HORIZON

Mercury 24
Specification

Revision XA1

January 20, 1993

Jeff Tingley
RasterOps Corporation

PART # 0002-0499

CONFIDENTIAL

1.0 Introduction

This document establishes the physical and electrical characteristics for the Mercury 24. The Mercury 24 is a high resolution, graphics display board for the Macintosh computer. The Mercury 24 occupies one NuBus slot in the Macintosh. It is capable of six pixel depths, multiple resolutions, and has a QuickDraw accelerator. The connectors for a MIPS co-processor or DSP based daughter board are also included. A block diagram of the Mercury 24 is shown in Figure 1-1. This document is the property of RasterOps. Reproduction of this document is prohibited.

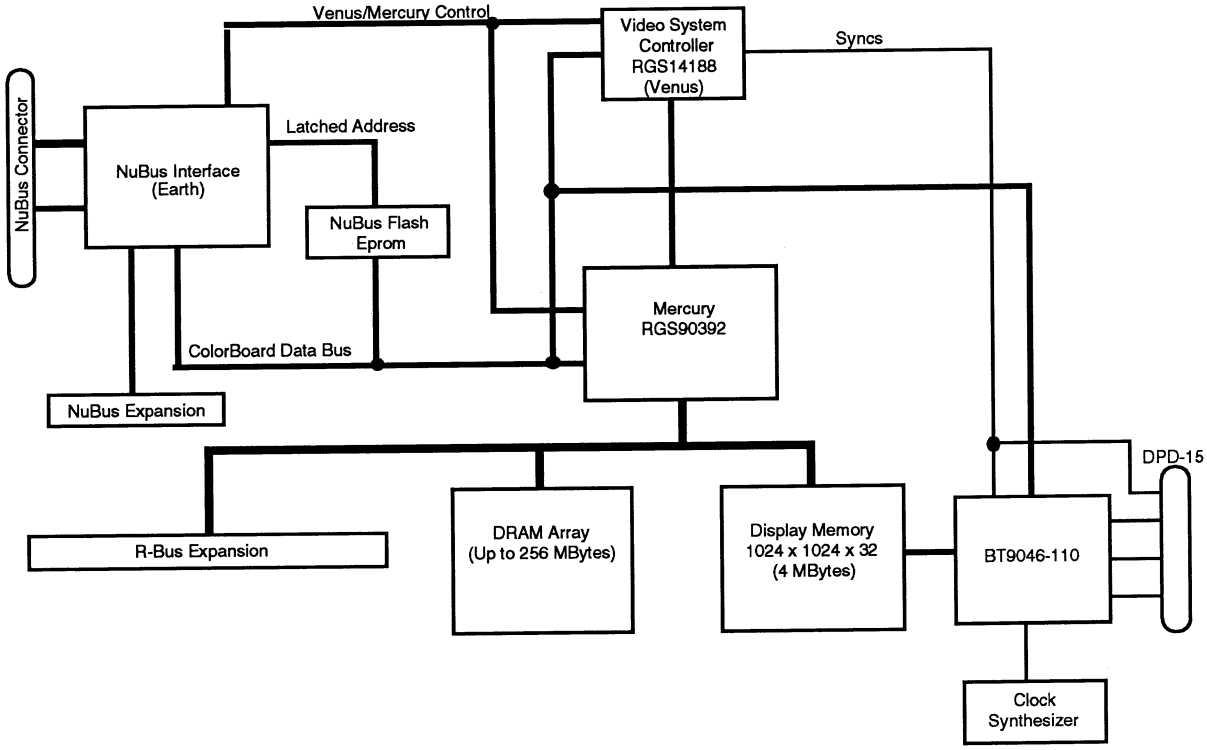


Figure 1-1 Mercury 24 Block Diagram

2.0 Hardware Description

2.1 Features

Switchable Resolutions of (Software Selectable stored in EEPROM or Monitor ID Selectable)

- 1280 x 1024 (16-bit, 75 Hz Vertical Refresh, Non-interlaced)
- 1152 x 870 (75 Hz Vertical Refresh, Non-interlaced)
- 1024 x 768 (60,75 Hz Vertical Refresh, Non-interlaced)
- 832 x 624 (75 Hz Vertical Refresh, Non-interlaced)
- 640 x 870 (75 Hz Vertical Refresh, Non-interlaced)
- 640 x 480 (66 Hz Vertical Refresh, Non-interlaced)

Pixel Depths of 1, 2, 4, 8, 16, and 24-bits

NuBus Block Mode Master/Slave Compatible FOR VRAM ACCESS

134.2 Million Color Palette (9 bit DACs)

Mercury based QuickDraw Accelerator

RS-343 Video Compatible

15-Pin D-Submininature Connector (Same as Apple)

Hardware Zoom of 1x, 2x, 3x... to 16x

Smooth Pan independent of pixel depth

2.2 Overview

The Mercury 24 is a stand alone frame buffer. It can display up to 24-bits per pixel in one of many resolutions. It is capable of integer zoom and implements smooth paning in all bit depths. The DACs are 9-bit instead of 8-bit to allow for gammas up to 2.4 without redundant colors. A Mercury based QuickDraw Accelerator is included for maximum performance. As an added feature, the Mercury 24 also supports an MIPS coprocessor or a DSP based daughter card.

3.0 Technical Data

3.1. Absolute Maximum Ratings

Supply Voltage, VCC	5.25 Volts
Operating Ambient Temperature Range	0 °C to 60 °C
Storage Temperature	-65 °C to 150 °C
ICC (Supply Current)	4.0 A @ 5.25 V

3.2 Reference Drawings

The following items should be referenced for information on the Mercury 24.

Mercury 24 Top Assembly LM	2642
Mercury 24 Schematics	0002-0496-11
Mercury 24 PCB Fab	0002-0497
Mercury 24 PCA	0002-0498-10
Mercury 24 Manual	0700-0230

4.0 Addressing

The Mercury 24 runs in "Fat Slot" (16 MByte of addressable space) and "Super Slot" (256 MBytes of addressable space) mode. A detailed address map for the 16 MByte space is shown in Figure 4-1.

Address Range	Access
FsF8.0000 - FsFF.FFFF	Flash Rom (512 KBytes)
FsF0.0600 - FsF0.06FF	RGS14188 (256 KBytes)
FsF0.0500 - FsF0.05FF	Mercury (256 Bytes)
FsF0.0400 - FsF0.04FF	Jupiter 4 (128 Bytes Aliased)
FsF0.0300 - FsF0.03FF	Jupiter 3 (128 Bytes Aliased)
FsF0.0200 - FsF0.02FF	Jupiter 2 (128 Bytes Aliased)
FsF0.0100 - FsF0.01FF	Jupiter 1 (128 Bytes Aliased)
FsF0.0000 - FsF0.00FF	DAC (16 Bytes Aliased)
FsE0.0000 - FsEF.FFFF	Accelerator Registers (No-ACK)
FsD0.0000 - FsDF.FFFF	Unused (No-ACK)
Fs80.0000 - FsCF.FFFF	Expansion Slot Space (No-ACK)
Fs40.0000 - Fs7F.FFFF	Frame Buffer Expansion (ACK)/G-World (No-ACK)
Fs00.0000 - Fs3F.FFFF	Frame Buffer

Figure 4-1 Address Map

The 256 MByte "Super Slot" address space is accessed with addresses of the form s000.0000 where s is the slot number. This "Super Slot" space contains up to 256 MBytes of Dynamic RAM usable for anything desired.

4.1 Video System Controller - RGS14188

A single RGS14188 is used to control the video display memory of the Mercury 24. A brief description of the pertinent registers is given in the following sections. For more information on the RGS14188 refer to the RasterOps RGS14188 User's Guide part number 0002-0272.

4.1.1 Interface Control Registers

These registers are used to control the interface and internal configuration of the RGS14188.

Register	Address	Function
Control 1	Base + 8'h00	RAM Size and Arbitration etc.
Control 2	Base + 8'h04	Page Mode Ctrl and General I/O Funct.
Control 3	Base + 8'h08	Video Timing Interface Control
Control 4	Base + 8'h0C	Interrupt and Ready Control.
Status	Base + 8'h10	Interrupt and CRT Control Status

Table 4-1 - Interface Control Registers

4.1.2 Refresh Control

This register sets the internal refresh request interval. It should be set to approximately '9C' hex.

Register	Address	Function
Rfsh Interval	Base + 8'h14	Refresh Interval Duration

Table 4-2 - Refresh Control

4.1.3 Screen Update Control Registers

These registers are used to control how the actual display is updated. They provide panning capability and other important functions for screen update operation.

Register	Byte	Address	Function
Display Start	Low	Base + 8'h2C	Upper Left Corner of Display
	Middle	Base + 8'h28	
	High	Base + 8'h24	
Half Row Increment	Low	Base + 8'h3C	Half Row Address for VRAM
	Middle	Base + 8'h38	
	High	Base + 8'h34	
Display Pitch	Low	Base + 8'h4C	Vert. Adj. Pixels
	Middle	Base + 8'h48	
	High	Base + 8'h44	
Column Address Mask		Base + 8'h50	Mask for Midlines
Horizontal Latency		Base + 8'h54	Holdoff Time for Xfer Cycles

Table 4-3 - Screen Update Control

4.1.4 Horizontal Timing Control

These registers are used to control all the horizontal screen timing parameters for interlaced and non-interlaced screens.

Register	Byte	Address	Function
Horizontal End Sync	Low	Base + 8'h5C	Sync Duration
	High	Base + 8'h58	
Horizontal End Blank	Low	Base + 8'h64	Sync + Back Porch
	High	Base + 8'h60	
Horizontal Start Blank	Low	Base + 8'h6C	Sync + Back Porch + Visible
	High	Base + 8'h68	
Horizontal Total	Low	Base + 8'h74	Total Visible
	High	Base + 8'h70	
Horizontal Half Line	Low	Base + 8'h7C	Duration for half a line
	High	Base + 8'h78	
Horizontal Count	Low	Base + 8'h84	
	High	Base + 8'h80	

Table 4-4 - Horizontal Timing Control

4.1.5 Vertical Timing Control

These registers are used to control all the vertical screen timing parameters for interlaced and non-interlaced screens.

Register	Byte	Address	Function
Vertical End Sync		Base + 8'h88	Sync Duration
Vertical End Blank		Base + 8'h8C	Sync + Back Porch
Vertical Start Blank	Low	Base + 8'h94	Sync + Back Porch + Visible
	High	Base + 8'h90	
Vertical Total	Low	Base + 8'h9C	Total Visible
	High	Base + 8'h98	
Vertical Count	Low	Base + 8'hA4	Start Value for Vertical Cntr.
	High	Base + 8'hA0	
Vertical Interrupt Line	Low	Base + 8'hAC	Interrupt Line
	High	Base + 8'hA8	
Y-Zoom		Base + 8'hB8	Y-Zoom Factor 1-256X

Table 4-5 - Vertical Timing Control

4.1.6 General Purpose Registers

These registers are used to control the general input/output functions of the RGS14188. Figure 4-2 shows exactly how each bit is defined for the Mercury 24 system. The GIO Configuration register should be set to 16'hFFBF.

Register	Byte	Address	Function
General I/O Configuration	Low	Base + 8'h18	Configure Low Byte (0=in)
	high	Base + 8'h1C	Configure High Byte (0=in)
General I/O	Low	Base + 8'hB0	Actual Input/Output Pins
	High	Base + 8'hB4	
Software Register		Base + 8'hBC	General Purpose Register

Table 4-6 - General Purpose Registers

High Byte

GIO15	GIO14	GIO13	GIO12	GIO11	GIO10	GIO9	GIO8
N.C.	N.C.	N.C.	N.C.	N.C.	ID2	ID1	ID0

Low Byte

GIO7	GIO6	GIO5	GIO4	GIO3	GIO2	GIO1	GIO0
Serial Chip Sel	Serial Data Out	Serial Data In	Serial Clock	N.C.	N.C.	ICD DAT	ICD CLK

Figure 4-2 General I/O Definitions

ICD CLK, ICD DATA A new clock synthesizer (ICD 2062) is used on the Mercury 24. The ICD 2062 is fully programmable. Both the pixel clock as well as the system clocks are generated by this IC. Table 4-7 shows the serial bit streams required for the various frequencies.

M Clock Frequency	Serial Bit Stream
40.0000 MHz	0637C2F

V Clock Frequency	Serial Bit Stream
30.240 MHz	009F0A8
57.296 MHz	0099C23
64.000 MHz	00BF426
80.000 MHz	00FF41E
100.000 MHz	01BFC18
107.619 MHz	03BB813

Table 4-7 Clock Chip Setup

Serial Clock
This bit is connected to the EEPROM's Serial Clock pin.

Serial Data In
This bit is connected to the EEPROM's Data In pin.

Serial Data Out
This bit is connected to the EEPROM's Data Out pin.

Serial Chip En
This bit is connected to the EEPROM's Chip Enable pin.

Monitor ID0
This bit is connected to the Monitor ID0 pin of the video connector.
It is first sampled along with ID1 and ID2 for valid patterns. If no valid pattern is found then it is driven low while sampling ID1 and ID2. If no pattern is still found then use the EEPROM data like our current products.

Monitor ID1
This bit is connected to the Monitor ID1 pin of the video connector.
It is first sampled along with ID0 and ID2 for valid patterns. If no valid pattern is found then it is driven low while sampling ID0 and ID2. If no pattern is still found then use the EEPROM data like our current products.

Monitor ID2
This bit is connected to the Monitor ID2 pin of the video connector.
It is first sampled along with ID0 and ID1 for valid patterns. If no valid pattern is found then it is driven low while sampling ID0 and ID1. If no pattern is still found then use the EEPROM data like our current products.

4.2 Mercury

4.2.1 Introduction

The Mercury chip is intended to be an acceleration co-processor and data path controller for the RGS14188 (Venus) chip. It takes advantage of a very fast, highly parallel architecture to speed up drawing, using Block Transfers (BLTs). In addition many Boolean, arithmetic and "special" operations can be performed on the data to be BLT. Mercury is designed to be interfaced to a Dynamic RAM array and a Video RAM array. It contains one bank of internal Static RAM (SRAM) which is 320 words deep by 128 bits per word (320 x 128). The SRAM is used for fast temporary storage of data. Mercury is a byte addressable machine. For the most detailed Mercury information please refer to the Mercury specification.

4.2.2 Acceleration

Mercury has several modes of acceleration and can manipulate linear, or 2-dimensional pixel arrays. It is also capable of doing pixel operations on four 32-bit pixels, eight 16-bit pixels, or sixteen 8-bit pixels at one time. In addition to all of its acceleration capabilities, Mercury acts as a data path controller for the RGS14188 chip and as a host interface for a large dynamic RAM array. It will automatically route a standard 32-bit Big Endian or Little Endian data bus to its high speed parallel bus for RAM read and write operations.

All acceleration is based on "Source/Destination BLTs" (Block Transfers). A Source/Destination BLT, using very high speed accesses, reads the data at the source (DRAM, VRAM or Internal Register), operates on it, and returns the new data to the destination. The internal Static RAM is used to store these blocks of data when necessary. Be sure to keep in mind that Mercury is a byte oriented machine.

4.2.3 Internal Registers

Mercury has several internal registers all of which are a maximum of 32-bits wide. Each register has read/write ability and must be accessed as a long (32-bit) quantity. AD[31:0] will contain the data to be written or read. Table 4-8 shows the address map for all the internal mercury registers.

Register	Address	Register	Address
Status	base + 8'h00	S_START	base + 8'h40
System Control	base + 8'h04	S_XEND	base + 8'h44
VRAS0_LB	base + 8'h08	S_YEND	base + 8'h48
VRAS0_UB	base + 8'h0C	S_PITCH	base + 8'h4C
VRAS1_LB	base + 8'h10	S_CEND	base + 8'h50
VRAS1_UB	base + 8'h14	S_INC	base + 8'h54
VRAS2_LB	base + 8'h18	OFFSET	base + 8'h58
VRAS2_UB	base + 8'h1C	D_START	base + 8'h5C
VRAS3_LB	base + 8'h20	D_XEND	base + 8'h60
VRAS3_UB	base + 8'h24	D_YEND	base + 8'h64
BANK0_END	base + 8'h28	D_PITCH	base + 8'h68
Unused	base + 8'h2C	D_CEND	base + 8'h6C
Unused	base + 8'h30	D_INC	base + 8'h70
Unused	base + 8'h34	Foreground	base + 8'h74
Unused	base + 8'h38	Background	base + 8'h78
Unused	base + 8'h3C	BLTer Control	base + 8'h7C

Table 4-8 Mercury Register Addresses

4.3 DAC

The Digital to Analog Converter is the Bt9046. The Bt9046 is a custom RamDac by Brooktree which integrates the pixel unpacking from the VRAM as well as CYMK conversion. The RamDac's data path is physically connected to D0 through D8. Only word and long accesses are allowed as the RamDac has a 9-bit data path For more information on DAC addressing and functions refer to the Bt9046 Data Sheets. Addresses A3 and A2 are connected to Bt9046 pins C1 and C0 respectively.

4.3.1 Sync Sources

"Syncs" will always appear on the GREEN analog output of the Bt9046. A pinout for the Video Output Connector is provided in Figure 4–3.

15-PIN DSUB CONNECTOR	
Pin 1	GROUND
Pin 2	RED
Pin 3	Composite Sync (Active Low)
Pin 4	Monitor ID0
Pin 5	GREEN
Pin 6	GROUND
Pin 7	Monitor ID1
Pin 8	NO CONNECT
Pin 9	BLUE
Pin 10	Monitor ID2
Pin 11	NO CONNECT
Pin 12	Vertical Sync. (Active Low)
Pin 13	GROUND
Pin 14	NO CONNECT
Pin 15	Horizontal Sync. (Active Low)

Figure 4-3 Video Output Connector Pinout

4.4 Flash EPROM

The Mercury 24 uses a Flash EPROM for the configuration ROM. The Size of the ROM is 128k x 8. The Flash EPROM can be written by following a prescribed sequence. For more information on programming see the Intel data sheet 28F010.

5.0 Frame Buffer

The frame buffer is 1024 x 1024 x 32. The data can be displayed using two different addressing mechanisms which we will call SAM Length Independent and SAM Length Dependent. The SAM Length Independent approach uses the concept of midline transfers (transfer cycles during active video) to make the screen size in the X direction independent of the length of the serial port in the VRAM. This allows more (or less) pixels to be displayed on one line of the screen then are in one row of VRAM. The display is then effectively compressed into the upper left corner of the VRAM array using every memory location on every row. The SAM Length Independent method uses the VRAM most efficiently allowing a theoretical resolution of 1024 x 1024 or 1152 x 910, 1280 x 819 etc.

The Mercury 24 frame buffer is designed to be accessible as an Extended Desktop. This means that as the pixel bit depths and screen size decrease there is more memory available which can then be used to create a larger, or "extended" screen. By the use of panning one can then view that area of VRAM which is usually not displayed. As the bit depth is reduced to 4-bit, 2-bit or 1-bit even more VRAM is available for extended desktops. When using the SAM Length Independent approach for generating screen refresh addresses the desktop can be extended in either X or Y-direction or both. When using the SAM Length Dependent approach the desktop can

be extended in the X direction only. The Y-direction is fixed at a maximum of 1024 lines. This is because each row of VRAM is displayed once every horizontal blanking interval (assuming no zoom) and there is no way to extend a line into the next row.

6.0 Expansion

The Mercury 24 gives two connector options for expansion. The NuBus is available on one 60-pin connector with buffered Clock and Reset signals. Two 96-pin connectors create the other expansion giving full access to the entire DRAM array and necessary arbitration circuitry.

Mercury Specification
Preliminary
Revision Level A2

~~RGS40259~~
RGS969S1

THIS DOCUMENT IS THE PROPERTY OF RASTEROPS AND IS CONFIDENTIAL

REPRODUCTION IS PROHIBITED

RasterOps
December 10, 1992
Jeff Tingley

Table of Contents

1.0 Mercury

1.1 Introduction

The Mercury chip is intended to be an acceleration co-processor and data path controller for the RGS14188 (Venus) chip. It takes advantage of a very fast, highly parallel architecture to speed up drawing, using Block Transfers (BLTs). In addition many Boolean, arithmetic and "special" operations can be performed on the data to be BLT. Figure 1-1 shows a functional drawing of the Mercury chip. Mercury is designed to be interfaced to a Dynamic RAM array and a Video RAM array. It contains two banks of internal Static RAM (SRAM) each bank being 320 words deep by 128 bits per word (320 x 128). The SRAM is used for fast temporary storage of data. Mercury is a byte addressable machine.

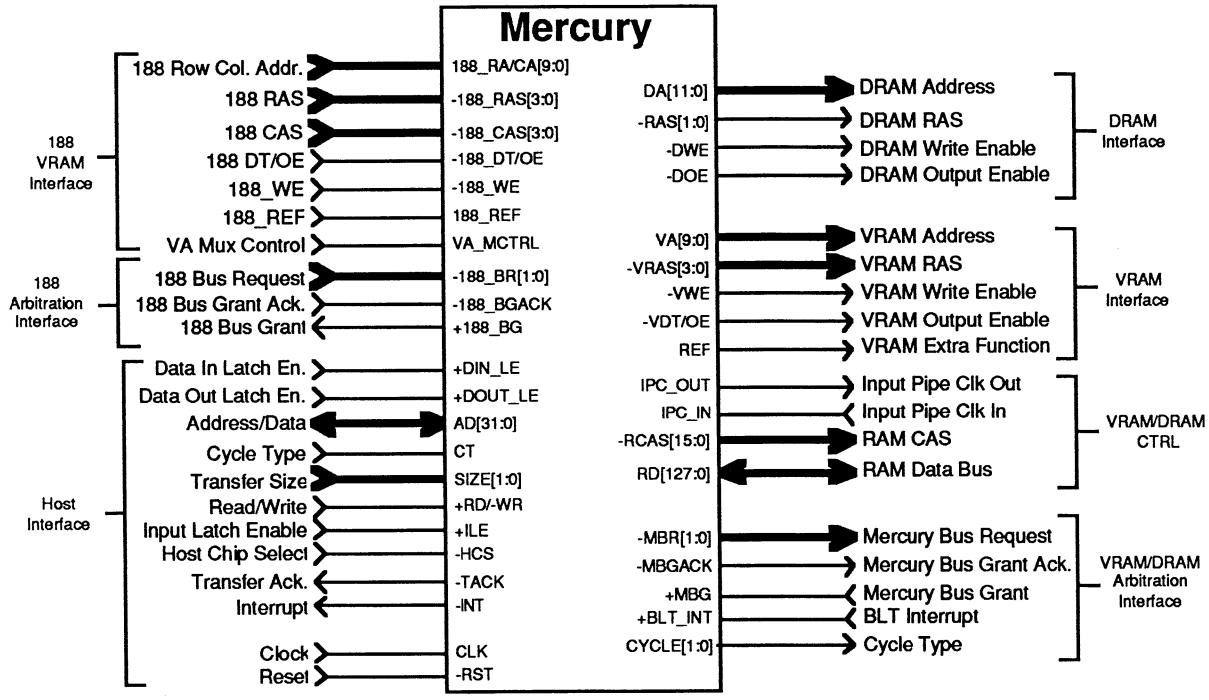


Figure 1-1 Functional Drawing

1.2 Pin Description

VRAM/DRAM CTRL

RD[127:0] - RAM Data Bus[127:0] (TTL Bidirectional, Pulled Up, 4mA Drive)
Mercury DRAM/VRAM data bus. This 16-byte bus is used to transfer data to and from the VRAM and DRAM arrays.

RCAS[15:0] - RAM CAS Bus[15:0] (3-State TTL Outputs, Pulled Up, 16mA Drive,)
These 16 outputs give a byte oriented CAS structure for control of the 16-Byte data bus.

IPC_OUT- Input Pipe Clock Out (TTL Output, 4mA Drive)
This output is used to feed back into the "IPC_IN" input for clocking of the input data pipe. This signal should be delayed from 0 - 6 nS depending on the system load configuration and cas access time.

IPC_IN- Input Pipe Clock In (TTL Input)
This input is used to clock the input pipe. It should be a delayed version of "IPC_OUT".

RGS14188 Arbitration Interface
Note: All these signals wire directly to the RGS14188 chip.

-188_BR[1:0] - 188 Bus Request [1:0] (TTL Inputs, Pulled Up)
These two inputs tell Mercury that the RGS14188 (188) needs to use the VRAM Interface. 188_BR[1:0] are encoded to mean special things, see the RGS14188 users guide for more information.

-188_BGACK - 188 Bus Grant Acknowledge (TTL Input, Pulled Up)
This is the Bus Grant Acknowledge output from the 188 chip. It indicates the 188 is using the VRAM Interface.

+188_BG - 188 Bus Grant (TTL Output, 4mA Drive)
This output hooks directly to the RGS14188 +BG input. When this signal is asserted Mercury is granting the VRAM Interface to the 188 chip.

Host Interface

CLK - Clock (TTL Input)
This is the main clock input for the Mercury Chip. Its maximum frequency is 50 MHz (20ns) .

-RST - Reset (TTL Input)
This input, when asserted, brings Mercury to a known state.

AD[31:0] - Address/Data[31:0] (TTL Bidirectional, Pulled Up, 4mA Drive)
This is the bidirectional, multiplexed address/data bus for the Mercury chip. This bus allows the host to read and write internal registers and access the Dynamic/Video RAM arrays.

SIZE[1:0] - Transfer Size[1:0] (TTL Inputs, SIZE[1] Pulled Down)
These two inputs, along with AD[1:0], are encoded to tell Mercury what the size of the host data transfer will be, byte, word, or long. Table 1-2 shows the encoding. Byte 0 is the byte with lowest address. Word 0 is the word (16-bits) with the lowest address. Long refers to a 32-bit quantity.

SIZE[1]	SIZE[0]	AD[1]	AD[0]	Function
0	0	0	0	Access Byte 0
0	0	0	1	Access Byte 1
0	0	1	0	Access Byte 2
0	0	1	1	Access Byte 3
0	1	0	0	Access Long
0	1	0	1	Access Word 0
0	1	1	0	Access Long
0	1	1	1	Access Word 1
1	0	0	0	Access Word 0
1	0	0	1	Access Word 0
1	0	1	0	Access Word 1
1	0	1	1	Access Word 1

Table 1-2 Transfer Size Encoding

CT- Cycle Type (TTL Input)
This input tells Mercury what type of cycle is to be executed. Table 1-3 shows the encoding.

CT	Function
0	Internal Register Access
1	DRAM Access

Table 1-3 Cycle Type Encoding

+RD/-WR - Read/Write (TTL Input)
This input tells whether a read or write host cycle is to be executed.

+ILE (TTL Input, Pulled Up)
This input is used to latch the address part of "AD[31:0]", "SIZE[1:0]", "CT", and the "+RD/-WR" inputs. The falling edge closes the latches.

-HCS - Host Chip Select (TTL Input)
The falling edge of this input signals the start of a host cycle. The rising edge of this input terminates the cycle.

-TACK - Transfer Acknowledge (Open Drain Output, 4mA Drive)
This output indicates that the requested host cycle has been executed and is ready to be terminated. Note: This output must be pulled up externally

+DIN_LE - Data Input Latch Enable (TTL Input, Pulled Up)
This input is used to latch the input data on the "AD[31:0]" bus if such a function is desired. The falling edge closes the latches.

+DOUT_LE - Data Output Latch Enable (TTL Input, Pulled Up)
This input is used to latch the data output to the "AD[31:0]" bus if such a function is desired. The falling edge closes the latches.

-INT (Open Drain Output, 4mA Drive)
This output can be used for interrupt driven BLTing. When enabled and driven low this output indicates the present BLT is done. Note: This output must be pulled up externally.

DRAM Interface

DA[11:0] - DRAM Address[11:0] (3-State TTL Outputs, Pulled Up, 32mA Drive)
These twelve outputs provide the row and column address for the DRAM array.

-DRAS[1:0] - DRAM RAS (3-State TTL Output, Pulled Up, 32mA Drive)
These outputs are used to latch the row address into the DRAM array.

-DWE - DRAM Write Enable (3-State TTL Output, Pulled Up, 48mA Drive)
This output controls the write function for the DRAM. It is used during write cycles.

-DOE - DRAM Output Enable (3-State TTL Output, Pulled Up, 48mA Drive)
This output controls the output enable function for the DRAM. It is used during read cycles.

VRAM Interface

VA[9:0] - VRAM Address[9:0] (3-State TTL Outputs, Pulled Up, 32mA Drive)
These ten outputs provide the row and column address for the VRAM array.

-VRAS[3:0] - VRAM RAS[3:0] (3-State TTL Outputs, Pulled Up, 32mA Drive)
These outputs are used to latch the row address into the VRAM array. They are intended to be used for bank selecting.

-VWE- VRAM Write Enable (3-State TTL Output, Pulled Up, 32mA Drive)
This output controls the write function for the VRAM. It is used during write cycles.

-VDT/OE- VRAM Transfer/Output Enable (3-State TTL Output, Pulled Up, 32mA Drive)
This output controls the output enable function for the VRAM. It is used during read cycles.

REF- RAM Extra Function (3-State TTL Output, Pulled Up, 32mA Drive)
This output controls the "Split SAM Transfer Mode" for the VRAM. It is used during transfer cycles.

VRAM/DRAM Arbitration Interface

-MBR[1:0] - Mercury Bus Request (TTL Output, 4mA Drive)
When asserted, these outputs indicates to other masters of the VRAM/DRAM Interface that mercury and/or the RGS14188 would like to use the port.

-MBGACK - Mercury Bus Grant Acknowledge (TTL Output, 4mA Drive)
When asserted, this output indicates to other masters of the VRAM/DRAM Interface that mercury is using the port.

+MBG - Mercury Bus Grant (TTL Input, Pulled Up)
When asserted, this input indicates that the VRAM/DRAM Interface is available for use.

CYCLE[1:0] - Cycle Type (TTL Output, 4mA Drive)
These outputs inform and external bus master of the type of cycle that was granted as a result of the external master asserting "+MBG. Table 1-4 shows the encoding.

CYCLE[1]	CYCLE[0]	Cycle Type Granted
0	0	BLT Cycle
0	1	Host DRAM
1	0	Venus Host Cycle
1	1	Transfer/Refresh Cycle

Table 1-4 Cycle Type Encoding

+BLT_INT - BLT Interrupt (TTL Input, Pulled Down)
When asserted, this input indicates that an external bus master would like to interrupt the current BLT so it can use the VRAM/DRAM interface. This input is only useful when BLTs are in progress.

188 VRAM Interface
Note: All these signals except VA_MCTRL wire directly to the RGS14188 Chip

188_RA/CA[9:0] - RGS14188 Row/Column Address[9:0] (TTL Input, Pulled Up)
These inputs are used in conjunction with the four 188 CAS inputs to map the four RGS14188 CAS structure to the 16 CAS structure of the Mercury design and create the VRAM Address Outputs for RGS14188 accesses to the VRAM.

-188_RAS[3:0] - RGS14188 Row Address Strobe[3:0] (TTL Input, Pulled Up)
These four inputs are used to latch the VRAM Row Address during RGS14188 accesses. They also help detect transfer cycles and refresh cycles for control of the VRAM and DRAM arrays.

-188_CAS[3:0] - RGS14188 Column Address Strobe[3:0] (TTL Input, Pulled Up)
These four inputs are used to latch the VRAM Column Address during RGS14188 accesses. Furthermore, they are used conjunction with the "188_RA/CA[1:0]" inputs, to map the RGS14188's four CAS structure to the sixteen CAS structure of the Mercury.

-188_DT/OE - RGS14188 Data Transfer/Output Enable (TTL Input, Pulled Up)
This input is used to indicate read and transfer cycles to the VRAM array.

-188_WE - RGS14188 Write Enable (TTL Input, Pulled Up)
This input is used to indicate write cycles to the VRAM array.

188_REF - RGS14188 RAM Extra Function (TTL Input, Pulled Up)
This input is used to control the "Split SAM" feature of the VRAM during Transfer Cycles.

VA_MCTRL - VRAM Address Mux Control (TTL Input, Pulled Up)
This input is used to control the shifting multiplexer that maps the "188_RA/CA" inputs to the "VA" outputs. It should be connected to a delayed version of the "-188_RAS" signals.

2.0 Mercury Functionality

2.1 Introduction

A block diagram of the Mercury chip is shown in Figure 2-1. Mercury has several modes of acceleration and can manipulate linear, or 2-dimensional pixel arrays. It is also capable of doing pixel operations on four 32-bit pixels, eight 16-bit pixels, or sixteen 8-bit pixels at one time. In addition to all of its acceleration capabilities, Mercury acts as a data path controller for the RGS14188 chip and as a host interface for a large dynamic RAM array. It will automatically route a standard 32-bit Big Endian or Little Endian data bus to its high speed parallel bus for RAM read and write operations.

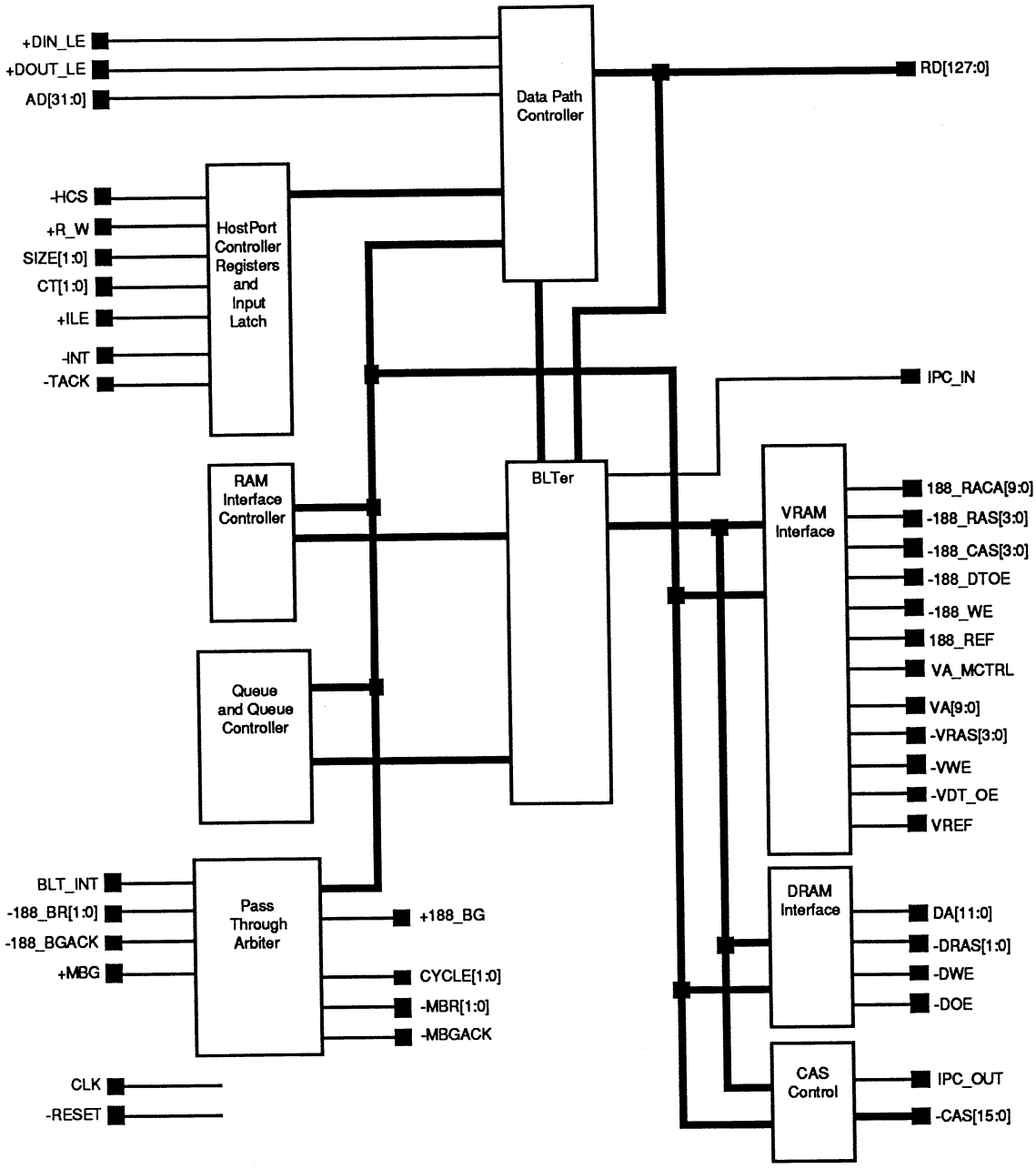


Figure 2-1 Mercury Block Diagram

A block diagram of a system designed around the mercury chip is shown in Figure 2-2.

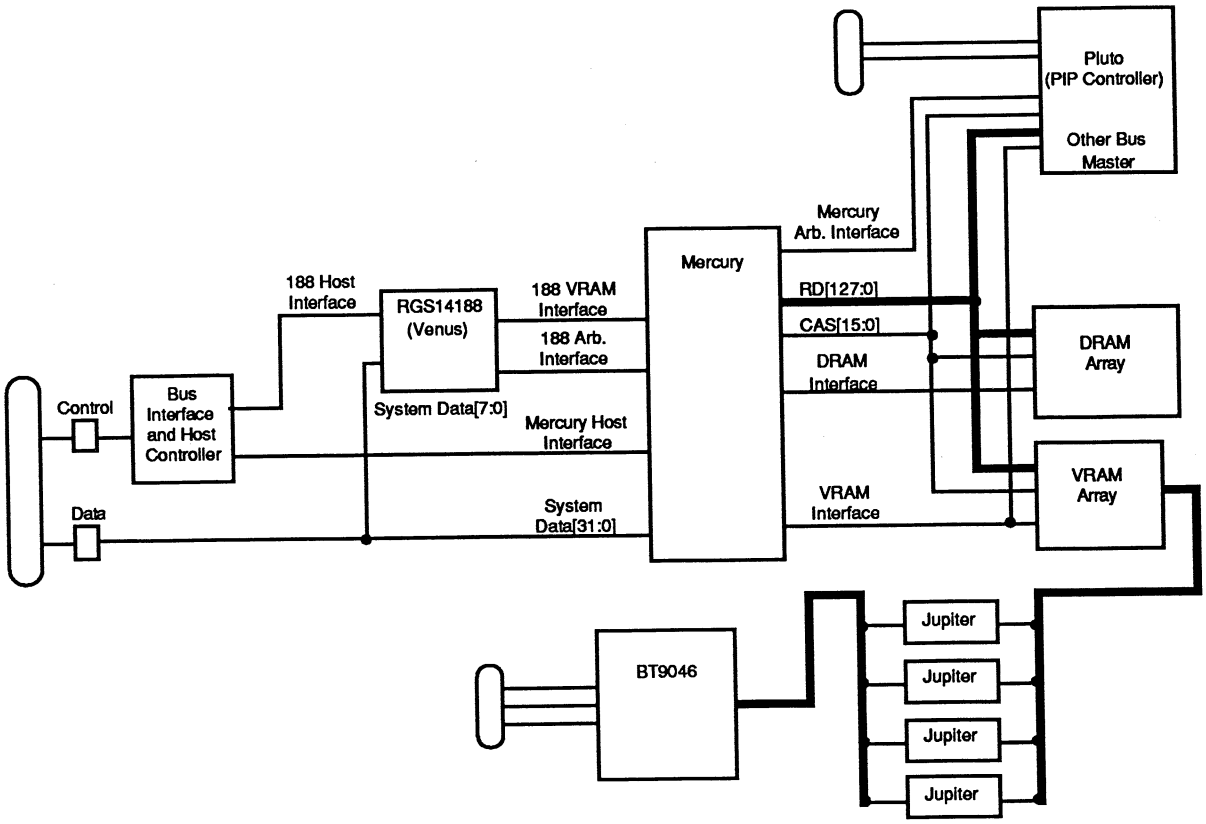


Figure 2-2 Mercury Based System

All acceleration is based on "Source/Destination BLTs" (Block Transfers). A Source/Destination BLT, using very high speed accesses, reads the data at the source (DRAM, VRAM or Internal Register) and the destination, operates on it, and returns the new data to the destination. The internal Static RAM is used to store these blocks of data when necessary. Be sure to keep in mind that Mercury is a byte oriented machine.

