W32p

ET4000/W32p Graphics Accelerator

Data Book

GRAPHICS BY

TSENG

LABS

I

GRAPHICS BY
TSENG
LABS

ET4000/W32p

Data Book Errata

7/7/94

Data Book Publication Date 3/94

---

1. 4/11/94 - Figure 2.12-2 ET4000/W32p Queued Registers and ACL Internal State found on page 30 is incorrect. The correct figure appears below.

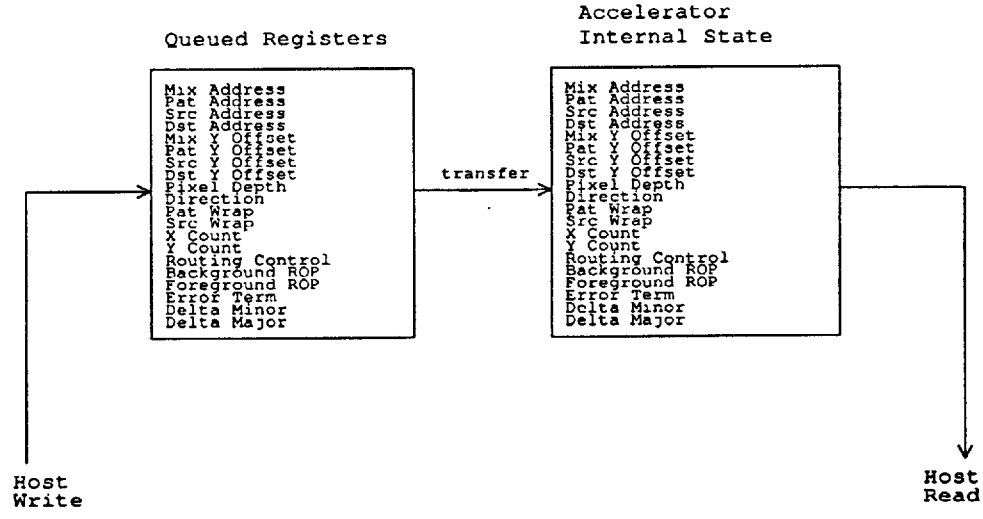


Figure 2.12-2 ET4000/W32p Queued Registers and ACL Internal State

11

2. 4/11/94 - The bit descriptions for bits <4> and <3> for the ACL Routing Control Register, address 9C and found on page 175, are missing information. The correct descriptions for these bits follow.

5.9.21 ACL Routing Control Register (cont'd)

Bit	Description
Bit 4	When set to 1, the MixMap is used as a "byte-enable" mask to control which bytes of the Destination Map will be modified. The MixMap is used in this way only if the Background Raster Operation is set to AA ("no-operation") and the Foreground Raster Operation is not a function of the Destination Map. This method of operation allows for faster drawing of transparent BitBlt's since it avoids reading the Destination Map data from the frame buffer. This bit may be set to 1 at all times; the only need to set it to 0 would be if the host wished to read Destination data from the accelerator using a transparent operation.
Bit 3	This bit controls whether the Mix Map participates in the current accelerator operation. When set to 1, the Mix Map data is taken from the CPU if the DARO field is set to 10. Otherwise, the Mix Map data is taken from the display memory, using the ACL Mix Address Register to determine the location of the map in display memory. When set to 0, the Mix Map data is taken from the CPU if the DARO field is set to 10. Otherwise, the Mix Map is not used (as if Mix data were fixed to "1"), and the accelerator applies the Foreground Raster Operation to all bytes of the operation.

3. 4/20/94 - Chip Revision IDs, found in section 5.6.14 CRTCB/Sprite Row Offset High (Index: EC) on page 148, are added for W32i Rev. C and W32p Rev. D. Bits <7:4> are now defined as follows:

5.9.21 CRTCB/Sprite Row Offset High Register (Index: EC)

Bit	Description
Bits 7:4	These bits are used to indicate the chip and revision level. The values are defined as follows:

Bit 7 6 5 4	Chip/Rev.
0 0 0 0	W32
0 0 0 1	W32i
0 0 1 0	W32p, Revision A
0 0 1 1	W32i, Revision B
1 0 1 1	W32i, Revision C
0 1 0 1	W32p, ***Revision B***
0 1 1 1	W32p, ***Revision C***
0 1 1 0	W32p, ***Revision D***
↓ ↓ ↓ ↓	
1 1 1 1	Reserved

111

4. 6/10/94 - The bit descriptions for bits <7> and <3> for Input Status Register One, address 3BA/3DA and found on page 95, are changed. The correct descriptions for these bits follow.

5.1.3 Input Status Register One

I/O address = 3BA (mono)/3DA (color)

Bit	Description	Access
7	Vertical retrace complement.	RO
6	CRTCB vertical display enable.	RO
5:4	Video display feedback test.	RO
3	Vertical retrace.	RO
2	CRTCB display enable.	RO
1	Horizontal Display Enable complement.	RO
0	Display enable complement.	RO

Bit Description

Bit 7 A value of 0 will be read during the vertical sync pulse; 1 otherwise. (See Figure 2.2-1)

Bit 6 A value of 1 indicates that the CRTCB window is active within the current scan line.

Bits 5:4 Used for diagnostic purposes. They are selectively connected to two of the eight color outputs of the Attribute Controller. The Color Plane Enable register (ATC Indexed Register 12) controls the multiplexer for the video wiring. Available combinations are:

Color Plane Register Bits		Input Status Register One Bits	
5	4	5	4
0	0	P2	AP0
0	1	P5	AP4
1	0	P3	AP1
1	1	P7	AP6

Bit 3 A value of 1 will be read during the vertical sync pulse; 0 otherwise.

Bit 2 A value of 1 indicates that the CRTCB window is active.

Bit 1 A value of 0 indicates CRTC horizontal display enabled.

Bit 0 A value of 1 indicates a vertical or horizontal retrace interval and is the real-time status of the inverted display enable signal.

IV

5. 6/10/94 - The bit description for bits <3:2> for GDC Indexed Register 6, address 3CF and found on page 132, is changed. The correct description for these bits follow.

5.4.10 GDC Indexed Register 6: Miscellaneous

I/O address = 3CF

Bit	Description	Access
7:4	Reserved (= 0).	
3:2	Memory map.	RW
1	Enable odd/even mode.	RW
0	Graphics mode enable.	RW

Bit Description

Bits 3:2 Memory Map—Control mapping of the Frame Buffer into CPU address space.

See Section 7.3 for the effect this bit has on the Video Memory Map.

Bit 1 When set to 1, enables odd/even mode, will cause the replacement of the CPU address bit 0 with a high-order bit, and the odd/even maps are "chained" via the CPU A0 bit.

Bit 0 When set to 1, enables graphics mode.

V

6. 6/14/94 - The frequency specifications for the Extension to INT 10, BIOS function AH=012 have changed. These are found in Section 8. Programming Interface on page 233. The updated specifications follow.

Extension to INT 10 Bios function AH=012:

BL=0F1 Set/Get Frequency Type

This function sets or gets the currently-selected frequency type (i.e. 60 Hz, 72 Hz, etc) for modes of a particular resolution.

Input:
AL=0: set frequency type
AL=1: get frequency type
BH=code for screen resolution group:
0=640x480
1=800x600
2=1024x768
3=1280x1024
if AL=0, CX=code for frequency type:
640x480: 0=60 Hz, 1=72Hz, 2=75Hz, 3=90Hz
800x600: 0=56 Hz, 1=60Hz, 2=72Hz, 3=75Hz, 4=90Hz
1024x768: 0=43.5Hz (interlaced), 1=60Hz, 2=70Hz, 3=75Hz
1280x1024: 0=43.5Hz (interlaced), 1=60 Hz, 2=70Hz, 3=75Hz

Output:
AL=012 (can be used to check whether this function is present)
CX=code for frequency type currently set (at the end of the call)

Notes:
1. Some frequency types are not available on all boards. (Check the returned frequency type to see if a specified frequency type got set or not.)
2. In some cases, a particular mode might get set at a lower frequency than indicated by the frequency type value returned from this function due to a bandwidth limitation (such as with a HiColor mode).
3. A few boards have alternate methods of determining the frequency to use.

7. 7/7/94 - The following registers are protected by the KEY but are not indicated as such in the publication:

Section	Page	Register
5.2.28	113	CRTC Indexed Register 30: Address Map Comparator
5.2.29	114	CRTC Indexed Register 31: General Purpose
5.2.36	120	CRTC Indexed Register 3F: Horizontal Overflow

V1

Life Support Disclaimer

Tseng Labs, Inc. does not sanction, authorize, or certify any chip-level or board-level products for use in the design, construction, or use in or as critical components of life-support systems or devices. Products of this nature are defined as follows:

a) Life-support systems and/or devices are those which are used to support/ sustain life, or are used to the same effect when applied as a surgical implant, and whose failure to perform may result in serious injury or fatality.

b) Critical components are those which are integral and essential to the design and construction of life-support devices which are used to support/sustain life, and whose failure to perform can be expected to significantly reduce the effectiveness of said devices, which may result in serious injury or fatality.

Contents

Figures

Tables

1. Introduction

The Tseng Labs ET4000/W32p video controller is a PCI/Local Bus-compatible graphics chip that delivers an 8-/16-/32-bit bus or CPU direct (local bus) interface. It features a Graphical User Interface (GUI) Accelerator, and advanced features for the developing imaging and multimedia markets. The host interface is the second generation of Tseng Labs cache/memory management architecture, and features performance that is approximately five times greater than its predecessor, the ET4000/AX. A unique feature of ET4000 architecture is that the design maximizes the capabilities of DRAM, and eliminates the requirement for VRAM. The ET4000/W32p will allow DRAM designs to significantly outperform competing and more costly VRAM 2D-based solutions, and supports from 512KB to 4MB memory configurations. The highest performance design requires high-speed data transfer from host to Accelerator utilizing a zero wait state 32-bit CPU direct connection, while providing enhanced Windows performance for graphics functions such as BitBLT, raster operations, pixel amplification™, and cursor control. The ET4000/W32p provides 32 multiplexed address/data lines, and can be connected to the processor via the local bus. The processor direct interface is most useful for applications such as live video decompression, and bit-map intensive applications such as true-color that require maximum bandwidth to the display memory.

GUI Optimization

The Graphics Accelerator design provides distribution of graphics functions from the CPU to the video controller, freeing the CPU to return to other tasks. Among the features are BitBLT, full 256 Raster Operation support, hardware cursor support or a second simultaneous display window, Imaging Port, memory-to-memory BLTs with masking, and pixel amplification™. The W32p is designed to optimize GUI application functionality and performance, speeding the operation of applications like Microsoft Windows significantly. Additionally, the W32p operates in all graphics modes, whether they be planar, or linear packed pixel modes. Among the supported modes are 1, 4, 16, 256, 32K, 64K, and 16.7 million colors. Pixel depths may be up to 32 bits per pixel. True-color modes (16.7 million colors) are optimized in the ET4000/W32p, providing the desktop computer market with the highest quality 2D images.

Multimedia and Imaging

Multimedia and imaging are enhanced through inclusion of a unique Image Memory Access port (IMA) that allows direct access to the frame buffer at high speed. An 8-bit port into the memory can be opened allowing images to be displayed on screen in real time, at color depths of up to 16.7 million simultaneous colors. Additionally, the second hardware display window (CRTCB) may be overlaid and resized up to the size of the display, creating an effective second simultaneous dual-window display. The frame buffer size can be up to 4 megabytes.

Future Display Standards

Up to two separate linear frame buffers may be displayed simultaneously by the ET4000/W32p. Previous SuperVGA drivers require the programmer to work in segments aligned on 64KB boundaries, requiring additional CPU overhead for managing segment crossings. This results in a reduction in performance. Advanced designs using 1-, 2-, or 4MB segments can write to any point in the video memory within a single operation. As a result, the simplified drivers create the maximum possible system performance for microcomputers.

The ET4000/W32p is packaged in a 208-pin QFP (Quad Flat Package) configuration.

1.1 ET4000/W32p Specifications

The following is an outline of the ET4000/W32p graphics controller specifications.

I. Input Interface
- A. Host Processor:
  1. Data Bus: 8-, 16-, or 32-bit memory and I/O bus
  2. Address: 20-, 22-, 24-bit address bus
  3. Bus Control: PCI or Local Bus
- B. IMA Port
  1. Data Bus: 8-bit
  2. Address: Implied (internally generated via programming)
  3. Bus Control: Tseng IMA bus protocols

II. Output Interface
- A. Monitor:
  1. Interlaced or non-interlaced V-SYNC and H-SYNC with polarity control
  2. 16-bit pixel output AP<15:0>
- B. External DAC look-up: pixel clock, blanking, and external DAC read/write decode controls

III. Resource Management
- A. Memory Management:
  1. Graphics Data Controller: VGA compatible data rotate/mask/logical functions
  2. Cache: Proprietary enhanced-LRU replacement policy and block transfer
  3. FIFO: up to two FIFOs for display pixel data
  4. Memory Control Unit (MCU):
     a. memory types: DRAM—fast page mode only
     b. memory size: 512KB example: (4) 256K x 4
        1MB example: (8) 256K x 4;
        2MB example: (16) 256K x 4 (Interleave);
     c. memory timing: programmable RAS/CAS timing in terms of system clock (independent of display clock)
- B. System Priority Controller (SPC): Intelligent SPC to resolve multiprocessors and ET4000/W32p resource requests to optimize MCU resource utilization.
- C. CRT Controllers (primary and secondary (CRTC, CRTCB)):
  1. Horizontal: 9-bit programmable display enable, blanking, and H-SYNC
  2. Vertical: 11-bit programmable line counter for display enable, blanking, split screen, and V-SYNC
- D. Display address controller:
  1. Linear Address Generator: 20-bit linear doubleword address with programmable starting address, row address offset
  2. Row Address Generator: 5-bit row scan address provides up to 32-line character height
  3. Hardware Cursor: a 64x64x2 sprite creating a second hardware window for simultaneous display of graphics, or even full motion (30 frame/second) digital video.
  4. Cursor: 20-bit cursor position and 5-bit cursor start, 5-bit cursor end control
- E. Attribute controller:
  1. Text: support for up to two 256 character sets;
     IBM-compatible text attribute decoding;
     IBM-compatible cursor blink/underline;
     AT&T-compatible underline decoding (in color text mode)
     Font width: programmable text font width: 6, 7, 8, 9, 10, 12, and 16 pixel

2. Graphics: plane graphics, linear-byte/word (packed pixel), monochrome and CGA color graphics format; VGA-compatible color LUT (look-up table)

F. Timing Interface: select up to 8 MCLK (pixel clock): 86MHz (graphics), 56MHz (text); system clock, 50MHz

G. Graphics Accelerator:

1. Data Path: 32-bit with 256 Raster Operation support
2. Address: Destination/Source/Pattern maps up to 4MB addressing
3. Functions: D/S/P BLT, tiling, pattern, FG/BG ROP, and CPU-assisted operations

IV. Display Data Format

A. Plane, Linear byte, Linear word graphics and VGA-compatible text format up to 16-bit wide character

B. Display Capability:

1. Resolution: up to 1280x1024 in 256 colors; 1024 x 768 in 65,536 colors; and 800x600 True Color (16.8 million colors) non-interlaced in graphics mode
2. Pixel Clock Rate: graphics mode, 86MHz; text mode, 56MHz

V. Compatibility

A. Register level: EGA/VGA

B. Display level: 8514A

C. Monitor: VGA, SuperVGA, and variable-frequency monitors with up to 1024 x 1024-pixel resolution and above

The ET4000/W32p offers performance starting with all of the IBM VGA/EGA features and provides enhanced performance with features summarized below.

1.2 ET4000/W32p Overview

- IBM Video Graphics Array (VGA) and Enhanced Graphics Adapter (EGA) register-level compatibility.
- Supports interface to 386/486 SX/DX local bus and PCI bus, with data bus width up to 32 bits.
- All display memory can be read from or written to with minimum wait states.
- Memory Management Unit (MMU) allows the host to access any location in the full 4MB range of display memory through any of three independent apertures.
- Internal Multiport Cache™ is capable of serving multiple masters. Host, Image Port, and internal Graphics Accelerator can all access the cache simultaneously. Multiport Cache paves the way for multiprocessor systems.
- Graphics Accelerator provides hardware drawing assist functions, such as bit block transfers (BitBlt), patterned lines, circles, and so forth.
- Pixel Amplification™ speeds text painting, color expansion, and rectangular area fill operations up to 8 times the speed of CPU-based processing.
- All 256 Raster Operations are included in hardware, freeing the CPU from time-consuming data manipulation.
- High-speed 8-bit Image Port allows data-intensive sources that cannot be transmitted across the system bus—a cost-efficient means of accessing display memory.
- Supports 64x64x2 sprite for use as hardware cursor.
- Secondary CRT Controller can be used to display picture-in-picture graphics or even full motion (30 frame/second) digital video.
- Supports Pixel Clock frequencies up to 86MHz.
- Supports resolutions up to 640x480 at 24 bits per pixel in interlaced format.
- 20-bit linear start and cursor address supports display from any location in the 4MB display memory.
- Split-screen feature, allows second independent window starting at display memory address zero. Split-screen window occupies full width of screen, with starting scan line software-selectable. Panning can be disabled within a split screen.
- Support for traditional VGA RAMDACs, HiColor DACs, and True Color DACs, and new 16-bit RAMDACs.
- Provides all controls for host interface to external RAMDAC, even on the local bus.
- Programmable memory timing allows connection to dynamic RAMs of any speed.
- Programmable CAS before RAS refresh, providing memory refresh during display blanking.
- Enhanced Memory Control Unit supports interleaved frame buffer memory for a 70 to 80% increase in available bandwidth.

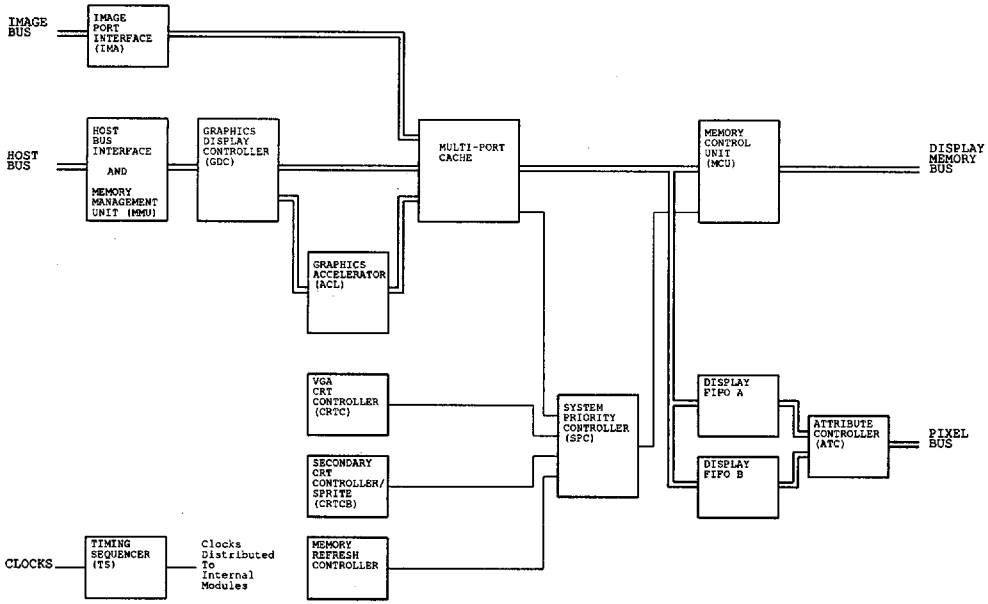
1.3 Notational Conventions

Some of the terms, acronyms, and abbreviations used in this document are defined in the following:

bpp	Bits per pixel
Byte	8 bits
CGA	Color Graphics Adapter
CPU	Central Processing Unit
CRTC	Cathode Ray Tube Controller (Primary video display)
CRTCB	Secondary video display
DAC	Digital-to-Analog Converter
Doubleword	4 bytes
DRAM	Dynamic Random-Access Memory
EGA	Enhanced Graphics Adapter
EISA	Extended Industry Standard Architecture
GDC	Graphics Display Controller
GUI	Graphical User Interface
IMA	Image Memory Access port (Image Port)
ISA	Industry Standard Architecture
Kb	Kilobit
KB	Kilobyte
MB	Megabyte
Pixel	Picture Element
Quadword	8 bytes
RAM	Random Access Memory
ROM	Read-Only Memory
ROP	Raster Operations
TS	Timing Sequencer
VESA	Video Electronic Standards Association
VGA	Video Graphics Array
Word	2 bytes

2. ET4000/W32p Functional Description

All major elements of the ET4000/W32p are contained within a single 208-pin Quad Flat Package. The block diagram below shows the internal architecture of the ET4000/W32p. The following sections provide a breakdown of the major elements of the chip.



