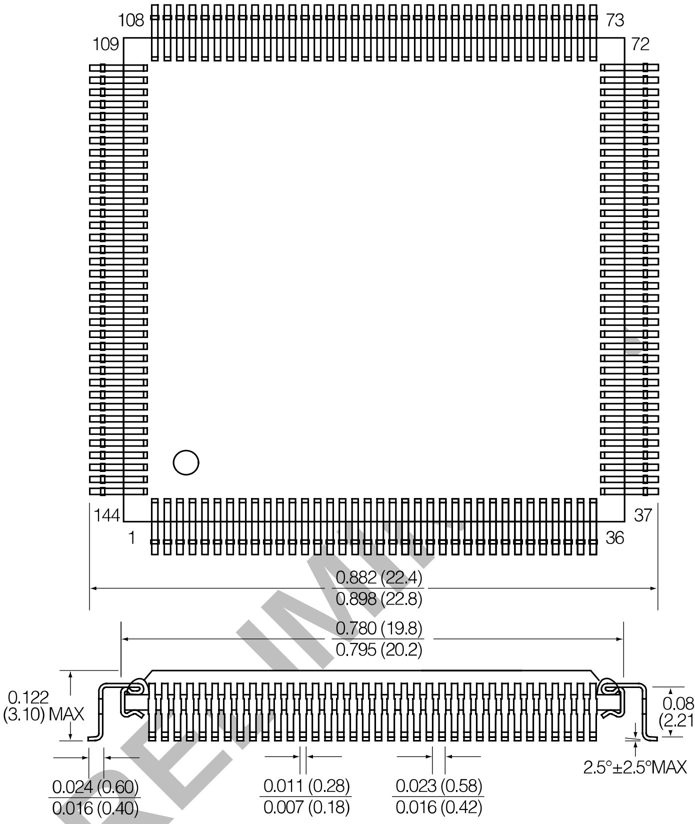
Table 22. Composite Video Output Waveform

	Doubly terminated 75 ohms, RREF=698 ohms											
Sync	**No**			**No**			**Yes**			**Yes**		
Pedestal	**No**			**Yes**			**No**			**Yes**		
Value	**IRE**	**mA**	**V**	**IRE**	**mA**	**V**	**IRE**	**mA**	**V**	**IRE**	**mA**	**V**
WHITE	100	18.65	0.70	92.5	19.05	0.714	100	26.67	1.00	92.5	26.67	1.00
BLACK		0	0	7.5	1.43	0.054		8.02	0.30	7.5	9.05	0.340
BLANK					0	0	-30				7.62	0.286
SYNC								0	0	-40	0	0
Note: RS-343A levels and tolerances assumed on all levels.												

Table 23. Composite Video Output Waveform

	Doubly terminated 100 ohms, RREF=927 ohms											
Sync	**No**			**No**			**Yes**			**Yes**		
Pedestal	**No**			**Yes**			**No**			**Yes**		
Value	**IRE**	**mA**	**V**	**IRE**	**mA**	**V**	**IRE**	**mA**	**V**	**IRE**	**mA**	**V**
WHITE	100	13.99	0.70	92.5	14.28	0.714	100	20.00	1.00	92.5	20.00	1.00
BLACK		0	0	7.5	1.07	0.054		6.01	0.30	7.5	6.78	0.340
BLANK					0	0	-30				5.71	0.286
SYNC								0	0	-40	0	0
Note: RS-343A levels and tolerances assumed on all levels.												

19.0 Package Information



20.0 Ordering Information

Table 24. Part Numbers

Product	Part Number	Speed
RGB624	IBM 37RGB624 CF 17	170 MHz
	IBM 37RGB624 CF 22	220 MHz
RGB624DB	IBM 37RGB624 DB 17	170 MHz
	IBM 37RGB624 DB 22	220 MHz