2.2 BLTer

The BLTer is the heart of Mercury's accelerator function; reading, combining and writing data at very high speeds on the 16-byte data bus. A block diagram of the BLTer is shown in Figure 2-3. Two of the main components of the BLTer are the SRAM and ROU. The SRAM consists of two banks of 320 - 128-bit words. This gives a total SRAM size of 81.92 KBits or 10.24 KBytes with each bank having 5.12 KBytes. Each bank of SRAM has the capacity to hold one 1280 pixel line at 32-bits per pixel. The SRAM is used only by the Mercury Chip for storing pixel data. The host port has no access to the SRAM.

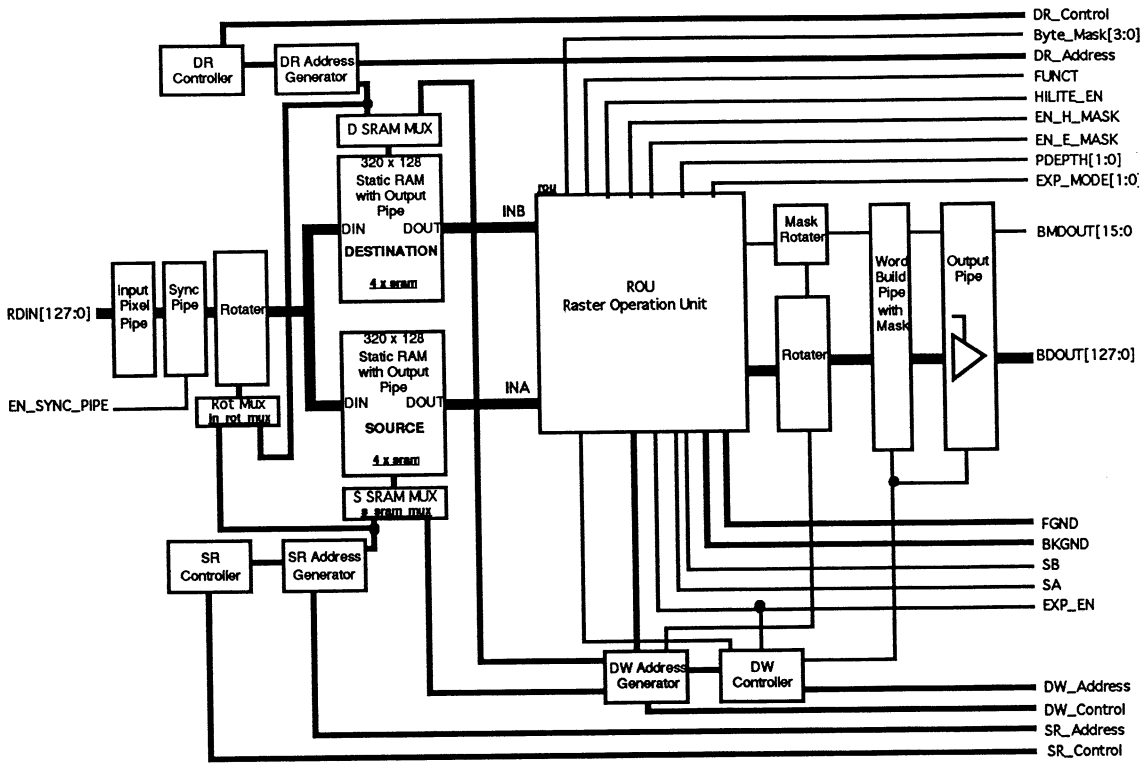


Figure 2-3 BLTer Block Diagram

The BLTer reads and writes data to and from the external Dynamic RAM (DRAM) and/or Video RAM (VRAM) arrays. The accesses to these RAM arrays is designed to be as fast as possible through the use of "page mode cycles". These page mode cycles can be of varying length depending on the amount of data to be operated on. When the BLTer is told to start it reads the source pixel array, if one has been defined, and stores the first line of it in the Source SRAM. It then reads the destination pixel array, if one has been defined, and stores the first line of it in the Destination SRAM. The data is then combined by the ROU and written back to the first line of the destination pixel array. This process continues until all the data has been operated on.

The Mercury chip can execute four different types of BLTs on source and destination data; Straight BLTs, Pattern BLTs, Color Expanding BLTs and Hilighting BLTs.

2.2.1 Straight BLTs

Straight BLTs take the data from the VRAM, DRAM, Foreground Register or Background Registers, combine it based on the ALU mode selected, then return the combined data to the destination. Straight BLTs can be done both left to right and right to left in the horizontal direction and both top to bottom and bottom to top in the vertical direction. See section 2.3 for more information on defining pixel arrays.

Some examples of Straight BLTs are:

A block of data in DRAM could be transferred to the same size block in VRAM.

The Foreground Register could be transferred to a block in VRAM

A block of data in DRAM could be logically combined with a block of data in VRAM and put back in the same location of VRAM.

The Background Register could be logically combined with a block of data in VRAM and put back in the same location of VRAM.

2.2.1 Pattern BLTS

Pattern BLTs take a n x m pattern stored as a rectangular source and copy that to the destination, repeating until the destination write is complete. The value for n, which is the number of bytes in the pattern in the X direction, must be divisible by 16 (16, 32, ...1152, Etc.). There are no restrictions on m, which is the number of lines in the pattern. All ROU operations can be performed on the pattern being BLT. Pattern BLTs can be done in any direction both horizontally and vertically. If patterns are not being used the "OFFSET" value does not need to be set to zero. **One important note; the starting address for a pattern must always be on a 16 byte boundary.**

Some examples of Pattern BLTs are:

A 32-byte x 4 line pattern in DRAM could be transferred to the entire visible screen in VRAM.

A 64-byte x 64 line pattern in DRAM could be logically combined with the entire visible screen in VRAM and transferred back to the entire visible screen in VRAM.

A four pixel (32-bits per pixel) x 100 line pattern in VRAM could be inverted and transferred to the entire VRAM.

2.2.3 Color Expanding BLT

Color Expanding BLTs store the shape of an object in a rectangular source array of 1-bit pixels. Each shape must always start on a byte address boundary but can have arbitrary length in the x-direction. The color information is obtained from the Foreground and/or Background Registers. Modes can be selected to determine if transparency is used (in other words, not writing specific data) and what each bit in the source array means. Mode information is given in Table 2-1.

Some examples of Color Expanding BLTs are:

A 32-bit x 4 line object in DRAM could be expanded to the color of the Foreground register and transferred to VRAM.

A 9-bit x 9 line object in DRAM could be expanded to the color of the Foreground Register, logically combined with the same size (expanded) block of data in VRAM and put back in the VRAM at the same location.

2.2.4 Hilighting BLTs

Hilighting BLTs are a special form of Standard BLT that takes the data at the destination and compares it to the Foreground Register (in this case the Hilight Color) and Background Register to decide what the resulting data should be. If the input data is equal to the Foreground Register the Hilighter passes the Background Register to the ALU. If the input data is equal to the Background Register the Hilighter passes the Foreground Register to the ALU. If the input data does not equal either the Foreground or Background Registers the data remains unchanged and is passed to the ALU. Hilighting can happen in parallel with color expanding if this operation is desired.

2.2.5 ROU (Raster Operation Unit)

The ROU (Raster Operation Unit) performs several functions for the BLTer. It color expands monochrome bit maps, hilights destination pixel arrays and logically/arithmetically combines data. A block diagram of the ROU is shown in figure 2-4. During processing the ROU will get data from the three sources, the internal SRAM, the Foreground Register, and the Background Register depending on the function and type of BLT selected.

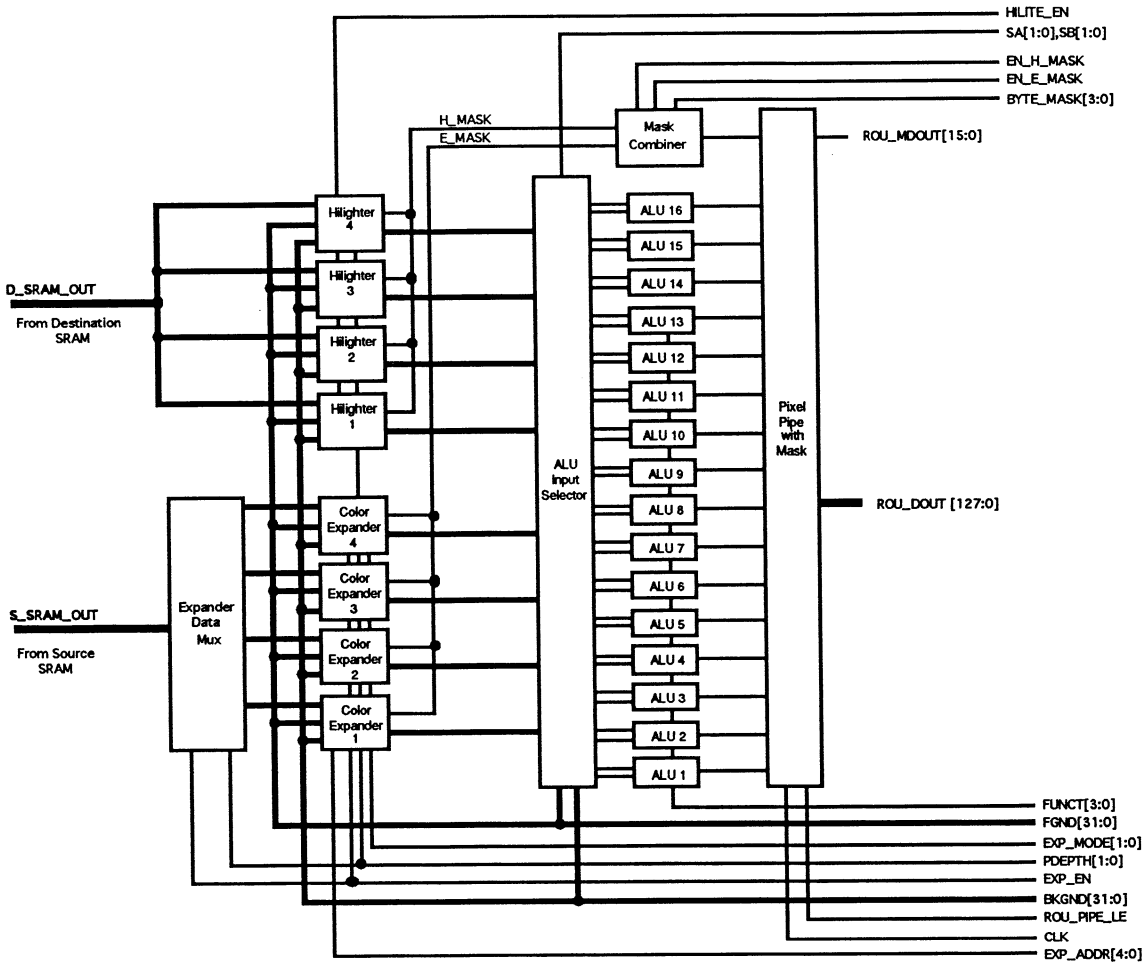


Figure 2-4 ROU Block Diagram

Color Expander

The ROU can treat the input data from the Source SRAM as a monochrome bit map expanding each bit to the color defined in the Foreground or Background Registers and the width defined by the selected pixel depth. Table 2-1 show how the input bits are treated based on the expansion mode selected.

Mode 0	1 = Foreground	0 = Background & E_MASK = 0
Mode 1	1 = Foreground	0 = Background & E_MASK = 1
Mode 2	1 = Background	0 = Foreground & E_MASK = 0
Mode 3	1 = Background	0 = Foreground & E_MASK = 1

Table 2-1 Expansion Modes.

When the mask function is enabled a pixel with the E_MASK bit set to a one will not be written.

If Color Expansion is not enabled the data from the SRAM is passed directly to the Color Expander Outputs without change.

Hilighter

Hilighting is a process where the data in the Destination SRAM is compared to the Foreground and Background Registers. If the SRAM data is exactly equal to the Foreground Register value, the Background Register value is substituted for the data before it is passed to the ALU. If the SRAM data is exactly equal to the Background Register value, the Foreground Register value is substituted for the data before it is passed to the ALU. If the SRAM data does not equal either Foreground or Background Register, the data is unchanged and the H_MASK bit is set for that pixel. When the mask function is enabled a pixel with the H_MASK bit set to a one will not be written.

If Hilighting is not enabled the data from the SRAM is passed directly to the Hilighter Outputs without change.

Mask Combiner

The Mask Combiner is used to route or logically combine the mask data obtained from the Color Expander and Hilighter. The output can be modally selected as 0, the H_MASK value from the Hilighter, the E_MASK value from the Color Expander, or the logical OR of the H_MASK and E_MASK values.

ALU (Arithmetic Logic Unit)

The ALU performs arithmetic and logical operation on the data passed to it depending on the function selected. Each input to the ALU can take data from four different sources. Table 2-2 and 2-3 show the possible data inputs for the "A" and "B" sides of the ALU.

Note: The operation is performed on 8, 16, or 32 bit quantities depending on the pixel depth selected except addition and subtraction which is performed on a component basis in 32-bit mode and a pixel basis in 8-bit mode. 16-bit additions and subtractions may not be useful because of the organization of the red green and blue components within the two bytes.

Mode 0	ina = Output from Color Expander
Mode 1	ina = Foreground Register
Mode 2	ina = Background Register
Mode 3	ina = Output from the Hilighter

Table 2-2 "ina" source for the ALU

Mode 0	inb = Output from Hilighter
Mode 1	inb = Foreground Register
Mode 2	inb = Background Register
Mode 3	inb = Output from the Color Expander

Table 2-3 "inb" source for the ALU

Table 2-4 shows all the possible operation the ALU can perform.

ina -> out
NOT(ina) -> out
ina AND inb -> out
NOT(ina) AND inb -> out
ina NAND inb -> out = NOT(ina) OR NOT(inb) -> out
ina OR inb -> out
NOT(ina) OR inb -> out
ina NOR inb -> out = NOT(ina) AND NOT(inb) -> out
ina XOR inb -> out
ina XNOR inb -> out
MAX(ina;inb)
MIN(ina;inb)
ina + inb -> out (adder wrap mode)
ina - inb -> out (subtractor wrap mode)
ina + inb -> out (adder saturate mode)
ina - inb -> out (subtractor saturate mode)

Table 2-4 ALU Operations

Note: ALU operations are applied after color expansion and hilighting.

2.2.6 Pixel Depth Support

The Mercury chip is capable of operating on 8, 16, or 32 bit pixels. 8-bit mode is very straight forward, all operations occur on bytes. In 16-bit mode all pixels are treated as two bytes, no attention is given to the number of bits in the red, green, blue and alpha components. In 32-bit mode all operations are performed on a component basis (8-bit). A 4-bit byte mask is provided to disable writing of any or all bytes within a 32-bit word. These four bits are located in the BLTer Control Register. One possible use for the 4-bit byte mask would be to disable the "alpha" byte within a 32-bit word so the 24-bits of pixel data could be manipulated without changing the "alpha" channel information.

2.3 Defining Pixel Arrays

Pixel arrays can be defined in four different ways depending on the horizontal and vertical directions desired for the BLT. Figure 2-5 through 2-9 show how pixel arrays are defined for both standard BLTs and Pattern BLTs. The corresponding source or destination registers are loaded with the desired values to define the BLT.

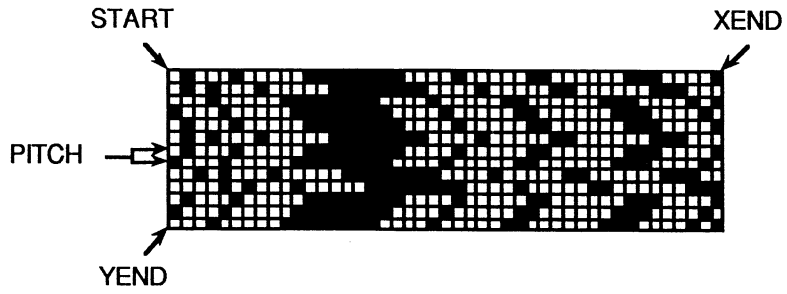


Figure 2-5 Left to Right, Top to Bottom

If the desired BLT direction is from right to left as shown below the 2's complement of the increment value is loaded into the Increment Register.

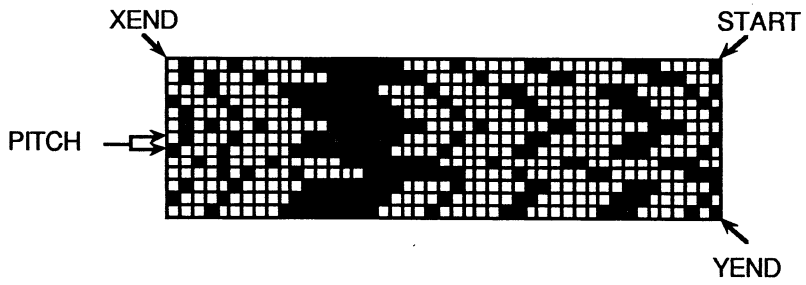


Figure 2-6 Right to Left, Top to Bottom

If the desired BLT direction is from bottom to top as shown below the 2's complement of the pitch is loaded into the Pitch Register.

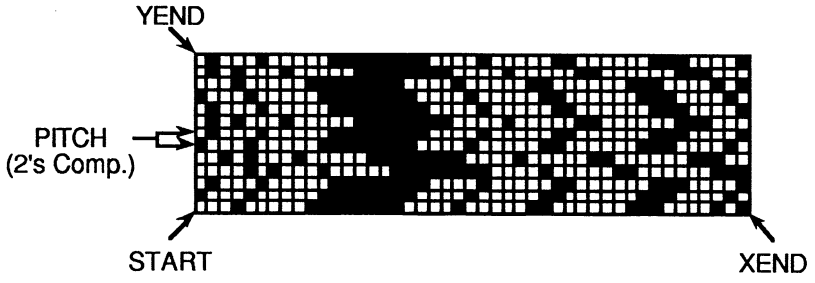


Figure 2-7 Left to Right, Bottom to Top

If the desired BLT direction is from right to left and bottom to top as shown below the 2's complement of the increment value is loaded into the Increment Register and the 2's complement of the pitch is loaded into the Pitch Register.

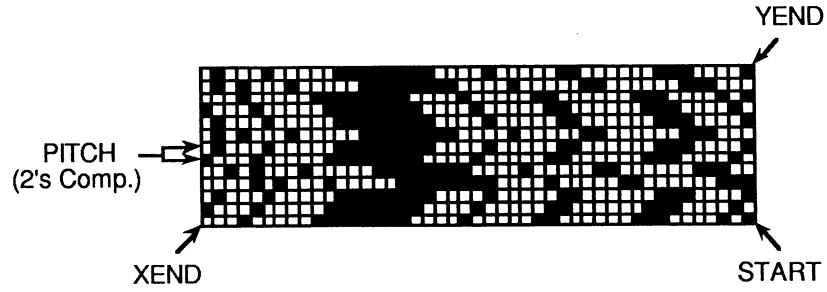


Figure 2-8 Right to Left, Bottom to Top

Patterns can also be defined for any direction horizontally and vertically as described above. In addition Pattern BLTs use an offset value which indicates where to start within the pattern. The offset value must be between the S_START and S_XEND value (S_START ≤ OFFSET ≤ S_XEND). If an offset in the Y direction is desired, the pattern must be duplicated in the Y direction and the S_START, S_XEND, and S_YEND would be changed to get the desired Y offset.

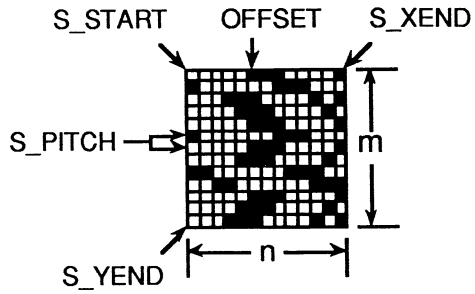


Figure 2-9 Pattern Source Array Definition

2.4 Host Interface

The Host Interface (see Figure 1-1) is used to communicate register data for configuration and control of the mercury chip. It is also used as data path and control for access to the external DRAM and VRAM arrays. The Host Interface uses a multiplexed, 32-bit address/data bus with the address part being latched on the falling edge of "+ILE". The Host Chip Select signal is used to start and terminate cycles and the Transfer acknowledge indicates the status of a current cycle.

2.5 RGS14188 Arbitration Interface

The RGS14188 Arbitration Interface (see Figure 1-1) is used to connect Mercury to the arbitration mechanism of an RGS14188 chip. It uses two levels of bus requesting and is

intended to be wired directly to the RGS14188. All other bus masters who would like to use either the VRAM or DRAM interface busses must request them from the Mercury VRAM/DRAM Arbitration Interface.

2.6 RGS14188 VRAM Interface

The RGS14188 VRAM Interface (see Figure 1-1) is used to connect Mercury to the RAS/CAS and address structure of an RGS14188 chip. Since mercury uses a very wide data bus the Address and CAS signals from the RGS14188 must be remapped to fit into Mercury's bus structure. This interface is connected directly to the RGS14188 chip.

2.7 VRAM/DRAM Arbitration Interface

The VRAM/DRAM Arbitration Interface is used to allow external devices such as PIP controller to access the VRAM or DRAM arrays. If arbitration is enabled Mercury will request the VRAM or DRAM buses from an external controller before each access. Mercury will also pass through all the Requests from the RGS14188 chip so it's Transfer/Refresh and host cycles can be completed. The arbitration is performed as follows: Mercury will request the bus by asserting "-MBR[1:0]". Table 2-5 shows the possible combinations for bus requests.

-MBR[1]	-MBR[0]	Request
1	1	No request
1	0	Host Cycle Pending
0	1	Transfer/Refresh Pending
0	0	Both Host and Transfer/Refresh Pending

Table 2-5 Bus Request Types.

At the "proper" time the external controller asserts the "+MBG" input to indicate the bus is now available. Mercury will then drive "-MBGACK" indicating it is using the bus. The signal "+BLT_INT" can be used to interrupt a BLT if one is in progress. "+BLT_INT" can be asserted anytime during the entire granted cycle, and if a BLT is in progress Mercury will relinquish the bus. It will then generate an immediate request for the bus so it can finish its BLT. If "+BLT_INT" is asserted when any other cycle is in progress (including no cycle) it will be ignored. Any bus request with "-MBR[1]" asserted must be serviced as soon as possible (immediately). A bus request with only "-MBR[0]" asserted can be delayed as long as desired with the only effect being reduced acceleration. If both bus requests are asserted Mercury will service the Transfer/Refresh cycle first, then rearbitrate for the other pending cycles. The external bus master can determine what kind of cycle was executed after relinquishing (driving "+MBG" active) the bus by looking at the "CYCLE" outputs.

3.0 Internal Registers

Mercury has several internal registers all of which are a maximum of 32-bits wide. Each register has read/write ability and must be accessed as a long (32-bit) quantity.

AD[31:0] will contain the data to be written or read. The following sections describe the registers in detail. All registers related to BLTing (including the BLT Control Register) are dual ranked. This allows one set of registers to be loaded while the current BLT is executing, maximizing system performance. To take advantage of this function the host must poll the Status Register before loading any BLT related registers to make sure the previous values have been transferred to the BLTer. As an alternative to polling, interrupt driven BLTing can be accomplished by loading the desired BLT and waiting for the "-INT" signal to be asserted. Interrupts must be enabled in the System Control Register for this function to work.

S_Start[28:0] - Source Starting Address[28:0] (Dual Ranked)

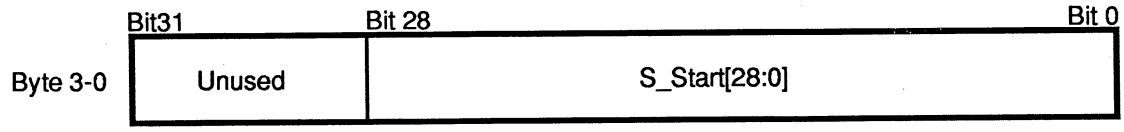


Figure 3-1 Source Start Address

Bit 28-0, Source Start Address
The byte address of the first pixel in the source array (see Figure 2-5).

S_XEnd28:0] - Source X-End Address[28:0] (Dual Ranked)

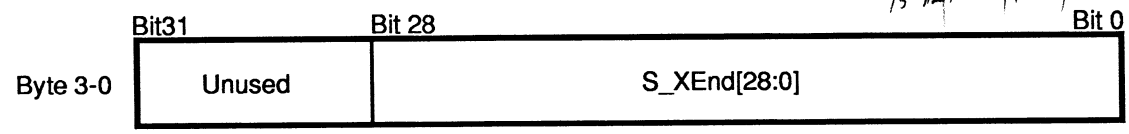


Figure 3-2 Source XEnd Address

Bit 28-0 Source XEnd Address
The byte address of the last pixel in the x-direction of the source array.

S_YEnd[28:0] - Source Y-End Address[28:0] (Dual Ranked)

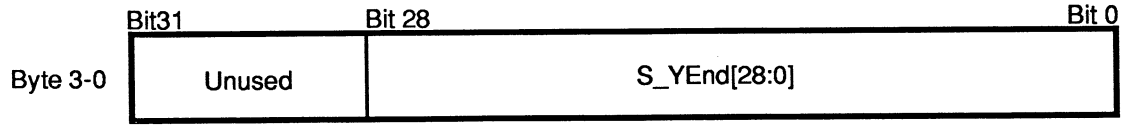


Figure 3-3 Source YEnd Address

Bit 28-0, Source YEnd Address
The byte address of the first pixel on the last line in the y-direction of the source array.

S_Pitch[28:0] - Source Pixel Array Pitch[28:0] (Dual Ranked)

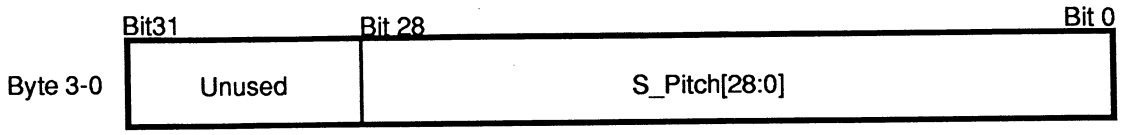


Figure 3-4 Source Pitch

Bit 28-0, Source Pitch
The Source Pitch is the number of bytes between vertically adjacent pixels for the specified source array.

Offset[28:0] - Source Pixel Array Pattern Offset[28:0] (Dual Ranked)

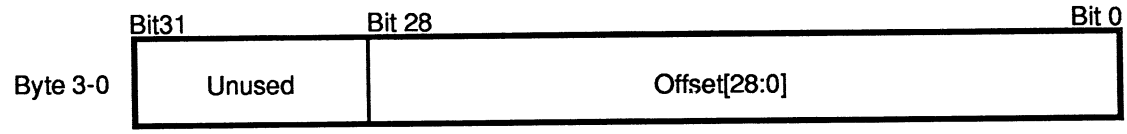


Figure 3-5 Pattern Offset

Bit 28-0, Source Pixel Array Pattern Offset
The Offset address, used only during pattern BLTs, specifies the byte address of an offset to start at within a pattern. See Figure 2-8.

S_CEND[28:0] - Source Column End Pointer[28:0] (Dual Ranked)

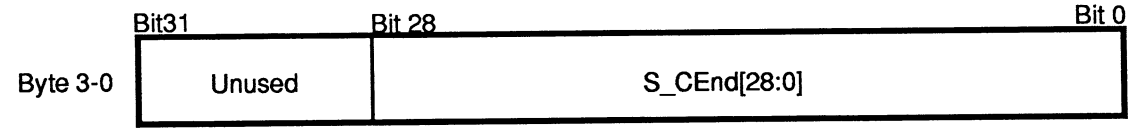


Figure 3-6 Source Column End Pointer

Bit 28-0, Source Column End
A mask value used to tell Mercury where the column address ends for the VRAMs or DRAMs of the pixel array being defined. Since there are 16 Bytes per address this value must be shifted 4 bits to the left. For example: 256K x 4 RAMs have nine column addresses so this value should be set to 'h0001FF0.

S_INC[28:0] - Source Increment[28:0] (Dual Ranked)

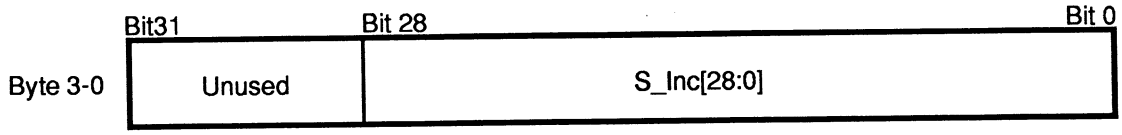


Figure 3-7 Source Increment Value

Bit 28-0, Source Increment Value
This register specifies the amount to increment the CAS address each time a CAS is asserted during page mode cycles. If BLTing from right to left this register must be loaded with the 2's complement of the actual increment value

D_Start[28:0] - Destination Starting Address[28:0] (Dual Ranked)

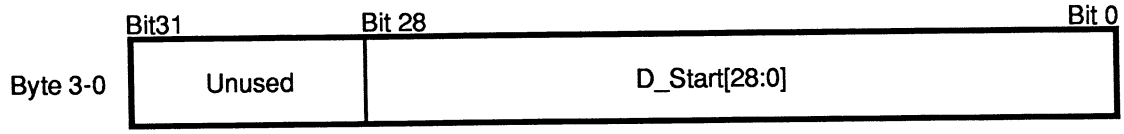


Figure 3-8 Destination Start Address

Bit 28-0, Destination Start Address
Designates the byte address of the first pixel in the destination array.

D_XEnd[28:0] - Destination X-End Address[28:0] (Dual Ranked)

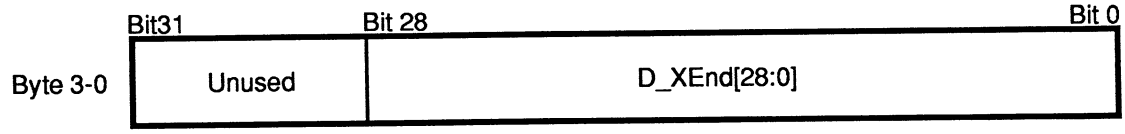


Figure 3-9 Destination X-End Address

Bit 28-0, Destination XEnd Address
The byte address of the last pixel in the x-direction of the destination array.

D_YEnd[28:0] - Destination Y-End Address[28:0] (Dual Ranked)

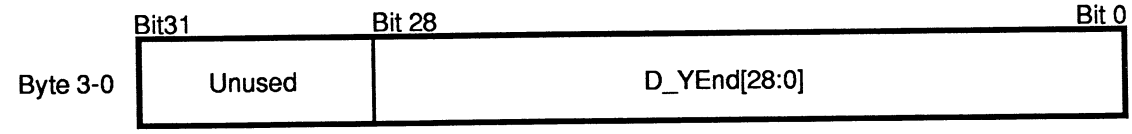


Figure 3-10 Destination Y-End Address

Bit 28-0, Destination YEnd Address
The byte address of the first pixel on the last line in the y-direction of the destination array.

D_Pitch[28:0] - Destination Pixel Array Pitch[28:0] (Dual Ranked)

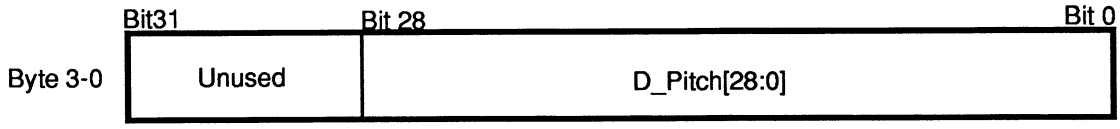


Figure 3-11 Destination Pitch

Bit 28-0, Destination Pitch
The number of bytes between vertically adjacent pixels in the destination array.

D_CEnd[28:0] - Destination Column End Pointer[28:0] (Dual Ranked)

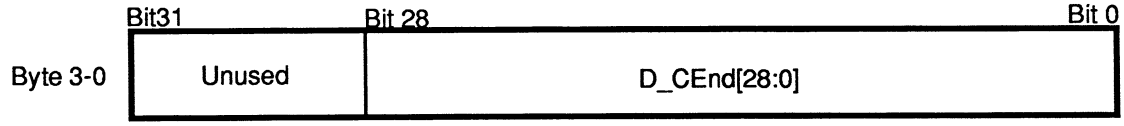


Figure 3-12 Destination Column End

Bit 28-0, Destination Column End
A mask value used to tell Mercury where the column address ends for the VRAMs or DRAMs of the pixel array being defined. Since there are 16 Bytes per address this value must be shifted 4 bits to the left. For example; 128K x 8 RAMs have eight column addresses so this value should be set to 'h0000FF0.

D_INC[28:0] - Destination Increment[28:0] (Dual Ranked)

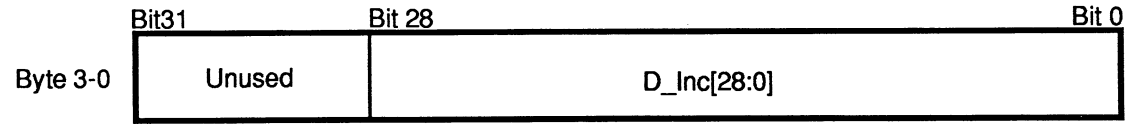


Figure 3-13 Destination Increment Value

Bit 28-0, Destination Increment Value
This register specifies the amount to increment the CAS address each time a CAS is asserted during page mode cycles. If BLTing from right to left this register must be loaded with the 2's complement of the actual increment Value

BLTer Control Register (Dual Ranked)

This register has two functions. It specifies all the control information Mercury needs to perform a specified BLT and it tells the BLTer that all the other necessary registers are loaded and the BLT is ready to start (A write to this register signals "BLT Go").

	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
Byte 0	+Hilight Enable	Expand Mode[1]	Expand Mode[0]	+Expand Enable	Dest. Select	+Use Dest.	Source Select	+Use Source
	Bit 15	Bit 14	Bit 13	Bit 12	Bit 11	Bit 10	Bit 9	Bit 8
Byte 1	Funct [3]	Funct [2]	Funct [1]	Funct [0]	ALU SB[1]	ALU SB[0]	ALU SA[1]	ALU SAI[0]
	Bit 23	Bit 22	Bit 21	Bit 20	Bit 19	Bit 18	Bit 17	Bit16
Byte 2	+Byte Mask[3]	+Byte Mask[2]	+Byte Mask[1]	+Byte Mask[0]	PDepth[1]	PDepth[0]	+EN H_Mask	+EN E_Mask
	Bit 31	Bit 30	Bit 29	Bit 28	Bit 27	Bit 26	Bit 25	Bit24
Byte 3	Unused	Unused	Unused	Unused	Unused	Unused	Unused	+Pattern Mode

Figure 3-14 BLTer Control Register

Bit 0, + Use Source
When set to a one, the BLTer reads the source array specified in the source registers. When set to a zero nothing is read for the source. (The source could be the Foreground Register for example)

Bit 1, Source Select
When told to Use Source (Bit 0) this bits select the RAM array for the BLTer to read. Table 3-1 shows the encoding.

Source Select	Selected Input for Source
0	DRAM
1	VRAM

Table 3-1 Source Select Encoding

Bit 2, +Use Destination
When set to a one, the BLTer reads the destination array specified in the destination registers. When set to a zero nothing is read for the destination. (The destination could be the Background Register for example)

Bit 3, Destination Select
This bits selects the destination RAM array for the BLTer to write and possibly read depending on (Bit 2). Table 3-2 shows the encoding.

Dest. Select	Selected Input for Destination
0	DRAM
1	VRAM

Table 3-2 Destination Select Encoding

Bit 4, +Expand Enable
When set to a one Mercury will treat the data in the Source SRAM as a monochrome bit map and expand it based on pixel depth and expand mode.

Bit 6-5, Expand Mode[1:0]
These bits specify the expand mode. Table 3-3 shows the encoding.

Expand Mode[1]	Expand Mode [0]	Function
0	0	1=Foreground:0=Background & E_MASK = 0
0	1	1=Foreground:0=Background & E_MASK = 1
1	0	1=Background:0=Foreground & E_MASK = 0
1	1	1=Background:0=Foreground & E_MASK = 1

Table 3-3 Expand Mode Encoding

Bit 7, Hilight Enable
When set to a one data in the destination SRAM will be hilighted based on the Foreground and Background Register values.

Bit 9-8, ALU SA[1:0] - ALU Select A[1:0]
These bits select the input for the "ina" side of the ALU as defined in Table 3-4.

ALU ASel[1]	ALU ASel[0]	Selected Input
0	0	Color Expander Out
0	1	Foreground Color
1	0	Background Color
1	1	Hilighter Output

Table 3-4 ALU A Side Selects

Bit 11-10, ALU SB[1:0] - ALU Select B[1:0]
These bits select the input for the "inb" side of the ALU as defined in Table 3-5

ALU ASel[1]	ALU ASel[0]	Selected Input
0	0	Hilighter Output
0	1	Foreground Color
1	0	Background Color
1	1	Color Expander Out

Table 3-5 ALU B Side Selects

Bit 15-12, Funct[3:0] - ALU Function Select[3:0]
The ALU Function Select bits apply a certain Boolean or arithmetic function on the selected "ina" and "inb" data. The operation is performed on 8, 16, 24, or 32 bit quantities depending on the pixel depth selected. Table 3-6 lists the possible operations.

Funct [3]	Funct [2]	Funct [1]	Funct [0]	ALU Operation
0	0	0	0	ina -> out
0	0	0	1	NOT(ina) -> out
0	0	1	0	ina AND inb -> out
0	0	1	1	NOT(ina) AND inb -> out
0	1	0	0	ina NAND inb -> out
0	1	0	1	ina OR inb -> out
0	1	1	0	NOT(ina) OR inb -> out
0	1	1	1	ina NOR inb -> out
1	0	0	0	ina XOR inb -> out
1	0	0	1	ina XNOR inb -> out
1	0	1	0	MAX(ina;inb)
1	0	1	1	MIN(ina;inb)
1	1	0	0	ina + inb -> out (adder wrap mode)
1	1	0	1	ina - inb -> out (subtractor wrap mode)
1	1	1	0	ina + inb -> out (adder saturate mode)
1	1	1	1	ina - inb -> out (subtractor saturate mode)

Table 3-6 ALU Operation Select

All logical operations are performed on a bit by bit basis independent of pixel depth. Min, Max, Addition and Subtraction operations are performed on a component basis in 24-bit and 32-bit modes. This means that the 8-bits of Red are added separately, as with green and blue. Overflow is handled either by wrapping (overflow mode) in each component or saturating (saturate mode) each component. The Min and Max functions replace the component that is either min or max depending on the function selected. Min, Max, Addition and Subtraction in 16-bit mode may not give useful results because the components of Red, Green, and Blue are 5-bits (sometimes 6-bits depending on the system) and Mercury is a byte oriented machine.

Bit 16, +EN E_Mask - +Enable Expander Mask
When set to a one the mask values output by the expander controls whether a pixel is written or not.

Bit 17, +EN H_Mask - +Enable Hilighter Mask
When set to a one the mask values output by the hilighter controls whether a pixel is written or not.

Note: when both +EN E_Mask and +EN H_Mask are set to a one, the mask values output by the Expander and Hilighter are logically ORed to produce the final mask value.

Bit 19-18, Pixel Depth[1:0]
Specifies either 8, 16, or 32 bits per as defined in Table 3-7.

PDepth[1]	Pdepth[0]	Pixel Depth
0	0	8-bits per Pixel
0	1	16-bits per Pixel
1	0	32-bits per Pixel
1	1	32-bits per Pixel

Table 3-7 Pixel Depth Control

Bit 23-20, +Byte Mask[3:0]
When set to a one, these bits disable writing of specific bytes within a 32-bit word. Table 3-7 shows how these bits are encoded. These bits only affect BLTs.