ET4000/W32p Block Diagram

2.1 Host Interface - PCI (Peripheral Component Interconnect)

2.1.1 Configuration Space Organization

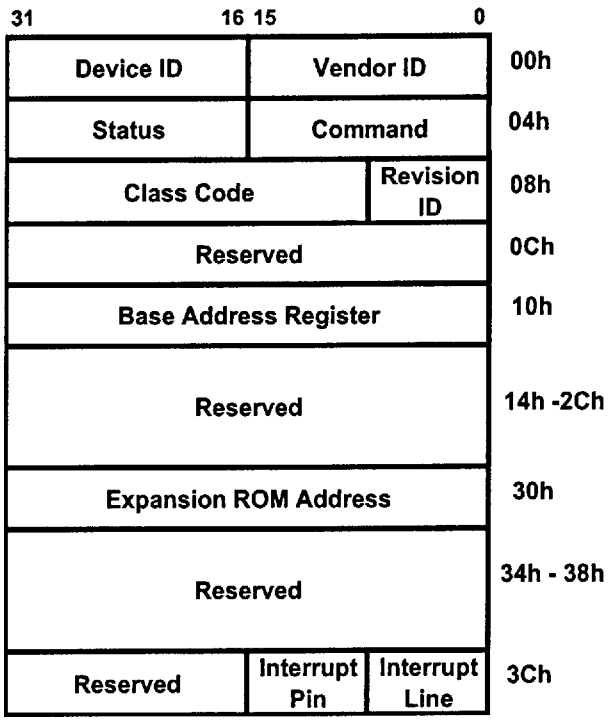
A 256-byte space restriction is placed on the specific record structure, which is divided into predefined header and device-dependent regions. The configuration space for a device must be accessible at all times, including during system boot.

Every device, including the W32p, must support the register layout of the 64-byte predefined header region. The region consists of fields that uniquely identify the device and allow it to be controlled generically. The remaining 192 bytes are device-specific.

In order for system software to scan the PCI bus to determine what devices are present, the vendor ID must be read in each PCI slot. The host bus to PCI bridge must unambiguously report attempts to read the vendor ID of non-existent devices. As 0FFFFh is an invalid vendor ID, it is sufficient for the host bus-to-PCI bridge to return a value of all 1's on read accesses to configuration space registers of non-existent devices. (Note that these accesses will be terminated with a master-abort.)

PCI devices must treat configuration space write operations as no-ops, meaning that the access must be completed normally on the bus and the data discarded. Read access to reserved or unimplemented registers must be completed normally and a data value of 0 returned.

Figure 2.1-1



The figure above illustrates the layout of the 64-byte predefined header portion of the 256-byte configuration space that must be supported by all PCI devices. Device-specific registers must be placed in locations 64 through 255 only. The lower addresses of multi-byte fields must contain the least significant parts of the field. Some fields contain bit-encoded reserved bits of no particular value, requiring software to use appropriate masks to extract defined bits. Meaning, the values of reserved bit positions must first be read, merged with the new values for other bit positions, and the data then written back.

The predefined header portion of the configuration space is divided into 2 parts. The first part consists of 16 bytes that are defined the same for all types of devices. The layout of the remaining 48 bytes can differ according to the base function of the supporting device.

PCI devices are required to carry Vendor ID, Device ID, Command, and Status fields in the header. Implementation of the other registers is optional, depending on device functionality. A device that supports the function that is defined by the register must implement it in the defined location and with the defined functionality.

2.1.2 Configuration Space Functions

PCI can potentially increase the ease of configuring systems as long as PCI devices provide certain functions that system configuration software can employ. Listed in the following sections are functions requiring support by PCI device register as defined in the predefined header portion of the configuration space. The exact format of the registers is germane to the device, though some rules are universal. Registers must be capable of being read back with the data returned indicating the value that the device is actually using.

Use of the configuration space should be restricted to initialization and error handling software, and is intended for configuration, initialization and catastrophic error handling functions. Device registers are manipulated by operational software's use of I/O and /or memory space accesses.

2.1.3 Device Identification

The following fields in the predefined header deal with device identification. PCI device are required to implement them so that generic software is able to easily determine available devices on the PCI bus. These registers are read-only.

Vendor ID - Identifies manufacturer of the device. Valid vendor identifiers are allocated by the PCI SIG to ensure uniqueness. The value 0FFFFh is not valid for vendor ID. The value assigned to Tseng Labs is 100C.

Device ID - Indicates specific device. The ID is allocated by the vendor. Tseng Labs incorporates the Revision ID in the last 4 bits of this field (320x where x is the 4 ID bits read from CRTCB Indexed Register EC <7:4>). Revision ID is the same as found in CRTCB/Sprite Row Offset High Register (Index EC), bit <7:4>.

Header Type - Identifies the layout of bytes 10h through 3Fh in the configuration space and specifies if the device contains multiple functions. Bit 7 is equal to 0 indicating the W32p is a single-function device. Bits 6-0 specify the layout of bytes 10h through 3Fh. The 00h encoding is illustrated in Figure 2.1-1 and all others are reserved.

Class Code - This read-only register identifies the generic function of the PCI device and can indicate a specific register-level programming interface. It is laid out in 3 byte-sized fields consisting of an upper byte at offset 0Bh, a middle byte at offset 0Ah, and a lower byte at offset 09h. The upper byte is a base class code indicating the type of function the device performs. The middle byte more specifically identifies the function of the device, and the lower byte is used to specify a particular register-level programming interface, if any, to facilitate interaction with device-independent software.

Base Class	Definition
00h	Class Code not finalized before device built
01h	Mass storage controller
02h	Network controller
03h	*Display controller*
04h	Multimedia device
05h	Memory controller
06h	Bridge device
07h-FEh	Reserved
FFh	Device does not fit in any defined classes

2.1.3.1 Base Class 03h

This base class is used to define display controllers. Here, the W32p is specified. No specific register-level programming interfaces are defined.

Base Class	Subclass	Progrmg. Interface	Definition
	00h	00h	VGA-compatible controller
03h	01h	00h	XGA controller
	80h	00h	Other display controller

The W32p is specified as a XGA-compatible controller returning a value of 01h and a programming interface value of 00h. ***Except for Revision A***, the W32p is specified as a VGA-compatible controller returning a value of 00h and a programming interface value of 00h.

2.1.4 Device Control

2.1.4.1 Command Register

Configuration space address 04h

This register provides undetailed control over a device's capability to generate and respond to PCI cycles. A zero written to the Command Register disconnects a device from the PCI bus except for configuration accesses. Bits that do not pertain to the device's functionality may not be used. For example, if a device does not implement an I/O space, it is not likely to implement a writeable element at bit location 0 of the Command Register.

Bit	Description	Access
15:10	Reserved.	
9:8	Reserved (always set to 0).	
7	Wait cycle control.	RW
6	Parity error response.	RW
5	VGA palette snoop.	RW
4	Reserved (always set to 0).	
3	Special cycles.	RW
2	Reserved (always set to 0).	
1	Memory space.	RW
0	I/O space.	RW

Bit **Description**

Bit 8 When set to 1, the system error driver for reporting address parity errors, data parity errors on the Special Cycle command, or any other catastrophic system errors, is enabled.

When set to 0, the system error driver is disabled.

Note that bit 6 must also be on to report address parity errors.

Bit 7 When set to 1, address/data stepping is enabled. This bit is hardwired to 1.

When set to 0, address/data stepping is disabled.

Bit 6 When set to 1, the device is set to respond to parity errors and takes prescribed action when parity errors are detected.

When set to 0, the device ignores parity errors and continues normal operation. This bit must reset to 0. Note that devices are still required to generate parity if parity checking is disabled.

Bit 5 When set to 1, special snooping behavior is enabled, meaning the device must not respond to DAC write operations.

When set to 0, palette accesses are treated like all other accesses.

Bit 3 When set to 1, the device is allowed to monitor Special Cycle operations.

When set to 0, the device ignores all Special Cycle operations.

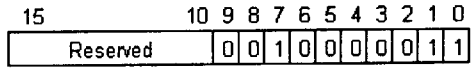
Bit 1 When set to 1, the device is allowed to respond to memory space accesses.
Except Revisions A & B: The W32p power-up condition has this bit set to 0.

When set to 0, the device response is disabled.

Bit 0 When set to 1, the device is allowed to respond to I/O space accesses.
Except Revisions A & B: The W32p power-up condition has this bit set to 0.

When set to 0, the device response is disabled.

Actual Command Register coding for the W32p is:



2.1.5 Device Status

2.1.5.1 Status Register

Configuration space address 06h

The status of PCI bus related events are recorded in the Status Register. Bits that do not pertain to the device's functionality may not be used. As an example, a device that acts as a target but will never signal target-abort would not implement bit 11. While reads to this register are handled normally, writes differ in that bits can only be reset, but not set. Bits are reset whenever the register is written with data (1) in the corresponding bit location (e.g., the value 0100 0000 0000 0000b clears bit 14 without affecting other bits). The register is then, in effect, a read/reset register.

Bit	Description	Access
15:11	Reserved (always set to 0).	
10:9	Device Select timing.	RO
8	Reserved. (always set to 0).	
7:0	Reserved.	

Bit	Description
Bit 11	This bit must be set by a target device whenever it terminates a transaction target-abort.

Bits 10:9 The timing of DEVSEL# is encoded by bits 10:9. Three available timings are allowed, as follows:

00 - fast
01 - medium
10 - slow

These read-only bits must indicate the slowest time that a device asserts DEVSEL# for any bus command except Configuration Read and Configuration Write. These bits are hardwired for the W32p as 0 0.

Bit 8 This bit is implemented by bus masters only. Set to 0 always.

2.1.5.2 Interrupt Line Register *Except Revisions A & B*

Configuration space address 3Ch

Bit	Description	Access
7:4	Reserved (always set to 0).	
3:0	System controller interrupt connect pin.	RW

Bit	Description
Bits 3:0	The value indicated by these bits indicate which input of the system controller(s) the W32p's interrupt pin is connected to.

2.1.5.2.1 Interrupt Line Register *Revision B*

Configuration space address 3Ch

Bit	Description	Access
7:4	Reserved (always set to 1).	
3:0	System controller interrupt connect pin.	RW

Bit	Description
Bits 3:0	The value indicated by these bits indicate which input of the system controller(s) the W32p's interrupt pin is connected to.

2.1.5.2.2 Interrupt Line Register *Revision A*

Configuration space address 3Ch

Bit	Description	Access
7:0	Reserved (always set to 1).	RO

2.1.6 Interrupt Pin Register

Configuration space address 3Ch

Bit	Description	Access
15:9	Reserved (always set to 0.)	RO
8	Permanently set to 1.	RO

Bit	Description
8	The interrupt pin register tells which interrupt pin a device uses. This register (bit <8>) is permanently set to Z value of 1 corresponding to INTA#.

2.1.7 Base Address Register

Configuration space address 10h

Bit	Description	Access
31:24	Upper byte of base register.	RW
23:0	Reserved (always set to 0).	

Bit Description

Bits 31:24 ***Except Revision B*** These bits are written to by the system BIOS to indicate where in the 32-bit address space to map a 16MB linear address space to be used when the W32p is put in system linear mode. Bits <31:30> will be compared with AD<31:30> in PCI mode, when CRTC Indexed Register 36 <4> is set to 1.

Bits <29:24> are not physically compared and should be copied into the Address Map Comparator bits <7:2> (CRTC Indexed Register 30) when using system linear addressing.

Revision B ONLY: Bits <31:30> are not physically compared with AD<31:30>.

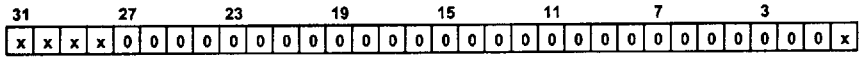
2.1.8 Expansion ROM Base Address Register

Revisions prior to C

The 4-byte register at offset 30h in configuration space is defined to handle the base address and size information for expansion ROM. Bits<31:1> are hard-wired to 000C0000. Bit 0 in the register is read/write and is used to control whether or not the device accepts accesses to its expansion ROM.

Revisions C and later

The 4-byte register at offset 30h in configuration space contains the following:



Bits <31:28> are read/writeable with a power-up condition of 0h. If the DISB PORI bit is set, then writes to these bits will have no effect and 0h will always be returned. Bits <31:28> represent the remapping location for the ROM. All zeroes in these registers locates the ROM into the standard VGA location of 0C000h. When any other value is written to these registers, the ROM is remapped to a 256MB space which is designated by these four bits. The ROM contents will appear as multiple images beginning on every 32KB boundary within this range.

Bits <27:1> are read-only and are hardwired to this value. Bit 0 in the register is read/write and is used to control whether or not the device accepts accesses to its expansion ROM. When this bit is reset, the device's Expansion ROM address space is disabled. When the bit is set, address decoding is enabled using the parameters in the other part of the base register. This allows the device to be used with or without an expansion ROM depending on system configuration. The power-up condition of bit 0 is 0. The Memory Space bit in the Command Register has precedence over the Expansion ROM enable bit. A device must respond to accesses to its expansion ROM only if the Memory Space bit and the Expansion ROM Base Address Enable bit are both set to 1.

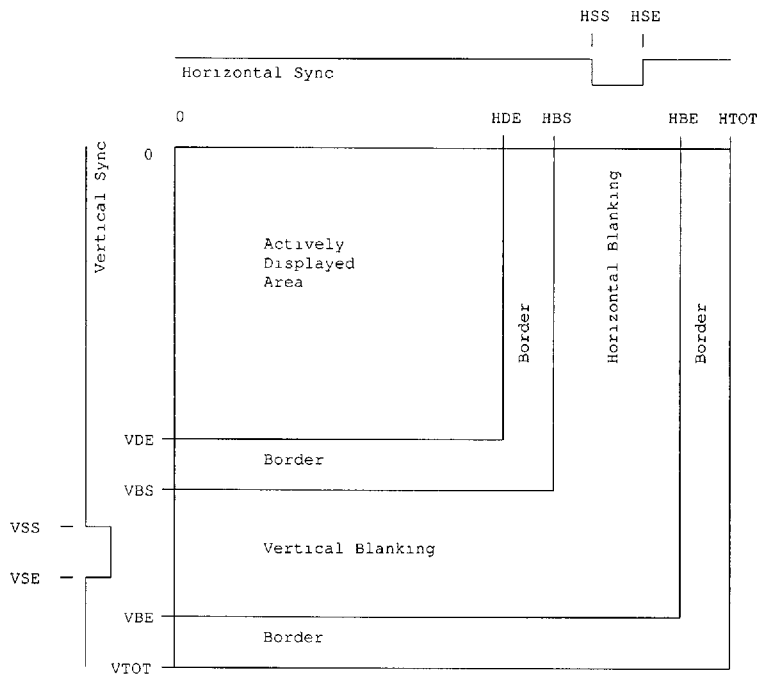
NOTE: Unlike local bus implementations, moving the memory address range from A0000-BFFFF to any other location will not automatically disable the ROM. Disabling the ROM should be handled through bit 0 of this register only.

2.2 VGA CRT Controller (CRTC)

The ET4000/W32p internal CRT Controller provides a 20-bit linear doubleword address, cursor control, and Vertical Sync and Horizontal Sync controls to external raster-scan CRT displays. Internally, the CRTC derives all reference timing in two dimensions: the horizontal display/blanking/sync and vertical display/blanking/sync. Each cycle in horizontal and vertical is evolved around the ET4000/W32p's CHARACTER and LINE reference logic. Each character is based on a multiple of 8 or 9 MCLK periods. Both CHARACTER and LINE reference logic can be asynchronously initialized via the SYNR input pin.

Figure 2.2-1 displays the role that the CRTC registers play to effect the horizontal and vertical timings of the CRT.

Figure 2.2-1: CRTC Video Timing Registers



HDE: Horizontal Display Enable
HBS: Horizontal Blank Start
HBE: Horizontal Blank End
HSS: Horizontal Sync Start
HSE: Horizontal Sync End
HTOT: Horizontal Total

VDE: Vertical Display Enable
VBS: Vertical Blank Start
VBE: Vertical Blank End
VSS: Vertical Sync Start
VSE: Vertical Sync End
VTOT: Vertical Total

2.3 Secondary CRT Controller (CRTCB)/Sprite

This module of the ET4000/W32p may be programmed as a hardware cursor (Sprite) or as a secondary display window (CRTCB). The two features cannot, however, be used at the same time. A control bit is provided in the CRTCB/Sprite Control Register (Index: EF) to select between the CRTCB and Sprite functions.

The ET4000/W32p can be programmed to inform the host processor when the last scan line of the CRTC, or CRTCB/Sprite has been displayed on each frame using a system interrupt. See Section 5.2.33, CRTC Index Register 35, bit 6.

2.3.1 CRTCB Overview

The CRTCB is a secondary CRTC display window. Its X/Y position, X/Y size, starting address, width, and color depth can be programmed via the CRTCB registers (see Section 5.6). The main differences between CRTC and CRTCB are:

1. CRTCB is programmed relative to the CRTC's X/Y display window in terms of X/Y position and width in pixel (X) and line (Y) resolution. The size of the CRTCB display can be programmed from 1 pixel x 1 line, to the entire CRTC display size.

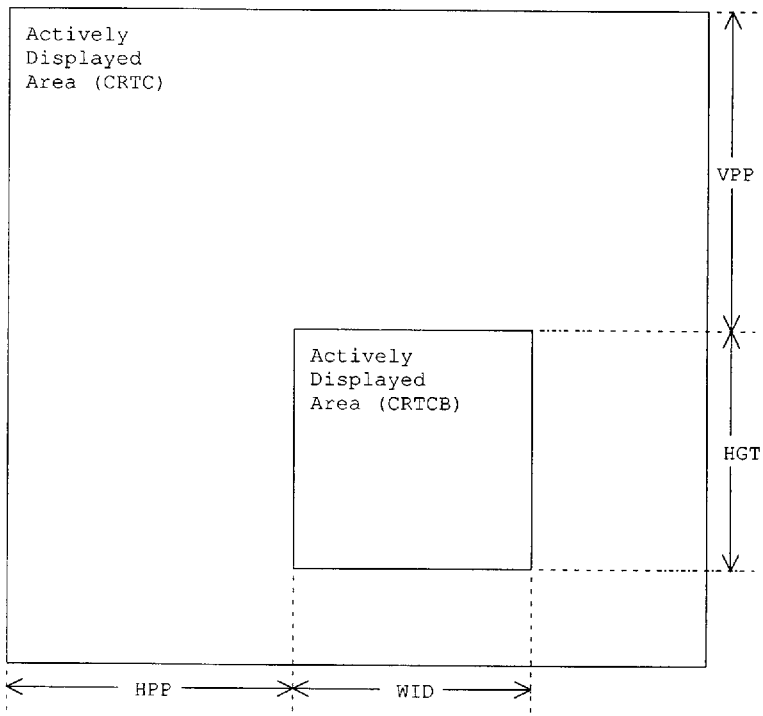
2. The color attributes displayed are not subject to internal "ATC" processing; i.e., the CRTCB display is packed pixel (linear graphics) format only.

3. The CRTCB display must be overlaid when the CRTC display is in timing state "1" (8 dots per character); i.e., CRTCB is always in 8 dots per character mode regardless of the CRTC's timing state.

2.3.2 Positioning the CRTCB Window

The CRTCB position on the screen is defined by the Pixel Position Registers. These registers indicate the point on the screen, relative to the actively displayed area of the CRTC, where the upper left-hand corner of the CRTCB window is displayed. The Width and Height Registers are used to control the pixel size of the CRTCB window. The following figure shows the use of these registers.

Figure 2.3.2-1: CRTCB Window Positioning



HPP: Horizontal Pixel Position
WID: Width
VPP: Vertical Pixel Position
HGT: Height

2.3.3 CRTCB Data Format

The data for the CRTCB window is stored in the display memory. The exact location of the beginning of the data in display memory is programmed into the Starting Address Registers, and the number of doublewords from one row of data to the next is programmed into the Row Offset Registers.

The Color Depth Register controls the formatting of the pixel data in memory.