21.0 Change Summary

Table 25. Summary of Changes

Date	Changes
02/06/95	1. First publication.
10/09/95	This revision adds the RGB624DB product; the document becomes a combined RGB624/RGB624DB data sheet. The following changes correct errors to the 02/06/95 publication: 1. In section 2.7.2, "Programming Summary" on page 8, for the item calling for a check that the frequencies of the two PLLs be separated by a certain amount: changed the separation amount from “2 MHz” to “3 MHz”. This matches the specification given in other sections. 2. In section 7.0, “Color Space Conversion,” the four programmable constants for the color space converter are labeled as K1, K2, K3, K4. But the registers themselves were labeled K0, K1, K2, K3. The registers are now named K1, K2, K3, K4 to match their names in the text. 3. For the DAC Operation register (index 0x0006), inverted the definition of the DSR bit (DAC Slew Rate) bit. When ‘0’, this bit sets the slew rate to “fast”, when ‘1’, the slew rate is “slow”. This description was also updated in section 11.3, “DSR - DAC Slew Rate,” and in appendix C.1.3, “DAC Operation- Slew Rate”. 4. In Table 16, "Pin Descriptions," for the SYSCLK output, deleted a statement that the SYSCLK source can be either the SYSCLK PLL or the REFCLK input. This is true for the RGB524, but for the RGB624/RGB624DB the only source is the SYSCLK PLL. Added section C.1.7, “SYSCLK Source,” which notes that REFCLK can no longer be used as a source of the SYSCLK output. 5. Fixed miscellaneous typos. The remaining changes are related to adding the RGB624DB product information: 6. In section 2.4.1, “SCLK,” added details of SCLK operation when double buffer operation is enabled. 7. To section 3.0, “Modes of Operation,” added sub-section 3.3, “RGB624DB Double Buffer Operation”. 8. To section 5.0, “VRAM Pixel Formats,” added sub-section 5.7, “16 BPP Double Buffered (RGB624DB Only)”. This is the detailed description of the RGB526DB double buffer operation. 9. Added Table 11, “Pixel Format Table - RGB624DB Double Buffer Operation”. 10. In Table 15, "Internal Register Summary," renamed register at index 0x0078 from “Key Control” to “Key Control/DB Operation”. 11. On page 44, add bit 0, DBUF MODE, to description of the Miscellaneous Control 3 register. 12. On page 50, for the 16 Bit Pixel Control register, added a note for bits 4-3 and on page 51, (RGB format), added notes for bits 7-6 and bit 1 regarding use with double buffer operation. 13. On page 68 the description of the Key Control register is broken into two discussions: RGB624 only and RGB624DB only. For the RGB624DB, the register is called the Key Control/DB Operation register, and bit 0 is assigned two uses: Key Enable (non-double buffer operation) and Suppress Buffer B (double buffer operation). 14. To Table 24, "Part Numbers," added part numbers for RGB624DB. 15. Added appendix A.0, “RGB624, RGB624DB Comparison”. 16. Added appendix B.0, “Relationship to RGB526/RGB526DB”. 17. Edited appendix C.0, “Relationship to RGB524” (previously A.0), to mesh with appendices A.0 and B.0.

Appendix

A.0 RGB624, RGB624DB Comparison

The RGB624DB is a superset of the RGB624 that adds the 16 BPP double buffer operation. The two products are pin and register compatible and are identical in all aspects except for the register bits that control the double buffer operation:

1. The Miscellaneous Control 3 register, bit 0, is unused on the RGB624 but is used as the DBUF MODE (double buffer mode enable) bit on the RGB624DB.
2. For the RGB624DB the Key Control register is relabeled as the Key Control/DB Operation register. Bit 0 of this register takes on a dual function for the RGB624DB. For 16 BPP YUV and 32 BPP formats, bit 0 retains its function as the Key Enable bit (the same as the RGB624). For the 16 BPP RGB format, bit 0 is defined as the SUPP BUFB (suppress buffer B) bit.

The actions of these bits are described in sections 2.4.1, 3.3 and 5.7.

Note that the contents of the ID register (index 0x0001) are the same for both the RGB624 and the RGB624DB.

B.0 Relationship to RGB526/RGB526DB

The RGB624 is a superset of the RGB526 that adds overlay/underlay layers, and an on-board YUV to RGB programmable color space converter for YUV pixel support. The two products are pin and register compatible and are identical in all aspects except for the register bits that control the YUV formats and the overlay/underlay control. The new pixel formats are:

- ❑ 16 bit 4:2:2 YUV
- ❑ 16 bit - mixed 4:2:2 YUV/5:5:5 RGB, selected with a tag bit
- ❑ 16 bit - 8 RGB indirect, 16 4:2:2 YUV (overlay)
- ❑ 24 bit packed 4:4:4 YUV
- ❑ 32 bit - 24 4:4:4 YUV, 8 unused
- ❑ 32 bit - 24 4:4:4 YUV, 8 RGB indirect (overlay)
- ❑ 32 bit - 24 RGB direct, 8 RGB indirect (overlay)

Four new registers K1, K2, K3, K4 (indices 0x00a0 - 0x00a3) are added to hold the constants for the programmable YUV to RGB color space converter.

Two new registers, the Key (index 0x0068) and the Key Mask (index 0x006c) are added for luma key/ chroma key matching of the overlay layer for the new overlay pixel formats.

A third register, the Key Control (index 0x0078) is also added. This register has a single bit, KEY ENAB, to enable the luma key/chroma key matching operation. This in effect enables the overlay format when the 16 BPP (YUV) or 32 BPP format is selected. The power on value of this bit is “disabled”, so that normal 16 BPP and 32 BPP operation is used following a reset.