+Byte Mask[0] = 1	Byte Addresses Disabled are 0, 4, 8, C
+Byte Mask[1] = 1	Byte Addresses Disabled are 1, 5, 9, D
+Byte Mask[2] = 1	Byte Addresses Disabled are 2, 6, A, E
+Byte Mask[3] = 1	Byte Addresses Disabled are 3, 7, B, F

Figure 3-15 Byte Mask Function

Bit 24, +Pattern Mode
When set to a one the specified source array is treated as a pattern and expanded to the destination array.

Bit 31-25, Unused
These bits do nothing.

FGND[31:0] - Foreground Color Register[31:0] (Dual Ranked)
This registers is used to specify the foreground color for color expanding BLTs and other operations.

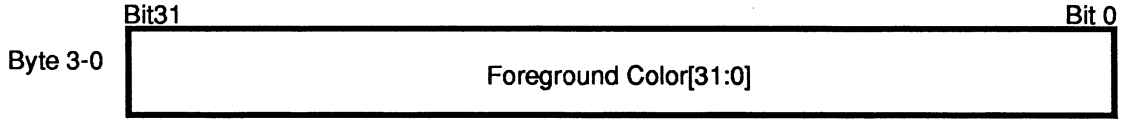


Figure 3-16 Foreground Color Register

8-bit Mode
Bits[7:0] specify the foreground color for expansion and hilighting.
Bits[31:0] specify the foreground color for standard BLTing.

16-bit Mode
Bits[15:0] specify the foreground color for expansion and hilighting.
Bits[31:0] specify the foreground color for standard BLTing.

32-bit Mode
Bits[31:0] specify the foreground color for expansion, hilighting and standard BLTing.

BKGND[31:0] - Background Color Register[31:0] (Dual Ranked)
This register is used to specify the background color for color expanding BLTs and other operations.

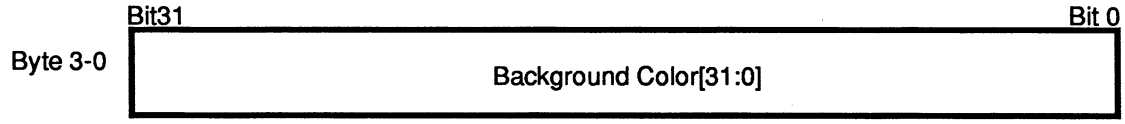


Figure 3-17 Background Color Register

8-bit Mode
Bits[7:0] specify the background color for expansion and hilighting.
Bits[31:0] specify the background color for standard BLTing.

16-bit Mode
Bits[15:0] specify the background color for expansion and hilighting.
Bits[31:0] specify the background color for standard BLTing.

32-bit Mode
Bits[31:0] specify the background color for expansion, hilighting and standard BLTing.

System Control Register

	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
Byte 0	+Enable NAND	-Reset Que	VRAM Size[1]	VRAM Size[0]	+Ext. Arb. On	+Swp AD Bytes	+Interrupt Enable	Unused

	Bit 15	Bit 14	Bit 13	Bit 12	Bit 11	Bit 10	Bit 9	Bit 8
Byte 1	Unused	Unused	Unused	Unused	Unused	TACK DLY[2]	TACK DLY[1]	TACK DLY[0]

Figure 3-18 System Control Register

Bit 0, Unused
Set this bit to do nothing.

Bit 1, +Interrupt Enable
Set this bit to a one to enable interrupts on BLT done. Useful for interrupt driven BLTing.

Bit 2, +Swp AD Bytes - Swap Address/Data Bytes
When set to a one the AD[31:0] bytes are swapped internally mapping AD[31:24] to bits[7:0], AD[23:16] to bits[15:8], AD[15:8] to bits[23:16] and AD[7:0] to bits[31:24].

Bit 3, +External Arbitration On
This bit enables arbitration for the DRAM and VRAM interface. When enabled it allows another system to read and write data into the VRAM and DRAM. This will be useful for PIP Controllers etc.

Bit 5-4, VRAM Size[1:0]
These bits specify the type of VRAM being used. Table 3-8 shows the encoding.

VRAM Size[1]	VRAM Size[0]	VRAM Type
0	0	512K x n
0	1	256K x n
1	0	1024K x n
1	1	128K x n

Table 3-8 VRAM Size Encoding

Bit 6, -Reset Queue
When set to a zero all dual rank control of the BLTer registers is reset and any waiting BLTs are discarded.

Bit 7, +Enable NAND
This bit is used for parametric testing, it should never be set.

Bit 10-8, TACK DLY[2:0] - Transfer Acknowledge Delay[3:0]
The value programmed in these bits specifies the number of system clock periods Mercury will wait to assert Transfer Acknowledge after it thinks the transfer is complete. This can be used to adjust register and RAM access to suite a given system.

Bit 15-11 Unused
These bits do nothing.

Status Register

This register is read only and provides status on mercury functions. A write of any value to the status registers clears the interrupt if it is active.

	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
Byte 0	+Queue1 Full	+Queue2 Full	+Interrupt Active	Unused	ID[4]	ID[3]	ID[2]	ID[1]

Figure 3-19 Status Register

Bit 3-0, ID[4:0]
These bits are used to identify the revision level of the chip, 4'b0001 = Rev A.

Bit 4, Unused
Set this bit to a zero to do nothing.

Bit 5, +Interrupt Active
When set to a one this bit indicates that the "-INT" pin is active. A write to this register clears the active interrupt.

Bit 6, +Queue 2 Full
When set to a one this bit indicates that the BLTer is busy.

Bit 7, +Queue 1 Full
When set to a one this bit indicates that the queue is full.

VRAS0_LB[28:0] - VRAM RAS[0] Lower Bound Pointer[28:0]

	Bit31	Bit 28	Bit 0
Byte 3-0	Unused	VRAS0_LB[28:0]	

Figure 3-20 VRAM RAS[0] Lower Bound Pointer

Bit 28-0, VRAM RAS[0] Lower Bound Pointer
This register is used to specify the lower bound address for the "VRAS[0]" output needed for bank designation during BLT address generation. When the address generated by the BLTer is greater then or equal to this value and less then or equal to the VRAS0_UB value "VRAS[0]" will be asserted if the VRAM is being accessed.

VRAS0_UB[28:0] - VRAM RAS[0] Upper Bound Pointer[28:0]

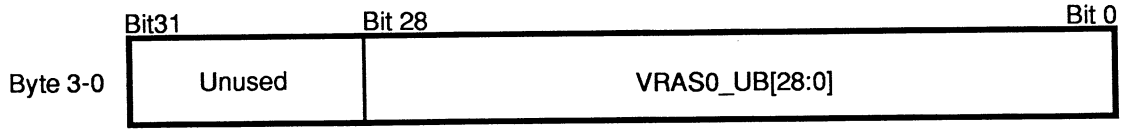


Figure 3-21 VRAM RAS[0] Upper Bound Pointer

Bit 28-0, VRAM RAS[0] Upper Bound Pointer
This register is used to specify the upper bound address for the "VRAS[0]" output needed for bank designation during BLT address generation. When the address generated by the BLTer is less then or equal to this value and greater then or equal to the VRAS0_LB value "VRAS[0]" will be asserted if the VRAM is being accessed.

VRAS1_LB[28:0] - VRAM RAS[1] Lower Bound Pointer[28:0]

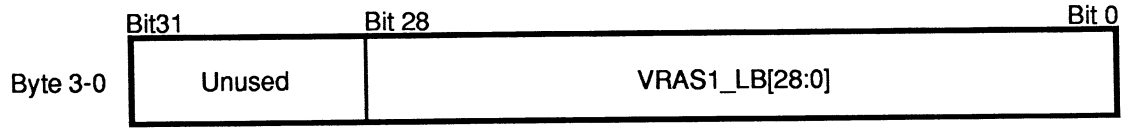


Figure 3-22 VRAM RAS[1] Lower Bound Pointer

Bit 28-0, VRAM RAS[1] Lower Bound Pointer
This register is used to specify the lower bound address for the "VRAS[1]" output needed for bank designation during BLT address generation. When the address generated by the BLTer is greater then or equal to this value and less then or equal to the VRAS0_UB value "VRAS[1]" will be asserted if the VRAM is being accessed.

VRAS1_UB[28:0] - VRAM RAS[1] Upper Bound Pointer[28:0]

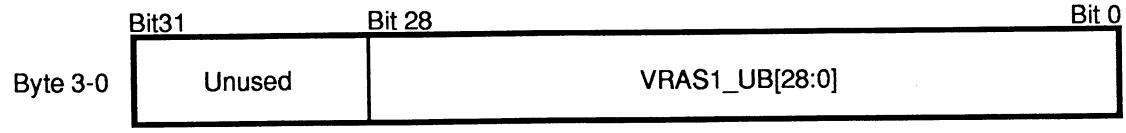


Figure 3-23 VRAM RAS[1] Upper Bound Pointer

Bit 28-0, VRAM RAS[1] Upper Bound Pointer

This register is used to specify the upper bound address for the "VRAS[1]" output needed for bank designation during BLT address generation. When the address generated by the BLTer is less then or equal to this value and greater then or equal to the VRAS0_LB value "VRAS[1]" will be asserted if the VRAM is being accessed.

VRAS2_LB[28:0] - VRAM RAS[2] Lower Bound Pointer[28:0]

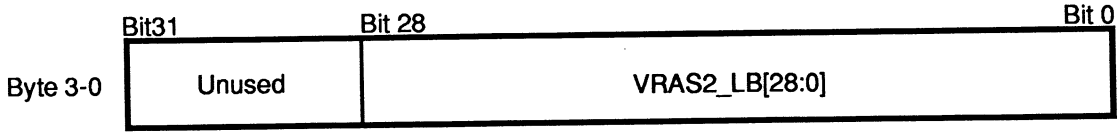


Figure 3-24 VRAM RAS[2] Lower Bound Pointer

Bit 28-0, VRAM RAS[2] Lower Bound Pointer
This register is used to specify the lower bound address for the "VRAS[2]" output needed for bank designation during BLT address generation. When the address generated by the BLTer is greater then or equal to this value and less then or equal to the VRAS0_UB value "VRAS[2]" will be asserted if the VRAM is being accessed.

VRAS2_UB[28:0] - VRAM RAS[2] Upper Bound Pointer[28:0]

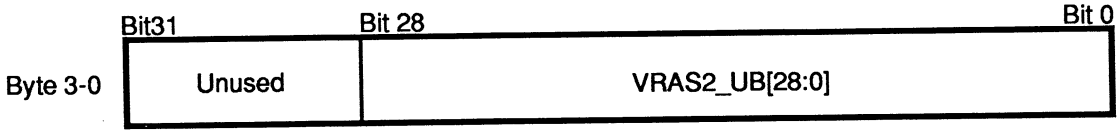


Figure 3-25 VRAM RAS[2] Upper Bound Pointer

Bit 28-0, VRAM RAS[2] Upper Bound Pointer
This register is used to specify the upper bound address for the "VRAS[2]" output needed for bank designation during BLT address generation. When the address generated by the BLTer is less then or equal to this value and greater then or equal to the VRAS0_LB value "VRAS[2]" will be asserted if the VRAM is being accessed.

VRAS3_LB[28:0] - VRAM RAS[3] Lower Bound Pointer[28:0]

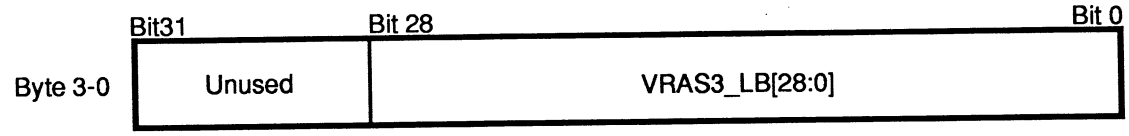


Figure 3-26 VRAM RAS[3] Lower Bound Pointer

Bit 28-0, VRAM RAS[3] Lower Bound Pointer
This register is used to specify the lower bound address for the "VRAS[3]" output needed for bank designation during BLT address generation. When the address

generated by the BLTer is greater then or equal to this value and less then or equal to the VRAS0_UB value "VRAS[3]" will be asserted if the VRAM is being accessed.

VRAS3_UB[28:0] - VRAM RAS[3] Upper Bound Pointer[28:0]

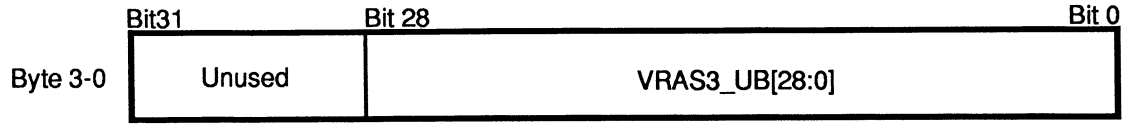


Figure 3-27 VRAM RAS[3] Upper Bound Pointer

Bit 28-0, VRAM RAS[3] Upper Bound Pointer
This register is used to specify the upper bound address for the "VRAS[3]" output needed for bank designation during BLT address generation. When the address generated by the BLTer is less then or equal to this value and greater then or equal to the VRAS0_LB value "VRAS[3]" will be asserted if the VRAM is being accessed.

BANK0_END[28:0] - DRAM Bank0 End Pointer[28:0]

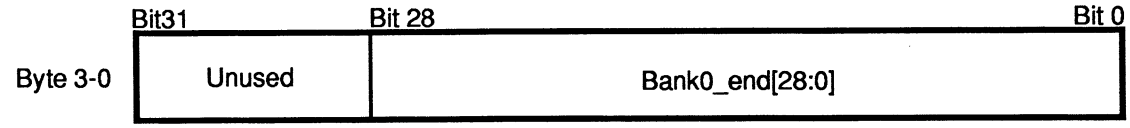


Figure 3-28 DRAM Bank0 End Pointer

Bit 28-0, DRAM Bank0 End Pointer
This register is used to specify the last address between banks of DRAM. When the address generated by the BLTer or input from the Host Port is less then or equal to this value "DRAS[0]" will be asserted. If the address is greater than this value "DRAS[1]" will be asserted.

4.0 Addressing

4.1 Introduction

Pixel arrays can be addressed in two different ways, square (not XY) and linear. Square addressing uses a start address value, an X-End address, a Y-End address, and a pitch (see Figures 2-5 to 2-9). The pitch is defined as the difference in addresses (bytes) between vertically adjacent pixels. The source and destination can have different pitches but in general the width and height should be relative. Linear addressing is a sub-addressing mode of square addressing. The Start and X-End address values are specified where X-End is just the ending address of the linear BLT. The Y-End value is the same as the Start and the pitch is that of the RAM being accessed (the Pitch Register is ignored since Y-End is the same as the Start). Keep in mind that when using linear addressing the BLT size cannot exceed 5120 bytes due to internal SRAM size.

4.2 Internal Registers

The internal registers are mapped into the address range from 0-7F (hex) using a cycle type of zero. Table 4-1 lists the addresses of each registers.

Register	Address	Register	Address
Status	0	S_START	40
System Control	4	S_XEND	44
VRAS0_LB	8	S_YEND	48
VRAS0_UB	C	S_PITCH	4C
VRAS1_LB	10	S_CEND	50
VRAS1_UB	14	S_INC	54
VRAS2_LB	18	OFFSET	58
VRAS2_UB	1C	D_START	5C
VRAS3_LB	20	D_XEND	60
VRAS3_UB	24	D_YEND	64
BANK0_END	28	D_PITCH	68
Unused	2C	D_CEND	6C
Unused	30	D_INC	70
Unused	34	Foreground	74
Unused	38	Background	78
Unused	3C	BLTer Control	7C

Table 4-1 Register Addresses

4.3 Dynamic RAM

Mercury controls an external Dynamic RAM (DRAM) array. This DRAM array can be various sizes ranging from 4 MBytes to 512 MBytes deep and can be accessed as bytes words or longs through the host port. DRAM refresh cycles are slaved to the RGS14188 refresh cycles. Table 4-2 shows the possible DRAM configurations and resulting sizes using RasterOps custom SIMMs.

Type	Bytes/SIMM	Bytes/Bank (4SIMMs)	Bytes/System(2 Banks)
256K x 4	1M	4M	8M
1M x 4	4M	16M	32M
4M x 4	16M	64M	128M
16M x 4	64M	256M	512M

Table 4-2 Possible DRAM Configurations

DRAM is only available on custom SIMM modules because of the 16-byte data path of the Mercury design. Standard SIMMs do not work with the Mercury chip.

4.4 Video RAM

Mercury can be interfaced to a frame buffer (Video RAM) 1, 2, 4, 8 , 12 or 16 MBytes deep. Mercury can control up to four different banks of VRAM with each bank consisting of up to 4 MBytes.

5.0 Performance

These performance values are based on current simulation results for a 200 by 200 pixel square which is 40,000 pixels.

5.1 Fill Constant

Pixel Depth	Speed (MPixels/Sec)
8-Bit Mode	270
16-bit Mode	164
24/32-bit Mode	90

5.2 Pattern Fills

Pixel Depth	Speed (MPixels/Sec)
8-Bit Mode	135
16-bit Mode	82
24/32-bit Mode	45

5.3 Single Operand Standard BLTs

Pixel Depth	Speed (MPixels/Sec)
8-Bit Mode	135
16-bit Mode	82
24/32-bit Mode	45

5.4 Double Operand Standard BLTs

Pixel Depth	Speed (MPixels/Sec)
8-Bit Mode	90
16-bit Mode	55
24/32-bit Mode	30

5.5 Single Operand Hilighting BLTs

Pixel Depth	Speed (MPixels/Sec)
8-Bit Mode	135
16-bit Mode	82
24/32-bit Mode	45

5.6 Single Operand Color Expanding BLTs

Pixel Depth	Speed (MPixels/Sec)
8-Bit Mode	135
16-bit Mode	82
24/32-bit Mode	45

5.7 Host Port DRAM Access

200 nS write, 300 nS read.

6.0 Design Tips and Pinout

6.1 Tips

Distribute the load on the _VRAS[3:0] signals. For example: If using one bank of VRAM wire RAS[0] from the VENUS chip to all the _VENUS_RAS inputs on the Mercury chip. Set the VRAS bound registers appropriately and wire the _VRAS[3:0] to equal numbers of VRAM. This will help guarantee RAS hold times with respect to address.

Use a silicon delay line for controlling the IPC_IN delay. Values should range from 0 to 6 nS.

Use a silicone delay line for controlling the VA_MCTRL delay. NAND all the used Venus RAS (outputs from the VENUS chip) signals and put the output into the delay line. The output from the delay line then connects to the VA_MCTRL input. The delay values should range for 15 to 30 ns.

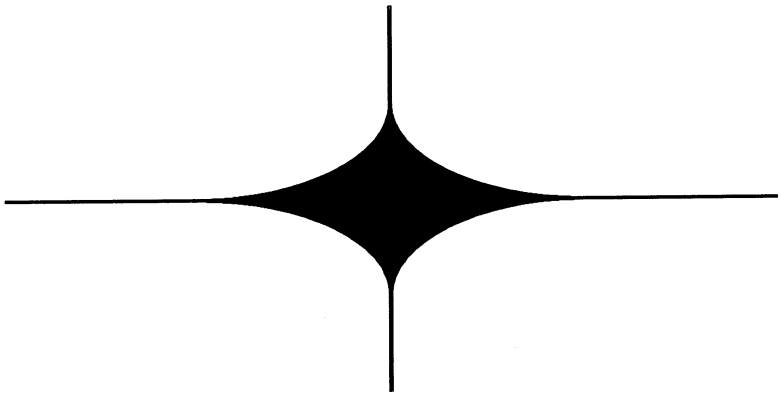
When using interrupt driven BLTing the queues could be dangerous due to interrupt service latency.

6.2 Pinout

1	AD[23]	41	RD[25]	81	RD[60]	121	RD[95]
2	GND	42	RD[26]	82	RD[61]	122	GND
3	AD[24]	43	RD[27]	83	RD[62]	123	RD[96]
4	AD[25]	44	RD[28]	84	RD[63]	124	RD[97]
5	AD[26]	45	RD[29]	85	GND	125	RD[98]
6	AD[27]	46	RD[30]	86	RD[64]	126	RD[99]
7	AD[28]	47	RD[31]	87	RD[65]	127	RD[100]
8	AD[29]	48	GND	88	RD[66]	128	RD[101]
9	AD[30]	49	RD[32]	89	RD[67]	129	RD[102]
10	AD[31]	50	RD[33]	90	RD[68]	130	RD[103]
11	GND	51	RD[34]	91	RD[69]	131	VCC
12	RD[0]	52	RD[35]	92	RD[70]	132	RD[104]
13	RD[1]	53	RD[36]	93	RD[71]	133	RD[105]
14	RD[2]	54	RD[37]	94	GND	134	RD[106]
15	RD[3]	55	RD[38]	95	RD[72]	135	RD[107]
16	RD[4]	56	RD[39]	96	RD[73]	136	RD[108]
17	RD[5]	57	VCC	97	RD[74]	137	RD[109]
18	RD[6]	58	GND	98	RD[75]	138	RD[110]
19	RD[7]	59	RD[40]	99	RD[76]	139	RD[111]
20	GND	60	RD[41]	100	RD[77]	140	GND
21	VCC	61	RD[42]	101	RD[78]	141	RD[112]
22	RD[8]	62	RD[43]	102	RD[79]	142	RD[113]
23	RD[9]	63	RD[44]	103	GND	143	RD[114]
24	RD[10]	64	RD[45]	104	VCC	144	RD[115]
25	RD[11]	65	RD[46]	105	RD[80]	145	RD[116]
26	RD[12]	66	RD[47]	106	RD[81]	146	RD[117]
27	RD[13]	67	GND	107	RD[82]	147	RD[118]
28	RD[14]	68	RD[48]	108	RD[83]	148	RD[119]
29	RD[15]	69	RD[49]	109	RD[84]	149	GND
30	GND	70	RD[50]	110	RD[85]	150	RD[120]
31	RD[16]	71	RD[51]	111	RD[86]	151	RD[121]
32	RD[17]	72	RD[52]	112	RD[87]	152	RD[122]
33	RD[18]	73	RD[53]	113	GND	153	RD[123]
34	RD[19]	74	RD[54]	114	RD[88]	154	RD[124]
35	RD[20]	75	RD[55]	115	RD[89]	155	RD[125]
36	RD[21]	76	GND	116	RD[90]	156	RD[126]
37	RD[22]	77	RD[56]	117	RD[91]	157	RD[127]
38	RD[23]	78	RD[57]	118	RD[92]	158	GND
39	GND	79	RD[58]	119	RD[93]	159	IPC_OUT
40	RD[24]	80	RD[59]	120	RD[94]	160	IPC_IN

Pinout (continued)

161	-CAS[0]	201	GND	241	SIZE[0]	281	AD[1]
162	-CAS[1]	202	VA[7]	242	+R_W	282	AD[2]
163	-CAS[2]	203	VA[8]	243	+ILE	283	AD[3]
164	GND	204	VA[9]	244	-HCS	284	AD[4]
165	VCC	205	GND	245	VA_MCTRL	285	AD[5]
166	-CAS[3]	206	-DOE	246	-TACK	286	AD[6]
167	-CAS[4]	207	-DWE	247	-INT	287	AD[7]
168	-CAS[5]	208	GND	248	VCC	288	GND
169	-CAS[6]	209	-DRAS[0]	249	-RESET	289	AD[8]
170	GND	210	-DRAS[1]	250	CLK	290	AD[9]
171	-CAS[7]	211	GND	251	GND	291	AD[10]
172	-CAS[8]	212	DA[0]	252	-188_BGACK	292	AD[11]
173	-CAS[9]	213	DA[1]	253	+188_BG	293	AD[12]
174	-CAS[10]	214	DA[2]	254	-188_BR[0]	294	AD[13]
175	GND	215	GND	255	-188_BR[1]	295	AD[14]
176	-CAS[11]	216	VCC	256	GND	296	AD[15]
177	-CAS[12]	217	DA[3]	257	188_REF	297	GND
178	-CAS[13]	218	DA[4]	258	-188_WE	298	AD[16]
179	-CAS[14]	219	DA[5]	259	-188_DTOE	299	AD[17]
180	GND	220	GND	260	-188_CAS[0]	300	AD[18]
181	-CAS[15]	221	DA[6]	261	-188_CAS[1]	301	AD[19]
182	-REF	222	DA[7]	262	-188_CAS[2]	302	AD[20]
183	-VDTOE	223	DA[8]	263	-188_CAS[3]	303	AD[21]
184	GND	224	GND	264	-188_RAS[0]	304	AD[22]
185	-VWE	225	DA[9]	265	-188_RAS[1]		
186	-VRAS[0]	226	DA[10]	266	-188_RAS[2]		
187	-VRAS[1]	227	DA[11]	267	-188_RAS[3]		
188	GND	228	GND	268	188_RACA[0]		
189	-VRAS[2]	229	-MBR[1]	269	188_RACA[1]		
190	-VRAS[3]	230	-MBR[0]	270	188_RACA[2]		
191	VA[0]	231	-MBGACK	271	188_RACA[3]		
192	GND	232	+MBG	272	188_RACA[4]		
193	VA[1]	233	+BLT_INT	273	188_RACA[5]		
194	VA[2]	234	CYCLE[1]	274	188_RACA[6]		
195	VA[3]	235	CYCLE[0]	275	188_RACA[7]		
196	VCC	236	GND	276	188_RACA[8]		
197	GND	237	+DIN_LE	277	188_RACA[9]		
198	VA[4]	238	DOUT_LE	278	GND		
199	VA[5]	239	CT	279	VCC		
200	VA[6]	240	SIZE[1]	280	AD[0]		



Darkstar Project Overview
Todd Witter
1/22/93
rev c

Darkstar is a general purpose high performance coprocessor board for the Mercury24. The daughtercard carries two AT&T DSP3210s running at 66MHz. These DSPs can access the Mercury24's large DRAM memory space extremely quickly using the DSP3210's block transfer capability. Each DSP also has a bank of very fast zero wait state SRAM available to it for quick data manipulation.

The DSP3210 is ideally suited and designed for the manipulation of digital signals such as those in captured or created computer images. In conjunction with Mercury 24, Darkstar is designed to provide performance increases in image filtering, format conversion, JPEG compression and decompression, bla bla bla.

DSP Control

Each DSP has a twelve bit control status (C/S) register. The layout of which is shown in figure 1.

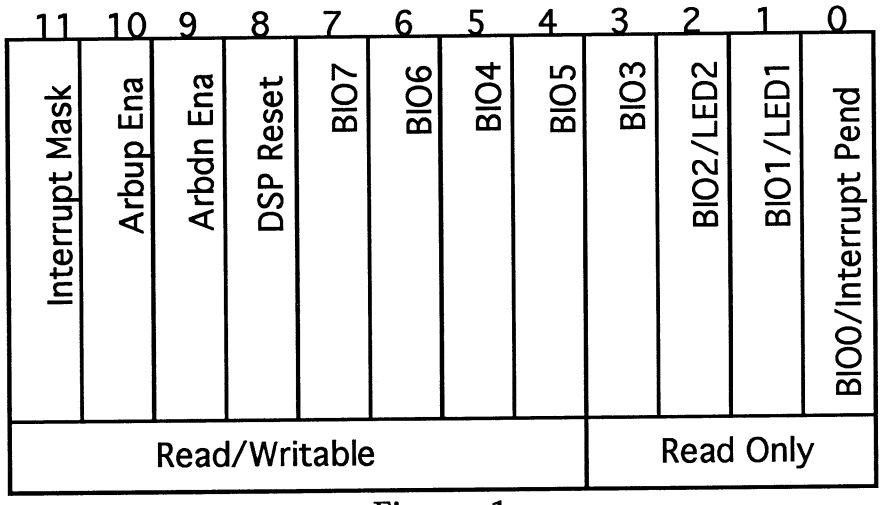


Figure 1

bits 11-9 are located within the Neptune DSP controller chip.
bit 11 is the interrupt mask bit (rw). A low value here enables the DSP's interrupt signal to be passed on to the Macintosh. It is set to one on a reset.

bit 10 is the arbup enable bit (rw). This bit is set to a one if there is another arbitrating device 'above' the Neptune chip. It is set to a zero if the Neptune chip is the top device in the arbitration chain. This bit is set to one on a reset. This bit's chief use is to speed up arbitration for the top device in the arbitration chain. This bit should only be changed while the DSP is inactive.

bit 9 is the arbdn enable bit (rw). This bit is set to a one if and only if it does not require use of the Mercury bus AND it is the top device in the arbitration chain or all devices 'above' it do not require the Mercury bus. It is set to a zero for operation of the Neptune/DSP. This bit is set to a one on a reset. This bit's chief use is to speed up arbitration for devices 'below' it in the arbitration chain. This bit should only be changed while the DSP is inactive.

bits 8-4 are located within the Darkstar controller chip.
bit 8 is the DSP reset bit (rw). This bit is used to manually reset the DSP and it's Neptune chip. A zero holds the two chips in their reset

state, a one releases them for operation. This bit is set to a zero on a Nubus reset.

bits 7-0 represent the BIO[7..0] pins of the DSP3210.
bits 7-4 are configured as outputs to the DSP and are read/writable. These four bits control the startup actions of the DSP after it is reset, it is important not to change the values of these bit until after the DSP reset cycle is complete. These bit are set to 0010 on a reset. Following the reset sequence these bits are user configurable.

bits 3-0 are configured as outputs from the DSP and are read only. These four bits represent the status of BIO bits 3-0. BIO[0] is used for the Nubus interrupt. If the DSP writes a zero to BIO[0] and the interrupt mask is zero then a Nubus interrupt occurs. BIO[2..1] are used to control the debug indicator lights. BIO[3] is user configurable.

The C/S register for DSP1 is located at 0xFs800000
The C/S register for DSP2 is located at 0xFs800004
On the Darkstar Card DSP2 and it's Neptune chip are at the 'top' of the arbitration chain. DSP1 resides above Mercury and below DSP2. Accesses to the Control/Status register do NOT affect the processing speed of the DSP.
The most significant nibble (D[31..28]) of DSP1's C/S register contains the daughterboard ID. The ID for Darkstar is 1. The ID Nibble is shown in Figure 2.

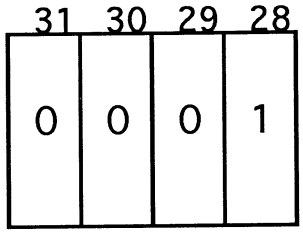


Figure 2

SRAM

Each DSP has either 128k or 512k of fast SRAM for it's use. Addressed from the DSP it is located at 0x00000000 and is repeated every 128k or 512kb depending on the amount of SRAM installed. The DSP's Memory map is shown in Figure 4.

This SRAM is also accessible from the Nubus. Each bank may be accessed separately or both banks may be written to simultaneously in a special memory location set aside for this use. Data read from this special area will only read one bank. The SRAM is fitted into the Mercury24's daughtercard space as follows:
SRAM for DSP1 has a base address of 0xFs880000.
SRAM for DSP2 has a base address of 0xFs900000.
The simultaneous write area has a base address of 0xFs980000.
A memory map is shown in Figure 3.

Accesses to the SRAM DO affect the processing speed of the DSP whose bank is being accessed but not the other DSP.

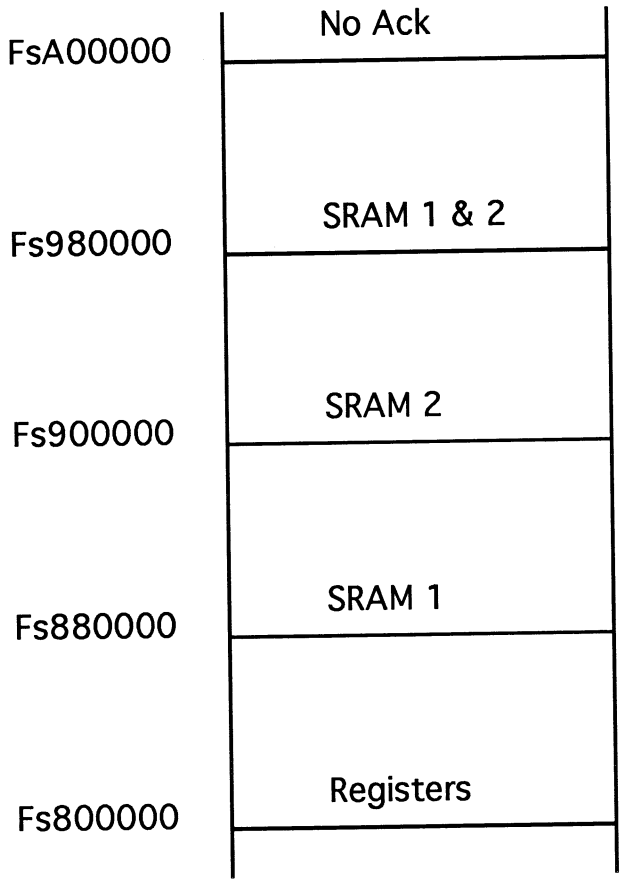


Figure 3

Mercury Bus

From the DSP the Mercury bus data space begins at location 0x80000000 and is repeated every 256MB. Addressed in this way Nubus DRAM addresses may be used untranslated by the DSP. The Address map as seen from the DSP is shown in Figure 4

From the Nubus this DRAM is accessible though the Mercury24's Super slot address space located at hex address 0xs0000000 where s is the slot in which the Mercury24 is installed. For more info on how the Mercury24 uses this DRAM, ask Jeff.

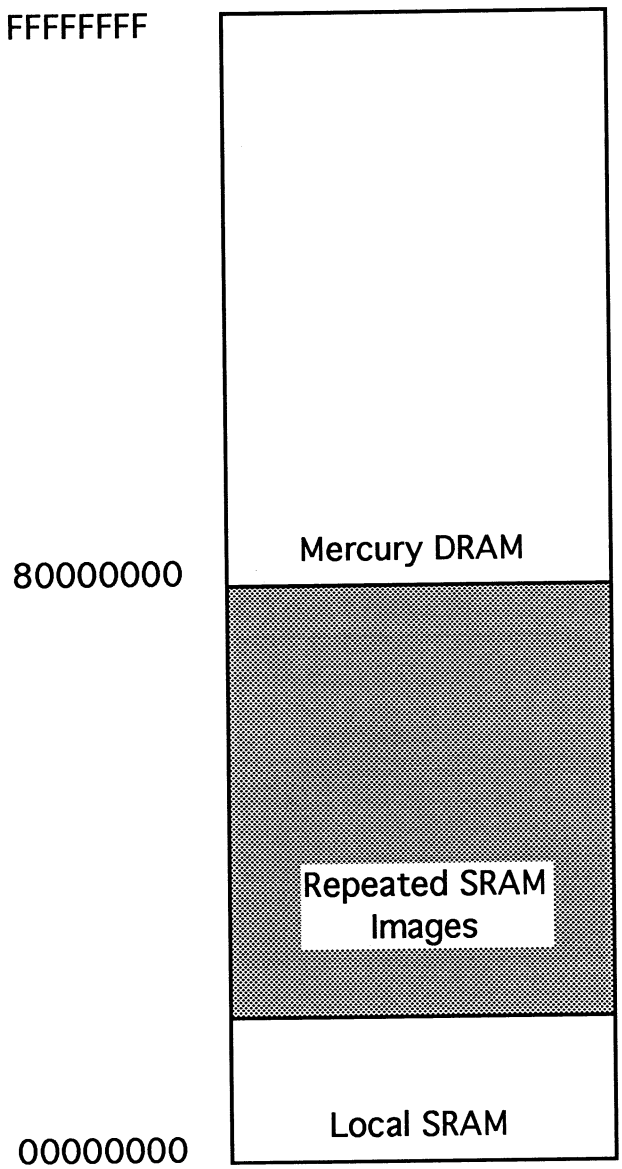


Figure 4

Mercury CY 8/31

- 1280 x 1024 16 BIT OR ~~LESS~~
- PIXEL CLOCK 107.573 MHZ
- VID CLK 6.70 MHZ

Mclock 37C2F

	34061	ACTUAL	RGS 14188	PIX/8 FOR VIDCLK
HES	A	B	14	
HEB	17	18	2E	
HSB	67	68	CE	
HT	68	69	DØ	
VES	2	3	2	
VEB	24	25	24	
VSB	424	425	424	
VT	427	428	427	

ACTUAL f_{TAG} = 107.619

SER WORD = 01BB813

```
rgs96981_status = 0,
rgs96981_system_control = 1,
rgs96981_vras0_lb = 2, 0
rgs96981_vras0_ub = 3, 3FFFFF
rgs96981_vras1_lb = 4, 0
rgs96981_vras1_ub = 5, 3FFFFF
rgs96981_vras2_lb = 6, 0
rgs96981_vras2_ub = 7, 3FFFFF
rgs96981_vras3_lb = 8, 0
rgs96981_vras3_ub = 9, 3FFFFF
rgs96981_bank0_end = 10, 1FFFFFFF
rgs96981_unused_0 = 11,
rgs96981_unused_1 = 12,
rgs96981_unused_2 = 13,
rgs96981_unused_3 = 14,
rgs96981_unused_4 = 15,
rgs96981_s_start = 16,
rgs96981_s_xend = 17,
rgs96981_s_yend = 18,
rgs96981_s_pitch = 19,
rgs96981_s_cend = 20,
rgs96981_s_inc = 21,
rgs96981_offset = 22,
rgs96981_d_start = 23,
rgs96981_d_xend = 24,
rgs96981_d_yend = 25,
rgs96981_d_pitch = 26,
rgs96981_d_cend = 27,
rgs96981_d_inc = 28,
rgs96981_foreground = 29,
rgs96981_background = 30,
rgs96981_blter_control = 31
```

RGS1433 Specification
(EARTH)
Preliminary
Revision Level A1

0002-0484
RasterOps
January 11, 1993
Jeff Tingley

1.0 Earth

1.1 Introduction

Earth is a NuBus, slave only, interface chip. Figure 1-1 shows a functional drawing of the Earth chip. Earth is designed to control the NuBus system level interface to configuration ROM, ASICs such as Venus, Mercury, and Jupiter, and a RAM DAC. It contains all the necessary address decode logic to map all 16 MBytes of a standard NuBus card as well as all 256 MBytes of a super slot card.