	Byte 3								Byte 2								Byte 1								Byte 0							
bit w/in dword	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	7	6	5	4	3	2	1	0	7	6	5	4	3	2	1	0	7	6	5	4	3	2	1	0	7	6	5	4	3	2	1	0
1 bpp																																
pixel number	24	25	26	27	28	29	30	31	16	17	18	19	20	21	22	23	8	9	10	11	12	13	14	15	0	1	2	3	4	5	6	7
bit significance	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
2 bpp																																
pixel number	12	12	13	13	14	14	15	15	8	8	9	9	10	10	11	11	4	4	5	5	6	6	7	7	0	0	1	1	2	2	3	3
bit significance	1	0	1	0	1	0	1	0	1	0	1	0	1	0	1	0	1	0	1	0	1	0	1	0	1	0	1	0	1	0	1	0
4 bpp																																
pixel number	6	6	6	6	7	7	7	7	4	4	4	4	5	5	5	5	2	2	2	2	3	3	3	3	0	0	0	0	1	1	1	1
bit significance	3	2	1	0	3	2	1	0	3	2	1	0	3	2	1	0	3	2	1	0	3	2	1	0	3	2	1	0	3	2	1	0
8 bpp																																
pixel number	3	3	3	3	3	3	3	3	2	2	2	2	2	2	2	2	1	1	1	1	1	1	1	1	0	0	0	0	0	0	0	0
bit significance	7	6	5	4	3	2	1	0	7	6	5	4	3	2	1	0	7	6	5	4	3	2	1	0	7	6	5	4	3	2	1	0
16 bpp																																
pixel number	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
bit significance	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0

2.3.4 Sprite Overview

The Sprite is a 64x64-pixel image. When active, it overlays the picture that is being displayed in CRTC. Each Sprite pixel is 2 bits, encoded to have the following effect on the display:

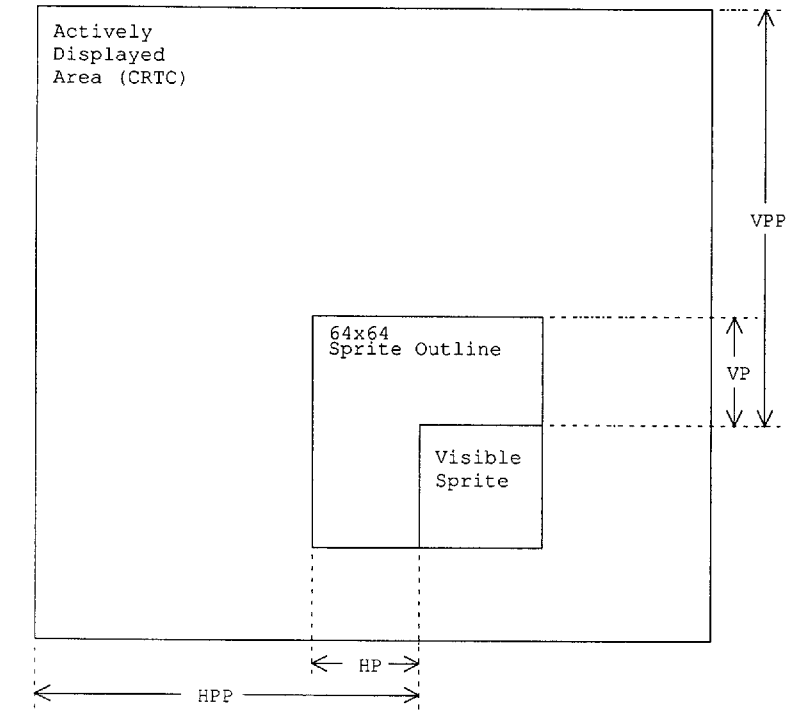
Bits<1:0>	Sprite Effect
0 0	Sprite Color 0 (defined as 00)
0 1	Sprite Color 1 (defined as FF)
1 0	Transparent (allow CRTC pixel pass through)
1 1	Invert (allow CRTC pixel invert through)

2.3.5 Positioning the Sprite

The Sprite position on the screen is defined by two types of registers: Pixel Position, and Preset. The Pixel Position registers specify where the first displayed Sprite pixel (upper left-hand corner) appears on the screen, and the Preset registers specify the offset into the 64x64 Sprite buffer as well as the X/Y size of the visible portion of the Sprite.

The main use of the Sprite Preset registers is to allow for displaying sprites which are less than 64x64 pixels in size.

Figure 2.3.5-1: Sprite Positioning



HPP: Horizontal Pixel Position
HP: Horizontal Preset
VPP: Vertical Pixel Position
VP: Vertical Preset

2.3.6 Sprite Data Format

The data for the Sprite is stored in the display memory. The exact location of the beginning of the data in display memory is programmed into the Starting Address Registers. The data is stored in a contiguous 1024 byte area, arranged in the following linear "packed" format:

	Byte 3								Byte 2								Byte 1								Byte 0							
bit w/in dword	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
bit w/in byte	7	6	5	4	3	2	1	0	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 number	15	15	14	14	13	13	12	12	11	11	10	10	9	9	8	8	7	7	6	6	5	5	4	4	3	3	2	2	1	1	0	0
bit significance	1	0	1	0	1	0	1	0	1	0	1	0	1	0	1	0	1	0	1	0	1	0	1	0	1	0	1	0	1	0	1	0

A single row of the Sprite thus occupies 4 contiguous doublewords. The second row occupies the next 4 contiguous doublewords after the first row, and so on in increasing fashion.

2.4 Memory Control Unit (MCU)

The Memory Control Unit provides programmable control of several aspects of the DRAM memory operation:

Memory control: RAS/CAS/MW timing/sequence control; the trp (RAS pre-charge), trcd (RAS to CAS delay), tcas (CAS pulse width), and tcp (CAS pre-charge), are programmable via CRTC Indexed Register 32.

Memory address: provides up to 4 megabyte addressing space via multiplexed AB<9:0> and AA<9:0> address interface.

Memory data: provides from 16-bit display memory data width to 32-bit data width via MD<31:0> data interface.

Memory refresh: programmable refresh frequency via CRTC Indexed Register 36.

2.4.1 Memory Interleave

Memory interleaving is done to increase DRAM bandwidth without doubling the DRAM data bus pin count. The ET4000/W32p provides the capability to interleave data from two banks of DRAM that share common RAS, address, write enable, and data signals, which are unique via the CAS signals. Through use of memory interleaving, The W32p delivers the performance of VRAM while using low-cost DRAM. Unlike CPU interleaving that may increase system performance overall by only twenty or thirty percent, the sequential nature of video accesses allows the W32p memory manager to leverage an additional seventy percent from its RAM. With VRAM, the resource allocated to the CPU and internal accelerator, or to the CRTC for screen refresh, is fixed at fifty percent for each. Demand on the memory may be higher than fifty percent on one side or the other, creating inefficiency. The W32p, however, can dynamically allocate its resource to the CPU, the graphics accelerator, the CRTC, or data being input through the IMA port.

2.5 System Priority Controller (SPC)

The SPC's main task is to maximize performance by orchestrating the ET4000/W32p's internal resource requests including: the Display FIFOs, Graphics Data Controller, Multiport Cache Controller, Accelerator, Image Port, and RAM refresh. The available memory bandwidth for system performance is based on two major factors: the CRTC's demand (i.e., the display resolution and color), and the memory bandwidth (i.e., the memory bus width and access time). Use of the Graphics Accelerator and/or the Image Port can substantially increase performance by reducing the number of host accesses.

Other factors also can contribute to the overall performance. For example, the cache controller provides optimum performance for sequential access rather than random access, and host write operations are generally faster than host read operations. The 32-bit Local Bus interface also results in faster data transfer, particularly in the plane graphics mode (a 32-bit CPU write equals up to a 128-bit data transfer). For further discussion of performance aspects, refer to Display Memory Design Considerations, Section 6.4.

2.6 Multiport Cache

The internal Multiport Cache™, an exclusive feature of the ET4000/W32p, allows access of multiple masters to the display memory cache simultaneously. Multiport Cache provides the ET4000/W32p with the ability to parallel-process tasks. Even if the IMA port is updating an active second display window while the CPU and Accelerator processor are updating the primary active display, the ET4000/W32p will allow all three masters to read/write into the display memory, while operating concurrently.

2.7 Timing Sequencer (TS)

The Timing Sequencer module is responsible for providing basic timing control for both the CRTCs and ATC. Timings controlled by the TS registers include:

-Horizontal count resolution: 8 or 9 dots/character
-MCLK/2, MCLK/4, and DCLK/2 (dotclock)

2.8 Graphics Display Controller (GDC)

The GDC assists the CPU in manipulating pixel data that is in planar format in display memory. This includes rotate/mask/z-plane, with any of four boolean functions—in response to a single CPU write. By putting basic bit map operations in high-speed hardware, the ET4000/W32p dramatically increases graphics processing throughput over software-driven solutions. The data manipulation capability implemented in the GDC is, however, applicable only for Plane systems and not for Linear Byte systems. This is because all the processing functions are designed to manipulate pixel data with one bit sourced from each plane. For example, the color compare function allows four bits across four planes (one pixel) to be compared to a pre-defined color, thereby allowing eight pixels to be color-compared simultaneously by processing 32 bits of video data (one byte from each plane).

2.9 Attribute Controller (ATC)

The internal Attribute Controller (ATC) provides flexible high-speed video shifting and attribute processing, and video load control every 8, 16, or 32 dot clocks, designed for both text and graphics video display applications. The ATC can process up to 16 bits of display data at the rate of 50MHz or 8-bit pixel data at a rate of 86MHz. In graphics modes, memory bits are reformatted into pixel color data in groups of 16, 8, 2, or 1 adjacent bits, translated through an internal 16-element color look-up table, and sent out serially to the video display. Through this pixel mapper, the ATC supports "PLANE" (for 16 colors), "BYTE" (256 colors) and "WORD" (65,536 colors) oriented pixel structures.

In text mode, eight bits of character code data and eight bits of attribute data are loaded; the character code is used as a lookup into a font table that is then loaded as the 16 bits of font data. The attribute is then applied to the font/cursor data, translated through the color lookup table, and sent out serially to produce 16 colors of text pixel data at speeds of up to 56MHz.

2.10 Image Port (IMA)

The Image Port is an 8-bit asynchronous input port capable of accepting CPU or image data directly into the display memory, and the ET4000/W32p will keep track of linear addresses being transferred as well as data transfer counts per line. The input data scan sequence can be interlaced, or non-interlaced, and may be sent along with a bit mask so that only the differential motion data is transferred. Once in display memory, the IMA data may be displayed through either the primary display (CRTC), or the secondary display (CRTCB). The combination of the internal bandwidth of the W32p chip, along with the IMA port, make possible 640x480, 16.7 million-color, full-motion (30 frame/second) digital video on desktop computer systems. The ET4000/W32p integrates Multimedia and Imaging capabilities beyond any SuperVGA class graphics controller.

The IMA port is a physical interface between an asynchronous processor such as an image processor for motion video, or simply a dedicated microprocessor high-speed direct connection, and the ET4000/W32p controller. The main mechanisms of this high-speed direct connection are:

1. Sustained asynchronous throughput rate up to 40 MB/sec.
2. The address generation is two-dimensional and sequential and is fully specified via IMA Registers E0-E6 prior to the data transfer.
3. The synchronization of address generation is by way of Frame/Line and odd/even interface signals.
4. The range of data transfer per line is specified by programming the image transfer length registers (IMA F3-F4).
5. A byte mask input can be used to specify only the changing motion video data to be transferred to the ET4000/W32p's frame buffer. The masked data transferred can be used to reduce the bandwidth requirement.

2.10.1 Image Port Interface Protocol

1. External Image Processor produces IXFS and IXLS pulses signaling initialization of linear address.
2. ET4000/W32p loads the image start address to the linear address generator.

If the interlace bit (IMA Indexed Register F7, bit 1) is set to:

0, then loads the image start address to the linear address generator.

1, then (If IXOF = 1) loads image start address and image row offset to the linear address generator

(If IXOF = 0) loads image start address to the linear address generator

3. After the trailing edge of IXLS and sensing IXRD ready acknowledgment, the image processor can begin to toggle the IXWQ* write request and place the 8-bit IM<7:0> and IDMK byte mask at each transfer. NOTE: IDMK, when equal to 0, can be used to "walk" the address generator's pointer without data being transferred.
4. The image processor continues to sample the IXRD ready (by clocking IXRD with the image processor's internal clock). If IXRD is asserted, the IXWQ* can be toggled, else IXWQ* is held at the high state.
5. If the number of doubleword count transfers has occurred, then IXRD will become inactive and wait for the IXLS input.
6. The image processor sends an IXLS line synchronization. If the IXLS input occurs before the doubleword count transfer is complete, the transfer counter is re-initialized to zero, and the remaining data will not be transferred.
7. Upon the leading edge of IXLS, the ET4000/W32p will advance the linear address pointer to the beginning edge of the next line by adding the image row offset, and return the IXRD, indicating that the IMA Port is ready for data transfers.

If the interlace bit (IMA Indexed Register F7, bit 1) is set to:

0, then image row offset value is added.
1, then twice image row offset value is added.

This process is repeated.

2.11 Memory Management Unit (MMU)

The ET4000/W32p Memory Management Unit (MMU) provides a mechanism to access the full 4MB range of display memory even though the display memory may occupy a much smaller region in the system's memory space. This is accomplished by providing a fixed-size "aperture" through which the display memory may be accessed. The aperture varies in size depending upon the system configuration; for typical VGA-compatible systems the aperture size is 8KB. See the Video Memory Map table in Section 7.3 to see how aperture size is related to system configuration. The aperture may be relocated to begin on any doubleword boundary in the 4MB display memory space.

2.11.1 Software Considerations

Each aperture has a Memory Base Pointer Register (MBP) which is 22 bits wide. The MBP Register specifies the starting address of the aperture in the linear display memory. The bottom two bits of the MBP Register must be set to "00", thus making the MBP aligned to the doubleword.

IMPORTANT NOTE FOR ***Revision A:*** If the accelerator is being used, it is recommended that aperture number 2 be used when data must be passed to/from the accelerator by the host. This will ensure compatibility with future revisions of Tseng Labs' accelerated chips.

IMPORTANT NOTE ***Except Revision A:*** The MBP for aperture number 2 is handled differently than the other two apertures. Namely, MBP2 is not directly programmable; it cannot be read or written by the host. Its value is determined by the programmed value of bit 4 of the CRTC Indexed Register 36 (AMC<0>). If AMC<0> is set to 0, MBP2 is internally set to zero; if AMC<0> is set to 1, MBP2 is internally set to 200000(hex). If the accelerator is being used, it is recommended that aperture number 2 be used when data must be passed to/from the accelerator by the host.

The MMU provides three independent apertures. The host implicitly selects which aperture to use by virtue of the address the host is accessing. As an example, consider the case of the MMU buffer space occupying the address region from B8000 through BDFFF (line six of the Video Memory Map table in section 7.3). All accesses in the range of B8000 through B9FFF will be directed through aperture number 0, BA000 through BBFFF use aperture number 1, and BC000 through BDFFF use aperture number 2. Figure 2.11.1-1 illustrates this address translation:

Associated with each MMU aperture are two control bits:

Linear Address Control (LAC), and
Aperture Type (APT)

The LAC bit controls the organization of data in display memory for the given aperture. (See section 5.8 MMU Register Descriptions, MMU Control Register for a description of the LAC bits). The LAC bit allows the programmer to access memory in a linear fashion, independent of the current display mode.

The APT bit indicates that accesses through this aperture should be directed to the accelerator. Namely, if the APT bit is a "1," an access through this aperture will be passed to the accelerator; otherwise the access will go through the GDC to the display memory.

The actual address translation that the MMU performs is quite simple. It adds bits <12:0> of the host address to the Memory Base Pointer for the selected aperture. Figure 2.11.1-2 depicts the address translation process.

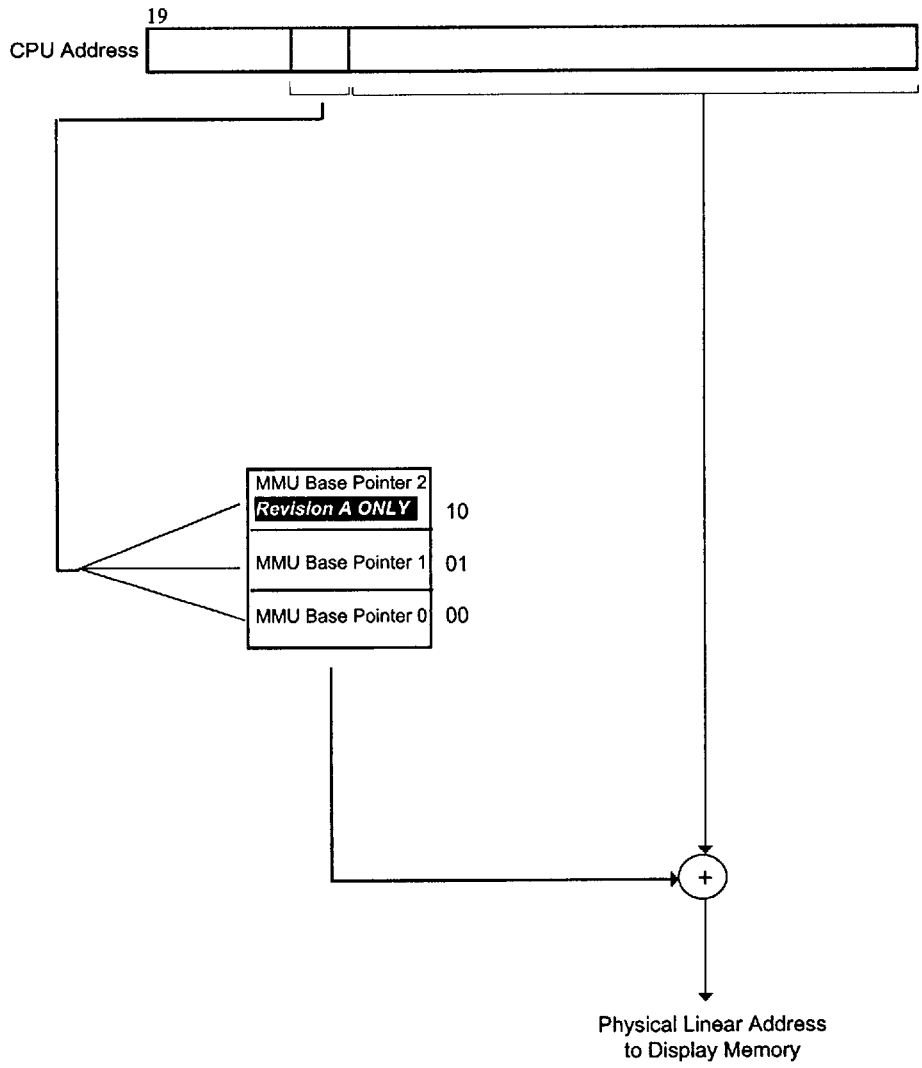


Figure 2.11.1-1 Three MMU Aperture Mapping

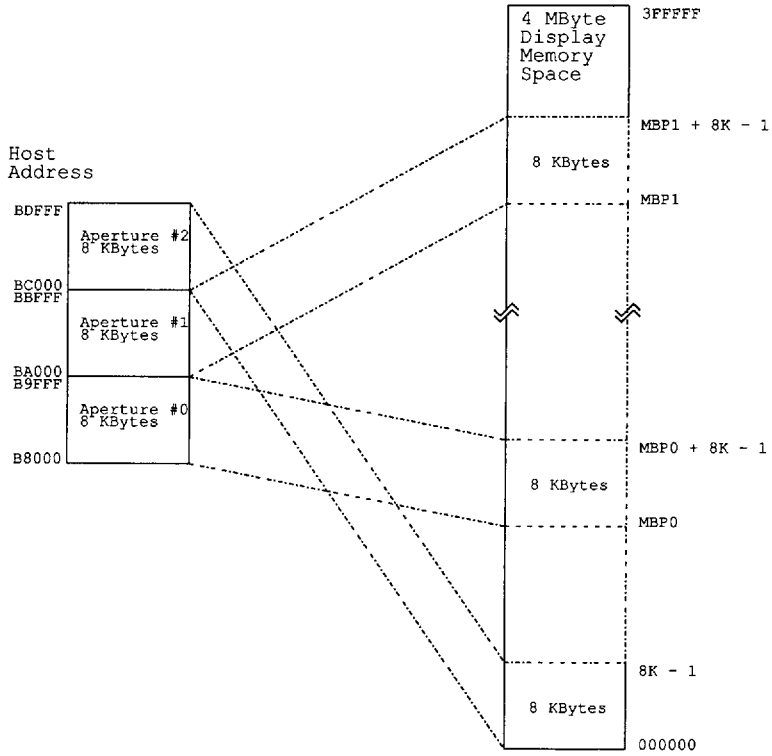


Figure 2.11.1-2: MMU Address Translation

2.12 Graphics Accelerator (ACL)

The ET4000/W32p Graphics Accelerator is the most cost-efficient method to expedite functions used in common applications such as Microsoft Windows and other graphical user interface (GUI) software. Typically, personal computer architecture and processor performance limit the performance of operations such as BitBLT, LineDraw, Font Painting, and Raster Operations. The ET4000/W32p allows the CPU to distribute these tasks to its Graphics Accelerator. The Accelerator registers are mapped to unused areas of the display memory address space, and reconfigured automatically when multiple adapters are present, or if the VGA switches from graphics into text mode.