For 16 BPP operation the 16 Bit Pixel Control register (index 0x000c) has been extended. Bits 4 and 3 were previously reserved. These two bits are now called the B16 MODE bits. When these two bits are 00 the 16 bit pixels are used as before. When 01 or 11, the 16 bit pixels are interpreted as the “tagged (mixed) RGB/YUV” format. When 10, the 16 bit pixels are interpreted as 16 bit 4:2:2 YUV. Also, when B16 MODE = 10 and key matching is enabled, 8 bits are used for luma keying and the remaining 8 bits are used as an 8 bit indirect RGB underlay.

For the tagged RGB/YUV formats, the high order bit is the tag bit which tags the remaining bits as RGB or YUV. For pixels tagged as YUV, the B16 MODE bits determine the allocation of the remaining 15 bits. Following the high order tag bit, when B16 MODE = 01 the next 8 bits contain a Y or U/V value, with the lowest 7 bits containing the remaining U/V or Y value. When B16 = 11, following the tag bit the next 7 bits contain Y or U/V, with the lowest 8 bits containing the remaining U/V or Y value.

When the B16 MODE bits are 01, 10 or 11 the remaining bits of the 16 Bit Pixel Control register are redefined to control various uses of the incoming pixel. One bit, SWAP YC, determines which value (high or low) is the Y value and which is the U/V value. See the detailed register description for more information.

For 24 BPP packed operation the 24 Bit Packed Pixel Control register (index 0x000d) has been extended with an additional bit, bit 7 = RGB/YUV. When 0, the format is 24 bit packed RGB, the same as before. When 1, the format is 24 bit packed 4:4:4 YUV. When YUV is selected bit 0, the B24P DCOL bit, is not used.

For 32 BPP operation, the 32 bit Pixel Control register (index 0x000e) has been extended with two bits. Bit 7 = RGB/YUV determines the use of the lower 24 pixel bits.

When 0, these bits are RGB. When 1, these bits are 4:4:4 YUV. When YUV is selected bits 1 and 0, the B32 DCOL bits, are not used.

If chroma key matching is enabled with the new Key Control register, the 24 low order bits, RGB or YUV, are the underlay, and the 8 high order bits are the overlay. If key matching is not enabled the 8 high order bits are unused and the 24 low order bits are used as before.

The second new bit, bit 6 = B8 DCOL, determines whether the 8 bit overlay is indirect (through the palette) or direct (to the DACs, for grey scale). This bit has no effect if chroma key matching is not enabled.

The power on value is 0x00 for the 16 Bit, 24 Bit Packed, and 32 Bit Pixel Control registers. Thus, none of the new pixel formats are enabled following a reset.

B.1 Double Buffer Operation

The double buffer operation of the RGB526DB is identical to the double buffer operation of the RGB624DB. The register usage for programming the double buffer operation (as noted in appendix A.0) is identical between the RGB526DB and the RGB624DB.

B.2 ID Register

The ID register (index 0x0001) contains the value 0x02 for the RGB526 and RGB526DB. The RGB624 and RGB624DB share the ID value 0x30.

C.0 Relationship to RGB524

The RGB524, RGB526, RGB526DB, RGB624 and RGB624DB products are a set of pin and register compatible Palette DACs that share these basic features: 144-pin QFP package, 64-bit pixel data path, packed 24-bit pixels, VGA pixel port, two on-chip programmable clock generators, video clocks up to 220 MHz, on-chip 64x64 hardware cursor, triple monotonic 8-bit DAC's to display 16.8 million colors.

The RGB526, RGB526DB, RGB624 and RGB624DB all improve on the RGB524 with the following common features:

- ❑ Enhanced PLL programming. While retaining the PLL programming features of the RGB524 for compatibility, the RGB526/RGB526DB and RGB624/RGB624DB offer a new mode for programming the PLLs. Called "full M over N", this new mode has larger M and N ranges, and removes restrictions on M and N combinations that exist with the previous generation. Full M over N gives finer control in programming for the desired output frequencies, and allows the internal operating points of the PLLs to be fine tuned.
- ❑ An "advanced function" hardware cursor. This provides the same cursor functions as the RGB524, and in addition provides a "translucent" cursor in which the underlying pixels appear to "shine through" the displayed cursor.
- ❑ 32 BPP Overlay/Underlay layers, with selection based on a "chroma key".
- ❑ Elimination of SCLK to LCLK skew specification (Table 21, "AC Characteristics," value t_{18}). Previous generation parts, may need programming or external hardware to guarantee the required SCLK to LCLK separation. With the RGB526/RGB526DB and RGB624/RGB624DB no special handing is required.
- ❑ Miscellaneous features described below.

In general the RGB524 can be used as the "base feature" set for a graphics adapter, and the additional features of the RGB526/RGB526DB or RGB624/RGB624DB can be added to an existing design by "dropping in" the new product, and enabling the new features by providing additional programming.