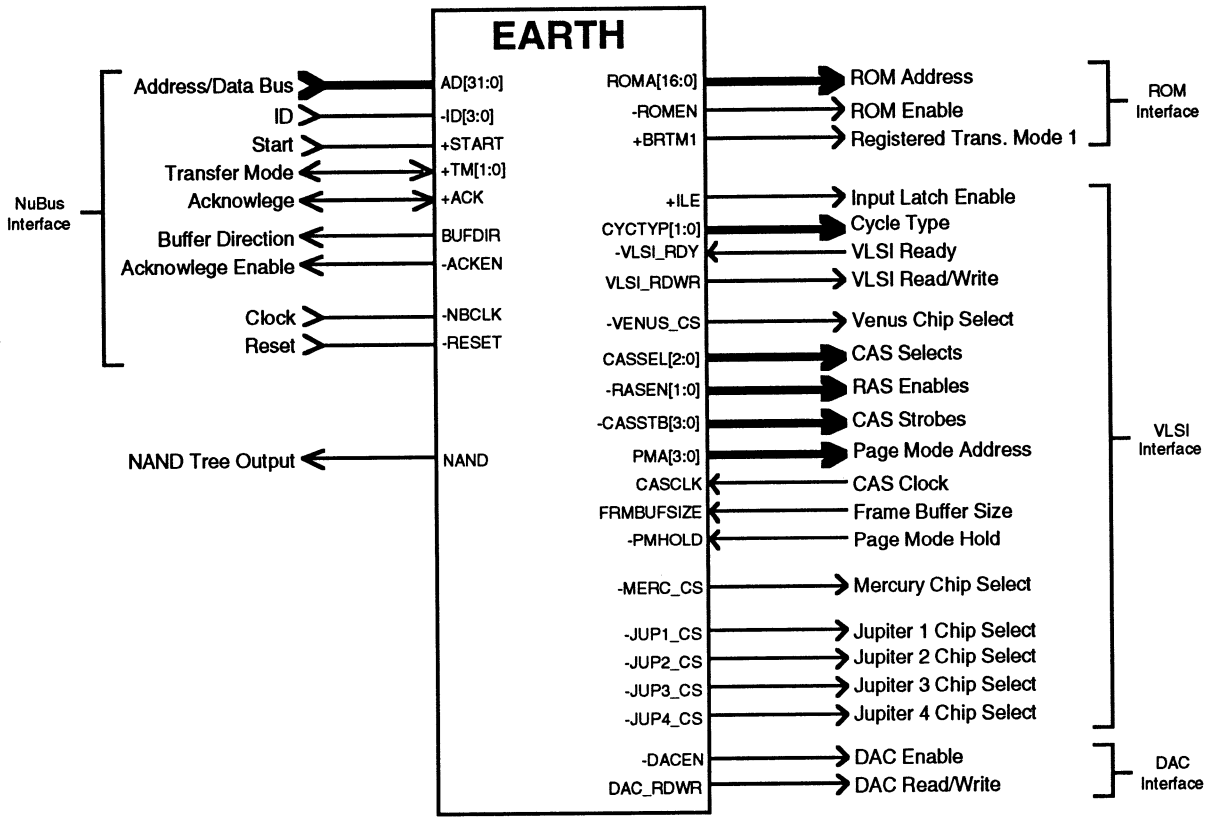


Figure 1-1 Functional Drawing

1.2 Pin Description

NuBus Interface

AD[31:0] - Address/Data Bus[31:0] (TTL Input)
NuBus Address/Data bus used for address space decoding. These inputs are just inverted versions of the Nubus Address/Data bus.

-ID[3:0] - ID[3:0] (TTL Inputs)
These four inputs identify the address of the slot earth is plugged into. This bits are necessary for both standard and super slot decoding. These inputs wire directly to the ID bits of the NuBus.

+START- START(TTL Input)
This is the inverted Start signal from the NuBus. It signals the beginning of a bus cycle.

+TM[1:0] - Transfer Mode [1:0] (TTL Bidirectional, 4mA Drive)
These I/O indicate the transfer mode and size as well as acknowledge type. They should be inverted/buffered and wired directly to the Nubus Transfer Mode signals.

+ACK - Acknowledge (TTL Bidirectional, 4mA Drive)
This output is used to acknowledge NuBus transactions. It should be inverted/buffered and wired directly to NuBus Acknowledge.

BUFDIR- Buffer Direction (TTL Output, 4mA Drive)
This output is used to control the direction of the NuBus Address/Data transceivers. This signal is driven high when data is to be driven onto the Nubus.

-ACKEN - Acknowledge Enable (TTL Output, 4mA Drive)
This output is used to control the direction of the transceiver used to buffer and invert "+TM[1:0]" and "+ACK". This signal is driven low when "+TM[1:0]" and "+ACK" are to be driven onto the NuBus.

-NBCLK - NuBus Clock (TTL Input)
This is the 75% duty cycle, 10 MHz, NuBus clock. Earth uses both the rising and falling edges of this clock to control internal functions.

-RESET - Reset (TTL Input)
This active low input is used to bring Earth to a know state. "-NBCLK" must be running during reset.

ROM Interface

ROMA[16:0] - ROM Address [16:0] (TTL Outputs, 4mA Drive)
These outputs are used to address the ROM and other devices the require the address to be stable throughout the entire NuBus cycle.

-ROMEN - ROM Enable (TTL Output, 4mA Drive)
This output is used to enable the ROM data bus drivers.

+BRTM1 - Buffered/Registered Transfer Mode 1 (TTL Output, 4mA Drive)
This output is used to control the write function of a Flash EPROM. It is just a registered version of the Nubus Transfer Mode 1 signal.

VLSI Interface

+ILE - Input Latch Enable (TTL Output, 4mA Drive)
This output is used to close the address and control latches on both the Venus and Mercury chips. "+ILE" should be wired directly to the +ILE inputs of both chips.

CYCTYP[1:0] - Cycle Type [1:0] (TTL Output, 4mA Drive)
These outputs tell Venus and Mercury what type of cycle to execute. They should be wired directly to their corresponding inputs on Venus and Mercury.

-VLSI_RDY- VLSI Ready (TTL Input)
This input should be the wire-or of the ready signals from Venus and Mercury. When asserted it indicates the current cycle is ready to proceed.

-VENUS_CS - Venus Chip Select (TTL Output, 4mA Drive)
When driven low this output initiates a Venus cycle. It should be wired directly to the corresponding Venus input.

VLSI_RDWR - VLSI Read/Write (TTL Output, 4mA Drive)
When low at the falling edge of "+ILE" this output indicates a write is to be performed, when high a read is to be performed. This signal should be wired directly to the corresponding inputs on both Venus and Mercury.

CASSEL[2:0]- CAS Select [2:0] (TTL Output, 4mA Drive)
These output specify to Venus which CAS signals to assert. They should be wired directly to the corresponding Venus inputs.

-RASEN[1:0] - RAS Enable [1:0] (TTL Output, 4mA Drive)
These signals indicate which RAS signals should be asserted for a given Venus cycle. "-RASEN[0]" is asserted for the first four megabytes of the frame buffer. "-RASEN[1]" is asserted for the second four megabytes. These outputs should be wired directly to the corresponding inputs on the Venus.

-CASSTB[3:0] - CAS Strobe [3:0] (TTL Output, 4mA Drive)
These outputs are used to strobe CAS during page mode cycles. They should be wired directly to the corresponding inputs on the Venus.

PMA[3:0]- Page Mode Address [3:0] (TTL Output, 4mA Drive)
These outputs contain the low four (AD[5:2]) address signals during standard cycles and the page mode address during page mode cycles. They should be wired to the low four address inputs of the Veuns.

CASCLK- CAS Clock (TTL Input)
This input is a delayed version of the NuBus clock. It is used to generate the proper timing for the "-CASSTB[3:0]" signals.

FRMBUFSIZE- Frame Buffer Size (TTL Input)
When low this input specifies a 4 MByte frame buffer (No ACK for the second 4 MBytes), high specifies a 8 MByte Frame buffer.

-PMHOLD - Page Mode Hold (TTL Input,)
When driven low this input indicates the Venus would like to hold the current page mode cycle. This input should be wired directly to the corresponding output of the Venus.

-MERC_CS - Mercury Chip Select (TTL Output, 4mA Drive)
When driven low this output initiates a Mercury cycle. It should be wired directly to the corresponding Mercury input.

-JUP1_CS - Jupiter 1 Chip Select (TTL Output, 4mA Drive)
When driven low this output initiates a Jupiter cycle. It should be wired directly to the corresponding Jupiter input.

-JUP2_CS - Jupiter 2 Chip Select (TTL Output, 4mA Drive)
When driven low this output initiates a Jupiter cycle. It should be wired directly to the corresponding Jupiter input.

-JUP3_CS - Jupiter 3 Chip Select (TTL Output, 4mA Drive)
When driven low this output initiates a Jupiter cycle. It should be wired directly to the corresponding Jupiter input.

-JUP4_CS - Jupiter 4 Chip Select (TTL Output, 4mA Drive)
When driven low this output initiates a Jupiter cycle. It should be wired directly to the corresponding Jupiter input.

DAC Interface

-DACEN - DAC Enable (TTL Output, 4mA Drive)
This output when low enables DAC read or write cycles. It should be wired directly to the BT496/497 corresponding input.

DAC_RDWR - DAC Read/Write (TTL Output, 4mA Drive)
This output when high indicates a read is to be performed and when low indicates a write is to be performed. It should be wired directly to the BT496/497 corresponding input.

2.0 Earth Functionality

2.1 Introduction

Earth is a bus specific (NuBus) interface controller. It allows an easy, inexpensive, and high performance connection of a Mercury and Venus based system to the NuBus. The address map for the Earth chip is shown in Table 2-1. Earth is packaged in a 144-pin Plastic Quad Flat Pack (PQFP).

Address Range	Access
Fs00.0000 - Fs3F.FFFF	Frame Buffer (1st 4 MBytes)
Fs40.0000 - Fs7F.FFFF	Frame Buffer (2nd 4 MBytes, FRSIZE = 1)
Fs80.0000 - FsCF.FFFF	Expansion Slot Space (No Ack)
FsD0.0000 - FsDF.FFFF	Unused (No Ack)
FsE0.0000 - FsEF.FFFF	RGS4909 Based Accelerator (No Ack)
FsF0.0000 - FsF0.00FF	DAC
FsF0.0100 - FsF0.01FF	Jupiter #1
FsF0.0200 - FsF0.02FF	Jupiter #2
FsF0.0300 - FsF0.03FF	Jupiter #3
FsF0.0400 - FsF0.04FF	Jupiter #4
FsF0.0500 - FsF0.05FF	Mercury Register
FsF0.0600 - FsF0.06FF	Venus Register
FsF8.0000 - FsFF.FFFF	Flash ROM
s000.0000 - sFFF.FFFF	Mercury DRAM (super slot space)

Table 2-1 Earth Address Map

A block diagram of a system designed around the earth chip is shown in Figure 2-1.

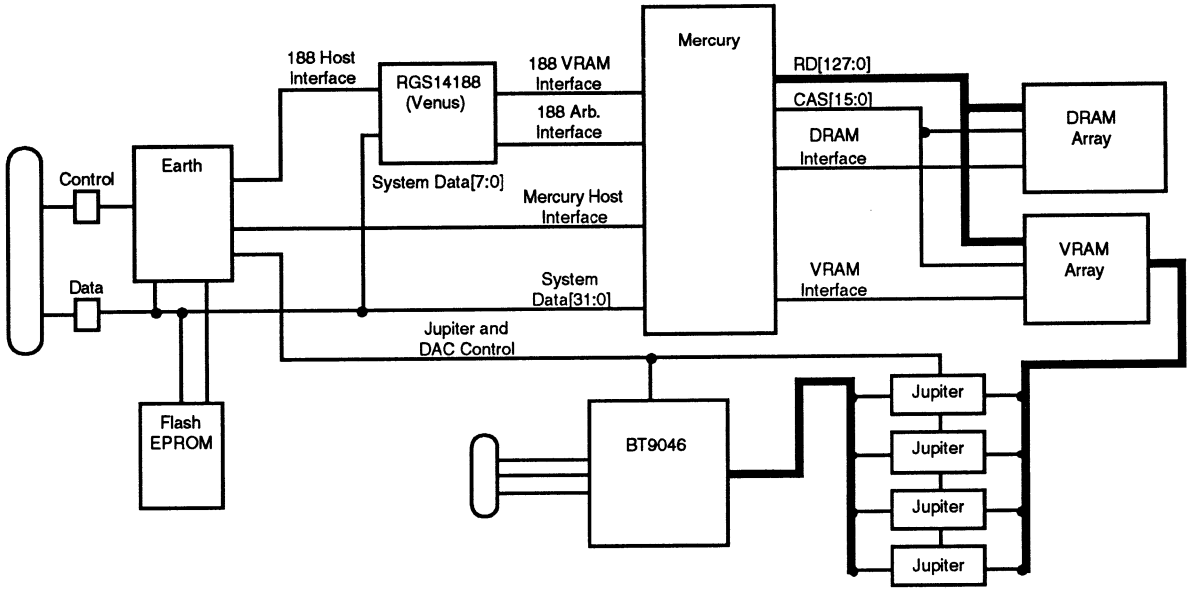


Figure 2-1 Earth Based System

6.0 Design Tips and Pinout

6.1 Tips

See Mercury 24 design for actual implementation.

6.2 Pinout

1	GND	41	AD[21]	81	NC	121	-CASSTB[3]
2	GND	42	AD[20]	82	-JUP1_CS	122	NAND
3	ROMA[4]	43	AD[19]	83	-JUP2_CS	123	-RESET
4	ROMA[5]	44	NC	84	-JUP3_CS	124	AD[24]
5	ROMA[6]	45	NC	85	-JUP4_CS	125	-NBCLK
6	ROMA[7]	46	NC	86	DAC_RDWR	126	NC
7	ROMA[8]	47	AD[18]	87	-DAC_EN	127	NC
8	ROMA[9]	48	AD[17]	88	NC	128	NC
9	NC	49	AD[16]	89	NC	129	+BRTM1
10	NC	50	AD[15]	90	VCC	130	CASCLK
11	NC	51	AD[14]	91	NC	131	AD[25]
12	ROMA[10]	52	AD[13]	92	NC	132	AD[26]
13	ROMA[11]	53	AD[12]	93	NC	133	ROMA[3]
14	ROMA[12]	54	NC	94	-VENUS_CS	134	ROMA[16]
15	ROMA[13]	55	NC	95	VLSI_RDWR	135	-ACKEN
16	ROMA[14]	56	NC	96	CYCTYP[0]	136	NC
17	ROMA[15]	57	AD[11]	97	CYCTYP[1]	137	NC
18	NC	58	AD[10]	98	+ILE	138	NC
19	NC	59	AD[9]	99	-RASEN[0]	139	BUFDIR
20	NC	60	AD[8]	100	GND	140	-ROMEN
21	NC	61	AD[7]	101	NC	141	ROMA[0]
22	+START	62	AD[6]	102	NC	142	ROMA[1]
23	+ACK	63	AD[5]	103	-RASEN[1]	143	ROMA[2]
24	-ID[0]	64	NC	104	+TM[0]	144	VCC
25	-ID[1]	65	NC	105	CASSEL[2]		
26	-ID[2]	66	AD[4]	106	CASSEL[0]		
27	-ID[3]	67	AD[3]	107	CASSEL[1]		
28	GND	68	AD[2]	108	VCC		
29	NC	69	AD[1]	109	GND		
30	NC	70	AD[0]	110	GND		
31	AD[31]	71	VCC	111	PMA[0]		
32	AD[30]	72	GND	112	PMA[1]		
33	AD[29]	73	GND	113	PMA[2]		
34	AD[28]	74	+TM[1]	114	PMA[3]		
35	AD[27]	75	FRMBUFSIZE	115	-CASSTB[0]		
36	VCC	76	-PMHOLD	116	NC		
37	GND	77	-VLSI_RDY	117	NC		
38	GND	78	-MERC_CS	118	NC		
39	AD[23]	79	NC	119	-CASSTB[1]		
40	AD[22]	80	NC	120	-CASSTB[2]		

7.0 Timing Specifications

7.1 Test Conditions

Timing characterization of the Earth Verilog model was done using both best case and worst case conditions. In both cases, the internal delays were calculated using the actual capacitance figures from the chip layout. Parameters for the test condtions are shown in Table 7-1. Test loads for the output pins are shown in Table 7-2.

Parameter	Best	Worst
Process	BCS	WCS
Temperature	15°C	80°C
Voltage	5.25 V	4.75 V

Table 7-1. Test Conditions

Signal	Load	Units
+TM[1:0]	10	pF
+ACK	10	pF
BUFDIR	50	pF
-ACKEN	10	pF
ROMA[16:0]	30	pF
-ROMEN	20	pF
+BRTM1	20	pF
+ILE	20	pF
+CYCTYP[1:0]	20	pF
VLSI_RDWR	20	pF
-VENUS_CS	10	pF
+CASSEL[2:0]	10	pF
-RASEN[1:0]	10	pF
-CASSTRB[3:0[	10	pF
PMA[3:0]	10	pF
-MERC_CS	10	pF
-JUP1_CS	10	pF
-JUP2_CS	10	pF
-JUP3_CS	10	pF
-JUP4_CS	10	pF
-DACEN	10	pF
DAC_RDWR	10	pF

Table 7-2. Output Pin Test Loads

7.2 Primary Nubus Timing

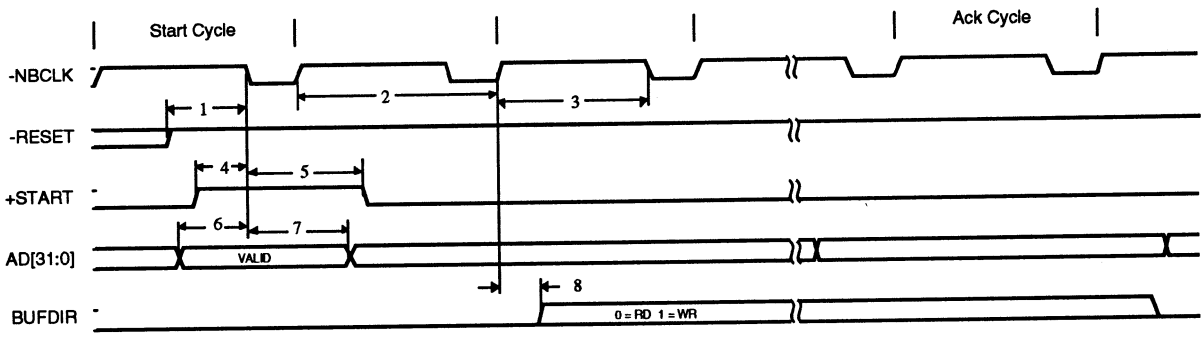


Figure 7-1. Primary Nubus Timing

Parameter	Symbol	Min	Max	Units
-RESET Setup	1	2		ns
-NBCLK Period	2	99.99	100.01	ns
-NBCLK Width	3	73	77	ns
+START Setup	4	3		ns
+START Hold	5	1		ns
AD[31:0] Setup	6	2		ns
AD[31:0] Hold	7	4		ns
BUFDIR Delay	8	7	23	ns

Table 7-3. Parameters for Figure 7-1

Notes:
1. All delays in this document are the same for both rising and falling edges unless otherwise specified.

7.3 +ACK and +TM Timing

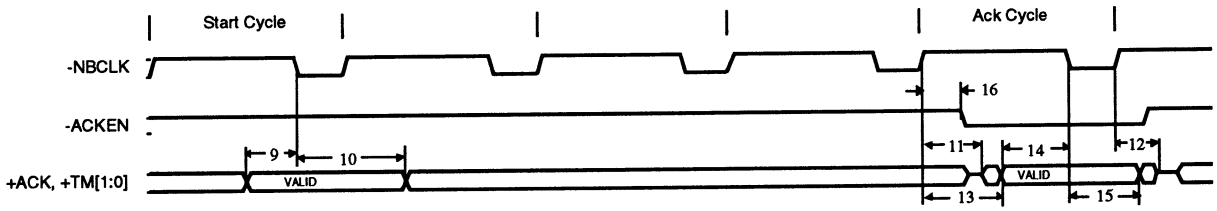


Figure 7-2. +ACK and +TM Timing for ROM, DAC, and Jupiter Cycles

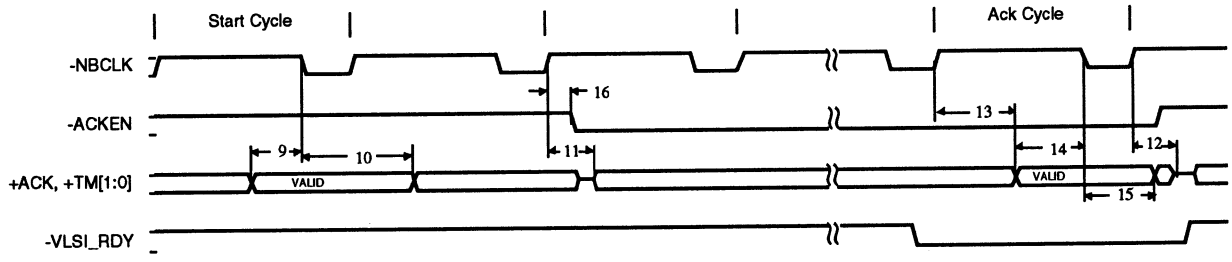


Figure 7-3. +ACK and +TM Timing for Mercury and Venus Cycles

Parameter	Symbol	Min	Max	Units
+ACK, +TM Setup Required	9	2		ns
+ACK, +TM Hold Required	10	3		ns
+ACK, +TM Access	11	6	15	ns
+ACK, +TM Recovery	12	7	20	ns
+ACK, +TM Delay	13	6	21	ns
+ACK, +TM Setup Provided	14	52		ns
+ACK, +TM Hold Provided	15	29		ns
-ACKEN Delay	16	6	19	ns

Table 7-4. Parameters for Figures 7-2 and 7-3

Notes:
1. -VLSI_RDY is shown in Figure 7-3 for clarity. See Figure 7-7 for timing information.

7.4 ROM, DAC, and Jupiter Interface Timing

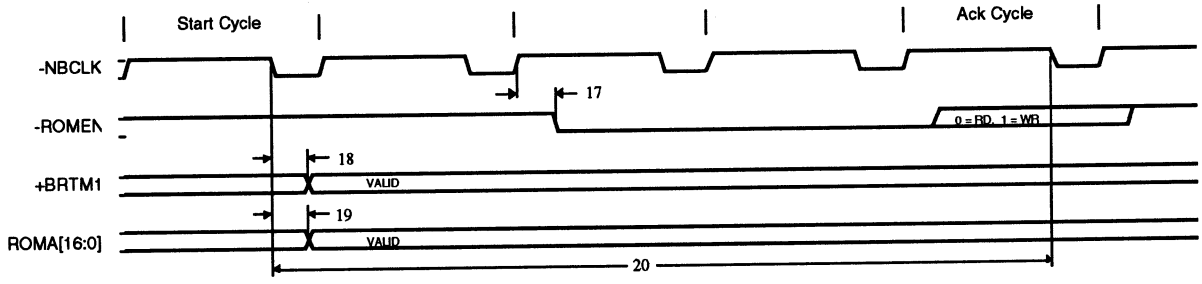


Figure 7-4. ROM Interface Timing

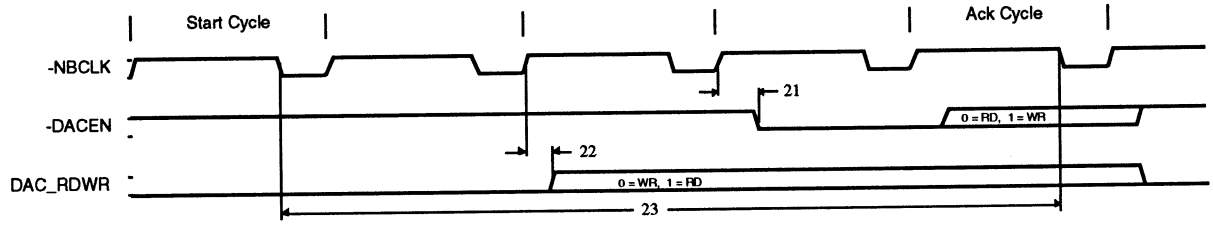


Figure 7-5. DAC Interface Timing

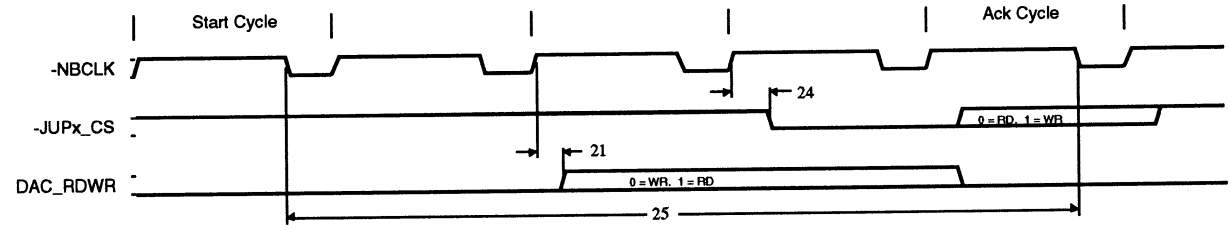


Figure 7-6. Jupiter Interface Timing

Parameter	Symbol	Min	Max	Units
-ROMEN Delay	17	7	21	ns
+BRTM1 Delay	18	6	20	ns
ROMA[16:0] Delay	19	7	18	ns
ROM Access Time	20	400	400	ns
-DACEN Delay	21	6	19	ns
DAC_RDWR Delay	22	6	18	ns
DAC Access Time	23	400	400	ns
-JUPx_CS Delay	24	6	19	ns
Jupiter Access Time	25	400	400	ns

Table 7-5. Parameters for Figures 7-4, 7-5, and 7-6

7.5 Mercury and Venus Interface Timing

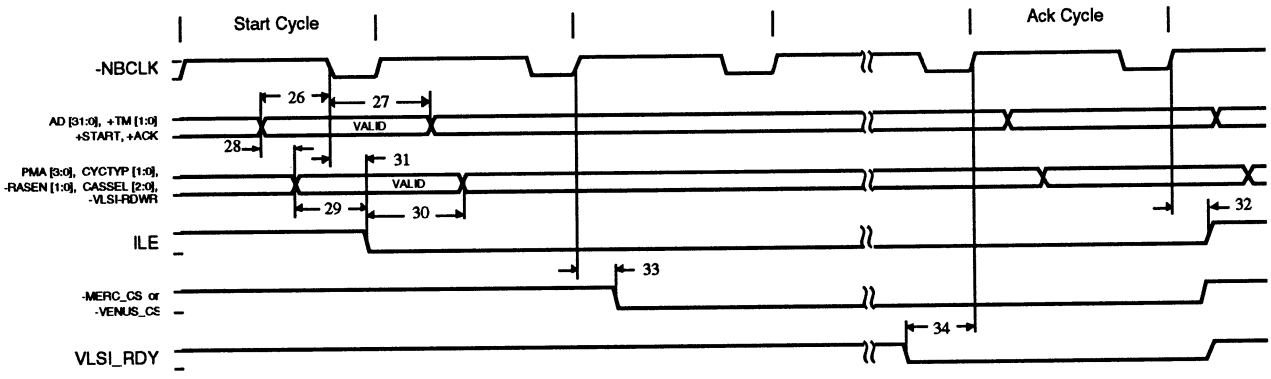


Figure 7-7. Mercury and Venus Interface Timing

Parameter	Symbol	Min	Max	Units
NB Signals Actual Setup	26	Note 4		ns
NB Signals Actual Hold	27	Note 4		ns
VLSI Signals Delay	28	4	20	ns
VLSI Signals Setup Provided	29	21		ns
VLSI Signals Hold Provided	30	14		ns
ILE Falling Edge Delay	31	6	20	ns
ILE Rising Edge Delay	32	6	19	ns
VLSI Chip Selects Delay	33	6	19	ns
-VLSI_RDY Setup Required	34	2		ns

Table 7-6. Parameters for Figure 7-7

Notes:
1. The signals AD, +TM, +START, and +ACK are referred to as "NB Signals."
2. The signals PMA, CYCTYP, -RASEN, CASSEL, and -VLSI_RDWR are referred to as "VLSI Signals."
3. The signals -MERC_CS and -VENUS_CS are referred to as "VLSI Chip Selects."
4. The "NB Signals" are shown in Figure 7-7 only to demonstrate the combinational delay from "NB Signals" to "VLSI Signals." The actual setup and hold times of these signals affect the setup and hold times of the "VLSI Signals" relative to ILE.
5. The "VLSI Signals" setup and hold times shown in Table 7-6 were calculated based on "NB Signals" setup and hold times of 21 ns and 23 ns respectively.
6. See Figures 7-1, 7-2, and 7-3 for minimum "NB Signal" setup and hold times required for the Earth chip to function properly.
7. -VLSI_RDY is an asynchronous input to the Earth chip. If the setup time shown in Table 7-6 is met, -VLSI_RDY will be detected during that Nubus cycle. Otherwise, it will be detected during the following Nubus cycle.

7.6 Page Mode Cycle Timing

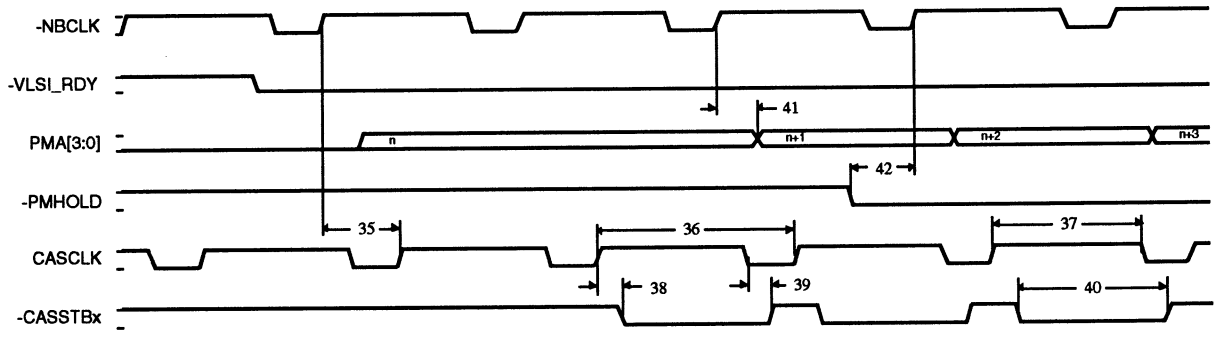


Figure 7-8. Page Mode Cycle Timing

Parameter	Symbol	Min	Max	Units
CASCLK Delay	35	32	76	ns
CASCLK Period	36	99.99	100.01	ns
CASCLK Width	37	73	77	ns
-CASSTB Fall Delay	38	5	15	ns
-CASSTB Rise Delay	39	4	13	ns
-CASSTB Width	40	72	77	ns
PMA Delay	41	7	21	ns
-PMHOLD Setup	42	1		ns

Table 7-7. Parameters for Figure 7-8

Notes:
1. -PMHOLD is an asynchronous input to the Earth chip. If the setup time shown in Table 7-7 is met, -PMHOLD will be detected during that Nubus cycle. Otherwise, it will be detected during the following Nubus cycle.
2. VLSI_RDY is shown in Figure 7-8 for clarity. See Figure 7-7 for timing information.

Advance Information

This document contains information on a product under development. The parametric information contains target parameters and is subject to change.

Bt496

100 MHz Monolithic CMOS 256 x 9 Triple Color Palette CMYK/RGB RAMDAC™

Distinguishing Features

- CMYK-to-RGB Conversion
- 100 MHz Operation
- 4:1 to 128:1 Multiplexed Pixel Port
- Three 256 x 9 Color Palette RAMs
- 1x to 16x Integer Zoom Support
- 1, 2, 4, 8, 16, or 32 Bits per Pixel
- Pixel Panning Support
- Simultaneous Support of Zoom, Panning, and Pixel Unpacking
- Programmable Setup (0 or 7.5 IRE)
- 9-Bit DACs
- VRAM Clock generation
- TTL SYNC output
- 207-pin PGA Package

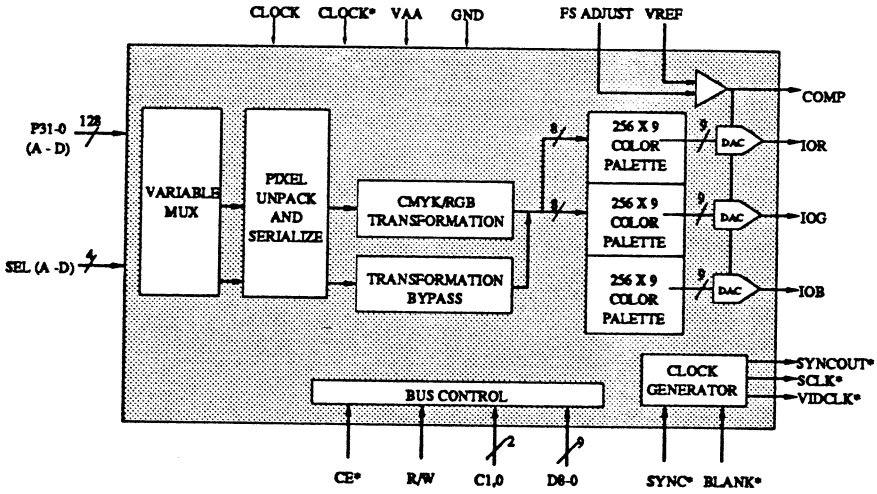
Applications

- Printer Prepress
- Desktop Color
- High-Resolution Color Graphics

Benefits

- Smaller boardspace
- Ease of design with turnkey solution
- Provides new capabilities to end customers

Functional Block Diagram



Product Description

The Bt496 triple 9-bit RAMDAC is designed specifically for high-performance, high-resolution color graphics used in printer prepress and desktop color applications. The multiple pixel ports and internal multiplexing enable TTL-compatible interfacing to the frame buffer, while maintaining the 100 MHz video data rates required for sophisticated color graphics.

The CMYK-to-RGB transformation block allows the frame buffer to manipulate data in the CMYK color space while the high-resolution monitor continues to display the image in RGB color space.

On-chip features include three 256 x 9 color palette RAMs, 4:1 input multiplexing of the pixel port, bit plane masking, programmable setup (0 or 7.5 IRE), pixel panning support, 1x to 16x integer zoom support, and CMYK-to-RGB color space conversion.

Pixel data may be input as 1, 2, 4, 8, 16, or 32 bits per pixel.

Brooktree Corporation • 9950 Barnes Canyon Rd. • San Diego, CA 92121
(619) 452-7580 • (800) VIDEO IC • TLX: 383 596 • FAX: (619) 452-1249
L496001 Rev. A 7/1/92

Brooktree®

Circuit Description

MPU Interface

As illustrated in the functional block diagram, the Bt496 supports a standard MPU bus interface, allowing the MPU direct access to the internal control registers and color palettes. The dual-port color palette RAMs allow color updating without contention with the display refresh process.

The control inputs, in conjunction with the internal address register, specify which control register or color palette location will be accessed by the MPU (see Table 1). The 9-bit address register eliminates the requirement for external address multiplexers. ADDR0 is the least significant bit.

To write color data, the MPU loads the address register with the address of the color palette RAM location to be modified. The MPU performs three successive write cycles (9 bits each of red, green, and blue), using the internal address register to select the primary color palette RAM. After the blue write cycle, the address register then increments to the next location, which the MPU may continue by simply writing another sequence of red, green, and blue data. Reading color data is similar to writing, except the MPU executes read cycles.

When accessing the color palette RAM, the address register increments after each blue read or write cycle. To keep track of the red, green, and blue read/write cycles, the address register has two additional bits (ADDRa, ADDRb) that count modulo three. They are reset to zero when the MPU reads or writes to the address register. The MPU does not have access to these bits. The other 9 bits of the address register (ADDR0–8) are accessible to the MPU. ADDR0 corresponds to D0.

Addr8–0	C1,C0	Addressed by MPU
$xxx	00	Address Register (Addr8-0)
$xxx	01	Reserved
$000	10	ID Register ($081)
$001	10	Revision Register
$002	10	Command Register 0
$003	10	Command Register 1
$004	10	Command Register 2
$006	10	Red Pixel Read Mask Register
$007	10	Green Pixel Read Mask Register
$008	10	Blue Pixel Read Mask Register
$00C	10	Red Signature Register
$00D	10	Green Signature Register
$00E	10	Blue Signature Register
$00F	10	Test Register
$100–$1FF	10	CMYK correction factor LUT
$000–$0FF	11	Color Palette RAMs

Table 1. MPU Addressing Map.

Circuit Description (continued)

Although the color palette RAM registers are dual-ported, if the pixel data is addressing the same palette entry being written to by the MPU during the write cycle, it is possible for one or more of the pixels on the display screen to be disturbed. Only one pixel is disturbed if the write data from the MPU is valid during the entire chip enable time.

Accessing the control registers is also done through the address register in conjunction with the C inputs, as shown in Table 1. All control registers may be written to or read by the MPU at any time. When accessing the control registers and CMYKLUT, the address register increments following a read or write cycle.

If an invalid address is loaded into the address register, data written to the device will be ignored and invalid data will be read by the MPU.

Figure 1 illustrates the MPU read/write timing of the Bt496.

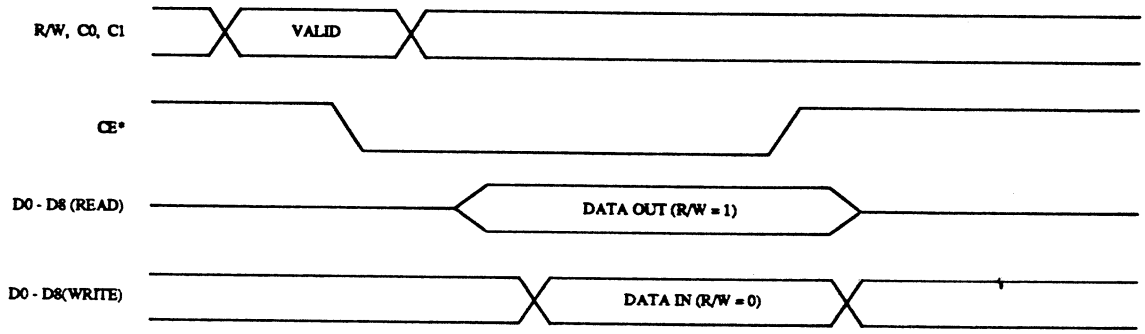


Figure 1. MPU Read/Write Timing.

Circuit Description (continued)

Frame Buffer Interface

To enable pixel data to be transferred from the frame buffer at TTL data rates, multiple pixels are supplied to the Bt496 each input pixel clock cycle. The number of pixels supplied each input cycle depends on the number of bits which each pixel contains. The Bt496 supports pixel depths of 32 (CMYK format), 24, 16 (RGB true color), 8, 4, 2, and 1 (pseudo color). Table 2 summarizes the available pixel formats.

When running in the 4:1 multiplex mode, CMYK/RGB selection is made via the SEL inputs on a pixel-by-pixel basis. Other modes are frame selectable via command register settings.

The Bt496 provides an output clock signal (SCLK*) whose falling edge is used to load input pixel data. The SCLK period is automatically adjusted to occur only as often as required for a given pixel depth and zoom factor.

The Bt496 also provides another output clock (VIDCLK*) which should be used for the generation of the SYNC* and BLANK* inputs. The frequency of VIDCLK* is selectable to be 1/4, 1/8, 1/16, or 1/32 of the pixel clock via command register settings.

SCLK* and VIDCLK* should be immediately redriven by a high-input impedance-inverting driver such as the Signetics 74F1804, whose output is then used to clock the frame buffer serial output data and video timing generator.

For further information on SYNC*, BLANK*, VIDCLK*, SCLK*, and input pixel timings, refer to the Video Generation and Timing Waveforms sections.

Pixel Depth	Multiplex Mode	Pixel Format
32	4:1	CMYK
24	4:1	True-Color RGB, 8-8-8
16	8:1	True-Color RGB, 5-5-5, 6-5-5, 6-6-4
8	16:1	Pseudo Color
4	32:1	Pseudo Color
2	64:1	Pseudo Color
1	128:1	Pseudo Color

Table 2. Pixel Formats Supported.

Circuit Description (continued)

Read Masking

Each CLOCK cycle, pixels are processed by the read mask and command registers. Through the use of the read mask registers, individual pixel inputs may be enabled or disabled for display.

In true-color modes, each pixel is processed by the Red, Green, and Blue Read Mask Registers prior to addressing the color lookup tables.

In pseudo-color modes, each of the RGB read mask registers are separately applied to the same expanded color index prior to addressing the corresponding palette RAM. For example, in 4-bit per pixel mode, if the pixel value is $05, and the RGB read mask registers contain $F3, $0F, and $7C, then the palette entries used would be $01 for red, $05 for green, and $04 for blue.

Pixel Zoom

The Bt496 supports 1x to 16x integer zoom through the use of pixel replication. All pixel inputs are zoomed.

If 2x zooming is specified, the first pixel is output for two clock cycles, followed by the second pixel for two clock cycles, etc. Zooming of 3x is similar, except each pixel is output for 3 clock cycles. Zoom is supported for all bit depths.

The SCLK* period is lengthened appropriately for each of the possible zoom values. If 2x zooming is specified, the SCLK* period would be twice as long as that for 1x zoom; for 3x zoom, the SCLK* period is three times as long as 1x zoom, etc.

