TVP3033
Data Manual

Video Interface Palette

Literature Number: SLAS149
April 1998

IMPORTANT NOTICE

Texas Instruments (TI) reserves the right to make changes to its products or to discontinue any semiconductor product or service without notice, and advises its customers to obtain the latest version of relevant information to verify, before placing orders, that the information being relied on is current.

TI warrants performance of its semiconductor products and related software to the specifications applicable at the time of sale in accordance with TI's standard warranty. Testing and other quality control techniques are utilized to the extent TI deems necessary to support this warranty. Specific testing of all parameters of each device is not necessarily performed, except those mandated by government requirements.

Certain applications using semiconductor products may involve potential risks of death, personal injury, or severe property or environmental damage ("Critical Applications").

TI SEMICONDUCTOR PRODUCTS ARE NOT DESIGNED, INTENDED, AUTHORIZED, OR WARRANTED TO BE SUITABLE FOR USE IN LIFE-SUPPORT APPLICATIONS, DEVICES OR SYSTEMS OR OTHER CRITICAL APPLICATIONS.

Inclusion of TI products in such applications is understood to be fully at the risk of the customer. Use of TI products in such applications requires the written approval of an appropriate TI officer. Questions concerning potential risk applications should be directed to TI through a local SC sales office.

In order to minimize risks associated with the customer's applications, adequate design and operating safeguards should be provided by the customer to minimize inherent or procedural hazards.

TI assumes no liability for applications assistance, customer product design, software performance, or infringement of patents or services described herein. Nor does TI warrant or represent that any license, either express or implied, is granted under any patent right, copyright, mask work right, or other intellectual property right of TI covering or relating to any combination, machine, or process in which such semiconductor products or services might be or are used.

Contents

List of Illustrations

List of Tables

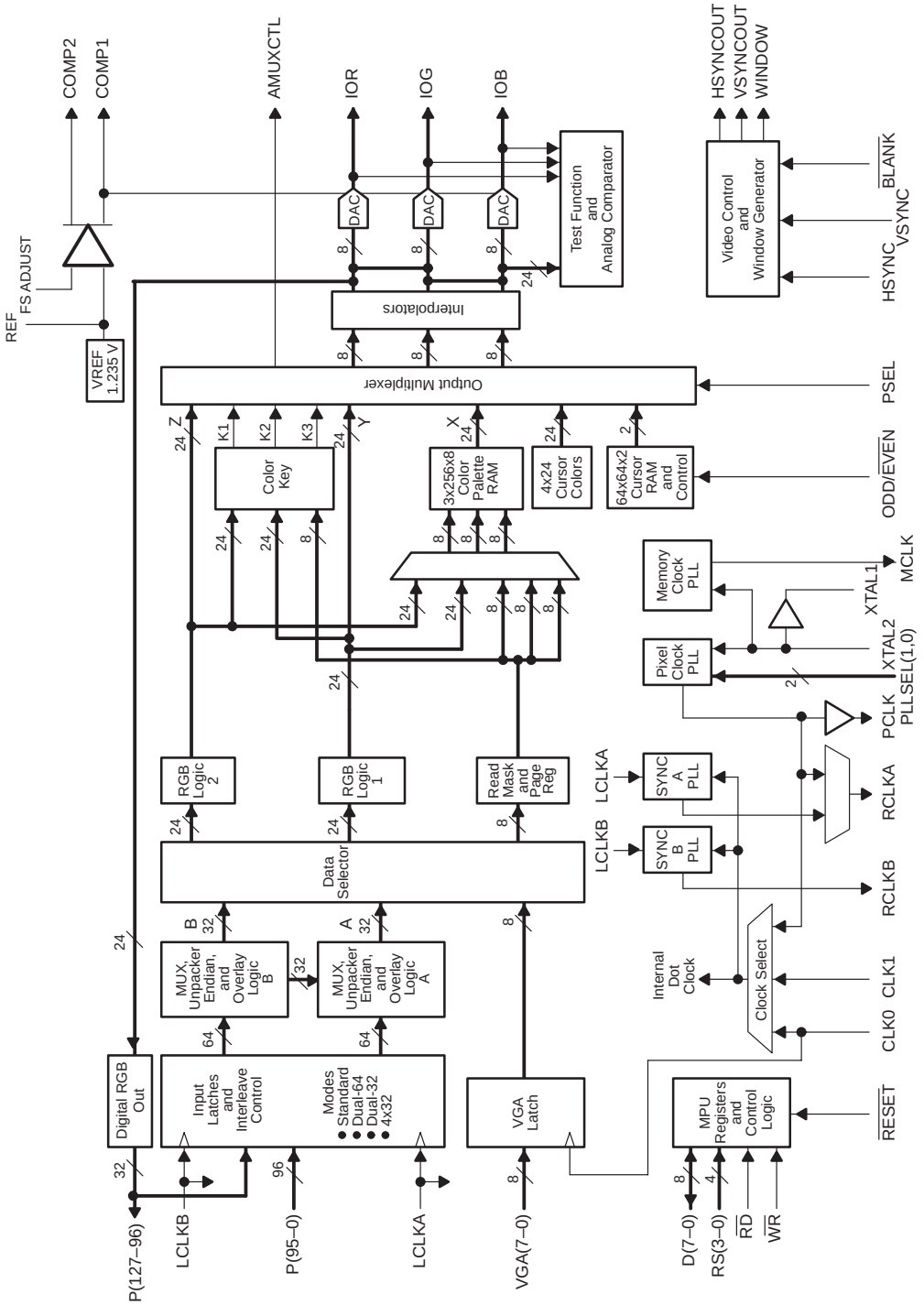
Table *Title* *Page*

1 Introduction

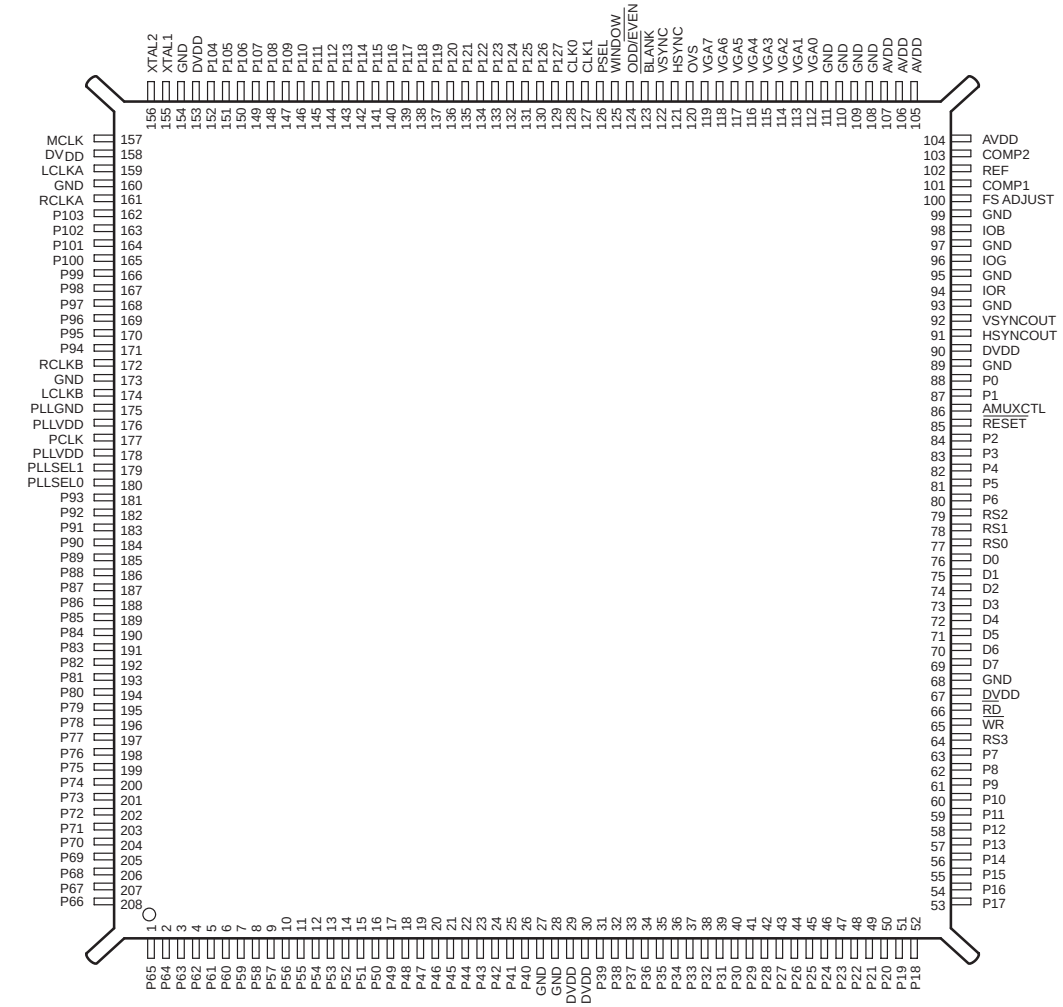
1.1 Features

- Supports System Resolutions up to 1600 × 1280 at 86-Hz Refresh Rate
- RGB Color Depths of 8, 16, 24, and 32 Bits/Pixel, All At Maximum Resolution
- Supports Interpolation for VGA Modes
- 128-Bit Pixel Bus for Shared Frame Buffer Applications
- Supports Dual, Independent 64- or 32-Bit Pixel Ports for Separate Frame Buffer Applications
- Programmable Window Output Controls Pixel Data Flow From Second Frame Buffer
- Gamma Correction
- RGB Modes:
  - 24-Bit/Pixel With 8-Bit Overlay
  - 24-Bit/Pixel Packed-24
  - 16-Bit/Pixel XGA Configuration (5–6–5)
  - 15-Bit/Pixel With 1-Bit Overlay (1–5–5–5)
  - 15-Bit/Pixel Double Buffered (5–5–5)
  - 12-Bit/Pixel Double Buffered (4–4–4)
- Supports WRAM Applications
- 175-, 220-, and 250-MHz Versions
- Power-Saving 3.3-V Supply Operation With 5-V Tolerant I/O
- Programmable Frequency Synthesis PLLs for Dot Clock and Memory Clock
- Two Sync PLLs to Compensate for System Delay and Ensure Reliable Data Latching
- Color Keying
- Hardware Cursor
  - 64 x 64 x 2 Cursor RAM
  - XGA and X-Windows Functional Compatible
- Versatile Pixel Bus Interface Supports Little- and Big-Endian Data Formats
- Triple 8-Bit Monotonic D/A Converters
- Analog Output Comparators for Monitor Detection
- RS-343A Compatible Outputs
- Direct VGA Pass-Through Capability
- Palette Page Register
- Horizontal Zooming Capability
- EPIC™ 0.72 mm CMOS Process

1.2 Functional Block Diagram



1.3 Terminal Assignments



1.4 Ordering Information

TVP3033 – XXX XXX

Pixel Clock Frequency Indicator
Must contain three characters:
–175: 175-MHz pixel clock
–220: 220-MHz pixel clock
–250: 250-MHz pixel clock

Package
Must contain three characters:
–175: PPA plastic quad flatpack
–220: PPA plastic quad flatpack
–250: PPA plastic quad flatpack

1.5 Terminal Functions

TERMINAL NAME	TERMINAL NO.	I/O	DESCRIPTION
AMUXCTL	86	O	Analog multiplexer control. AMUXCTL is the logical combination of the color/luma key and port select functions as specified by the AMUXCTL key logic function register. AMUXCTL is a digital output and is pipeline delayed to be synchronized with the analog output data. AMUXCTL can be used to mix an external analog video stream with the TVP3033 analog outputs.
AV_{DD}	104–107	PWR	Analog power. All AV_{DD} terminals must be connected. A separate cutout in the DV_{DD} plane should be made for AV_{DD}. The DV_{DD} and AV_{DD} planes should be connected only at a single point through a ferrite bead close to where power enters the board.
$\overline{\text{BLANK}}$	123	I	Blank input. $\overline{\text{BLANK}}$ control is normally received from the graphics controller and is latched with CLK0 in VGA modes and is latched with LCLKA otherwise. The $\overline{\text{BLANK}}$ signal is pipeline delayed before being passed to the DACs.
CLK0	128	I	Dot clock 0 TTL input. The CLK0 input can be selected to drive the dot clock at frequencies up to 140 MHz. When using the VGA port, the maximum frequency is 85 MHz. CLK0 is the data latch clock for the VGA port.
CLK1	127	I	Dot clock 1 TTL input. The CLK1 input can be selected to drive the dot clock at frequencies up to 140 MHz.
COMP1, COMP2	101, 103	I	Compensation. COMP1 and COMP2 provide compensation for the internal reference amplifier. A 0.1-uF ceramic capacitor is required between COMP1 and COMP2. The capacitor must be as close to the device as possible to avoid noise pick-up.
D7–D0	69–76	I/O	MPU interface data bus. D7–D0 are used to transfer data in and out of the register map, palette RAM, and cursor RAM.
DV_{DD}	29, 30, 67, 90, 153, 158	PWR	Digital Power. All DV_{DD} terminals must be connected to the digital power plane with sufficient nearby decoupling capacitors.
FS ADJUST	100	I	Full-scale adjustment. A resistor connected between the FS ADJUST terminal and ground controls the full-scale range of the DACs.
GND	27, 28, 68, 89, 93, 95, 97, 99, 108–111, 154, 160, 173	GND	Ground. All GND terminals must be connected. A common ground plane should be used.

NOTE 1: All unused inputs should be tied to a logic level and not be allowed to float. All digital inputs and outputs are TTL compatible unless otherwise stated.

1.5 Terminal Functions (Continued)

TERMINAL NAME	TERMINAL NO.	I/O	DESCRIPTION
HSYNC, VSYNC	121, 122	I	Horizontal and vertical sync inputs. The HSYNC and VSYNC controls are normally received from the graphics controller and are latched with CLK0 in VGA modes and are latched with LCLKA otherwise. The HSYNC and VSYNC signals are pipeline delayed and each may be inverted before being passed to the HSYNCOUT and VSYNCOUT terminals. If HSYNC and VSYNC are used to generate the sync level on the green current output, HSYNC and VSYNC must be active low signals.
HSYNCOUT, VSYNCOUT	91, 92	O	Horizontal and vertical sync outputs. The HSYNCOUT and VSYNCOUT signals are a pipeline delayed version of the selected sync inputs. Output polarity inversion may be independently selected.
IOR, IOG, IOB	94, 96, 98	O	Analog current outputs. The IOR, IOG, and IOB outputs can drive a 37.5-Ω load directly (doubly terminated 75-Ω line), thus eliminating the requirement for external buffering.
LCLKA	159	I	Latch clock input A. The LCLKA input is used to latch pixel bus data and video controls. LCLKA is phase-locked to the internal dot clock by the SYNC A PLL. The external path from RCLKA to LCLKA must be such that linear phase changes in RCLKA cause corresponding linear phase changes in LCLKA.
LCLKB	174	I	Latch clock input B. The LCLKB input is used to latch pixel bus data. LCLKB is phase-locked to the internal dot clock by the SYNC B PLL. The external path from RCLKB to LCLKB must be such that linear phase changes in RCLKB cause corresponding linear phase changes in LCLKB.
MCLK	157	O	Memory clock output. MCLK is the output of an independently programmable PLL frequency synthesizer. The dot clock may be output on the MCLK terminal while the MCLK frequency is reprogrammed to prevent transition effects.
ODD/$\overline{\text{EVEN}}$	124	I	Indicator of odd or even field during interlaced display for cursor operation. A low signal indicates the even field and a high signal indicates the odd field.
OVS	120	I	Overscan border control. OVS is a timing signal that defines a screen border area outside of the horizontal and vertical active display. In the overscan border area, the color stored in the overscan color register is displayed. OVS is active high during the overscan border and during active video.
P127–P0	1–26, 31–63, 80–84, 87, 88, 129–152, 162–171, 181–208	I	Pixel input port. The P127–P0 port can be used in various multiplexing modes. The pixel bus has no internal pull-ups by default. Weak internal pull-ups may be turned on via software. P127–P96 can be configured as outputs to send the digital pixel data to a video encoder or a similar device.
PCLK	177	O	Pixel clock PLL test output. The PCLK output is independent of the dot clock source selected by the CLK CNTL register.
PLLGND	175	GND	Ground for regulated PLL supplies. Decoupling capacitors should be connected between $PLLV_{DD}$ and PLLGND. PLLGND should be connected to the system ground plane through a ferrite bead.
PLLSEL(1,0)	179, 180	I	PLL frequency selection. PLLSEL(1,0) selects among three independently programmed frequency settings for the PCLK and SYNC A PLLs.

NOTE 1. All unused inputs should be tied to a logic level and not be allowed to float. All digital inputs and outputs are TTL compatible unless otherwise stated.

1.5 Terminal Functions (Continued)

TERMINAL NAME	TERMINAL NO.	I/O	DESCRIPTION
$PLLV_{DD}$	176, 178	PWR	PLL power supply. $PLLV_{DD}$ must be a well regulated 3.3-V power supply voltage. Decoupling capacitors should be connected between $PLLV_{DD}$ and PLLGND. Terminal 178 supplies power to the PCLK PLL. Terminal 176 supplies power to the MCLK PLL and the sync PLLs.
PSEL	126	I	Port select. PSEL is a timing signal that can be used to switch between pixel streams. PSEL can be latched by LCLKA or LCLKB.
RCLKA	161	O	Reference clock output A. The RCLKA output can be programmed to output either the pixel clock PLL for use with the VGA port (power-up default), or the SYNC A PLL output for extended modes. RCLKA can be used by the controller to generate CRT controls and the VRAM shift clock. The SYNC A PLL adjusts the phase of RCLKA to phase-lock the received LCLKA with the internal dot clock. RCLKA is not gated off during blank.
RCLKB	172	O	Reference clock output B. The RCLKB output is the SYNC B PLL output. RCLKB can be used by a controller to generate the VRAM shift clock. SYNC B PLL adjusts the phase of RCLKB to phase-lock the received LCLKB with the internal dot clock. RCLKB is not gated off during blank.
$\overline{RD}$	66	I	Read strobe input. A low signal on $\overline{RD}$ initiates a read from the register map. Read transfer data is enabled onto the D(7–0) bus when $\overline{RD}$ is low.
REF	102	I/O	Voltage reference for DACs. An internal voltage reference is provided, which requires an external 0.1-uF ceramic capacitor between REF and analog GND. However, the internal reference voltage can be overdriven by an externally supplied reference voltage.
$\overline{RESET}$	85	I	Master reset. All registers assume their default state after reset. The default state is VGA mode.
RS3–RS0	64, 77–79	I	Register select inputs. RS3–RS0 specify the location in the direct register map that is to be accessed, as shown in Table 2–1.
VGA7–VGA0	112–119	I	VGA port. The VGA7–VGA0 bus can be selected as the pixel input bus for VGA modes. It does not allow for any multiplexing.
WINDOW	125	O	Window control. WINDOW is a timing signal that is active during a programmed rectangular window on the display. WINDOW is used when mixing graphics and video or 2D and 3D graphics to indicate when data should be placed onto the pixel bus.
$\overline{WR}$	65	I	Write strobe input. A low signal on $\overline{WR}$ initiates a write to the register map. Write transfer data is latched from the D(7–0) bus with the rising edge of $\overline{WR}$.
XTAL2, XTAL1	155, 156	I/O	Connection for quartz crystal resonator as a reference for the MCLK and PCLK frequency synthesizer PLLs. XTAL2 may be used as a TTL reference clock input, in which case XTAL1 is left unconnected.

NOTE 1. All unused inputs should be tied to a logic level and not be allowed to float. All digital inputs and outputs are TTL compatible unless otherwise stated.

2 Detailed Description

2.1 MPU Interface

A standard microprocessor interface is supported, giving the MPU direct access to the registers and memories of the TVP3033. The processor interface is controlled via read and write strobes ($\overline{RD}$, $\overline{WR}$), the four register select terminals RS(3–0) and the D(7–0) data terminals. A software selectable 8/6-function is used to select between an 8-bit or 6-bit wide data path to the color palette RAM and is provided in order to maintain VGA compatibility.

Table 2–1 shows the direct register map. These registers are addressed directly by the register select lines RS(3–0). Table 2–2 shows the indirect register map. The index for the indirect register map is loaded into the index register (direct register: 0000). This register is also used as the palette RAM write address and cursor RAM write address. The indexed data register (direct register: 1010) is then used to read or write the register pointed to in the indirect register map. The index register does not post-increment following accesses to the indirect map.

Table 2–1. Direct Register Map

RS3	RS2	RS1	RS0	REGISTER ADDRESSED BY MPU	R/W	DEFAULT	MNEMONIC
0	0	0	0	Palette/Cursor RAM Write Address/Index	R/W		PRAM WAD
0	0	0	1	Palette RAM Data	R/W		PRAM DAT
0	0	1	0	Pixel Mask	R/W	0xFF	PIX MASK
0	0	1	1	Palette/Cursor RAM Read Address	R/W		PRAM RAD
0	1	0	0	Cursor Color Write Address	R/W		COLR WAD
0	1	0	1	Cursor Color Data	R/W		COLR DAT
0	1	1	0	Reserved			
0	1	1	1	Cursor Color Read Address	R/W		COLR RAD
1	0	0	0	Reserved			
1	0	0	1	Direct Cursor Control	R/W	0x00	CUR DCTL
1	0	1	0	Indexed Data	R/W		INDX DAT
1	0	1	1	Cursor RAM Data	R/W		CRAM DAT
1	1	0	0	Cursor Position X LSB	R/W		CUR XL
1	1	0	1	Cursor Position X MSB	R/W		CUR XH
1	1	1	0	Cursor Position Y LSB	R/W		CUR YL
1	1	1	1	Cursor Position Y MSB	R/W		CUR YH

Table 2–2. Indirect Register Map

INDEX REGISTER	REGISTER ADDRESSED BY INDEX	R/W	DEFAULT	MNEMONIC
0x00	Reserved			
0x01	Silicon Revision	R	0x00	SILC REV
0x02–0x05	Reserved			
0x06	Indirect Cursor Control	R/W	0x00	CUR ICTL
0x07–0x0F	Reserved			
0x10	System Configuration	R/W	0x80	SYS CNFG
0x11	Interleave Control	R/W	0x00	ITLV CTL
0x12–0x17	Reserved			
0x18	Pixel Port A Control 1	R/W	0x02	PPA CTL1
0x19	Pixel Port A Control 2	R/W	0x00	PPA CTL2
0x1A	Clock Control	R/W	0x18	CLK CNTL
0x1B	Reserved			
0x1C	Palette Page	R/W	0x00	PAL PAGE
0x1D	General Control 1	R/W	0x00	GEN CTL1
0x1E	Power Down Control	R/W	0x00	PWR CNTL
0x1F	Video Encoder Interface Control	R/W	0x00	VEI CNTL
0x20–23	Reserved			
0x24	Byte Router Control 1	R/W	0x60	BR CTL1
0x25	Byte Router Control 2	R/W	0x18	BR CTL2
0x26–0x2A	Reserved			
0x2B	SYNC B PLL Data	R/W		PLL SNCB
0x2C	Extended Mode PLL Address	R/W		PLL ADDR
0x2D	Extended Mode PCLK PLL Data	R/W		PLL PCLK
0x2E	Memory Clock PLL Data	R/W		PLL MCLK
0x2F	Extended Mode SYNC A PLL Data	R/W		PLL SNCA
0x30	Video Key Mask	R/W		VID MASK
0x31	Video Key	R/W		VID KEY
0x32	Red Range Lower Limit	R/W		RED RNGL
0x33	Red Range Upper Limit	R/W		RED RNGH
0x34	Green Range Lower Limit	R/W		GRN RNGL
0x35	Green Range Upper Limit	R/W		GRN RNGH
0x36	Blue Range Lower Limit	R/W		BLU RNGL
0x37	Blue Range Upper Limit	R/W		BLU RNGH
0x38	Color-Key Control 1	R/W	0x00	KEY CTL1
0x39	Color-Key Control 2	R/W	0x00	KEY CTL2

Table 2–2. Indirect Register Map (Continued)

INDEX REGISTER	REGISTER ADDRESSED BY INDEX	R/W	DEFAULT	MNEMONIC
0x3A	Sense Test	R/W	0x00	SENS TST
0x3B	Test Mode Data	R		TST DATA
0x3C	CRC Remainder LSB	R		CRC LSB
0x3D	CRC Remainder MSB	R		CRC MSB
0x3E	CRC Bit Select	W		CRC SEL
0x3F	Device ID	R	0x33	DEV ID
0x40	Graphics Mask Red	R/W		GM RED
0x41	Graphics Mask Green	R/W		GM GRN
0x42	Graphics Mask Blue	R/W		GM BLU
0x43	Reserved			
0x44	Graphics Key Red	R/W		GK RED
0x45	Graphics Key Green	R/W		GK GRN
0x46	Graphics Key Blue	R/W		GK BLU
0x47	Reserved			
0x48	Pixel Port B Control 1	R/W	0x02	PPB CTL1
0x49	Pixel Port B Control 2	R/W	0x00	PPB CTL2
0x4A–0x4B	Reserved			
0x4C	VGA PLL Address	R/W		VGA ADDR
0x4D	VGA PCLK PLL Data	R/W		VGA PCLK
0x4E	Reserved			
0x4F	VGA SYNC A PLL Data	R/W		VGA SNCA
0x50	Window Start X LSB	R/W		WIN STXL
0x51	Window Start X MSB	R/W		WIN STXM
0x52	Window Width LSB	R/W		WIN WIDL
0x53	Window Width MSB	R/W		WIN WIDM
0x54	Window Start Y LSB	R/W		WIN STYL
0x55	Window Start Y MSB	R/W		WIN STYM
0x56	Window Height LSB	R/W		WIN HGTL
0x57	Window Height MSB	R/W		WIN HGTM
0x58	Reserved			
0x59	Reserved			
0x5A	DACMUX Logic Function 1	R/W	0x00	DAC FCN1
0x5B	DACMUX Logic Function 2	R/W	0x00	DAC FCN2
0x5C	Interpolator Logic Function 1	R/W	0x00	ITP FCN1
0x5D	Interpolator Logic Function 2	R/W	0x00	ITP FCN2
0x5E	AMUXCTL Logic Function 1	R/W	0x00	AMX FCN1
0x5F	AMUXCTL Logic Function 2	R/W	0x00	AMX FCN2
0x60–0xFE	Reserved			
0xFF	Software Reset	W		SOFT RST

2.2 Color Palette RAM

The color palette RAM is a single-port memory that performs a color look-up-table function. The color palette RAM is 24 bits wide for each location and 8 bits wide for each color. Access to the RAM through the MPU port is addressed by an internal 8-bit address register for reading/writing data from/to the RAM. This register is automatically incremented following a RAM transfer, allowing the entire palette to be read/written with only one access of the address register. When the address register increments beyond the last location in RAM, it is reset to the first location (address 0).

Since the RAM is single-ported, anti-sparkle circuitry is provided to prevent sparkling when the RAM is accessed during active video. When a palette RAM write cycle occurs, the pixel color from the RAM is held constant for three dot clocks.

When a palette RAM read cycle occurs, the pixel color from the RAM is held constant for one dot clock.

2.2.1 Writing to Color-Palette RAM

To load the color palette, the MPU must first write to the PRAM WAD register (direct register: 0000) with the address where the modification is to start. The selected palette RAM location is loaded a byte at a time by writing a sequence of three bytes (red, green, and blue) to the PRAM DAT register (direct register: 0001). After the blue write cycle, the palette RAM address increments to the next location, which the MPU may modify by simply writing another sequence of red, green, and blue bytes.

2.2.2 Reading From Color-Palette RAM

Reading from the palette is performed by writing to the PRAM RAD register (direct register: 0011) with the location to be read. Three successive MPU reads from the PRAM DAT register then produces red, green, and blue color data (8 or 6 bits depending on the 8/$\overline{6}$ mode) for the specified location. Following the blue read cycle, the palette RAM address is incremented. While the RAM is read during active display, the pixel color is held constant.

2.2.3 Eight or Six-Bit Mode Selection

The 8-bit or 6-bit DAC resolution is software selectable. The default is 6-bit resolution. The DAC BITS bit in the GEN CTL1 register selects 8-bit resolution (1) or 6-bit resolution (0). This specifies the number of bits used to specify the red, green, and blue color fields stored in the color palette RAM. Since the MPU access is 8 bits wide, the color data stored in the palette is 8 bits even when 6-bit resolution is chosen. If 6-bit resolution is chosen, the two MSBs of color data in the palette RAM have the values previously written. However, if they are read back with 6-bit resolution, the two MSBs are zeros. The output multiplexer after the color palette shifts the six LSB bits to the six MSB positions and fills the two LSBs with zeros, then feeds the data to the DAC.

Since the cursor/overscan color registers are actually physically part of the color palette RAM, they are also affected by the setting of the DAC BITS bit.

2.2.4 Pixel Mask Register

The PIX MASK register (direct register: 0010) is an 8-bit register used to enable or disable a bit plane from addressing the color-palette RAM in the pseudo-color, overlay, and VGA modes. Each palette address bit is logically ANDed with the corresponding bit from the PIX MASK register before going to the PAL PAGE register and addressing the palette RAM.

2.2.5 Palette Page Register

The PAL PAGE register (index: 0x1C) allows selection of multiple color look-up tables stored in the palette RAM when using a mode that addresses the palette RAM with less than eight bits. When using 1 or 4 bit planes in the direct-color + overlay modes, the additional planes are provided by the PAL PAGE register before the data addresses the color palette. This is illustrated in Table 2–3.

NOTE:

The additional bits from the PAL PAGE register are inserted after the pixel mask. The PAL PAGE register specifies the additional bit planes for the overlay field in direct-color modes with less than 8 bits-per-pixel overlay.

Table 2–3. Allocation of Palette Page Register Bits

NUMBER OF BIT PLANES	MSB	PALETTE ADDRESS BITS						LSB
8	M	M	M	M	M	M	M	M
4	P7	P6	P5	P4	M	M	M	M
1	P7	P6	P5	P4	P3	P2	P1	M

Pn = bit n of PAL PAGE
M = bit from pixel port

2.3 Cursor Color Registers

The registers for the four cursor colors are accessed through the direct register map. Since these registers are actually physically part of the color palette RAM, they are also affected by the setting of the DAC BITS bit in the GEN CTL1 register (see Section 2.13.3, *Three-Color 64 × 64 Cursor* for use of the cursor colors).