The ET4000/W32p Graphics Accelerator provides a simple, yet powerful mechanism to accelerate the movement and processing of graphics data. The accelerator architecture adheres to the RISC philosophy of providing the basic building block for manipulation of graphics data at a high rate of performance, while allowing the software to manage the complexity of higher-level drawing algorithms. The accelerator has the capability to operate without CPU intervention on graphics data in the display memory, or it may take data from the CPU and mix it with data from the display memory.

Inside the accelerator are two primary functional blocks:

- An address sequencer, and
- A graphics data processor.

By using these two functions, much of the CPU-processing required to perform a Bit Block Transfer ("BitBLT") or to draw a line can be off-loaded from the host CPU to the Graphics Accelerator. The Address Sequencer maintains address pointers to locate data in display memory, and the Graphics Data Processor combines data from a Mix Map, Source Map, Pattern Map, and Destination Map, and writes the result back to the Destination Map under the control of parameters programmed into the chip.

Maximum performance is achieved by:

- Minimizing the amount of information that must be passed across the host bus between the host processor and the ET4000/W32p.
- The CPU need not perform any combinatorial functions on the graphics data; all Raster Operations are performed by the Graphics Accelerator.
- For most rectangular BitBLTs, the CPU need not maintain address pointers.
- The inner loop of a LineDraw operation can be executed by the Accelerator, freeing the CPU for other tasks.
- The accelerator supports a 1-bit-per-pixel "Mix Map", which speeds-up font-painting (via Color Expansion) and filling of irregular-shaped objects.
- Accesses to the display memory are localized within the ET4000/W32p, allowing the best possible utilization of display memory bandwidth.

2.12.1 Overview

The accelerator supports the notion of a "Pixel Map," which is defined by two things:

- A starting address in display memory, and
- A byte-offset from one scan line of the map to the next scan line. (In the case of the Mix Map, the offset is in bits.)

The accelerator operates on four "Pixel Maps":
1. Source Map
2. Pattern Map
3. Destination Map
4. Mix Map

Data from these maps are combined according to the following rule:

D = FgdRop(S,P,D) if MixMap bit is 1
D = BgdRop(S,P,D) if MixMap bit is 0

The Foreground Raster Operation (FgdRop) and Background Raster Operation (BgdRop) are 8-bit values which cover any possible combinational mix of the three maps (Pattern, Source, and Destination). The encoding of these 256 ROPs is 100% compatible with the Microsoft Windows specification. See Appendix A for a list of Raster Operations.

The Destination and Pattern maps must reside in the display memory. The Source and Mix map data may reside in display memory or be supplied by the CPU during an accelerated graphics operation. The Mix Map data may be fixed to "1" which causes the accelerator to always apply the Foreground ROP. For a given graphics operation, the CPU can provide either Source data or Mix data (or neither), but not both. The Mix Map differs from the other three maps in that it is a "monochrome" map; that is, for each bit processed in the Mix Map, a byte is processed from the other three maps. The resultant Destination data that the accelerator produces may be read-back by the CPU instead of being written back to the display memory.

The figure above shows the path of one byte through the accelerator's Graphics Data Processor. In reality, more than one byte is processed at a time.

2.12.2 Starting an Accelerator Operation

After loading all the necessary accelerator control registers (e.g., X/Y Count Registers, Map Starting Addresses, Y Offsets, X/Y Wrap values, Raster Operations, etc...), a graphics operation is initiated by simply writing the starting address of the Destination Map into the ACL Destination Address Register. Strictly speaking, a write to byte 3 of the ACL Destination Address Register initiates the operation; so the same effect is achieved if the programmer performs a doubleword-write to byte 0, a word-write to byte 2, or a byte-write to byte 3. Note that the ASEN bit (bit 4 of the ACL Operation State Register) must be set to 1 in order to enable the accelerator to run.

An accelerated graphics operation is defined as a two-dimensional "walk" through display memory. The X Count Register and Y Count Register specify the limits of each dimension. The accelerator offers two different drawing functions: BitBlt, and LineDraw.

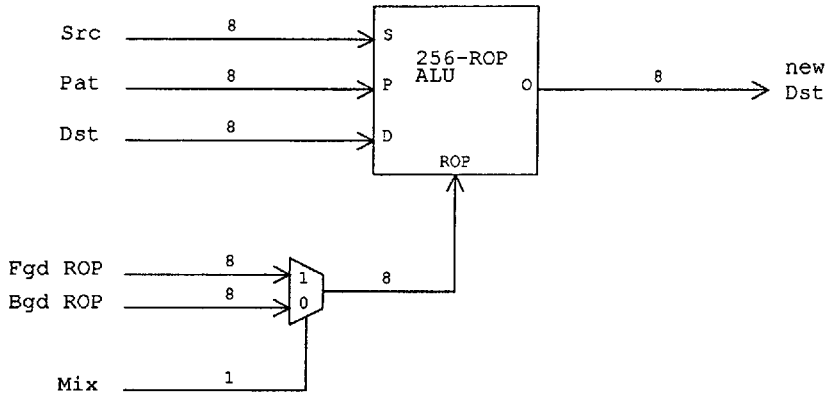


Figure 2.12-1 ET4000/W32p Graphics Accelerator Data Path

In the case of BitBlt, sequential bytes are processed until the programmed limit of the X Count Register has been reached. At that time, each map's Y Offset is added to that map's starting address, and the X-walk is repeated. This sequence repeats until the programmed limit of the Y Count Register has been reached. In the case of LineDraw, the accelerator uses the Bresenham algorithm to compute the address of each pixel on the line, stepping in the X and/or Y dimension until the programmed limit of either the X Count Register or the Y Count Register has been reached.

The accelerator always operates on pixel maps as if they are in linear format in display memory.

2.12.3 The Accelerator's Queue

The ET4000/W32p has a one-level queue of registers for the Graphics Accelerator. In addition to the queue, there are registers internal to the accelerator that it uses as a "working set" while it is performing a graphics operation. This configuration allows the host to be modifying the registers in the queue while the accelerator is running. The figure below shows the data path between the host, queued registers, and accelerator registers. For most graphics operations, the queue starts out as being "empty" and the host will load the registers in the queue to set-up the operation. Then the host will initiate the operation by the method outlined previously. At the time when the operation is initiated, the queue becomes "full" and all of the values stored in the registers in the queue are transferred into the accelerator's internal state. After the transfer, the queue becomes "empty" again, waiting for the host to set-up the next operation.

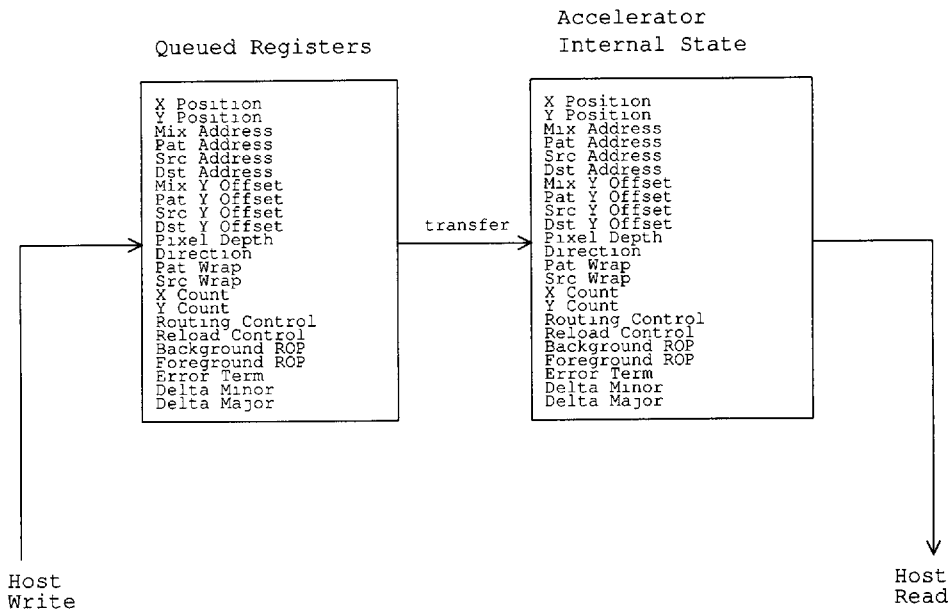


Figure 2.12-2 ET4000/W32p Queued Registers and ACL Internal State

The queued registers retain their values after the transfer takes place, so the programmer need not load the entire queue for each graphics operation.

Since only the internal state of the accelerator is readable by the host, the host cannot simply read back a value that it has just written to a register in the queue. If the host wishes to read the contents of the queue, it must first cause a "transfer" to take place to move the data from the queue into the accelerator's internal state. This transfer is achieved by writing to the ACL Operation State Register with bit 0 (the "Restore" bit) set to "1." Naturally, the host must ensure that the accelerator is idle when the Restore is performed.

2.12.4 Accelerator Operation

The accelerator maintains internal copies of the Address Registers for each map (Mix, Pattern, Source, and Destination) which it increments as an accelerator operation progresses. After an accelerator operation has completed, the internal Address Registers will point to the first byte of the next scan line to be processed. For example, if a BitBLT is initiated from (0,0) with a width of 4 (XCNT=3), a height of 2 (YCNT=1), and a Direction of X/Y Increasing (DIR=00), then the final Source Address will be the linear address that corresponds to the byte at an X/Y-position of (0,2).

The accelerator also maintains an internal copy of the X Position and Y Position as an accelerator operation progresses. These position registers serve as a "reference" pointer into each of the maps to indicate how far the operation has progressed. The X Position and Y Position Registers are automatically initialized to zero by the accelerator when a graphics operation is initiated.

The internal address register for a map is only updated if that map is needed to perform the programmed Raster Operation.

If the host is supplying Source data, the bottom 2 bits of the Source Address Register are used to indicate the byte alignment within a doubleword. This allows the programmer to have control over the internal byte rotator of the accelerator. As an example of utilizing the internal rotator, consider the case of moving data from system memory at address 203 (Hex) to address 400 (Hex) in the display memory. The Source Address Register is programmed such that the bottom two bits are "3", and the Destination Address Register contains 400. The host then writes one byte, and then as many doublewords as required to transfer the rest of the data. This mechanism provides the best possible performance since there are no unaligned reads from system memory—all of the data alignment is done by the accelerator. There is no penalty for unaligned Destination writes inside the accelerator.

Similarly, if the host is supplying Mix data, the bottom 5-bits of the Mix Address Register are used to indicate the bit alignment within a doubleword. Once again, this frees the CPU from the task of doing bitwise-rotation when passing the 1-bit-per-pixel Mix Map to the accelerator; the accelerator has the bitwise-rotator built into its data path. Of course, if the programmer wishes to have the CPU do the rotation, that option is always available.

Basically, by using the Accelerator Data Routing (ACRO), Data Route (DARO), and MixMap Enable (MXEN) control fields in the ACL Routing Control Register (see Section 5.9.20), the programmer can perform the following types of operations (also, see Figure 2.12-3, Accelerator Data Path Routing):

ACRO	MXEN	DARO	Mix Data	Src Data	Dst Write Data
0	x	00	display memory	display memory	display memory
0	0	01	fixed to "1"	from host	display memory
0	1	01	display memory	from host	display memory
0	x	10	from host	display memory	display memory
1	x	00	display memory	display memory	goes to host
1	0	01	fixed to "1"	from host	goes to host
1	1	01	display memory	from host	goes to host
1	x	10	from host	display memory	goes to host

NOTE: All functions feature 3-way ROP between Src, Pat, Dst, with ROP selected by the MixMap. PAttern is always assumed to be in the display memory. The MixMap data has three possible options: fixed to "1," read from display memory, or provided by the host.

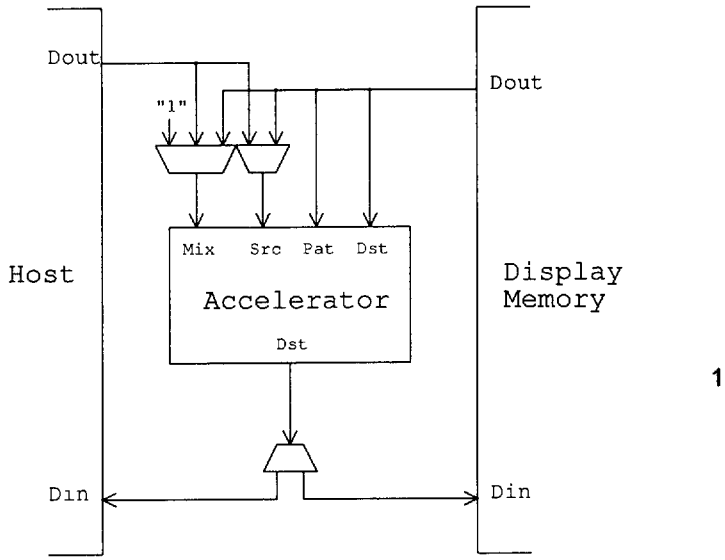


Figure 2.12-3 ET4000/W32p Accelerator Data Path Routing

2.12.5 Passing Map Data to the Accelerator

2.12.5.1 Data Alignment and Rotation

Map data is passed to the accelerator by writing to an accelerated MMU aperture while the accelerator is operating. The ACL Routing Control Register should have the DARO and MXEN fields programmed appropriately to inform the accelerator that data for the operation will be coming from the host. The host may supply either Source Map data or Mix Map data, but not both. The only requirement is that the host write in such a way that the data is in alignment with the programmed map Address Register. For example, if the host is passing Source Map data to the accelerator and the Source Address Register has its bottom two bits set to "11," the host must start writing data to the MMU aperture starting at byte 3. Note that only the bottom two bits of the host address are significant; the rest are ignored. Internally, the accelerator looks at the relationship between the Source Address and the Destination Address and computes what rotation is required to align the maps to each other.

Similar rules apply when passing Mix Map data; the bottom 5 bits of the Mix Address indicate the bit-alignment of the Mix Map.

In order to eliminate double-cycle accesses to the system memory, it is recommended that the transfer become doubleword-aligned as quickly as possible. Namely, if the starting address is not doubleword aligned, the first cycle(s) may have to be byte- or word-sized until the address becomes aligned. With this in mind, it is clear that the transfer of one scan line can be broken into three sections:

1. Alignment (to get doubleword aligned).
2. Middle (to move doublewords).
3. Remainder (to process remaining bytes in last doubleword).

Any given transfer may only require steps 1 and 2, 2 and 3, or just 2 alone; it all depends on the initial starting address alignment and the number of bytes per scan line of the BitBlt. As an example, consider a BitBlt of 405 bytes per scan line with an initial Source Address of 102(Hex). The transfer is broken into three sections as shown:

1. Perform a word-write or two byte-writes in order to get aligned.
2. Perform 100 doubleword-writes (which are aligned).

3. With three bytes remaining, can perform a word-write followed by a byte-write; a byte-write followed by a word-write; or three byte-writes.

Note that the accelerator has no strict requirement that the transfer be performed in these three sections; this is simply an example of how to avoid costly double-cycle unaligned accesses to system memory when performing Memory-to-Screen accelerator operations.

Due to the way that Intel's 80x86 processors perform unaligned doubleword accesses out-of-order, it is possible to get the host/accelerator interface into a deadlock situation. For this reason, unaligned doubleword writes should be strictly avoided. The accelerator has the ability to detect the deadlock situation and ignore that write cycle, thus allowing the operation to continue. A Write Fault Interrupt can be generated to indicate that the deadlock has occurred.

It is imperative that the host send the exact number of bytes that the accelerator requires to complete the operation, as specified by the X Count and Y Count Registers.

When the host is supplying the Mix Map for a graphics operation, the processing order of bits in the Mix Map varies depending on the programmed X Direction:

• If the X Direction is increasing ("0"), the least-significant bit of the Mix data is processed first. In other words the least-significant bit of the Mix Map is anchored to the left-most edge of the Destination Map.

• If the X Direction is decreasing ("1"), the most-significant bit of the Mix data is processed first. In other words the most-significant bit of the Mix Map is anchored to the right-most edge of the Destination Map.

2.12.5.2 Synchronization (for writing)

Another important issue when passing data from the host to the accelerator is synchronization. Essentially, there are two processors (the host and the accelerator) working in concert to perform a BitBLT. The transfer of data must be throttled so that the two processors remain in sync. This throttling can be done at one of two levels, the software level or the hardware level.

At the software level, the host can poll the Write-Status bit before each write to determine if there is room in the queue for another data write. Of course, this can add overhead to the inner loop for writing data to the accelerator, and is not generally recommended.

At the hardware level, the host bus WAIT (aka Ready) line can be used for synchronization. Programming the Sync Enable bit to "1" forces the ET4000/W32p to insert wait-states into a host data transfer when the queue is full.

The primary concern when operating in this fashion is to limit the amount of time that the host bus is held waiting, as inordinately long wait times can crash some PC systems as the main memory cannot be refreshed. For this reason it is important to try to match the speed of the accelerator to that of the host processor. A general rule of thumb is to move a doubleword or word per write when passing Source Map data; and move one byte per write when passing Mix Map data.

2.12.6 Reading Map Data from the Accelerator

2.12.6.1 Data Alignment and Rotation

Map data is read from the accelerator by reading from an accelerated MMU aperture while the accelerator is operating. The ACL Routing Control Register should have the ACRO programmed appropriately to inform the accelerator that the resultant Destination Map data from the operation will be going to the host.

The host MUST perform 32-bit doubleword-aligned accesses when reading data from the accelerator. Note that only the bottom two bits of the host address are significant; the rest are ignored.

As an example, consider the case of copying a pixel map from the display memory at address 0x1003 to the host's main memory. The Source Address Register would be programmed to 0x1003, and the Destination Address Register would be programmed to any value that is doubleword aligned (for example, 0). Once the accelerator is started, the host would begin reading data from the accelerated MMU aperture (from a doubleword-aligned address). The accelerator takes care of rotating the Source Map data from byte 3 to byte 0, and presenting it as doublewords for the host to read.

2.12.6.2 Synchronization (for reading)

The synchronization issues for reading are similar to those discussed in Section 2.11.5.2 for writing. The recommended mode of operation is to program the Sync Enable bit to "1" and use the WAIT (aka Ready) line of the system bus to throttle the transfer of data from the accelerator to the host. If the Zero Wait-State Enable bit of the Sync Enable Register is set to 1, the ET4000/W32p will not add any wait states to the reads from the accelerator. Generally, the accelerator has ample bandwidth to support steady zero-wait-state reads from the host.

2.12.7 Accelerator as Remote Processor

It is possible to write Source or Mix Map data to the accelerator AND read Destination Map data from the accelerator during the same graphics operation. The only rule that must be followed is that the writing operation must precede the read operation by at least one "step", where a step is the unit of data that is passing through the accelerator at one time. In most cases, the step is 4 bytes.

2.12.8 Support for Common Graphics Operations

This section offers programming suggestions to perform some common graphics operations. This is by no means an exhaustive discussion; the programmer is encouraged to gain an understanding of the core functions that the Graphics Accelerator provides and decide how to best put them to use for specific operations.

2.12.8.1 Line Drawing

The ET4000/W32p Graphics Accelerator has a built-in Bresenham LineDraw Engine. The LineDraw Engine operates in units of "pixels" rather than "bytes," and the Pixel Depth Register is used to tell the accelerator how many bytes to process per pixel.