The exact differences between the RGB524, RGB526/RGB526DB and RGB624/RGB624DB are detailed below. Several changes have been made to enhance the usability of the parts or to work with the new features that are "non transparent". For drop in replacement of the RGB524 these changes should be examined to determine the effect on existing designs, if any. Users of other members of the RGB51x and RGB52x product family should also examine these changes to determine the effects on programming compatibility. It is felt that for the most part these changes will be considered desirable or non-consequential.

The remaining changes are considered "transparent" because they remove previous restrictions, or they are added features that are enabled by setting register bits that are reserved in the RGB51x and RGB52x products.

C.1 Non-Transparent Changes

C.1.1 ID Register

The ID register (index 0x0001) contains the value 0x02 for the RGB524, RGB526 and RGB526DB. The RGB624 and RGB624DB share the ID value 0x30.

C.1.2 Revision Register

In general, the contents of the Revision register (index 0x0000) varies among the RGB524, RGB526, RGB526DB, RGB624 and RGB624DB products.

C.1.3 DAC Operation- Slew Rate

The DSR bit (DAC Slew Rate) in the DAC Operation register (index 0x0006) is inverted: ‘0’ is now the “fast” setting and ‘1’ is now the “slow” setting.

The power on value for this register is changed from 0x00 to 0x02. This sets the DSR bit for a “slow” slew rate.

C.1.4 Pixel PLL Power On Values

The power on values for three of the Pixel PLL programming registers is changed as follows:

The Fixed Pixel PLL Reference Divider (index 0x0014) is changed from “undefined” to 0x05; the F0 register (index 0x0020) is changed from 0x00 to 0x05; and the F1 register (index 0x0021) is changed from 0x00 to 0x0e.

As described in section 2.2.1, “REFCLK,” these values are chosen for use with a 14.31818 MHz reference clock to produce the VGA frequencies of 25.057 MHz and 28.322 MHz.

C.1.5 Pixel PLL Power On Enable

The power on value for the Miscellaneous Clock Control register (index 0x0002) is changed from 0x00 to 0x01. This sets the PPLL ENAB bit (Pixel PLL Enable) “on” (enabled), so that at power on time the Pixel PLL is running (using the power on programming values described above).

C.1.6 PLL Behavior During Reset

For the RGB524, when $\overline{\text{RESET}}$ is low both the SYSCLK and Pixel PLLs oscillate at some indeterminate frequency, generally in the range of 5 KHz to 250 KHz.

For the RGB526/RGB526DB and RGB624/RGB624DB, when $\overline{\text{RESET}}$ is low both the SYSCLK and Pixel PLLs will oscillate at their power on programmed values. When $\overline{\text{RESET}}$ becomes high the PLLs will continue to oscillate at these same frequencies.

C.1.7 SYSCLK Source

For the RGB524 the source of the SYSCLK output can be either the SYSCLK PLL, or the REFCLK input. The System Clock Control register (index 0x0008), bit 1 (SYSC SRC) selects the SYSCLK source.

For the RGB526/RGB526DB and RGB624/RGB624DB, REFCLK cannot be used as the source of the SYSCLK output. Bit 1 of the System Clock Control register becomes “reserved”.

C.1.8 MISR Operation

For the RGB51x and RGB52x products the diagnostic MISR (Multiple Input Signature Register) is armed to accumulate a signature by “pulsing” the MISR CNTL bit in the Miscellaneous Control 1 register. That is, writing this register to change the MISR CNTL bit from ‘0’ to ‘1’ and back to ‘0’.

For the RGB526/RGB526DB and RGB624/RGB624DB the MISR is armed by changing the MISR CNTL bit from ‘0’ to ‘1’. Also, MISR CNTL must stay at ‘1’ throughout the frame, until the entire signature has been accumulated.

To detect the end of signature accumulation a new register, MISR Status, has been added at index 0x0083. This register contains a single bit, MISR DONE, that indicates when a signature has finished accumulating.

C.2 Transparent Changes

C.2.1 SCLK to LCLK Relationship

For the RGB51x and RGB52x products there is a skew specification that requires the rising edge of LCLK to not be too close to the rising edge of SCLK (a “dead zone” must be maintained.)

For the RGB526/RGB526DB and RGB624/RGB624DB this restriction is eliminated.

C.2.2 SYSCLK Glitch on DF Change

For the RGB524 and RGB528, if the DF programming bits are changed when the SYSCLK programming values are changed then the SYSCLK output can glitch.

For the RGB526/RGB526DB and RGB624/RGB624DB this behavior does not occur. The DF bits (or the P bits, using the new “standard programming mode”) can be changed along with the other programming values, and the SYSCLK frequency will change smoothly to the new value.

C.2.3 Generated Composite Sync

Miscellaneous Control 1 register (index 0x0070) changes bit 2 from “reserved” to XOR SYNC. This bit, when set, generates composite sync internally as the Exclusive OR of the incoming horizontal and vertical sync signals.