The COLR WAD register (direct register: 0100) must be initialized before writing to the color registers. The lower two bits of the COLR WAD register select one of the four color registers according to Table 2–5. The selected 24-bit color register is loaded a byte at a time by writing a sequence of three bytes (red, green, and blue) to the COLR DAT register (direct register: 0101). After the blue byte is written, the color address increments to the next color. All four colors may be loaded with a single write to the COLR WAD register followed by 12 consecutive writes to the COLR DAT register.

The COLR RAD register (direct register: 0111) must be initialized before reading from the color registers. The lower two bits of the COLR RAD register select one of the four color registers according to Table 2–5. Next, the COLR DAT register (direct register: 0101) is read three times, producing red, green, and blue bytes from the selected color register. After the blue byte is read, the color address is incremented to the next color. All four colors may be read with a single write to the COLR RAD register followed by 12 consecutive reads of the COLR DAT register.

The sequence followed by the color address counter is shown in Table 2–5. The starting point depends on what was written to the COLR WAD register or the COLR RAD register.

Table 2–4. Cursor Color Address Registers (COLR WAD, COLR RAD)

BIT7	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0
RESERVED						PTR CLR	

Table 2–5. Cursor Color Address Registers (COLR WAD, COLR RAD) Direct: 0100, 0111
Access: R/W Default: Uninitialized

BIT NAME	VALUE	DESCRIPTION
RESERVED		RESERVED
PTR CLR	00: Overscan border color	Pointer to cursor color registers
	01: Cursor color 0	
	10: Cursor color 1	
	11: Cursor color 2	

2.4 Clock Control

The TVP3033 provides two TTL clock inputs (CLK0 and CLK1) which can be used for pixel rates up to 140 MHz. CLK0 must be selected as the dot clock source for VGA modes since CLK0 also serves as the latch clock for the VGA port. In the VGA modes, the PCLK PLL is usually the clock generator and supplies the pixel clock to the controller through the RCLKA terminal. The controller returns the clock to the CLK0 terminal synchronous with the VGA(7–0) data. The received CLK0 is then used to drive the internal dot clock. This is the default configuration which supports immediate operation in VGA modes without software intervention. See Table 2–7 for the CLK CNTL register definition.

The CLK1 input is available for driving the internal dot clock with an additional external clock source. One application for this is when the color palette output is to be synchronized with another video source.

The MCLK STB and MCLK SIG bits are used to route either the MCLK PLL output or the internal dot clock to the MCLK terminal. This may be used to ensure a stable output on the MCLK terminal when changing the MCLK PLL frequency. This procedure is described in Section 2.5.4.1, *Changing the MCLK Frequency*.

The RCKA SIG bits select the signal to output on the RCLK terminal. The PCLK PLL is output at power-up/reset to supply the pixel clock to the graphics controller to support VGA modes. In VGA modes, the graphics accelerator receives RCLKA and returns its VGA output clock to the CLK0 terminal along with synchronous VGA data. The TVP3033 slaves itself to the received CLK0 and uses it as the dot clock source. The RCKA SIG bits should be set to select the SYNC A PLL output for most other modes. When using interpolation with VGA modes, the dot clock/NVALUE option can be used to supply the dot clock divided by two to the controller. The SYNC A PLL N register should be programmed to two in this case.

Table 2–6. Clock Control Register (CLK CNTL)

BIT7	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0
RSVD	RCKA SIG		MCLK SIG	MCLK STB	LCLKSRC	CLKSRC	

Table 2–7. Clock Control Register (CLK CNTL)Index: 0x1A Access: R/W Default: 0x18

BIT NAME	VALUE	DESCRIPTION
RESERVED		
RCKA SIG	00: PCLK PLL output (supports VGA mode)	RCLKA output terminal signal selection
	01: SYNC A PLL output	
	10: Dot clock	
	11: Dot clock / NVALUE. Using NVALUE of SYNC A PLL.	
MCLK SIG	0: Dot clock (to ensure stable output when changing MCLK)	MCLK output terminal signal selection
	1: MCLK PLL output	
MCLK STB	0: Low-to-high transition on MCLK STB causes MCLK SIG to take effect. MCLK SIG should not change during low-to-high transition on MCLK STB.	Strobe for changing MCLK SIG
	1: Default	
LCLKSRC	0: LCLK is taken from LCLK input terminals	LCLKA, LCLKB source selection
	1: LCLKA, LCLKB are taken from CLK0 input. This is for 2× zoom for interpolation in VGA mode.	
CLKSRC	00: CLK0	Dot clock source selector
	01: CLK1	
	10: PCLK PLL	
	11: Reserved	

2.5 PLL Clock Generators

The TVP3033 has four on-chip, fully programmable phase-locked loops (PLLs). The first PLL (PCLK PLL) is intended for pixel clock generation for frequencies up to the device limit. The second PLL (MCLK PLL) is provided for general clocking such as the system clock or memory clock. The third and fourth PLLs (SYNC A PLL and SYNC B PLL) are provided for synchronizing pixel data and latch timing by compensating for system loop delay.

The clock generators enable a wide range of precise frequencies. The advanced PLLs utilize an internal loop filter to provide maximum noise immunity and reduce jitter. Except for the reference crystal or oscillator, no external components or adjustments are necessary. Each PLL can be independently enabled or disabled for maximum system flexibility. The TVP3033 clocking scheme is shown in Figure 2–1.

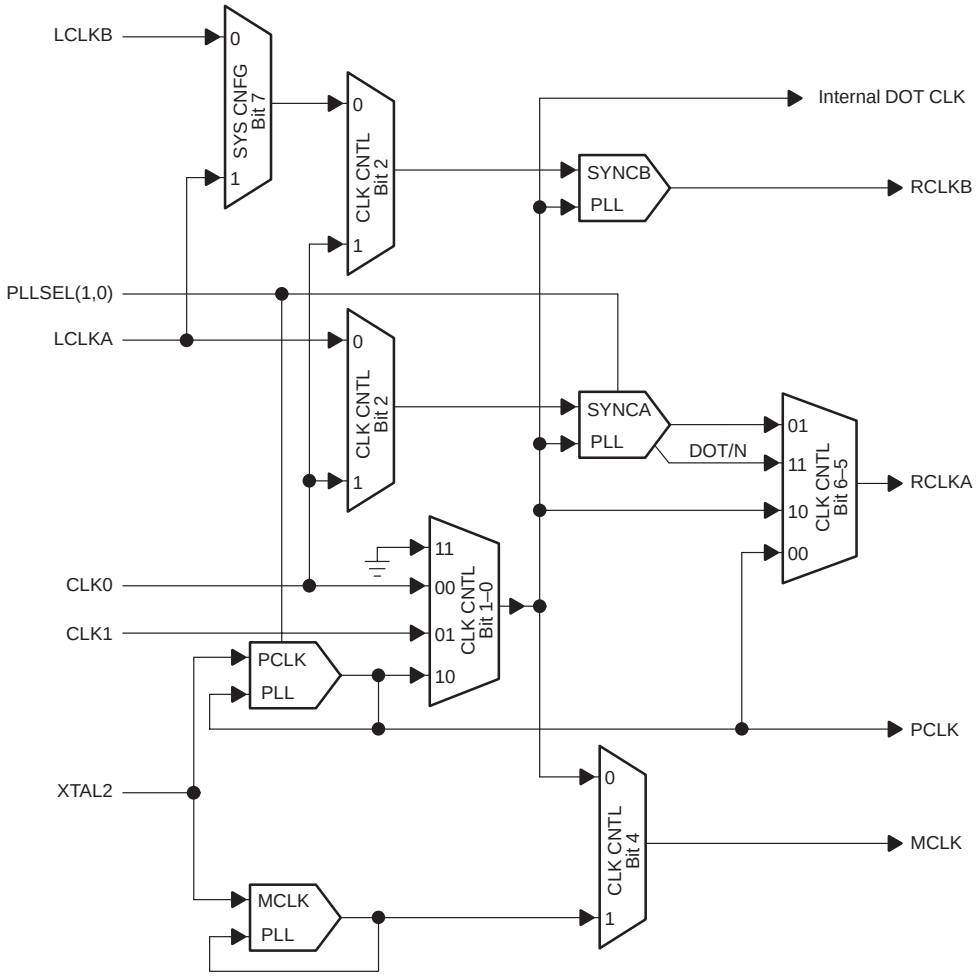


Figure 2–1. TVP3033 Clocking Scheme

2.5.1 Pixel Clock Frequency Selection

The PLLSEL(1,0) inputs provide a hardware controlled means of selecting the pixel clock frequency, which is required to maintain VGA compatibility. Table 2–8 shows that PLLSEL(1,0) selects one of three programmable register sets for the PCLK PLL and SYNC A PLL. These are the VGA 0, VGA 1, and extended mode register sets. The default frequencies shown are output on the RCLKA and PCLK terminals at power-up/reset depending on the state of PLLSEL(1,0).

The MCLK PLL and SYNC B PLL are not effected by PLLSEL(1,0) and have only the extended mode programmable register set. The MCLK PLL defaults to 50.11 MHz. The SYNC B PLL defaults to passing the dot clock.

Table 2–8. Pixel Clock Frequency Selection

PLLSEL(1,0)	PCLK PLL, SYNC A PLL REGISTER SET	PCLK PLL DEFAULT FREQUENCY†	SYNC A PLL DEFAULT FREQUENCY
00	VGA 0	25.185 MHz	Pass dot clock
01	VGA 1	28.311 MHz	Pass dot clock
1X	Extended Mode	25.185 MHz	Pass dot clock

† With the standard 14.31818 MHz reference crystal. For other crystals, the default frequency is proportional to the crystal frequency.

2.5.2 PLL Register Sets

The extended mode PLL register sets are accessed through the five registers shown in Table 2–9. These registers are used to program the extended mode register set for the PCLK, MCLK, SYNC A and SYNC B PLLs. The PCLK and SYNC A PLLs have two additional register sets (VGA 0 and VGA 1) to provide programmable frequencies for the VGA modes. These are accessed through the three registers shown in Table 2–10.

Table 2–9. Extended Mode PLL Registers

INDEX	R/W	REGISTER ADDRESSED BY INDEX	MNEMONIC
0x2B	R/W	SYNC B PLL data	PLL SNCB
0x2C	R/W	Extended mode PLL address	PLL ADDR
0x2D	R/W	Extended mode PCLK PLL data	PLL PCLK
0x2E	R/W	MCLK PLL data	PLL MCLK
0x2F	R/W	Extended mode SYNC A PLL data	PLL SNCA

Table 2–10. VGA Mode PLL Registers

INDEX	R/W	REGISTER ADDRESSED BY INDEX	MNEMONIC
0x4C	R/W	VGA PLL address	VGA ADDR
0x4D	R/W	VGA PCLK PLL data	VGA PCLK
0x4F	R/W	VGA SYNC A PLL data	VGA SNCA

Each PLL register set contains four registers (N, M, P, and status). The PLL ADDR register is used to point to the extended mode register set of each PLL. The PLL ADDR register allows read and write access and contains four 2-bit pointers, one for each PLL, according to Table 2–12. The extended mode PLL registers are then accessed through the PLL PCLK, PLL MCLK, PLL SNCA and PLL SNCB registers (index: 0x2D–0x2F and 0x2B).

The VGA ADDR register is used to point to the two VGA register sets of the PCLK and SYNC A PLLs. The VGA ADDR register allows read and write access and contains two 3-bit pointers, according to Table 2–14. The VGA mode PLL registers are then accessed through the VGA PCLK and VGA SNCA registers (index: 0x4D and 0x4F).

The PLL register pointers are independently auto-incremented following a write cycle to the corresponding PLL data register. The PLL register pointers do not auto-increment following a read cycle of the PLL data registers. The most efficient way to program the PLLs is to first write zeros to the PLL register pointer followed by three consecutive writes to the PLL data register to program the N, M, and P registers. Following the third write, the PLL register pointer points to the read-only status register. The status register can then be polled until the LOCK bit is set (the pointer does not auto-increment on reads).

Table 2–11. Extended Mode PLL Address Register (PLL ADDR)

BIT7	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0
PTR SNCB		PTR SNCA		PTR MCLK		PTR PCLK	

Table 2–12. Extended Mode PLL Address Register (PLL ADDR)
Index: 0x2C Access: R/W Default: Uninitialized

BIT NAME	VALUE	DESCRIPTION
PTR SNCB	00: N register	Pointer to SYNC B PLL registers
	01: M register	
	10: P register	
	11: Status register	
PTR SNCA	00: N register	Pointer to SYNC A PLL extended registers
	01: M register	
	10: P register	
	11: Status register	
PTR MCLK	00: N register	Pointer to MCLK PLL registers
	01: M register	
	10: P register	
	11: Status register	
PTR PCLK	00: N register	Pointer to PCLK PLL extended registers
	01: M register	
	10: P register	
	11: Status register	

Table 2–13. VGA Mode PLL Address Register (VGA ADDR)

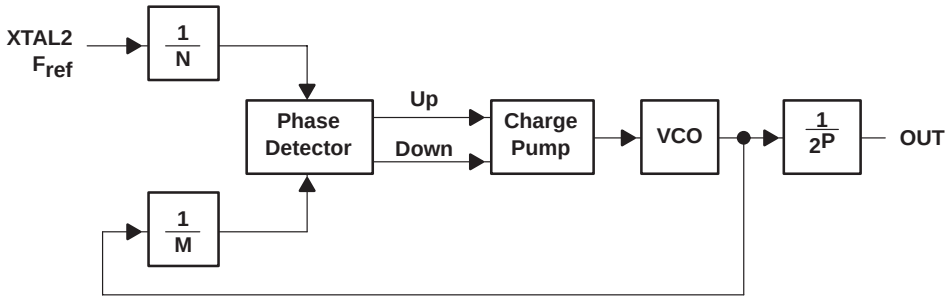
BIT7	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0
RESERVED		VGA SPTR			VGA PPTR		

Table 2–14. VGA Mode PLL Address Register (VGA ADDR)
Index: 0x4C Access: R/W Default: Uninitialized

BIT NAME	VALUE	DESCRIPTION
RESERVED		
VGA SPTR	000: VGA 0 N register	Pointer to SYNC A PLL VGA register sets
	001: VGA 0 M register	
	010: VGA 0 P register	
	011: VGA 0 status register	
	100: VGA 1 N register	
	101: VGA 1 M register	
	110: VGA 1 P register	
	111: VGA 1 status register	
VGA PPTR	000: VGA 0 N register	Pointer to PCLK PLL VGA register sets
	001: VGA 0 M register	
	010: VGA 0 P register	
	011: VGA 0 status register	
	100: VGA 1 N register	
	101: VGA 1 M register	
	110: VGA 1 P register	
	111: VGA 1 status register	

2.5.3 PCLK PLL

The PCLK PLL can be used at frequencies up to the device limit. A model of the PCLK PLL is shown in Figure 2–2. The resulting equations describe the voltage controlled oscillator frequency and the PLL output frequency as a function of the N, M, and P values and the reference frequency.



$$F_{VCO} = F_{ref} \times \frac{M}{N} \qquad F_{OUT} = \frac{F_{VCO}}{2^P}$$

Provided : Minimum VCO Frequency $\leq F_{VCO} \leq$ Maximum VCO Frequency

Figure 2–2. Sync PLL Operation

The N, M, and P registers can be programmed to any value within the following limits:

$2 \leq NVALUE \leq 7$
$2 \leq MVALUE \leq 255$
$0 \leq PVALUE \leq 4$

If several N, M, and P selections meet the above criteria, the selection with the smallest NVALUE should be used. The bit assignments of the N, M, P, and status registers for the PCLK PLL are listed in Table 2–16.

Table 2–15. PCLK PLL N, M, P, and Status Registers (PLL PCLK, VGA PCLK)

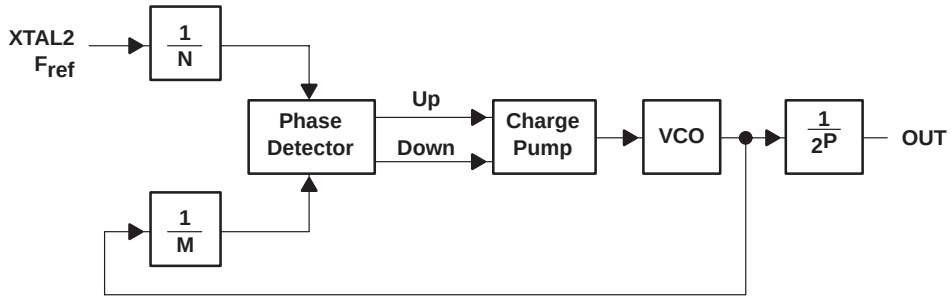
BIT7	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0
NVALUE							
MVALUE							
PLLEN	PCLKEN	RESERVED			PVALUE		
RSVD	LOCK	RESERVED					

Table 2–16. PCLK PLL N, M, P, and Status Registers (PLL PCLK, VGA PCLK) Index: 0x2D, 0x4D Access: R/W Default: PLLSEL(1,0) = 00/10/11 N=0x0E, M=0xC5, P=0x83, PLLSEL(1,0) = 01 N=0x0B, M=0x57, P=0x82

BIT NAME	VALUE	DESCRIPTION
N REGISTER ACCESS: R/W		
NVALUE	0x02–0x07: Valid frequency division values (2–7 decimal)	N frequency prescaler
M REGISTER ACCESS: R/W		
MVALUE	0x02–0xFF: Valid frequency division values (2–255 decimal)	M frequency prescaler
P REGISTER ACCESS: R/W		
PLLEN	0: PLL disabled. VCO does not oscillate.	PLL enable
	1: PLL enabled	
PCLKEN	0: PCLK output terminal is logic 0	PCLK output enable
	1: PCLK output terminal is driven by PCLK PLL	
RESERVED	000: Always program to 000	Reserved
PVALUE	0–4: Actual frequency division is 2**PVALUE. Valid division factors are 1, 2, 4, 8, and 16.	P frequency post-scaler
STATUS REGISTER ACCESS: READ ONLY		
RESERVED		Reserved
LOCK	0: VCO is not locked to the selected frequency	VCO locked
	1: VCO is locked to the selected frequency	
RESERVED		Reserved

2.5.4 MCLK PLL

The memory clock PLL (MCLK PLL) can be used at frequencies up to 100 MHz. The MCLK PLL maximum output frequency of 100 MHz cannot be exceeded. A model of the MCLK PLL is shown in Figure 2–3 and it is identical to the model of the PCLK PLL. The resulting equations describe the voltage controlled oscillator frequency and the PLL output frequency as a function of the N, M, and P values and the reference frequency.



$$F_{VCO} = F_{ref} \times \frac{M}{N} \qquad F_{OUT} = \frac{F_{VCO}}{2^P}$$

Provided : Minimum VCO Frequency $\leq F_{VCO} \leq$ Maximum VCO Frequency

Figure 2–3. PCLK and MCLK PLL Model

The N, M, and P registers can be programmed to any value within the following limits:

$2 \leq \text{NVALUE} \leq 7$
$2 \leq \text{MVALUE} \leq 255$
$0 \leq \text{PVALUE} \leq 4$

If several N, M, and P selections meet the above criteria, the selection with the smallest NVALUE should be used. The bit assignments of the N, M, P, and status registers for the MCLK PLL are listed in Table 2–18.

Table 2–17. MCLK PLL N, M, P and Status Registers (PLL MCLK)

BIT7	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0
NVALUE							
MVALUE							
PLLEN	RESERVED				PVALUE		
RSVD	LOCK	RESERVED					

Table 2–18. MCLK PLL N, M, P, and Status Registers (PLL MCLK) Index: 0x2E Access: R/W Default: N=0x02, M=0x1C, P=0x82

BIT NAME	VALUE	DESCRIPTION
N REGISTER ACCESS: R/W		
NVALUE	0x02–0x0E: Valid frequency division values (2–14 decimal)	N frequency prescaler
M REGISTER ACCESS: R/W		
MVALUE	0x02–0xFF: Valid frequency division values (2–255 decimal)	M frequency prescaler
P REGISTER ACCESS: R/W		
PLLEN	0: PLL disabled. VCO does not oscillate.	PLL enable
	1: PLL enabled	
RESERVED	0000: Always program to 0000	
PVALUE	0–4: Actual frequency division is 2**PVALUE. Valid division factors are 1, 2, 4, 8, and 16.	P frequency postscaler
STATUS REGISTER ACCESS: READ ONLY DEFAULT: UNINITIALIZED		
RESERVED		
LOCK	0: VCO is not locked to the selected frequency	VCO locked
	1: VCO is locked to the selected frequency	
RESERVED		

2.5.4.1 Changing the MCLK Frequency

The MCLK frequency is normally used as the graphics controller system clock and memory clock. During reprogramming of the PLLs a wide range of frequencies are generated as the PLL transitions to the new programmed frequency. These transition effects can produce unwanted results in some systems. The TVP3033 provides a mechanism for smooth transitioning of the MCLK PLL. The following programming steps are recommended:

1. Program the PCLK PLL to the same frequency to which MCLK will be changed to, and poll the PCLK PLL status until LOCK = 1.
2. Select the PCLK PLL as the dot clock source if it is not already selected.
3. Output the dot clock on the MCLK terminal by writing bits MCLK SIG, MCLK STB in the CLK CNTL register to 0,0 followed by 0,1.
4. Program the MCLK PLL for the new frequency and poll the MCLK PLL status until LOCK = 1.
5. Output MCLK on the MCLK terminal by writing bits MCLK SIG, MCLK STB to 1,0 followed by 1,1.
6. Reprogram the PCLK PLL to its operating frequency.

2.5.5 Synchronizer PLLs

In high-performance graphics systems, the interface between the controller, frame buffer, and video palette requires special treatment to guarantee reliable operation over all supported resolutions and refresh rates. These systems are particularly sensitive to variations in graphics accelerator propagation delays from device to device which can produce severe production problems at the board level. Since the video palette is the source of the highest frequency clocks in the system, it becomes very difficult to resynchronize the received pixel data with the internal dot clock. To solve this problem, the TVP3033 has incorporated unique synthesizer PLL circuits (SYNC A PLL and SYNC B PLL) to synchronize the pixel data and load clock with the internal dot clock.

The SYNC A PLL is used both for the split pixel bus modes and standard modes. Three sets of programmable registers are provided to support two VGA settings and an extended mode setting. The PLLSEL(1,0) inputs select which register set is used. A multiplexer is included to route the SYNC A PLL output or the PCLK PLL output to the RCLKA terminal. At power-up/reset, the PCLK PLL output is output to support VGA modes. In VGA modes, the graphics accelerator receives the RCLKA signal and returns its VGA output clock to the CLK0 terminal along with synchronous VGA data. The TVP3033 slaves itself to the received CLK0 signal and uses it as the dot clock source. The SYNC A PLL should be output on RCLKA for most other modes. When using interpolation with VGA modes, the PCLK PLL output is used as the dot clock. The RCLKA output is the output of SYNC A PLL. The RCLKA is given back by the controller as CLK0. The CLK0 signal is then routed to the SYNC A PLL through a multiplexer. The CLK0 signal is then synchronized with the dot clock by using M = 4, N = 2, and P = 2 in the SYNC A PLL registers. This setting supplies a synchronized dot clock divided by two to the controller.

The SYNC B PLL is used to support the split pixel bus modes (dual-64 and dual-32 modes). In these modes, two reference clocks and two load clocks are used which can be a different frequency division of the internal dot clock. A single register set is provided for the SYNC B PLL. The SYNC B PLL output drives the RCLKB terminal directly. At power-up/reset, the dot clock frequency is output.

The pixel data latching structure of the TVP3033 is shown in Figure 2–4. The PCLK PLL signal is selected as the dot clock source whenever the SYNC PLLs are used. The dot clock is used as a reference frequency by the SYNC PLL and is prescaled as specified by the N register. The incoming LCLKx signal is used as the other input of the PLL and is prescaled as specified by the M register. The PLL generates the RCLKx signal with the proper frequency and phase shift to phase align the prescaled dot clock and prescaled LCLKx signal. The first two pixel bus pipeline stages are latched with the rising edge of the LCLKx signal and all subsequent stages are latched with the dot clock.

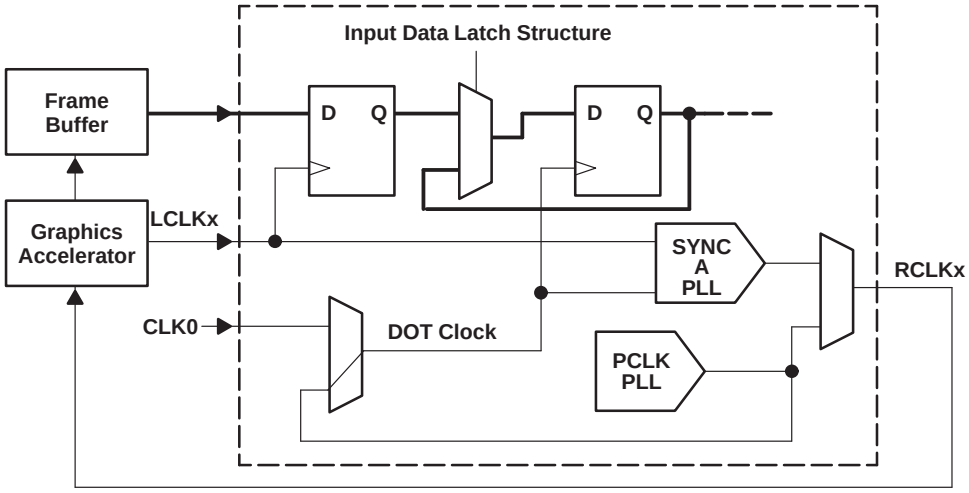


Figure 2–4. SYNC PLL Operation

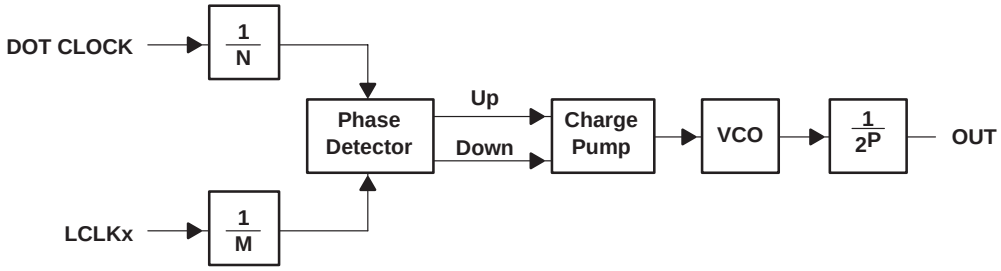


Figure 2–5. SYNC PLL Model

The bit assignments of the N, M, P, and status registers for the SYNC PLLs are listed in Table 2–20. The N, M, and P registers can be programmed to any value within the following limits:

$2 \leq \text{NVALUE} \leq 255$
$2 \leq \text{MVALUE} \leq 255$
$0 \leq \text{PVALUE} \leq 7$

The LCLK edge synchronizer function inserts a delay of the specified number of LCLKs, relative to the end of BLANK, at which time synchronization with the internal dot clock is achieved. This function is enabled by setting the ES ENBL bit to 1 and is required for all packed-24 modes and whenever the dual-64 or dual-32 configurations are used. For a standard configuration with nonpacked modes, this function has no effect. The EDGE SNC bits are programmable but should always be set to 01 for proper operation in split pixel bus configurations and packed-24 modes.

Table 2–19. SYNC A PLL and SYNC B PLL N, M, P, and Status Registers (PLL SNCA, VGA SNCA, PLL SNCB)

BIT7	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0
NVALUE							
MVALUE							
PLLEN	RSVD	EDGE SNC		ES ENBL	PVALUE		
RSVD	LOCK	RESERVED					

Table 2–20. SYNC A PLL and SYNC B PLL N, M, P, and Status Registers (PLL SNCA, VGA SNCA, PLL SNCB)
Index: 0x2F, 0x4F, 0x2B Access: R/W Default: N=0x02, M=0x10, P=0x9B (Pass dot clock)

BIT NAME	VALUE	DESCRIPTION
N REGISTER ACCESS: R/W		
NVALUE	0x02–0xFF: Valid frequency division values (2–255 decimal)	N frequency prescaler
M REGISTER ACCESS: R/W		
MVALUE	0x02–0xFF: Valid frequency division values (2–255 decimal)	M frequency prescaler
P REGISTER ACCESS: R/W		
PLLEN	0: PLL disabled. VCO does not oscillate.	PLL enable
	1: PLL enabled	
RESERVED	1: Always program to 1	
EDGE SNC	0x00–0x02: Selects alignment of the nth LCLK rising edge from start of BLANK with the internal dot clock. Applies only to SYNC A PLL.	LCLK edge synchronizer delay in LCLKs
ES ENBL	0: Disabled	LCLK edge synchronizer function enable
	1: Enabled	
PVALUE	0–7: Actual frequency division is 2**PVALUE. Valid selections are 1, 2, 4,...128.	P frequency postscaler
STATUS REGISTER ACCESS: READ ONLY		
RESERVED		
LOCK	0: VCO is not locked to the selected frequency	VCO locked
	1: VCO is locked to the selected frequency	
RESERVED		

2.6 Pixel Bus Interface

The pixel bus interface supports four primary configurations: standard, dual-64, dual-32, and 4 × 32. These are illustrated in Figure 2–6. The configurations having pixel ports A and B are capable of receiving two independent pixel streams. Ports A and B may load pixel data using a different frequency division of the dot clock.

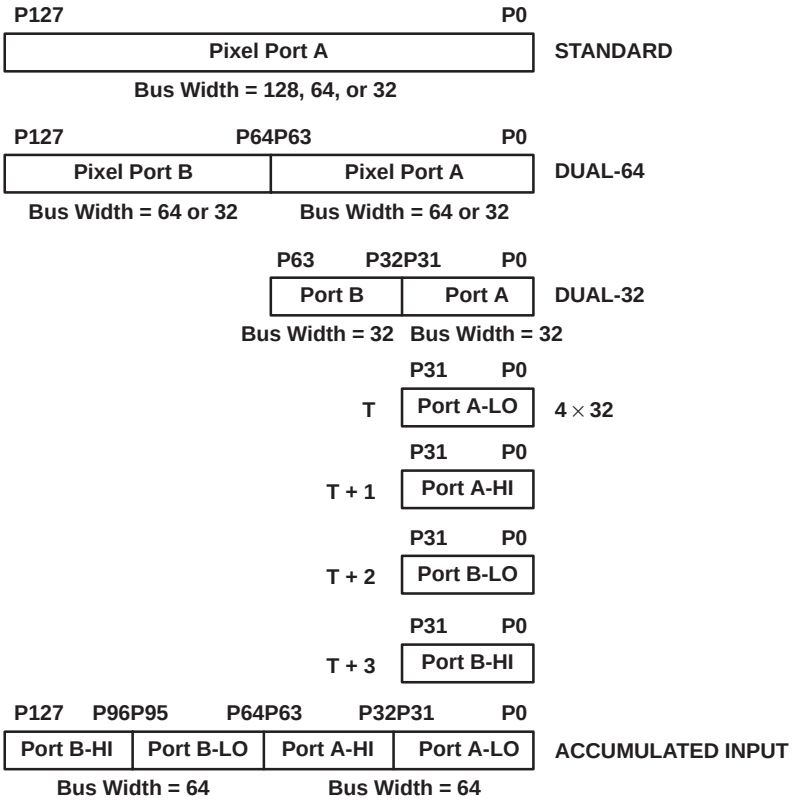


Figure 2–6. Pixel Bus Configurations

2.6.1 Standard Configuration

The standard configuration supports the shared-memory system architecture. The entire 128-bit pixel bus is used as a single port (pixel port A).