To draw a line from pixel coordinate (x1,y1) to (x2,y2), the following steps should be executed:

1. Compute DeltaX:

```
dx = x2-x1;
```

2. Compute X Direction, and take absolute value of DeltaX:

```
if (dx < 0) {  xdir=1,  dx=-dx;  }
else        {  xdir=0;           }
```

3. Compute DeltaY:

```
dy = y2-y1;
```

4. Compute Y Direction, and take absolute value of DeltaY:

```
if (dy < 0) {  ydir=1,  dy=-dy;  }
else        {  ydir=0;           }
```

5. Compute Axial Direction and Load Registers:

```
if (dx >= dy) {       /* X is Major Axis */
        adir = 1;
        ACL X Count Reg          <== (dx-1) * "Bytes-per-pixel";
ACL Y Count Reg          <== 0xFFF;
        ACL Delta Minor Reg      <== dy;
        ACL Delta Major Reg      <== dx;
}
else {                /* Y is Major Axis */
        adir = 0;
        ACL X Count Reg          <== 0xFFF;
        ACL Y Count Reg          <== dy-1;
        ACL Delta Minor Reg      <== dx;
        ACL Delta Major Reg      <== dy;
}
```

6. Select LineDraw Algorithm and Load the Direction Reg:

```
if (Microsoft VGA Line Algorithm) {
        algrthm = ydir;
        ACL Direction Reg <== 0x80 | (algrthm<<4) | (adir<<2) | (ydir<<1) | (xdir);          }

else {                /* XGA LineDraw Algorithm */
        algrthm = (!ydir) & 1;
        ACL Direction Reg <== 0x80 | (algrthm<<4) | (adir<<2) | (ydir<<1) | (xdir);          }
```

7. Load the Destination Address Register with the address of the pixel at coordinate (x1,y1) to start the accelerator. If a pixel is more than one byte, the starting address depends on the X Direction in the following manner:

- If the X Direction is 1, the address must point to the highest addressed byte in the pixel.
- If the X Direction is 0, the address must point to the lowest addressed byte in the pixel.

Note that the Count Register along the Minor Axis is loaded with the maximum possible value (0xFFF). This ensures that the length of the line is limited by the value programmed into the Count Register for the major axis. The accelerator stops drawing the line when it reaches the limit of the smaller of the two Count values. This feature helps support clipping of lines as discussed in the following section.

2.12.8.2 Clipping of Lines

The accelerator offers two unique features that help support clipping when drawing lines:

1. Loadable Initial Error Term.
2. Length of line can be limited by either X or Y Count.

Consider the case of a line who's starting point (x1, y1) is outside of the clip region, and the line passes through an edge of the clip region. The point at which the line crosses the edge is called the intersection point (let's call it (xi,yi)). By loading the Error Term with the value that it would have at that point in the line, and also loading the Destination Address with the address of the point of intersection, the programmer can draw the remainder of the line. Normally, the accelerator computes the initial Bresenham Error Term based on the values of dx and dy (namely, initial ET = DeltaMajor). But with the LETQ bit of the ACL Direction Register set to "1," the programmed value of the Error Term is used as the initial Error Term for the LineDraw Engine.

The equation to compute the Error Term at the point (xi,yi) is:

X Major: ETi = ET1 + (|yi-y1| * (2*dx)) - (|xi-x1| * (2*dy))
Y Major: ETi = ET1 + (|xi-x1| * (2*dy)) - (|yi-y1| * (2*dx))

Note that a software algorithm must be employed to find the coordinates of the intersection point (xi,yi).
As another example, consider the case of a line who's starting point is inside the clip region. In this case, there is no need for the CPU to compute if the line passes through the edge of the clip region. The programmer can simply load the X Count Register with the minimum value of DeltaX and the horizontal distance from the starting point to the left or right edge, depending on the X direction. The Y Count is loaded in a similar fashion. The table below summarizes how to program the X and Y Count Registers, based on LineDraw direction.

Direction Y	Direction X	X Count	Y Count
0	0	MIN(DeltaX,xRight-x1)	MIN(DeltaY,yBottom-y1)
0	1	MIN(DeltaX,x1-xLeft)	MIN(DeltaY,yBottom-y1)
0	0	MIN(DeltaX,xRight-x1)	MIN(DeltaY,y1-yTop)
0	1	MIN(DeltaX,x1-xLeft)	MIN(DeltaY,y1-yTop)

2.12.8.3 Tiled and Fixed-Color Fills

Both the Source and Pattern Maps can be programmed to "wrap" or "tile" as the graphics operation progresses. There is one requirement on the location of the data for the tiled map. Namely, the data for the tiled map must be stored at a base address that is on a modulo XWrap times YWrap boundary. For example, a 4x1 map must be on a 4-byte boundary, a 4x4 map must be on a 16-byte boundary, an 8x8 map must be on a 64-byte boundary, and so forth. With this approach, the address programmed into the Source (Pattern) Address Register points to the byte that will be anchored to the STARTING CORNER

of the BitBlt. A side effect of this approach is that the wrap combinations of None-by-2, None-by-4, and None-by-8 will not work as expected, and should not be used.

The Source and Pattern Maps can be configured as fixed-color maps by programming the X/Y Wrap values to 4-by-1. If the Destination is 8 bit-per-pixel, the 8-bit fixed color must be written into all four bytes of the Source (Pattern) maps. This can be thought of as an X Wrap value of one byte. Similarly, an X Wrap of two bytes can be achieved by duplicating the two bytes to fill the four bytes of the map.

For ease of use when the Pixel Depth Register is programmed to 3 bytes-per-pixel, a wrap of 4-by-1 will correctly repeat the 3 bytes as the fill color. In other words, it really acts as a "3-by-1" wrap.

2.12.8.4 Color Expansion and Font Painting

The accelerator is capable of expanding a 1 bit-per-pixel (monochrome) pixel map into an 8 bit-per-pixel map. This is a very common operation when painting fonts. Opaque text (that is, text of a solid foreground color painted with a solid background color) is drawn by setting the Foreground ROP to "Src," and the Background ROP to "Pat," and supplying the 1 bit-per-pixel map as the Mix Map. A "0" in the Mix Map will result in the fixed color in the Pattern map to be written to the Destination, and a "1" in the MixMap will draw the Source color into the Destination. Of course, there is no restriction that the Pattern and Source Maps be solid color; they could just as easily be 2-dimensional pixel maps. The important point to understand is how the 1-bit-per-pixel Mix Map is used to select between the programmed Foreground and Background ROP's.

A similar approach is used to draw transparent text (that is, text of a solid foreground color painted such that the Destination Map shows through in the background). It is a simple matter of setting the Foreground ROP to "Src", and the Background ROP to "Dst." Of course, it is also possible to make the Foreground ROP and Background ROP functions of any of the three pixel maps to obtain the desired result.

If the programmer wishes to operate as if the pixel maps are 16, 24, or 32-bits-per-pixel, each bit of the Mix Map must be replicated 2, 3, or 4 times, respectively, because the accelerator ALWAYS expands one BIT of the Mix Map to one BYTE of the other maps.

2.12.8.5 Clipping

For the most part, rectangular clipping of BitBlt operations must be done by the software. However, clipping with a data mask can be accomplished by using the Mix Map and programming the Background ROP to "Dest" and the Foreground ROP as desired. This allows each bit in the Mix Map to control whether the corresponding byte is processed or left alone.

The architecture of the accelerator LineDraw Engine provides some assistance to the clipping of lines; see Section 2.11.7.1 for a more detailed discussion of drawing clipped lines.

2.12.8.6 Bit Masking

A common graphics operation involves the use of a bit-oriented mask to control whether certain bits of the Destination data are operated on or not. An example of this is the standard VGA "Bit Mask" Register (GDC Indexed Register 8).

A bit value of 0 in the mask leaves the corresponding bit of the Destination data unchanged, and a bit value of 1 causes that bit of the Destination data to take on some value, which could be constant, or a function of Source and/or Destination data.

This type of operation is easily accomplished with the accelerator by using the Pattern Map to hold the desired Bit Mask value, and setting the Pattern X/Y Wrap to 4-by-1. Then the Foreground ROP can be chosen properly so that a bit value of 0 in the

Pattern Map applies a certain 2-operand ROP to that Destination bit, and a bit value of 1 in the Pattern Map applies another 2-operand ROP to that Destination bit. The table below summarizes the resultant Destination bit value ("Dst<n>") when various Foreground ROP's are applied:

ROP	Pat<n> == 0	Pat<n> == 1
CA	unchanged	Src<n>
AC	Src<n>	unchanged
FA	unchanged	1
0A	unchanged	0
8A	unchanged	Src<n> & Dst<n>
EA	unchanged	Src<n> | Dst<n>
6A	unchanged	Src<n> XOR Dst<n>
9A	unchanged	Src<n> XNOR Dst<n>

2.12.8.7 Compound BitBLT's

Some types of graphics operations require a complex combination of Pixel Maps. In these cases, it makes sense to perform the operation as two separate BitBLT's, or a "compound BitBLT." It is recommended, however, that the problem be thoroughly analyzed before deciding to use a compound BitBLT since in many cases the solution is subtle and may not be immediately apparent (such as the Bit Masking described in the previous section).

The following example is just one case of applying compound BitBLT's:

Consider a situation that has memory organized as one byte per pixel (and displayed as one byte per pixel), but each byte is broken into 2 nibbles, such that two different processes share the one byte, with each process appearing to have 4-bits per pixel.

One process would use a BitMask of 00001111, and the other uses a BitMask of 11110000. Now, in order to perform a BitBLT that uses the MixMap (stored in system memory) to select between a Foreground Color and a Background Color, the following two BitBLT's would be performed:

BitBLT #1

1. Program Data Routing to take MixMap data from the host.
2. Write Foreground Color to off-screen memory (replicated to fill a doubleword).
3. Write Background Color to off-screen memory (replicated to fill a doubleword).
4. Load Source Address Register with off-screen address where Foreground Color was stored.
5. Load Pattern Address Register with off-screen address where Background Color was stored.
6. Load Destination Address Register to point to an area of off-screen memory.
7. Load Pattern Wrap Register with 02 (4-by-1 wrap).
8. Load Source Wrap Register with 02 (4-by-1 wrap).
9. Load Foreground ROP Register with CC (SRCCOPY).
10. Load Background ROP Register with F0 (PATCOPY).
11. Perform the BitBLT.

The resultant map in off-screen memory is a color-expanded version of the MixMap.

BitBLT #2

1. Program Data Routing for Screen-to-Screen operation.
2. Write BitMask (00001111) to off-screen memory (replicated to fill a doubleword).

3. Load Foreground ROP Register with CA.
4. Load Source Address Register with off-screen address where previous BitBLT Destination Map is stored.
5. Load Destination Address Register with on-screen address of the actual Pixel Map that we wish to operate on.
6. Perform the BitBLT.

The resultant map will have the Foreground Color in the bottom nibble of every byte for which the MixMap had a corresponding bit value of "1", and have the Background Color in the bottom nibble of every byte for which the MixMap had a corresponding bit value of "0".

The upper nibble of each byte will be unmodified.

2.12.9 Accelerator Interrupts

The ET4000/W32p is capable of generating a system interrupt on four possible conditions:

1. Write Interrupt - This interrupt is generated while the queue is in the state of being "not-full." This status indicates that the queue is ready for another write to it. This is a state-triggered interrupt; i.e., the interrupt line is asserted while the queue is in the state of being "not-full." The interrupt is cleared by disabling it (writing a "0" to bit 0 of the ACL Interrupt Mask Register).
2. Read Interrupt - This interrupt is generated when the queue is empty and the accelerator goes from busy to idle, indicating that the accelerator is no longer performing a graphics operation, and will not start to perform another operation without a command from the host. The terminology of "read interrupt" conveys the fact that the host is ensured of reading correct results from the display memory and from any of the accelerator registers that are modified during the course of an accelerated graphics operation. This is an event-triggered interrupt; i.e., the interrupt line asserts when the accelerator goes from busy to idle, and stays asserted until the interrupt is cleared (by a write of "1" to the corresponding bit in the ACL Interrupt Status Register).
3. Write Fault Interrupt - This interrupt is generated when the host writes to the queue when it is full and the Sync Enable bit is "0." Under these conditions, the write is ignored. This is an event-triggered interrupt; i.e., the interrupt line asserts when the host write occurs, and stays asserted until the interrupt is cleared (by a write of "1" to the corresponding bit in the ACL Interrupt Status Register). This interrupt is also generated when a deadlock situation has occurred when passing Source or Mix data to the accelerator.
4. Read Fault Interrupt - This interrupt is generated when the host reads from an accelerated MMU aperture when the ADST bit (ACL Accelerator Status Register, bit 7) is "0," and the Sync Enable bit is "0." Under these conditions, the data returned for the read may be undefined. This is an event-triggered interrupt; i.e., the interrupt line asserts when the host read occurs, and stays asserted until the interrupt is cleared (by a write of "1" to the corresponding bit in the ACL Interrupt Status Register).

2.12.10 Accelerator State Save/Restore

The ET4000/W32p provides a mechanism for suspending an active graphics operation, saving the state of the operation, restoring the state of the operation, and resuming the operation. The Suspend/Save/Restore/Resume feature is often required by multi-tasking operating systems to allow several tasks to share the display hardware. Generally, the host will take an interrupt when a task-switch is required; the pseudo-code in the following sections outlines the steps required to save and restore the state of the Graphics Accelerator.

The mechanism for saving and restoring is quite different between the Revision A chip and other versions of the chip. For **Revision A**, the state of accelerator operations that require the host to participate in moving data cannot be Saved/Restored in the middle of the operation. In other words the operation must be allowed to complete before the accelerator state can be saved. With **Revision B**, this restriction was removed, also resulting in a new algorithm for Suspending, Saving, and Restoring. The new algorithm uses a backtracking approach to restore the state of the accelerator; the information to compute the backtrack amount is taken from the accelerator during the Suspending operation.

2.12.10.1 State-Save Interrupt Handler (Rev. A only)

Write "1" to bit 0 (SO) of ACL Suspend/Terminate Register. /* suspends operation */
while (STAT.RDST==1); /* wait for accel op to complete */

Write "0" to bit 0 (SO) of ACL Suspend/Terminate Register.

/* By now, accelerator is completely idle */

Read all accelerator registers from chip (including STAT) and save into local array, called SAVE1.

/* "shift" state from queue */
Write "1" to bit 0 (RSO) of ACL Operation State Register.

Read all accelerator registers from chip and save into local array, called SAVE2.

/* terminate operation and reset accelerator */
Write "1" to bit 4 (TO) of ACL Suspend/Terminate Register.

while (STAT.RDST==1); /* wait for Read-status OK */

Write "0" to bit 4 (TO) of ACL Suspend/Terminate Register.

Done with State Save.

2.12.10.2 State-Restore Interrupt Handler (Rev. A only)

Load the array SAVE1 back into chip (including STAT).

/* "shift" state from queue */
Write "1" to bit 0 (RSO) of ACL Operation State Register.

Load the array SAVE2 back into chip.

/* resume screen-to-screen op if necessary */
Write (SAVE1.STAT & 8) to ACL Operation State Register.

Done with State Restore.

2.12.10.3 State-Save Interrupt Handler

Please contact Tseng Labs for information on the State-Save Interrupt Handler for ET4000/W32p revisions other than Revision A.

2.12.10.4 State-Restore Interrupt Handler

Please contact Tseng Labs for information on the State-Restore Interrupt Handler for ET4000/W32p revisions other than Revision A.

3. ET4000/W32p Pin Descriptions

The ET4000/W32p provides a flexible interface to different types of CPU buses as well as options such as the image port, high color DAC, and hardware sprite. It is compatible with the following CPU buses:

- Local Bus—386DX; 486SX/DX/DX2; 32 bits
- VESA LBUS
- PCI Bus

A specific host bus type is selected by pulling the LBPI input pins to appropriate levels. If LBPI is low then PCI bus is selected. If LBPI is high then one of the local bus configurations is selected. A Power On Reset Initialize (PORI) scheme is used to determine different configurations for each bus type.

Section 3.2 lists the pin descriptions for six practical configurations. Designers can choose from among them to customize their W32p-based products. Section 3.3 lists the pin descriptions common to all specific configurations.

3.1 Power On Reset Initialize (PORI)

During the high-to-low transition of the RESET signal, PORI bits (DD<7:0>, RS<1:0>, CS<2:0>, PMEW*, BDE, and MUX<0>) are latched internally. These latched data bits are used to determine the host interface configuration. PORI bits are normally pulled high internally; these signals can be pulled down via resistor to ground, or driven low with a tri-state buffer during reset.

3.1.1 Description of Terms

I	= Input
O	= Output
IO	= Bidirectional
PWR	= Power input pin
TTL	= Pin has standard TTL input and output thresholds
CMOS	= Pin has standard CMOS input and output thresholds
S	= Schmitt Trigger on input
TS	= Tri-state
OC	= Open-collector (these are actually tri-state outputs, driven low, float high)
PU	= Internal passive pull-up
PD	= Internal passive pull-down
B2	= Output buffer can source/sink 2mA
B4	= Output buffer can source/sink 4mA
B8	= Output buffer can source/sink 8mA
High	= Voltage level between 2.0V and VDD (also abbreviated "H")
Low	= Voltage level between VSS and 0.7V (also abbreviated "L")
*	= Active Low
L	= Ouput is Low during RESET
H	= Ouput is High during RESET
Z	= Ouput is Tristate on RESET
U	= Output has active drive but state is undetermined during RESET
PCIDB	= PCI bus bi-directional ***Except Revisions A & B***
PCIT	= PCI output tristate ***Except Revisions A & B***

3.1.2 Configuration-Specific Definitions

F11100 =PCI pin configuration
F10011 =Local bus with external decode, no buffers, 16-bit pixel bus, no image port, feature connector control
F01000 =Local bus, no buffers, 8-bit pixel bus, no image port, feature connector control
F01011 =Local bus, no buffers, 16-bit pixel bus, no image port
F10001 =Local bus with external decode, no buffers, 8-bit pixel bus, 8-bit image port
F11100 =LBB pin configuration, all functions, multiplexed host address bus
COMMON=Applies to all pin configurations

See also Appendix E. Specific Configuration Pin-outs.

3.2 Pin Descriptions - Configuration Specific

3.2.1 PCI Bus Configuration F11100

SYMBOL	PIN#	I/O	TYPE	FUNCTION
AD<31:0>	2-9, 11-14, 16-25, 28-37	IO,Z	TTL,TS, PU,B8, PCIDB	32-bit multiplexed PCI address/data bus.
CBE<3:0>*	44,42, 41,40	I	TTL	PCI command /byte enable bus.
PAR	50	IO,Z	TTL,TS, PU,B8, PCIDB	PCI parity signal.
IDSEL	51	I	TTL,PU	PCI ID select signal.
FRAME*	47	I	TTL	Active low input signal indicating active PCI cycle.
BCLK	39	I	TTL,PU	PCI Bus system clock input.
IRDY*	45	I	TTL	PCI initiator READY signal.
DEVSEL*	48	O,Z	TTL,B8,TS, PCIT	PCI device select signal. When active this signal indicates the availability of ET4000/W32p's memory or I/O resources to the PCI Bus.
TRDY*	46	IO,Z	TTL,TS, OC,PU,B8, PCIDB	PCI target ready signal. When active this signal indicates the completion of a current data phase of a PCI access cycle.
RESET*	52	I	CMOS,S	Active low reset signal to initialize the ET4000/W32p. When low, the chip is held in the reset state.
ADDL	55	IO,H	TTL,TS, PU,B8	***Except Revisions A & B:*** When DVCK PORI is high, this is the active low signal used to latch AD<14:2> of the PCI bus for BIOS addresses. When DVCK PORI is high, this is the active high signal used to latch DD<7:2> into ROM address <7:2>. ***Revisions A & B ONLY:*** Active high signal used to latch AD<14:2> of the PCI bus for BIOS accesses.
STOP*	49	IO,Z	TTL,TS, PU,B8, PCIDB	This signal indicates that a ET4000/W32p is requesting termination of the current PCI cycle.
INT*	53	O,L	TTL,OC, TS,B8, PCIDB	***Except Revisions A & B:*** Active low tri-state interrupt request signal. ***Revisions A & B ONLY:*** Active high tri-state interrupt request signal.