C.2.4 VRAM Input Byte Swapping

Miscellaneous Control 3 register (index 0x0072) changes bits 3 and 2 from “reserved” to SWAP BYTE. These bits, when set, cause incoming bytes on the PIX[63:00] inputs to be swapped within byte pairs.

C.2.5 Enhanced PLL Programming

The RGB526/RGB526DB and RGB624/RGB624DB add a new mode for programming the PLL frequencies. This new mode (now called the “standard” mode, or sometimes called “full M over N”), allows a wider range of “M”, “N”, and “P” values (multiply, prescale divide, and postscaler divide). Also, a link between the postscaler divide factor (DF bits in the old mode) and the prescaler divide value is removed, eliminating a programming restriction.

With the new programming mode an additional programming value, “C”, must be supplied to adjust the operating point of the analog circuits.

For compatibility with the RGB51x and RGB52x products the “old” programming mode is retained (now called “restricted” programming). The power on mode is the “old” mode, and the power on programming values are “old” mode values.

System Clock Control register (index 0x0008) changes bit 2 from “reserved” to PROG MODE. This bit, when set, enables the new standard programming mode for the SYSCLK PLL. Also, with the new standard programming, the contents of the two SYSCLK programming registers located at indices 0x0015 and 0x0016 are redefined to supply the N and M values, and two new registers at indices 0x0017 and 0x0018 are added to supply the P and C values.

Pixel PLL Control 1 register (index 0x0010) changes the EXT/INT bit values of 100 and 101 from “reserved” to “Standard Mode”, to enable the new standard programming mode for the Pixel PLL. ‘100’ is for external frequency select and ‘101’ is for internal frequency select.

Pixel PLL Control 2 register (index 0x0011) extends the meaning of the INT FS bits for use with the new standard programming mode, when the EXT/INT bits of Pixel PLL Control 1 are set to ‘101’.

With standard programming the F0 - F15 pixel PLL programming register values are redefined. The old programming mode organizes the F0 - F15 pixel PLL programming registers as 8 banks of 2 programming values, for a 1-of-8 frequency selection. The new standard programming mode organizes F0 - F15 as 4 banks of 4 programming values, for a 1-of-4 frequency selection. This is a consequence of having more programming bits needed for the new programming mode.

For diagnostic readback, with the new standard programming mode the contents of the readback registers at indices 0x008e and 0x008f are redefined to reflect the pixel PLL M and N values, and registers are added at indices 0x008c and 0x008d to reflect the P and C values.

C.2.6 Advanced Cursor

The cursor operation has been extended with the ability to display a “translucent” cursor pixel. With this pixel type the underlying pixel appears to “shine through” the translucent cursor color.

A new register, Advanced Cursor Control (index 0x0037) is added. This register has a single bit, ACA ENAB (Advanced Cursor Attribute Enable) to enable the new cursor operation. The power on value of this bit is “disabled”, so that the standard cursor operation is used following a reset.

A second new register, Advanced Cursor Attribute (index 0x0038) is added. When the advanced cursor operation is enabled the two bit cursor value specifies one of four colors. Each of these colors has a two bit entry in the Advanced Cursor Attribute register which specifies whether the cursor pixel is transparent, solid, translucent, or highlighted.

C.2.7 Double-buffer operation

The register differences for the double-buffer operation of the RGB526DB and RGB624DB are described in appendix A.0.

C.2.8 YUV and Overlays

The register differences for the YUV formats and overlay/underlay layers of the RGB624 and RGB624DB are described in appendix B.0.

D.0 PLL Compatibility Programming

The RGB51x and RGB52x products all have a programmable PLL for generating a pixel clock, and the RGB524 and RGB528 have a second PLL for driving a “SYSCLK” output. All of these PLLs are programmed identically.

As discussed in section 2.6, "PLL Operation Compatibility with RGB51x/RGB52x" on page 6, the RGB624/RGB624DB provides a new method of programming the PLLs. This new method is less restrictive and is preferred for new applications, hence it is now called the “standard” programming method.

But for software compatibility with the RGB51x and RGB52x products the RGB624/RGB624DB additionally provides the same, “old” method of PLL programming. When reset the RGB624/RGB624DB PLL control registers are set for “old” programming, and the programming registers contain power on values appropriate for “old” programming.

This appendix describes the “old” programming modes.

D.1 PLL Programming

The two PLLs are programmed identically. Three values are used:

REF DIV COUNT (Reference Divide Count) This number provides a count value for dividing down the incoming REFCLK. It must be between 2 and 31. Operation of the PLL is indeterminate if this number is 0 or 1.

VCO DIV COUNT (VCO Divide Count) This number provides a count value for the divider in the PLL feedback loop. The value can range from 0 through 63. Internally, 65 is added to VCO DIV COUNT, so that the PLL feedback divider value ranges from 65 through 128.