The interleave mode and packed/unpacked mode are selected for pixel port A. After these operations, the pixel port A data is multiplexed onto the 32-bit A bus. This is the input to the data selector block. The data type and the specific multiplexing mode are programmed for pixel port A. A maximum bus width of 128 bits can be used.

The SYNC A PLL is used to generate the reference clock to the controller (RCLKA). LCLKA is used to latch the received pixel data on P(127–0). RCLKB and LCLKB are not used.

2.6.2 Dual-64 Configuration

The dual-64 configuration supports the separate frame-buffer system architecture. The 128-bit pixel bus is split into two independent 64-bit pixel ports (A and B). Pixel port A receives data on P(63–0) and pixel port B receives data on P(127–64).

The interleave mode and packed/unpacked mode can be independently selected for pixel port A and pixel port B. After these operations, the pixel port A data is multiplexed onto the 32-bit A bus and the pixel port B data is multiplexed onto the 32-bit B bus. The data type and the specific multiplexing mode are independently programmed for pixel port A and pixel port B. A maximum bus width of 64 bits can be used for each pixel port.

Since the device has one palette RAM, both pixel port A and pixel port B cannot both operate in a mode requiring use of the color palette RAM. Pixel ports A and B can latch data at a different frequency, each using a different division of the internal dot clock. Independent clocking and SYNC PLLs (RCLKA, LCLKA, SYNC A PLL and RCLKB, LCLKB, SYNC B PLL) are provided to facilitate communication with two separate controllers.

2.6.3 Dual-32 Configuration

The dual-32 configuration operates much like the dual-64 configuration, but pixel ports A and B are each a maximum of 32 bits wide. The lower half of the 128-bit pixel bus is split into two independent 32-bit pixel ports (A and B). Pixel port A receives data on P(31–0) and pixel port B receives data on P(63–32).

The interleave mode and packed/unpacked mode can be independently selected for pixel port A and pixel port B. After these operations, the pixel port A data is multiplexed onto the 32-bit A bus and the pixel port B data is multiplexed onto the 32-bit B bus. The data type and the specific multiplexing mode are independently programmed for pixel port A and pixel port B. A maximum bus width of 32 bits can be used for each pixel port.

Both pixel port A and pixel port B cannot both operate in a mode requiring use of the color palette RAM. Pixel ports A and B can latch data at a different frequency, each using a different division of the internal dot clock. Independent clocking and SYNC PLLs (RCLKA, LCLKA, SYNC A PLL and RCLKB, LCLKB, SYNC B PLL) are provided to facilitate communication with two separate controllers.

2.6.4 4×32 Configuration

The 4×32 configuration supports dual pixel ports using only P(31–0). Pixel data is clocked in on P(31–0) on four consecutive rising edges of the LCLKA signal. The first and second words latched in are the lower half and upper half of the pixel port A data. The third and fourth words latched in are the lower half and upper half of the pixel port B data.

The interleave mode (none or 16-bit interleave) is programmed using pixel port A. Packed mode is not supported. After these operations, the pixel port A data is multiplexed onto the 32-bit A bus and the pixel port B data is multiplexed onto the 32-bit B bus. The data type and the specific multiplexing mode are independently programmed for pixel port A and pixel port B. A maximum bus width of 64 bits can be used for each pixel port.

Both pixel port A and pixel port B cannot both operate in a mode requiring use of the color palette RAM. The SYNC A PLL is used to generate the reference clock RCLKA to the controller. LCLKA is used to latch the received pixel data on P(31–0) for both pixel port A and pixel port B. RCLKB and LCLKB are not used.

2.7 Pixel Bus De-Interleave for WRAM Applications

A generic pixel bus de-interleave scheme provides support for systems using window RAM (WRAM) for frame buffer memory. WRAM is a dual-port memory similar to video RAM (VRAM). The WRAM device achieves a higher serial output clock rate by multiplexing the serial outputs onto half the normal number of terminals. As a result, data written to the WRAM in a linear fashion comes out of the serial outputs in some form of interleave scheme, depending on the memory configuration and graphics controller design. The degree to which the color palette device must provide de-interleave capability depends on how much of this function is performed by the graphics controller. The pixel bus de-interleave function of the TVP3033 can perform all, none, or any portion of the de-interleave task.

The de-interleave scheme (see Figure 2–7) assumes that two consecutive words are latched into the pixel bus. These two words have the even data groups on the first word, and the odd data groups on the second word. The size of the groups can be 16, 32, or 64 bits. After rearranging the even and odd data groups, two data words of the same size as was input are passed on with the data groups in the proper order to be displayed.

Table 2–21 specifies the de-interleave modes that can be used in each of the pixel bus configurations and bus width selections. The de-interleave mode is selected by programming the ITLV A and ITLV B bits in the ITLV CTL register. For dual-64 and dual-32 configurations, the de-interleave mode is independently programmable for pixel ports A and B and the bus width in Table 2–21 refers to the bus width for a single pixel port. For other modes, only the ITLV A bits are used. For a 4×32 configuration, pixel ports A and B are used and each uses a 64-bit bus width.

The pixel bus data format tables in Appendix A, *Pixel Bus Data Formats* do not comprehend the de-interleave function. Therefore, the data in linear order (after the de-interleave function) corresponds to the pixel bus bit positions shown in Tables A1 through A10 in Appendix A, *Pixel Bus Data Formats*.

Table 2–21. Available De-Interleave Modes

PIXEL BUS CONFIGURATION	BUS WIDTH	DE-INTERLEAVE MODES			
		NONE	×16	×32	×64
Standard	128	√	√	√	√
	64	√	√	√	
	32	√	√		
Dual-64	64	√	√	√	
	32	√	√		
Dual-32	32	√	√		
4×32	64	√	√		

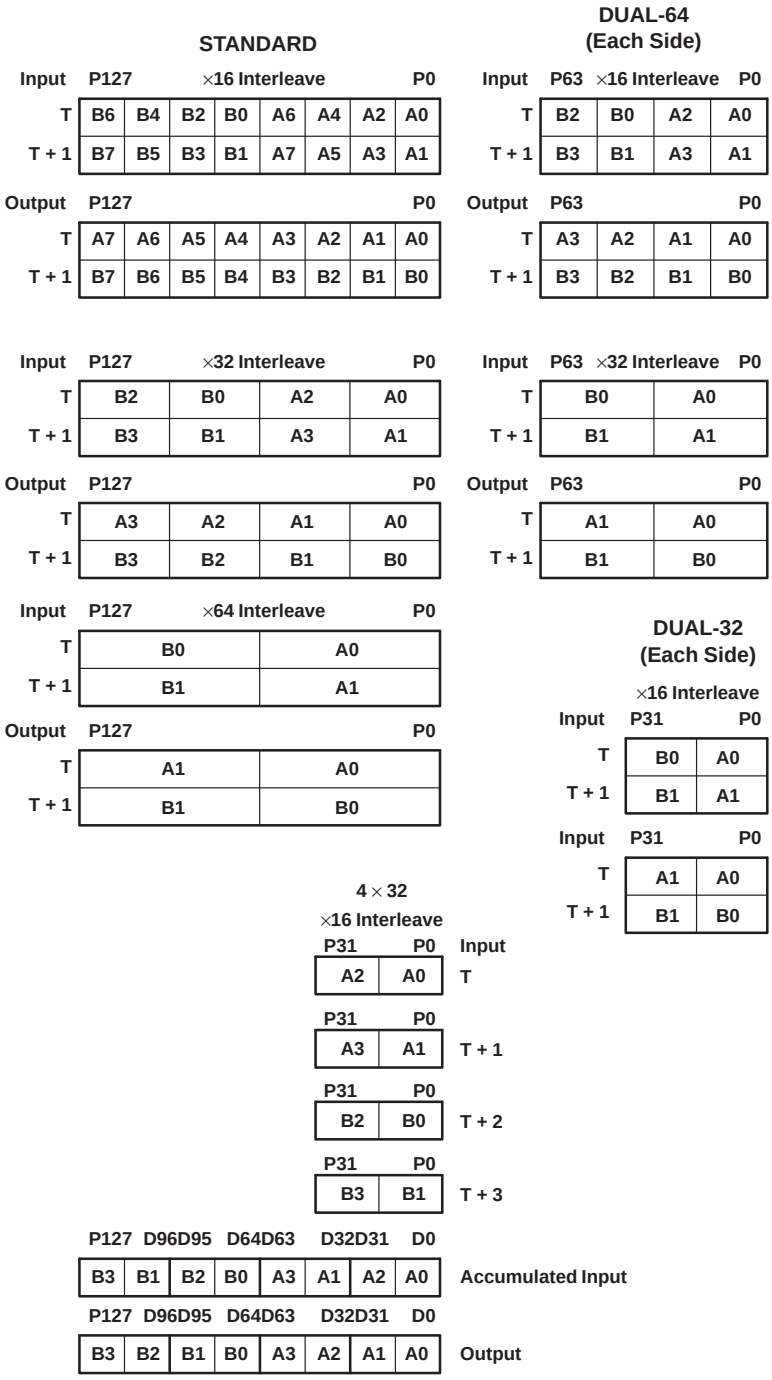


Figure 2–7. De-Interleave Modes

2.7.1 Standard Configuration

For a standard configuration, Figure 2–7 shows the de-interleave scheme for a 128-bit bus width. The A and B represent the first and second linear groups of 128 bits to be displayed for a single pixel stream. For the ×16 interleave case, A and B are each broken down into eight 16-bit groups (A0, A1, A2,...,A7 and B0, B1, B2,...,B7). For the ×32 interleave case, A and B are each broken down into four 32-bit groups (A0, A1, A2, A3 and B0, B1, B2, B3). For the ×64 interleave case, A and B are each broken down into two 64-bit groups (A0, A1 and B0, B1). After rearranging data groups, two 128-bit words are output in the proper order to be displayed. The output data is then multiplexed onto the A bus and is controlled by programming the pixel port A control registers.

When the bus width is 64, the same de-interleave schemes shown for the dual-64 configuration are used.

When the bus width is 32, the same de-interleave scheme shown for the dual-32 configuration is used.

2.7.2 Dual-64 Configuration

For a dual-64 configuration, the de-interleave mode is independently programmable for pixel ports A and B. Figure 2–7 shows the de-interleave scheme for pixel port A (P63–P0) and a 64-bit bus width. The same scheme applies to pixel port B (P127–P64). The A and B in the figure represent the first and second linear groups of 64 bits to be displayed in a single pixel stream. For the ×16 interleave case, A and B are each broken down into four 16-bit groups (A0, A1, A2, A3 and B0, B1, B2, B3). For the ×32 interleave case, A and B are each broken down into two 32-bit groups (A0, A1 and B0, B1). After rearranging data groups, two 64-bit words are output in the proper order to be displayed. For pixel port A (P63–P0), the output data is multiplexed onto the A bus and is controlled by the pixel port A control registers. For pixel port B (P127–P64), the output data is multiplexed onto the B bus and is controlled by the pixel port B control registers.

When the bus width is 32, the same de-interleave scheme shown for dual-32 configuration is used.

2.7.3 Dual-32 Configuration

For a dual-32 configuration, the de-interleave mode is independently programmable for pixel ports A and B. Figure 2–7 shows the de-interleave scheme for pixel port A (P31–P0). The only allowable bus width is 32 bits. The same scheme applies to pixel port B (P63–P32). The A and B in the figure represent the first and second linear groups of 32 bits to be displayed in a single pixel stream. The only allowable de-interleave mode is ×16. The A and B are each broken down into two 16-bit groups (A0, A1 and B0, B1). After rearranging data groups, two 32-bit words are output in the proper order to be displayed. For pixel port A (P31–P0), the output data is multiplexed onto the A bus and is controlled by the pixel port A control registers. For pixel port B (P63–P32), the output data is multiplexed onto the B bus and is controlled by the pixel port B control registers.

2.7.4 4 × 32 Configuration

For a 4 × 32 configuration, the only allowable de-interleave mode is ×16 and pixel port A and pixel port B are each programmed for a 64-bit bus width. The pixel bus data is received on P31–P0 and is accumulated into a 128-bit word prior to the de-interleave function. The A in Figure 2–7 represents the first linear group of 64 bits to be passed to the A bus (pixel port A) and the B represents the first linear group of 64 bits to be passed to the B bus (pixel port B).

The A and B are each broken down into four 16-bit groups (A0, A1, A2, A3 and B0, B1, B2, B3). After rearranging data groups, a single 128-bit word is output. The lower 64 bits of the output data is multiplexed onto the A bus and is controlled by the pixel port A control registers. The upper 64 bits of the output data is multiplexed onto the B bus and is controlled by the pixel port B control registers.

2.8 Pixel Bus Clocking

Figure 2–8 shows the pixel bus clocking scheme. The SYNC PLLs generate the RCLKA and RCLKB signals to the controllers. The RCLK signals need not be the same frequency as the corresponding LCLK signals. The LCLK signals from the controllers are used to latch the corresponding pixel bus data. The LCLKA signal is used to latch the video controls. The SYNC PLLs are used to align their received LCLK signals with the internal dot clock allowing the received data to be transferred to the internal dot clock reliably.

The phase relationship between LCLKA and LCLKB is critical in order to correctly align the two pixel streams. This alignment is achieved by internal circuitry linking the SYNC A PLL and SYNC B PLL. The synchronization scheme mutually aligns the LCLKA and LCLKB signals with the internal dot clock on the fourth rising edge of the LCLKA signal after $\overline{\text{BLANK}}$ goes inactive. This properly aligns the two pixel streams through the three internal LCLK latching stages. After the three LCLK stages, the two pixel streams are transferred to the internal dot clock pipeline. When pixel port A and pixel port B are both used, the pixel port B data must be started the required number of LCLKB cycles before or after the start of pixel port A data. This is achieved by properly programming the window generator or by external means.

Figure 2–8 shows the typical pixel bus timing. The first LCLKx rising edge that samples blank inactive also latches the first pixel group. The last LCLKx rising edge that samples blank inactive also latches the last pixel group. In Figure 2–8, the delay from RCLKA to $\overline{\text{BLANKA}}$ and P(127–0) depends on the total system delay through the controller. The delay may be as long as is required, it need not be less than the RCLKx cycle time.

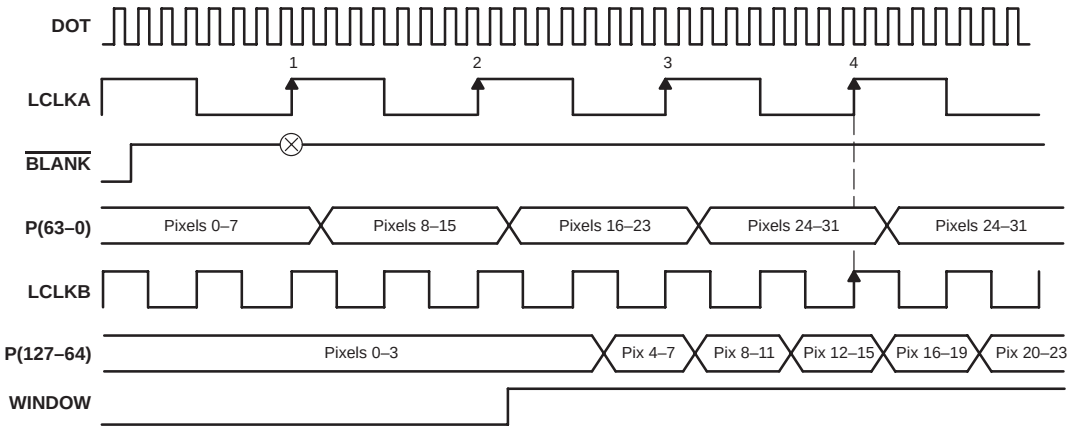


Figure 2–8. Example Pixel Bus Interface Timing

2.9 Pixel Multiplexing Control

Six control registers are used to program the pixel multiplexing functions of the device. The SYS CNFG and ITLV CTL provide control of global functions. SYS CNFG controls the pixel bus configuration, VGA mode, big-/little-endian mode and gamma correction. ITLV CTL selects the interleave mode (or no interleave) for pixel port A and pixel port B. The interleave modes allow pixel data in two consecutive words to be combined as is sometimes necessary when using WRAMs. Pixel port A and pixel port B each use two registers to program the multiplexing mode.

Table 2–22. System Configuration Register (SYS CNFG)
Index: 0x10 Access: R/W Default: 0x80

BIT7	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0
VGAMODE	RAMINPUT	OLAYSEL	RSVD	ENDIAN	GAMMA	BUS CNFG	

Table 2–23. System Configuration Register (SYS CNFG)
Index: 0x10 Access: R/W Default: 0x80

BIT NAME	VALUE	DESCRIPTION
VGAMODE	0: VGA mode disabled	VGA mode enable
	1: VGA mode enabled. When enabled, overrides modes programmed for pixel ports A and B (default).	
RAMINPUT	0: Forces palette RAM input to be overlay or pseudo-color data (default). 1: Forces palette RAM input to be RGB data for gamma correction. The RGB data is from the Y-Bus or the Z-Bus as selected by the GAMMA bit.	Color palette RAM input data selector
OLAYSEL	0: Overlay or pseudo-color data from pixel port A is selected (default) 1: Overlay or pseudo-color data from pixel port B is selected	Overlay port selector
Reserved		
ENDIAN	0: Little-endian mode (default)	Pixel endian control
	1: Big-endian mode	
GAMMA	0: Gamma correction applied to Y bus data if palette RAM is available (default)	Gamma correction source selector
	1: Gamma correction applied to Z bus data if palette RAM is available	
BUS CNFG	00: Standard (default)	Pixel bus configuration
	01: Dual-64	
	10: Dual-32	
	11: 4 $\times$ 32	

Table 2–24. Interleave Control Register (ITLV CTL)

BIT7	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0
RESERVED				ITLV B		ITLV A	

Table 2–25. Interleave Control Register (ITLV CTL)
Index: 0x11 Access: R/W Default: 0x00

BIT NAME	VALUE	DESCRIPTION
RESERVED		
ITLV B	00: No interleave (default)	Pixel bus interleave control for pixel port B
	01: 2:1 interleave in groups of 16 bits	
	10: 2:1 interleave in groups of 32 bits	
	11: 2:1 interleave in groups of 64 bits	
ITLV A	00: No interleave (default)	Pixel bus interleave control for pixel port A
	01: 2:1 interleave in groups of 16 bits	
	10: 2:1 interleave in groups of 32 bits	
	11: 2:1 interleave in groups of 64 bits	

Table 2–26. Pixel Port A Control Register 1 (PPA CTL1)

BIT7	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0
RSVD	HZOOM			COL DEPTH		BUSWIDTH	

Table 2–27. Pixel Port A Control Register 1 (PPA CTL1)
Index: 0x18 Access: R/W Default: 0x02

BIT NAME	VALUE	DESCRIPTION
RESERVED		
HZOOM	000: ×1 (default–no zoom)	Horizontal zoom factor for pixel port A
	001: ×2	
	010: ×4	
	011: ×8	
	100: ×16	
	101: ×32	
	110 - 111: Reserved	
COL DEPTH	00: 8 bits per pixel (default)	Total bits per pixel for pixel port A
	01: 16 bits per pixel	
	10: 24 bits per pixel	
	11: 32 bits per pixel	
BUSWIDTH	00: 32	Bus width for pixel port A
	01: 64	
	10: 128 (default)	
	11: Reserved	

Table 2–28. Pixel Port A Control Register 2 (PPA CTL2)

BIT7	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0
MODE			RESERVED		DBUF SEL	SUB MODE	

Table 2–29. Pixel Port A Control Register 2 (PPA CTL2)
Index: 0x19 Access: R/W Default: 0x00

BIT NAME	VALUE	DESCRIPTION
MODE	000: Pseudo-color (default)	Multiplexing mode selector for pixel port A
	001: Overlay + RGB	
	010: Reserved	
	011: RGB	
	100: Packed RGB	
	101–111: Reserved	
RESERVED		
DBUF SEL	0: Select lower nibbles when in 4–4–4 DB mode (default)	Double buffer selector for pixel port A
	1: Select upper nibbles when in 4–4–4 DB mode	
SUB MODE		See Table 2–34

Table 2–30. Pixel Port B Control Register 1 (PPB CTL1)

BIT7	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0
RSVD	HZOOM			COL DEPTH		BUSWIDTH	

Table 2–31. Pixel Port B Control Register 1 (PPB CTL1)
Index: 0x48 Access: R/W Default: 0x02

BIT NAME	VALUE	DESCRIPTION
RESERVED		
HZOOM	000: ×1 (default–no zoom)	Horizontal zoom factor for pixel port B
	001: ×2	
	010: ×4	
	011: ×8	
	100: ×16	
	101: ×32	
	110 – 111: Reserved	
COL DEPTH	00: 8 bits per pixel (default)	Total bits per pixel for pixel port B
	01: 16 bits per pixel	
	10: 24 bits per pixel	
	11: 32 bits per pixel	
BUSWIDTH	00: 32	Bus width for pixel port B
	01: 64	
	10: 128 (default)	
	11: Reserved	

Table 2–32. Pixel Port B Control Register 2 (PPB CTL2)

BIT7	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0
MODE			RESERVED		DBUF SEL	SUB MODE	

Table 2–33. Pixel Port B Control Register 2 (PPB CTL2)
Index: 0x49 Access: R/W Default: 0x00

BIT NAME	VALUE	DESCRIPTION
MODE	000: Pseudo-color (default)	Multiplexing mode selector for pixel port B
	001: Overlay + RGB	
	010: Reserved	
	011: RGB	
	100: Packed RGB	
	101–111: Reserved	
RESERVED		
DBUF SEL	0: Select lower nibbles when in 4–4–4 DB mode (default)	Double buffer selector for pixel port B
	1: Select upper nibbles when in 4–4–4 DB mode	
SUB MODE		See Table 2–34

Table 2–34. Definition of MODE and SUB MODE Fields

MODE	SUB MODE			
	00	01	10	11
000 Pseudo-color	8-bit Pseudo-color			
001 Overlay + RGB	4–4–4–4 O–R–G–B	1–5–5–5 O–R–G–B	8–8–8–8 O–R–G–B	8–4–4–4 DB See DBUF SEL bit, Table 2–29 and Table 2–39
011 RGB		5–5–5 DB (Bit 31 selects buffer)		5–6–5 R–G–B
100 Packed RGB		8–8–8 R–G–B		4–4–4 DB See DBUF SEL bit, Table 2–29 and Table 2–33

Table 2–35 indicates how the X, Y, and Z buses are utilized when using the standard configuration for each of the pixel port modes. Any two of the X, Y, and Z pixel streams may be combined on screen on a pixel basis by making use of the color key functions. Where more than one type of pixel data is listed in the X-bus column, the OLAYSEL, RAMINPUT, and GAMMA controls must be programmed via the MPU port to select the desired pixel data type.

Table 2–35. X Bus, Y Bus, and Z Bus Utilization for Standard Configuration

PIXEL PORT A	OLAYSEL, RAMINPUT AND GAMMA			X BUS	Y BUS	Z BUS
Pseudo-color	0	0	X	Pseudo-color		
Overlay + RGB	0 0	0 1	X 0	Overlay GAMMA Y	RGB RGB	
RGB, 5–6–5	X	1	0	GAMMA Y	RGB	
RGB, 5–5–5 DB	X X	1 1	0 1	GAMMA Y GAMMA Z	RGB-HI RGB-HI	RGB-LO RGB-LO
Packed RGB	X	1	0	GAMMA Y	Packed RGB	

Table 2–36 indicates how the X, Y, and Z buses are utilized for each of the possible mode combinations on pixel port A and pixel port B. Any two of the X, Y, and Z pixel streams may be combined on screen on a pixel basis by making use of the color key functions. Where more than one type of pixel data is listed in the X-bus column, the OLAYSEL, RAMINPUT, and GAMMA controls must be programmed via the MPU port to select the desired pixel data type. When necessary A and B are used to indicate that the pixel stream originated from pixel port A or pixel port B respectively.

Table 2–36. X Bus, Y Bus, and Z Bus Utilization for Dual-64, Dual-32, and 4 × 32 Configurations

PIXEL PORT A	PIXEL PORT B	OLAYSEL, RAMINPUT AND GAMMA			X BUS	Y BUS	Z BUS
Pseudo-color	Pseudo-color	0 1	0 0	X X	Pseudo-color A Pseudo-color B		
	Overlay + RGB	0 1 X	0 0 1	X X 0	Pseudo-color OVERLAY GAMMA Y	RGB RGB RGB	
	RGB	0 X	0 1	X 0	Pseudo-color GAMMA Y	RGB RGB	
	Packed RGB	0 X	0 1	X 0	Pseudo-color GAMMA Y	Packed RGB Packed RGB	
Overlay + RGB	Pseudo-color	1 0 X	0 0 1	X X 1	Pseudo-color OVERLAY GAMMA Z		RGB RGB RGB
	Overlay + RGB	0 1 X X	0 0 1 1	X X 0 1	OVERLAY-A OVERLAY-B GAMMA Y GAMMA Z	RGB-B RGB-B RGB-B RGB-B	RGB-A RGB-A RGB-A RGB-A
	RGB	0 X X	0 1 1	X 0 1	OVERLAY GAMMA Y GAMMA Z	RGB-B RGB-B RGB-B	RGB-A RGB-A RGB-A
	Packed RGB	0 X X	0 1 1	X 0 1	OVERLAY GAMMA Y GAMMA Z	Packed RGB Packed RGB Packed RGB	RGB RGB RGB
RGB	Pseudo-color	1 X	0 1	X 1	Pseudo-color GAMMA Z		RGB RGB
	Overlay + RGB	1 X X	0 1 1	X 0 1	OVERLAY GAMMA Y GAMMA Z	RGB-B RGB-B RGB-B	RGB-A RGB-A RGB-A
	RGB	X X	1 1	0 1	GAMMA Y GAMMA Z	RGB-B RGB-B	RGB-A RGB-A
	Packed RGB	X X	1 1	0 1	GAMMA Y GAMMA Z	Packed RGB Packed RGB	RGB RGB
Packed RGB	Pseudo-color	1 X	0 1	X 1	Pseudo-color GAMMA Z		Packed RGB Packed RGB
	Overlay + RGB	1 X X	0 1 1	X 0 1	OVERLAY GAMMA Y GAMMA Z	RGB RGB RGB	Packed RGB Packed RGB Packed RGB
	RGB	X X	1 1	0 1	GAMMA Y GAMMA Z	RGB RGB	Packed RGB Packed RGB
	Packed RGB	X X	1 1	0 1	GAMMA Y GAMMA Z	Packed RGB-B Packed RGB-B	Packed RGB-A Packed RGB-A

Table 2–37. Byte Router Control Register 1 (BR CTL1)

BIT7	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0
GRN SEL1		BLU SEL1		RESERVED			

Table 2–38. Byte Router Control Register 1 (BR CTL1)
Index: 0x24 Access: R/W Default: 0x60

BIT NAME	VALUE	DESCRIPTION
GRN SEL1	00: Route RED data to GREEN output	Byte selector for GREEN output of RGB logic 1 block
	01: Route GREEN data to GREEN output (default)	
	10: Route BLUE data to GREEN output	
	11: Reserved	
BLU SEL1	00: Route RED data to BLUE output	Byte selector for BLUE output of RGB logic 1 block
	01: Route GREEN data to BLUE output	
	10: Route BLUE data to BLUE output (default)	
	11: Reserved	
RESERVED		

Table 2–39. Byte Router Control Register 2 (BR CTL2)

BIT7	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0
RED SEL2		GRN SEL2		BLU SEL2		RED SEL1	

Table 2–40. Byte Router Control Register 2 (BR CTL2)
Index: 0x25 Access: R/W Default: 0x18

BIT NAME	VALUE	DESCRIPTION
RED SEL2	00: Route RED data to RED output (default)	Byte selector for RED output of RGB logic 2 block
	01: Route GREEN data to RED output	
	10: Route BLUE data to RED output	
	11: Reserved	
GRN SEL2	00: Route RED data to GREEN output	Byte selector for GREEN output of RGB logic 2 block
	01: Route GREEN data to GREEN output (default)	
	10: Route BLUE data to GREEN output	
	11: Reserved	
BLU SEL2	00: Route RED data to BLUE output	Byte selector for BLUE output of RGB logic 2 block
	01: Route GREEN data to BLUE output	
	10: Route BLUE data to BLUE output (default)	
	11: Reserved	
RED SEL1	00: Route RED data to RED output (default)	Byte selector for RED output of RGB logic 1 block
	01: Route GREEN data to RED output	
	10: Route BLUE data to RED output	
	11: Reserved	

2.10 Interpolation

Interpolation is a means of increasing the apparent display resolution by computing intermediate pixel values.

For example, there is currently a large installed base of game software that operates in 320 $\times$ 200 resolution. Simple interpolation can yield much improvement in display quality. Also, when video and graphics are mixed on-screen, the video often requires pixel duplication. Here interpolation also improves display quality.