3.2.1.1 Display Interface - Configuration F11100

SYMBOL	PIN#	I/O	TYPE	FUNCTION
AP<15:0>	78-85, 87-94	O,Z	TTL,TS, PU[1],B8	Display pixel data bus connects to bits <15:0> of external Digital to Analog Converter (DAC) pixel data inputs. [1]AP<15:8> have internal pullups.
BLANK*	76	O,Z	TTL,TS,B8	Active low display blank signal, when active indicates a blanking period. During blanking time, the video output line shall be cleared. This signal shall be used in conjunction with the digital video output in external DAC to produce the required R,G,B analog output.
BDE	77	IO,Z	TTL,PU,B8	CRTCB/Sprite display enable timing, synchronized to first and last pixel of CRTCB/Sprite display data of each line.
PCLK	75	O,Z	TTL,TS,B8	Used to sample pixel data AP<15:0>, BLANK, BDE, and SP<1:0>.
VS	73	O,Z	TTL,TS,B4	Vertical retrace synchronization, supplied to the CRT monitor.
HS	72	O,Z	TTL,TS,B4	Horizontal retrace synchronization, supplied to the CRT monitor.
EDCK	115	I	TTL,PU	Input from the feature connector. Low : Pixel clock (PCLK) from the W32p will be tristated. High: Pixel clock is not tristated. ***Except Revisions A & B:*** This signal is sampled once per scan line, and can be shared with OEOA*. See Section 6.4.2.1 OE Controlled Interleave DRAM Interface.
ESYC	116	I	TTL,PU	Input from the feature connector. Low : Vertical and horizontal sync (VS,HS), and the BLANK from the W32p will be tristated. High: VS,HS, and BLANK are not tristated. ***Except Revisions A & B:*** This signal is sampled once per scan line, and can be shared with OEEB*. See Section 6.4.2.1 OE Controlled Interleave DRAM Interface.
EVID	117	I	TTL,PU	Input from the feature connector. Low : The DAC pixel bus (AP<15:0>) from the W32p will be tristated. High: AP<15:0> are not tristated. ***Except Revisions A & B:*** This signal is sampled once per scan line, and can be shared with OEEA*. See Section 6.4.2.1 OE Controlled Interleave DRAM Interface.

3.2.1.2 Image Port Interface - Configuration F11100

The Image Port Interface provides image data as well as command decode for an external image processing device. All of the pins for the Image Port are shared with other chip interfaces. IMA Indexed Register F7, bit 0 determines whether the pins are used for Image Port functions, or other purposes.

SYMBOL	PIN#	I/O	TYPE	FUNCTION
IMD<15:8>	95, 98-104	IO,Z	TTL,TS, PU,B2	Upper byte of Image Port data bus. Not used on W32p. Reserved for future enhancements.
IMD<7:0>	106-113	IO,Z	TTL,TS, PU,B2	Image Port Data bus. These signals are inputs from an external image processing device.
IDMK<1>	118	IO,Z	TTL ,TS, PU,B2	Image Port Data Mask for upper byte of Image Port data bus.Not used on W32p. Reserved for future enhancements. Low : ET4000/W32p ignores the current data byte; it only steps the internal address sequencer. High: ET4000/W32p accepts the current data byte and writes it to display memory.
IDMK<0>	120	IO,Z	TTL ,TS, PU,B2	Image Port Data Mask. Low : ET4000/W32p ignores the current address sequencer. High: ET4000/W32p accepts the current data byte and writes it to display memory.
IXWQ*	125	IO,Z	TTL,TS, PU,B2	External Write Request. IM<7:0> and IDMK are strobed into the ET4000/W32p during the low to high transition of this signal.
IXRD	124	IO,Z	TTL,TS, PU,B8	External Data Port Ready. When active, the external image processor can transfer the next image data byte by pulsing IXWQ*. This signal should be synchronized by the external processor.
IXFS	121	IO,Z	TTL,PU, TS,B2	External Frame Synchronization. Low to high edge indicates the start of a new frame of image data.
IXLS	122	IO,Z	TTL,PU, TS,B2	External Line Synchronization. Low to high edge indicates the start of a new line of image data.
IXOF	123	IO,Z	TTL,PU, TS,B2	External Odd Field Status. This signal indicates odd or even field information during interlaced image data transfer. Should be pulled low if data transfer format is non-interlaced.
IXCM*	126	O,Z	TTL,TS, PU,B4	External Command. Active low external memory-mapped register decode. The external device, such as an Image Processor, can use this signal as a read/write command for connection to the host bus.

3.2.2 Bus Interface Configuration F10011

SYMBOL	PIN#	I/O	TYPE	FUNCTION
D<31:0>	2-9, 11-14, 16-25, 28-37	IO,Z	TTL,TS, PU,B8	32-bit host data bus.
A<27:2>	95, 98-104, 126, 121-125, 120, 106-113, 54-55, 118	IO,Z	TTL,TS, PU,B2, B4,B8	Host address bus. These signals, along with SEGI are used to decode and address the W32p's memory for CPU access. These signals are strictly inputs.The W32p's IO registers use A<15:2> for address and decode.
SEGI	49	IO,Z	TTL,TS, PU,B8	Host bus memory access enable. This pin is used to decode the remaining upper address signals.In this configuration this would be A<31:28>. However, if partial decoding after 512MB is acceptable, this signal may be attached directly to A<28> of the host bus.
BE3*	44	I	TTL	Active low input signal to indicate that valid data will be transferred on D<31:24>.
BE2*	42	I	TTL	Active low input signal to indicate that valid data will be transferred on D<23:16>.
BE1	41	I	TTL	Active low input signal to indicate that valid data will be transferred on D<15:8>.
BE0	40	I	TTL	Active low input signal to indicate that valid data will be transferred on D<7:0>.
MIO*	50	IO,Z	TTL,TS, PU,B8	Memory or I/O status input. Low : I/O. High: Memory.
W/R*	51	I	TTL,PU	This signal defines the current bus cycle as read or write. Low : Read. High: Write.
ADS*	47	I	TTL	Active low input signal indicates address and status valid.
LCLK	39	I	TTL,PU	Local Bus system clock input.
RDYRTN*	45	I	TTL	Ready Return. Active low signal used to extend the current Local Bus cycle by holding READ data active.
LOCAL*	48	O,Z	TTL,B8,TS	Active low signal to indicate the availability of ET4000/W32p's memory or I/O resources to the Local Bus.
READY*	46	IO,Z	TTL,TS, OC,PU,B8,	Active low output indicates termination of a host access cycle.
RESET	52	I	CMOS,S	Active high reset signal to initialize the ET4000/W32p. When high, the chip is held in the reset state.
INT*	53	O,L	TTL,OC,B8, TS	Interrupt Request. Active tri-state signal to indicate an interrupting condition.

3.2.2.1 Display Interface - Configuration F10011

SYMBOL	PIN#	I/O	TYPE	FUNCTION
AP<15:0>	78-85, 87-94	O,Z	TTL,TS, PU[1],B8	Display pixel data bus connects to bits <15:0> of external Digital to Analog Converter (DAC) pixel data inputs. [1]AP<15:8> have internal pullups.
BLANK*	76	O,Z	TTL,TS,B8	Active low display blank signal, when active indicates a blanking period. During blanking time, the video output line shall be cleared. This signal shall be used in conjunction with the digital video output in external DAC to produce the required R,G,B analog output.
BDE	77	IO,Z	TTL,PU,B8	CRTCB/Sprite display enable timing, synchronized to first and last pixel of CRTCB/Sprite display data of each line.
PCLK	75	O,Z	TTL,TS,B8	Used to sample pixel data AP<15:0>, BLANK, BDE, and SP<1:0>.
VS	73	O,Z	TTL,TS,B4	Vertical retrace synchronization, supplied to the CRT monitor.
HS	72	O,Z	TTL,TS,B4	Horizontal retrace synchronization, supplied to the CRT monitor.
EDCK	115	I	TTL,PU	Input from the feature connector. Low : Pixel clock (PCLK) from the W32p will be tristated. High: Pixel clock is not tristated. ***Except Revisions A & B:*** This signal is sampled once per scan line, and can be shared with OEOA*. See Section 6.4.2.1 OE Controlled Interleave DRAM Interface.
ESYC	116	I	TTL,PU	Input from the feature connector. Low : Vertical and horizontal sync (VS,HS), and the BLANK from the W32p will be tristated. High: VS,HS, and BLANK are not tristated. ***Except Revisions A & B:*** This signal is sampled once per scan line, and can be shared with OEEB*. See Section 6.4.2.1 OE Controlled Interleave DRAM Interface.
EVID	117	I	TTL,PU	Input from the feature connector. Low : The DAC pixel bus (AP<15:0>) from the W32p will be tristated. High: AP<15:0> are not tristated. ***Except Revisions A & B:*** This signal is sampled once per scan line, and can be shared with OEEA*. See Section 6.4.2.1 OE Controlled Interleave DRAM Interface.

3.2.3 Bus Interface Configuration F01000

SYMBOL	PIN#	I/O	TYPE	FUNCTION
D<31:0>	2-9, 11-14, 16-25, 28-37	IO,Z	TTL,TS, PU,B8	32-bit host data bus.
A<31:2>	95, 98-104, 49,83-85 126, 121-125, 120, 106-113, 54-55, 118	IO,Z	TTL,TS, PU,B2, B4,B8	Host address bus. These signals are used to decode and address the W32p's memory for CPU access. These signals are strictly inputs. The W32p's IO registers use A<15:2> for address and decode.
BE3*	44	I	TTL	Active low input signal to indicate that valid data will be transferred on D<31:24>.
BE2*	42	I	TTL	Active low input signal to indicate that valid data will be transferred on D<23:16>.
BE1	41	I	TTL	Active low input signal to indicate that valid data will be transferred on D<15:8>.
BE0	40	I	TTL	Active low input signal to indicate that valid data will be transferred on D<7:0>.
MIO*	50	IO,Z	TTL,TS, PU,B8	Memory or I/O status input. Low : I/O. High: Memory.
W/R*	51	I	TTL,PU	This signal defines the current bus cycle as read or write. Low : Read. High: Write.
ADS*	47	I	TTL	Active low input signal indicates address and status valid.
LCLK	39	I	TTL,PU	Local Bus system clock input.
RDYRTN*	45	I	TTL	Ready Return. Active low signal used to extend the current Local Bus cycle by holding READ data active.
LOCAL*	48	O,Z	TTL,B8,TS	Active low signal to indicate the availability of ET4000/W32p's memory or I/O resources to the Local Bus.
READY*	46	IO,Z	TTL,TS, OC,PU,B8,	Active low output indicates termination of a host access cycle.
RESET	52	I	CMOS,S	Active high reset signal to initialize the ET4000/W32p. When high, the chip is held in the reset state.
INT*	53	O,L	TTL,OC,B8, TS	Interrupt Request. Active tri-state signal to indicate an interrupting condition.

3.2.3.1 Display Interface - Configuration F01000

SYMBOL	PIN#	I/O	TYPE	FUNCTION
AP<7:0>	87-94	O,Z	TTL,TS, B8	Display pixel data bus connects to bits <7:0> of external Digital to Analog Converter (DAC) pixel data inputs.
BLANK*	76	O,Z	TTL,TS,B8	Active low display blank signal, when active indicates a blanking period. During blanking time, the video output line shall be cleared. This signal shall be used in conjunction with the digital video output in external DAC to produce the required R,G,B analog output.
BDE	77	IO,Z	TTL,PU,B8	CRTCB/Sprite display enable timing, synchronized to first and last pixel of CRTCB/Sprite display data of each line.
PCLK	75	O,Z	TTL,TS,B8	Used to sample pixel data AP<7:0>, BLANK, BDE, and SP<1:0>.
VS	73	O,Z	TTL,TS,B4	Vertical retrace synchronization, supplied to the CRT monitor.
HS	72	O,Z	TTL,TS,B4	Horizontal retrace synchronization, supplied to the CRT monitor.
EDCK	115	I	TTL,PU	Input from the feature connector. Low : Pixel clock (PCLK) from the W32p will be tristated. High: Pixel clock is not tristated. ***Except Revisions A & B:*** This signal is sampled once per scan line, and can be shared with OEOA*. See Section 6.4.2.1 OE Controlled Interleave DRAM Interface.
ESYC	116	I	TTL,PU	Input from the feature connector. Low : Vertical and horizontal sync (VS,HS), and the BLANK from the W32p will be tristated. High: VS,HS, and BLANK are not tristated. ***Except Revisions A & B:*** This signal is sampled once per scan line, and can be shared with OEEB*. See Section 6.4.2.1 OE Controlled Interleave DRAM Interface.
EVID	117	I	TTL,PU	Input from the feature connector. Low : The DAC pixel bus (AP<7:0>) from the W32p will be tristated. High: AP<7:0> are not tristated. ***Except Revisions A & B:*** This signal is sampled once per scan line, and can be shared with OEEA*. See Section 6.4.2.1 OE Controlled Interleave DRAM Interface.

3.2.4 Bus Interface Configuration F01011

SYMBOL	PIN#	I/O	TYPE	FUNCTION
D<31:0>	2-9, 11-14, 16-25, 28-37	IO,Z	TTL,TS, PU,B8	32-bit host data bus.
A<31:2>	95, 98-104, 49,115-117, 126,121-125, 120, 106-113, 54-55, 118	IO,Z	TTL,TS, PU,B2, B4,B8	Host address bus. These signals are used to decode and address the W32p's memory for CPU access. These signals are strictly inputs. The W32p's IO registers use A<15:2> for address and decode.
BE3*	44	I	TTL	Active low input signal to indicate that valid data will be transferred on D<31:24>.
BE2*	42	I	TTL	Active low input signal to indicate that valid data will be transferred on D<23:16>.
BE1	41	I	TTL	Active low input signal to indicate that valid data will be transferred on D<15:8>.
BE0	40	I	TTL	Active low input signal to indicate that valid data will be transferred on D<7:0>.
MIO*	50	IO,Z	TTL,TS, PU,B8	Memory or I/O status input. Low : I/O. High: Memory.
W/R*	51	I	TTL,PU	This signal defines the current bus cycle as read or write. Low : Read. High: Write.
ADS*	47	I	TTL	Active low input signal indicates address and status valid.
LCLK	39	I	TTL,PU	Local Bus system clock input.
RDYRTN*	45	I	TTL	Ready Return. Active low signal used to extend the current Local Bus cycle by holding READ data active.
LOCAL*	48	O,Z	TTL,B8,TS	Active low signal to indicate the availability of ET4000/W32p's memory or I/O resources to the Local Bus.
READY*	46	IO,Z	TTL,TS, OC,PU,B8,	Active low output indicates termination of a host access cycle.
RESET	52	I	CMOS,S	Active high reset signal to initialize the ET4000/W32p. When high, the chip is held in the reset state.
INT*	53	O,L	TTL,OC, TS,B8,	Interrupt Request. Active tri-state signal to indicate an interrupting condition.

3.2.4.1 Display Interface - Configuration F01011

SYMBOL	PIN#	I/O	TYPE	FUNCTION
AP<15:0>	78-85, 87-94	O,Z	TTL,TS, PU[1],B8	Display pixel data bus connects to bits <15:0> of external Digital to Analog Converter (DAC) pixel data inputs. [1]AP<15:8> have internal pullups.
BLANK*	76	O,Z	TTL,TS,B8	Active low display blank signal, when active indicates a blanking period. During blanking time, the video output line shall be cleared. This signal shall be used in conjunction with the digital video output in external DAC to produce the required R,G,B analog output.
BDE	77	IO,Z	TTL,PU,B8	CRTCB/Sprite display enable timing, synchronized to first and last pixel of CRTCB/Sprite display data of each line.
PCLK	75	O,Z	TTL,TS,B8	Used to sample pixel data AP<15:0>, BLANK, BDE and SP<1:0>.
VS	73	O,Z	TTL,TS,B4	Vertical retrace synchronization, supplied to the CRT monitor.
HS	72	O,Z	TTL,TS,B4	Horizontal retrace synchronization, supplied to the CRT monitor.

3.2.5 Bus Configuration F10001

SYMBOL	PIN#	I/O	TYPE	FUNCTION
D<31:0>	2-9, 11-14, 16-25, 28-37	IO,Z	TTL,TS, PU,B8	32-bit host data bus.
A<22:2>	115-117, 95,98-103, 78-85, 54,55, 118	IO,Z	TTL,TS, PU,B2, B4,B8	Host address bus. These signals are used to decode and address the W32p's memory for CPU access. These signals are strictly inputs. The W32p's IO registers use A<15:2> for address and decode.
SEGI	49	IO,Z	TTL,TS, PU,B8	Host bus memory access enable. This pin is used to decode the remaining upper address signals. In this configuration this would be A<31:23>.
BE3*	44	I	TTL	Active low input signal to indicate that valid data will be transferred on D<31:24>.
BE2*	42	I	TTL	Active low input signal to indicate that valid data will be transferred on D<23:16>.
BE1	41	I	TTL	Active low input signal to indicate that valid data will be transferred on D<15:8>.
BE0	40	I	TTL	Active low input signal to indicate that valid data will be transferred on D<7:0>.
MIO*	50	IO,Z	TTL,TS, PU,B8	Memory or I/O status input. Low : I/O. High: Memory.
W/R*	51	I	TTL,PU	This signal defines the current bus cycle as read or write. Low : Read. High: Write.
ADS*	47	I	TTL	Active low input signal indicates address and status valid.
LCLK	39	I	TTL,PU	Local Bus system clock input.
RDYRTN*	45	I	TTL	Ready Return. Active low signal used to extend the current Local Bus cycle by holding READ data active.
LOCAL*	48	O,Z	TTL,B8,TS	Active low signal to indicate the availability of ET4000/W32p's memory or I/O resources to the Local Bus.
READY*	46	IO,Z	TTL,TS, OC,PU,B8,	Active low output indicates termination of a host access cycle.
RESET	52	I	CMOS,S	Active high reset signal to initialize the ET4000/W32p. When high, the chip is held in the reset state.
INT*	53	O,L	TTL,OC,B8, TS	Interrupt Request. Active tri-state signal to indicate an interrupting condition.

3.2.5.1 Display Interface - Configuration F10001

SYMBOL	PIN#	I/O	TYPE	FUNCTION
AP<7:0>	87-94	O,Z	TTL,TS, PU,B8	Display pixel data bus connects to bits <7:0> of external Digital to Analog Converter (DAC) pixel data inputs.
BLANK*	76	O,Z	TTL,TS,B8	Active low display blank signal, when active indicates a blanking period. During blanking time, the video output line shall be cleared. This signal shall be used in conjunction with the digital video output in external DAC to produce the required R,G,B analog output.
BDE	77	IO,Z	TTL,PU,B8	CRTCB/Sprite display enable timing, synchronized to first and last pixel of CRTCB/Sprite display data of each line.
PCLK	75	O,Z	TTL,TS,B8	Used to sample pixel data AP<7:0>, BLANK, BDE, and SP<1:0>.
VS	73	O,Z	TTL,TS,B4	Vertical retrace synchronization, supplied to the CRT monitor.
HS	72	O,Z	TTL,TS,B4	Horizontal retrace synchronization, supplied to the CRT monitor.

3.2.5.2 Image Port Interface F10001

The Image Port Interface provides image data as well as command decode for an external image processing device. All of the pins for the Image Port are shared with other chip interfaces. IMA Indexed Register F7, bit 0 determines whether the pins are used for Image Port functions, or other purposes.

SYMBOL	PIN#	I/O	TYPE	FUNCTION
IMD<7:0>	106-113	IO,Z	TTL,TS, PU,B2	Image Port Data bus. These signals are inputs from an external image processing device.
IDMK	120	IO,Z	TTL ,TS, PU,B2	Image Port Data Mask. Low : ET4000/W32p ignores the current data byte; it only steps the internal address sequencer. High: ET4000/W32p accepts the current data byte and writes it to display memory.
IXWQ*	125	IO,Z	TTL,TS, PU,B2	External Write Request. IM<7:0> and IDMK are strobed into the ET4000/W32p during the low to high transition of this signal.
IXRD	124	IO,Z	TTL,TS, PU,B8	External Data Port Ready. When active, the external image processor can transfer the next image data byte by pulsing IXWQ*. This signal should be synchronized by the external processor.
IXFS	121	IO,Z	TTL,PU, TS,B2	External Frame Synchronization. Low to high edge indicates the start of a new frame of image data.
IXLS	122	IO,Z	TTL,PU, TS,B2	External Line Synchronization. Low to high edge indicates the start of a new line of image data.
IXOF	123	IO,Z	TTL,PU, TS,B2	External Odd Field Status. This signal indicates odd or even field information during interlaced image data transfer. Should be pulled low if data transfer format is non-interlaced.
IXCM*	126	O,Z	TTL,TS, PU,B4	External Command. Active low external memory-mapped register decode. The external device, such as an Image Processor, can use this signal as a read/write command for connection to the host bus.