Regardless of the zoom value, pixel data is latched only once per SCLK* period.

Pixel Panning

Pixel panning is specified to a four pre-zoomed pixel granularity. Valid pan values depend on the bit per pixel mode. Panning is accomplished partially by delaying Sync and Blank signals beyond their normal position by no more than 32 screen pixels. Panning fields bits used depend on the number of bits per pixel. Table 3 shows the interpretation of the pan select field in command register 1.

Block mode

The Bt496 supports block modes for bit per pixel depths of 1, 2, 4, 8, 16, and 32 bits (Table 4). All pixel inputs are used regardless of pixel depth. Pixel unpacking may be from the MSB or LSB of each pixel port, selectable by command register bit CR03. Pixel mapping for these modes are summarized in Table 5 for MSB unpacking and Table 6 for LSB unpacking.

The SCLK* period is automatically adjusted for different pixel depth. For example, changing the pixel depth from 16 to 8 bits would cause the SCLK* period to be twice as long.

Circuit Description (continued)

Bits per Pixel	CR18–CR14	Number of Pixels Panned
32	(xxxxx) 0 pixels	0
16	(00000) 0 pixels (10000) 4 pixels	0, or 4
8	(00000) 0 pixels (01000) 4 pixels (10000) 8 pixels (11000) 12 pixels	0, 4, 8, or 12
4	(00000) 0 pixels (00100) 4 pixels (01000) 8 pixels . . (11100) 28 pixels	0, 4, 8, 12, 16, 20, 24, or 28
2	(00000) 0 pixels . . . (11110) 60 pixels	0, 4, 8, 12, 16, 20, 24, 28, 32, 36, 40, 44, 48, 52, 56, or 60.
1	(00000) 0 pixels . . . (11111) 124 pixels	0, 4, 8, 12, 16, 20, 24, 28, 32, 36, 40, 44, 48, 52, 56, 60, 64, 68, 72, 76, 80, 84, 88, 92, 96, 100, 104, 108, 112, 116, 120, or 124.

Table 3. Pixel Pan.

Bits per Pixel	Pixels per SCLK*	Colors Displayable
1	128	2
2	64	4
4	32	16
8	16	256
16	8	64K
32	4	16.7M

Table 4. Block Mode Operation.

Circuit Description (continued)

Pixel Port Bits		Bit-Per Pixel Depth					
		32/24	16	8	4	2	1
PA	31–24	Pixel 1 (Black)	Pixel 1	Pixel 1	Pixel 1	Pixel 1	Pixel 1
	23–16	Pixel 1 (Cyan/Red)		Pixel 2	.	.	.
	15–8	Pixel 1 (Magenta/Green)	Pixel 2	Pixel 3	.	.	.
	7–0	Pixel 1 (Yellow/Blue)		Pixel 4	. Pixel 8	. Pixel 16	. Pixel 32
PB	31–24	Pixel 2 (Black)	Pixel 3	Pixel 5	Pixel 9	Pixel 17	Pixel 33
	23–16	Pixel 2 (Cyan/Red)		Pixel 6	.	.	.
	15–8	Pixel 2 (Magenta/Green)	Pixel 4	Pixel 7	.	.	.
	7–0	Pixel 2 (Yellow/Blue)		Pixel 8	. Pixel 16	. Pixel 32	. Pixel 64
PC	31–24	Pixel 3 (Black)	Pixel 5	Pixel 9	Pixel 17	Pixel 33	Pixel 65
	23–16	Pixel 3 (Cyan/Red)		Pixel 10	.	.	.
	15–8	Pixel 3 (Magenta/Green)	Pixel 6	Pixel 11	.	.	.
	7–0	Pixel 3 (Yellow/Blue)		Pixel 12	. Pixel 24	. Pixel 48	. Pixel 96
PD	31–24	Pixel 4 (Black)	Pixel 7	Pixel 13	Pixel 25	Pixel 49	Pixel 97
	23–16	Pixel 4 (Cyan/Red)		Pixel 14	.	.	.
	15–8	Pixel 4 (Magenta/Green)	Pixel 8	Pixel 15	.	.	.
	7–0	Pixel 4 (Yellow/Blue)		Pixel 16	. Pixel 32	. Pixel 64	. Pixel 128

Table 5. Pixel Port Unpacking from MSB (CR03 = 0).

Circuit Description (continued)

Pixel Port Bits		Bit Per Pixel Depth					
		32/24	16	8	4	2	1
PA	31–24	Pixel 1 (Black)	Pixel 2	Pixel 4	Pixel 8	Pixel 16	Pixel 32
	23–16	Pixel 1 (Cyan/Red)		Pixel 3	.	.	.
	15–8	Pixel 1 (Magenta/Green)	Pixel 1	Pixel 2	.	.	.
	7–0	Pixel 1 (Yellow/Blue)		Pixel 1	. Pixel 1	. Pixel 1	. Pixel 1
PB	31–24	Pixel 2 (Black)	Pixel 4	Pixel 8	Pixel 16	Pixel 32	Pixel 64
	23–16	Pixel 2 (Cyan/Red)		Pixel 7	.	.	.
	15–8	Pixel 2 (Magenta/Green)	Pixel 3	Pixel 6	.	.	.
	7–0	Pixel 2 (Yellow/Blue)		Pixel 5	. Pixel 9	. Pixel 17	. Pixel 33
PC	31–24	Pixel 3 (Black)	Pixel 6	Pixel 12	Pixel 24	Pixel 48	Pixel 96
	23–16	Pixel 3 (Cyan/Red)		Pixel 11	.	.	.
	15–8	Pixel 3 (Magenta/Green)	Pixel 5	Pixel 10	.	.	.
	7–0	Pixel 3 (Yellow/Blue)		Pixel 9	. Pixel 17	. Pixel 33	. Pixel 65
PD	31–24	Pixel 4 (Black)	Pixel 8	Pixel 16	Pixel 32	Pixel 64	Pixel 128
	23–16	Pixel 4 (Cyan/Red)		Pixel 15	.	.	.
	15–8	Pixel 4 (Magenta/Green)	Pixel 7	Pixel 14	.	.	.
	7–0	Pixel 4 (Yellow/Blue)		Pixel 13	. Pixel 25	. Pixel 49	. Pixel 97

Table 6. Pixel Port Unpacking from LSB (CR03 = 1).

Circuit Description (continued)

Shift Clock Generation

The Bt496 facilitates the generation of the VRAM shift clock by providing SCLK*. SCLK* should be used to clock the VRAM shift registers which provide pixel data to the Bt496. The ratio of SCLK* to Pixel CLOCK is adjusted so that all pixel input bits are presented once during each SCLK* period. The Bt496 supports zoom concurrently with each of the block modes, requiring a number of different SCLK* periods. For example, with 32-bit per pixel depth and 1x zoom, SCLK* occurs every 4 CLOCK cycles; for 16-bit per pixel depth and 1x zoom or 8-bit per pixel depth and 2x zoom, SCLK* occurs every 8 CLOCK cycles, etc. The SCLK* period my be computed using the following formula:

$$\lambda = 128\ (Z/D)$$

where λ is the SCLK* period (in pixel clock cycles), Z is the zoom factor, and D is the pixel depth. A list of SCLK* periods for all possible combinations of pixel depth and zoom is shown in Table 7.

SCLK* is stopped (in a logical "1" state) during blanking to allow the system to reload the VRAM serial shift registers. System implementations utilizing "midline" transfer may need to insert a VRAM shift clock pulse during blanking time to load the shift register tap address. The system may insert this additional clock without incurring additional gate delays by using a NAND driver for SCLK*. The unused (and normally high) input on the NAND driver may then be used to insert additional SCLK*s. These implementations already present valid pixel information at the end of the blanking period. Therefore, the first SCLK* asserted by the Bt496 should load pixel data. For these implementations, CR08 should be a logical zero. See Figure 2 and the Video Generation section for more information.

Zoom Factor	Pixel Depth					
	32	16	8	4	2	1
1	4	8	16	32	64	128
2	8	16	32	64	128	256
3	12	24	48	96	192	384
4	16	32	64	128	256	512
5	20	40	80	160	320	640
6	24	48	96	192	384	768
7	28	56	112	224	448	896
8	32	64	128	256	512	1024
9	36	72	144	288	576	1152
10	40	80	160	320	640	1280
11	44	88	176	352	704	1408
12	48	96	192	384	768	1536
13	52	104	208	416	832	1664
14	56	112	224	448	896	1792
15	60	120	240	480	960	1920
16	64	128	256	512	1024	2048

Table 7. SCLK Period Table (in Pixel Clock Cycles).*

Circuit Description (continued)

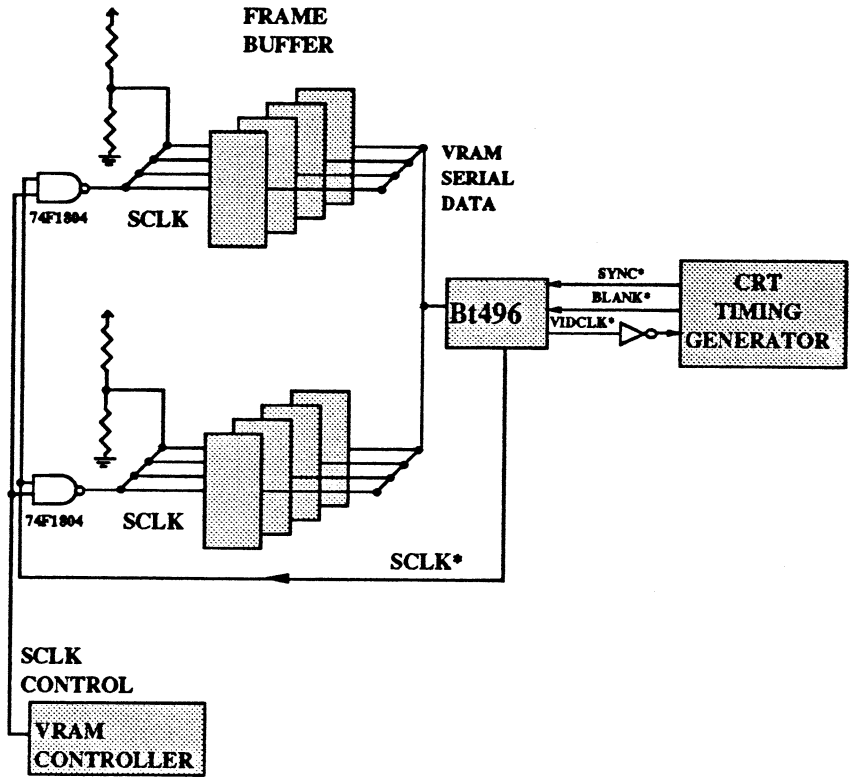


Figure 2. VRAM Serial Clock Interface.

Circuit Description (continued)

Color Modes

The Bt496 processes the pixel types shown in Table 8.

Pixel Type	Description
32 bit, CMYK, True Color	Subtractive color space, 8 bits each of black, cyan, magenta, and yellow for each pixel. Pixels are converted to RGB color space, masked, then are used to address color tables.
24 bit, RGB, 8-8-8, True Color	8 bits each of red, green, and blue for each pixel. Additional 8 supplied pixel bits are ignored. Pixels are RGB masked, then used to address color tables.
16 bit, RGB, 5-5-5, True Color	5 bits each of red, green, and blue for each pixel. Additional supplied pixel bit in MSB of pixel data is ignored. Color component values are left-justified into three 8-bit values.
16 bit, RGB, 6-5-5, True Color	6 bits of red, 5 bits each of green and blue for each pixel. Color component values are left-justified into three 8-bit values.
16 bit, RGB, 6-6-4, True Color	6 bits each of red and green, 4 bits of blue for each pixel. Color component values are left-justified into three 8-bit values.
8 bit, RGB, Pseudo Color	8 bits of supplied pixel data are read masked into three color palette addresses, then used to address color palettes.
4 bit, RGB, Pseudo Color	4 bits of supplied pixel data are right-justified into an 8-bit value, then read masked into 3 color palette addresses.
2 bit, RGB, Pseudo Color	2 bits of supplied pixel data are right-justified into an 8-bit value, then read masked into 3 color palette addresses.
1 bit, RGB, Pseudo Color	1 bit of supplied pixel data is right-justified into an 8-bit value, then read masked into 3 color palette addresses.

Table 8. Pixel Processing Modes.

When 32-bit per pixel depth is specified, each of the 4 pixels may be selected for CMYK or RGB processing on a pixel basis by using the CMYK control inputs. A CMYK pixel has 8 bits each for black, cyan, magenta, and yellow. The mapping of CMYK pixels is shown in Table 9. If the CMYK controls specify RGB processing, 8 bits each for red, green, and blue are used. The remaining 8 bits are ignored.

When 16-bit per pixel depth is specified, 8 pixels are presented each SCLK period. One of three modes for pixel interpretation is selected by CR05 and CR04. The pixel bits supplied for each channel are used as the MSBs of the color table addresses. To maintain the ability to produce a full-scale output without reloading a gamma corrected color lookup table, the supplied bits are duplicated from left (MSB) to right (LSB) into the unspecified address bits of the color table. This value is then subject to read masking. This operation is illustrated by Table 10.

When 8 bits or less per pixel depth are specified, the Bt496 operates in pseudo-color mode, presenting the same 8-bit value to all three color palette tables (subject to read masking).

Pixel Depth	SEL	Pixel Number	Transformation Input Pixel Mapping			
			Black (7–0)	Cyan (7–0)	Magenta (7–0)	Yellow (7–0)
32	1	1	P31A–P24A	P23A–P16A	P15A–P8A	P7A–P0A
		2	P31B–P24B	P23B–P16B	P15B–P8B	P7B–P0B
		3	P31C–P24C	P23C–P16C	P15C–P8C	P7C–P0C
		4	P31D–P24D	P23D–P16D	P15D–P8D	P7D–P0D

Table 9. CMYK Pixel Format.

		Red Address								Green Address								Blue Address							
5:5:5	RAM Address	7	6	5	4	3	2	1	0	7	6	5	4	3	2	1	0	7	6	5	4	3	2	1	0
	Pixel Input	14	13	12	11	10	14	13	12	9	8	7	6	5	9	8	7	4	3	2	1	0	4	3	2
		30	29	28	27	26	30	29	28	25	24	23	22	21	25	24	23	20	19	18	17	16	20	19	18
6:5:5	RAM Address	7	6	5	4	3	2	1	0	7	6	5	4	3	2	1	0	7	6	5	4	3	2	1	0
	Pixel Input	15	14	13	12	11	10	15	14	9	8	7	6	5	9	8	7	4	3	2	1	0	4	3	2
		31	30	29	28	27	26	31	30	25	24	23	22	21	25	24	23	20	19	18	17	16	20	19	18
6:6:4	RAM Address	7	6	5	4	3	2	1	0	7	6	5	4	3	2	1	0	7	6	5	4	3	2	1	0
	Pixel Input	15	14	13	12	11	10	15	14	9	8	7	6	5	4	9	8	3	2	1	0	3	2	1	0
		31	30	29	28	27	26	31	30	25	24	23	22	21	20	25	24	19	18	17	16	19	18	17	16

Table 10. Color Table Address Generation for 16-bit True Color.

Circuit Description (continued)

Pixel unpacking starting from the MSBs may be accomplished by setting CR03 = 0. To unpack starting from the LSBs, set CR03 = 1. Tables 5 and 6 list the pixel input bit mapping for all available pixel depths. Tables 11 and 12 provide greater detail on pixel unpacking for true-color modes, as does Table 13 for pseudo-color modes.

Bit(s) per pixel	SEL Input	CR05–CR04	Bits per Band			Pixel Number	Red(7–0)	Green(7–0)	Blue(7–0)
			Red	Green	Blue				
24	0	XX	8	8	8	1	R7–0 = P23A–P16A	G7–0 = P15A–P8A	B7–0 = P7A–P0A
						2	R7–0 = P23B–P16B	G7–0 = P15B–P8B	B7–0 = P7B–P0B
						3	R7–0 = P23C–P16C	G7–0 = P15C–P8C	B7–0 = P7C–P0C
						4	R7–0 = P23D–P16D	G7–0 = P15D–P8D	B7–0 = P7D–P0D
16	X	00	5	5	5	1	R7–3 = P30A–P26A	G7–3 = P25A–P21A	B7–3 = P20A–P16A
						2	R7–3 = P14A–P10A	G7–3 = P9A–P5A	B7–3 = P4A–P0A
						3	R7–3 = P30B–P26B	G7–3 = P25B–P21B	B7–3 = P20B–P16B
						4	R7–3 = P14B–P10B	G7–3 = P9B–P5B	B7–3 = P4B–P0B
						5	R7–3 = P30C–P26C	G7–3 = P25C–P21C	B7–3 = P20C–P16C
						6	R7–3 = P14C–P10C	G7–3 = P9C–P5C	B7–3 = P4C–P0C
						7	R7–3 = P30D–P26D	G7–3 = P25D–P21D	B7–3 = P20D–P16D
						8	R7–3 = P14D–P10D	G7–3 = P9D–P5D	B7–3 = P4D–P0D
16	X	01	6	5	5	1	R7–2 = P31A–P26A	G7–3 = P25A–P21A	B7–3 = P20A–P16A
						2	R7–2 = P15A–P10A	G7–3 = P9A–P5A	B7–3 = P4A–P0A
						3	R7–2 = P31B–P26B	G7–3 = P25B–P21B	B7–3 = P20B–P16B
						4	R7–2 = P15B–P10B	G7–3 = P9B–P5B	B7–3 = P4B–P0B
						5	R7–2 = P31C–P26C	G7–3 = P25C–P21C	B7–3 = P20C–P16C
						6	R7–2 = P15C–P10C	G7–3 = P9C–P5C	B7–3 = P4C–P0C
						7	R7–2 = P31D–P26D	G7–3 = P25D–P21D	B7–3 = P20D–P16D
						8	R7–2 = P15D–P10D	G7–3 = P9D–P5D	B7–3 = P4D–P0D
16	X	10	6	6	4	1	R7–2 = P31A–P26A	G7–2 = P25A–P20A	B7–4 = P19A–P16A
						2	R7–2 = P15A–P10A	G7–2 = P9A–P4A	B7–4 = P3A–P0A
						3	R7–2 = P31B–P26B	G7–2 = P25B–P20B	B7–4 = P19B–P16B
						4	R7–2 = P15B–P10B	G7–2 = P9B–P4B	B7–4 = P3B–P0B
						5	R7–2 = P31C–P26C	G7–2 = P25C–P20C	B7–4 = P19C–P16C
						6	R7–2 = P15C–P10C	G7–2 = P9C–P4C	B7–4 = P3C–P0C
						7	R7–2 = P31D–P26D	G7–2 = P25D–P20D	B7–4 = P19D–P16D
						8	R7–2 = P15D–P10D	G7–2 = P9D–P4D	B7–4 = P3D–P0D

Table 11. True Color Pixel Formats, MSB Pixel Unpacking (CR03 = 0).

Circuit Description (continued)

Bit(s) per pixel	SEL Input	CR05–CR04	Bits per Band			Pixel Number	Color Palette Addressing		
			Red	Green	Blue		Red(RA7–0)	Green(RA7–0)	Blue(RA7–0)
24	0	XX	8	8	8	1	R7–0 = P23A–P16A	G7–0 = P15A–P8A	B7–0 = P7A–P0A
						2	R7–0 = P23B–P16B	G7–0 = P15B–P8B	B7–0 = P7B–P0B
						3	R7–0 = P23C–P16C	G7–0 = P15C–P8C	B7–0 = P7C–P0C
						4	R7–0 = P23D–P16D	G7–0 = P15D–P8D	B7–0 = P7D–P0D
16	X	00	5	5	5	1	R7–3 = P14A–P10A	G7–3 = P9A–P5A	B7–3 = P4A–P0A
						2	R7–3 = P30A–P26A	G7–3 = P25A–P21A	B7–3 = P20A–P16A
						3	R7–3 = P14B–P10B	G7–3 = P9B–P5B	B7–3 = P4B–P0B
						4	R7–3 = P30B–P26B	G7–3 = P25B–P21B	B7–3 = P20B–P16B
						5	R7–3 = P14C–P10C	G7–3 = P9C–P5C	B7–3 = P4C–P0C
						6	R7–3 = P30C–P26C	G7–3 = P25C–P21C	B7–3 = P20C–P16C
						7	R7–3 = P14D–P10D	G7–3 = P9D–P5D	B7–3 = P4D–P0D
						8	R7–3 = P30D–P26D	G7–3 = P25D–P21D	B7–3 = P20D–P16D
16	X	01	6	5	5	1	R7–2 = P15A–P10A	G7–3 = P9A–P5A	B7–3 = P4A–P0A
						2	R7–2 = P31A–P26A	G7–3 = P25A–P21A	B7–3 = P20A–P16A
						3	R7–2 = P15B–P10B	G7–3 = P9B–P5B	B7–3 = P4B–P0B
						4	R7–2 = P31B–P26B	G7–3 = P25B–P21B	B7–3 = P20B–P16B
						5	R7–2 = P15C–P10C	G7–3 = P9C–P5C	B7–3 = P4C–P0C
						6	R7–2 = P31C–P26C	G7–3 = P25C–P21C	B7–3 = P20C–P16C
						7	R7–2 = P15D–P10D	G7–3 = P9D–P5D	B7–3 = P4D–P0D
						8	R7–2 = P31D–P26D	G7–3 = P25D–P21D	B7–3 = P20D–P16D
16	X	10	6	6	4	1	R7–2 = P15A–P10A	G7–2 = P9A–P4A	B7–4 = P3A–P0A
						2	R7–2 = P31A–P26A	G7–2 = P25A–P20A	B7–4 = P19A–P16A
						3	R7–2 = P15B–P10B	G7–2 = P9B–P4B	B7–4 = P3B–P0B
						4	R7–2 = P31B–P26B	G7–2 = P25B–P20B	B7–4 = P19B–P16B
						5	R7–2 = P15C–P10C	G7–2 = P9C–P4C	B7–4 = P3C–P0C
						6	R7–2 = P31C–P26C	G7–2 = P25C–P20C	B7–4 = P19C–P16C
						7	R7–2 = P15D–P10D	G7–2 = P9D–P4D	B7–4 = P3D–P0D
						8	R7–2 = P31D–P26D	G7–2 = P25D–P20D	B7–4 = P19D–P16D

Table 12. True-Color Pixel Formats, LSB Pixel Unpacking (CR03 = 1).

Circuit Description (continued)

Bit(s) per pixel	Pixel Number	Color Table Addressing (RA7–0)	
		MSB Unpacking (CR03 = 0)	LSB Unpacking (CR03 = 1)
8	1	RA7–0 = P31A–P24A	RA7–0 = P7A–P0A
	2	RA7–0 = P23A–P16A	RA7–0 = P15A–P8A
	3	RA7–0 = P15A–P8A	RA7–0 = P23A–P16A
	4	RA7–0 = P7A–P0A	RA7–0 = P31A–P24A
	5	RA7–0 = P31B–P24B	RA7–0 = P7B–P0B
	.	.	.
	.	.	.
	.	.	.
	16	RA7–0 = P7D–P0D	RA7–0 = P31D–P24D
4	1	RA3–0 = P31A–P28A	RA3–0 = P3A–P0A
	.	.	.
	.	.	.
	.	.	.
	8	RA3–0 = P3A–P0A	RA3–0 = P31A–P28A
	9	RA3–0 = P31B–P28B	RA3–0 = P3B–P0B
	.	.	.
	.	.	.
	.	.	.
	32	RA3–0 = P3D–P0D	PA3–0 = P31D–P28D
2	1	RA1–0 = P31A–P30A	RA1–0 = P1A–P0A
	.	.	.
	.	.	.
	.	.	.
	16	RA1–0 = P1A–P0A	RA1–0 = P31A–P30A
	17	RA1–0 = P31B–P30B	RA1–0 = P1B–P0B
	.	.	.
	.	.	.
	.	.	.
	64	RA1–0 = P1D–P0D	RA1–0 = P31D–P30D
1	1	RA0 = P31A	RA0 = P0A
	.	.	.
	.	.	.
	.	.	.
	32	RA0 = P0A	RA0 = P31A
	33	RA0 = P31B	RA0 = P0B
	.	.	.
	.	.	.
	.	.	.
	128	RA0 = P0D	RA0 = R31D

Table 13. Pseudo-Color Pixel Formats.

Circuit Description (continued)

CMYK-to-RGB Conversion

For printer prepress applications, image data is generally specified in the subtractive color space, i.e., cyan, magenta, yellow, and black (CMYK). The four dimensional CMYK space has an extra degree of freedom used to control the amount of black. One way the extra degree of freedom may be used is in the control of the gray component replacement (GCR), which governs the amount of the gray component of a CMY color that is replaced by black. The specification of a GCR value locks in the extra degree of freedom and establishes a one-to-one mapping between RGB and CMYK.

One algorithm which may be used to convert from RGB (r, g, b in the range {0,1}) to CMYK (c, m, y, k in the range {0,1}) is as follows:

$$c' = 1 - r$$
$$m' = 1 - g$$
$$y' = 1 - b$$

$$V_{min} = \min(c', m', y')$$
$$V_{max} = \max(c', m', y')$$
$$V_{rem} = V_{min}\, G^{\alpha}(1 - (V_{max} - V_{min}))^{\beta}$$

$$c = c' - V_{rem}$$
$$m = m' - V_{rem}$$
$$y = y' - V_{rem}$$
$$k = V_{min} G$$

In these equations, G is the GCR value in the range [0, 1]. The constant α is used to control the relationship between the black level and the CMY levels as a function of GCR. Current literature suggests a value for α of five. The constant β is used to give further control of the K to CMY relationship as a function of saturation. The value for β is considered subjective, but two may be used as a starting point.

The inverse of this color space conversion algorithm has been implemented in the Bt496:

$$V_{min} = \min(c, m, y)$$
$$V_{max} = \max(c, m, y)$$
$$V_{rem} = V\min\, G^{\alpha-1}[1 - (V_{max} - V_{min})]^{\beta}$$
$$V_{rem} = \min\,(V_{rem}, 1 - V_{max})]$$

$$r = 1 - c - V_{rem}$$
$$g = 1 - m - V_{rem}$$
$$b = 1 - y - V_{rem}$$

The non-linear portion of the Vrem computation is implemented in the Bt496 by using a lookup table, indexed by (Vmax–Vmin).

A GCR value in the range 0.6–0.8 produces a black with the highest density, and therefore offers the greatest dynamic range. Although the maximum GCR value of 1.0 provides the best economy of the more expensive colored printing inks, the resulting printed images suffer from reduced shadow density and increased sensitivity to registration errors between the black and colored printers, causing light outlines to appear on dark picture elements.

Circuit Description (continued)

A simplifed block diagram of the CMYK-to-RGB transformation is shown in Figure 3. The pipeline registers are not shown for simplicity.

The 256 x 8 RAM is used to evaluate the non-linear portion of equation $V_{rem} = Vmin\ G^{\alpha-1}[1-(V_{max} - V_{min})]^{\beta}$. This RAM should be loaded by the MPU with the values for the function

$f(x) = G^{(\alpha-1)}(1-x)^{\beta}$

Sample RAM contents for a GCR of .7, alpha of 5, and beta of 2 are shown in Table 14.

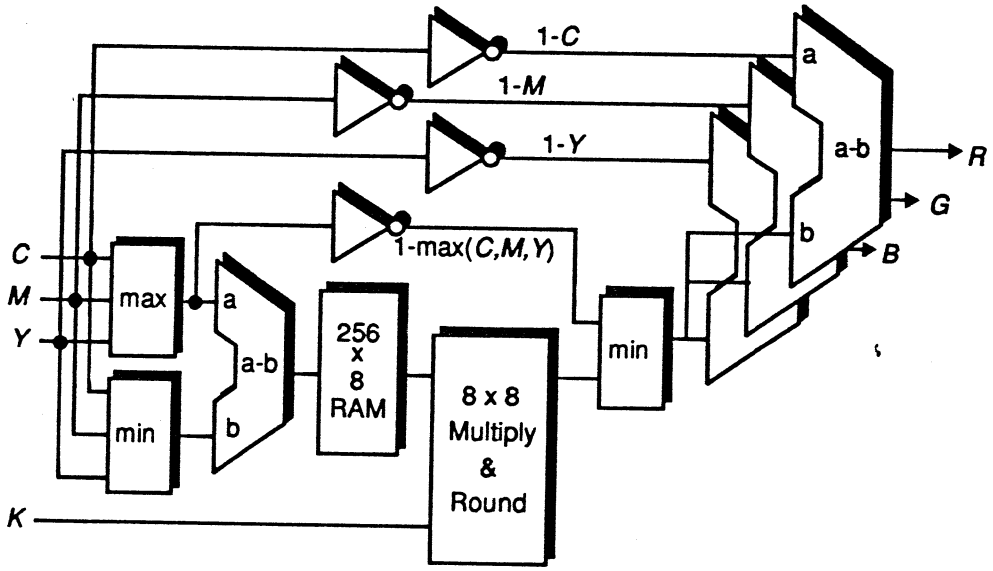


Figure 3. CMYK–to–RGB Conversion.

Circuit Description (continued)

Entry	Value	Entry	Value
$00	$3D	$4A–$4C	$1E
$01–$02	$3C	$4D–$4F	$1D
$03–$04	$3B	$50–$52	$1C
$05–$06	$3A	$53–$55	$1B
$07–$08	$39	$56–$58	$1A
$09–$0B	$38	$59–$5C	$19
$0C–$0D	$37	$5D–$5F	$18
$0E–$0F	$36	$60–$62	$17
$10–$11	$35	$63–$66	$16
$12–$13	$34	$67–$69	$15
$14–$16	$33	$6A–$6D	$14
$17–$18	$32	$6E–$70	$13
$19–$1A	$31	$71–$74	$12
$1B–$1D	$30	$75–$78	$11
$1E–$1F	$2F	$79–$7C	$10
$20–$21	$2E	$7D–$80	$0F
$22–$24	$2D	$81–$85	$0E
$25–$26	$2C	$86–$89	$0D
$27–$29	$2B	$8A–$8E	$0C
$2A–$2B	$2A	$8F–$92	$0B
$2C–$2E	$29	$93–$97	$0A
$2F–$30	$28	$98–$9D	$09
$31–$33	$27	$9E–$A2	$08
$34–$36	$26	$A3–$A8	$07
$37–$38	$25	$A9–$AF	$06
$39–$3B	$24	$B0–$B6	$05
$3C–$3E	$23	$B7–$BD	$04
$3F–$40	$22	$BE–$C6	$03
$41–$43	$21	$C7–$D0	$02
$44–$46	$20	$D1–$DE	$01
$47–$49	$1F	$DF–$FF	$00

Table 14. CMYK-to-RGB Transformation RAM Contents for GCR = .7, $\alpha = 5$, $\beta = 2$.

Circuit Description (continued)

Video Generation

Video input pixel timing is shown in Figure 4.

The VIDCLK* output is a free-running clock typically used for clocking the display timing generator. The MPU controls the period of *VIDCLK** by setting bits CR07 and CR06 in command register 0. VIDCLK* may be CLOCK/4, CLOCK/8, CLOCK/16, or CLOCK/32. SYNC and BLANK information are latched with each falling edge of VIDCLK* and inserted into the pipelined pixel stream at the appropriate time. No relationship should be considered to exist between the SYNC*/BLANK* signals and the incoming pixel data. The SYNC* and BLANK* inputs are used to provide the RAMDAC with timing information; i.e., blank duration, front porch, sync duration, and back porch.

Since VIDCLK* generally runs with a different period from SCLK*, SYNC* and BLANK* may be provided at a higher resolution than the number of pixels loaded each SCLK*. Thus, not all pixels loaded with the last active SCLK* will necessarily be displayed on the screen. However, on the leading edge of the scan line, the first pixel loaded by SCLK* will always be the first pixel displayed.

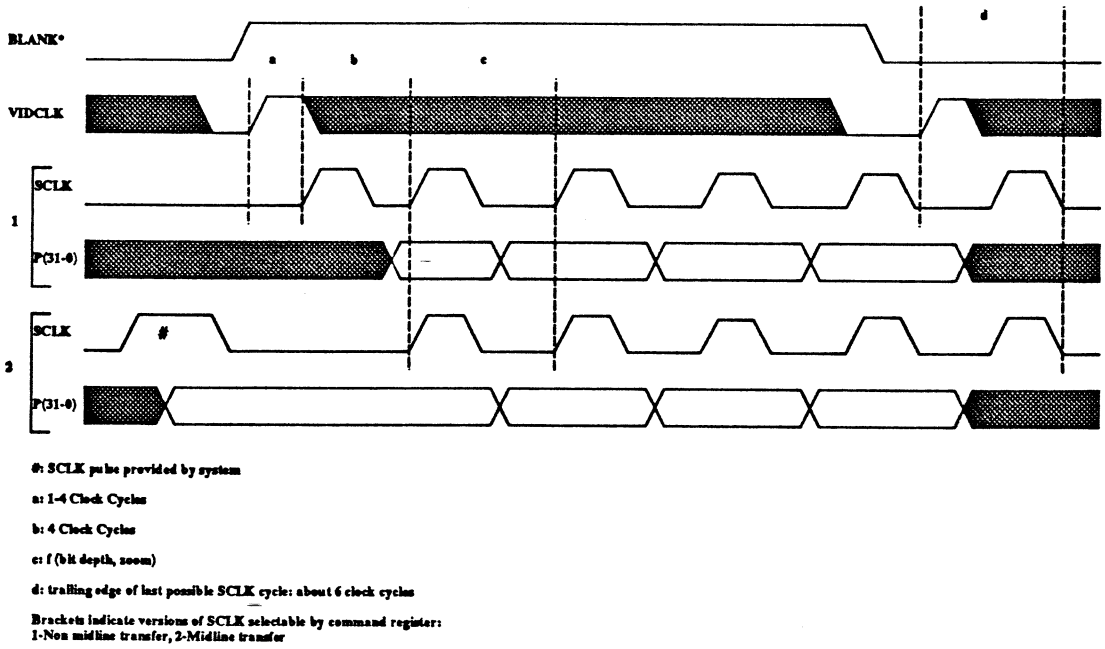
The SYNCOUT* signal follows the SYNC* input by the number of pipeline stages internal to the Bt496. It is positioned with pixel resolution, and should be used to derive the SYNC signal for displays requiring separate TTL sync. The RAMDAC pipeline delay present from the input pixel port to the output is constant for a given pixel pan amount.

It is recommended that MPU operations requiring display synchronization to blank and/or sync (e.g., command register updates) read command register CR21 AND CR20 to synchronize. Using the SYNC* and/or BLANK* input signals will not provide accurate synchronization due to the size and variability of the Bt496's pipeline depth.

The SYNC* and BLANK* inputs, pipelined to maintain synchronization with the pixel data, add appropriately weighted currents to the analog outputs. This produces the specific output levels required for video applications, as illustrated in Figures 5 and 6. Command register 2 specifies whether a 0 IRE or 7.5 IRE blanking pedestal is to be generated, and whether or not sync information is to be encoded on the video output. A 0 IRE pedestal will force the black level and the blank level to be the same.

The varying output current from the D/A converters produces a corresponding voltage level, which is used to drive the CRT monitor. Tables 15 and 16 detail how the SYNC* and BLANK* inputs modify the output levels.

The D/A converters on the Bt496 use a segmented architecture in which bit currents are routed to either the current output or GND by a sophisticated decoding scheme. This architecture eliminates the need for precision component ratios and greatly reduces the switching transients associated with turning current sources on or off. Monotonicity and low glitch are guaranteed by using identical current sources and current steering their outputs. An on-chip operational amplifier stabilizes the full-scale output current against temperature and power supply variations.

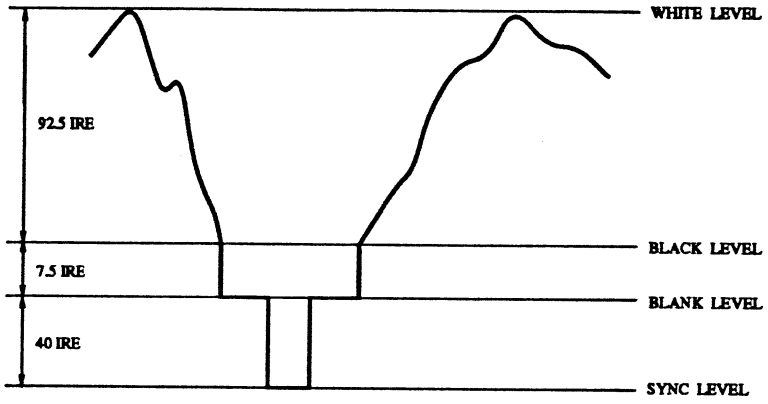


Note: For clarity, this figure shows SCLK & VIDCLK outputs inverted .

Figure 4. Video Input Pixel Timing.

Circuit Description (continued)

RED, BLUE		GREEN	
MA	V	MA	V
19.05	0.714	26.67	1.000
1.44	0.054	9.05	0.340
0.00	0.000	7.62	0.286
		0.00	0.000



Note: 75 Ω doubly-terminated load, RSET = 562 Ω (0.5%), VREF = 1.235 V. Blank pedestal = 7.5 IRE. RS-343A levels and tolerances assumed on all levels.

Figure 5. Composite Video Output Waveform (SETUP = 7.5 IRE).

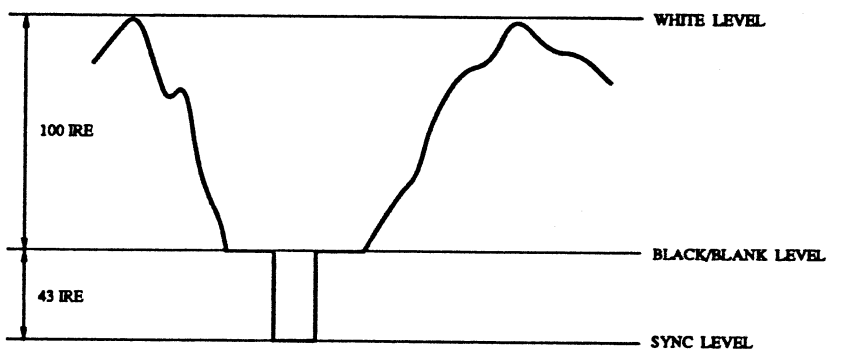
Description	IOG (mA)	IOR, IOB (mA)	SYNC*	BLANK*	DAC Input Data
WHITE	26.67	19.05	1	1	$FF
DATA	data + 9.05	data + 1.44	1	1	data
DATA–SYNC	data + 1.44	data + 1.44	0	1	data
BLACK	9.05	1.44	1	1	$00
BLACK–SYNC	1.44	1.44	0	1	$00
BLANK	7.62	0	1	0	$xx
SYNC	0	0	0	0	$xx

Note: Typical with full-scale IOG = 26.67 mA. RSET = 562 Ω (0.5%), VREF = 1.235 V. Blank pedestal = 7.5 IRE.

Table 15. Video Output Truth Table (SETUP = 7.5 IRE).

Circuit Description (continued)

RED, BLUE		GREEN	
MA	V	MA	V
18.60	0.698	26.67	1.000
0.00	0.000	8.05	0.302
		0.00	0.000



Note: 75 Ω doubly-terminated load, RSET = 530 Ω (0.5%), VREF = 1.235 V. Blank pedestal = 0 IRE. RS-343A levels and tolerances assumed on all levels.

Figure 6. Composite Video Output Waveform (SETUP = 0 IRE).

Description	IOG (mA)	IOR, IOB (mA)	SYNC*	BLANK*	DAC Input Data
WHITE	26.67	18.60	1	1	$FF
DATA	data + 8.05	data	1	1	data
DATA–SYNC	data	data	0	1	data
BLACK	8.05	0	1	1	$00
BLACK–SYNC	0	0	0	1	$00
BLANK	8.05	0	1	0	$xx
SYNC	0	0	0	0	$xx

Note: Typical with full-scale IOG = 26.67 mA. RSET = 530 Ω (0.5%), VREF = 1.235 V. Blank pedestal = 0 IRE.