A simple horizontal interpolation function is provided as shown in Figure 2–9. The data from the output multiplexer block may be selected directly or the interpolated data may be selected under the control of the interpolator logic function. The interpolator logic function allows selective interpolation of an area on-screen using the color-key signals (K1, K2, and K3) and the PSEL terminal (K4). When interpolation is used, the pixel port(s) using interpolation must be programmed for a 2$\times$ horizontal zoom (PPA CTL1 or PPB CTL1 registers). This results in duplication of each received pixel. The interpolator function then replaces the duplicated pixel with the interpolated value. The interpolated value is determined for each color field by:

Received pixel sequence : $P_0,\ P_1,\ P_2\ , \ldots$

Duplicated pixel sequence : $P_0,\ P_0,\ P_1,\ P_1,\ P_2\ ,\ P_2\ , \ldots$

Interpolated pixel sequence : $P_0,\ \frac{P_0 + P_1}{2},\ \ P_1,\ \frac{P_1 + P_2}{2},\ P_2,\ \frac{P_2 + P_3}{2}, \ldots$

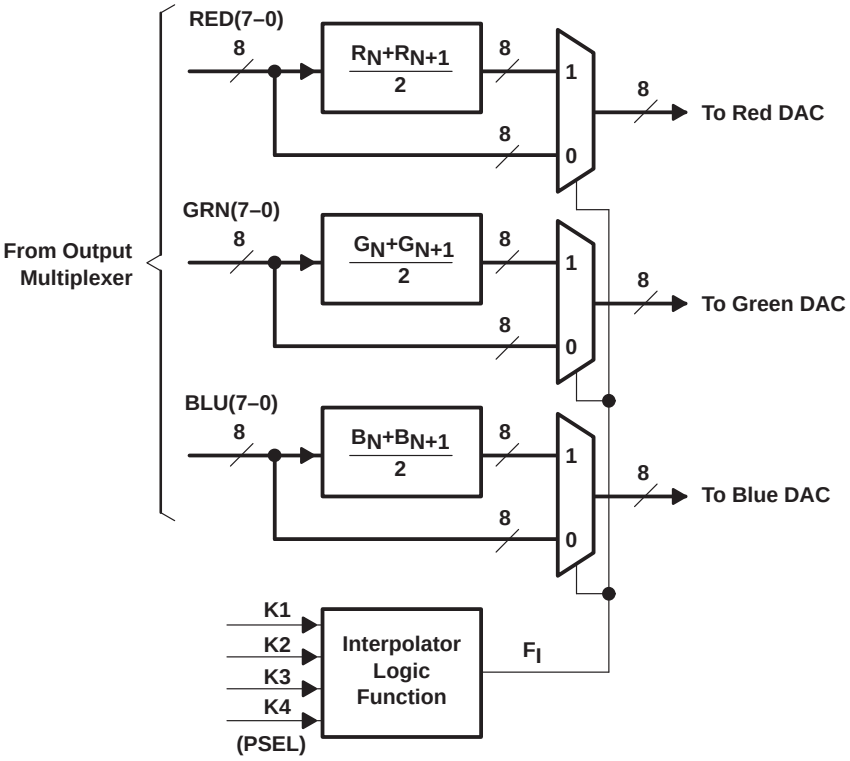


Figure 2–9. Interpolation Function

2.11 Color Key Functions

The color-key switching facility provides three functions (K1, K2, and K3) for mixing video and 2D/3D graphics on screen on a pixel-by-pixel basis. These are shown in Figure 2–10. These functions provide compatibility with the Open MPEG specification (OM/1). The variables named VM, VK, GM, and GK can be associated with either video or graphics data. The K1, K2, and K3 functions and the port select input (denoted as K4 in descriptions below) may be logically combined to control the DAC output, interpolator function, and the analog multiplexer control (AMUXCTL). These logic functions are specified by the DAC FCN1 and DAC FCN2, ITPL FCN1 and ITPL FCN2, and AMUX FCN1 and AMUX FCN2 registers.

The K1 function may be used to key on overlay or pseudo-color data.

The VID MASK register is programmed to zero the unused bits of the overlay field. The VID KEY register is loaded to specify the bit pattern to be matched. The K1 function is described by:

$$K1 = (INPUT\_DATA \times VM) \oplus VK$$

For the RGB 5–5–5 DB mode, the tag bit comes into the color key K1 function as overlay bit 0. The PAIR SEL bits in the KEY CTL2 register selects switching between the Y-bus and Z-bus. The DACMUX logic function is programmed to choose the polarity for buffer selection. Alternately, the PAIR SEL bits may be used for software buffer selection.

The K2 function performs a 24-bit comparison. The input can be from the Y-bus or Z-bus, or from the pseudo-color path and is selected by the K2 MUX bits in the KEY CTL2 register. This function can be used to monitor the graphics pixel stream for a specific color (8-, 16-, or 24-bits/pixel) for displaying video in a

window. The GM RED, GM GRN, and GM BLU registers are programmed to zero the bits to be ignored in the comparison. The GK RED, GK GRN, and GK BLU registers are loaded with the key pattern. The K2 function is described by:

$$K2 = (INPUT\_DATA \times GM) \oplus GK$$

The K3 function performs a 24-bit range comparison. The input can be from the Y-bus or Z-bus and is selected by the K3 MUX bit in the KEY CTL1 register. This function can be used to key on a range of colors within a digitized video stream when a tolerance is required in the key value.

The RED RNGL, GRN RNGL, and BLU RNGL registers are programmed with the 24-bit lower limit for the comparison, and the RED RNGH, GRN RNGH, and BLU RNGH registers are programmed with the 24-bit upper limit. The KEY CTL1 register is programmed to enable/disable the red, green, and blue range comparators. The incoming red, green, and blue color fields are compared with their respective color range registers according to the equation below. The K3 function is described by:

$$K3 = ((REDRNGL \leq INPUT\ (RED) \leq REDRNGH) + \overline{REDMASK})$$
$$\times\ ((GRNRNGL \leq INPUT\ (GRN) \leq GRNRNGH) + \overline{GRNMASK})$$
$$\times\ ((BLURNGL \leq INPUT\ (BLU) \leq BLURNGH) + \overline{BLUMASK})$$

NOTE:

Direct-color data has been shifted to the MSBs of the RGB color fields and the LSBs filled with zeros prior to input to the color-key function.

Bit positions which are masked (set to zero) in the mask registers (VID MASK, GM RED, GM GRN, and GM BLU) should also be set to zero in the corresponding key registers (VID KEY, GK RED, GK GRN, and GK BLU).

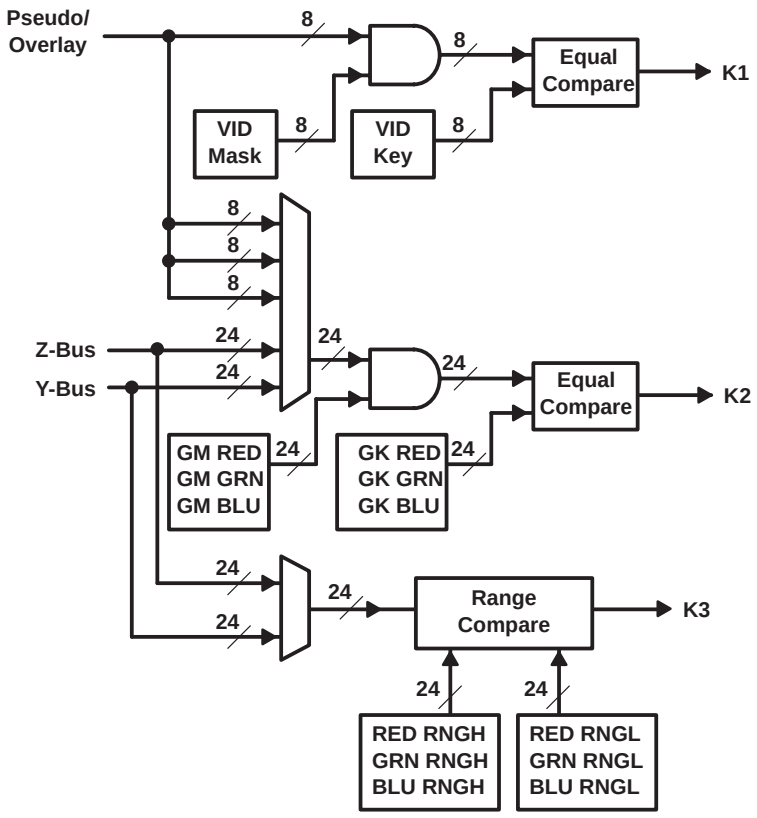


Figure 2–10. Color Key Functions

2.11.1 Color Key Logic Functions

$$P1 = ((K1 \oplus K1I) \times K1M1) + ((K2 \oplus K2I) \times K2M1) + ((K3 \oplus K3I) \times K3M1) + ((K4 \oplus K4I) \times K4M1)$$

$$P2 = ((K1 \oplus K1I) \times K1M2) + ((K2 \oplus K2I) \times K2M2) + ((K3 \oplus K3I) \times K3M2) + ((K4 \oplus K4I) \times K4M2)$$

$$F = ((P1 \oplus P1INV) + (P2 \oplus P2INV)) \oplus INVERT$$

When AND OR = 0

$$F = ((P1 \oplus P1INV) \times (P2 \oplus P2INV)) \oplus INVERT$$

When AND OR = 1

Where:

$\times$ = AND
$+$ = OR
$\oplus$ = Exclusive OR

Table 2–41. DACMUX, Interpolator, and AMUXCTL Logic Function 1 Registers (DAC FCN1, ITP FCN1, AMX FCN1)

BIT7	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0
K1I	K2I	K3I	K4I	P1INV	P2INV	AND OR	INVERT
K1I	K2I	K3I	K4I	P1INV	P2INV	AND OR	INVERT
K1I	K2I	K3I	K4I	P1INV	P2INV	AND OR	INVERT

Table 2–42. DACMUX, Interpolator and AMUXCTL Logic Function 1 Registers (DAC FCN1, ITP FCN1, AMX FCN1) Index: 0x5A, 0x5C, 0x5E Access: R/W Default: 0x00

BIT NAME	VALUE	DESCRIPTION
K1I	0: No inversion (default)	Inversion control for color key function K1
	1: Inversion	
K2I	0: No inversion (default)	Inversion control for color key function K2
	1: Inversion	
K3I	0: No inversion (default)	Inversion control for color key function K3
	1: Inversion	
K4I	0: No inversion (default)	Inversion control for color key function K4 (PSEL input)
	1: Inversion	
P1INV	0: No inversion (default)	Inversion control for P1 partial sum
	1: Inversion	
P2INV	0: No inversion (default)	Inversion control for P2 partial sum
	1: Inversion	
AND OR	0: Combine P1 and P2 with a logical OR	Controls combination of partial sums P1 and P2
	1: Combine P1 and P2 with a logical AND	
INVERT	0: No inversion (default)	Inversion control for overall logic function
	1: Inversion	

Table 2–43. DACMUX, Interpolator, and AMUXCTL Logic Function 2 Registers (DAC FCN2, ITP FCN2, AMX FCN2)

BIT7	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0
K1M1	K2M1	K3M1	K4M1	K1M2	K2M2	K3M2	K4M2
K1M1	K2M1	K3M1	K4M1	K1M2	K2M2	K3M2	K4M2
K1M1	K2M1	K3M1	K4M1	K1M2	K2M2	K3M2	K4M2

Table 2–44. DACMUX, Interpolator, and AMUXCTL Logic Function 2 Registers (DAC FCN2, ITP FCN2, AMX FCN2) Index: 0x5B, 0x5D, 0x5F Access: R/W Default: 0x00

BIT NAME	VALUE	DESCRIPTION
K1M1	0: Bit masked. Associated product term contributes nothing to P1 or P2.	Bit mask for color key function K1 for product term P1
	1: Bit not masked	
K2M1	0: Bit masked	Bit mask for color key function K2 for product term P1
	1: Bit not masked	
K3M1	0: Bit masked	Bit mask for color key function K3 for product term P1
	1: Bit not masked	
K4M1	0: Bit masked	Bit mask for color key function K4 for product term P1
	1: Bit not masked	
K1M2	0: Bit masked	Bit mask for color key function K1 for product term P2
	1: Bit not masked	
K2M2	0: Bit masked	Bit mask for color key function K2 for product term P2
	1: Bit not masked	
K3M2	0: Bit masked	Bit mask for color key function K3 for product term P2
	1: Bit not masked	
K4M2	0: Bit masked	Bit mask for color key function K4 for product term P2
	1: Bit not masked	

Table 2–45. Color Key Control Register 1 (KEY CTL1)

BIT7	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0
RESERVED		AMX DLY		BLUMASK	GRNMASK	REDMASK	RSVD

Table 2–46. Color Key Control Register 1 (KEY CTL1)
Index: 0x38 Access: R/W Default: 0x00

BIT NAME	VALUE	DESCRIPTION
RESERVED		
AMX DLY	00: AMUXCTL synchronous with corresponding pixel boundary on analog RGB outputs (IOR, IOG, IOB) (default)	Pipeline delay select analog multiplexer control (AMUXCTL)
	01: AMUXCTL switches 1 dot clock prior to corresponding pixel boundary on analog RGB outputs	
	10: AMUXCTL switches 2 dot clocks prior to corresponding pixel boundary on analog RGB outputs	
	11: AMUXCTL switches 3 dot clocks prior to corresponding pixel boundary on analog RGB outputs	
BLUMASK	0: Masked (default)	Mask for range compare of blue color field
	1: Not masked	
GRNMASK	0: Masked (default)	Mask for range compare of green color field
	1: Not masked	
REDMASK	0: Masked (default)	Mask for range compare of red color field
	1: Not masked	
RESERVED		

Table 2–47. Color-Key Control Register 2 (KEY CTL2)

BIT7	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0
PAIR SEL			RESERVED	K2 MUX		K3 MUX	K4 CLK

Table 2–48. Color-Key Control Register 2 (KEY CTL2)
Index: 0x39 Access: R/W Default: 0x00

BIT NAME	VALUE	DESCRIPTION
PAIR SEL	000: X bus always selected by DACMUX (default)	Select for pair of pixel streams to switch between
	001: Y bus always selected by DACMUX	
	010: Z bus always selected by DACMUX	
	011: Reserved	
	100: X bus (when F_D = 0) and Y bus (when F_D = 1)	
	101: X bus (when F_D = 0) and Z bus (when F_D = 1)	
	110: Y bus (when F_D = 0) and Z bus (when F_D = 1)	
	111: Reserved	
RESERVED	Always set to 0	
K2 MUX	00: Pseudo-color/overlay data. This data is the output of the read mask/page register block (default).	Select input data for color-key K2 function
	01: Y bus	
	10: Z bus	
	11: Reserved	
K3 MUX	0: Y bus (default)	Select input data for color-key K3 function
	1: Z bus	
K4 CLK	0: PSEL latched by LCLKA (default)	Select latch clock for PSEL terminal
	1: PSEL latched by LCLKB	

2.12 Window Function

The window function provides a timing signal which is active during a programmed rectangular window on the display. This is intended to control the flow of pixel data into pixel port B when in the dual-32 or dual-64 configurations. The controller function connected to pixel port A is always the source of the CRT timing controls. The controller function connected to pixel port B then supplies pixel data in a window within the active display defined by the CRT controls.

In order to allow for a window that borders on the top or left edges of the screen, the window dimensions are specified relative to the trailing edge of HSYNC and the trailing edge of VSYNC. Twelve bit numbers are specified for the X,Y start position and window width and height as shown in Table 2–54 and Table 2–56. The units are scan lines in the vertical dimension and LCLKB cycles in the horizontal dimension.

When pixel port A and pixel port B are both used, the pixel port B data must be started the required number of LCLKBs before or after the start of the pixel port A data. Figure 2–11 shows example window timing. Table 2–49 defines the terms used in this discussion. Table 2–50 describes the CRT timing and PLL programming restrictions when using window.

Table 2–49. Symbol Parameters

SYMBOL	DESCRIPTION
HBP_A	Horizontal back porch (trailing edge of HSYNC to start of active video) time in LCLKAs
HBP_B	Horizontal back porch in LCLKBs
R_A	Multiplex ratio for pixel port A (BUSWIDTH/COLDEPTH)
R_B	Multiplex ratio for pixel port B (BUSWIDTH/COLDEPTH)
N_A	SYNC A PLL N prescaler value in dot clocks
N_B	SYNC B PLL N prescaler value in dot clocks
D	Controller delay in LCLKBs. This is the number of LCLKBs from window going active to the first pixel group latched into pixel port B and is solely dependent on the external pixel port B controller.

Table 2–50. CRT Timing Restrictions for use of WINDOW

NUMBER	RESTRICTION	DESCRIPTION
1	$N = N_A = N_B$	SYNC A PLL and SYNC B PLL N prescalers are the same.
2	$HBP_A = K \times \frac{N}{R_A}$	Horizontal back porch in LCLKAs is an integral multiple of the N prescaler after conversion to LCLKAs.
3	$HTOT_A = K \times \frac{N}{R_A}$	Horizontal total in LCLKAs is an integral multiple of the N prescaler after conversion to LCLKAs.

Table 2–51. CRT Timing Specification for Window Function Example

SCREEN PARAMETERS	
Screen resolution	1280 × 1024
Vertical refresh	75 Hz
Pixel clock frequency	133.33 MHz
Horizontal total time	1152 LCLKA
Horizontal active time	960 LCLKA
Horizontal blank time	192 LCLKA
Horizontal back porch time	96 LCLKA

Table 2–52. Parameter Settings for Window Function Example

PARAMETERS – DUAL-64 CONFIGURATION		
PARAMETER	**PORT A**	**PORT B**
MODE	Packed-RGB	RGB
SUBMODE	8–8–8	5–6–5
BUSWIDTH	32	64
COLDEPTH	24	16
Multiplex ratio	4:3	4:1
SYNC PLL N/M	16/12	16/4
LCLKA frequency	100 MHz	33.33 MHz

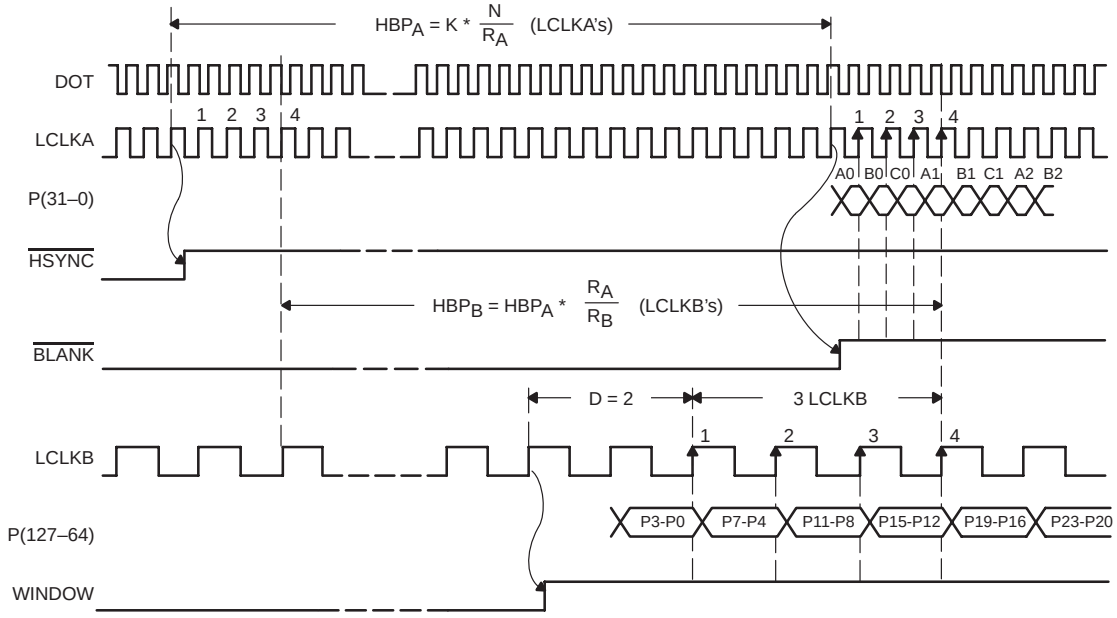


Figure 2–11. Window Timing Example 1

Example 1 mixes graphics data in packed-24 RGB format and 16-bit 5–6–5 XGA format. The timing for example 1 is shown in Figure 2–11. The parameters for this example are shown in Table 2–52. The multiplex ratios R_A and R_B are determined by dividing the BUSWIDTH by the COLDEPTH. The SYNC PLL M-value of 4 is chosen for SYNC B PLL. Since the N/M ratio must be the multiplex ratio, the SYNC B PLL N-value must be 16. To meet restriction 1 in Table 2–50, the SYNC A PLL N-value must also be 16. To obtain the proper 4/3 N/M ratio, the SYNC B PLL M-value must be 12. For restrictions 2 and 3, the horizontal total and back porch times in LCLKAs must be a multiple of [16 / (4/3)] = 12. This is satisfied by the values 1152 and 96 respectively.

$$STARTX = HBP_A \times \frac{R_A}{R_B} - 3 - D + \frac{X}{3 \times R_B}$$

Where $X = 0,\ 3 \times R_B,\ 6 \times R_B,\ 9 \times R_B,\ \ldots$

Equation 1 STARTX for Pixel Port B in Packed-RGB Mode

$$STARTX = HBP_A \times \frac{R_A}{R_B} - 3 - D + \frac{X}{R_B}$$

Where $\times = 0,\ R_B,\ 2 \times R_B,\ 3 \times R_B,\ \ldots$

Equation 2 STARTX for Pixel Port B in all Other Modes

Equation 1 and equation 2 describe how to determine the number to store in the WINSTXL and WINSTXM registers to specify the horizontal starting point for window. In Figure 2–11, because of the internal synchronization mechanism and the observance of retrictions 1–3, the proper window position can be found by, first, converting the horizontal back porch from LCLKA units to LCLKB units. This locates the fourth rising edge of LCLKA into active video, at which point LCLKA, LCLKB, and the internal dot clock are aligned. On this edge, the fourth pixel group of the line should be latched into both pixel port A and pixel port B. Subtract

3 LCLKBs to locate where the first pixel group is latched into pixel port B. Subtract the controller dependent factor D which takes into account the controller latency. Finally, add the required number of LCLKBs required for the X-coordinate on screen where the window begins as shown in equation 1 or equation 2. For this example, the STARTX parameter is calculated to be:

$$\text{STARTX} = 96 \times \frac{\left[\frac{4}{3}\right]}{4} - 3 - 2 + \frac{X}{4} = 27 + \frac{X}{4}$$

Where:

X = 0, 4, 8, . . .

The window granularity is the number of pixels latched into pixel port B per LCLKB (or per group of 3 LCLKBs for packed-RGB mode). To achieve single pixel granularity in positioning the window, the window is programmed to overlap all pixels in the desired window. The color key functions can then be used in conjunction with the window function to obtain single pixel granularity. In this case, the controller connected to pixel port B must be able to position the first pixel at the proper position on the pixel bus. In example 1, if the window must start at X = 3, then the window is programmed to start at X = 0. The first pixel group latched into pixel port B must have the first pixel on terminals P127 – P112. Terminals P111 – P64 for the first load would be don't cares. Pixel port A could then utilize a reserved color to control the rectangular window with single pixel precision.

Table 2–53. Window Start Registers (WIN STXL, WIN STXM, WIN STYL, WIN STYM)

BIT7	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0
START XL							
RESERVED				START XM			
START YL							
WINENBL	RESERVED			START YM			

Table 2–54. Window Start Registers (WIN STXL, WIN STXM, WIN STYL, WIN STYM)
Index: 0x50, 0x51, 0x54, 0x55 Access: R/W Default: Uninitialized

BIT NAME	VALUE	DESCRIPTION
WINSTXL		Window start X LSB register
START XL	0x00–0xFF: Lower eight bits of the 12-bit window start X count. Specify window start X position as number of LCLKB periods after trailing edge of HSYNC.	
WINSTXM		Window start X MSB register
RESERVED		
START XM	0x00–0x0F: Upper four bits of the 12-bit window start X count.	
WINSTYL		Window start Y LSB register
START YL	0x00–0xFF: Lower eight bits of the 12-bit window start Y count. Specify window start Y position as number of scan lines after trailing edge of VSYNC.	
WINSTYM		Window start Y MSB register
WINENBL	0: Default	
	1: Enables the window function	
RESERVED		
START YM	0x00–0x0F: Upper four bits of the 12-bit window start Y count.	

Table 2–55. Window Width and Height Registers (WIN WIDL, WIN WIDM, WIN HGTL, WIN HGTM)

BIT7	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0
WIDTH L							
RESERVED				WIDTH M			
HEIGHT L							
RESERVED				HEIGHT M			

Table 2–56. Window Width and Height Registers (WIN WIDL, WIN WIDM, WIN HGTL, WIN HGTM)
Index: 0x52, 0x53, 0x56, 0x57 Access: R/W Default: Uninitialized

BIT NAME	VALUE	DESCRIPTION
WIN WIDL		Window width LSB register
WIDTH L	0x00–0xFF: Lower eight bits of the 12-bit window width count. Specify window width as number of LCLKB periods.	
WIN WIDM		Window width MSB register
RESERVED		
WIDTH M	0x00–0x0F: Upper four bits of the 12-bit window width count.	
WIN HGTL		Window height LSB register
HEIGHT L	0x00–0xFF: Lower eight bits of the 12-bit window height count. Specify window height as number of scan lines.	
WIN HGTM		Window height MSB register
RESERVED		
HEIGHT M	0x00–0x0F: Upper four bits of the 12-bit window height count.	

2.13 On-Chip Cursor

TVP3033 has an on-chip three-color 64 × 64 pixel user-definable cursor. The cursor operation defaults to the XGA standard, but X-Windows, 3-color, and advanced modes are also available (see Section 2.13.3, *Three-Color 64 × 64 Cursor*). The cursor operates in both noninterlaced and interlaced applications.

The pattern for the 64 × 64 cursor is provided by the cursor RAM, which may be accessed by the MPU at any time. Cursor positioning is performed using the CUR XL, CUR XH, CUR YL, and CUR YH registers in the direct register map. Positions X and Y are defined as increasing from left to right and from top to bottom respectively, as seen on the display screen.

On-chip cursor control is performed by the CUR ICTL register (index: 0x06) shown in Table 2–58. The CUR DCTL register provides an alternate means of enabling and disabling the cursor and selecting the cursor mode with the direct register map.

Table 2–57. Indirect Cursor Control Register (CUR ICTL)

BIT7	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0
REG SEL	FIELD INV	INTRLACE	V DETECT	CRAM A98		MODE SEL	

Table 2–58. Indirect Cursor Control Register (CUR ICTL)
Index: 0x06 Access: R/W Default: 0x00

BIT NAME	VALUE	DESCRIPTION
REG SEL	0: Use CUR ICTL register in indirect map to enable/disable cursor (default)	Cursor control register selector
	1: Use CUR DCTL register in direct map to enable/disable cursor	
FIELD INV	0: ODD/$\overline{\text{EVEN}}$ terminal indicates odd field when 1 (default)	Inversion control for ODD/$\overline{\text{EVEN}}$ input terminal
	1: ODD/$\overline{\text{EVEN}}$ terminal indicates odd field when 0	
INTRLACE	0: Cursor operates in non-interlaced mode (default)	Interlaced mode enable
	1: Cursor operates in interlaced mode. ODD/$\overline{\text{EVEN}}$ terminal determines field with polarity as specified by the FIELD INV bit.	
V DETECT	0: Vertical blank is detected when 2048 consecutive dot clocks have occurred between rising edges of $\overline{\text{BLANK}}$.	Vertical blank detection method
	1: Vertical blank is detected when 4096 consecutive dot clocks have occurred between rising edges of $\overline{\text{BLANK}}$.	
CRAM A98	Cursor RAM upper address bits. CRAM A98 are bits 9 and 8 of the 10-bit cursor RAM address. These are used with the lower 8 bits of the cursor RAM address supplied by the PRAM WAD and PRAM RAD registers in the direct map. CRAM A98 should be written first if it is to be changed.	Cursor RAM address bits 9,8
MODE SEL	00: Cursor disabled (default)	Cursor enable and mode selector. Only effective when REG SEL bit in CUR ICTL register is 0. See Table 2–61 for mode definitions.
	01: Three-color cursor	
	10: XGA cursor	
	11: X-Windows cursor	

Table 2–59. Direct Cursor Control Register (CUR DCTL)

BIT7	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0
RESERVED						MODE SEL	

Table 2–60. Direct Cursor Control Register (CUR DCTL)
Direct Register: 1001 Access: R/W Default: 0x00

BIT NAME	VALUE	DESCRIPTION
RESERVED		
MODE SEL	00: Cursor disabled (default)	Cursor enable and mode selector. Only effective when REG SEL bit in CUR ICTL register is 1. See Table 2–61 for mode definitions.
	01: Three-color cursor	
	10: XGA cursor	
	11: X-Windows cursor	

NOTE 1: The cursor RAM upper address bits (CRAM A98 bits in the CUR ICTL register) default to zeros after reset. Since, normally, software sets these bits to zeroes before accessing the cursor RAM, it may not be necessary to write to the CRAM A98 bits.

2.13.1 Cursor RAM

The 64 × 64 × 2 cursor RAM is used to define the pixel pattern within the 64 × 64 pixel cursor window. It is not initialized and may be written to or read by the MPU at any time, even when the cursor is enabled.

The cursor RAM address zero is at the top left corner of the RAM as shown in Figure 2–12. The cursor plane 0 bits for the entire cursor array are stored in the first 512 bytes of the RAM, and the cursor plane 1 bits for the entire cursor array are stored in the last 512 bytes of the RAM. Information for eight cursor pixels is stored in each byte. The MSB (D7) corresponds with the first or leftmost pixel displayed on the screen.

The 64 × 64 × 2 cursor RAM stores a total of 8192 bits and is accessed through the 8-bit MPU data bus. There are therefore 1024 bytes stored in the RAM and a 10-bit address is used. The upper two bits of the cursor RAM address are written to the CRAM A98 bits of the CUR ICTL register. The lower eight bits of the cursor RAM address are written to the PRAM WAD register (direct register: 0000) for writing to the RAM and to the PRAM RAD register (direct register: 0011) for reading the RAM . Then the plane 0 or 1 data for the first eight pixels is written to the CRAM DAT register (direct register: 1011). This stores the cursor pixel data in the cursor RAM and automatically increments the PRAM WAD register. The upper two bits of the cursor RAM address also increment when the lower eight bits roll over from 0xFF to 0x00. A second write to the CRAM DAT register loads the plane 0 or 1 data for the next eight cursor pixels, and so on. Update of the entire cursor RAM requires 1024 writes to the CRAM DAT register.

To read from the cursor RAM, the address of the first cursor RAM location to be read is loaded using the CRAM A98 bits in the CUR ICTL register and the PRAM RAD register. Then, a read is performed on the CRAM DAT register (direct register: 1011) which reads the plane 0 or 1 data for eight consecutive pixels. Similar to the cursor RAM write operation, when the read is completed, the CRAM A98 registers and the PRAM RAD register are automatically incremented and further reads read successive cursor RAM locations. Upload of the entire cursor RAM requires 1024 reads of the CRAM DAT register.

NOTE:

Internally, the entire 10-bit address is loaded into the address counter after a write to the PRAM WAD or PRAM RAD register (direct register: 0000 or 0011), so the CRAM A98 bits should be written to first, if they are to be changed.