DF (Desired Frequency) These are two bits with values of 00, 01, 10, and 11. The intent of these bits is to divide the operation of the PLL into four frequency ranges. Following the divide of the REFCLK provided by the REF DIV COUNT there is an additional divide-by-two which is selected or bypassed with the DF bits. Also, the output of the PLL has a divider stage, or postscaler, that is controlled by the DF bits.

Table 26, "PLL Equations," gives the general formulas for programming the PLLs. Because of the action of the DF bits there are four equations, one for each DF bit setting.

It is possible to program the PLLs with values that generate illegal operating conditions:

1. The reference frequency VRF (Video Reference Frequency), which is internal to the PLL, cannot be less than 1 MHz.

Table 26. PLL Equations

DF	Output Frequency	Internal VRF	Max Output Freq. (MHz)	
			170	220
00	$\frac{\text{FREF} \times (\text{VCO DIV COUNT} + 65)}{(\text{REF DIV COUNT}) \times 8}$	$\frac{\text{FREF}}{(\text{REF DIV COUNT}) \times 2}$	42.5	55.0
01	$\frac{\text{FREF} \times (\text{VCO DIV COUNT} + 65)}{(\text{REF DIV COUNT}) \times 4}$	$\frac{\text{FREF}}{(\text{REF DIV COUNT}) \times 2}$	85	110 (*)
10	$\frac{\text{FREF} \times (\text{VCO DIV COUNT} + 65)}{(\text{REF DIV COUNT}) \times 2}$	$\frac{\text{FREF}}{(\text{REF DIV COUNT}) \times 2}$	170 (*)	220 (*)
11	$\frac{\text{FREF} \times (\text{VCO DIV COUNT} + 65)}{\text{REF DIV COUNT}}$	$\frac{\text{FREF}}{\text{REF DIV COUNT}}$	170 (*)	220 (*)

1. FREF = REFCLK frequency
2. Frequencies marked with (*) are maximum pixel PLL frequencies. The SYSCLK PLL maximum output frequency is 100 MHz.

2. The internal VCO (Voltage Controlled Oscillator) cannot exceed the rated speed of the product (170/220 MHz).
3. The SYSCLK output driven by the SYSCLK PLL cannot exceed 100 MHz.
4. The DDOTCLK output driven by the pixel PLL cannot exceed 100 MHz.

Table 26, "PLL Equations," gives the equations for calculating the internal VRF. Table 26 also gives the maximum allowable output frequency for each setting of DF. This reflects the action of the VCO postscaler. If the PLLs are programmed so that these maximum dot clock frequencies are not exceeded then the maximum VCO frequency will not be exceeded.

D.2 PLL Frequency Selection

The REF DIV COUNT, VCO DIV COUNT, and DF bits are provided to the PLLs in a pair of 8-bit registers. REF DIV COUNT is 5 bits and occupies one register, with the 3 high order bits unused. The 6 VCO DIV COUNT bits occupy the second register, with the 2 high order bits used by DF.

For the SYSCLK PLL, the two programming registers are the System PLL Reference Frequency, which holds REF DIV COUNT, and the System PLL VCO Divider, which holds VCO DIV COUNT and the DF bits.

For the pixel PLL, there are actually 17 registers which can be used to hold the programming values: Fixed Pixel PLL Reference Divider and F0 - F15. A pair of registers is selected from this group to provide the pixel PLL programming values. This selection is controlled by the pixel PLL Control 1 and pixel PLL Control 2 registers.

Two different programming styles are supported:

Direct Programming In this scheme only the register holding VCO DIV COUNT and DF is altered to change the frequencies. The register for REF DIV COUNT holds a value that is constant for all frequencies. This method is discussed in more detail below.

M over N In this scheme both register values are changed to program a new frequency. The name refers to the general PLL concept in which

$$Output\ frequency = Input\ reference \times (M/N)$$

where VCO DIV COUNT serves as M and REF DIV COUNT serves as N, with modifications to the equation as shown in Table 26, "PLL Equations."

For the SYSCLK PLL there is not much distinction between the two programming styles. Both registers are written to provide an initial operating frequency. Then to change frequency either one register is changed (System PLL VCO Divider), or both registers are changed, depending on the programming style.

For the pixel PLL, a set of preprogrammed frequencies can be stored in the F0 - F15 registers, and the number of stored frequencies depends on the programming style.

When the direct programming style is used a single REF DIV COUNT is stored in the Fixed PLL Reference Divider register. Up to 16 values of VCO DIV COUNT and DF can be stored in the F0 - F15 registers, allowing 16 preprogrammed pixel clock frequencies.

With the M/N style the Fixed Pixel PLL Reference Divider register is not used. The F0 - F15 are reconfigured as 8 pairs of M and N values (M0,N0,M1,N1, ... M7,N7). This allows 8 preprogrammed pixel clock frequencies.

The selection of the programming registers, either 1 of the 16 F0 - F15 registers or 1 of the 8 M/N pairs, is done either externally with the FS[1:0] inputs to the module, or internally with the INT FS[3:0] bits of Pixel PLL Control 2 register. When the M/N style is used INT FS[3] is ignored.