Table 16. Video Output Truth Table (SETUP = 0 IRE).

Internal Registers

Command Register 0

This register is located at internal address $002, and may be written or read by the MPU at any time.

Bit(s)	Function	Description
CR08	SCLK* Control (0) Extra pulse not needed (1) Extra pulse needed	This bit specifies whether the first SCLK* pulse after a blanked time is needed (logical one) to read the first pixel item. A logical zero indicates that the system has externally provided the first VRAM shift clock and the Bt9046 may latch valid pixel data with the first SCLK*.
CR07, CR06	VIDCLK Select (00) CLOCK/4 (01) CLOCK/8 (10) CLOCK/16 (11) CLOCK/32	Video Clock Output Period Select
CR05,CR04	16 bit per pixel true color mode (00) 5 red, 5 green, 5 blue (01) 6 red, 5 green, 5 blue (10) 6 red, 6 green, 4 blue (11) reserved	These bits specify the interpretation of the pixel data when processing in 16 bit per pixel mode. These bits are ignored in other modes.
CR03	Pixel Unpacking Mode (0) Unpacked from MSB (1) Unpacked from LSB	This bit specifies the unpacking mode for pixels having less than 32 bits per pixel. When this bit is a logical 0, pixels are unpacked starting from PA bit 31. When this bit is a logical 1, pixels are unpacked starting from PA bit 0.
CR02–CR00	Block Mode Select (11X) – Reserved (101) – 32 bits per pixel (100) – 16 bits per pixel (011) – 8 bits per pixel (010) – 4 bits per pixel (001) – 2 bits per pixel (000) – 1 bit per pixel	These bits specify whether the pixel data is input as 1, 2, 4, 8, 16, or 32 bits per pixel.

Internal Registers (continued)

Command Register 1

This register is located at internal address $003 and may be written or read by the MPU at any time.

Bit(s)	Function	Description
CR18–CR14	Pan Select	These bits specify the pan amount. Pan may be specified to a 4-pixel pre-zoomed boundary. Valid pan ranges depend on the bits per pixel setting. See Table 3 in the pan description values for a bit definition.
CR13, CR10	Zoom Factor (0000) – 1x . . . (1111) – 16x	These bits specify the zoom amount. For 2x zoom, the first pixel is output for two clock cycles, followed by the second pixel for two clock cycles, etc. For 3x zoom, the first pixel is output for 3 cycles, the second pixel for 3 cycles, etc.

Internal Registers (continued)

Command Register 2

This register is located at internal address $004, and may be written or read by the MPU at any time.

Bit(s)	Function	Description
CR28	Reserved	
CR27	Sync enable (0) disable sync (1) enable sync	This bit specifies whether sync information is to be output onto IOG (logical one) or not (logical zero). A logical zero on this bit will disable CR21 and the SYNCOUT* pin.
CR26	Pedestal enable (0) 0 IRE pedestal (1) 7.5 IRE pedestal	This bit specifies whether a 0 or 7.5 IRE blanking pedestal is to be generated on the video outputs. 0 IRE specifies that the black and blank levels are the same.
CR25–CR22	Reserved	
CR21	SYNC	This bit tracks the SYNC level on IOG and specifies whether SYNC is being output onto IOG (logical one) or not (logical zero). This bit is read only. Any attempts to write to this bit will be ignored.
CR20	BLANK	This bit tracks the BLANK level and specifies whether BLANK is being output onto IOR, IOG, & IOB (logical one) or not (logical zero). This bit should be used to synchronize MPU events to display blanking time. This bit is read only. Any attempts to write to this bit will be ignored.

Internal Registers (continued)

ID Register ($000)

This 8-bit register may be read by the MPU to determine the type of RAMDAC being used in the system. The value is different for each RAMDAC. For the Bt496, the value read by the MPU will be $081. Data written to this register is ignored.

Revision Register ($001)

This 9-bit register may be read by the MPU to determine the revision level of the Bt496. The upper 4 bits indicate revisions while the lower 5 bits should be ignored. The current revision is $A. Any data written to this register is ignored.

Pixel Read Mask Registers ($006-$008)

The 24-bit pixel read mask register is used to enable (logical one) or disable (logical zero) a bit plane from addressing the color palette RAM. Each register bit is logically ANDed with the corresponding bit plane input. This register may be written to or read by the MPU at any time and is not initialized. D0 corresponds to P0.

Red, Green, and Blue Signature Registers ($00C-$00E)

The test register must be enabled in order to use the Signature or the Data Strobe mode. Refer to the test register ($00F) description. During normal operation, disable the test registers for power consideration.

Signature Mode

These three 9-bit signature registers (one each for red, green, and blue) may be read by the MPU while BLANK* is a logical zero. While BLANK* is a logical one, the signatures are being acquired. The MPU may read from or write to the signature registers while BLANK* is a logical zero to load the seed values.

By loading a test display into the frame buffer, a deterministic value for the red, green, and blue signature registers will be read from these registers if all circuitry is working properly. Refer to the Application Information Test Register section for more information.

Data Strobe Mode

If "data strobe testing" is selected, the operation of the signature registers changes slightly. Rather than determining the signatures, they capture red, green, and blue data being presented to the three DACs. Refer to the Application information Test Register section for more information.

Internal Registers (continued)

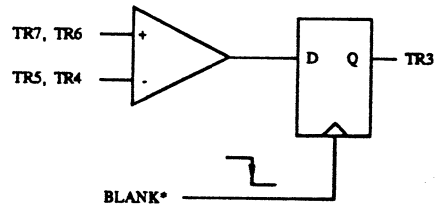
Test Register ($00F)

This 9-bit register is used for testing the Bt496.

Bits	Function	Description
TR8	Test mode enable (0) Disable test mode (1) Enable test mode	The test register must be enabled in order to select signature analysis or data strobe test modes. For normal operation, disable test registers for power considerations.
TR7–TR3	Comparator	Described below.
TR2–TR1	Continuity result	These pins are used to assist in an ATE continuity test. Their function is not defined at this time.
TR0	Test mode select (0) signature analysis test (1) data strobe test	This bit determines the method of high-speed test used. The signature analysis registers are used to hold the test result for both test modes.

TR7–TR3 are used to compare the analog RGB outputs to each other and to a 150 mV reference. This enables the MPU to determine whether the CRT monitor is connected to the analog RGB outputs or not, and whether the DACs are functional.

TR7	TR6	TR5	TR4	TR3
red select	green select	blue select	150 mV ref. select	result



TR7–TR4		If TR3 = 1	If TR3 = 0
0000	normal operation	—	—
1010	red DAC compared to blue DAC	red > blue	blue > red
1001	red DAC compared to 150 mV reference	red > 150 mV	red < 150 mV
0110	green DAC compared to blue DAC	green > blue	blue > green
0101	green DAC compared to 150 mV reference	green > 150 mV	green < 150 mV

The table above lists the valid comparison combinations. A logical one enables that function to be compared; the result is TR3. For normal operation, TR7–TR3 must be a logical zero.

Pin Descriptions

Pin Name	Description
BLANK*	Composite blank control input (TTL compatible). A logical zero drives the analog output to the blanking level, as illustrated in Tables 14 and 15. It is latched on the falling edge of VIDCLK*.
SYNC*	Composite sync control input (TTL compatible). A logical zero typically switches off a 40 IRE current source on the IOG output (see Figures 4 and 5). SYNC* does not override any other control or data input, as shown in Tables 14 and 15; therefore, it should be asserted only during the blanking interval. SYNC* is latched on the falling edge of VIDCLK* .
SYNCOUT*	Composite sync control output (TTL compatible). This output is typically used for generating the composite sync signal for CRTs requiring separate sync.
SCLK*	VRAM Shift Clock output (TTL compatible). The falling edge of this output is used to clock the input pixel data. The Shift Clock frequency is a function of the number of bits per pixel and the zoom factor. An external inverting buffer should be used for driving the pixel VRAM serial clock inputs. Waveform is active (low) for 4 CLOCK cycles, then inactive (high) as required to achieve cycle time. The shift Clock is automatically stopped/started for blanked intervals.
VIDCLK*	Video Clock output (TTL compatible). This output is CLOCK divided by 4, 8, 16, or 32, as programmed in command register 0. Sync and Blank inputs are latched by this clock. 50/50 duty cycle.
P31–P0 {A–D}	Pixel inputs (TTL compatible). These inputs are used to specify color palette locations in RGB or pseudo-color modes. In CMYK mode, these pixels are converted to RGB color space, then used to specify which locations of the color palette RAM are used to provide color information. These inputs are latched on the falling edge of SCLK*.
SEL(A-D)	SEL mode select inputs (TTL compatible). These inputs specify the interpretation of the pixels supplied on the pixel inputs. A high level causes the corresponding pixel inputs to be interpreted as CMYK data. A low level indicates RGB pixel data. These inputs are valid only in 32-bit per pixel mode. These inputs are latched in the falling edge of SCLK*.
IOR, IOG, IOB	Red, green, and blue current outputs. These high-impedance current sources are capable of directly driving a doubly-terminated 75-Ω coaxial cable (Figure 6). Each of these outputs, whether used or not, should have the same output load.
VAA	Analog power. All VAA pins must be connected.
GND	Analog ground. All GND pins must be connected.
COMP	Compensation pin. This pin provides compensation for the internal reference amplifier. A 0.1 µF ceramic capacitor must be connected between this pin and VAA (Figure 6). Connecting the capacitor to VAA rather than to GND provides the highest possible power supply noise rejection. The COMP capacitor must be as close to the device as possible to keep lead lengths to an absolute minimum and maximize the capacitor's self-resonant frequency to be greater than the LD* frequency.
FS ADJUST	Full-scale adjust control. A resistor (RSET) connected between this pin and GND controls the magnitude of the full-scale video signal (Figure 6). Note that the IRE relationships in Figures 4 and 5 are maintained, regardless of the full-scale output current. The relationship between RSET and the full-scale output current on IOG is: RSET (Ω) = K1 * VREF (V) / IOG (mA) The full-scale output current on IOR and IOB for a given RSET is: IOR, IOB (mA) = K2 * VREF (V) / RSET (Ω)

Pin Descriptions (continued)

Pin Name	Description

where K1 and K2 are defined as:

Setup	IOG	IOR, IOB
7.5 IRE	K1 = 11,294	K2 = 8,067
0 IRE	K1 = 10,684	K2 = 7,457

VREF — Voltage reference input. An external voltage reference circuit must supply this input with a 1.235 V (typical) reference. The use of a resistor network to generate the reference is not recommended, as any low-frequency power supply noise on VREF will be directly coupled onto the analog outputs. A 0.1 µF ceramic capacitor must be used to decouple this input to VAA, as shown in Figure 13. The decoupling capacitor must be as close to the device as possible to keep lead lengths to an absolute minimum.

CLOCK, CLOCK* — Clock inputs. These differential clock inputs are designed to be driven by ECL logic configured for single supply (+5 V) operation. The clock rate is typically the pixel clock rate of the system.

CE* — Chip enable control input (TTL compatible). This input must be a logical zero to enable data to be written to or read from the device. During write operations, data is internally latched on the rising edge of CE*. Care should be taken to avoid glitches on this edge-triggered input.

R/W — Read/write control input (TTL compatible). To write data to the device, both CE* and R/W must be a logical zero. To read data from the device, CE* must be a logical zero and R/W must be a logical one. R/W is latched on the falling edge of CE*.

C0, C1 — Command control inputs (TTL compatible). C0 and C1 specify the type of read or write operation being performed, as illustrated in Table 1. They are latched on the falling edge of CE*.

D0–D8 — Data bus (TTL compatible). Data is transferred into and out of the device over this 9-bit bidirectional data bus. D0 is the least significant bit.

Pin Descriptions (continued)—207-pin PGA Package

Signal	Pin Number
BLANK*	N14
SYNC*	P15
SYNCOUT*	C3
CLOCK	T16
CLOCK*	T17
SCLK*	S2
VIDCLK*	R15
SELA	B14
SELB	A15
SELC	A14
SELD	C13
P0A	D5
P0B	C4
P0C	A2
P0D	B3
P1A	C2
P1B	B1
P1C	D3
P1D	E4
P2A	D2
P2B	C1
P2C	D1
P2D	E3
P3A	F2
P3B	E2
P3C	G3
P3D	F3
P4A	G1
P4B	G2
P4C	H3
P4D	H4
P5A	H1
P5B	H2
P5C	J2
P5D	J4
P6A	J3
P6B	J1
P6C	K1
P6D	K2
P7A	K3
P7B	K4
P7C	L2
P7D	L1
P8A	M3
P8B	L3
P8C	N2
P8D	M2

Signal	Pin Number
P9A	N3
P9B	P1
P9C	R1
P9D	P2
P10A	N4
P10B	P3
P10C	T1
P10D	R2
P11A	R4
P11B	P5
P11C	T4
P11D	U3
P12A	U4
P12B	R5
P12C	P6
P12D	T5
P13A	R6
P13B	U5
P13C	T6
P13D	P7
P14A	R7
P14B	U6
P14C	U7
P14D	T7
P15A	R8
P15B	P8
P15C	U8
P15D	T8
P16A	T9
P16B	P9
P16C	R9
P16D	U9
P17A	U10
P17B	T10
P17C	R10
P17D	P10
P18A	T11
P18B	U11
P18C	P11
P18D	R11
P19A	T12
P19B	U12
P19C	U13
P19D	R12

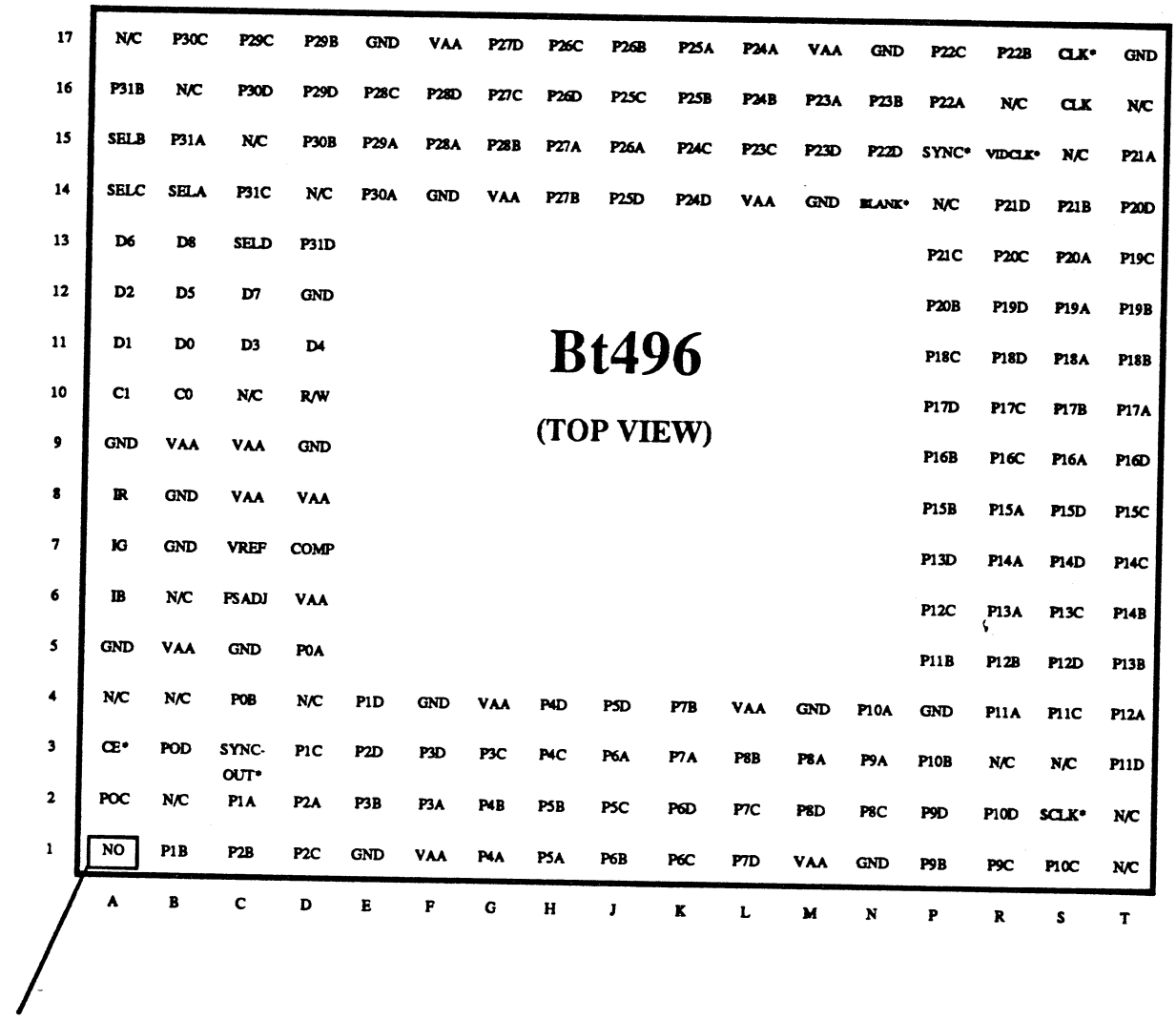
Signal	Pin Number
P20A	T13
P20B	P12
P20C	R13
P20D	U14
P21A	U15
P21B	T14
P21C	P13
P21D	R14
P22A	P16
P22B	R17
P22C	P17
P22D	N15
P23A	M16
P23B	N16
P23C	L15
P23D	M15
P24A	L17
P24B	L16
P24C	K15
P24D	K14
P25A	K17
P25B	K16
P25C	J16
P25D	J14
P26A	J15
P26B	J17
P26C	H17
P26D	H16
P27A	H15
P27B	H14
P27C	G16
P27D	G17
P28A	F15
P28B	G15
P28C	E16
P28D	F16
P29A	E15
P29B	D17
P29C	C17
P29D	D16

Pin Descriptions (continued)—207-pin PGA Package

Signal	Pin Number
P30A	E14
P30B	D15
P30C	B17
P30D	C16
P31A	B15
P31B	A16
P31C	C14
P31D	D13
IR	A8
IG	A7
IB	A6
VAA	B5
VAA	B9
VAA	C8
VAA	C9
VAA	D6
VAA	D8
VAA	F1
VAA	F17
VAA	G4
VAA	G14
VAA	L4
VAA	L14
VAA	M1
VAA	M17
GND	A5
GND	A9
GND	B7
GND	B8
GND	C5
GND	D9
GND	D12
GND	E1
GND	E17
GND	F4
GND	F14
GND	M4
GND	M14
GND	N1
GND	N17
GND	P4
GND	U17
COMP	D7
FSADJ	C6
VREF	C7
CE*	A3
R/W	D10
C1	A10
C0	B10

Signal	Pin Number
D0	B11
D1	A11
D2	A12
D3	C11
D4	D11
D5	B12
D6	A13
D7	C12
D8	B13

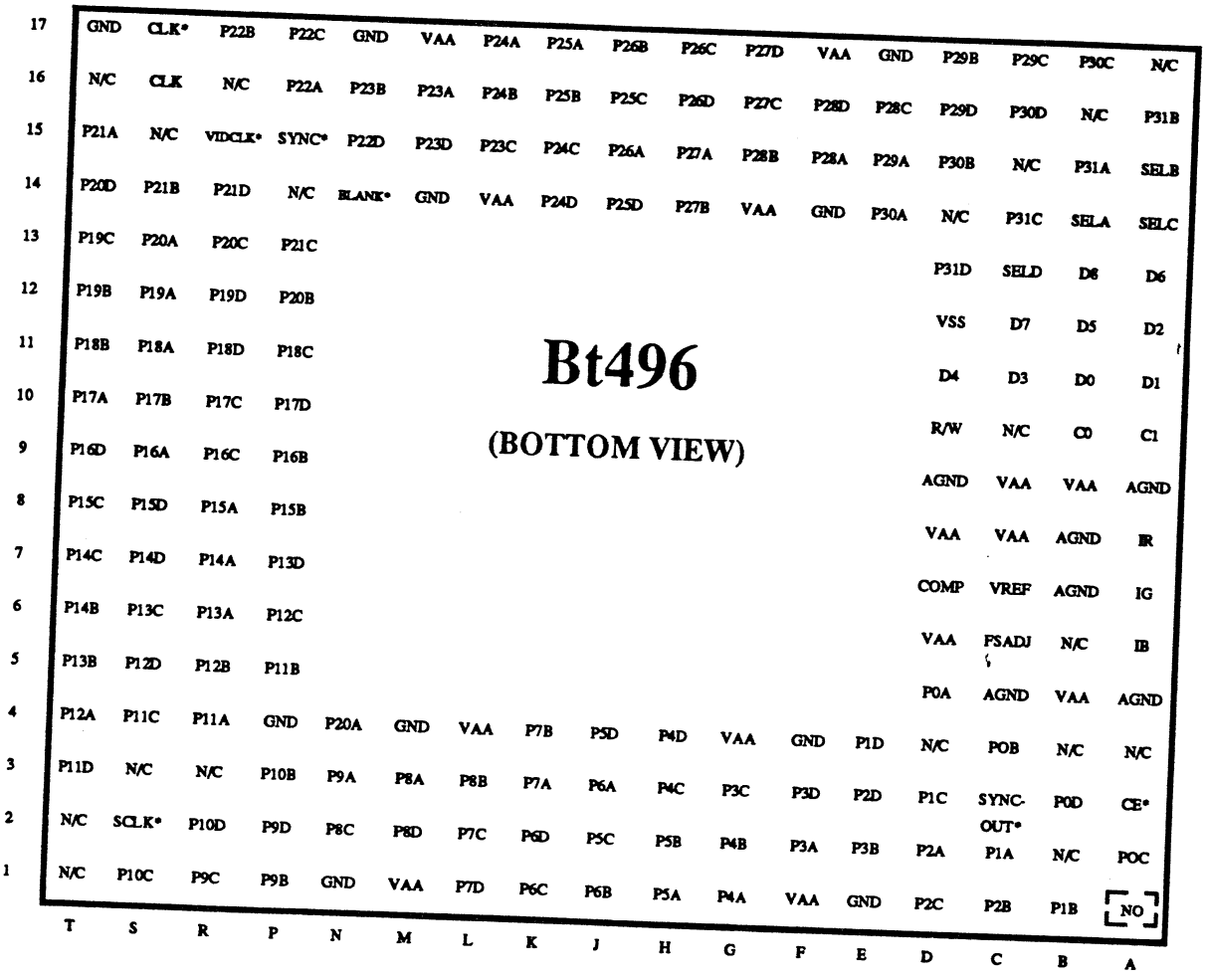
Pin Descriptions (continued)—207-pin PGA Package



	A	B	C	D	E	F	G	H	J	K	L	M	N	P	R	S	T
17	N/C	P30C	P29C	P29B	GND	VAA	P27D	P26C	P26B	P25A	P24A	VAA	GND	P22C	P22B	CLK*	GND
16	P31B	N/C	P30D	P29D	P28C	P28D	P27C	P26D	P25C	P25B	P24B	P23A	P23B	P22A	N/C	CLK	N/C
15	SELB	P31A	N/C	P30B	P29A	P28A	P28B	P27A	P26A	P24C	P23C	P23D	P22D	SYNC*	VIDCLK*	N/C	P21A
14	SELC	SELA	P31C	N/C	P30A	GND	VAA	P27B	P25D	P24D	VAA	GND	BLANK*	N/C	P21D	P21B	P20D
13	D6	D8	SELD	P31D										P21C	P20C	P20A	P19C
12	D2	D5	D7	GND										P20B	P19D	P19A	P19B
11	D1	D0	D3	D4										P18C	P18D	P18A	P18B
10	C1	C0	N/C	R/W										P17D	P17C	P17B	P17A
9	GND	VAA	VAA	GND										P16B	P16C	P16A	P16D
8	IR	GND	VAA	VAA										P15B	P15A	P15D	P15C
7	IG	GND	VREF	COMP										P13D	P14A	P14D	P14C
6	IB	N/C	RSADJ	VAA										P12C	P13A	P13C	P14B
5	GND	VAA	GND	P0A										P11B	P12B	P12D	P13B
4	N/C	N/C	P0B	N/C	P1D	GND	VAA	P4D	P5D	P7B	VAA	GND	P10A	GND	P11A	P11C	P12A
3	CE*	POD	SYNC-OUT*	P1C	P2D	P3D	P3C	P4C	P6A	P7A	P8B	P8A	P9A	P10B	N/C	N/C	P11D
2	POC	N/C	P1A	P2A	P3B	P3A	P4B	P5B	P5C	P6D	P7C	P8D	P8C	P9D	P10D	SCLK*	N/C
1	NO	P1B	P2B	P2C	GND	VAA	P4A	P5A	P6B	P6C	P7D	VAA	GND	P9B	P9C	P10C	N/C

alignment marker (on top)
No pin at location A1

Pin Descriptions—207-pin PGA Package



	T	S	R	P	N	M	L	K	J	H	G	F	E	D	C	B	A
17	GND	CLK*	P22B	P22C	GND	VAA	P24A	P25A	P26B	P26C	P27D	VAA	GND	P29B	P29C	P30C	N/C
16	N/C	CLK	N/C	P22A	P23B	P23A	P24B	P25B	P25C	P26D	P27C	P28D	P28C	P29D	P30D	N/C	P31B
15	P21A	N/C	VIDCLK*	SYNC*	P22D	P23D	P23C	P24C	P26A	P27A	P28B	P28A	P29A	P30B	N/C	P31A	SELB
14	P20D	P21B	P21D	N/C	BLANK*	GND	VAA	P24D	P25D	P27B	VAA	GND	P30A	N/C	P31C	SELA	SELC
13	P19C	P20A	P20C	P21C										P31D	SELD	D8	D6
12	P19B	P19A	P19D	P20B										VSS	D7	D5	D2
11	P18B	P18A	P18D	P18C										D4	D3	D0	D1
10	P17A	P17B	P17C	P17D										R/W	N/C	C0	C1
9	P16D	P16A	P16C	P16B										AGND	VAA	VAA	AGND
8	P15C	P15D	P15A	P15B										VAA	VAA	AGND	IR
7	P14C	P14D	P14A	P13D										COMP	VREF	AGND	IG
6	P14B	P13C	P13A	P12C										VAA	FSADJ	N/C	IB
5	P13B	P12D	P12B	P11B										P0A	AGND	VAA	AGND
4	P12A	P11C	P11A	GND	P20A	GND	VAA	P7B	P5D	P4D	VAA	GND	P1D	N/C	POB	N/C	N/C
3	P11D	N/C	N/C	P10B	P9A	P8A	P8B	P7A	P6A	P4C	P3C	P3D	P2D	P1C	SYNC-OUT*	P0D	CE*
2	N/C	SCLK*	P10D	P9D	P8C	P8D	P7C	P6D	P5C	P5B	P4B	P3A	P3B	P2A	P1A	N/C	POC
1	N/C	P10C	P9C	P9B	GND	VAA	P7D	P6C	P6B	P5A	P4A	VAA	GND	P2C	P2B	P1B	NO

PC Board Layout Considerations

PC Board Considerations

The layout should be optimized for lowest noise on the Bt496 power and ground lines by shielding the digital inputs and providing good decoupling. The trace length between groups of VAA and GND pins should be as short as possible to minimize inductive ringing.

A well-designed power distribution network is critical to eliminate digital switching noise. Ground planes must provide a low-impedance return path for the digital circuits. A minimum of a six-layer PC board is recommended. The ground layer should be used as a shield to isolate noise from the analog traces with layer one (top) for the analog traces, layer two the ground plane (preferably analog ground plane), layer three for the analog power plane. The remaining layers should be used for digital traces and digital power supplies.

The optimum layout enables the Bt496 to be located as close as possible to the power supply connector and the video output connector.

Ground Planes

For optimum performance, a common digital and analog ground plane with tub isolation (at least a 1/8 inch gap) and connected together only at the power supply connector (or the lowest impedance source) is recommended. Ground plane partitioning should extend the analog ground plane no more than 2 inches from the power supply connector to preserve digital noise margins during MPU read cycles. Thus, the ground partitioning isolation technique is constrained by the noise margin degradation during digital readback of the Bt496.

The digital ground plane should be under all digital signal traces to minimize radiated noise and crosstalk. For maximum performance, a separate isolated ground plane for the analog output termination resistors, RSET resistor, and VREF circuitry should be used, as shown in Figure 7. Another isolated ground plane is used for the GND pins of the Bt496 and supply decoupling capacitors.

Power Planes

Separate digital and analog power planes are necessary. The digital power plane should provide power to all digital logic on the PC board, and the analog power plane should provide power to all Bt496 power pins, VREF circuitry, and COMP and VREF decoupling. There should be at least a 1/8 inch gap between the digital power plane and the analog power plane.

The analog power plane should be connected to the digital power plane (VCC) at a single point through a ferrite bead, as illustrated in Figure 7. This bead should be located within 3 inches of the Bt496 and provides resistance to switching currents, acting as a resistance at high frequencies. A low-resistance bead should be used, such as Ferroxcube 5659065-3B, Fair-Rite 2743001111, or TDK BF45-4001.

Plane-to-plane noise coupling can be reduced by ensuring that portions of the digital power and ground planes do not overlay portions of the analog power and ground planes, unless they can be arranged so that the plane-to-plane noise is common mode.

Device Decoupling

For optimum performance, all capacitors should be located as close to the device as possible, using the shortest leads possible (consistent with reliable operation) to reduce the lead inductance. Although chip capacitors have minimum inductance, radial lead ceramic capacitors may be substituted for chip capacitors and are better than axial lead capacitors for self-resonance. Values are chosen to have self-resonance above the pixel clock.

Power Supply Decoupling

Best power supply decoupling performance is obtained with a 0.1 µF ceramic capacitor in parallel with a 0.01 µF chip capacitor decoupling each of four groups of VAA pins to GND. The capacitors should be placed as close as possible to the device.

The 33 µF capacitor is for low-frequency power supply ripple; the 0.1 µF and 0.01 µF capacitors are for high-frequency power supply noise rejection.

A linear regulator to filter the analog power supply is recommended if the power supply noise is ≥ 200 mV or greater than 10 LSBs. This is especially important when a switching power supply is used and the switching frequency is close to the raster scan frequency. Note that about 10 percent of power supply hum and ripple noise less than 1 MHz will couple onto the analog outputs.

COMP Decoupling

The COMP pin must be decoupled to VAA, typically using a 0.1 µF ceramic capacitor. Low-frequency supply noise will require a larger value. Lead lengths should be minimized for best performance so that the self-resonance frequency is greater than the SCLK* frequency.

If the display has a "ghosting" problem, additional capacitance in parallel with the COMP capacitor may help to fix the problem.

PC Board Layout Considerations (continued)

Digital Signal Interconnect

The digital signals to the Bt496 should be isolated as much as possible from the analog outputs and other analog circuitry. Also, these signals should not overlay the analog power and ground planes.

Most noise on the analog outputs will be caused by excessive edge speeds (less than 3 ns), overshoot, undershoot, and ringing on the digital signals.

The digital edge speeds should be no faster than necessary, as feedthrough noise is proportional to the digital edge speeds. Lower speed applications will benefit by using lower speed logic (3–5 ns edge rates) to reduce data-related noise on the analog outputs.

Transmission line mismatch will exist if the line length reflection time is greater than one-fourth the signal edge time. This results in ringing, overshoot, and undershoot that can generate noise onto the analog outputs. Line termination or reducing the line length is the solution. For example, logic edge rates of 2 ns require line lengths of less than 4 inches without using termination. Ringing may be reduced by damping the line with a series resistor (10–50 Ω).

Radiation of digital signals can also be picked up by the analog circuitry. This is prevented by reducing the digital edge speeds (rise/fall time), minimizing ringing by using damping resistors, and minimizing coupling through PC board capacitance by routing 90 degrees to any analog signals.

Ensure that the power pins for the clock driver are properly decoupled to minimize transients. Minimize edge speeds and ringing, using damping resistors (10–50 Ω) or parallel termination where necessary.

If using parallel termination on digital signals, the resistors should be connected to the digital power and ground planes, not the analog power and ground planes.

Analog Signal Interconnect

The Bt496 should be located as close as possible to the output connectors to minimize noise pickup and reflections due to impedance mismatch.

The video output signals should overlay the analog ground plane, and not the analog power plane, to maximize the high-frequency power supply rejection.

For maximum performance, the analog outputs should have a source load resistor equal to the destination termination (via a clean isolated ground return path). The load resistor connection between the current output and GND should be as close as possible to the Bt496 to minimize reflections. Unused analog outputs should be connected to GND.

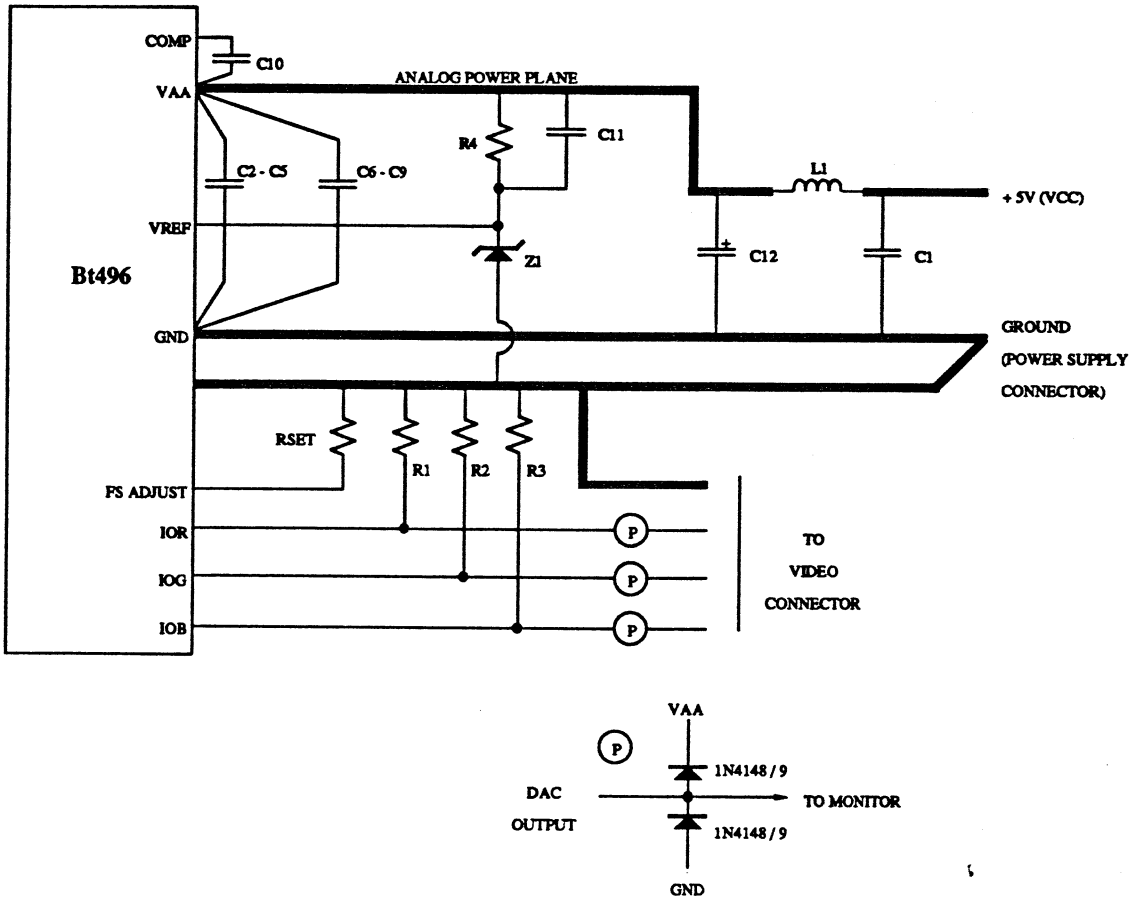
Analog edges exceeding the CRT monitor bandwidth can be reflected, producing cable-length dependent ghosts. Simple pulse filters can reduce high-frequency energy, reducing EMI and noise.

Analog Output Protection

The Bt496 analog outputs 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 7 can prevent latchup under severe discharge conditions without adversely degrading analog transition times. The 1N4148/9 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).

PC Board Layout Considerations (continued)



Location	Description	Vendor Part Number
C1–C5, C10, C11	0.1 µF ceramic capacitor	Erie RPE110Z5U104M50V
C6–C9	0.01 µF ceramic chip capacitor	AVX 12102T103QA1018
C12	33 µF tantalum capacitor	Mallory CSR13F336KM
L1	ferrite bead	Fair-Rite 2743001111
R1, R2, R3	75-Ω 1% metal film resistor	Dale CMF-55C
R4	1000-Ω 1% metal film resistor	Dale CMF-55C
RSET	562 or 530-Ω 0.5% metal film resistor	Dale CMF-55C
Z1	1.2 V voltage reference	National Semiconductor LM385Z-1.2

Note: The vendor numbers above are listed only as a guide. Substitution of devices with similar characteristics will not affect the performance of the Bt496.

Figure 7. Typical Connection Diagram and Parts List.

Application Information

Clock Interfacing

Due to the high clock rates at which the Bt496 may operate, it is designed to accept differential clock signals (CLOCK and CLOCK*). These clock inputs are designed to be generated by a differential ECL crystal oscillator operating from a +5 V supply. Note that the CLOCK and CLOCK* inputs require termination resistors (220 Ω to GND). The termination resistors should be located as close as possible to the CLOCK generator (Bt438 shown). A 150 Ω resistor between CLOCK and CLOCK* should be placed as close to the Bt496 as possible.

Typically, SCLK* is used to generate the pixel VRAM serial clock. Due to the low drive capability of the SCLK output, this signal should be immediately buffered by a high performance inverting buffer such as a Signetics 74F1804.

Figure 8 illustrates a sample VRAM serial clock generation circuit. Typically, VIDCLK is used for clocking the SYNC and BLANK generation logic.

The CLOCK and CLOCK* inputs must be differential signals and greater than 0.6 V peak-to-peak due to the noise margins of the CMOS process. The Bt496 will not function using a single-ended clock with CLOCK* connected to ground.

ESD and Latchup Considerations

Correct ESD-sensitive handling procedures are required to prevent device damage, which can produce symptoms of catastrophic failure or erratic device behavior with somewhat "leaky" inputs.

Latchup can be prevented by ensuring that all VAA pins are at the same potential, and that the VAA supply voltage is applied before the signal pin voltages. The correct power-up sequence ensures that any signal pin voltage will never exceed the power supply voltage by more than +0.5 V.

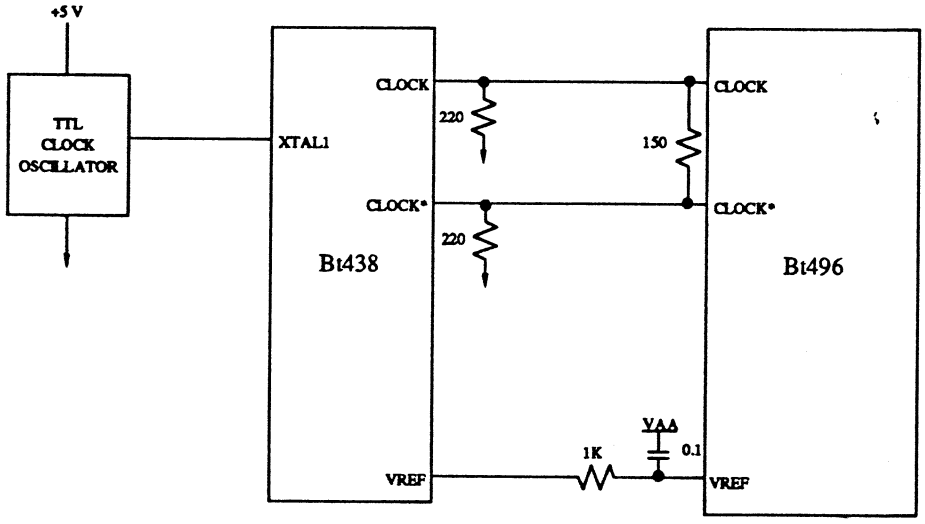


Figure 8. Generating the Bt496 Clock Signals.