Vertical retrace is determined by detecting 2048 or 4096 pixel clocks between rising edges of the internal BLANK signal. The V DETECT bit in the CUR ICTL register selects 2048 when 0 and 4096 when 1.

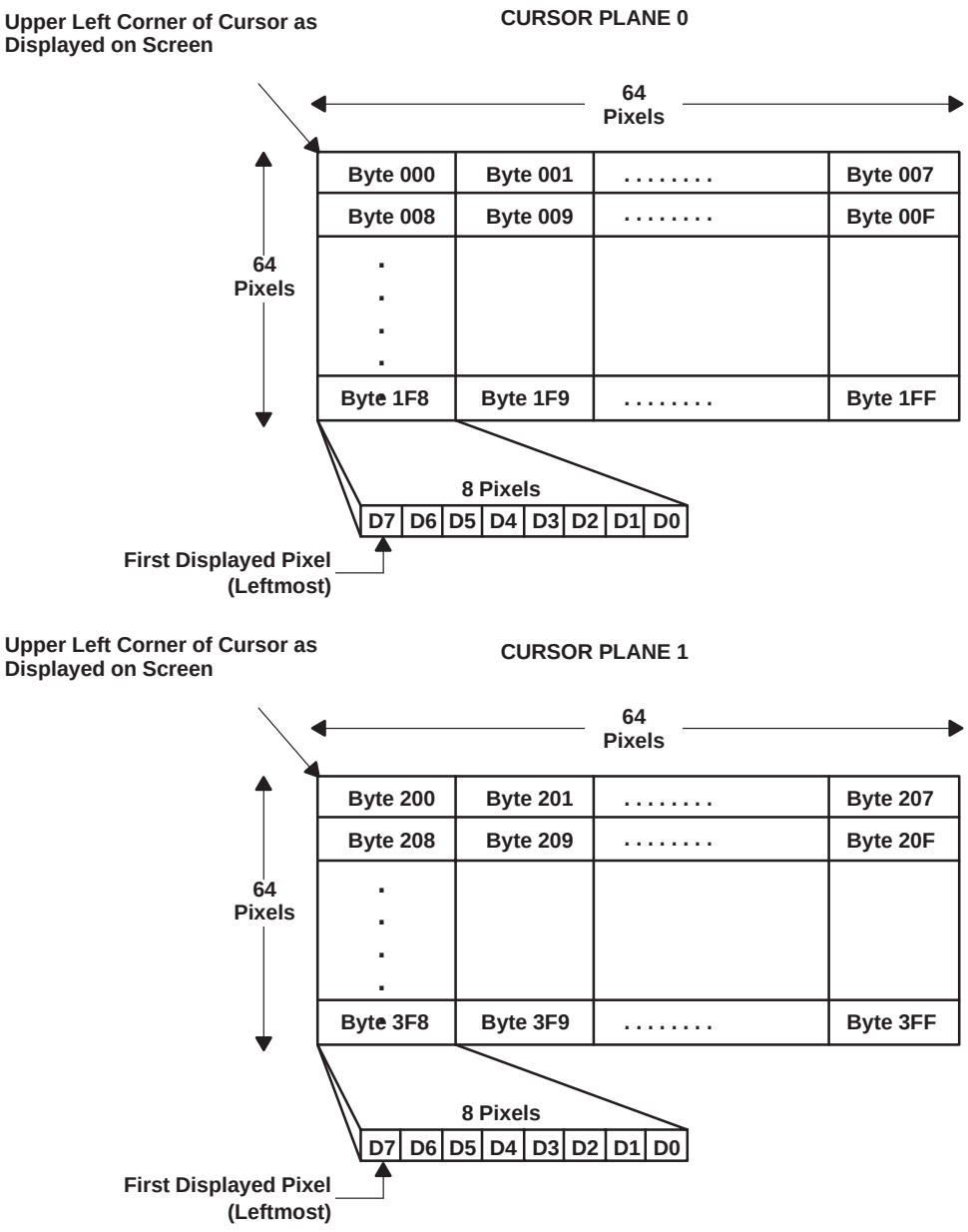


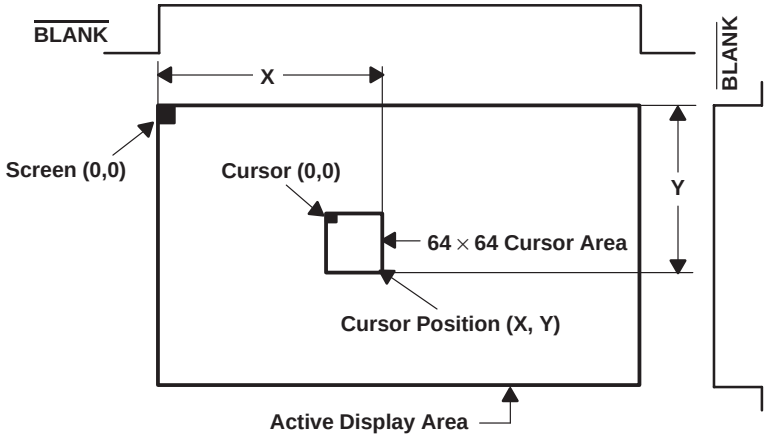
Figure 2–12. Cursor-RAM Organization

2.13.2 Cursor Positioning

The cursor position (x,y) registers are used to position the 64 × 64 cursor on the display screen. The cursor position (x,y) registers specify the location of the cursor bottom right corner on the display screen relative to the end of the internal $\overline{\text{BLANK}}$ signal. Figure 2–13 shows the orientation of the x,y coordinates for positioning the cursor.

The values written to the cursor position registers represent the position of the bottom right corner of the cursor. If zero is written to the cursor position x or cursor position y registers the cursor is off screen. If the cursor position (x,y) is (1,1), only a single pixel of the cursor [cursor (63,63)] is displayed and it appears at the upper left corner of the screen [screen (0, 0)].

If the upper left corner of the cursor is preferred as a reference, determine the screen (x,y) coordinate where cursor (0, 0) is to be positioned. Then add 64 (0x40) to the x coordinate and add 64 (0x40) to the y coordinate and write these values to the cursor position (x, y) registers. For example, if the upper left corner of the cursor is to be positioned at screen (0, 0) write (0x40, 0x40) to the cursor position (x, y) registers.



Cursor Position (X,Y) = Screen (X,Y) Where Cursor (0,0) is Located + (64,64)

Figure 2–13. Cursor-Positioning

2.13.3 Three-Color 64 × 64 Cursor

The 64 × 64 × 2 cursor RAM provides two bits of cursor information on every dot clock cycle during the 64 × 64 cursor window. CCR(1,0) specify whether the XGA mode (10) or X-Windows mode (11) or 3-color mode (01) is used to interpret the cursor information. When CCR(1,0) = 00, the cursor is disabled. The cursor enable/disable and mode select may also be programmed with the direct cursor control register. The two bits of cursor pixel data determine the cursor appearance as shown in Table 2–61.

Table 2–61. Cursor Color Selection Modes

RAM		COLOR SELECTION		
PLANE 1	PLANE 0	3-COLOR MODE	XGA MODE	X-WINDOWS MODE
0	0	Transparent	Cursor color 0	Transparent
0	1	Cursor color 0	Cursor color 1	Transparent
1	0	Cursor color 1	Transparent	Cursor color 0
1	1	Cursor color 2	Complement	Cursor color 1

NOTES: 2. Cursor color 0, 1, and 2: These colors are set by writing to the cursor color registers.
3. Transparent: The underlying pixel color is displayed.
4. Complement: The ones complement of the underlying pixel color is displayed.

2.13.4 Interlaced Cursor Operation

The cursor supports an interlaced display when the INTRLACE bit in the CUR ICTL register is enabled. For the following discussion, an interlaced display consisting of an even field of scan lines numbered 0, 2, 4,...etc., and an odd field of scan lines numbered 1, 3, 5,...etc is assumed. Scan line 0 is the first scan line at the top of the display. When interlaced mode is enabled and cursor-position y (CPy) is greater than 64 (0x40) and less than or equal to 4095 (0xFFF), the first cursor line displayed depends on the state of the ODD/$\overline{\text{EVEN}}$ terminal and value of CPy.

If CPy is an even number, the data in row 0 of the cursor RAM array is displayed during the even field followed by rows 2, 4, ..., 62 on successive scan lines. The data in row 1 of the cursor RAM array is displayed during the odd field followed by rows 3, 5, ..., 63 on successive scan lines.

If CPy is an odd number, the data in row 0 of the cursor RAM array is displayed during the odd field followed by rows 2, 4, ..., 62 on successive scan lines. The data in row 1 of the cursor RAM array is displayed during the even field followed by rows 3, 5, ..., 63 on successive scan lines.

If CPy is between 0 and 64 (0x40), the cursor is partially off the top of the screen. In this case, the data in the first displayed row of the cursor RAM (row N) is always displayed on scan line 0, which is the first scan line of the even field, followed by cursor rows N + 2, N + 4,...etc., on successive scan lines. The data in cursor row N+1 is displayed on scan line 1, which is the first scan line of the odd field, followed by cursor rows N + 3, N + 5,...etc., on successive scan lines.

The FIELD INV bit of the CUR ICTL register allows the polarity of the received ODD/$\overline{\text{EVEN}}$ signal to be inverted.

2.14 Overscan Border

The TVP3033 provides the capability to produce a custom screen border using the overscan function. The overscan function is enabled by the OVS ENBL bit of the GEN CTL1 register. The overscan color is user-programmable by loading the overscan color red, green, and blue registers as described in Section 2.3, *Cursor Color Registers*.

If the overscan function is enabled (OVS ENBL = 1), the overscan color is displayed any time that OVS is high and $\overline{\text{BLANK}}$ is low (active). The blanking pedestal will be imposed on the analog outputs when both OVS and $\overline{\text{BLANK}}$ are low. If overscan is disabled, then the blanking pedestal occurs whenever $\overline{\text{BLANK}}$ is low.

The OVS terminal is always sampled by LCLKA. Figure 2–14 demonstrates the use of the OVS terminal to produce a custom overscan screen border.

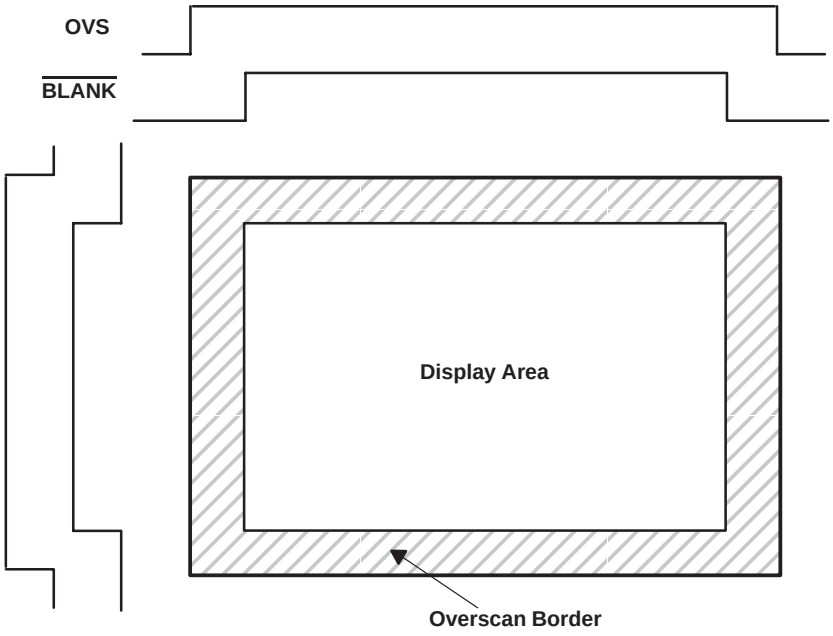


Figure 2–14. Overscan Border

2.15 Video Encoder Interface

The video encoder interface enables simultaneous output of computer graphics to a computer monitor and to a TV or VCR when used with a companion video encoder device. Since the TV signal is encoded from the color palette internal pixel stream, the TV display can make use of the hardware cursor, digital mixing, and interpolation features of the device.

Terminals P(127–96) of the pixel bus assume this alternate function when the video encoder interface is enabled as shown in Table 2–62. The pixel data is driven out with an independent set of video controls (VEHSYNC, VEVSYNC, VEBLANK) and a synchronous clock (VECLK). Operation of this interface is intended for up to 640 × 480 resolution with a maximum pixel clock frequency of 30 MHz.

When the video encoder interface is enabled, P(63–0) are available for pixel data input using the standard, dual-32, or 4 × 32 configurations. The VEI CNTL register shown in Table 2–64 is used to enable this function and to select the output polarity of VEHSYNC and VEVSYNC.

Table 2–62. Terminal Definitions for Video Encoder Interfacing

ALTERNATE TERMINAL DEFINITIONS WHEN VIDEO ENCODER INTERFACE IS ENABLED		
PIXEL BUS TERMINAL	**VIDEO ENCODER OUTPUT SIGNAL**	**COMMENT**
P127	VECLK	Synchronous clock output
P126	$\overline{\text{VEBLANK}}$	Blank output
P125	VEHSYNC	HSYNC output
P124	VEVSYNC	VSYNC output
P123 – P120	RESERVED	
P119 – P112	VERED7 – VERED0	Red pixel data outputs (7 = MSB)
P111– P104	VEGRN7 – VEGRN0	Green pixel data outputs (7 = MSB)
P103 – P96	VEBLU7 – VEBLU0	Blue pixel data outputs (7 = MSB)

Table 2–63. Video Encoder Interface Control Register (VEI CNTL)

BIT7	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0
VEI ENBL	RESERVED					VS INVRT	HS INVRT

Table 2–64. Video Encoder Interface Control Register (VEI CNTL)
Index: 0x1F Access: R/W Default: 0x00

BIT NAME	VALUE	DESCRIPTION
VEI ENBL	0: Disable (default)	Video encoder interface function enable
	1: Enable	
RESERVED		
VS INVRT	0: Output on VEVSYNC has same polarity as VSYNC input terminal (default)	Vertical sync output inversion control
	1: Output on VEVSYNC has opposite polarity as VSYNC input terminal	
HS INVRT	0: Output on VEHSYNC has same polarity as HSYNC input terminal (default)	Horizontal sync output inversion control
	1: Output on VEHSYNC has opposite polarity as HSYNC input terminal	

2.16 Test Functions

The TVP3033 provides several functions that enable system testing and verification. These are detailed in the following sections.

2.16.1 16-Bit CRC

A 16-bit cyclic redundancy check (CRC) is provided so that video data integrity can be verified at the input to the DACs. The CRC is updated when two consecutive HSYNC pulses are detected while $\overline{\text{BLANK}}$ is active (vertical retrace). For use of the CRC function, HSYNC must be active low at the input to the TVP3033. The CRC is only calculated on the active screen area; i.e., active blank stops the calculation. One complete vertical screen must be completed to generate a valid CRC.

The CRC can be performed on any of the 24 data lines that enter the DACs and is controlled by the CRC SEL register. Values from 0 to 23 (0x17) may be written to this register to select between the 24 different DAC data inputs. Value 0 corresponds to DAC data red 0 (LSB), value 7 to red 7 (MSB), value 8 to green 0 (LSB), value 15 to green 7 (MSB), value 16 to blue 0 (LSB), and value 23 to blue 7 (MSB). The 16-bit remainder that is calculated on the individual DAC data line can be read from the CRC LSB and CRC MSB registers.

As long as the display pattern for each screen remains fixed, the CRC result should remain constant. If the CRC result changes, an error condition should be assumed. The CRC is calculated using the algorithm depicted by the circuit in Figure 2–15. The user could calculate and store the CRC remainder for a test screen in software, and compare this to the TVP3033 calculated CRC remainder to verify data integrity.

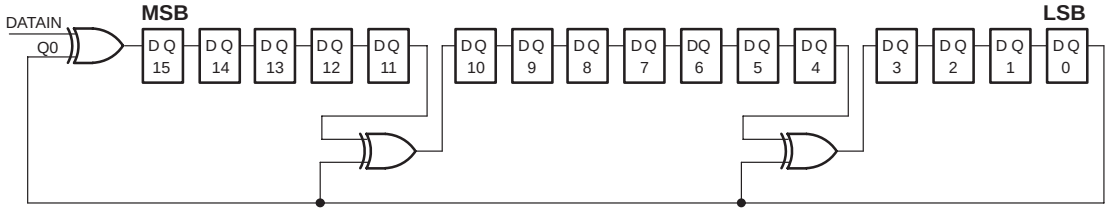


Figure 2–15. CRC Algorithm

2.16.2 Sense Comparator Output and Test Register

The TVP3033 provides a set of analog comparators that can be used to determine the presence of the CRT monitor or verify that the RGB termination is correct. Each analog output is compared with an internal 350-mV reference. The internal reference has a tolerance of ±50 mV when using an external 1.235-V reference. If the internal voltage reference is used, the tolerance will be higher.

The SENS TST register is used to enable the comparator function and to read the comparison results for the red, green, and blue comparators independently. When the sense test register is read, the results are indicated as shown in Table 2–66.

Table 2–65. Sense Test Register (SENS TST)

BIT7	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0
DISABLE	RESERVED				RED CMP	GRN CMP	BLU CMP

Table 2–66. Sense Test Register (SENS TST)
Index: 0x3A Access: R/W Default: 0x00

BIT NAME	VALUE	DESCRIPTION
DISABLE	0: Sense comparator function enabled (default)	Disable control for sense comparator function
	1: Sense comparator function disabled	
Reserved	0000: Always program to 0000	
RED CMP	0: IOR is < 350 mV	Red comparator result
	1: IOR is > 350 mV	
GRN CMP	0: IOG is < 350 mV	Green comparator result
	1: IOG is > 350 mV	
BLU CMP	0: IOB is < 350 mV	Blue comparator result
	1: IOB is > 350 mV	

NOTES: 5. The DISABLE bit can be set to 1 to disable the sense comparison function. At reset, the sense comparison is enabled. If the SENS TST register is written to disable the sense comparator function, bits 6–0 need to be set to 0.

6. The SENS TST register is latched by the falling edge of the internally sampled $\overline{\text{BLANK}}$ signal. In order to have stable voltage inputs to the comparators, the frame buffer inputs should be set such that data entering the DACs remains unchanged for a sufficient period of time prior to and after the $\overline{\text{BLANK}}$ signal falling edge.

2.16.3 Device Identification Code

The DEV ID register (index: 0x3F) allows software identification of the device for different versions of the system design. The DEV ID register is read-only. The value defined for the TVP3033 is 0x33.

2.16.4 Silicon Revision

The silicon revision register (index: 0x01) is a read-only register that enables software to identify the silicon revision of the TVP3033. This number is initially 0x00. A major revision number is stored in bits 7–4 and a minor revision number is stored in bits 3–0.

2.17 Reset

There are two ways to reset the TVP3033. The $\overline{\text{RESET}}$ input terminal can be used to perform a hardware reset. Alternatively, the device has an integrated software reset function. A hardware reset is initiated by pulling the $\overline{\text{RESET}}$ input terminal low. When this is done, all TVP3033 registers go to default states. This reset is asynchronous, and any glitch on this terminal could change the intended register setup. The default state at reset is VGA mode, and all default register settings are listed in Table 2–2. If a reset is desired at power-up, an external resistor capacitor diode network can be connected to the $\overline{\text{RESET}}$ terminal. If TTL logic is employed to provide the signal to the $\overline{\text{RESET}}$ terminal, a pull-up resistor should be used to make sure that CMOS levels are achieved.

For software reset, anytime the SOFT RST register (index: 0xFF) is written to, all registers are initialized to the default settings. The data written into the reset register is ignored.

2.18 Analog Output Specifications

The DAC outputs are controlled by three current sources (only two for IOR and IOB) as shown in Figure 2–16. The default condition is to have 0 IRE difference between blank and black levels, which is shown in Figure 2–18. If a 7.5 IRE pedestal is desired, it can be selected by setting bit GCR4 of the general control register. This video output is shown in Figure 2–17.

A resistor (RSET) is needed between the FS ADJUST terminal and GND to control the magnitude of the full-scale video signal. The IRE relationships in Figure 2–17 and Figure 2–18 are maintained regardless of the full-scale output current.

The relationship between RSET and the full-scale output current IOG is:

RSET (ohms) = K1 X VREF (v) / IOG (mA)

The full-scale output current on IOR and IOB for a given RSET is:

IOR, IOB (mA) = K2 X VREF (v) / RSET (ohms)

where K1 and K2 are defined as:

	IOG		IOR, IOB	
Pedestal	8-Bit Output	6-Bit Output	8-Bit Output	6-Bit Output
7.5 IRE	K1 = 11,294	K1 = 11,206	K2 = 8,067	K2 = 7,979
0 IRE	K1 = 10,684	K1 = 10,600	K2 = 7,462	K2 = 7,374

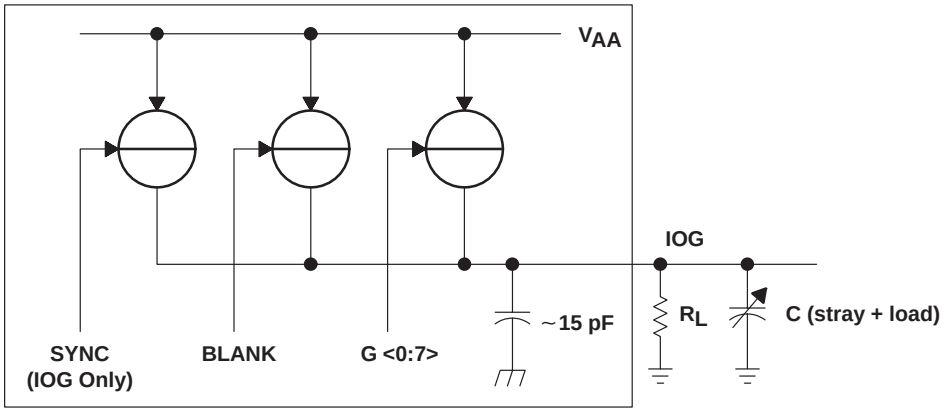


Figure 2–16. Equivalent Circuit of the IOG Current Output

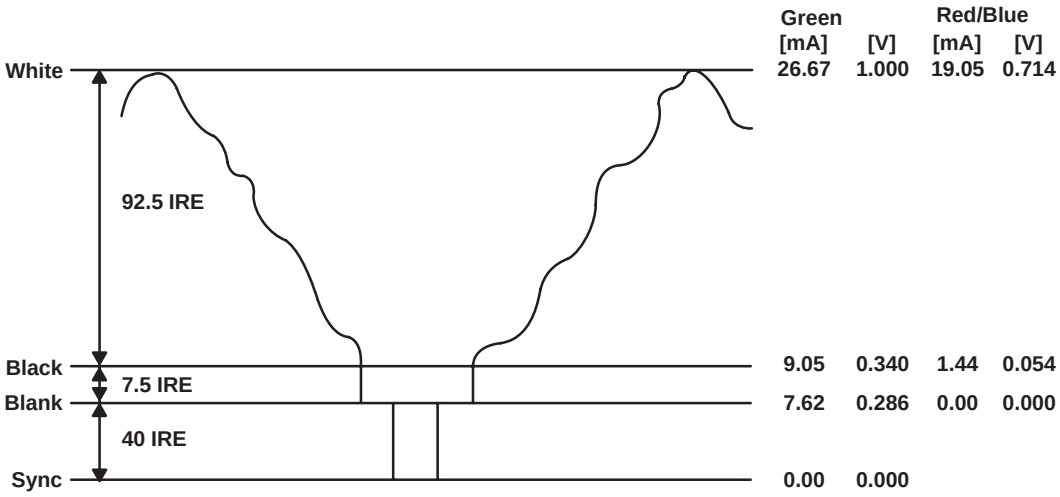


Figure 2–17. Composite Video Output (With 7.5 IRE, 8-Bit Output)

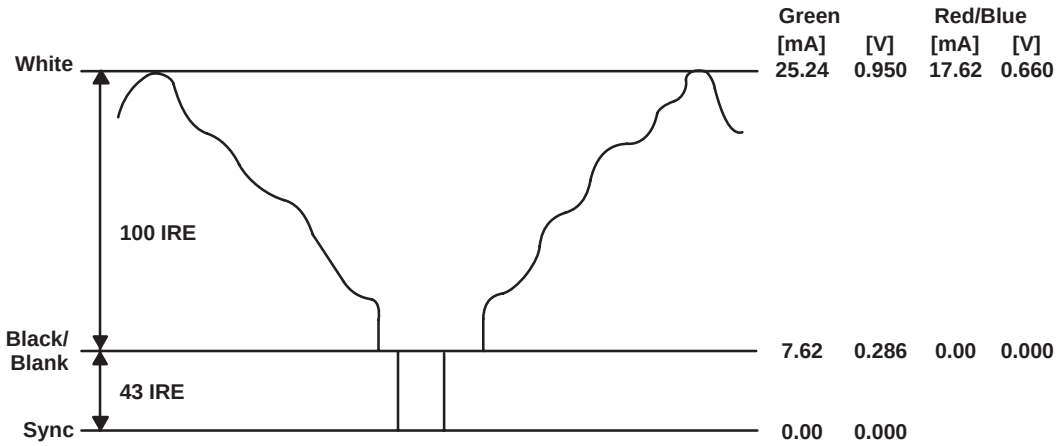


Figure 2–18. Composite Video Output (With 0 IRE, 8-Bit Output)

2.19 Other Register Definitions

Table 2–67. General Control Register 1 (GEN CTL1)

BIT7	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0
RSVD	OVS ENBL	SOG ENBL	PEDESTAL	DAC BITS	RSVD	VS INV	HS INV

Table 2–68. General Control Register 1 (GEN CTL1)
Index: 0x1D Access: R/W Default: 0x00

BIT NAME	VALUE	DESCRIPTION
RESERVED		
OVS ENBL	0: Disabled (default)	Overscan border function enable
	1: Enabled	
SOG ENBL	0: IOG output includes no sync information (default)	Output sync on green analog output (IOG) enable
	1: IOG output includes horizontal and vertical sync information	
PEDESTAL	0: 0 IRE blanking pedestal. Black level and blank level are the same (default).	Blanking pedestal control
	1: 7.5 IRE blanking pedestal. Black level is 7.5 IRE above blank level.	
DAC BITS	0: 6-bit resolution. Six bits each are used to specify the red, green, and blue color fields stored in the color palette RAM (default).	Color palette RAM bits per color
	1: 8-bit resolution. Eight bits each are used to specify the red, green, and blue color fields stored in the color palette RAM.	
RESERVED		
VS INV	0: Do not invert VSYNC before passing to VSYNCOUT output (default)	Inversion control for VSYNCOUT
	1: Invert VSYNC before passing to VSYNCOUT output	
HS INV	0: Do not invert HSYNC before passing to HSYNCOUT output (default)	Inversion control for HSYNCOUT
	1: Invert HSYNC before passing to HSYNCOUT output	

Table 2–69. Power Down Control Register (PWR CNTL)

BIT7	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0
PULL UPS	RESERVED		CRC PWR	KEY PWR	RAM PWR	DAC PWR	CLK PWR

Table 2–70. Power Down Control Register (PWR CNTL)
Index: 0x1E Access: R/W Default: 0x00

BIT NAME	VALUE	DESCRIPTION
PULL UPS	0: Disabled. No pull-up current on pixel bus terminals P127–P0 (default).	Pixel bus pull-up enable
	1: Enabled. Weak 5-uA pull-up current on pixel bus terminals P127–P0.	
RESERVED		
CRC PWR	0: Normal operation (default)	CRC power down control
	1: Power down CRC circuitry	
KEY PWR	0: Normal operation (default)	Color-key power down control
	1: Power down color-key circuitry	
RAM PWR	0: Normal operation (default)	Palette RAM power down control
	1: Power down palette RAM (contents preserved)	
DAC PWR	0: Normal operation (default)	DAC power down control
	1: Power down DACs	
CLK PWR	0: Normal operation (default)	Internal dot clock power down control
	1: Disable internal dot clock	

3 Electrical Characteristics

3.1 Absolute Maximum Ratings Over Operating Free-Air Temperature Range (unless otherwise noted)†

Supply voltage, V_{DD} (see Note 1) . . . . . 7 V
Input voltage range, V_I . . . . . –0.5 V to V_{DD} + 0.5 V
Analog output short-circuit duration to any power supply or common . . . . . unlimited
Operating free-air temperature range, T_A . . . . . 0°C to 70°C
Storage temperature range, T_{stg} . . . . . –65°C to 150°C
Junction temperature, T_J . . . . . 175°C
Case temperature for 10 seconds: PPA package . . . . . 260°C
Lead temperature 1,6 mm (1/16 inch) from case for 10 seconds . . . . . 260°C

† Stresses beyond those listed under “absolute maximum ratings” may cause permanent damage to the device. These are stress ratings only, and functional operation of the device at these or any other conditions beyond those indicated under “recommended operating conditions” is not implied. Exposure to absolute-maximum-rated conditions for extended periods may affect device reliability.

NOTE 1: All voltage values are with respect to GND.

3.2 Recommended Operating Conditions

	MIN	NOM	MAX	UNIT
Supply voltages, AV_{DD}, DV_{DD}	3.1	3.3	3.5	V
Reference voltage, V_{ref}	1.15	1.235	1.26	V
High-level input voltage, V_{IH}	2.4		V_{DD}+0.5	V
Low-level input voltage, V_{IL}			0.8	V
Output load resistance, R_L		37.5		Ω
FS ADJUST resistor, R_{SET}		523		Ω
Operating free-air temperature, T_A	0		70	°C

3.3 Electrical Characteristics

PARAMETER			TEST CONDITIONS	MIN	TYP†	MAX	UNIT
V_{OH}	High-level output voltage		I_{OH} = –800 μA	2.4			V
V_{OL}	Low-level output voltage	D(7–0), RCLKA, RCLKB, PCLK, MCLK, WINDOW, AMUXCTL	I_{OL} = 3.2 mA			0.4	V
		HSYNCOUT, VSYNCOUT	I_{OL} = 15 mA			0.4	
I_{IH}	High-level input current	TTL inputs	V_I = 2 V			1	μA
I_{IL}	Low-level input current	TTL inputs	V_I = 0.8 V			–1	μA
I_{DD}	Supply current	TVP3033-175				800	mA
		TVP3033-220				950	
		TVP3033-250				1100	
I_{OZ}	High-impedance-state output current					10	μA
C_i	Input capacitance	TTL inputs	f = 1 MHz, V_I = 2 V		4		pF

† All typical values are at V_{DD} = 3.3 V, T_A = 25°C.