The programming style and selection source is chosen with the EXT/INT bits of the Pixel PLL Control 1 register, as shown in Table 27, "Pixel PLL Control 1 EXT/INT Freq. Selection."

Table 27. Pixel PLL Control 1 EXT/INT Freq. Selection

EXT /INT	Frequency Selection	REF DIV COUNT	VCO DIV COUNT, DF
000	External FS[1:0]	Fixed Reference Divider	F0-F3
001	External FS[1:0]	N0-N3	M0-M3
010	Internal FS[3:0]	Fixed Reference Divider	F0-F15
011	Internal FS[2:0]	N0-N7	M0-M7

Note that there are more selections available with the internal register INT FS[3:0] than using the external FS[1:0] inputs. With the external inputs, only 4 frequencies can be chosen, using either direct programming or M/N. With the INT FS[3:0] bits 8 frequencies can be chosen using M/N or 16 frequencies can be chosen using direct programming.

Table 28. Direct Programming Reference Divider Values

REFCLK (MHz)	Fixed PLL Reference Divider Register Value
4	0x0002
6	0x0003
8	0x0004
10	0x0005
12	0x0006
14	0x0007
16	0x0008
18	0x0009
20	0x000a
22	0x000b
24	0x000c
26	0x000d
28	0x000e
30	0x000f
32	0x0010
34	0x0011
36	0x0012
38	0x0013
40	0x0014
42	0x0015
44	0x0016
46	0x0017
48	0x0018
50	0x0019
52	0x001a
54	0x001b
56	0x001c
58	0x001d
60	0x001e
62	0x001f

D.3 Direct Programming

Use the following steps to calculate the values used with direct programming:

1. Look up the REFCLK frequency in Table 28, "Direct Programming Reference Divider Values," and write the given programming value into the PLL Reference Divider register (System PLL Reference Divider for the SYSCLK PLL, Fixed Pixel PLL Reference Divider for the pixel PLL). If the incoming REFCLK frequency does not appear in this table, then the direct programming method cannot be used.
2. Use Table 29, "PLL Direct Programming Equations," to determine the values to write into the VCO Divider register (System PLL VCO Divider for the SYSCLK PLL, F0 - F15 for the pixel PLL). First, pick the row of the table whose frequency range covers the frequency of interest. This will determine the value of the DF bits to write. Next, use the given equation to calculate the value of the VCO DIV BITS. Write these two values together to the appropriate register.

   The generated pixel clock frequency is designated in this table as VF, for Video Frequency. Note that within each range the desired VF frequency must lie on a given step value (e.g., with DF = 11 a frequency of 159 MHz is invalid because it does not lie on a 2 MHz step; but either 158 MHz or 160 MHz is valid).

Table 29. PLL Direct Programming Equations

DF	VCO Divide Count	Frequency Range	Step (MHz)
00	(4 x VF) - 65	16.25 - 32 MHz	0.25
01	(2 x VF) - 65	32.5 - 64 MHz	0.5
10	VF - 65	65.0 - 128 MHz	1.0
11	(VF / 2) - 65	130.0 - 220 MHz	2.0
VF = Desired Video Frequency			

D.4 M/N Programming

For the “M over N” programming style use Table 26, "PLL Equations," in the following steps:

1. Select values for REF DIV COUNT, VCO DIV COUNT, and DF that generate the desired frequency (or come close enough). Note that the values 0 and 1 are illegal for REF DIV COUNT under all conditions.
2. Calculate the internal reference frequency VRF. Verify that this frequency is not less than 1 MHz.
3. Verify that the dot clock frequency does not exceed the maximum value specified in the table.
4. If conditions 2 and 3 are not met then the selected values cannot be used.

D.5 General PLL Programming

Fundamentally the only differences between the two programming styles are these:

1. Direct programming can be used only if the REFCLK frequency falls on a 2 MHz boundary from 4 MHz through 62 MHz.
2. With direct programming, for a given pixel clock frequency there is only one set of programming values. These values are obtained from Table 28, "Direct Programming Reference Divider Values," and Table 29, "PLL Direct Programming Equations." Illegal conditions cannot be generated as long as the correct value from Table 28, "Direct Programming Reference Divider Values," is used.
3. M/N can be used with any REFCLK frequency from 2 MHz through 100 MHz
4. A given pixel clock can be generated with multiple combinations of programming values. Some of these values can produce illegal internal conditions. Table 26, "PLL Equations," is used to calculate the resulting pixel clock and to determine if conditions are violated.

If the incoming REFCLK does not meet the requirements for direct programming, then M/N programming must be used for both PLLs.

If REFCLK is suitable for direct programming, then the programming style for the SYSCLK PLL is generally a matter of convenience.