3.2.6 LBB Bus Configuration F11100

SYMBOL	PIN#	I/O	TYPE	FUNCTION
AD<31:0>	2-9, 11-14, 16-25, 28-37	IO,Z	TTL,TS, PU,B8	32-bit multiplexed host address/data bus.
BE3*	44	I	TTL	Active low input signal to indicate that valid data will be transferred on AD<31:24>.
BE2*	42	I	TTL	Active low input signal to indicate that valid data will be transferred on AD<23:16>.
BE1	41	I	TTL	Active low input signal to indicate that valid data will be transferred on AD<15:8>.
BE0	40	I	TTL	Active low input signal to indicate that valid data will be transferred on AD<7:0>.
MIO*	50	IO,Z	TTL,TS, PU,B8	Memory or I/O status input. Low : I/O. High: Memory.
W/R*	51	I	TTL,PU	This signal defines the current bus cycle as read or write. Low : Read. High: Write.
ADS*	47	I	TTL	Active low input signal indicates address and status valid.
LCLK	39	I	TTL,PU	Local Bus system clock input.
RDYRTN*	45	I	TTL	Ready Return. Active low signal used to extend the current Local Bus cycle by holding READ data active.
LOCAL*	48	O,Z	TTL,B8,TS	Active low signal to indicate the availability of ET4000/W32p's memory or I/O resources to the Local Bus.
READY*	46	IO,Z	TTL,TS, OC,PU,B8,	Active low output indicates termination of a host access cycle.
RESET	52	I	CMOS,S	Active high reset signal to initialize the ET4000/W32p. When high, the chip is held in the reset state.
DE*	54	IO,H	TTL,TS, PU,B8	Active low signal used to enable AD<31:0> data bus buffer.
AE*	55	IO,H	TTL,TS, PU,B8	Active low signal used to enable AD<31:0> address bus buffer.
DIR	49	IO,Z	TTL,TS, PU,B8	Signal used to control the direction of bi-directional data bus buffers connected to AD<31:0> between ET4000/W32p and the local data bus. Low : The signal is low during I/O and memory write operations. High: The signal is high during I/O and memory read operations.
INT*	53	O,L	TTL,OC, TS,B8,	Interrupt Request. Active tri-state signal to indicate an interrupting condition.

3.2.6.1 Display Interface - Configuration F11100

SYMBOL	PIN#	I/O	TYPE	FUNCTION
AP<15:0>	78-85, 87-94	O,Z	TTL,TS, PU[1],B8	Display pixel data bus connects to bits <15:0> of external Digital to Analog Converter (DAC) pixel data inputs. [1]AP<15:8> have internal pullups.
BLANK*	76	O,Z	TTL,TS,B8	Active low display blank signal, when active indicates a blanking period. During blanking time, the video output line shall be cleared. This signal shall be used in conjunction with the digital video output in external DAC to produce the required R,G,B analog output.
BDE	77	IO,Z	TTL,PU,B8	CRTCB/Sprite display enable timing, synchronized to first and last pixel of CRTCB/Sprite display data of each line.
PCLK	75	O,Z	TTL,TS,B8	Used to sample pixel data AP<15:0>, BLANK, BDE, and SP<1:0>.
VS	73	O,Z	TTL,TS,B4	Vertical retrace synchronization, supplied to the CRT monitor.
HS	72	O,Z	TTL,TS,B4	Horizontal retrace synchronization, supplied to the CRT monitor.
EDCK	115	I	TTL,PU	Input from the feature connector. Low : Pixel clock (PCLK) from the W32p will be tristated. High: Pixel clock is not tristated. ***Except Revisions A & B:*** This signal is sampled once per scan line, and can be shared with OEOA*. See Section 6.4.2.1 OE Controlled Interleave DRAM Interface.
ESYC	116	I	TTL,PU	Input from the feature connector. Low : Vertical and horizontal sync (VS,HS), and the BLANK from the W32p will be tristated. High: VS,HS, and BLANK are not tristated. ***Except Revisions A & B:*** This signal is sampled once per scan line, and can be shared with OEEB*. See Section 6.4.2.1 OE Controlled Interleave DRAM Interface.
EVID	117	I	TTL,PU	Input from the feature connector. Low : The DAC pixel bus (AP<15:0>) from the W32p will be tristated. High: AP<15:0> are not tristated. ***Except Revisions A & B:*** This signal is sampled once per scan line, and can be shared with OEEA*. See Section 6.4.2.1 OE Controlled Interleave DRAM Interface.

3.2.6.2 Image Port Interface - Configurations 11100

The Image Port Interface provides image data as well as command decode for an external image processing device. All of the pins for the Image Port are shared with other chip interfaces. IMA Indexed Register F7, bit 0 determines whether the pins are used for Image Port functions, or other purposes.

SYMBOL	PIN#	I/O	TYPE	FUNCTION
IMD<15:8>	95, 98-104	IO,Z	TTL,TS, PU,B2	Upper byte of Image Port data bus. Not used on W32p. *Reserved for future enhancements.*
IMD<7:0>	106-113	IO,Z	TTL,TS, PU,B2	Image Port Data bus. These signals are inputs from an external image processing device.
IDMK<1>	118	IO,Z	TTL ,TS, PU,B2	Image Port Data Mask for upper byte of Image Port data bus. Not used on W32p. *Reserved for future enhancements.* Low : ET4000/W32p ignores the current data byte; it only steps the internal address sequencer. High: ET4000/W32p accepts the current data byte and writes it to display memory.
IDMK<0>	120	IO,Z	TTL ,TS, PU,B2	Image Port Data Mask. Low : ET4000/W32p ignores the current data byte; it only steps the internal address sequencer. High: ET4000/W32p accepts the current data byte and writes it to display memory.
IXWQ*	125	IO,Z	TTL,TS, PU,B2	External Write Request. IM<7:0> and IDMK are strobed into the ET4000/W32p during the low to high transition of this signal.
IXRD	124	IO,Z	TTL,TS, PU,B8	External Data Port Ready. When active, the external image processor can transfer the next image data byte by pulsing IXWQ*. This signal should be synchronized by the external processor.
IXFS	121	IO,Z	TTL,PU, TS,B2	External Frame Synchronization. Low to high edge indicates the start of a new frame of image data.
IXLS	122	IO,Z	TTL,PU, TS,B2	External Line Synchronization. Low to high edge indicates the start of a new line of image data.
IXOF	123	IO,Z	TTL,PU, TS,B2	External Odd Field Status. This signal indicates odd or even field information during interlaced image data transfer. Should be pulled low if data transfer format is non-interlaced.
IXCM*	126	O,Z	TTL,TS, PU,B4	External Command. Active low external memory-mapped register decode. The external device, such as an Image Processor, can use this signal as a read/write command for connection to the host bus.

3.3 Pin Descriptions - Common

3.3.1 Bus Configurations and PORI - Common

SYMBOL	PIN#	I/O	TYPE	FUNCTION
LBPI	57	I	TTL	Local bus/PCI bus configuration pin. This is a static state signal and is *not latched with RESET*. Low : PCI bus. High: Local bus.
FCG<4:0>	204,203, 205,62, 63	IO,Z	TTL,TS, PU,B2, B4	Chip feature configuration bits. Pins are shared with CS<1>, CS<0>, CS<2>, RS<1>, and RS<0>, respectively.See Section 6.2.1for configurations.
SNPE	66	IO,Z	TTL,TS, PU,B4	Snoop Enable. When snoop feature is enabled, the W32p will not respond with LOCAL* and READY* during I/O writes to 03C6-03C9. Pin is shared with DD<5>. Low : External RAMDAC snoop enabled. High: External RAMDAC snoop disabled.
DELC	64	IO,Z	TTL,TS, PU,B4	Command delay. Pin is shared with DD<7>. Low : Internal command starts at first T2 after the low-to-high transition of ADS*. High: Internal command starts at first T2.
DISB	67	IO,Z	TTL,TS, PU,B4	Disable BIOS. Pin is shared with DD<4>. Low : Enable BIOS operation via DD<7:0> bus. High: Disable BIOS operation via DD<7:0> bus.
		Except Revision A		Low: ROME* is generated for BIOS ROM residing on the W32p's DD<7:0> data bus. ROME* will be generated multiple times per CPU access cycle depending on the BE<3:0>* signals. Also, LOCAL* and READY* will be generated. High: ROME* is generated based on address decode only. LOCAL* and READY* will not be generated. ROME* is intended to be used as the chip select for the BIOS ROM which resides on an external bus (such as ISA bus).
DVCK	65	IO,Z	TTL,TS, PU,B4	Internal local bus (LCLK) clock divide. Pin is shared with DD<6>. *Except for Revisions A & B in PCI mode* this signal is used to enable the PCI ROM address bus function of DD<7:0>.In local bus mode, this PORI bit controls DVCK *only* in all revisions. Low : LCLK = BCLK High: LCLK = BCLK/2
		Except Revisions A & B		Low (PCI bus mode): Disables PCI ROM address function of DD<7:0>. PCI ROM address derived from AD<14:2>. High (PCI bus mode: Enables PCI ROM address function of DD<7:0>.
IOD<2:0>	59,77, 208	IO,Z	TTL,TS, PU,B2, B4,B8	CRTCB/Sprite I/O register map for I/O addresses 21xA, 21xB where x = IOD<2:0>. Pins are shared with PMEW, BDE, and MUX0, respectively. This affects the I/O address mapping. See also Table 5.0-1 and section 6.2.1 Local Bus PORI.
MONID<3:0>	68-71	IO,Z	TTL,TS, PU,B4	Bits 0 and 1 can be read back via Status Register 0. Bits 3 and 2 can be read back via the Feature Control Register. Pins are shared with DD<3:0>.

3.3.2 Display Memory Interface - Common

3.3.2.1 Bank A (MD<15:0>)

SYMBOL	PIN#	I/O	TYPE	FUNCTION
MD<15:0>	163-158, 156-148, 146	IO,Z	TTL,TS, PU,B4,	Display memory data bus.
AA<9:0>	138-129	O,U	TTL,TS,B8	Row/column time multiplexed address bus outputs.
MWA*	145	O,H	TTL,TS,B8	Active low write command output.
RASA*	144	O,H	TTL,TS,B8	Active low row address strobe.
CAS<0>*	139	O,H	TTL,TS,B4	Active low column address strobe for MD<7:0>.
CAS<1>*	140	O,H	TTL,TS,B4	Active low column address strobe for MD<15:8>.
CAS<4>*	194	O,H	TTL,TS,B4	Active low column address strobe for MD<7:0> (interleaved).
CAS<5>*	195	O,H	TTL,TS,B4	Active low column address strobe for MD<15:8> (interleaved).

3.3.2.2 Bank B (MD<31:16>)

SYMBOL	PIN#	I/O	TYPE	FUNCTION
MD<31:16>	181-166	IO,Z	TTL,TS, PU,B4	Display memory data bus.
AB<9:0>	192-183	O,U	TTL,TS,B8	Row/column time multiplexed address bus outputs.
MWB*	199	O,H	TTL,TS,B8	Active low write command output.
RASB*	198	O,H	TTL,TS,B8	Active low row address strobe.
CAS<2>*	141	O,H	TTL,TS,B4	Active low column address strobe for MD<23:16>.
CAS<3>*	143	O,H	TTL,TS,B4	Active low column address strobe for MD<31:24>.
CAS<6>*	196	O,H	TTL,TS,B4	Active low column address strobe for MD<23:16> (interleaved).
CAS<7>*	197	O,H	TTL,TS,B4	Active low column address strobe for MD<31:24> (interleaved).

3.3.3 Clock Interface - Common

SYMBOL	PIN#	I/O	TYPE	FUNCTION
SCLK W32p	206	I	TTL	System Clock used to generate all display memory timing and ET4000/ internal sequencing.
MCLK	202	I	TTL	A variable frequency clock used to generate the necessary video, vertical and horizontal timing for the ET4000/W32p. The programmable clock select output pins (CS<4:0>) can be used to select the proper MCLK frequency for the display mode.
CS<2:0>	205-203	O,Z	TTL,B2, PU,TS	Clock Select. Used to select 1 of 8 possible MCLK frequencies.

3.3.4 Peripheral Interface - Common

SYMBOL	PIN#	I/O	TYPE	FUNCTION
DD<7:0>	64-71	IO,Z	TTL,PU, TS,B4	Peripheral Data Bus and PCI ROM address bus. These signals are used to transfer host bus data to/from the RAMDAC, BIOS ROM, and IMA devices. Except for Revisions A & B, these signals can provide the ROM address which needs to be latched with ADDL and ROME* in PCI mode. The PCI ROM address function is enabled by the DVCK PORI pin. See Section 6.3 PCI ROM.
RS<1:0>	62-63	IO,Z	TTL,TS, PU,B4	Lower 2 bits of peripheral address bus for RAMDAC and BIOS ROM. RS<1> is the write strobe for the IMA device's host interface.
PMER*	58	O,H	TTL,B2	Active low external Digital to Analog Converter (DAC) register read command.
PMEW*	59	IO,Z	TTL,TS, PU,B4	Active low external Digital to Analog Converter (DAC) register write command.
ROME*	56	O,H	TTL,B2	Active low signal, used to enable external BIOS ROM during ROM read decode. NOTES: ***Except Revision A***: The DISB PORI input affects the definition of ROME*. Low : ROME* = ROM space decode and bus read operation. High: ROME* = ROM space decode only. (***Except Revision A*** — see Sections 3.3.1 and 6.2.3.1.) ***Except Revisions A & B:*** In PCI mode, when DVCK PORI is low, this signal is used to latch DD<6:0> into ROM addresses <14:8> and also enable the ROM.

3.3.5 Multiplexed Signals - Common

SYMBOL	PIN#	I/O	TYPE	FUNCTION
CS<3> - PCI	54	IO,H	TTL,TS, PU,B8	Clock Select. Additional MCLK select signal. This signal is available on this pin in PCI mode only. It is always available on this pin in PCI mode and may also appear on the MUX<0> pin depending on the conditions specified by the table below.
MUX<1:0>	1,208	IO,Z	TTL,TS, PU,B2	Multiplexed pins for Input Status (SWSE), Synchronous Reset (SYNR), Clock Select (CS<4:3>), and Sprite Data (SP<1:0>). IMA Indexed Register F7 bits 6 and 5 determine the function of these pins.

MUX<1>	F7<6>	CRTCB EF<1:0>	ATC 16<6>	Value	IO
	0	X	X	SWSE	I
	1	0,0	X	SP<1>	O
	1	≠0,0	0	CS<4>	O
	1	≠0,0	1	TKN<1>	O

MUX<0>	F7<5>	CRTCB EF<1:0>	ATC 16<6>	Value	IO
	0	X	X	SYNR	I
	1	0,0	X	SP<0>	O
	1	≠0,0	0	CS<3>	O
	1	≠0,0	1	TKN<0>	O

SYNR — Active high synchronous reset signal. When active, it indicates a request to reset the ET4000/W32p's internal LINE and CHARACTER counters.

SWSE — Input status, can be read back via Input Status Register 0, bit 4.

TKN<1:0> — Token Status output.

If CRTC Index 35, bit 5 = 0, then TKN<1> = ET4000/W32p's MCU is processing FONT cycle.
TKN<0> = ET4000/W32p's MCU is processing pixel cycle.
If CRTC Index 35, bit 5 = 1, then TKN<1> = interlace mode is active.
TKN<0> = even field.

SP<1:0> — Sprite data bus to external DAC. Requires DAC with sprite inputs.

CS<4> — Clock Select. Additional MCLK select signal.

CS<3> — Clock Select. Additional MCLK select signal.

Except Revisions A & B: MUX<1> can be shared between SWSE (input) and OEOB* (output). This occurs when CRTC Indexed Register 37 <2> is set to 0. Additionally, MUX<1> must be in input mode (when IMA Indexed Register F7 <6> = 0). See Section 6.4.2.1 OE Controlled Interleave DRAM Interface.

3.3.6 Power Source Interface - Common

SYMBOL	PIN#	I/O	TYPE	FUNCTION
VSS	207,200, 193,182, 164,157, 142,127, 119,105, 96,86, 74,60, 38,26, 15	-	PWR	Ground.
VDD	201,165, 147,128, 114,97, 61,43, 27,10	-	PWR	+5V

3.4 Electrical Specifications

Maximum Ratings

Storage temperature	-40 to + 125 deg. C
Operating free-air temperature range	0 to +70 deg. C
Supply voltage applied to ground potential	-0.5 to +7.0 V
DC voltage applied to outputs for high output state	-0.5 to Vdd max.
DC input voltage	+4.75V to +5.25 V
Supply current	200mA typ / 300 max.

3.4.1 Electrical Characteristics

The following condition applies unless otherwise specified:

T(A) = 0 + 70 deg. Vdd = 5.0 V ± 5%

DC Characteristics Over Operating Temperature

Symbol	Parameter		Condition	Min.	Typ.	Max.	Unit
V_{IH}	High level input voltage						V
		TTL		2.0			
		CMOS SCHMITT trigger		4.0			
V_{IL}	Low level input voltage						V
		TTL				0.8	
		CMOS level SCHMITT trigger				1.0	
I_{IH}	High level input current		$V_{IN} = V_{DD}$	-10		10	µA
	Input buffer with pull-down			10		200	
I_{IL}	Low level input current		$V_{IN} = V_{SS}$	-10		10	µA
	Input buffer with pull-up			-200		-10	
V_{OH}	High level output voltage						V
	BUFFER	B2	I_{OH} = -2 mA	2.4			
		B4	I_{OH} = -4 mA				
		B8	I_{OH} = -8 mA				
			I_{OL} = 1 µA	V_{DD} -0.05			
V_{OL}	Low level output voltage						
	BUFFER	B2	I_{OL} = 2 mA	2.4			
		B4	I_{OL} = 4 mA				
		B8	I_{OL} = 8 mA				
			I_{OL} = 1 µA			0.05	
I_{OZ}	High impedance leakage current		$V_{OUT} = V_{DD}$ or V_{SS}	-10		10	µA
	Output buffer with pull-up			-200		-10	
	Output buffer with pull-down			10		200	
VH	SCHMITT trigger hysteresis voltage						V
	CMOS				0.6		

4. ET4000/W32p Timing Specifications

Timing is specified with signals driving the following load capacitance:

AA<9:0>, AB<9:0>	64pf
CAS<7:0>*	28pf
RASA*, RASB*	80pf
MWA*, MWB*	80pf
MD<31:0>	28pf
OEOB*, OEOA*, OEEB*, OEEA*	48pf
AP<15:0>	15pf
PCLK	12pf
All others	7pf