3.4 Operating Characteristics

PARAMETER		TEST CONDITIONS	MIN	TYP	MAX	UNIT
Resolution (each DAC)		8-bit mode		8		bits
		6-bit mode		6		
E_L	End-point linearity error (each DAC)	8-bit mode			1	LSB
		6-bit mode			1/4	
E_D	Differential linearity error (each DAC)	8-bit mode			1	LSB
		6-bit mode			1/4	
Gray scale error					5%	
Output current (see Note 2)		White level relative to blank	17.69	19.05	20.4	mA
		White level relative to black (7.5 IRE only)	16.74	17.62	18.5	mA
		Black level relative to blank (7.5 IRE only)	0.95	1.44	1.9	mA
		Blank level on IOR, IOB	0	5	50	μA
		Blank level on IOG (with SYNC enabled)	6.29	7.6	8.96	mA
		Sync level on IOG (with SYNC enabled)	0	5	50	μA
		One LSB (8/$\overline{6}$ high)		69.1		μA
		One LSB (8/$\overline{6}$ low)		276.4		μA
DAC-to-DAC matching				2%	5%	
DAC-to-DAC crosstalk				–20		dB
Output compliance			–1		1.2	V
Voltage reference output voltage			1.15	1.235	1.26	V
Output impedance				50		kΩ
Output capacitance		f = 1 MHz, I_{OUT} = 0		13		pF
Sense voltage reference			300	350	400	mV
Clock and data feedthrough				–20		dB
Glitch area (see Note 3)				50		pV–s
Pipeline delay, pixel port				4 LCLK +27 DOT		periods
Pixel clock PLL, MCLK PLL, SYNC A PLL, SYNC B PLL	Lock time			5		ms
	Jitter			±200		ps

NOTES: 2. Test conditions for RS343-A video signals (unless otherwise specified): "Recommended Operating Conditions", using external voltage reference V_{ref} = 1.235 V, R_{SET} = 523 Ω. When using the internal voltage reference, R_{SET} may need to be adjusted in order to meet these limits.

3. Glitch area does not include clock and data feedthrough. The –3-dB test bandwidth is twice the clock rate.

3.5 Timing Requirements (see Note 4)

			TVP3033 -175		TVP3033 -220		TVP3033 -250		UNIT
			MIN	MAX	MIN	MAX	MIN	MAX	
	DOTCLK frequency			175		220		250	MHz
	Pixel clock PLL	Internal frequency		175		220		250	MHz
		PCLK frequency		110		110		110	MHz
	MCLK PLL frequency			100		100		100	MHz
	VCO frequency for pixel clock PLL, MCLK PLL, SYNC A PLL, and SYNC B PLL		110	220	110	220	110	250	MHz
	CLK0 frequency for VGA mode			85		85		85	MHz
t_{cyc}	Clock cycle time	TTL	7.1		7.1		7.1		ns
t_{su1}	Setup time, RS(3–0) valid before $\overline{RD}$ or $\overline{WR}\downarrow$		17		17		17		ns
t_{h1}	Hold time, RS(3–0) valid after $\overline{RD}$ or $\overline{WR}\downarrow$		17		17		17		ns
t_{su2}	Setup time, D(7–0) valid before $\overline{WR}\uparrow$		35		35		35		ns
t_{h2}	Hold time, D(7–0) valid after $\overline{WR}\uparrow$		0		0		0		ns
t_{su3}	Setup time, VGA(7–0) and $\overline{HSYNC}$, $\overline{VSYNC}$, and $\overline{BLANK}$ valid before CLK0$\uparrow$		4		4		4		ns
t_{h3}	Hold time, VGA(7–0) and $\overline{HSYNC}$, $\overline{VSYNC}$, and $\overline{BLANK}$ valid after CLK0$\uparrow$		4		4		4		ns
t_{su4}	Setup time, P(127–0) and PSEL valid before LCLKA$\uparrow$		5		5		5		ns
t_{h4}	Hold time, P(127–0) and PSEL valid after LCLKA$\uparrow$		4		4		4		ns
t_{su5}	Setup time, $\overline{HSYNC}$, $\overline{VSYNC}$, and OVS valid before LCLKA$\uparrow$		5		5		5		ns
t_{h5}	Hold time, $\overline{HSYNC}$, $\overline{VSYNC}$, and OVS valid after LCLKA$\uparrow$		1		1		1		ns
t_{su6}	Setup time, $\overline{BLANK}$ valid before LCLKA$\uparrow$		3		3		3		ns
t_{h6}	Hold time, $\overline{BLANK}$ valid after LCLKA$\uparrow$		4		4		4		ns
t_{w1}	Pulse duration, $\overline{RD}$ or $\overline{WR}$ low		60		60		60		ns
t_{w2}	Pulse duration, $\overline{RD}$ or $\overline{WR}$ high		40		40		40		ns
t_{w3}	Pulse duration, clock high	TTL	3		2		2		ns
t_{w4}	Pulse duration, clock low	TTL	3		2		2		ns

NOTES: 4. TTL input signals are 0 to 3 V with less than 3 ns rise/fall time between the 10% and 90% levels unless otherwise specified. For input and output signals, timing reference points are at the 10% and 90% signal levels. Analog output loads are less than 10 pF. D7–D0 output loads are less than 50 pF. All other output loads are less than 50 pF unless otherwise specified.

3.6 Switching Characteristics

PARAMETER		TVP3033-175			TVP3033-220			TVP3033-250			UNIT
		MIN	TYP	MAX	MIN	TYP	MAX	MIN	TYP	MAX	
	RCLK frequency (see Note 5)			85			85			85	MHz
t_{en1}	Enable time, $\overline{RD}$ low to D(7–0) valid			40			40			40	ns
t_{dis1}	Disable time, $\overline{RD}$ high to D(7–0) disabled			17			17			17	ns
t_{v1}	Valid time, D(7–0) valid after $\overline{RD}$ high	5			5			5			ns
t_{d1}	Delay time, $\overline{RD}$ low to D(7–0) starting to turn on	5			5			5			ns
t_{d2}	Delay time, CLK0 to DOTCLK (internal signal) high/low		7			7			7		ns
t_{d6}	Analog output settling time (see Note 6)		5			5			4		ns
t_r	Analog output rise time (see Note 7)		2			2			2		ns
	Analog output skew	0		2	0		2	0		2	ns

NOTES: 5. RCLKA and RCLKB can drive an output capacitive load up to 15 pF. The worst-case transition time between the 10% and 90% levels is less than 4 ns (typical 3 ns).
6. Measured from 50% point of full-scale transition to output settling, within ± 1 LSB (settling time does not include clock and data feedthrough).
7. Measured between 10% and 90% of full-scale transition.

3.7 Timing Diagrams

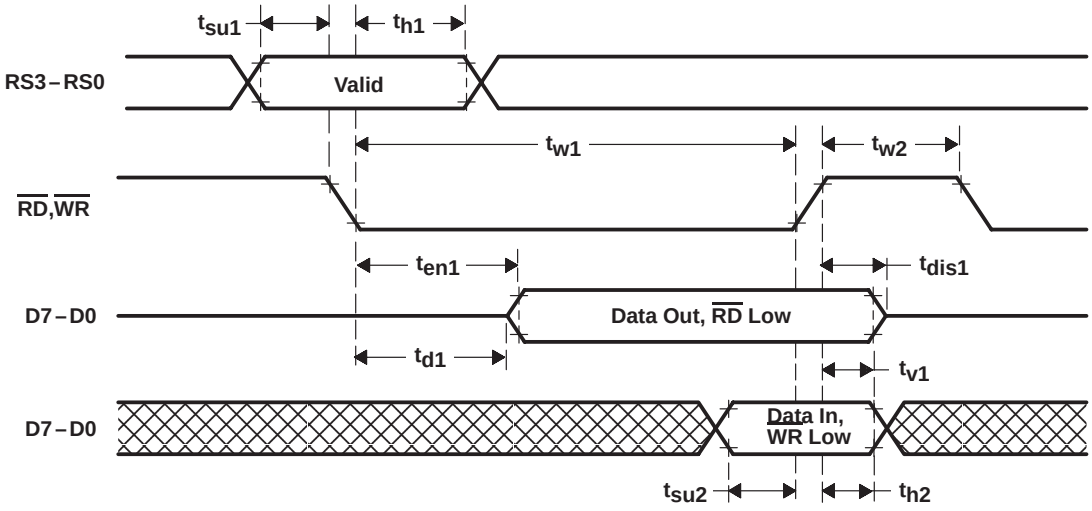


Figure 3–1. MPU Interface Timing

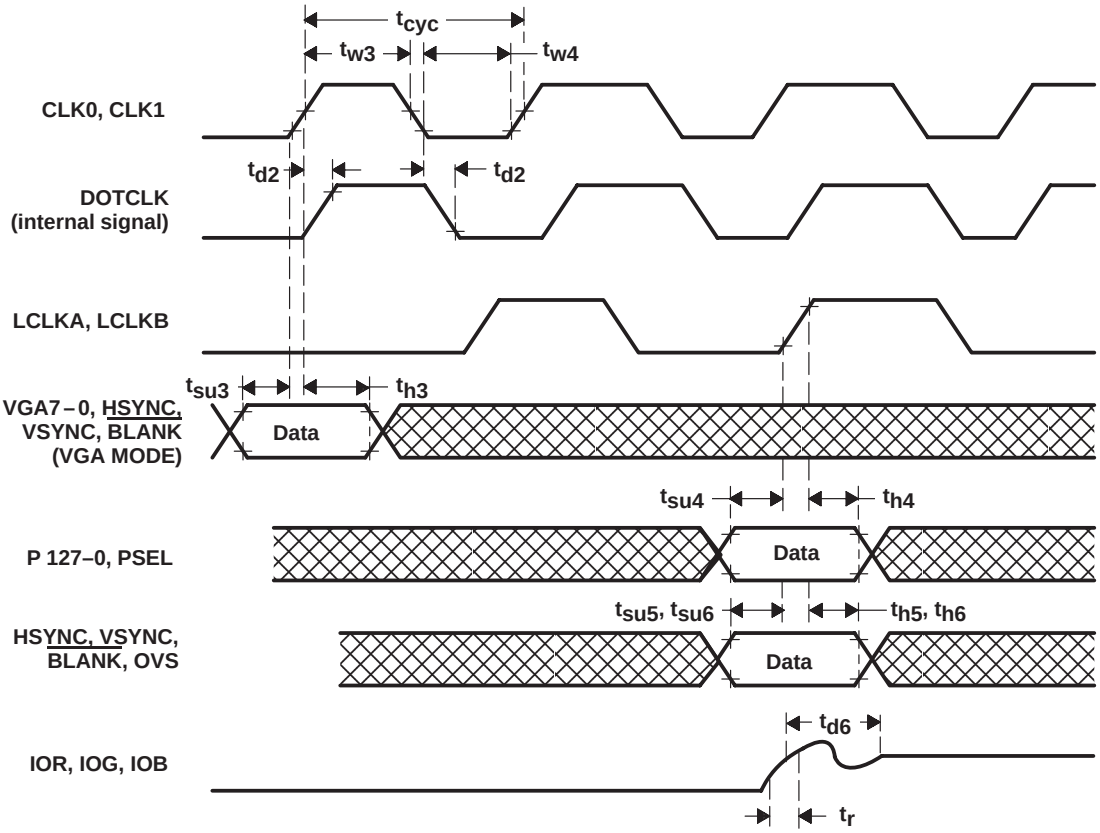


Figure 3–2. Video Input/Output Timing

Appendix A
Pixel Bus Data Formats

A.1 List of Tables

Table A-1. RGB Modes, Non-Packed, Bits 63–0, Little-Endian

MODE	0 = PSEUDO	1 = OVERLAY + RGB				3 = RGB		
SUB MODE	N/A	0 = 4–4–4–4	1 = 1–5–5–5	2 = 8–8–8–8	3 = 8–4–4–4 DB	1 = 5–5–5 DB		3 = 5–6–5
DESCRIPT	INDEXED	O–R–G–B	O–R–G–B	O–R–G–B	DBUF SEL	FD = 0	FD = 1	R–G–B
COL DEPTH	8	16	16	32	32	32		16
BUS WIDTH	128	128	128	128	128	128		128
P63	P7H	O3D	O0D	O7B	O7B	T0B	T0B	R4D
P62	P6H	O2D	R4D	O6B	O6B	R4B		R3D
P61	P5H	O1D	R3D	O5B	O5B	R3B		R2D
P60	P4H	O0D	R2D	O4B	O4B	R2B		R1D
P59	P3H	R3D	R1D	O3B	O3B	R1B		R0D
P58	P2H	R2D	R0D	O2B	O2B	R0B		G5D
P57	P1H	R1D	G4D	O1B	O1B	G4B		G4D
P56	P0H	R0D	G3D	O0B	O0B	G3B		G3D
P55	P7G	G3D	G2D	R7B	R3Bb	G2B		G2D
P54	P6G	G2D	G1D	R6B	R2Bb	G1B		G1D
P53	P5G	G1D	G0D	R5B	R1Bb	G0B		G0D
P52	P4G	G0D	B4D	R4B	R0Bb	B4B		B4D
P51	P3G	B3D	B3D	R3B	R3Bf	B3B		B3D
P50	P2G	B2D	B2D	R2B	R2Bf	B2B		B2D
P49	P1G	B1D	B1D	R1B	R1Bf	B1B		B1D
P48	P0G	B0D	B0D	R0B	R0Bf	B0B		B0D
P47	P7F	O3C	O0C	G7B	G3Bb			R4C
P46	P6F	O2C	R4C	G6B	G2Bb		R4B	R3C
P45	P5F	O1C	R3C	G5B	G1Bb		R3B	R2C
P44	P4F	O0C	R2C	G4B	G0Bb		R2B	R1C
P43	P3F	R3C	R1C	G3B	G3Bf		R1B	R0C
P42	P2F	R2C	R0C	G2B	G2Bf		R0B	G5C
P41	P1F	R1C	G4C	G1B	G1Bf		G4B	G4C
P40	P0F	R0C	G3C	G0B	G0Bf		G3B	G3C
P39	P7E	G3C	G2C	B7B	B3Bb		G2B	G2C
P38	P6E	G2C	G1C	B6B	B2Bb		G1B	G1C
P37	P5E	G1C	G0C	B5B	B1Bb		G0B	G0C
P36	P4E	G0C	B4C	B4B	B0Bb		B4B	B4C
P35	P3E	B3C	B3C	B3B	B3Bf		B3B	B3C
P34	P2E	B2C	B2C	B2B	B2Bf		B2B	B2C
P33	P1E	B1C	B1C	B1B	B1Bf		B1B	B1C
P32	P0E	B0C	B0C	B0B	B0Bf		B0B	B0C

Table A-1. RGB Modes, Non-Packed, Bits 63–0, Little-Endian (Continued)

MODE	0 = PSEUDO	1 = OVERLAY + RGB				3 = RGB		
SUB MODE	N/A	0 = 4–4–4–4	1 = 1–5–5–5	2 = 8–8–8–8	3 = 8–4–4–4 DB	1 = 5–5–5 DB		3 = 5–6–5
DESCRIPT	INDEXED	O–R–G–B	O–R–G–B	O–R–G–B	DBUF SEL	FD = 0	FD = 1	R–G–B
COL DEPTH	8	16	16	32	32	32		16
BUS WIDTH	128	128	128	128	128	128		128
P31	P7D	O3B	O0B	O7A	O7A	T0A	T0A	R4B
P30	P6D	O2B	R4B	O6A	O6A	R4A		R3B
P29	P5D	O1B	R3B	O5A	O5A	R3A		R2B
P28	P4D	O0B	R2B	O4A	O4A	R2A		R1B
P27	P3D	R3B	R1B	O3A	O3A	R1A		R0B
P26	P2D	R2B	R0B	O2A	O2A	R0A		G5B
P25	P1D	R1B	G4B	O1A	O1A	G4A		G4B
P24	P0D	R0B	G3B	O0A	O0A	G3A		G3B
P23	P7C	G3B	G2B	R7A	R3Ab	G2A		G2B
P22	P6C	G2B	G1B	R6A	R2Ab	G1A		G1B
P21	P5C	G1B	G0B	R5A	R1Ab	G0A		G0B
P20	P4C	G0B	B4B	R4A	R0Ab	B4A		B4B
P19	P3C	B3B	B3B	R3A	R3Af	B3A		B3B
P18	P2C	B2B	B2B	R2A	R2Af	B2A		B2B
P17	P1C	B1B	B1B	R1A	R1Af	B1A		B1B
P16	P0C	B0B	B0B	R0A	R0Af	B0A		B0B
P15	P7B	O3A	O0A	G7A	G3Ab			R4A
P14	P6B	O2A	R4A	G6A	G2Ab		R4A	R3A
P13	P5B	O1A	R3A	G5A	G1Ab		R3A	R2A
P12	P4B	O0A	R2A	G4A	G0Ab		R2A	R1A
P11	P3B	R3A	R1A	G3A	G3Af		R1A	R0A
P10	P2B	R2A	R0A	G2A	G2Af		R0A	G5A
P09	P1B	R1A	G4A	G1A	G1Af		G4A	G4A
P08	P0B	R0A	G3A	G0A	G0Af		G3A	G3A
P07	P7A	G3A	G2A	B7A	B3Ab		G2A	G2A
P06	P6A	G2A	G1A	B6A	B2Ab		G1A	G1A
P05	P5A	G1A	G0A	B5A	B1Ab		G0A	G0A
P04	P4A	G0A	B4A	B4A	B0Ab		B4A	B4A
P03	P3A	B3A	B3A	B3A	B3Af		B3A	B3A
P02	P2A	B2A	B2A	B2A	B2Af		B2A	B2A
P01	P1A	B1A	B1A	B1A	B1Af		B1A	B1A
P00	P0A	B0A	B0A	B0A	B0Af		B0A	B0A

Table A-2. RGB Modes, Non-Packed, Bits 127–64, Little-Endian

MODE	0 = PSEUDO	1 = OVERLAY + RGB				3 = RGB		
SUB MODE	N/A	0 = 4–4–4–4	1 = 1–5–5–5	2 = 8–8–8–8	3 = 8–4–4–4 DB	1 = 5–5–5 DB		3 = 5–6–5
DESCRIPT	INDEXED	O–R–G–B	O–R–G–B	O–R–G–B	DBUF SEL	FD = 0	FD = 1	R–G–B
COL DEPTH	8	16	16	32	32	32		16
BUS WIDTH	128	128	128	128	128	128		128
P127	P7P	O3H	O0H	O7D	O7D	T0D	T0D	R4H
P126	P6P	O2H	R4H	O6D	O6D	R4D		R3H
P125	P5P	O1H	R3H	O5D	O5D	R3D		R2H
P124	P4P	O0H	R2H	O4D	O4D	R2D		R1H
P123	P3P	R3H	R1H	O3D	O3D	R1D		R0H
P122	P2P	R2H	R0H	O2D	O2D	R0D		G5H
P121	P1P	R1H	G4H	O1D	O1D	G4D		G4H
P120	P0P	R0H	G3H	O0D	O0D	G3D		G3H
P119	P7O	G3H	G2H	R7D	R3Db	G2D		G2H
P118	P6O	G2H	G1H	R6D	R2Db	G1D		G1H
P117	P5O	G1H	G0H	R5D	R1Db	G0D		G0H
P116	P4O	G0H	B4H	R4D	R0Db	B4D		B4H
P115	P3O	B3H	B3H	R3D	R3Df	B3D		B3H
P114	P2O	B2H	B2H	R2D	R2Df	B2D		B2H
P113	P1O	B1H	B1H	R1D	R1Df	B1D		B1H
P112	P0O	B0H	B0H	R0D	R0Df	B0D		B0H
P111	P7N	O3G	O0G	G7D	G3Db			R4G
P110	P6N	O2G	R4G	G6D	G2Db		R4D	R3G
P109	P5N	O1G	R3G	G5D	G1Db		R3D	R2G
P108	P4N	O0G	R2G	G4D	G0Db		R2D	R1G
P107	P3N	R3G	R1G	G3D	G3Df		R1D	R0G
P106	P2N	R2G	R0G	G2D	G2Df		R0D	G5G
P105	P1N	R1G	G4G	G1D	G1Df		G4D	G4G
P104	P0N	R0G	G3G	G0D	G0Df		G3D	G3G
P103	P7M	G3G	G2G	B7D	B3Db		G2D	G2G
P102	P6M	G2G	G1G	B6D	B2Db		G1D	G1G
P101	P5M	G1G	G0G	B5D	B1Db		G0D	G0G
P100	P4M	G0G	B4G	B4D	B0Db		B4D	B4G
P99	P3M	B3G	B3G	B3D	B3Df		B3D	B3G
P98	P2M	B2G	B2G	B2D	B2Df		B2D	B2G
P97	P1M	B1G	B1G	B1D	B1Df		B1D	B1G
P96	P0M	B0G	B0G	B0D	B0Df		B0D	B0G

Table A-2. RGB Modes, Non-Packed, Bits 127–64, Little-Endian (Continued)

MODE	0 = PSEUDO	1 = OVERLAY + RGB				3 = RGB		
SUB MODE	N/A	0 = 4–4–4–4	1 = 1–5–5–5	2 = 8–8–8–8	3 = 8–4–4–4 DB	1 = 5–5–5 DB		3 = 5–6–5
DESCRIPT	INDEXED	O–R–G–B	O–R–G–B	O–R–G–B	DBUF SEL	FD = 0	FD = 1	R–G–B
COL DEPTH	8	16	16	32	32	32		16
BUS WIDTH	128	128	128	128	128	128		128
P95	P7L	O3F	O0F	O7C	O7C	T0C	T0C	R4F
P94	P6L	O2F	R4F	O6C	O6C	R4C		R3F
P93	P5L	O1F	R3F	O5C	O5C	R3C		R2F
P92	P4L	O0F	R2F	O4C	O4C	R2C		R1F
P91	P3L	R3F	R1F	O3C	O3C	R1C		R0F
P90	P2L	R2F	R0F	O2C	O2C	R0C		G5F
P89	P1L	R1F	G4F	O1C	O1C	G4C		G4F
P88	P0L	R0F	G3F	O0C	O0C	G3C		G3F
P87	P7K	G3F	G2F	R7C	R3Cb	G2C		G2F
P86	P6K	G2F	G1F	R6C	R2Cb	G1C		G1F
P85	P5K	G1F	G0F	R5C	R1Cb	G0C		G0F
P84	P4K	G0F	B4F	R4C	R0Cb	B4C		B4F
P83	P3K	B3F	B3F	R3C	R3Cf	B3C		B3F
P82	P2K	B2F	B2F	R2C	R2Cf	B2C		B2F
P81	P1K	B1F	B1F	R1C	R1Cf	B1C		B1F
P80	P0K	B0F	B0F	R0C	R0Cf	B0C		B0F
P79	P7J	O3E	O0E	G7C	G3Cb			R4E
P78	P6J	O2E	R4E	G6C	G2Cb		R4C	R3E
P77	P5J	O1E	R3E	G5C	G1Cb		R3C	R2E
P76	P4J	O0E	R2E	G4C	G0Cb		R2C	R1E
P75	P3J	R3E	R1E	G3C	G3Cf		R1C	R0E
P74	P2J	R2E	R0E	G2C	G2Cf		R0C	G5E
P73	P1J	R1E	G4E	G1C	G1Cf		G4C	G4E
P72	P0J	R0E	G3E	G0C	G0Cf		G3C	G3E
P71	P7I	G3E	G2E	B7C	B3Cb		G2C	G2E
P70	P6I	G2E	G1E	B6C	B2Cb		G1C	G1E
P69	P5I	G1E	G0E	B5C	B1Cb		G0C	G0E
P68	P4I	G0E	B4E	B4C	B0Cb		B4C	B4E
P67	P3I	B3E	B3E	B3C	B3Cf		B3C	B3E
P66	P2I	B2E	B2E	B2C	B2Cf		B2C	B2E
P65	P1I	B1E	B1E	B1C	B1Cf		B1C	B1E
P64	P0I	B0E	B0E	B0C	B0Cf		B0C	B0E

Table A-3. Packed RGB Mode, Bits 63–0, Little-Endian

MODE	4 = PACKED RGB								
SUB MODE	1 = 8–8–8								
COL DEPTH	24			24			24		
BUS WIDTH	32 (4:3 MUX RATIO)			64 (8:3 MUX RATIO)			128 (16:3 MUX RATIO)		
BUS LOAD	1ST	2ND	3RD	1ST	2ND	3RD	1ST	2ND	3RD
P63				G7C	B7F	R7H	G7C	R7H	B7N
P62				G6C	B6F	R6H	G6C	R6H	B6N
P61				G5C	B5F	R5H	G5C	R5H	B5N
P60				G4C	B4F	R4H	G4C	R4H	B4N
P59				G3C	B3F	R3H	G3C	R3H	B3N
P58				G2C	B2F	R2H	G2C	R2H	B2N
P57				G1C	B1F	R1H	G1C	R1H	B1N
P56				G0C	B0F	R0H	G0C	R0H	B0N
P55				B7C	R7E	G7H	B7C	G7H	R7M
P54				B6C	R6E	G6H	B6C	G6H	R6M
P53				B5C	R5E	G5H	B5C	G5H	R5M
P52				B4C	R4E	G4H	B4C	G4H	R4M
P51				B3C	R3E	G3H	B3C	G3H	R3M
P50				B2C	R2E	G2H	B2C	G2H	R2M
P49				B1C	R1E	G1H	B1C	G1H	R1M
P48				B0C	R0E	G0H	B0C	G0H	R0M
P47				R7B	G7E	B7H	R7B	B7H	G7M
P46				R6B	G6E	B6H	R6B	B6H	G6M
P45				R5B	G5E	B5H	R5B	B5H	G5M
P44				R4B	G4E	B4H	R4B	B4H	G4M
P43				R3B	G3E	B3H	R3B	B3H	G3M
P42				R2B	G2E	B2H	R2B	B2H	G2M
P41				R1B	G1E	B1H	R1B	B1H	G1M
P40				R0B	G0E	B0H	R0B	B0H	G0M
P39				G7B	B7E	R7G	G7B	R7G	B7M
P38				G6B	B6E	R6G	G6B	R6G	B6M
P37				G5B	B5E	R5G	G5B	R5G	B5M
P36				G4B	B4E	R4G	G4B	R4G	B4M
P35				G3B	B3E	R3G	G3B	R3G	B3M
P34				G2B	B2E	R2G	G2B	R2G	B2M
P33				G1B	B1E	R1G	G1B	R1G	B1M
P32				G0B	B0E	R0G	G0B	R0G	B0M

Table A-3. Packed RGB Mode, Bits 63–0, Little-Endian (Continued)

MODE	4 = PACKED RGB								
SUB MODE	1 = 8–8–8								
COL DEPTH	24			24			24		
BUS WIDTH	32 (4:3 MUX RATIO)			64 (8:3 MUX RATIO)			128 (16:3 MUX RATIO)		
BUS LOAD	1ST	2ND	3RD	1ST	2ND	3RD	1ST	2ND	3RD
P31	B7B	G7C	R7D	B7B	R7D	G7G	B7B	G7G	R7L
P30	B6B	G6C	R6D	B6B	R6D	G6G	B6B	G6G	R6L
P29	B5B	G5C	R5D	B5B	R5D	G5G	B5B	G5G	R5L
P28	B4B	G4C	R4D	B4B	R4D	G4G	B4B	G4G	R4L
P27	B3B	G3C	R3D	B3B	R3D	G3G	B3B	G3G	R3L
P26	B2B	G2C	R2D	B2B	R2D	G2G	B2B	G2G	R2L
P25	B1B	G1C	R1D	B1B	R1D	G1G	B1B	G1G	R1L
P24	B0B	G0C	R0D	B0B	R0D	G0G	B0B	G0G	R0L
P23	R7A	B7C	G7D	R7A	G7D	B7G	R7A	B7G	G7L
P22	R6A	B6C	G6D	R6A	G6D	B6G	R6A	B6G	G6L
P21	R5A	B5C	G5D	R5A	G5D	B5G	R5A	B5G	G5L
P20	R4A	B4C	G4D	R4A	G4D	B4G	R4A	B4G	G4L
P19	R3A	B3C	G3D	R3A	G3D	B3G	R3A	B3G	G3L
P18	R2A	B2C	G2D	R2A	G2D	B2G	R2A	B2G	G2L
P17	R1A	B1C	G1D	R1A	G1D	B1G	R1A	B1G	G1L
P16	R0A	B0C	G0D	R0A	G0D	B0G	R0A	B0G	G0L
P15	G7A	R7B	B7D	G7A	B7D	R7F	G7A	R7F	B7L
P14	G6A	R6B	B6D	G6A	B6D	R6F	G6A	R6F	B6L
P13	G5A	R5B	B5D	G5A	B5D	R5F	G5A	R5F	B5L
P12	G4A	R4B	B4D	G4A	B4D	R4F	G4A	R4F	B4L
P11	G3A	R3B	B3D	G3A	B3D	R3F	G3A	R3F	B3L
P10	G2A	R2B	B2D	G2A	B2D	R2F	G2A	R2F	B2L
P09	G1A	R1B	B1D	G1A	B1D	R1F	G1A	R1F	B1L
P08	G0A	R0B	B0D	G0A	B0D	R0F	G0A	R0F	B0L
P07	B7A	G7B	R7C	B7A	R7C	G7F	B7A	G7F	R7K
P06	B6A	G6B	R6C	B6A	R6C	G6F	B6A	G6F	R6K
P05	B5A	G5B	R5C	B5A	R5C	G5F	B5A	G5F	R5K
P04	B4A	G4B	R4C	B4A	R4C	G4F	B4A	G4F	R4K
P03	B3A	G3B	R3C	B3A	R3C	G3F	B3A	G3F	R3K
P02	B2A	G2B	R2C	B2A	R2C	G2F	B2A	G2F	R2K
P01	B1A	G1B	R1C	B1A	R1C	G1F	B1A	G1F	R1K
P00	B0A	G0B	R0C	B0A	R0C	G0F	B0A	G0F	R0K