Application Information (continued)

Test Features of the Bt496

The Bt496 contains a test register, three signature analysis registers, and an analog output comparator that assist the user in evaluating the performance and functionality of the part. This section explains the operation and use of these test features. For proper use, the test register must be enabled and the proper test method selected. Please refer back to the *Test Register ($00F)* section. During normal operation, disable the test register for power considerations.

Signature Register (Signature Mode)

The signature register, when enabled (TR0 = 0), operates with three 9-bit vectors of data that are output from the color palette RAM. These three 9-bit vectors each represent a single color component of the pixel-color, and are presented as inputs simultaneously to the red, green, and blue signature analysis registers (SARs), as well as the three on-chip DACs. The SARs act as three 9-bit wide Linear Feedback Shift Registers on each succeeding pixel that is latched.

The Bt496 will only generate signatures while in active-display (BLANK* Negated). The SARs are available for reading and writing via the MPU port when the Bt496 is in a blanking state (BLANK* asserted).

Typically, the user will write three specific 9-bit "seed" values into the SARs. Then, a known pixel stream will be input to the chip, for example, one scan-line worth of pixels. At the succeeding blank state, the resultant 9-bit signatures can be read out by the MPU. The 9-bit signature register data is a result of the same captured data that is fed to the DACs.

It is not simple to specify the algorithm that the linear feedback shift operation uses in the Bt496. The linear feedback configuration is shown in Figure 9. Experienced users have developed tables of specific seeds and pixel streams and recorded the signatures that result from those inputs applied to "known-good" parts. Note that a good signature from one given pixel stream can be used as the seed for the succeeding stream to be tested. Any signature is deterministically created from a starting seed and the succeeding pixel stream fed to the SARs.

Signature Register (Data Strobe Mode)

Setting test register bit TR0 to "1" puts the SARs into data strobe mode. In this instance, the linear feedback circuits of the SARs are disabled, which stops the SARs from generating signatures. Instead, the SARs simply capture and hold the pixel data that is selected. Any MPU data written to the SARs is ignored. This mode is most useful when using a sophisticated VLSI semiconductor tester.

Analog Comparator

The other dedicated test structure in the Bt496 is the analog output comparator. It allows the user to measure the DACs against each other, as well as against a specific reference voltage.

Four combinations of tests are selected via the Test Register. With a given setting, the respective signals (DAC outputs or the 150 mV reference) will be continuously input to the comparator. The result of the comparator is latched into the Test Register.

Due to the simple design of the comparator, it is recommended that the DAC outputs be stable for 5 µs before capture. At a display rate of 100 MHz, 5 µs corresponds to 500 pixels. Furthermore, either the color palette RAM or the pixel inputs (or both) should be configured to guarantee a single continuous output from the DACs under test, up until capture.

Typically, users will create line-wide test bands of various colors. For each test, the result is obtained by reading Test Register bit D3.

Application Information (continued)

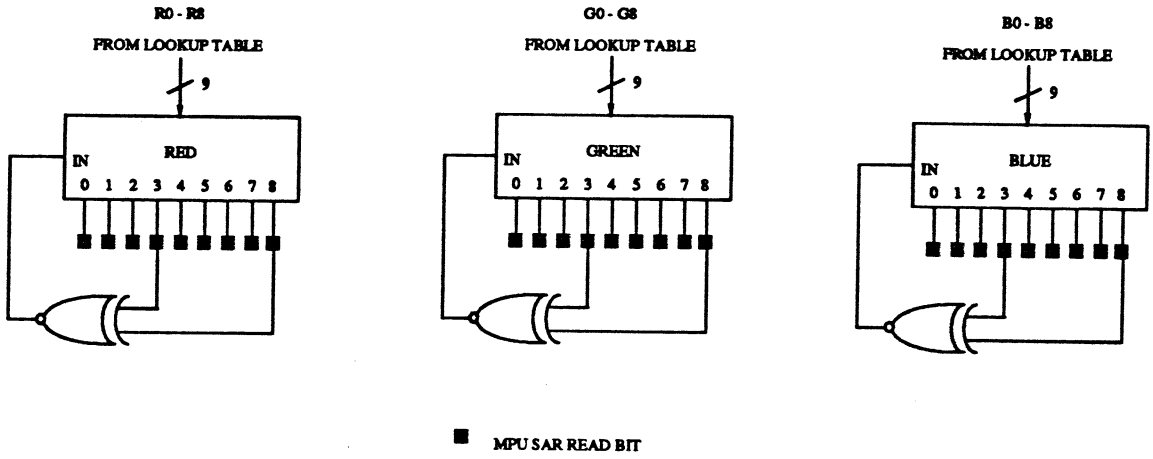


Figure 9. Signature Analysis Register Circuit.

Application Information (continued)

Initializing the Bt496

Following a power-on, the Bt496 must be initialized. This sequence will configure the Bt496 as follows:

Pixel Depth: 16 bits per pixel
Pixel Unpacking Mode: from MSB
RGB partitioning: 6 bits red, 5 bits green, 5 bits blue
All bits used in lookup table addressing (i.e., no masking)
Pan: 0 pixels; Zoom: 1x
Sync Enabled on IOG, 7.5 IRE blanking Pedestal
VIDCLK = CLOCK/4
Mid-line VRAM transfer used
Signature analysis testing used

Control Register Initialization

Write $000 to Address Register
Write $014 to Command Register 0
Write $000 to Command Register 1
Write $0C0 to Command Register 2
Write $0FF to Red Pixel Read Mask Register
Write $0FF to Green Pixel Read Mask Register
Write $0FF to Blue Pixel Read Mask Register

Color Palette RAM initialization

Write $000 to Address Register
Write Red Data to Palette RAM (location $00)
Write Green Data to Palette RAM (location $00)
Write Blue Data to Palette RAM (location $00)
Write Red Data to Palette RAM (location $01)
Write Green Data to Palette RAM (location $01)
Write Blue Data to Palette RAM (location $01)

Write Red Data to Palette RAM (location $FF)
Write Green Data to Palette RAM (location $FF)
Write Blue Data to Palette RAM (location $FF)

Recommended Operating Conditions

Parameter	Symbol	Min	Typ	Max	Units
Power Supply	VAA	4.75	5.00	5.25	V
Ambient Operating Temperature	TA	0		+70	°C
Output Load	RL		37.5		Ω
Reference Voltage	VREF	1.20	1.235	1.26	V
FS ADJUST Resistor	RSET		562		Ω

Absolute Maximum Ratings

Parameter	Symbol	Min	Typ	Max	Units
VAA (measured to GND)				6.5	V
Voltage on Any Signal Pin*		GND–0.5		VAA + 0.5	V
Analog Output Short Circuit Duration to any Power Supply or Common	ISC		indefinite		
Ambient Operating Temperature	TA	–55		+125	°C
Storage Temperature	TS	–65		+150	°C
Junction Temperature	TJ				
PQFP	TJ			+150	°C
PGA	TJ			+175	°C
Soldering Temperature (5 seconds, 1/4" from pin)	TSOL			260	°C
Vapor Phase Soldering (1 minute)	TVSOL			220	°C

Note: Stresses above those listed under "Absolute Maximum Ratings" may cause permanent damage to the device. This is a stress rating only and functional operation of the device at these or any other conditions above those listed in the operational sections of this specification is not implied. Exposure to absolute maximum rating conditions for extended periods may affect device reliability.

* This device employs high-impedance CMOS devices on all signal pins and should be handled as an ESD-sensitive device. Voltage on any signal pin that exceeds the power supply voltage by more than +0.5 V can induce destructive latchup.

DC Characteristics

Parameter	Symbol	Min	Typ	Max	Units
Analog Outputs					
Resolution (each DAC)		9	9	9	Bits
Accuracy (each DAC)					
Integral Linearity Error	IL			±1	LSB
Differential Linearity Error	DL			±1	LSB
Gray Scale Error				±5	% Gray Scale
Monotonicity			guaranteed		
Coding					Binary
Digital Inputs (except CLOCK, CLOCK*)					
Input High Voltage	VIH	2.0		VAA + 0.5	V
Input Low Voltage	VIL	GND–0.5		0.8	V
Input High Current (Vin = 2.4 V)	IIH			1	µA
Input Low Current (Vin = 0.4 V)	IIL			–1	µA
Input Capacitance (f = 1 MHz, Vin = 2.4 V)	CIN		4	10	pF
Clock Inputs (CLOCK, CLOCK*)	DVIN	.6		6	V
Input High Current (Vin = 4.0 V)	IKIH			1	µA
Input Low Current (Vin = 0.4 V)	IKIL			–1	µA
Input Capacitance (f = 1 MHz, Vin = 4.0 V)	CKIN		4	10	pF
Digital Outputs (D0–D8), SCLK*, VIDCLK.					
Output High Voltage (IOH = –400 µA)	VOH	2.4			V
Output Low Voltage (IOL = 1.0 mA)	VOL			0.4	V
3-state Current	IOZ			10	µA
Output Capacitance	CDOUT		10		pF

See test conditions on next page.

DC Characteristics (continued)

Parameter	Symbol	Min	Typ	Max	Units
Analog Outputs					
Output Current					
White Level Relative to Blank		17.69	19.05	20.40	mA
White Level Relative to Black		16.74	17.62	18.50	mA
Black Level Relative to Blank					
SETUP = 7.5 IRE		0.95	1.44	1.90	mA
SETUP = 0 IRE		0	5	50	µA
Blank Level on IOG		6.29	7.62	8.96	mA
Blank Level on IOR, IOB		0	5	50	µA
Sync Level on IOG		0	5	50	µA
LSB Size			37		µA
DAC-to-DAC Matching			2	5	%
Output Compliance	VOC	-0.5		+1.2	V
Output Impedance	RAOUT		50		kΩ
Output Capacitance (f = 1 MHz, IOUT = 0 mA)	CAOUT		13	20	pF
Voltage Reference Input Current	IREF		10		µA
Power Supply Rejection Ratio (COMP = 0.1 µF, f = 1 KHz)	PSRR		0.5		% / % ΔVAA

Test conditions (unless otherwise specified): "Recommended Operating Conditions" with RSET = 562 Ω, VREF = 1.235 V. SETUP = 7.5 IRE. As the above parameters are guaranteed over the full temperature range, temperature coefficients are not specified or required.

AC Characteristics

Parameter	Symbol	Min/Typ/ Max	100 MHz	Units
Clock Rate	Fmax	max	110	MHz
R/W, C1,C0 Setup Time	1	min	0	ns
R/W, C1,C0 Hold Time	2	min	15	ns
CE* Low Time	3	min	50	ns
CE* High Time	4	min	25	ns
CE* Asserted to Data Bus Driven	5	min	7	ns
CE* Asserted to Data Valid	6	max	75	ns
CE* Negated to Data Bus 3-Stated	7	max	15	ns
Write Data Setup Time	8	min	35	ns
Write Data Hold Time	9	min	3	ns
Pixel and Control Setup Time	10	min	0	ns
Pixel and Control Hold Time	11	min	6	ns
Clock Cycle Time	12	min	10	ns
Clock Pulse Width High Time	13	min	4.5	ns
Clock Pulse Width Low Time	14	min	4.5	ns

See test conditions on next page.

AC Characteristics (continued)

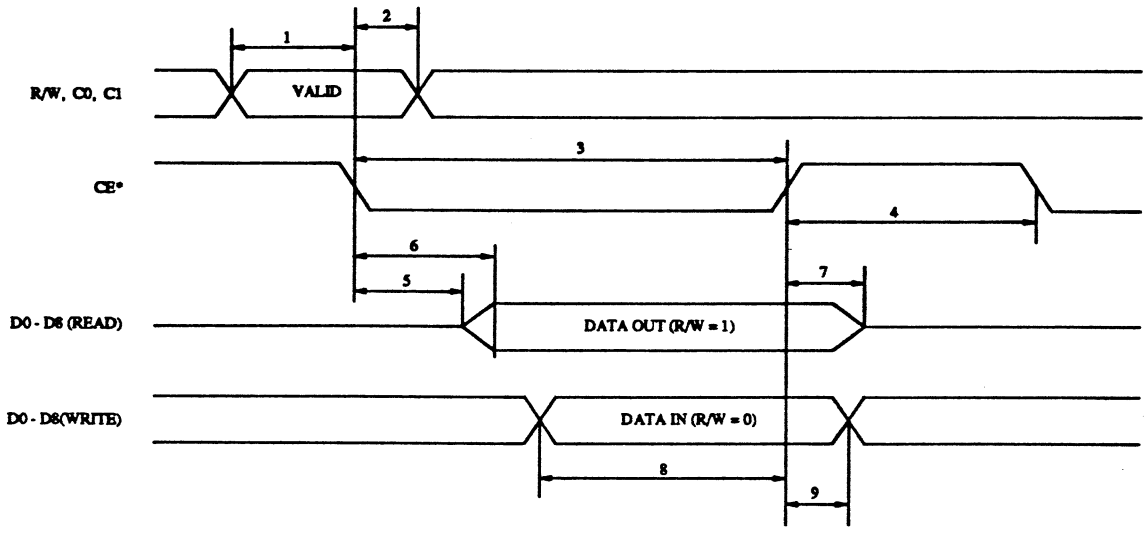
Parameter	Symbol	Min/Typ/ Max	100 MHz	Units
Analog Output Delay	18	typ	12	ns
Analog Output Rise/Fall Time	19	typ	1.5	ns
Analog Output Settling Time	20	max	8	ns
Clock and Data Feedthrough*		typ	tbd	dB
Glitch Impulse*		typ	50	pV – sec
DAC to DAC Crosstalk		typ	tbd	dB
Analog Output Skew		typ	0	ns
		max	2	ns
SYNC*, BLANK* Setup Time	21	min	3	ns
SYNC*, BLANK* Hold Time	22	min	10	ns
Pipeline Delay		min	tbd	Clocks
		max	tbd	Clocks
VAA Supply Current**	IAA	typ	tbd	mA
		max	tbd	mA

Test conditions (unless otherwise specified): "Recommended Operating Conditions" with RSET = 562 Ω, VREF = 1.235 V. TTL input values are 0–3 V, with input rise/fall times ≤ 3 ns, measured between the 10 percent and 90 percent points. ECL input values are VAA–0.8 to VAA–1.8 V, with input rise/fall times ≤2 ns, measured between the 20 percent and 80 percent points. Timing reference points at 50 percent for inputs and outputs. Analog output load ≤10 pF, D0–D8 output load ≤40 pF. See Notes 1–3 in the Pixel Input/Output Timing diagram. As the above parameters are guaranteed over the full temperature range, temperature coefficients are not specified or required.

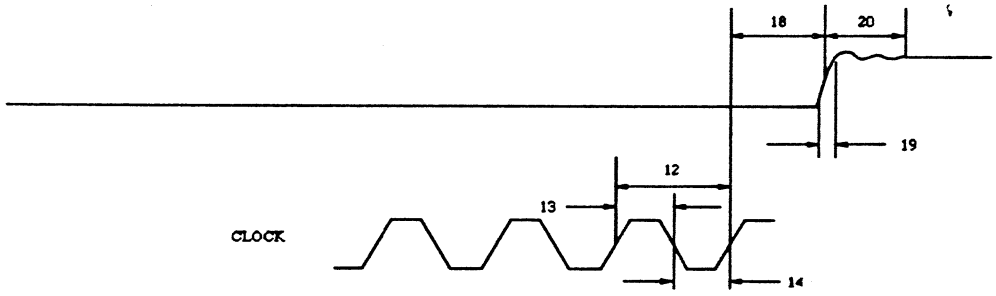
*Clock and data feedthrough is a function of the amount of edge rates, overshoot, and undershoot on the digital inputs. For this test, the TTL digital inputs have a 1 kΩ resistor to GND and are driven by 74HC logic. Settling time does not include clock and data feedthrough. Glitch impulse includes clock and data feedthrough, –3 dB test bandwidth = 2x clock rate.

**At Fmax. IAA (typ) at VAA = 5.0 V, TA = 20° C. IAA (max) at VAA = 5.25 V, TA = 0° C.

Timing Waveforms



MPU Read/Write Timing Dimensions.



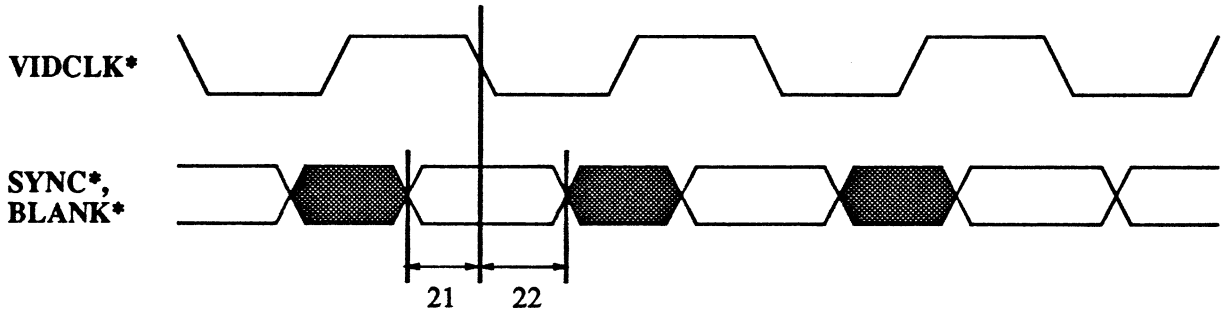
Note 1: Output delay time measured from 50 percent point of the rising clock edge to 50 percent point of full- scale transition.

Note 2: Output settling time measured from 50 percent point of full-scale transition to output settling within ±1 LSB.

Note 3: Output rise/fall time measured between 10 percent and 90 percent points of full-scale

Pixel Input/Output Timing.

Timing Waveforms (continued)



Video Control Input Timing.

Ordering Information

Model Number	Speed	Package	Ambient Temperature Range
Bt496KG100	100 MHz	207-pin Ceramic PGA	0° to +70° C

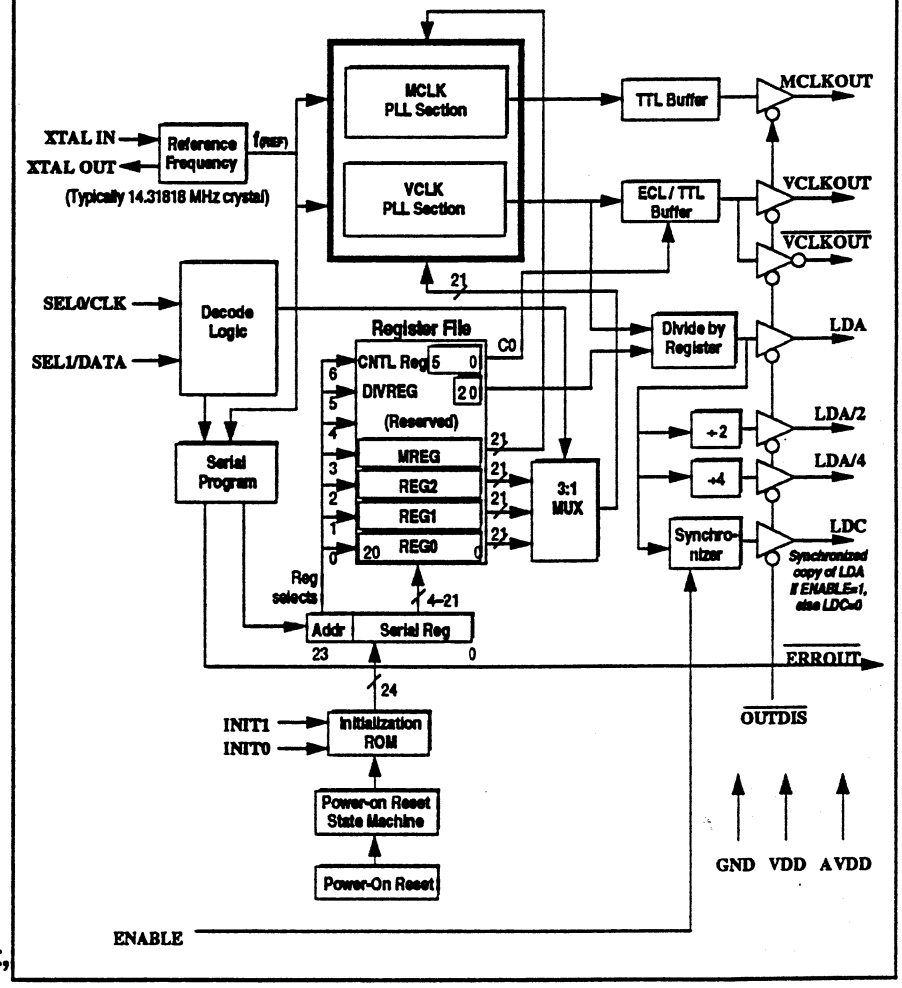
ICD2062

Dual Programmable ECL/TTL Clock Generator

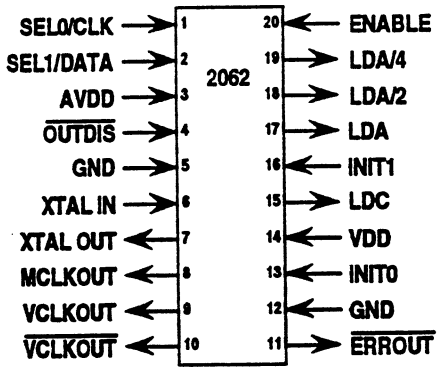
Single-Chip Dual Programmable Oscillator Handles All Frequency Requirements of Popular Graphic Chip Sets

- 2nd Generation Dual Oscillator Graphics Clock Generator
- 2 Independent Clock Outputs from 320 KHz – 160 MHz (Differential ECL Video Clock Generation) and 320 KHz – 120 MHz (CMOS Outputs)
- Individually Programmable Oscillators Using a Highly Reliable Manchester-Encoded 21–Bit Serial Data Word
- 2–Pin Serial Programming Interface Allows Direct Connection to Most Graphic Chip Sets with no External Hardware Required
- Programmable Video Clock Dividers Allow for Easy Interface to Most RAMDACs and VRAMs
- Tri-State Oscillator Control Disables Outputs for Test Purposes
- Phase-Locked Loop Oscillator Input Derived from Single 14.31818 MHz Crystal Input
- Sophisticated Internal Loop-Filter Requires no External Components or Manufacturing "Tweaks" as Commonly Required with External Filters
- Low-Power, High-Speed, 5–Volt, 1.25µ CMOS Technology
- Available in 20–Pin DIP or SOIC Package Configuration

ICD2062 Dual-Programmable ECL/TTL Clock Generator Block Diagram



Pin Descriptions



Signal Descriptions

Signal	Pin Number	Signal Function
SEL0/CLK	1	Bit 0 (LSB) of frequency select logic, used to select oscillator frequencies; Clock Input in serial programming mode.
SEL1/DATA	2	Bit 1 (MSB) of frequency select logic, used to select oscillator frequencies; Data Input in serial programming mode.
AVDD	3	+5 Volts to Analog Core
OUTDIS–	4	Output Disable (3–State Output Enable) when signal is pulled low. (Internal pull-up allows no-connect if 3–state operation not required.)
GND	5	Ground
XTAL IN	6	Input Reference Oscillator for all Phase-Locked Loops (nominally 14.31818 MHz). An optional PC System Bus Clock signal may be used as input if available.
XTAL OUT	7	Oscillator Output to a Reference Series-Resonant Crystal (nominally 14.31818 MHz). For higher accuracy, a parallel-resonant crystal may be used. Assume C_{LOAD} = 17pf. For more specifications, see the IC DESIGNS Application Note on crystal oscillators. (Pin is no-connect if external reference oscillator or System Bus Clock Signal is used.)
MCLKOUT	8	Memory Clock out
VCLKOUT	9	Differential clock outputs. Connect directly to RAMDAC CLOCK & CLOCK– inputs. Can drive 4 RAMDACs.
VCLKOUT–	10	Output levels equivalent to 10KH ECL circuit operating from single supply. VCLKOUT– is skew-free.
ERROUT–	11	Error Output: a low signals an error in the Serial Programming Word.
GND	12	Ground
INIT0	13	Select power-up initial conditions (LSB).
VDD	14	+5 Volts
LDC	15	Load output (TTL compatible). When ENABLE is high, has same timing as LDA output. Can drive up to 4 capacitive loads without buffering.
INIT1	16	Select power-up initial conditions (MSB).
LDA	17	Skew-free Load Outputs (TTL compatible). Generated by dividing VCLKOUT by Div Register (1, 2, 3, 4, 5, 8). Each output can drive up to 4 capacitive loads without buffering.
LDA/2	18	Generated by dividing LDA by two
LDA/4	19	Generated by dividing LDA by four.
ENABLE	20	Synchronous load enable input. Internally synched to LDA, used to start/stop LDC output synchronously. If ENABLE is low, LDC is held low; when high, LDC is free-running.

Table of Contents

Introduction

The ICD2062 is a clock generator for high-resolution video displays. It uses a low-frequency (and low-cost) reference crystal to produce the following: a 10KH compatible complementary ECL oscillator signal for high-speed video RAMDACs, a high-speed TTL oscillator signal for video RAMs and system logic operation, and the requisite load, control and clock signals to control the loading of data between the CRT controller, VRAM and RAMDACs.

The ICD2062 Dual Programmable Clock Generator offers 2 fully user-programmable phase-locked loops in a single package. The outputs may be changed "on the fly" to any desired frequency value in the range 320 KHz to 160 MHz (VCLKOUT) and 320 KHz to 120 MHz (MCLKOUT). The ICD2062 is ideally suited for any design where multiple or varying frequencies are required, replacing more expensive metal can oscillators, particularly where the application requires expensive complementary ECL oscillators.

The Video Clock output may be programmatically divided by 1, 2, 3, 4, 5 or 8 in order to generate the Load Signal, which is further divided by 2 and 4 for clocking video timing logic. A second Load Signal may be synchronously gated in order to enable starting and stopping the clocking of video RAMs. The ICD2062 can also configure the pipeline delay of certain RAMDACs (such as the Bt457/458) to a fixed pipeline delay.

Being able to change the output frequency dynamically adds a whole new degree of freedom for the electrical engineer heretofore unavailable with existing crystal oscillator devices. Some examples of the uses for this device include: graphics board dot clocks to allow dynamic synchronization with different brands of monitors or display formats; and on-board test strategies where the ability to skew a system's desired frequency (for example: ±10%) allows worst case evaluations.

Register Definitions

Register File

The Register File consists of the following registers and their selection addresses:

Register Addressing

Address	Register	Usage
000	REG0	Video Clock Register 1
001	REG1	Video Clock Register 2
010	REG2	Video Clock Register 3
011	MREG	Memory or I/O Timing Clock
100	(Reserved)	
101	DIVREG	Load Divisor Register
110	CNTL Reg	Control Register

Register Selection

Video clock output is controlled not only by the SEL0 & SEL1 bits, but also by the OUTDIS– signal, as follows:

VCLKOUT Selection

OUTDIS–	SEL1	SEL0	VCLKOUT
0	X	X	High-Z
1	0	0	REG0
1	0	1	REG1
1	1	X	REG2

The Memory Clock output is controlled by the OUTDIS– signal as indicated below:

MCLKOUT Selection

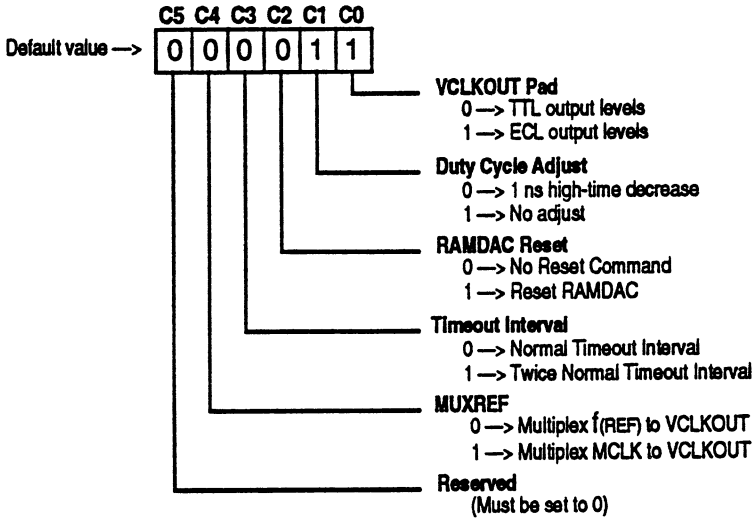
OUTDIS–	MCLKOUT
0	High-Z
1	MREG

The Clock Select pins SEL0 & SEL1 have a dual purpose. When these pins are performing serial down-load, the VCLKOUT signal remains unchanged (unless the currently selected register is the one being programmed). When the pins SEL0 & SEL1 are functioning as register selects, a timeout interval is required to determine whether the user desires register select or serial programming. At the end of the timeout interval, new register selection occurs. At this point, the VCLKOUT signal will be multiplexed to the reference signal $f_{(REF)}$ for an additional timeout interval to give the VCO time to settle to its new value. [The timeout interval in both cases is approximately 5 msec — see the timeout interval spec in the AC Timing Section.]

Control Register Definition

The Control Register (CNTL Reg) allows the user to adjust various internal options. Most of these options are for special cases, and should have no applicability to standard graphics usage. The register word is defined as follows:

Control Register



VCLKOUT Pad — This control bit determines whether the VCLKOUT Pad is at ECL or TTL levels. The default is ECL levels. When in TTL mode, the VCLKOUT– Pad is nonfunctional, and remains tri-stated.

Duty Cycle Adjust — This control bit causes a 1ns decrease in the output waveform high time. The default is no adjustment. In situations in which the capacitive load is beyond device specifications, or where the Threshold Voltage VTH is to be changed from CMOS to TTL levels, this adjustment can sometimes bring the output closer to 50% duty cycle.

RAMDAC Reset — This control bit, when set, will cause the ICD2062 to issue a RAMDAC reset sequence, which is required by some specific RAMDACs (such as the Bt457/458). For more specifics on this operation, see the paragraph titled "Internal Reset Sequence" in the section titled "RAMDAC/VRAM Interface". NOTE: This operation will only take place the first time this bit is set.

Timeout Interval — The Timeout Interval is normally defined as in the AC Specifications. It is derived from the MCLK VCO, and if this VCO is programmed to certain extremes, the timeout may be too short. If this control bit is set, the Timeout Interval is doubled.

MUXREF — This control bit determines which clock is multiplexed to the VCLKOUT output during frequency changes. While the VCLK VCO changes to a different frequency, a known clock is multiplexed to the output. The default is to multiplex the f(REF) reference frequency, but some graphic controllers cannot run as slow as f(REF). This bit, when set, allows the MCLK to be used as an alternative frequency.

Divide Register Definition

The output signals LDA, LDA/2, LDA/4, and LDC are all a function of the VCLK VCO value divided by the division factor stored in the Divide Register (DIVREG). DIVREG is at address 101.

DIVREG Division Factors

D2	D1	D0	Division Factor	Clock Low (cycles)	Clock High (cycles)
0	0	0	+3	1	2
0	0	1	+4	2	2
0	1	0	+5	2	3
0	1	1	+8	4	4
1	0	X	+1	1/2	1/2
1	1	X	+2	1	1

Register Initialization

The ICD2062 Clock Synthesizer has several of its registers in a known state upon power-up. This is implemented by the Power-On initialization circuitry. Three VGA registers are initialized based on the state of the INIT1 and INIT0 pins at power-up. Also, the Memory Clock is initialized based on the INIT pins.

The Power-On Reset function operates transparently to the video subsystem. It performs its initialization function and is cleared before the system Power-On Reset permits the system to begin its boot process. The INIT pins must ramp up with VDD if a 1 on either of these pins is desired. They are internally pulled down, and so will default to 0 if left unconnected.

The various registers are initialized as follows (all frequencies in MHz):

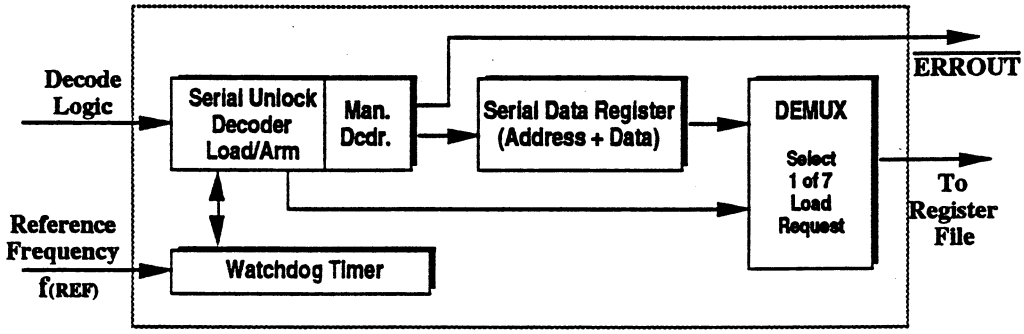
Register Initialization — ROM Option 1

INIT1	INIT0	MREG	REG0	REG1	REG2
0	0	32.5	25.175	28.322	28.322
0	1	40.0	25.175	28.322	28.322
1	0	50.350	40.00	28.322	28.322
1	1	56.644	40.00	50.35	25.175

Serial Programming Architecture

The ICD2062 programming scheme is simple, yet impenetrable to accidental access. Because the only common denominator between most VGA and 8514 controllers is a few clock select pins, these have to perform the dual function of clock selection and serial programming. The Serial Program Block (See diagram on p. 1) contains several components: a Serial Unlock Decoder (containing the Unlocking Mechanism and Manchester Decoder), a Watchdog Timer, the Serial Data Register (Serial Reg) and a Demultiplexer to the Register File.

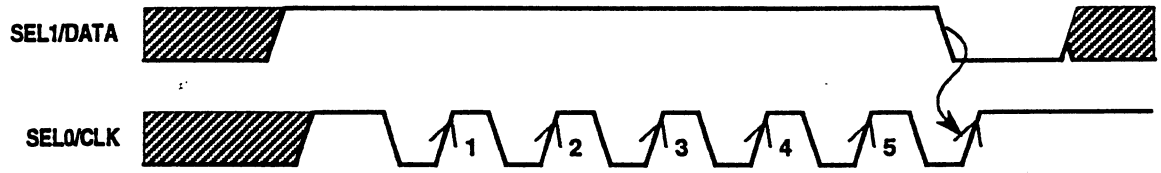
Serial Program Block Diagram — Detail



Unlocking Mechanism

The Unlocking Mechanism watches for an initial break sequence detailed in the following timing diagram:

Unlock Sequence



The initial unlock sequence consists of at least five low-to-high transitions of CLK with DATA high, followed immediately by a single low-to-high transition of CLK with DATA low. Following this unlock sequence, the encoded serial data is clocked into the Serial Data Register.

Watchdog timer

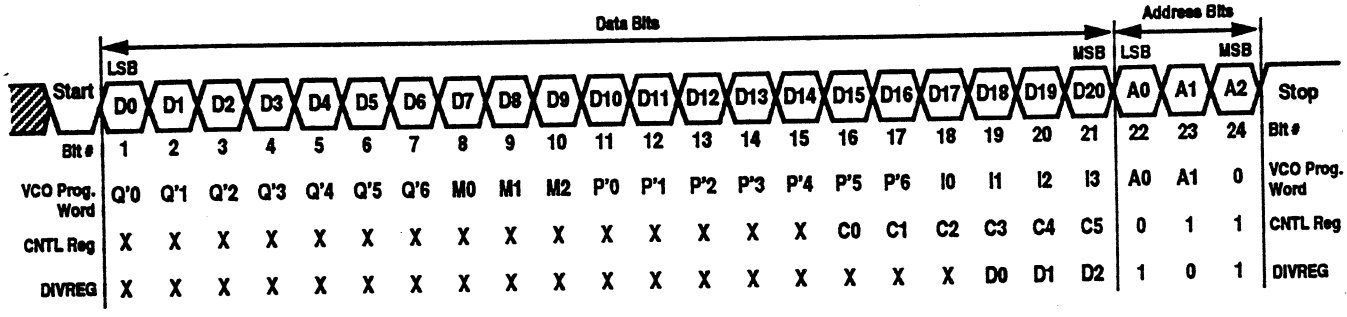
Following any transition of CLK or DATA, the Watchdog Timer is reset and begins counting. Throughout the entire programming process, the Watchdog Timer ensures that rising edges of CLK do not violate the timeout specification (of 1ms — see AC specs). If a timeout does occur, the Lock Mechanism is rearmed and the current data in the Serial Data Register is lost.

Since the VCLK registers are selected by the SEL0 or SEL1 bits, and since any change in their state may affect the resultant output frequency, new data input on the Selection Bits is only permitted to pass through to the Decode Logic after the Watchdog Timer has timed out. This delay of SEL0 or SEL1 data permits a serial program cycle to take place without affecting the current register selection. The process of serial programming has no effect on the performance of the graphics subsystem. [Note that there is a latency amounting to the duration of the Watchdog Timer before any new VCLK register selections take effect.]

The Serial Data Register

Serial data is clocked into the Serial Data Register in the following order:

Serial Data Timing



The serial data is sent using a modified Manchester encoded data format. This is defined as follows:

1 — An individual data bit is sampled on the rising edge of CLK.

2 — The complement of the data bit must be sampled on the previous falling edge of CLK.

3 — The Setup and Hold Time requirements must be met on both CLK edges.

4 — The unlock sequence, start, and stop bits are not Manchester encoded.

For specifics on timing, see the timing diagram in the Device Specifications Section at the end of this Datasheet.

The bits are shifted in this order: a start bit, 21 data bits, 3 address bits (which designate the desired register), and a stop bit (which also functions as a load strobe to transfer the data from the Serial Reg into the desired register). For the VCO registers (REG0, REG1, REG2, MREG), the data is made up of 4 fields: D[20:17] = Index; D[16:10] = P'; D[9:7] = Mux; D[6:0] = Q'. [See the section on "Programming the 2061" for more details on the VCO data word.] For the other registers of fewer than 21 bits (DIVREG, CNTL Reg), the upper bits are used (starting with the MSB). A total of 24 bits must always be loaded into the Serial Data Register (or an error is issued).

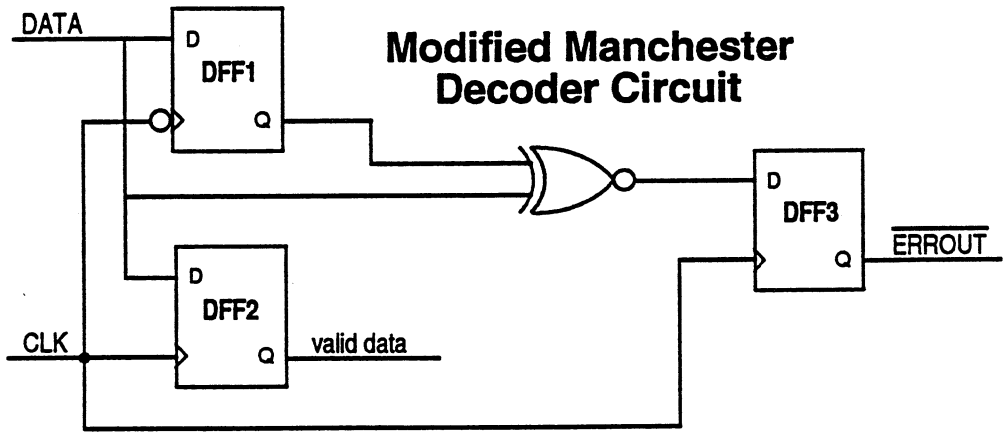
Following the entry of the last address bit, a stop bit or "load command" is issued by bringing DATA high and toggling CLK high-to-low and low-to-high. The Unlocking Mechanism then automatically rearms itself following the load. Only when the Watchdog Timer has timed out are the SEL0 & SEL1 pins permitted to return to their normal clock select function.