For the pixel PLL, the use of preprogrammed frequencies, and the number required, 8 or 16, can affect the choice of programming styles.

However, the register selection specified with the EXT/INT bits of the Pixel PLL Control 1 register does not necessarily force the selection of programming style, direct or M/N. For example, there is nothing to prevent an arbitrary value from being written into the Fixed Pixel PLL Reference Divider and writing an appropriate value into one of the F0 - F15 registers as calculated with the M/N method. Of course, if multiple “N” values are used and they have to be re-written to the Fixed Pixel PLL Reference Divider every time the FS[3:0] value changes, this defeats the purpose of the FS[3:0] selection mechanism.

Likewise, when the 1-of-8 M/N register selection is used there is nothing to prevent using the direct programming equations from being used for the values. The same value will wind up being used for all of the “N” values.

D.6 Start-up Values

As mentioned previously when the RGB624/RGB624DB is reset the PLLs are initialized for “old” style programming.

The System PLL VCO Divider (index 0x0016) is reset to 0x41. This yields a DF value of ‘01’, and a VCO DIV COUNT value of decimal 1.

The System PLL Reference Divider (index 0x0015) is reset to 0x08. This gives a REF DIV COUNT of decimal 8.

Using the equation in Table 26, "PLL Equations," for DF = 01, the SYSCLK frequency will be REFCLK times:

$$(1 + 65)/(8 \times 4) = 66/32 = 33/16$$

The F0 and F1 (indices 0x0020 and 0x0021) hold the VCO Divider values for the Pixel PLL. F0 is initialized to 0x05, for a DF value of 00 and a VCO DIV COUNT of decimal 5. F1 is initialized to 0x0e, for a DF value of 00 and a VCO DIV COUNT of decimal 14.

The Fixed Pixel PLL Reference Divider (index 0x0014) is initialized to 0x05, for a REF DIV COUNT of decimal 5.

Using the equation in Table 26 for DF = 00, the F0 frequency will be REFCLK times:

$$(5 + 65)/(5 \times 8) = 70/40 = 7/4$$

and the F1 frequency will be REFCLK times:

$$(14 + 65)/(5 \times 8) = 79/40$$

E.0 Switching Into VGA Mode

The RGB624/RGB624DB has two fundamental modes of operation which depend on the input pixel port selected, VGA or VRAM. The port is selected with the “PORT SEL” bit (bit 0) of Miscellaneous Control 2 register.

On the RGB513, RGB514, and RGB525 products, there are two problems that can occur when switching from the VRAM port back to the VGA port. Internally latched data from the previously selected VRAM port can corrupt the VGA data, and the SCLK can stop running momentarily. Both of these problems can be circumvented with a software work-around.

For software compatibility with the RGB513, RGB514, and RGB525 it is recommended that this software work-around also be incorporated in the software used with the RGB624/RGB624DB. The RGB624/RGB624DB does not require this software modification, but it does continue to operate correctly (that is, the software patch is transparent to the RGB624/RGB624DB.)

The software modification is as follows:

When doing a “mode switch” into VGA mode, the following additional steps should be taken:

1. Set bits 1 and 0 to ‘1’s in VRAM Mask Low register, to mask off the lowest VRAM byte. The remaining VRAM Mask bits are “don’t care”.
2. Set bit 6 (VMSK CNTL) in Miscellaneous Control 1 register to ‘1’, to enable the VRAM MASKing.
3. Make sure at least one SCLK occurs. This means setting up the chip for VRAM pixel data operation. In particular, make sure that the Pixel Format register is set to one of the valid formats (4 BPP…32 BPP). A valid pixel format must be set or SCLK will not run.
4. At this point the low byte of the internal VRAM pixel data should be ‘0’s, and will not interfere with the VGA data.

The VGA Port can now be selected. A two step process is required:

1. Write to the Miscellaneous Control 2 register. Set bit 0 (PORT SEL) to ‘0’ for VGA, but write bits 7 and 6 as ‘01’ (PCLK SEL = Internal PLL.)
2. Do a second write to the Miscellaneous Control 2 register. Again, set bit 0 (PORT SEL) to ‘0’ for VGA. But now set bits 7 and 6 to ‘00’ (PCLK SEL = LCLK.) The VGA port is now selected.

When doing a mode switch back to VRAM port operation, make sure that bit 6 (VMSK CNTL) in Miscellaneous Control 1 register is set back to ‘0’, to disable the VRAM MASKing.

The information provided is believed to be accurate and reliable. IBM reserves the right to make changes to the product described without notice. No liability is assumed as a result of its use nor for any infringement of the rights of others.

IBM Microelectronics Division
1580 Route 52, Bldg. 504
Hopewell Junction, NY
12533-6531

The IBM home page can be found at http://www.ibm.com.

The IBM Microelectronics Division home page can be found at
http://www.chips.ibm.com.

Fast Fax Service 415-885-4121

Document No. SC22-9867-00