Table A-4. Packed RGB Mode, Double Buffered, Bits 63–0, Little-Endian

MODE	4 = PACKED RGB								
SUB MODE	3 = 4–4–4 DB								
COL DEPTH	24			24			24		
BUS WIDTH	32 (4:3 MUX RATIO)			64 (8:3 MUX RATIO)			128 (16:3 MUX RATIO)		
BUS LOAD	1ST	2ND	3RD	1ST	2ND	3RD	1ST	2ND	3RD
P63				G3Cb	B3Fb	R3Hb	G3Cb	R3Hb	B3Nb
P62				G2Cb	B2Fb	R2Hb	G2Cb	R2Hb	B2Nb
P61				G1Cb	B1Fb	R1Hb	G1Cb	R1Hb	B1Nb
P60				G0Cb	B0Fb	R0Hb	G0Cb	R0Hb	B0Nb
P59				G3Cf	B3Ff	R3Hf	G3Cf	R3Hf	B3Nf
P58				G2Cf	B2Ff	R2Hf	G2Cf	R2Hf	B2Nf
P57				G1Cf	B1Ff	R1Hf	G1Cf	R1Hf	B1Nf
P56				G0Cf	B0Ff	R0Hf	G0Cf	R0Hf	B0Nf
P55				B3Cb	R3Eb	G3Hb	B3Cb	G3Hb	R3Mb
P54				B2Cb	R2Eb	G2Hb	B2Cb	G2Hb	R2Mb
P53				B1Cb	R1Eb	G1Hb	B1Cb	G1Hb	R1Mb
P52				B0Cb	R0Eb	G0Hb	B0Cb	G0Hb	R0Mb
P51				B3Cf	R3Ef	G3Hf	B3Cf	G3Hf	R3Mf
P50				B2Cf	R2Ef	G2Hf	B2CF	G2Hf	R2Mf
P49				B1Cf	R1Ef	G1Hf	B1Cf	G1Hf	R1Mf
P48				B0Cf	R0Ef	G0Hf	B0Cf	G0Hf	R0Mf
P47				R3Bb	G3Eb	B3Hb	R3Bb	B3Hb	G3Mb
P46				R2Bb	G2Eb	B2Hb	R2Bb	B2Hb	G2Mb
P45				R1Bb	G1Eb	B1Hb	R1Bb	B1Hb	G1Mb
P44				R0Bb	G0Eb	B0Hb	R0Bb	B0Hb	G0Mb
P43				R3Bf	G3Ef	B3Hf	R3Bf	B3Hf	G3Mf
P42				R2Bf	G2Ef	B2Hf	R2Bf	B2Hf	G2Mf
P41				R1Bf	G1Ef	B1Hf	R1Bf	B1Hf	G1Mf
P40				R0Bf	G0Ef	B0Hf	R0Bf	B0Hf	G0Mf
P39				G3Bb	B3EB	R3Gb	G3Bb	R3Gb	B3Mb
P38				G2Bb	B2Eb	R2Gb	G2Bb	R2Gb	B2Mb
P37				G1Bb	B1Eb	R1Gb	G1Bb	R1Gb	B1Mb
P36				G0Bb	B0Eb	R0Gb	G0Bb	R0Gb	B0Mb
P35				G3Bf	B3Ef	R3Gf	G3Bf	R3Gf	B3Mf
P34				G2Bf	B2Ef	R2Gf	G2Bf	R2Gf	B2Mf
P33				G1Bf	B1Ef	R1Gf	G1Bf	R1Gf	B1Mf
P32				G0Bf	B0Ef	R0Gf	G0Bf	R0Gf	B0Mf

Table A-4. Packed RGB Mode, Double Buffered, Bits 63–0, Little-Endian (Continued)

MODE	4 = PACKED RGB								
SUB MODE	3 = 4–4–4 DB								
COL DEPTH	24			24			24		
BUS WIDTH	32 (4:3 MUX RATIO)			64 (8:3 MUX RATIO)			128 (16:3 MUX RATIO)		
BUS LOAD	1ST	2ND	3RD	1ST	2ND	3RD	1ST	2ND	3RD
P31	B3Bb	G3Cb	R3Db	B3Bb	R3Db	G3Gb	B3Bb	G3Gb	R3Lb
P30	B2Bb	G2Cb	R2Db	B2Bb	R2Db	G2Gb	B2Bb	G2Gb	R2Lb
P29	B1Bb	G1Cb	R1Db	B1Bb	R1Db	G1Gb	B1Bb	G1Gb	R1Lb
P28	B0Bb	G0Cb	R0Db	B0Bb	R0Db	G0Gb	B0Bb	G0Gb	R0Lb
P27	B3Bf	G3Cf	R3Df	B3Bf	R3Df	G3Gf	B3Bf	G3Gf	R3Lf
P26	B2Bf	G2Cf	R2Df	B2Bf	R2Df	G2Gf	B2Bf	G2Gf	R2Lf
P25	B1Bf	G1Cf	R1Df	B1Bf	R1Df	G1Gf	B1Bf	G1Gf	R1Lf
P24	B0Bf	G0Cf	R0Df	B0Bf	R0Df	G0Gf	B0Bf	G0Gf	R0Lf
P23	R3Ab	B3Cb	G3Db	R3Ab	G3Db	B3Gb	R3Ab	B3Gb	G3Lb
P22	R2Ab	B2Cb	G2Db	R2Ab	G2Db	B2Gb	R2Ab	B2Gb	G2Lb
P21	R1Ab	B1Cb	G1Db	R1Ab	G1Db	B1Gb	R1Ab	B1Gb	G1Lb
P20	R0Ab	B0Cb	G0Db	R0Ab	G0Db	B0Gb	R0Ab	B0Gb	G0Lb
P19	R3Af	B3Cf	G3Df	R3Af	G3Df	B3Gf	R3Af	B3Gf	G3Lf
P18	R2Af	B2Cf	G2Df	R2Af	G2Df	B2Gf	R2Af	B2Gf	G2Lf
P17	R1Af	B1Cf	G1Df	R1Af	G1Df	B1Gf	R1Af	B1Gf	G1Lf
P16	R0Af	B0Cf	G0Df	R0Af	G0Df	B0Gf	R0Af	B0Gf	G0Lf
P15	G3Ab	R3Bb	B3Db	G3Ab	B3Db	R3Fb	G3Ab	R3Fb	B3Lb
P14	G2Ab	R2Bb	B2Db	G2Ab	B2Db	R2Fb	G2Ab	R2Fb	B2Lb
P13	G1Ab	R1Bb	B1Db	G1Ab	B1Db	R1Fb	G1Ab	R1Fb	B1Lb
P12	G0Ab	R0Bb	B0Db	G0Ab	B0Db	R0Fb	G0Ab	R0Fb	B0Lb
P11	G3Af	R3Bf	B3Df	G3Af	B3Df	R3Ff	G3Af	R3Ff	B3Lf
P10	G2Af	R2Bf	B2Df	G2Af	B2Df	R2Ff	G2Af	R2Ff	B2Lf
P09	G1Af	R1Bf	B1Df	G1Af	B1Df	R1Ff	G1Af	R1Ff	B1Lf
P08	G0Af	R0Bf	B0Df	G0Af	B0Df	R0Ff	G0Af	R0Ff	B0Lf
P07	B3Ab	G3Bb	R3Cb	B3Ab	R3Cb	G3Fb	B3Ab	G3Fb	R3Kb
P06	B2Ab	G2Bb	R2Cb	B2Ab	R2Cb	G2Fb	B2Ab	G2Fb	R2Kb
P05	B1Ab	G1Bb	R1Cb	B1Ab	R1Cb	G1Fb	B1Ab	G1Fb	R1Kb
P04	B0Ab	G0Bb	R0Cb	B0Ab	R0Cb	G0Fb	B0Ab	G0Fb	R0Kb
P03	B3Af	G3Bf	R3Cf	B3Af	R3Cf	G3Ff	B3Af	G3Ff	R3Kf
P02	B2Af	G2Bf	R2Cf	B2Af	R2Cf	G2Ff	B2Af	G2Ff	R2Kf
P01	B1Af	G1Bf	R1Cf	B1Af	R1Cf	G1Ff	B1Af	G1Ff	R1Kf
P00	B0Af	G0Bf	R0Cf	B0Af	R0Cf	G0Ff	B0Af	G0Ff	R0Kf

Table A-5. Packed Modes, Bits 127–64, Little-Endian

MODE	4 = PACKED RGB					
SUB MODE	1 = 8–8–8			3 = 4–4–4 DB		
COL DEPTH	24			24		
BUS WIDTH	128 (16:3 MUX RATIO)			128 (16:3 MUX RATIO)		
BUS LOAD	1ST	2ND	3RD	1ST	2ND	3RD
P127	B7F	G7K	R7P	B3Fb	G3Kb	R3Pb
P126	B6F	G6K	R6P	B2Fb	G2Kb	R2Pb
P125	B5F	G5K	R5P	B1Fb	G1Kb	R1Pb
P124	B4F	G4K	R4P	B0Fb	G0Kb	R0Pb
P123	B3F	G3K	R3P	B3Ff	G3Kf	R3Pf
P122	B2F	G2K	R2P	B2Ff	G2Kf	R2Pf
P121	B1F	G1K	R1P	B1Ff	G1Kf	R1Pf
P120	B0F	G0K	R0P	B0Ff	G0Kf	R0Pf
P119	R7E	B7K	G7P	R3Eb	B3Kb	G3Pb
P118	R6E	B6K	G6P	R2Eb	B2Kb	G2Pb
P117	R5E	B5K	G5P	R1Eb	B1Kb	G1Pb
P116	R4E	B4K	G4P	R0Eb	B0Kb	G0Pb
P115	R3E	B3K	G3P	R3Ef	B3Kf	G3Pf
P114	R2E	B2K	G2P	R2Ef	B2Kf	G2Pf
P113	R1E	B1K	G1P	R1Ef	B1Kf	G1Pf
P112	R0E	B0K	G0P	R0Ef	B0Kf	G0Pf
P111	G7E	R7J	B7P	G3Eb	R3Jb	B3Pb
P110	G6E	R6J	B6P	G2Eb	R2Jb	B2Pb
P109	G5E	R5J	B5P	G1Eb	R1Jb	B1Pb
P108	G4E	R4J	B4P	G0Eb	R0Jb	B0Pb
P107	G3E	R3J	B3P	G3Ef	R3Jf	B3Pf
P106	G2E	R2J	B2P	G2Ef	R2Jf	B2Pf
P105	G1E	R1J	B1P	G1Ef	R1Jf	B1Pf
P104	G0E	R0J	B0P	G0Ef	R0Jf	B0Pf
P103	B7E	G7J	R7O	B3Eb	G3Jb	R3Ob
P102	B6E	G6J	R6O	B2Eb	G2Jb	R2Ob
P101	B5E	G5J	R5O	B1Eb	G1Jb	R1Ob
P100	B4E	G4J	R4O	B0Eb	G0Jb	R0Ob
P99	B3E	G3J	R3O	B3Ef	G3Jf	R3Of
P98	B2E	G2J	R2O	B2Ef	G2Jf	R2Of
P97	B1E	G1J	R1O	B1Ef	G1Jf	R1Of
P96	B0E	G0J	R0O	B0Ef	G0Jf	R0Of

Table A-5. Packed Modes, Bits 127-64, Little-Endian (Continued)

MODE	6 = PACKED RGB					
SUB MODE	1 = 8–8–8			3 = 4–4–4 DB		
COL DEPTH	24			24		
BUS WIDTH	128 (16:3 MUX RATIO)			128 (16:3 MUX RATIO)		
BUS LOAD	1ST	2ND	3RD	1ST	2ND	3RD
P95	R7D	B7J	G7O	R3Db	B3Jb	G3Ob
P94	R6D	B6J	G6O	R2Db	B2Jb	G2Ob
P93	R5D	B5J	G5O	R1Db	B1Jb	G1Ob
P92	R4D	B4J	G4O	R0Db	B0Jb	G0Ob
P91	R3D	B3J	G3O	R3Df	B3Jf	G3Of
P90	R2D	B2J	G2O	R2Df	B2Jf	G2Of
P89	R1D	B1J	G1O	R1Df	B1Jf	G1Of
P88	R0D	B0J	G0O	R0Df	B0Jf	G0Of
P87	G7D	R7I	B7O	G3Db	R3Ib	B3Ob
P86	G6D	R6I	B6O	G2Db	R2Ib	B2Ob
P85	G5D	R5I	B5O	G1Db	R1Ib	B1Ob
P84	G4D	R4I	B4O	G0Db	R0Ib	B0Ob
P83	G3D	R3I	B3O	G3Df	R3If	B3Of
P82	G2D	R2I	B2O	G2Df	R2If	B2Of
P81	G1D	R1I	B1O	G1Df	R1If	B1Of
P80	G0D	R0I	B0O	G0Df	R0If	B0Of
P79	B7D	G7I	R7N	B3Db	G3Ib	R3Nb
P78	B6D	G6I	R6N	B2Db	G2Ib	R2Nb
P77	B5D	G5I	R5N	B1Db	G1Ib	R1Nb
P76	B4D	G4I	R4N	B0Db	G0Ib	R0Nb
P75	B3D	G3I	R3N	B3Df	G3If	R3Nf
P74	B2D	G2I	R2N	B2Df	G2If	R2Nf
P73	B1D	G1I	R1N	B1Df	G1If	R1Nf
P72	B0D	G0I	R0N	B0Df	G0If	R0Nf
P71	R7C	B7I	G7N	R3Cb	B3Ib	G3Nb
P70	R6C	B6I	G6N	R2Cb	B2Ib	G2Nb
P69	R5C	B5I	G5N	R1Cb	B1Ib	G1Nb
P68	R4C	B4I	G4N	R0Cb	B0Ib	G0Nb
P67	R3C	B3I	G3N	R3Cf	B3If	G3Nf
P66	R2C	B2I	G2N	R2Cf	B2If	G2Nf
P65	R1C	B1I	G1N	R1Cf	B1If	G1Nf
P64	R0C	B0I	G0N	R0Cf	B0If	G0Nf

Table A-6. RGB Modes, Non-Packed, Bits 63–0, Big-Endian

MODE	0 = PSEUDO	1 = OVERLAY + RGB				3 = RGB		
SUB MODE	N/A	0 = 4–4–4–4	1 = 1–5–5–5	2 = 8–8–8–8	3 = 8–4–4–4 DB	1 = 5–5–5 DB		3 = 5–6–5
DESCRIPT	INDEXED	O–R–G–B	O–R–G–B	O–R–G–B	DBUF SEL	FD = 0	FD = 1	R–G–B
COL DEPTH	8	16	16	32	32	32		16
BUS WIDTH	128	128	128	128	128	128		128
P63	P0H	B0D	B0D	B0B	B0Bf		B0B	B0D
P62	P1H	B1D	B1D	B1B	B1Bf		B1B	B1D
P61	P2H	B2D	B2D	B2B	B2Bf		B2B	B2D
P60	P3H	B3D	B3D	B3B	B3Bf		B3B	B3D
P59	P4H	G0D	B4D	B4B	B0Bb		B4B	B4D
P58	P5H	G1D	G0D	B5B	B1Bb		G0B	G0D
P57	P6H	G2D	G1D	B6B	B2Bb		G1B	G1D
P56	P7H	G3D	G2D	B7B	B3Bb		G2B	G2D
P55	P0G	R0D	G3D	G0B	G0Bf		G3B	G3D
P54	P1G	R1D	G4D	G1B	G1Bf		G4B	G4D
P53	P2G	R2D	R0D	G2B	G2Bf		R0B	G5D
P52	P3G	R3D	R1D	G3B	G3Bf		R1B	R0D
P51	P4G	O0D	R2D	G4B	G0Bb		R2B	R1D
P50	P5G	O1D	R3D	G5B	G1Bb		R3B	R2D
P49	P6G	O2D	R4D	G6B	G2Bb		R4B	R3D
P48	P7G	O3D	O0D	G7B	G3Bb			R4D
P47	P0F	B0C	B0C	R0B	R0Bf	B0B		B0C
P46	P1F	B1C	B1C	R1B	R1Bf	B1B		B1C
P45	P2F	B2C	B2C	R2B	R2Bf	B2B		B2C
P44	P3F	B3C	B3C	R3B	R3Bf	B3B		B3C
P43	P4F	G0C	B4C	R4B	R0Bb	B4B		B4C
P42	P5F	G1C	G0C	R5B	R1Bb	G0B		G0C
P41	P6F	G2C	G1C	R6B	R2Bb	G1B		G1C
P40	P7F	G3C	G2C	R7B	R3Bb	G2B		G2C
P39	P0E	R0C	G3C	O0B	O0B	G3B		G3C
P38	P1E	R1C	G4C	O1B	O1B	G4B		G4C
P37	P2E	R2C	R0C	O2B	O2B	R0B		G5C
P36	P3E	R3C	R1C	O3B	O3B	R1B		R0C
P35	P4E	O0C	R2C	O4B	O4B	R2B		R1C
P34	P5E	O1C	R3C	O5B	O5B	R3B		R2C
P33	P6E	O2C	R4C	O6B	O6B	R4B		R3C
P32	P7E	O3C	O0C	O7B	O7B	T0B	T0B	R4C

Table A-6. RGB Modes, Non-Packed, Bits 63–0, Big-Endian (Continued)

MODE	0 = PSEUDO	1 = OVERLAY + RGB				3 = RGB		
SUB MODE	N/A	0 = 4–4–4–4	1 = 1–5–5–5	2 = 8–8–8–8	3 = 8–4–4–4 DB	1 = 5–5–5 DB		3 = 5–6–5
DESCRIPT	INDXED	O–R–G–B	O–R–G–B	O–R–G–B	DBUF SEL	FD = 0	FD = 1	R–G–B
COL DEPTH	8	16	16	32	32	32		16
BUS WIDTH	128	128	128	128	128	128		128
P31	P0D	B0B	B0B	B0A	B0Af		B0A	B0B
P30	P1D	B1B	B1B	B1A	B1Af		B1A	B1B
P29	P2D	B2B	B2B	B2A	B2Af		B2A	B2B
P28	P3D	B3B	B3B	B3A	B3Af		B3A	B3B
P27	P4D	G0B	B4B	B4A	B0Ab		B4A	B4B
P26	P5D	G1B	G0B	B5A	B1Ab		G0A	G0B
P25	P6D	G2B	G1B	B6A	B2Ab		G1A	G1B
P24	P7D	G3B	G2B	B7A	B3Ab		G2A	G2B
P23	P0C	R0B	G3B	G0A	G0Af		G3A	G3B
P22	P1C	R1B	G4B	G1A	G1Af		G4A	G4B
P21	P2C	R2B	R0B	G2A	G2Af		R0A	G5B
P20	P3C	R3B	R1B	G3A	G3Af		R1A	R0B
P19	P4C	O0B	R2B	G4A	G0Ab		R2A	R1B
P18	P5C	O1B	R3B	G5A	G1Ab		R3A	R2B
P17	P6C	O2B	R4B	G6A	G2Ab		R4A	R3B
P16	P7C	O3B	O0B	G7A	G3Ab			R4B
P15	P0B	B0A	B0A	R0A	R0Af	B0A		B0A
P14	P1B	B1A	B1A	R1A	R1Af	B1A		B1A
P13	P2B	B2A	B2A	R2A	R2Af	B2A		B2A
P12	P3B	B3A	B3A	R3A	R3Af	B3A		B3A
P11	P4B	G0A	B4A	R4A	R0Ab	B4A		B4A
P10	P5B	G1A	G0A	R5A	R1Ab	G0A		G0A
P09	P6B	G2A	G1A	R6A	R2Ab	G1A		G1A
P08	P7B	G3A	G2A	R7A	R3Ab	G2A		G2A
P07	P0A	R0A	G3A	O0A	O0A	G3A		G3A
P06	P1A	R1A	G4A	O1A	O1A	G4A		G4A
P05	P2A	R2A	R0A	O2A	O2A	R0A		G5A
P04	P3A	R3A	R1A	O3A	O3A	R1A		R0A
P03	P4A	O0A	R2A	O4A	O4A	R2A		R1A
P02	P5A	O1A	R3A	O5A	O5A	R3A		R2A
P01	P6A	O2A	R4A	O6A	O6A	R4A		R3A
P00	P7A	O3A	O0A	O7A	O7A	T0A	T0A	R4A

Table A-7. RGB Modes, Non-Packed, Bits 127–64, Big-Endian

MODE	0 = PSEUDO	1 = OVERLAY + RGB				3 = RGB		
SUB MODE	N/A	0 = 4–4–4–4	1 = 1–5–5–5	2 = 8–8–8–8	3 = 8–4–4–4 DB	1 = 5–5–5 DB		3 = 5–6–5
DESCRIPT	INDEXED	O–R–G–B	O–R–G–B	O–R–G–B	DBUF SEL	FD = 0	FD = 1	R–G–B
COL DEPTH	8	16	16	32	32	32		16
BUS WIDTH	128	128	128	128	128	128		128
P127	P0P	B0H	B0H	B0D	B0Df		B0D	B0H
P126	P1P	B1H	B1H	B1D	B1Df		B1D	B1H
P125	P2P	B2H	B2H	B2D	B2Df		B2D	B2H
P124	P3P	B3H	B3H	B3D	B3Df		B3D	B3H
P123	P4P	G0H	B4H	B4D	B0Db		B4D	B4H
P122	P5P	G1H	G0H	B5D	B1Db		G0D	G0H
P121	P6P	G2H	G1H	B6D	B2Db		G1D	G1H
P120	P7P	G3H	G2H	B7D	B3Db		G2D	G2H
P119	P0O	R0H	G3H	G0D	G0Df		G3D	G3H
P118	P1O	R1H	G4H	G1D	G1Df		G4D	G4H
P117	P2O	R2H	R0H	G2D	G2Df		R0D	G5H
P116	P3O	R3H	R1H	G3D	G3Df		R1D	R0H
P115	P4O	O0H	R2H	G4D	G0Db		R2D	R1H
P114	P5O	O1H	R3H	G5D	G1Db		R3D	R2H
P113	P6O	O2H	R4H	G6D	G2Db		R4D	R3H
P112	P7O	O3H	O0H	G7D	G3Db			R4H
P111	P0N	B0G	B0G	R0D	R0Df	B0D		B0G
P110	P1N	B1G	B1G	R1D	R1Df	B1D		B1G
P109	P2N	B2G	B2G	R2D	R2Df	B2D		B2G
P108	P3N	B3G	B3G	R3D	R3Df	B3D		B3G
P107	P4N	G0G	B4G	R4D	R0Db	B4D		B4G
P106	P5N	G1G	G0G	R5D	R1Db	G0D		G0G
P105	P6N	G2G	G1G	R6D	R2Db	G1D		G1G
P104	P7N	G3G	G2G	R7D	R3Db	G2D		G2G
P103	P0M	R0G	G3G	O0D	O0D	G3D		G3G
P102	P1M	R1G	G4G	O1D	O1D	G4D		G4G
P101	P2M	R2G	R0G	O2D	O2D	R0D		G5G
P100	P3M	R3G	R1G	O3D	O3D	R1D		R0G
P99	P4M	O0G	R2G	O4D	O4D	R2D		R1G
P98	P5M	O1G	R3G	O5D	O5D	R3D		R2G
P97	P6M	O2G	R4G	O6D	O6D	R4D		R3G
P96	P7M	O3G	O0G	O7D	O7D	T0D	T0D	R4G

Table A-7. RGB Modes, Non-Packed, Bits 127–64, Big-Endian (Continued)

MODE	0 = PSEUDO	1 = OVERLAY + RGB				3 = RGB		
SUB MODE	N/A	0 = 4–4–4–4	1 = 1–5–5–5	2 = 8–8–8–8	3 = 8–4–4–4 DB	1 = 5–5–5 DB		3 = 5–6–5
DESCRIPT	INDEXED	O–R–G–B	O–R–G–B	O–R–G–B	DBUF SEL	FD = 0	FD = 1	R–G–B
COL DEPTH	8	16	16	32	32	32		16
BUS WIDTH	128	128	128	128	128	128		128
P95	P0L	B0F	B0F	B0C	B0Cf		B0C	B0F
P94	P1L	B1F	B1F	B1C	B1Cf		B1C	B1F
P93	P2L	B2F	B2F	B2C	B2Cf		B2C	B2F
P92	P3L	B3F	B3F	B3C	B3Cf		B3C	B3F
P91	P4L	G0F	B4F	B4C	B0Cb		B4C	B4F
P90	P5L	G1F	G0F	B5C	B1Cb		G0C	G0F
P89	P6L	G2F	G1F	B6C	B2Cb		G1C	G1F
P88	P7L	G3F	G2F	B7C	B3Cb		G2C	G2F
P87	P0K	R0F	G3F	G0C	G0Cf		G3C	G3F
P86	P1K	R1F	G4F	G1C	G1Cf		G4C	G4F
P85	P2K	R2F	R0F	G2C	G2Cf		R0C	G5F
P84	P3K	R3F	R1F	G3C	G3Cf		R1C	R0F
P83	P4K	O0F	R2F	G4C	G0Cb		R2C	R1F
P82	P5K	O1F	R3F	G5C	G1Cb		R3C	R2F
P81	P6K	O2F	R4F	G6C	G2Cb		R4C	R3F
P80	P7K	O3F	O0F	G7C	G3Cb			R4F
P79	P0J	B0E	B0E	R0C	R0Cf	B0C		B0E
P78	P1J	B1E	B1E	R1C	R1Cf	B1C		B1E
P77	P2J	B2E	B2E	R2C	R2Cf	B2C		B2E
P76	P3J	B3E	B3E	R3C	R3Cf	B3C		B3E
P75	P4J	G0E	B4E	R4C	R0Cb	B4C		B4E
P74	P5J	G1E	G0E	R5C	R1Cb	G0C		G0E
P73	P6J	G2E	G1E	R6C	R2Cb	G1C		G1E
P72	P7J	G3E	G2E	R7C	R3Cb	G2C		G2E
P71	P0I	R0E	G3E	O0C	O0C	G3C		G3E
P70	P1I	R1E	G4E	O1C	O1C	G4C		G4E
P69	P2I	R2E	R0E	O2C	O2C	R0C		G5E
P68	P3I	R3E	R1E	O3C	O3C	R1C		R0E
P67	P4I	O0E	R2E	O4C	O4C	R2C		R1E
P66	P5I	O1E	R3E	O5C	O5C	R3C		R2E
P65	P6I	O2E	R4E	O6C	O6C	R4C		R3E
P64	P7I	O3E	O0E	O7C	O7C	T0C	T0C	R4E

Table A-8. Packed RGB Mode, Bits 63–0, Big-Endian

MODE	4 = PACKED RGB								
SUB MODE	1 = 8–8–8								
COL DEPTH	24			24			24		
BUS WIDTH	32 (4:3 MUX RATIO)			64 (8:3 MUX RATIO)			128 (16:3 MUX RATIO)		
BUS LOAD	1ST	2ND	3RD	1ST	2ND	3RD	1ST	2ND	3RD
P63				G0C	B0F	R0H	G0C	R0H	B0N
P62				G1C	B1F	R1H	G1C	R1H	B1N
P61				G2C	B2F	R2H	G2C	R2H	B2N
P60				G3C	B3F	R3H	G3C	R3H	B3N
P59				G4C	B4F	R4H	G4C	R4H	B4N
P58				G5C	B5F	R5H	G5C	R5H	B5N
P57				G6C	B6F	R6H	G6C	R6H	B6N
P56				G7C	B7F	R7H	G7C	R7H	B7N
P55				B0C	R0E	G0H	B0C	G0H	R0M
P54				B1C	R1E	G1H	B1C	G1H	R1M
P53				B2C	R2E	G2H	B2C	G2H	R2M
P52				B3C	R3E	G3H	B3C	G3H	R3M
P51				B4C	R4E	G4H	B4C	G4H	R4M
P50				B5C	R5E	G5H	B5C	G5H	R5M
P49				B6C	R6E	G6H	B6C	G6H	R6M
P48				B7C	R7E	G7H	B7C	G7H	R7M
P47				R0B	G0E	B0H	R0B	B0H	G0M
P46				R1B	G1E	B1H	R1B	B1H	G1M
P45				R2B	G2E	B2H	R2B	B2H	G2M
P44				R3B	G3E	B3H	R3B	B3H	G3M
P43				R4B	G4E	B4H	R4B	B4H	G4M
P42				R5B	G5E	B5H	R5B	B5H	G5M
P41				R6B	G6E	B6H	R6B	B6H	G6M
P40				R7B	G7E	B7H	R7B	B7H	G7M
P39				G0B	B0E	R0G	G0B	R0G	B0M
P38				G1B	B1E	R1G	G1B	R1G	B1M
P37				G2B	B2E	R2G	G2B	R2G	B2M
P36				G3B	B3E	R3G	G3B	R3G	B3M
P35				G4B	B4E	R4G	G4B	R4G	B4M
P34				G5B	B5E	R5G	G5B	R5G	B5M
P33				G6B	B6E	R6G	G6B	R6G	B6M
P32				G7B	B7E	R7G	G7B	R7G	B7M

Table A-8. Packed RGB Mode, Bits 63–0, Big-Endian (Continued)

MODE	4 = PACKED RGB								
SUB MODE	1 = 8–8–8								
COL DEPTH	24			24			24		
BUS WIDTH	32 (4:3 MUX RATIO)			64 (8:3 MUX RATIO)			128 (16:3 MUX RATIO)		
BUS LOAD	1ST	2ND	3RD	1ST	2ND	3RD	1ST	2ND	3RD
P31	B0B	G0C	R0D	B0B	R0D	G0G	B0B	G0G	R0L
P30	B1B	G1C	R1D	B1B	R1D	G1G	B1B	G1G	R1L
P29	B2B	G2C	R2D	B2B	R2D	G2G	B2B	G2G	R2L
P28	B3B	G3C	R3D	B3B	R3D	G3G	B3B	G3G	R3L
P27	B4B	G4C	R4D	B4B	R4D	G4G	B4B	G4G	R4L
P26	B5B	G5C	R5D	B5B	R5D	G5G	B5B	G5G	R5L
P25	B6B	G6C	R6D	B6B	R6D	G6G	B6B	G6G	R6L
P24	B7B	G7C	R7D	B7B	R7D	G7G	B7B	G7G	R7L
P23	R0A	B0C	G0D	R0A	G0D	B0G	R0A	B0G	G0L
P22	R1A	B1C	G1D	R1A	G1D	B1G	R1A	B1G	G1L
P21	R2A	B2C	G2D	R2A	G2D	B2G	R2A	B2G	G2L
P20	R3A	B3C	G3D	R3A	G3D	B3G	R3A	B3G	G3L
P19	R4A	B4C	G4D	R4A	G4D	B4G	R4A	B4G	G4L
P18	R5A	B5C	G5D	R5A	G5D	B5G	R5A	B5G	G5L
P17	R6A	B6C	G6D	R6A	G6D	B6G	R6A	B6G	G6L
P16	R7A	B7C	G7D	R7A	G7D	B7G	R7A	B7G	G7L
P15	G0A	R0B	B0D	G0A	B0D	R0F	G0A	R0F	B0L
P14	G1A	R1B	B1D	G1A	B1D	R1F	G1A	R1F	B1L
P13	G2A	R2B	B2D	G2A	B2D	R2F	G2A	R2F	B2L
P12	G3A	R3B	B3D	G3A	B3D	R3F	G3A	R3F	B3L
P11	G4A	R4B	B4D	G4A	B4D	R4F	G4A	R4F	B4L
P10	G5A	R5B	B5D	G5A	B5D	R5F	G5A	R5F	B5L
P09	G6A	R6B	B6D	G6A	B6D	R6F	G6A	R6F	B6L
P08	G7A	R7B	B7D	G7A	B7D	R7F	G7A	R7F	B7L
P07	B0A	G0B	R0C	B0A	R0C	G0F	B0A	G0F	R0K
P06	B1A	G1B	R1C	B1A	R1C	G1F	B1A	G1F	R1K
P05	B2A	G2B	R2C	B2A	R2C	G2F	B2A	G2F	R2K
P04	B3A	G3B	R3C	B3A	R3C	G3F	B3A	G3F	R3K
P03	B4A	G4B	R4C	B4A	R4C	G4F	B4A	G4F	R4K
P02	B5A	G5B	R5C	B5A	R5C	G5F	B5A	G5F	R5K
P01	B6A	G6B	R6C	B6A	R6C	G6F	B6A	G6F	R6K
P00	B7A	G7B	R7C	B7A	R7C	G7F	B7A	G7F	R7K