Note that the Serial Data Register (Serial Reg) which receives the address and data bits is exactly the correct length to accept the data being sent. The stop bit is used as a load command which passes the Serial Reg contents on to the register file location indicated by the address bits. If a stop bit is not received after the Serial Data Register has been filled, but rather more valid encoded data is received, then all of the received serial data is ignored, the Unlocking Mechanism rearmed, and an error is issued. The device counts the serial data clock edges to know exactly when the Serial Buffer is full, and thus to know which bit is the stop bit. Following the stop bit, the Unlocking Mechanism rearms itself. If corrupt data is detected (i.e., incorrectly Manchester-encoded data), then the Unlocking Mechanism is rearmed, the Serial Counter reset, all received data ignored, and ERROUT– is asserted.

ERROUT– Operation

The ERROUT– signal is used to inform when a program error has been detected internally by the ICD2062. The signal remains low until the next unlock sequence.

The following circuit shows the basic mechanism used to detect valid and erroneous serial data:



RULE: Must have different values on the rising and falling edges when sampling the falling edge first. Valid data is read on the rising edge of CLK.

The ERROUT– signal is invoked for any of the following error conditions: incorrect start bit; incorrect Manchester encoding; incorrect length of data word; incorrect stop bit.

NOTE: If there is no input pin available on the target VGA controller chip to monitor ERROUT–, a software routine which counts VSYNC pulses to measure output frequency may be used as a determination of programming accuracy.

Programming the ICD2062

The desired output frequency is defined via a serial interface, with a 21–bit number shifted in. The ICD2062 has two programmable oscillators, requiring a 21–bit programming word (W) to be loaded into each channel's respective registers independently. This word contains 4 fields:

Field	# Bits	
Index (I)	4 bits	(MSB—Most Significant Bits)
P counter (P')	7 bits	
Mux (M)	3 bits	
Q counter (Q')	7 bits	(LSB—Least Significant Bits)

The frequency of the programmable oscillator $f_{(VCO)}$ is determined by these fields as follows:

$$P' = \overline{130 - P} \qquad Q' = \overline{129 - Q}$$

$$f_{(VCO)} = (2 \cdot f_{(REF)} \cdot \frac{P}{Q})$$

where $f_{(REF)}$ = Reference frequency (between 1 MHz – 60 MHz; typically 14.31818 MHz). NOTE: If a reference frequency other than 14.31818 MHz is used, then the initially loaded ROM frequencies will not be correct.

The value of $f_{(VCO)}$ must remain between 40 MHz and 160 MHz inclusive. Therefore, for output frequencies below 40 MHz, $f_{(VCO)}$ must be brought into range. To accomplish this, a post-VCO Divisor is selected by setting the values of the Mux field (M) as follows:

M	Divisor
000	1
001	2
010	4
011	8
100	16
101	32
110	64
111	128

The Index field (I) is used to preset the VCO to an appropriate range. The value for this field should be chosen from the following table. (Note that this table is referenced to the VCO frequency, $f_{(VCO)}$, rather than to the desired output frequency.) Note that VCLK may be shut off, but that MCLK must be left running.

$f_{(VCO)}$ (MHz)	I
40.0 – 42.5	0000
42.5 – 47.5	0001
47.5 – 53.5	0010
53.5 – 58.5	0011
58.5 – 62.5	0100
62.5 – 68.5	0101
68.5 – 76.0	0110
76.0 – 82.0	0111
82.0 – 87.0	1000
87.0 – 92.0	1001
92.0 – 97.0	1010
97.0 – 105.0	1011
105.0 – 115.0	1100
115.0 – ∞	1101
Turn off VCLK	1110
Mux MCLK to VCLK	1111

When the Index Field is programmed to 1111, VCLK is turned off and both channels run from the same VCLK VCO.

To assist with these calculations, IC DESIGNS provides the SERDATA program. SERDATA is a program for the IBM PC which automatically generates the appropriate programming word from the user's reference and desired output frequencies. SERDATA is also available for the Apple Macintosh as a HyperCard 2.0 stack.

Programming Constraints

There are five primary programming constraints the user must be aware of:

$$1\text{MHz} \leq f_{(REF)} \leq 60\text{MHz}$$

$$200\text{KHz} \leq \frac{f_{(REF)}}{Q} \leq 1\text{MHz}$$

$$40\text{MHz} \leq f_{(VCO)} \leq 160\text{MHz}$$

$$3 \leq Q \leq 129$$

$$4 \leq P \leq 130$$

The constraints have to do with trade-offs between optimum speed with lowest noise, VCO stability, and factors affecting the loop equation. The factors are listed here for completeness' sake; however, by using the aforementioned SERDATA program, these constraints become transparent.

Programming Example

The following is an example of the calculations SERDATA performs:

Derive the proper programming word for a 39.5 MHz output frequency, using 14.31818 MHz as the reference frequency:

Since 39.5 MHz < 40 MHz, double it to 79.0 MHz. Set M to 001. Set I to 0111. The result:

$$f_{(VCO)} = 79.0 = (2 \bullet 14.31818 \bullet \frac{P}{Q})$$

$$\frac{P}{Q} = 2.7587$$

Several choices of P and Q are available:

P	Q	$f_{(VCO)}$	Error (PPM)
69	25	79.0363	460
80	29	78.9969	40
91	33	78.9669	419

Choose (P, Q) = (80,29) for best accuracy (40 PPM).

Therefore:

$$P' = \overline{130 - P} = \overline{130 - 80} = \overline{50} = \overline{0110010} = 1001101$$

$$Q' = \overline{129 - Q} = \overline{129 - 29} = \overline{100} = \overline{1100100} = 0011011$$

and the full programming word, W, is:

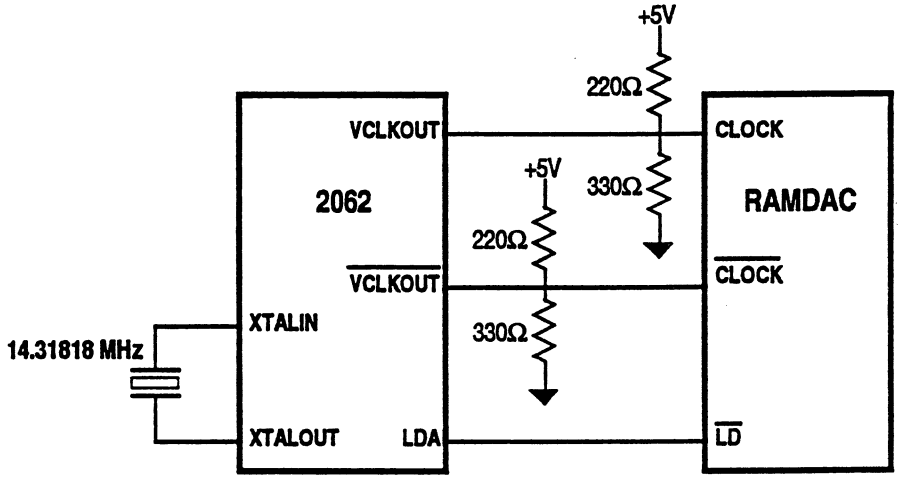
$$W = I, P', M, Q' = 0111, 1001101, 001, 0011011 = 01111001101001011011$$

The programming word W is then sent as a serial bit stream, LSB first. Appropriate address bits and start & stop bits must also be included as defined in the "Serial Programming Scheme" section.

RAMDAC/VRAM Interface

Interfacing to the RAMDAC

The figure below shows how to interface the ICD2062 to a RAMDAC. The part should be located as close to the RAMDAC as possible. Termination resistors are needed on the VCLKOUT outputs, and should be located as close as possible to the RAMDAC.



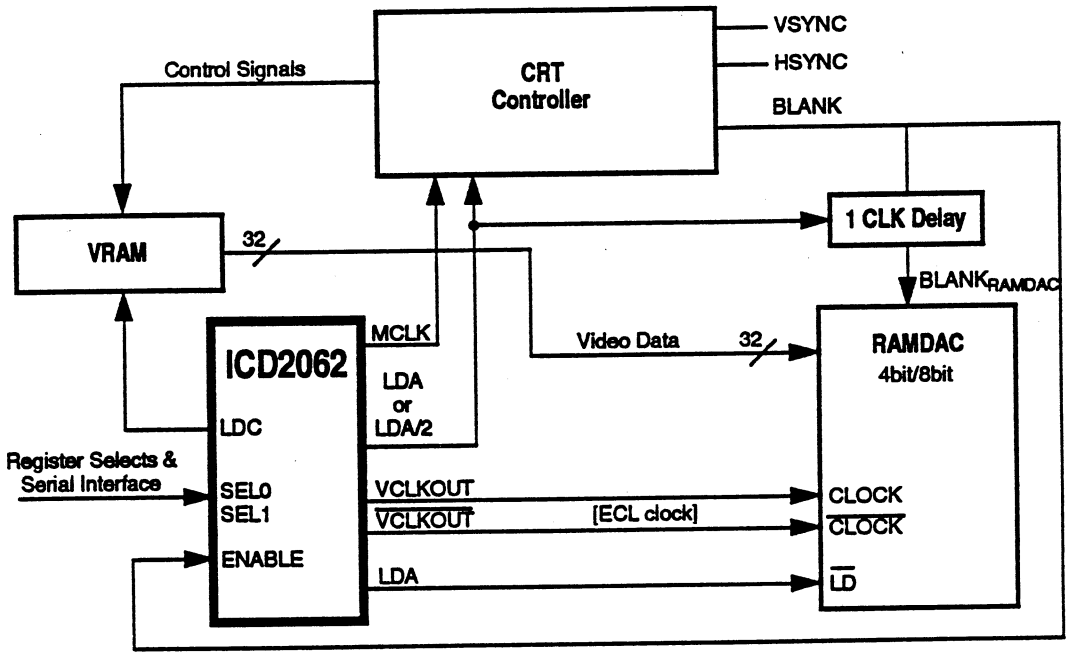
The ICD2062 may drive the CLOCK inputs of up to four RAMDACs, if they are located physically adjacent to each other. In this case, only 2 sets of termination resistors should be used, and these should be located nearest the farthest RAMDAC from the ICD2062.

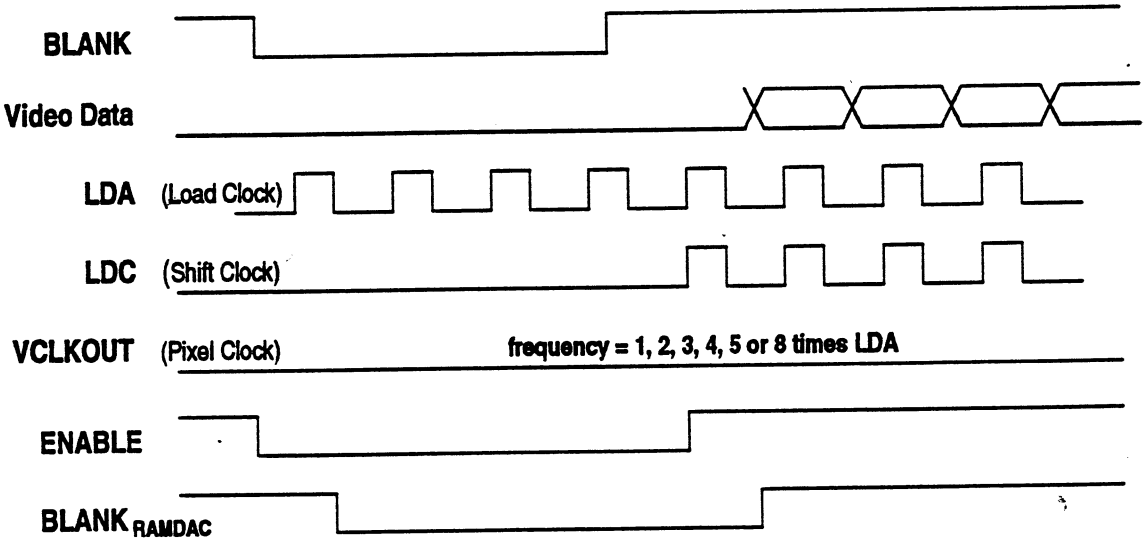
Typical ICD2062 Usage

The DIVREG Register holds the divisor, which can be 1, 2, 3, 4, 5 or 8, by which the pixel clock is divided to generate the load signals: LDA, LDA/2 and LDA/4.

The ENABLE input is synchronized internally to LDA; it may be used to start and stop the LDC output synchronously. When ENABLE is low, LDC is held low. When ENABLE is high, then LDC will be free-running and in phase with LDA. This allows the video DRAM shift registers to be non-clocked during the retrace intervals.

NOTE: For fanouts > 4, LDC needs to be buffered.

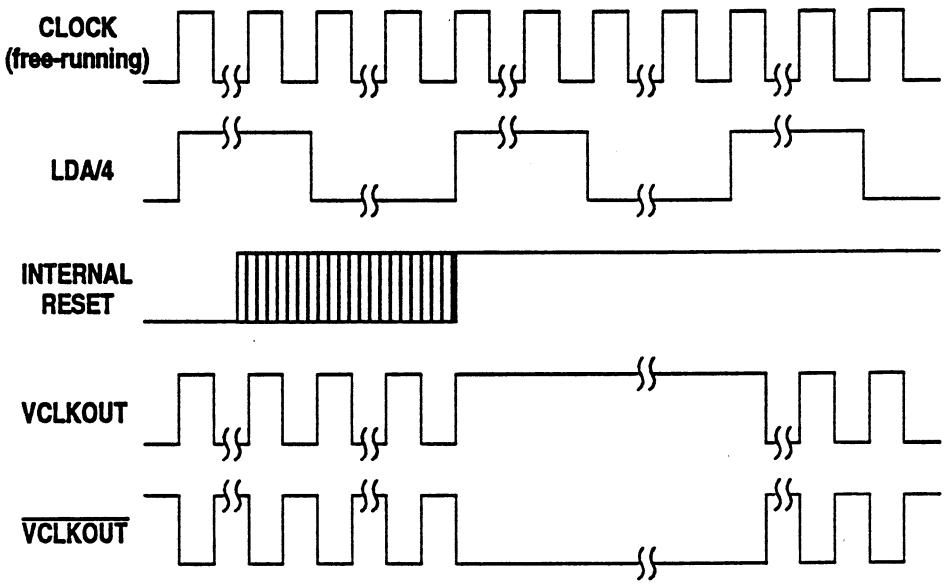

Control Signals
CRT
Controller
VSYNC
HSYNC
BLANK
VRAM
32
1 CLK Delay
BLANK RAMDAC
ICD2062
MCLK
Video Data
32
RAMDAC
4bit/8bit
LDA
or
LDA/2
LDC
Register Selects &
Serial Interface
SEL0
SEL1
ENABLE
VCLKOUT
VCLKOUT
[ECL clock]
LDA
CLOCK
CLOCK
LD


BLANK
Video Data
LDA (Load Clock)
LDC (Shift Clock)
VCLKOUT (Pixel Clock)
frequency = 1, 2, 3, 4, 5 or 8 times LDA
ENABLE
BLANK RAMDAC

Internal RESET Sequence

The internal RESET signal allows the ICD2062 to set the RAMDAC pipeline delay to a specific cycle count, depending on the RAMDAC. Reset takes place the first time the CNTL Register's Reset Bit is set. Following the first rising edge of LDA/4 after the Reset Bit is set, the VCLKOUT and VCLKOUT– outputs are stopped in their respective high and low states; at the next rising edge of LDA/4, these outputs are again allowed to be free-running. The figure below shows the operation of the internal RESET signal:

Internal RESET Timing



Power Management Issues

Estimating Total Current Drain

Actual current drain is a function of frequency and of circuit loading. The operating current of a given output is given by the equation: I = C • V • f, where I=current, C=load capacitance (max. 25pf), V=output voltage (usually 5V for TTL pads, 1.5V for ECL pads), and f=output frequency (MHz).

To calculate total operating current, sum the following:

MCLKOUT	—>	C • V • f(MCLKOUT)	
VCLKOUT	—>	C • V • f(VCLKOUT)	; (ECL pad, V = 1.5V)
VCLKOUT–	—>	C • V • f(VCLKOUT–)	; (ECL pad, V = 1.5V)
LDA	—>	C • V • f(LDA)	
LDA/2	—>	C • V • f(LDA/2)	
LDA/4	—>	C • V • f(LDA/4)	
LDC	—>	C • V • f(LDC)	
Internal	—>	12ma	

This gives an approximation of the actual operating current. For unconnected output pins, one can assume 5–10pf loading, depending on package type.

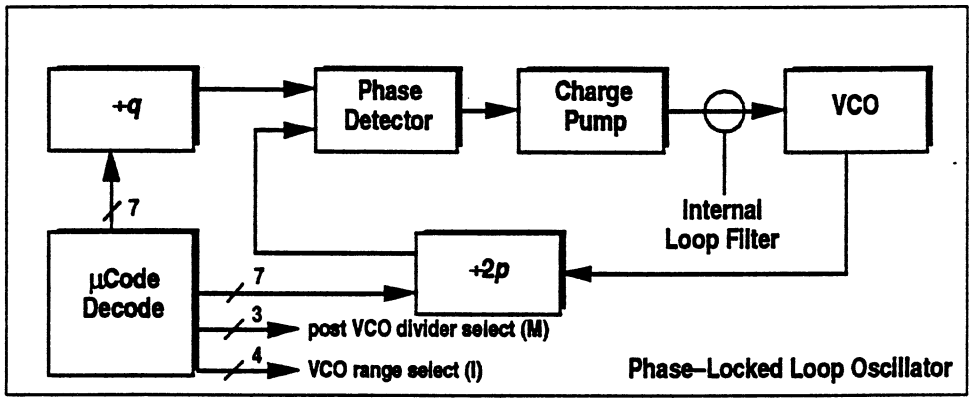
Typical values:

Frequency	Capacitive Load	Current (ma)
low	none	15
high	none	50
high	high	100

Circuit Operation

Circuit Description

The ICD2062 is designed to use an inexpensive TTL crystal and to generate the high-frequency ECL clock signals required by RAMDACs. The VCLKOUT and VCLKOUT– signals interface directly with the RAMDAC CLOCK and CLOCK– inputs. Output levels of the complementary ECL pads are compatible with 10KH ECL circuitry operating from a single +5V power supply.



Each oscillator block is a classical phase-locked loop connected as shown above. The external input frequency $f_{(REF)}$ goes into a "divide-by-n" block. The resultant signal becomes the reference frequency for the phase-locked loop circuitry.

The phase-locked loop is a feedback system which phase matches the reference signal and the variable synthesized signal. The system averages zero phase error between the negative edges arriving at the phase detector. The phase error at the charge pump tells the VCO either to go faster or slower as required. The greater the change in control voltage, the greater the change in the VCO's output frequency. This up and down movement of the variable frequency will quickly lock on to the reference frequency, resulting in an output oscillation as stable as the input reference. An internal loop filter provides stability and damping.

Stability and "Bit-Jitter"

The long-term frequency stability of the IC DESIGNS phase-locked loop frequency synthesis components is good due to the nature of the feedback mechanism employed internally in the design. As a result, stability of the devices is affected more by the accuracy of the external reference source than by the internal frequency synthesis circuits.

Short-term stability (also called "bit-jitter") is a manifestation of the frequency synthesis process. The IC DESIGNS frequency synthesis parts have been designed with an emphasis on reduction of "bit-jitter". The primary cause of this phenomenon is the "dance" of the VCO as it strives to maintain lock. Low-gain VCO's and sufficient loop filtering are design elements specifically included to minimize this "bit-jitter". The IC DESIGNS families of frequency synthesis components are all guaranteed to operate at a jitter rate low enough to be acceptable for graphic designs.

Frequency Range

The frequency range of the video clock VCO is 320 KHz – 160 MHz. The Memory Clock VCO operates in the range of 320 KHz – 120 MHz.

Output Disable

When the OUTDIS– pin is asserted (active low), all the output pins except XTAL OUT and ERROUT– enter a high impedance mode, to support automated board testing.

PC Board Routing Issues

Traditionally, having multiple crystals has allowed the designer to locate them in those places on the board where they are needed. Using a monolithic circuit puts some constraints on the PC board layout to accommodate a single source of all clocks, particularly at frequencies above 50 MHz.

A full power and ground plane layout should be employed both under and around the IC package. The analog power pin (AVDD) should be bypassed to ground with a 0.1µf multi-layer ceramic capacitor and a 2.2µf/10V tantalum capacitor wired in parallel. Both capacitors should be placed within 0.15" of the power pin. A 22Ω resistor placed between the power supply and the AVDD pin can help to filter noisy supply lines. Refer to IC DESIGNS Application Note, *Power Feed and Board Layout Issues*, for more details and for illustrative schematics.

The designer should also avoid routing any of the output traces of the ICD2062 in close parallel proximity. Large routing lengths and large fanouts add capacitance to output drivers. Capacitance affects the rise and fall times of the outputs. Large fanouts should therefore be buffered, particularly for the highest frequencies. When designing with this device, it is best to locate the ICD2062 closest to the device requiring the highest frequency.

FCC & Noise issues

A conscious design effort was made to achieve the optimum rise & fall times at the output pads in order to produce acceptable signals at the clock destinations when operating at high frequencies. Unfortunately, the production of the squarest possible square waves can lead to the generation of high-energy odd harmonics, which can result in extraneous emissions.

For techniques on how to design with this device while taking FCC emission issues into consideration, please refer to the IC DESIGNS Application Note: *Minimizing Radio Frequency Emissions.*

ECL Design Issues

Please refer to the IC DESIGNS Application Note: *ECL Design Guidelines.*

Minimized Parasitic Problems

All of the IC DESIGNS families of frequency synthesis components have been optimized to reduce internal noise and crosstalk problems. To minimize adjacency problems, all the synthesis blocks are physically separated into discrete elements with their output oscillator pins placed on opposite sides of the die. Further, all the synthesis VCO's are separated from their digital logic. Finally, separate power and ground buses for the analog and digital circuitry are used.

Temperature and Process Sensitivity

Because of its feedback circuitry, the ICD2062 is inherently stable over temperature, voltage and manufacturing process variations. Incorporating the loop filter internal to the chip assures that the loop filter will track the same process variations as does the VCO. With the ICD2062, no manufacturing "tweaks" to external filter components are required as is the case with external "de-coupled" filters.

Order Entry Information

Part Number	Package Type	Temperature Range	Chip Options
ICD2062	P = 20–Pin Plastic DIP	C = Commercial (0°C – +70°C)	–1
	C = 20–Pin Ceramic DIP		(Other ROM options are available by special order)
	S = 20–Pin SOIC		

Example: order *ICD2062PC* for the ICD2062, 20–pin plastic DIP, commercial temperature range device with the initial frequencies shown in the Register Initialization Table.

Device Specifications

Electrical Data

Maximum Ratings

Name	Description	Min	Max	Units
VDD	Supply voltage relative to GND	–0.5	7.0	Volts
V_{IN}	Input Voltage with respect to GND	–0.5	VDD + 0.5	Volts
T_{OPER}	Operating Temperature	0	+70	°C
T_{STOR}	Storage Temperature	–65	+150	°C
T_{SOL}	Max Soldering Temperature (10 sec)		+260	°C
T_J	Junction Temperature		+125	°C
P_{DISS}	Package Power Dissipation		375	mWatts

DC Characteristics

VDD = +5V ±5%

$0°C \leq T_{CASE} \leq +70°C$

Name	Description	Min	Typ	Max	Units	Conditions
V_{IH}	High-level input voltage	2.0			Volts	
V_{IL}	Low-level input voltage			0.8	Volts	
$V_{OH(ECL)}$	ECL High-level output*	VDD–1.0		VDD–0.8	Volts	
$V_{OL(ECL)}$	ECL Low-level output*	VDD–2.0		VDD–1.6	Volts	
$V_{OH(TTL)}$	TTL High-level output‡	2.4			Volts	IOH= –4ma
$V_{OL(TTL)}$	TTL Low-level output‡			0.4	Volts	IOL= 4ma
I_{IH}	Input high current			100	µa	V_{IH} = 5.2V
I_{IL}	Input low current			–250	µa	V_{IL} = 0V
I_{OL}	Output leakage current			10	µa	
I_{CC}	Power supply current	15		100	ma	
I_{CC-TYP}	Power supply curr. (typical)		45		ma	@ 60 MHz
C_{IN}	Input Capacitance			10	pf	
$C_{OUT(ECL)}$	Output Capacitance		7		pf	

* ECL outputs: VLKCOUT, VCLKOUT–

‡ TTL outputs: MCLKOUT, LDA, LDA/2, LDA/4, LDC, ERROUT–

Electrical Data

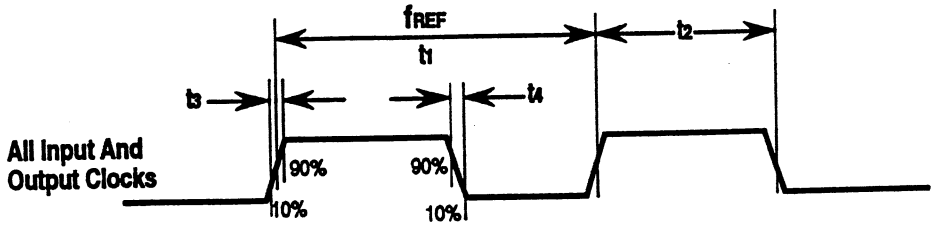
AC Characteristics

VDD = +5V ±5%

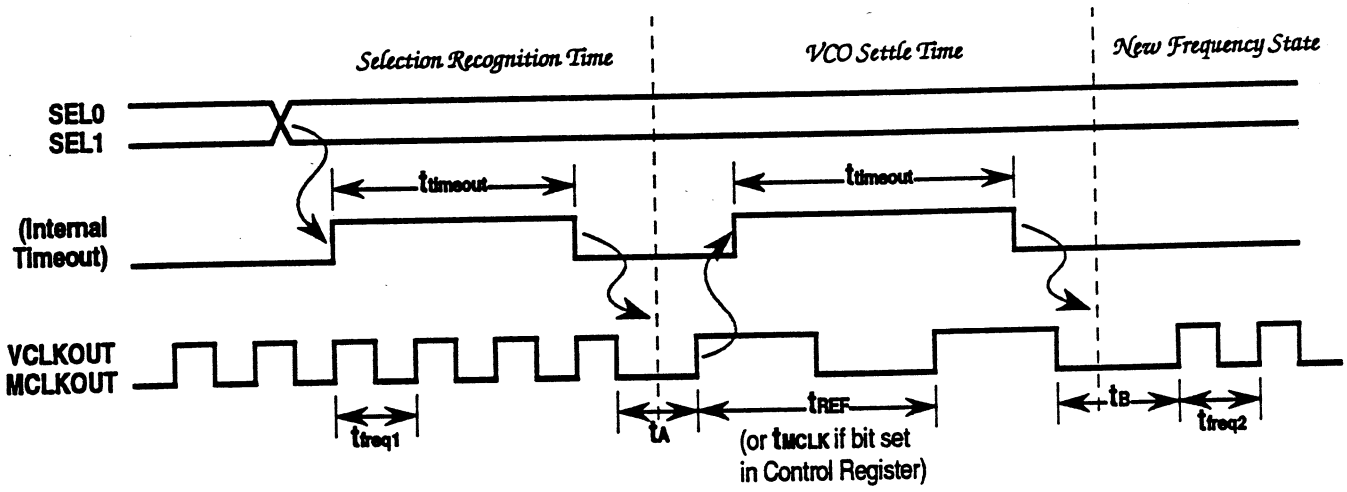
0°C ≤ T_{CASE} ≤ +70°C

Symbol	Name	Description	Min	Typ	Max	Units
f_{REF}	reference frequency	Reference Oscillator nominal value (Note: for references of other than 14.31818 MHz, the pre-loaded ROM frequencies will not be accurate.)	1	14.31818	60	MHz
t_1	output clock period		6.25		3000	ns
t_2/t_1	duty cycle	Duty cycle for the output oscillators (Note: duty cycle is measured at CMOS threshold levels. At 5 volts, V_{TH} = 2.5 Volts.)	45%		55%	
t_3	rise time	Rise time for the output oscillators into a 25 pf load			3	ns
t_4	fall time	Fall time for the output oscillators into a 25 pf load			3	ns
t_{REF}	reference period	$1 / f_{REF}$	16.6		1000	ns
t_{freq1}	freq1 output	Old frequency output				
t_{freq2}	freq2 output	New frequency output				
t_A	f_{REF} mux time	Time clock output remains low while output muxes to reference frequency	$1/2\ t_{REF}$		$3/2\ t_{REF}$	
$t_{timeout}$	timeout interval	Internal interval for serial programming and for VCO changes to settle. If the interval is too short, see the timeout interval section in the control register definition.	2	5	10	ms
t_B	t_{freq2} mux time	Time clock output remains low while output muxes to new frequency value	$1/2\ t_{freq2}$		$3/2\ t_{freq2}$	
t_5	tri-state	Time for the output oscillators to go into tri-state mode after OUTDIS– signal assertion	0		12	ns
t_6	clk valid	Time for the output oscillators to recover from tri-state mode after OUTDIS– signal goes high	0		12	ns
fmax		VCLKOUT clock rate			160	MHz
$t_{SKEW-LDA}$		VCLKOUT to LDA output skew	2		6	ns
$t_{SKEW-LDA/2}$		LDA to LDA/2 output skew	0	1	2	ns
$t_{SKEW-LDA/4}$		LDA to LDA/4 output skew	0	1	2	ns
$t_{SKEW-LDC}$		LDA to LDC output skew	0	1	2	ns
t_{EN-SU}		ENABLE setup time to LDA	12			ns
t_{EN-HD}		ENABLE hold time to LDA	0			ns
t_{serclk}		Clock period of serial clock	$2 / t_{REF}$		1	ms
t_{SU}		Setup time	20			ns
t_{HD}		Hold time	10			ns
t_{ldcmd}		Load command	0		$t_1 + 30$	ns

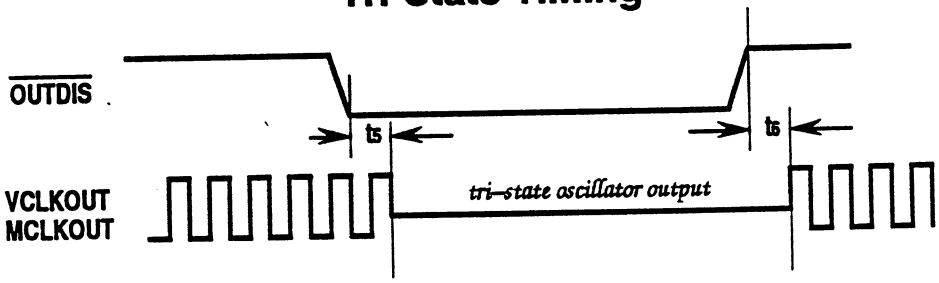
Rise and Fall Times



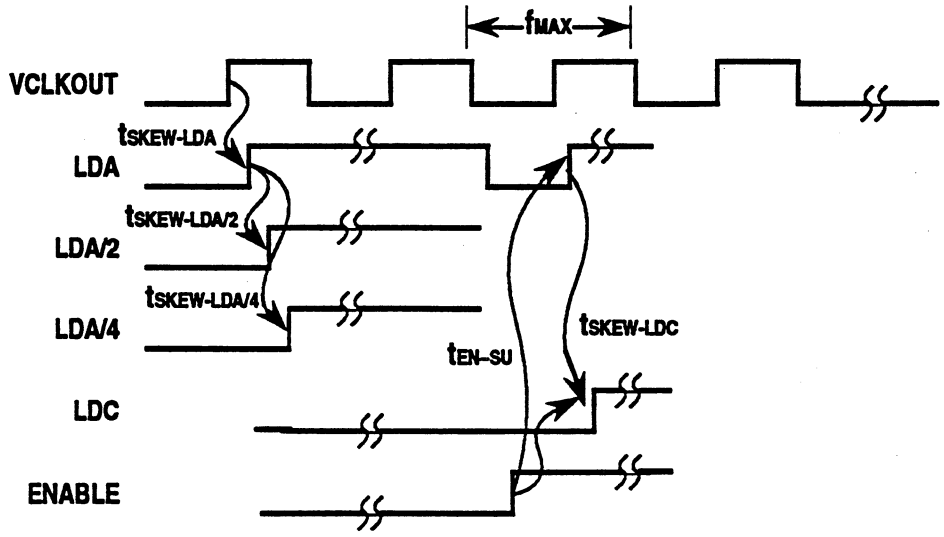
Selection Timing



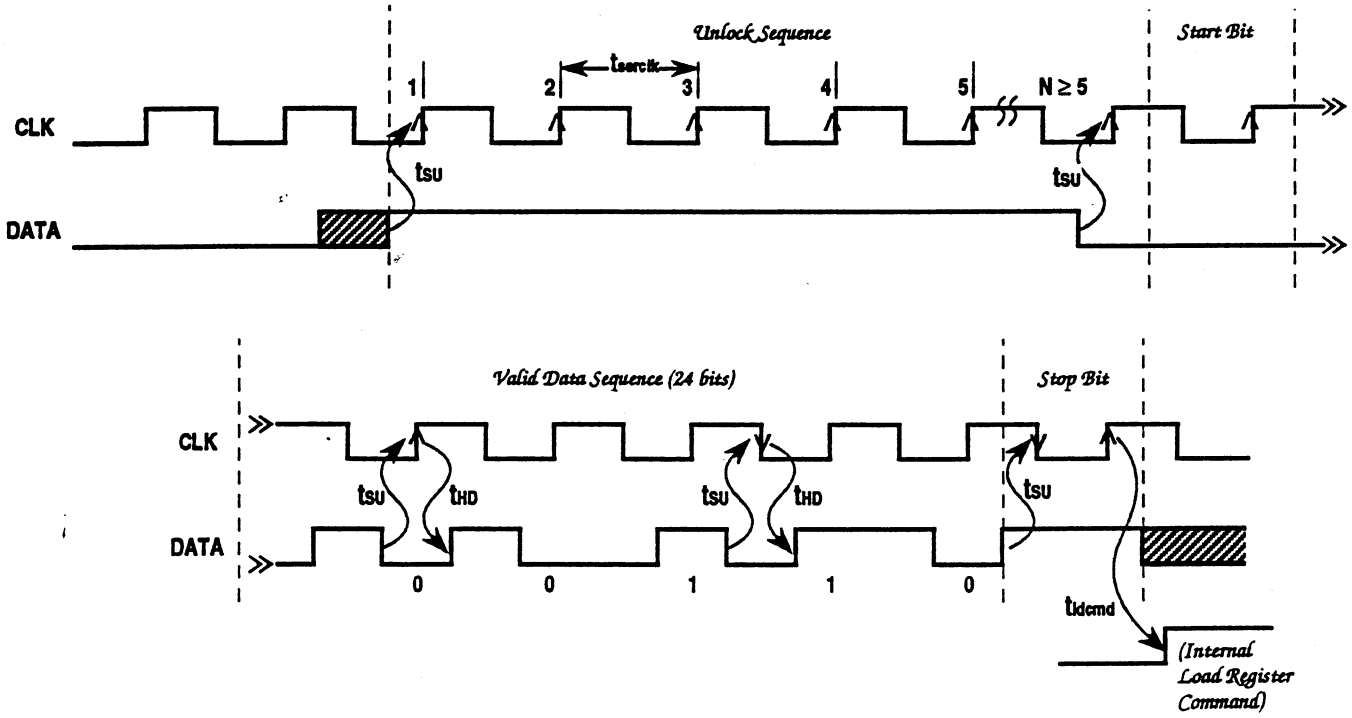
Tri-State Timing



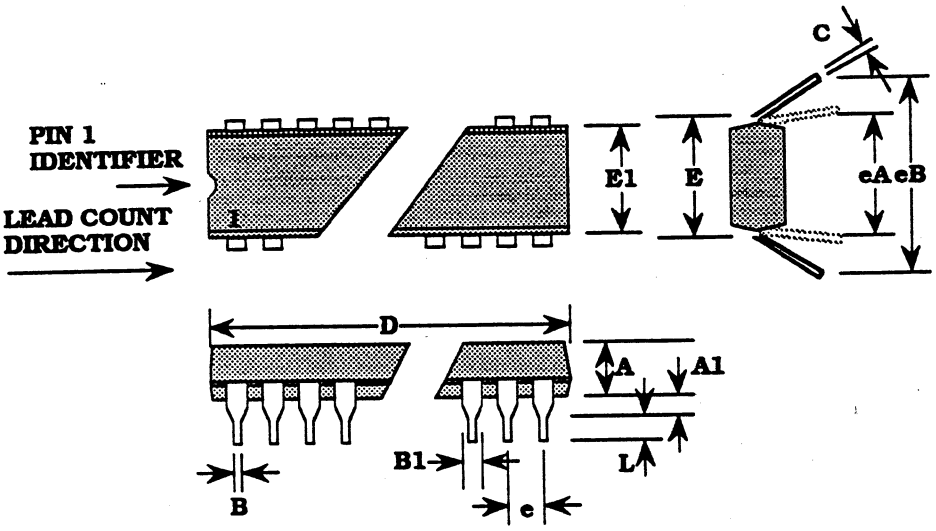
RAMDAC / VRAM Interface Timing



Serial Programming Timing



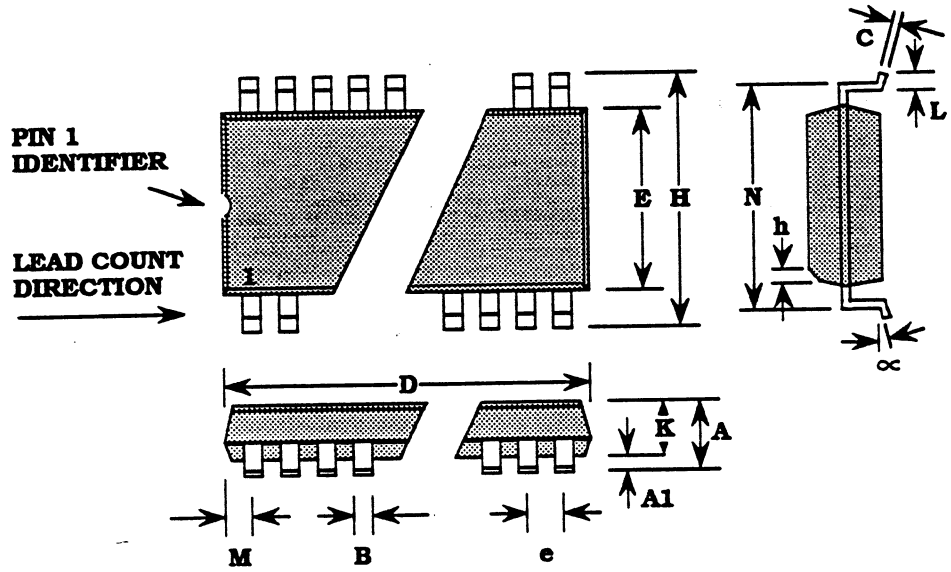
Packaging Information



PDIP Outline

	Lead Count 20	
SYMBOL	MIN	MAX
A	.195	.200
A1	.015	–
B	.015	.020
B1	.050	.070
C	.008	.012
D	.925	1.030
E	.290	.310
E1	.230	.280
e	.100	REF
eA	.290	–
eB	–	.310
L	.100	–

(Dimensions in Inches)



SOIC Outline

	Lead Count 20	
SYMBOL	MIN	MAX
A	.099	.104
A1	.004	.009
B	.014	.019
C	.010	REF
D	.505	.512
E	.294	.299
e	.050	TYP
H	.402	.419
h	.025 x 45°	
L	.030	.040
∝	0°	8°
K	.088	.098
M	.020	.030
N	.335	.351

(Dimensions in Inches)

IC DESIGNS
12020 — 113th Ave NE
Kirkland, WA 98034
Tel: 206 / 821 – 9202
Fax: 206 / 820 – 8959