Table A-9. Packed RGB Mode, Double Buffered, Bits 63–0, Big-Endian

MODE	4 = PACKED RGB								
SUB MODE	3 = 4–4–4 DB								
COL DEPTH	24			24			24		
BUS WIDTH	32 (4:3 MUX RATIO)			64 (8:3 MUX RATIO)			128 (16:3 MUX RATIO)		
BUS LOAD	1ST	2ND	3RD	1ST	2ND	3RD	1ST	2ND	3RD
P63				G0Cf	B0Ff	R0Hf	G0Cf	R0Hf	B0Nf
P62				G1Cf	B1Ff	R1Hf	G1Cf	R1Hf	B1Nf
P61				G2Cf	B2Ff	R2Hf	G2Cf	R2Hf	B2Nf
P60				G3Cf	B3Ff	R3Hf	G3Cf	R3Hf	B3Nf
P59				G0Cb	B0Fb	R0Hb	G0Cb	R0Hb	B0Nb
P58				G1Cb	B1Fb	R1Hb	G1Cb	R1Hb	B1Nb
P57				G2Cb	B2Fb	R2Hb	G2Cb	R2Hb	B2Nb
P56				G3Cb	B3Fb	R3Hb	G3Cb	R3Hb	B3Nb
P55				B0Cf	R0Ef	G0Hf	B0Cf	G0Hf	R0Mf
P54				B1Cf	R1Ef	G1Hf	B1Cf	G1Hf	R1Mf
P53				B2Cf	R2Ef	G2Hf	B2Cf	G2Hf	R2Mf
P52				B3Cf	R3Ef	G3Hf	B3Cf	G3Hf	R3Mf
P51				B0Cb	R0Eb	G0Hb	B0Cb	G0Hb	R0Mb
P50				B1Cb	R1Eb	G1Hb	B1Cb	G1Hb	R1Mb
P49				B2Cb	R2Eb	G2Hb	B2Cb	G2Hb	R2Mb
P48				B3Cb	R3Eb	G3Hb	B3Cb	G3Hb	R3Mb
P47				R0Bf	G0Ef	B0Hf	R0Bf	B0Hf	G0Mf
P46				R1Bf	G1Ef	B1Hf	R1Bf	B1Hf	G1Mf
P45				R2Bf	G2Ef	B2Hf	R2Bf	B2Hf	G2Mf
P44				R3Bf	G3Ef	B3Hf	R3Bf	B3Hf	G3Mf
P43				R0Bb	G0Eb	B0Hb	R0Bb	B0Hb	G0Mb
P42				R1Bb	G1Eb	B1Hb	R1Bb	B1Hb	G1Mb
P41				R2Bb	G2Eb	B2Hb	R2Bb	B2Hb	G2Mb
P40				R3Bb	G3Eb	B3Hb	R3Bb	B3Hb	G3Mb
P39				G0Bf	B0Ef	R0Gf	G0Bf	R0Gf	B0Mf
P38				G1Bf	B1Ef	R1Gf	G1Bf	R1Gf	B1Mf
P37				G2Bf	B2Ef	R2Gf	G2Bf	R2Gf	B2Mf
P36				G3Bf	B3Ef	R3Gf	G3Bf	R3Gf	B3Mf
P35				G0Bb	B0Eb	R0Gb	G0Bb	R0Gb	B0Mb
P34				G1Bb	B1Eb	R1Gb	G1Bb	R1Gb	B1Mb
P33				G2Bb	B2Eb	R2Gb	G2Bb	R2Gb	B2Mb
P32				G3Bb	B3Eb	G3Gb	G3Bb	R3Gb	B3Mb

Table A-9. Packed RGB Mode, Double Buffered, Bits 63–0, Big-Endian (Continued)

MODE	4 = PACKED RGB								
SUB MODE	3 = 4–4–4								
COL DEPTH	24			24			24		
BUS WIDTH	32 (4:3 MUX RATIO)			64 (8:3 MUX RATIO)			128 (16:3 MUX RATIO)		
BUS LOAD	1ST	2ND	3RD	1ST	2ND	3RD	1ST	2ND	3RD
P31	B0Bf	G0Cf	R0Df	B0Bf	R0Df	G0Gf	B0Bf	G0Gf	R0Lf
P30	B1Bf	G1Cf	R1Df	B1Bf	R1Df	G1Gf	B1Bf	G1Gf	R1Lf
P29	B2Bf	G2Cf	R2Df	B2Bf	R2Df	G2Gf	B2Bf	G2Gf	R2Lf
P28	B3Bf	G3Cf	R3Df	B3Bf	R3Df	G3Gf	B3Bf	G3Gf	R3Lf
P27	B0Bb	G0Cb	R0Db	B0Bb	R0Db	G0Gb	B0Bb	G0Gb	R0Lb
P26	B1Bb	G1Cb	R1Db	B1Bb	R1Db	G1Gb	B1Bb	G1Gb	R1Lb
P25	B2Bb	G2Cb	R2Db	B2Bb	R2Db	G2Gb	B2Bb	G2Gb	R2Lb
P24	B3Bb	G3Cb	R3Db	B3Bb	R3Db	G3Gb	B3Bb	G3Gb	R3Lb
P23	R0Af	B0Cf	G0Df	R0Af	G0Df	B0Gf	R0Af	B0Gf	G0Lf
P22	R1Af	B1CF	G1Df	R1Af	G1Df	B1Gf	R1Af	B1Gf	G1Lf
P21	R2Af	B2Cf	G2Df	R2Af	G2Df	B2Gf	R2Af	B2Gf	G2Lf
P20	R3Af	B3Cf	G3Df	R3Af	G3Df	B3Gf	R3Af	B3Gf	G3Lf
P19	R0Ab	B0Cb	G0Db	R0Ab	G0Db	B0Gb	R0Ab	B0Gb	G0Lb
P18	R1Ab	B1Cb	G1Db	R1Ab	G1Db	B1Gb	R1Ab	B1Gb	G1Lb
P17	R2Ab	B2Cb	G2Db	R2Ab	G2Db	B2Gb	R2Ab	B2Gb	G2Lb
P16	R3Ab	B3Cb	G3Db	R3Ab	G3Db	B3Gb	R3Ab	B3Gb	G3Lb
P15	G0Af	R0Bf	B0Df	G0Af	B0Df	R0Ff	G0Af	R0Ff	B0Lf
P14	G1Af	R1Bf	B1Df	G1Af	B1Df	R1Ff	G1Af	R1Ff	B1Lf
P13	G2Af	R2Bf	B2Df	G2Af	B2Df	R2Ff	G2Af	R2Ff	B2Lf
P12	G3Af	R3Bf	B3Df	G3Af	B3Df	R3Ff	G3Af	R3Ff	B3Lf
P11	G0Ab	R0Bb	B0Db	G0Ab	B0Db	R0Fb	G0Ab	R0Fb	B0Lb
P10	G1Ab	R1Bb	B1Db	G1Ab	B1Db	R1Fb	G1Ab	R1Fb	B1Lb
P09	G2Ab	R2Bb	B2Df	G2Ab	B2Db	R2Fb	G2Ab	R2Fb	B2Lb
P08	G3Ab	R3Bb	B3Df	G3Ab	B3Db	R3Fb	G3Ab	R3Fb	B3Lb
P07	B0Af	G0Bf	R0Cf	B0Af	R0Cf	G0Ff	B0Af	G0Ff	R0Kf
P06	B1Af	G1Bf	R1Cf	B1Af	R1Cf	G1Ff	B1Af	G1Ff	R1Kf
P05	B2Af	G2Bf	R2Cf	B2Af	R2Cf	G2Ff	B2Af	G2Ff	R2Kf
P04	B3Af	G3Bf	R3Cf	B3Af	R3Cf	G3Ff	B3Af	G3Ff	R3Kf
P03	B0Ab	G0Bb	R0Cb	B0Ab	R0Cb	G0Fb	B0Ab	G0Fb	R0Kb
P02	B1Ab	G1Bb	R1Cb	B1Ab	R1Cb	G1Fb	B1Ab	G1Fb	R1Kb
P01	B2Ab	G2Bb	R2Cb	B2Ab	R2Cb	G2Fb	B2Ab	G2Fb	R2Kb
P00	B3Ab	G3Bb	R3Cb	B3Ab	R3Cb	G3Fb	B3Ab	G3Fb	R3Kb

Table A-10. Packed Modes, Bits 127–64, Big-Endian

MODE	4 = PACKED RGB					
SUB MODE	1 = 8–8–8			3 = 4–4–4 DB		
COL DEPTH	24			24		
BUS WIDTH	128 (16:3 MUX RATIO)			128 (16:3 MUX RATIO)		
BUS LOAD	1ST	2ND	3RD	1ST	2ND	3RD
P127	B0F	G0K	R0P	B0Ff	G0Kf	R0Pf
P126	B1F	G1K	R1P	B1Ff	G1Kf	R1Pf
P125	B2F	G2K	R2P	B2Ff	G2Kf	R2Pf
P124	B3F	G3K	R3P	B3Ff	G3Kf	R3Pf
P123	B4F	G4K	R4P	B0Fb	G0Kb	R0Pb
P122	B5F	G5K	R5P	B1Fb	G1Kb	R1Pb
P121	B6F	G6K	R6P	B2Fb	G2Kb	R2Pb
P120	B7F	G7K	R7P	B3Fb	G3Kb	R3Pb
P119	R0E	B0K	G0P	R0Ef	B0Kf	G0Pf
P118	R1E	B1K	G1P	R1Ef	B1Kf	G1Pf
P117	R2E	B2K	G2P	R2Ef	B2Kf	G2Pf
P116	R3E	B3K	G3P	R3Ef	B3Kf	G3Pf
P115	R4E	B4K	G4P	R0Eb	B0Kb	G0Pb
P114	R5E	B5K	G5P	R1Eb	B1Kb	G1Pb
P113	R6E	B6K	G6P	R2Eb	B2Kb	G2Pb
P112	R7E	B7K	G7P	R3Eb	B3Kb	G3Pb
P111	G0E	R0J	B0P	G0Ef	R0Jf	B0Pf
P110	G1E	R1J	B1P	G1Ef	R1Jf	B1Pf
P109	G2E	R2J	B2P	G2Ef	R2Jf	B2Pf
P108	G3E	R3J	B3P	G3Ef	R3Jf	B3Pf
P107	G4E	R4J	B4P	G0Eb	R0Jb	B0Pb
P106	G5E	R5J	B5P	G1Eb	R1Jb	B1Pb
P105	G6E	R6J	B6P	G2Eb	R2Jb	B2Pb
P104	G7E	R7J	B7P	G3Eb	R3Jb	B3Pb
P103	B0E	G0J	R0O	B0Ef	G0Jf	R0Of
P102	B1E	G1J	R1O	B1Ef	G1Jf	R1Of
P101	B2E	G2J	R2O	B2Ef	G2Jf	R2Of
P100	B3E	G3J	R3O	B3Ef	G3Jf	R3Of
P99	B4E	G4J	R4O	B0Eb	G0Jb	R0Ob
P98	B5E	G5J	R5O	B1Eb	G1Jb	R1Ob
P97	B6E	G6J	R6O	B2Eb	G2Jb	R2Ob
P96	B7E	G7J	R7O	B3Eb	G3Jb	R3Ob

Table A-10. Packed Modes, Bits 127–64, Big-Endian (Continued)

MODE	4 = PACKED RGB					
SUB MODE	1 = 8–8–8			3 = 4–4–4 DB		
COL DEPTH	24			24		
BUS WIDTH	128 (16:3 MUX RATIO)			128 (16:3 MUX RATIO)		
BUS LOAD	1ST	2ND	3RD	1ST	2ND	3RD
P95	R0D	B0J	G0O	R0Df	B0Jf	G0Of
P94	R1D	B1J	G1O	R1Df	B1Jf	G1Of
P93	R2D	B2J	G2O	R2Df	B2Jf	G2Of
P92	R3D	B3J	G3O	R3Df	B3Jf	G3Of
P91	R4D	B4J	G4O	R0Db	B0Jb	G0Ob
P90	R5D	B5J	G5O	R1Db	B1Jb	G1Ob
P89	R6D	B6J	G6O	R2Db	B2Jb	G2Ob
P88	R7D	B7J	G7O	R3Db	B3Jb	G3Ob
P87	G0D	R0I	B0O	G0Df	R0If	B0Of
P86	G1D	R1I	B1O	G1Df	R1If	B1Of
P85	G2D	R2I	B2O	G2Df	R2If	B2Of
P84	G3D	R3I	B3O	G3Df	R3If	B3Of
P83	G4D	R4I	B4O	G0Db	R0Ib	B0Ob
P82	G5D	R5I	B5O	G1Db	R1Ib	B1Ob
P81	G6D	R6I	B6O	G2Db	R2Ib	B2Ob
P80	G7D	R7I	B7O	G3Db	R3Ib	B3Ob
P79	B0D	G0I	R0N	B0Df	G0If	R0Nf
P78	B1D	G1I	R1N	B1Df	G1If	R1Nf
P77	B2D	G2I	R2N	B2Df	G2If	R2Nf
P76	B3D	G3I	R3N	B3Df	G3If	R3Nf
P75	B4D	G4I	R4N	B0Db	G0Ib	R0Nb
P74	B5D	G5I	R5N	B1Db	G1Ib	R1Nb
P73	B6D	G6I	R6N	B2Db	G2Ib	R2Nb
P72	B7D	G7I	R7N	B3Db	G3Ib	R3Nb
P71	R0C	B0I	G0N	R0Cf	B0If	G0Nf
P70	R1C	B1I	G1N	R1Cf	B1If	G1Nf
P69	R2C	B2I	G2N	R2Cf	B2Ib	G2Nf
P68	R3C	B3I	G3N	R3Cf	B3If	G3Nf
P67	R4C	B4I	G4N	R0Cb	B0Ib	G0Nb
P66	R5C	B5I	G5N	R1Cb	B1Ib	G1Nb
P65	R6C	B6I	G6N	R2Cb	B2Ib	G2Nb
P64	R7C	B7I	G7N	R3Cb	B3Ib	G3Nb

Appendix B
PLL Programming

The C program below illustrates an algorithm which can be used to determine register values for the PCLK and MCLK PLLs. The user enters the target frequency at the PLL output (in MHz). The program scans all possible N, M, and P combinations and tests for a VCO frequency within the required limits. The program output is the N, M, and P register settings which results in an output frequency closet to the target frequency.

Often, several N, M, and P combinations result in the same output frequency. In this case, the combination with the smallest N is chosen. The smallest N produces the highest frequency at the PLLs phase detector and results in the best jitter performance.

The algorithm chooses the smallest N because it starts with the minimum N value and increments up. The first time the final output frequency is found is the N, M, and P combination with the smallest N. Any subsequent combinations with the same output frequency are discarded since the criteria for replacement is less than (not less than or equal to).

```
#include <math.h>
#include <stdio.h>
#define REFERENCE   14.31818
#define MIN_N       2
#define MAX_N       7
#define MIN_M       2
#define MAX_M       255
#define MIN_P       1
#define MAX_P       16
#define MIN_VCO     110.0
#define MAX_VCO     250.0

struct PLLInfo
{  int    n;
   int    m;
   int    p;
   double vco;
   double out; };

int PowOf2 (int p);

int main (void)
{
struct PLLInfo temp, result;
double ftarget = 220.0;
int first_pass = 1;

fprintf (stderr, "\nTarget Frequency (MHz) ->");
scanf ("%lf",&ftarget);
for (temp.p=MIN_P; temp.p<=MAX_P; temp.p*=2)
{
   for (temp.m=MIN_M; temp.m<=MAX_M; temp.m++)
   {
```

```
       for (temp.n=MIN_N; temp.n<=MAX_M; temp.n++)
       {
          temp.vco=REFERENCE* (double)temp.m/(double)temp.n;
          temp.out=temp.vco/temp.p;
          if ((temp.vco>=MIN_VCO) && (temp.vco<=MAX_VCO))
          {
             if (first_pass)
             {
                result=temp;
                first_pass=0;
             }
             else if (fabs(temp.out-ftarget)<fabs(result.out-ftarget))
                result=temp;
             }
          }/*for (n...*/
       }/*for (m...*/
    }/*for (p=...*/
    printf("\nPLL Frequency    ->%6.2lf MHz", result.out);
    printf("\nVCO Frequency    ->%6.2lf MHz", result.vco);
    printf("\nN-Value Register ->%2.2X HEX", result.n);
    printf("\nM-Value Register ->%2.2X HEX", result.m);
    printf("\nP-Value Register ->%2.2X HEX\n",PowOf2(result.p)+0x80);
    return (0);
}
int PowOf2 (int i)
{
    in power=0;
    for (;i>1;i/=2)
       power++;
    return (power);
}
-------------------------
Example Program Execution
-------------------------
Target Frequency (MHz) -> 100
PLL Frequency          -> 100.23 MHz
VCO Frequency          -> 200.45 MHz
N-Value Register       -> 02 HEX
M-Value Register       -> 1C HEX
P-Value Register       -> 81 HEX
```

Appendix C
PC-Board Layout Considerations

C.1 PC-Board Considerations

It is recommended that a 4-layer PC board be used with the TVP3033 video interface palette: one layer for 3.3-V power, one for GND, and two for signals. The layout should be optimized for the lowest noise on the TVP3033 power and ground lines by shielding the digital inputs and providing good decoupling. The lead length between groups of analog V_{DD} and GND terminals (see Figure C–1) should be minimized so as to minimize inductive ringing. The TVP3033 should be located as close as possible to the output connectors to minimize noise pickup and reflections due to impedance mismatch.

The analog outputs are susceptible to crosstalk from digital lines; digital traces must not be routed under or adjacent to the analog output traces.

For maximum performance, the analog-video-output impedance, cable impedance, and load impedance should be the same. The load resistor connection between the video outputs and GND should be as close as possible to the TVP3033 to minimize reflections. Unused analog outputs should be connected to GND.

Analog output video edges exceeding the CRT monitor bandwidth can be reflected, producing cable-length-dependent ghosts. Simple pulse filters can reduce high-frequency energy, thus reducing EMI and noise. The filter impedance must match the line impedance.

C.2 Ground Plane

It is also recommended that only one ground plane be used for both the TVP3033 and the rest of the logic. Separate digital and analog ground planes are not needed and can potentially cause system problems.

C.3 Power Plane

Split-power planes for the TVP3033 and the rest of the logic are recommended. The TVP3033 VIP analog circuitry should have its own power plane, referred to as AV_{DD}. These two power planes should be connected at a single point through a ferrite bead. This bead should be located as near as possible to where the power supply connects to the board. To maximize the high-frequency power supply rejection, the video output signals should not overlay the analog power plane.

C.4 Supply Decoupling

All capacitors should be in surface mount packages. This reduces the lead inductance and is consistent with reliable operation.

For the best performance, a 0.1-µF ceramic capacitor in parallel with a 0.01-µF chip capacitor should be used to decouple each of the groups of power terminals to GND. These capacitors should be placed as close as possible to the device.

If a switching power supply is used, the designer should pay close attention to reducing power supply noise and consider using a 3-terminal voltage regulator for supplying power to AV_{DD}.

C.5 COMP and REF Terminals

A 0.1-µF ceramic capacitor should be connected between COMP1 and COMP2 to avoid noise and color-smearing problems. A 0.1-µF ceramic capacitor is also recommended between GND and REF to further stabilize the output image. This 0.1-µF capacitor is needed for either internal or external voltage references. These capacitor values may depend on the board layout; experimentation may be required in order to determine optimum values.

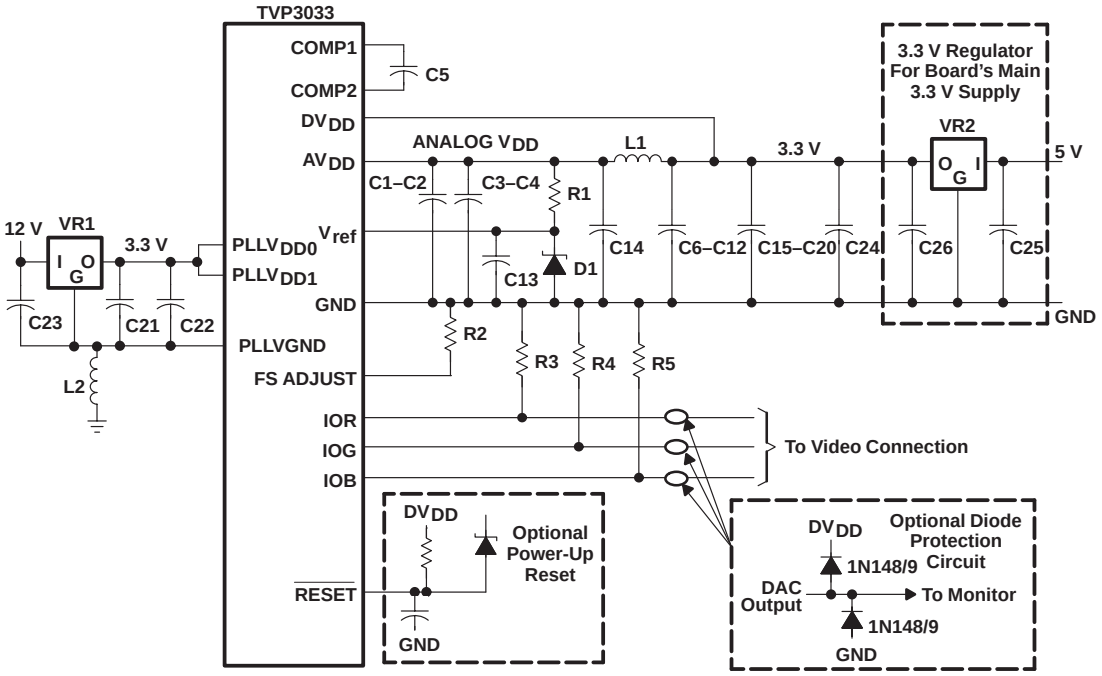
C.6 Analog Output Protection

The TVP3033 analog output should be protected against high-energy discharges, such as those from monitor arc-over or from hot-switching ac-coupled monitors.

The diode protection circuit shown in Figure C–1 can prevent latch-up under severe discharge conditions without adversely degrading analog transition times. The IN4148/9 parts are low-capacitance, fast-switching diodes, which are also available in multiple-device packages (FSA250X or FSA270X) or surface-mountable pairs (BAV99 or MMBD7001).

C.7 PLL Supply

A separate 3.3-V regulator is required for the PLL supply. A typical circuit is shown in Figure C–1.

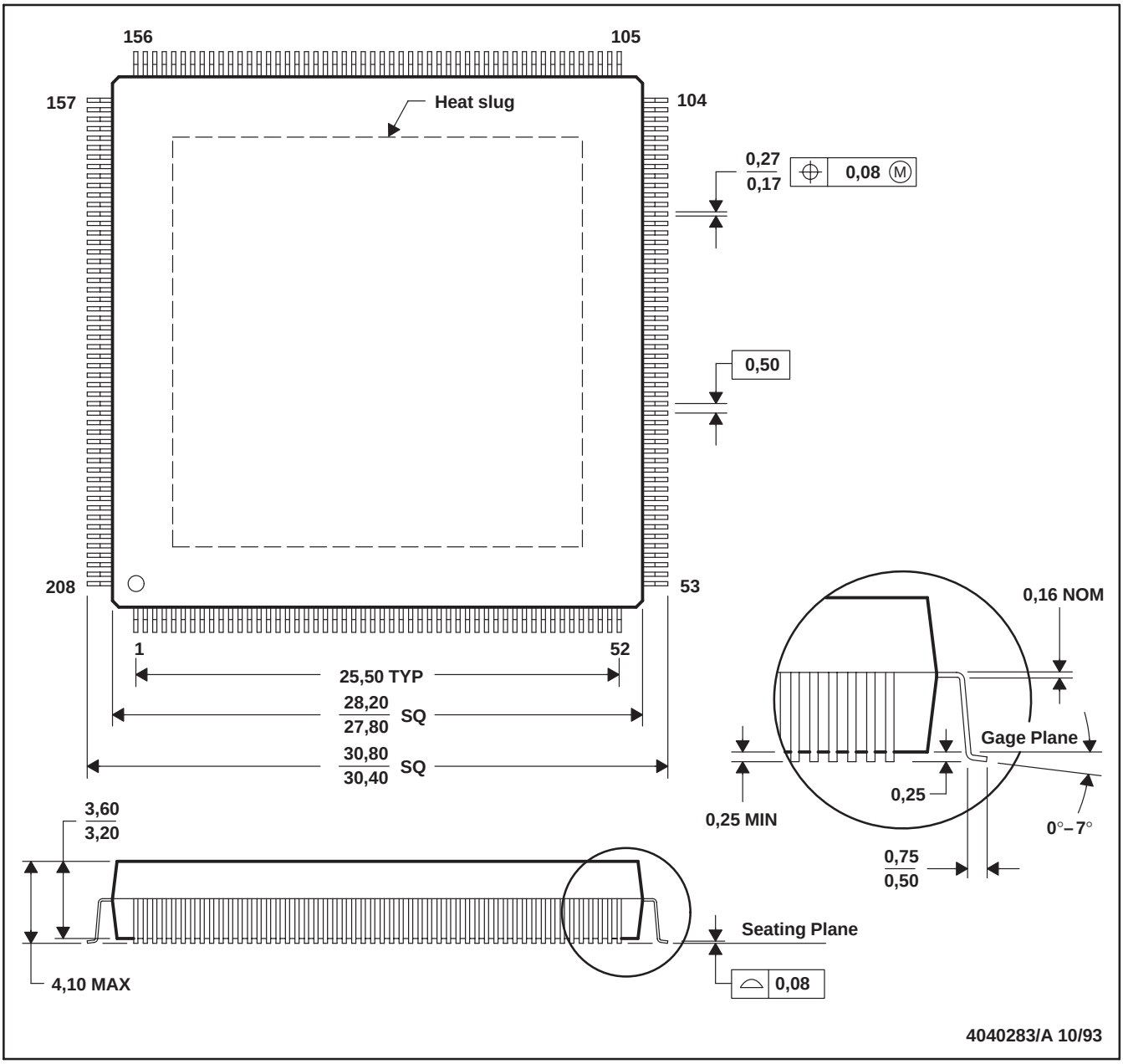


REFERENCE DESIGNATOR	DESCRIPTION
C1, C2, C5–C13, C24–C25	0.1 μF-surface-mount, ceramic capacitor
C3, C4, C15–C20, C22, C23	0.01 μF-surface-mount, chip capacitor
C21, C26	10 μF-tantalum capacitor
C14	33 μF-tantalum capacitor
D1	TI LM385-1.2 1.2 V voltage reference
L1, L2	ferrite bead
R1	1 kΩ, 1%, surface mount resistor
R2	523 Ω, 1%, surface mount resistor
R3–R5	75 Ω, 1%, surface mount resistor
VR1	TI TPS7233QD 3.3 V voltage regulator, 100 mA
VR2	3.3 V voltage regulator, 5–10 A

Figure C–1. Typical Connection Diagram and Parts List

Appendix D
Mechanical Data

PPA (S-PQFP-G208) **PLASTIC QUAD FLATPACK**



NOTES: A. All linear dimensions are in millimeters.
B. This drawing is subject to change without notice.
C. Thermally enhanced molded plastic package with a heat slug (HSL).
D. Falls within JEDEC MO-143

IMPORTANT NOTICE

Texas Instruments and its subsidiaries (TI) reserve the right to make changes to their products or to discontinue any product or service without notice, and advise customers to obtain the latest version of relevant information to verify, before placing orders, that information being relied on is current and complete. All products are sold subject to the terms and conditions of sale supplied at the time of order acknowledgement, including those pertaining to warranty, patent infringement, and limitation of liability.

TI warrants performance of its semiconductor products to the specifications applicable at the time of sale in accordance with TI's standard warranty. Testing and other quality control techniques are utilized to the extent TI deems necessary to support this warranty. Specific testing of all parameters of each device is not necessarily performed, except those mandated by government requirements.

CERTAIN APPLICATIONS USING SEMICONDUCTOR PRODUCTS MAY INVOLVE POTENTIAL RISKS OF DEATH, PERSONAL INJURY, OR SEVERE PROPERTY OR ENVIRONMENTAL DAMAGE ("CRITICAL APPLICATIONS"). TI SEMICONDUCTOR PRODUCTS ARE NOT DESIGNED, AUTHORIZED, OR WARRANTED TO BE SUITABLE FOR USE IN LIFE-SUPPORT DEVICES OR SYSTEMS OR OTHER CRITICAL APPLICATIONS. INCLUSION OF TI PRODUCTS IN SUCH APPLICATIONS IS UNDERSTOOD TO BE FULLY AT THE CUSTOMER'S RISK.

In order to minimize risks associated with the customer's applications, adequate design and operating safeguards must be provided by the customer to minimize inherent or procedural hazards.

TI assumes no liability for applications assistance or customer product design. TI does not warrant or represent that any license, either express or implied, is granted under any patent right, copyright, mask work right, or other intellectual property right of TI covering or relating to any combination, machine, or process in which such semiconductor products or services might be or are used. TI's publication of information regarding any third party's products or services does not constitute TI's approval, warranty or endorsement thereof.