4.1 Local Bus Timing

NO.	DESCRIPTION	MIN(ns)	MAX(ns)
C1	BCLK min. high pulse width	5.3	—
C2	BCLK min. low pulse width	3.6	—
C3	BCLK cycle time	30.0	—
C4	ADS* setup time to BCLK high	3.4	—
C5	ADS* hold time from BCLK high	3.0	—
C6	BHE*,SEG<2:0>,BE3* setup time to BCLK high	1.2	—
C7	BHE*,SEG<2:0>,BE3* hold time from BCLK high	6.0	—
C8	WR*,MIO* setup time to BCLK high	4.9	—
C9	WR*,MIO* hold time to BCLK high	3.0	—
C10	A<23:01,BLE*,BE<1:0>* setup time to BCLK high	0.61	—
C11	A<23:01,BLE*,BE<1:0>* hold time from BCLK high	7.0	—
C12	AD<31:0> input delay	—	1.5s+.3
C13	AD<31:0> setup time to BCLK high	-1.7	—
C14	AD<31:0> hold time from BCLK high	4.0	—
C15	AD<31:0> delay from BCLK high	—	20.8
C16	AD<31:0> float delay from BCLK high	—	5.5
C17	ADRE* high delay from BCLK high	—	9.0
C18	ADRE* low delay from BCLK high	—	9.0
C19	RDMX* low delay from BCLK high	—	9.8
C20	RDMX* high delay from BCLK high	—	8.2
C21	DIR delay from BCLK high	—	20.0
C22	DIR delay from BCLK high	—	5.2
C23	BS16* low delay from SEG,A<23:0>	—	15.0
C24	BS16* high delay from BCLK high	—	15.0
C25	READY* low delay from BCLK high	—	12.0
C26	READY* high delay from BCLK high	—	8.5
C27	LOCAL* low delay from SEG,A<23:0>	—	16.31
C28	LOCAL* high delay from BCLK high	—	14.2
C29	RDYRTN* setup time to BCLK high	11.0	—
C30	RDYRTN* hold time from BCLK high	7.1	—
C31	ROME* low delay from BCLK high	—	15.4
C32	ROME* high delay from BCLK high	—	15.2
C33	IXCM* low delay from BCLK high	—	15.4
C34	IXCM* high delay from BCLK high	—	15.2
C35	READY* tri-state delay from BCLK high	—	8.5

s = SCLK period

4.2 PCI Bus Interface Timing

NO.	DESCRIPTION	MIN(ns)	MAX(ns)
N1	FRAME* setup to BCLK high	5.1	—
N2	FRAME* hold from BCLK high	3.7	—
N3	Adress setup to BCLK high	4.9	—
N4	Address hold from BCLK high	1.6	—
N5	Read data active delay from BCLK high	—	25.0
N6	Read data tristate delay from BCLK high	4.1	—
N7			
N8	Command setup to BCLK high	2.0	—
N9	Command hold from BCLK high	0.8	—
N10	BE<3:0>* setup to BCLK high	1.3	—
N11	BE<3:0>* hold from BCLK high	2.0	—
N12	IRDY* setup to BCLK high	3.0	—
N13	IRDY* hold from BCLK high	0.3	—
N14	DEVSEL* low delay from BCLK high	—	14.4
N15	DEVSEL* high delay from BCLK high	3.8	12.1
N16	DEVSEL* tristate delay from BCLK high	3.5	11.5
N17	TRDY* low delay from BCLK high	—	15.4
N18	TRDY* high delay from BCLK high	4.0	12.7
N19	TRDY* tristate delay from BCLK high	3.6	11.0
N20	STOP* low delay from BCLK high	—	19.3
N21	STOP* high delay from BCLK high	3.6	11.8
N22	STOP* tristate delay from BCLK high	3.9	12.7
N23	PAR active delay from BCLK high	—	10.4
N24	PAR tristate delay from BCLK high	3.1	10.4

4.3 Video Bus Interface

NO.	DESCRIPTION	MIN(ns)	MAX(ns)
D1	MCLK high pulse width (PCLK high= 6.0 ns)	2.5	—
D2	MCLK low pulse width (PCLK low = 6.0 ns)	5.5	—
D3	MCLK cycle time	10.0	—
D4	PCLK high delay from MCLK	—	4.5
D5	AP<7:0> delay time from PCLK	3.5	8.1
D6	MBS* delay time from PCLK	3.6	8.3
D7	HS,VS delay time from PCLK	4.3	15.8
D8	AP<15:8> delay time from PCLK		
D9	SP<1:0> delay time from PCLK	1.4	5.8
D10	BDE delay time from PCLK	3.5	9.1
D11	PCLK high pulse width		

4.4 Display Memory Timing - Non-interleaved

NO.	DESCRIPTION	MIN(ns)	MAX(ns)
E1	AA,AB address setup time to RAS[5]	(3*SCLK)-(Shigh)-0.66	(3*SCLK)-(Shigh)-0.05
	AA,AB address setup time to RAS[7]	(3*SCLK)-Shigh+0.88	(3*SCLK)-(Shigh)+2.55
	AA,AB address setup time to RAS[6]	(3*SCLK)-(Shigh)+1.17	(3*SCLK)-(Shigh)+3.68
E2	AA,AB address hold time to RAS[5]	(SCLK)+(Shigh)-3.27	(SCLK)+(Shigh)-1.31
	AA,AB address hold time to RAS[7]	(SCLK)+(Shigh)-6.48	(SCLK)+(Shigh)-2.24
	AA,AB address hold time to RAS[6]	(SCLK)+(Shigh)-7.61	(SCLK)+(Shigh)-2.53
E3	AA,AB address setup time to CAS[5]	(SCLK)-(Shigh)-6.00	(SCLK)-(Shigh)-1.52
	AA,AB address setup time to CAS[8]	(SCLK)-(Shigh)-2.79	(SCLK)-(Shigh)-0.59
	AA,AB address setup time to CAS[9]	0.45	1.40
	AA,AB address setup time to CAS[6]	0.73	2.53
E4	AA,AB address hold time to CAS[5]	(Shigh)+0.30	(Shigh)+1.68
	AA,AB address hold time to CAS[8]	(Shigh)-1.53	(Shigh)-0.73
	AA,AB address hold time to CAS[9]	(SCLK)-5.80	(SCLK)-1.61
	AA,AB address hold time to CAS[6]	(SCLK)-6.80	(SCLK)-1.90
E5	RAS low to CAS low delay[1,2,3]	(3*SCLK)-5.42	(3*SCLK)-1.47
	RAS low to CAS low delay[4] *Except Revisions A & B*	(2*SCLK)+(Shigh)-1.19	(2*SCLK)+(Shigh)-0.46
E6	*NA*		
E7	CAS low pulse width[1,2,3]	SCLK-2.88	SCLK-0.78
	CAS low pulse width[4] *Except Revisions A & B*	SCLK+3.88	SCLK+9.21
E8	CAS high pulse width[1,2,3]	SCLK+0.78	SCLK+2.88
	CAS high pulse width[4] *Except Revisions A & B*	SCLK-9.21	SCLK-3.88
E9	*NA*		
E10	MWA*,MWB* setup time to RAS	0.51	1.91
E11	CAS low to MWA*,MWB* high[1,2,3]	SCLK-1.34	SCLK-0.38
	CAS low to MWA*,MWB* high[4] *Except Revisions A & B*	(2*SCLK)-(Shigh)-5.58	(2*SCLK)-(Shigh)-1.60
E12	MD out setup time to CAS[1,2,3]	(SCLK)-(Shigh)-8.52	(SCLK)-(Shigh)-2.35
	MD out setup time to CAS[4] *Except Revisions A & B*	1.42	5.18
E13	MD out hold time[1,2,3]	(Shigh)+1.60	(Shigh)+5.98
	MD out hold time[4] *Except Revisions A & B*	(SCLK)-7.67	(SCLK)-2.31
E14	MD in setup time[1]	3.583	—
	MD in setup time[2]	2.162	—
	MD in setup time *Except Revisions A & B*	2.13	6.88
E15	MD in hold time[1]	6.374	—
	MD in hold time[2]	11.204	—
	MD in hold time *Except Revisions A & B*	-2.21	-0.81
E16	RAS high pulse width[10]	(n*SCLK)+1.38	(n*SCLK)+4.82
E17	CAS cycle time	(2*SCLK)	(2*SCLK)
E18	OE low pulse width *Except Revisions A & B*	SCLK-3.18	SCLK-0.85
E19	OE high pulse width *Except Revisions A & B*	SCLK+ 0.85	SCLK+3.18
E20	CAS low to OE low delay[3] *Except Revisions A & B*	0.61	2.31
	CAS low to OE low delay[4] *Except Revisions A & B*	(SCLK)-(Shigh)-1.93	(SCLK)-(Shigh)-0.40
E21	CAS high to OE high delay *Except Revisions A & B*	0.48	2.01

[1] = CRTC 37<2> = 1 *Revisions A & B*
[2] = CRTC 37<2> = 0 *Revisions A & B*
[3] = CRTC 37<2> = 1 *Except Revisions A & B*
[4] = CRTC 37<2> = 0 *Except Revisions A & B*
[5] = CRTC 37<2> = 1 and CRTC 34<6> = 1 *Except Revisions A & B*
[6] = CRTC 37<2> = 0 and CRTC 34<6> = 0 *Except Revisions A & B*
[7] = CRTC 34<6> = 1 *Except Revisions A & B*
[8] = CRTC 37<2> = 1 and CRTC 34<6> = 0 *Except Revisions A & B*
[9] = CRTC 37<2> = 0 and CRTC 34<6> = 1 *Except Revisions A & B*
[10] = n determined from CRTC 37<4:3>

SCLK = SCLK period
Shigh = SCLK high pulse width

4.5 Clock Interface (Reset Initialize and Clock Timing)

NO.	DESCRIPTION	MIN(ns)	MAX(ns)
F1	REST high pulse width	2c+.2	—
F2	SCLK high pulse width	7.2	—
F3	SCLK low pulse width	8.5	—
F4	SCLK cycle time	20.0	—
F5	BCLK hold time from REST low	5.5	—
F6	PORI setup time to REST low	-0.9	—
F7	PORI hold time from REST low	5.4	—

c = greatest of SCLK, MCLK, or BCLK period

4.6 ROM BIOS Read Timing

NO.	DESCRIPTION	MIN(ns)	MAX(ns)
M1	ROME* active delay from BCLK high	2xBCLK+2.3	2xBCLK+8.0
M2	ROME* mininum pulse width (Rev C)	7xBCLK-2.6	—
	(Rev A,B)	4xBCLK-2.6	—
M3	ADDL delay from BCLK high	—	7.0
M4	ADDL mininum pulse width	20.1	—
M5	DD port address out setup to ADDL low	1xBCLK+2.0	—
M6	DD port address out hold from ADDL low	2.0	
M7	DD port address out setup to ROME* low	1xBCLK-1.1	—
M8	DD port address out hold from ROME* low	2.4	—
M9	RS<1:0> setup to ROME* low	3xBCLK-5.0	—
M10	RS<1:0> hold from rome* high	3xBCLK+2.0	—
M11	ROME* cycle delay (WORD or DWORD)	6xBCLK+0.3	6xBCLK+1.2
M12	ROME data setup to BCLK high	15.0	—
M13	ROME data hold to BCLK high	0.0	—

4.7 Image Port Timing

NO.	DESCRIPTION	MIN(ns)	MAX(ns)
H1	IXFS/IXLS/IXOF min. pulse width	s+5.0	—
H2	IXWQ* low min. pulse width	5.0	—
H3	IXWQ* high min. pulse width	5.0	—
H4	IDMK/IM<7:0> setup time to IXWQ*	2.0	—
H5	IDMK/IM<7:0> hold time to IXWQ*	2.0	—

s= SCLK period

4.8 Display Memory Timing - Interleaved

NO.	DESCRIPTION	MIN(ns)	MAX(ns)
J1	AA,AB address setup time to RAS[5]	(3*SCLK)-(Shigh)-0.66	(3*SCLK)-(Shigh)-0.05
	AA,AB address setup time to RAS[7]	(3*SCLK)-Shigh+0.88	(3*SCLK)-(Shigh)+2.55
	AA,AB address setup time to RAS[8]	(3*SCLK)-(Shigh)+1.17	(3*SCLK)-(Shigh)+3.68
J2	AA,AB address hold time to RAS[5]	(SCLK)+(Shigh)-3.27	(SCLK)+(Shigh)-1.31
	AA,AB address hold time to RAS[7]	(SCLK)+(Shigh)-6.48	(SCLK)+(Shigh)-2.24
	AA,AB address hold time to RAS[6]	(SCLK)+(Shigh)-7.61	(SCLK)+(Shigh)-2.53
J3	AA,AB address setup time to CAS[5]	(SCLK)-(Shigh)-6.00	(SCLK)-(Shigh)-1.52
	AA,AB address setup time to CAS[8]	(SCLK)-(Shigh)-2.79	(SCLK)-(Shigh)-0.59
	AA,AB address setup time to CAS[9]	0.45	1.40
	AA,AB address setup time to CAS[6]	0.73	2.53
J4	AA,AB address hold time to CAS[5]	(Shigh)+0.30	(Shigh)+1.68
	AA,AB address hold time to CAS[8]	(Shigh)-1.53	(Shigh)-0.73
	AA,AB address hold time to CAS[9]	(SCLK)-5.80	(SCLK)-1.61
	AA,AB address hold time to CAS[6]	(SCLK)-6.80	(SCLK)-1.90
J5	RAS low to CAS low delay[1,2,3]	(3*SCLK)-5.42	(3*SCLK)-1.47
	RAS low to CAS low delay[4] *Except Revisions A & B*	(2*SCLK)+(Shigh)-1.19	(2*SCLK)+(Shigh)-0.46
J6	*NA*		
J7	CAS low pulse width[1,2,3]	SCLK-2.88	SCLK-0.78
	CAS low pulse width[4] *Except Revisions A & B*	SCLK+3.88	SCLK+9.21
J8	CAS high pulse width[1,2,3]	SCLK+0.78	SCLK+2.88
	CAS high pulse width[4] *Except Revisions A & B*	SCLK-9.21	SCLK-3.88
J9	RAS to second CAS delay[1,2,3]	(4*SCLK)-6.04	(4*SCLK)-1.76
	RAS to second CAS delay[4] *Except Revisions A & B*	(3*SCLK)+(Shigh)-1.65	(3*SCLK)+(Shigh)-0.61
J10	MWA*,MWB* setup time to RAS	0.51	1.91
J11	CAS low to MWA*,MWB* high[1,2,3]	SCLK-1.34	SCLK-0.38
	CAS low to MWA*,MWB* high[4] *Except Revisions A & B*	(2*SCLK)-(Shigh)-5.58	(2*SCLK)-(Shigh)-1.60
J12	MD out setup time to CAS[1,2,3]	(SCLK)-(Shigh)-8.52	(SCLK)-(Shigh)-2.35
	MD out setup time to CAS[4] *Except Revisions A & B*	1.42	5.18
J13	MD out hold time[1,2,3]	(Shigh)+1.60	(Shigh)+5.98
	MD out hold time[4] *Except Revisions A & B*	(SCLK)-7.67	(SCLK)-2.31
J14	MD in setup time[1]	3.583	—
	MD in setup time[2]	2.162	—
	MD in setup time	2.13	6.88
J15	MD in hold time[1]	6.374	—
	MD in hold time[2]	11.204	—
	MD in hold time *Except Revisions A & B*	-2.21	-0.81
J16	RAS high pulse width[10]	(n*SCLK)+1.38	(n*SCLK)+4.82
J17	CAS cycle time	(2*SCLK)	(2*SCLK)
J18	OE low pulse width *Except Revisions A & B*	SCLK-3.18	SCLK-0.85
J19	OE high pulse width *Except Revisions A & B*	SCLK+ 0.85	SCLK+3.18
J20	CAS low to OE low delay[3] *Except Revisions A & B*	0.61	2.31
	CAS low to OE low delay[4] *Except Revisions A & B*	(SCLK)-(Shigh)-1.93	(SCLK)-(Shigh)-0.40
J21	CAS high to OE high delay *Except Revisions A & B*	0.48	2.01

[1] = CRTC 37<2> = 1 *Revisions A & B*
[2] = CRTC 37<2> = 0 *Revisions A & B*
[3] = CRTC 37<2> = 1 *Except Revisions A & B*
[4] = CRTC 37<2> = 0 *Except Revisions A & B*
[5] = CRTC 37<2> = 1 and CRTC 34<6> = 1 *Except Revisions A & B*
[6] = CRTC 37<2> = 0 and CRTC 34<6> = 0 *Except Revisions A & B*
[7] = CRTC 34<6> = 1 *Except Revisions A & B*
[8] = CRTC 37<2> = 1 and CRTC 34<6> = 0 *Except Revisions A & B*
[9] = CRTC 37<2> = 0 and CRTC 34<6> = 1 *Except Revisions A & B*
[10] = n determined from CRTC 37<4:3>

SCLK = SCLK period
Shigh = SCLK high pulse width

4.9 DAC READ/WRITE Timing

NO.	DESCRIPTION	MIN(ns)	MAX(ns)
K1	PMEW* active delay from BCLK high	3xBCLK+4.5	3xBCLK+15.6
	PMER* active delay from BCLK high	2xBCLK+4.5	2xBCLK+15.6
K2	PMEW* min low pulse width	11xBCLK-1.0	—
K3	PMEW* cycle delay (WORD or DWORD)	7xBCLK+0.3	—
K4	DD port write data setup to PMEW* low	1xBCLK-0.5	—
K5	DD port write data hold from PMEW* high	1xBCLK+0.0	—
K6	PMER* min low pulse width	15xBCLK-0.5	—
K7	DD port data setup to BCLK high	15.0	—
K8	DD port data hold to BCLK high	0.0	
K9	RS<1:0> setup to PMEW/R* low	1xBCLK-0.5	—
K10	RS<1:0> hold from PMEW/R* high	4.0	—

4.10 Extended Memory-Mapped Register Access

NO.	DESCRIPTION	MIN(ns)	MAX(ns)
X1	DD<7:0> hold from IMAD falling (address phase)	0.37	2.0
X2	DD<7:0> setup to IMAD falling (address phase)	11.4	BCLK-6
X3	DD<7:0> hold from IMAD rising (data phase)	0.37	2.0
X4	DD<7:0> setup to IMAD rising (data phase)	11.4	BCLK-6
X5	RS<0> setup to IMAD falling	BCK-.3	BCLK-1.8
X6	RS<0> hold from IXMAD falling	0.80	2.0
X7	Read data setup to BCLK	10.0	—
X8	Read data hold from BCLK	5.0	—
X9	IXMAD low delay from BCLK	—	8.0
X10	IXMAD high delay from BCLK	—	8.0

4.11 Timing Diagrams

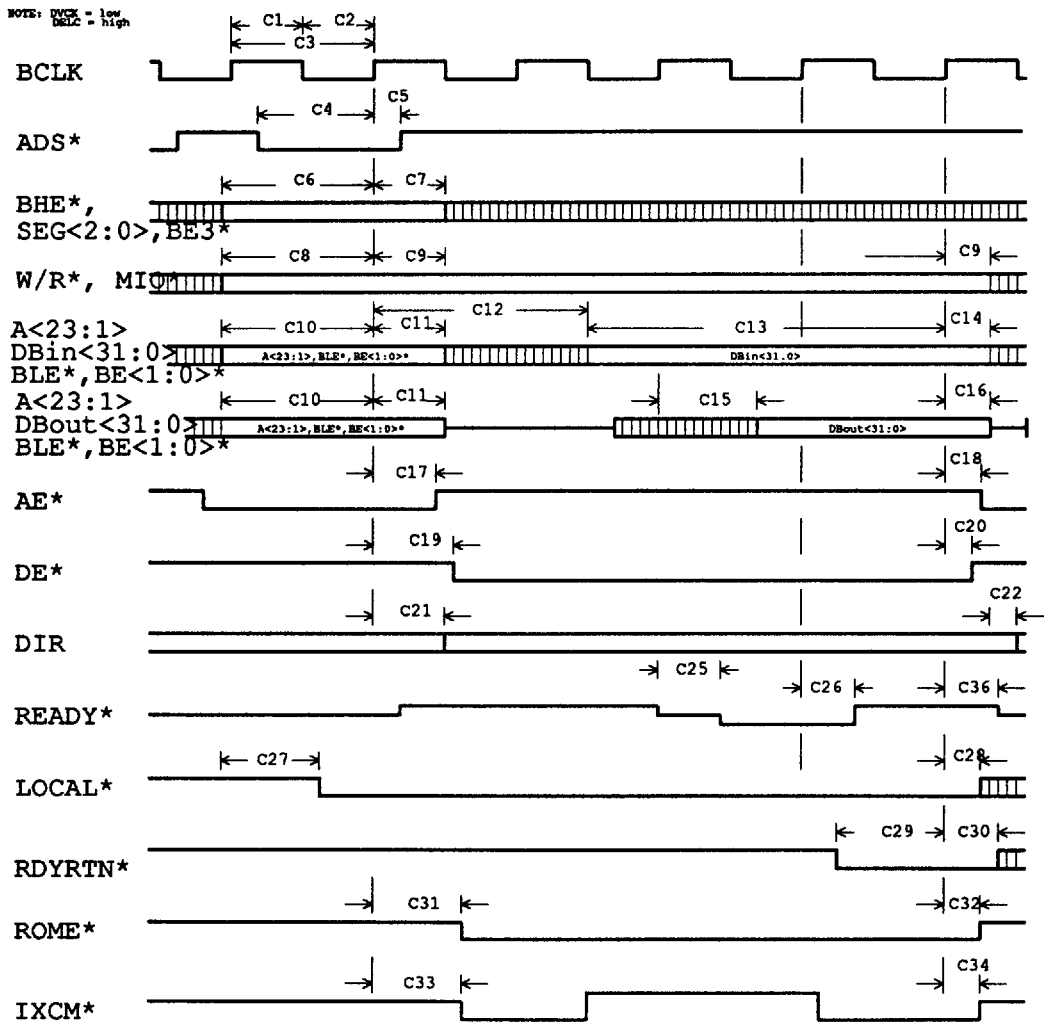


Figure 4.11-1 Local Bus Read/Write Timing

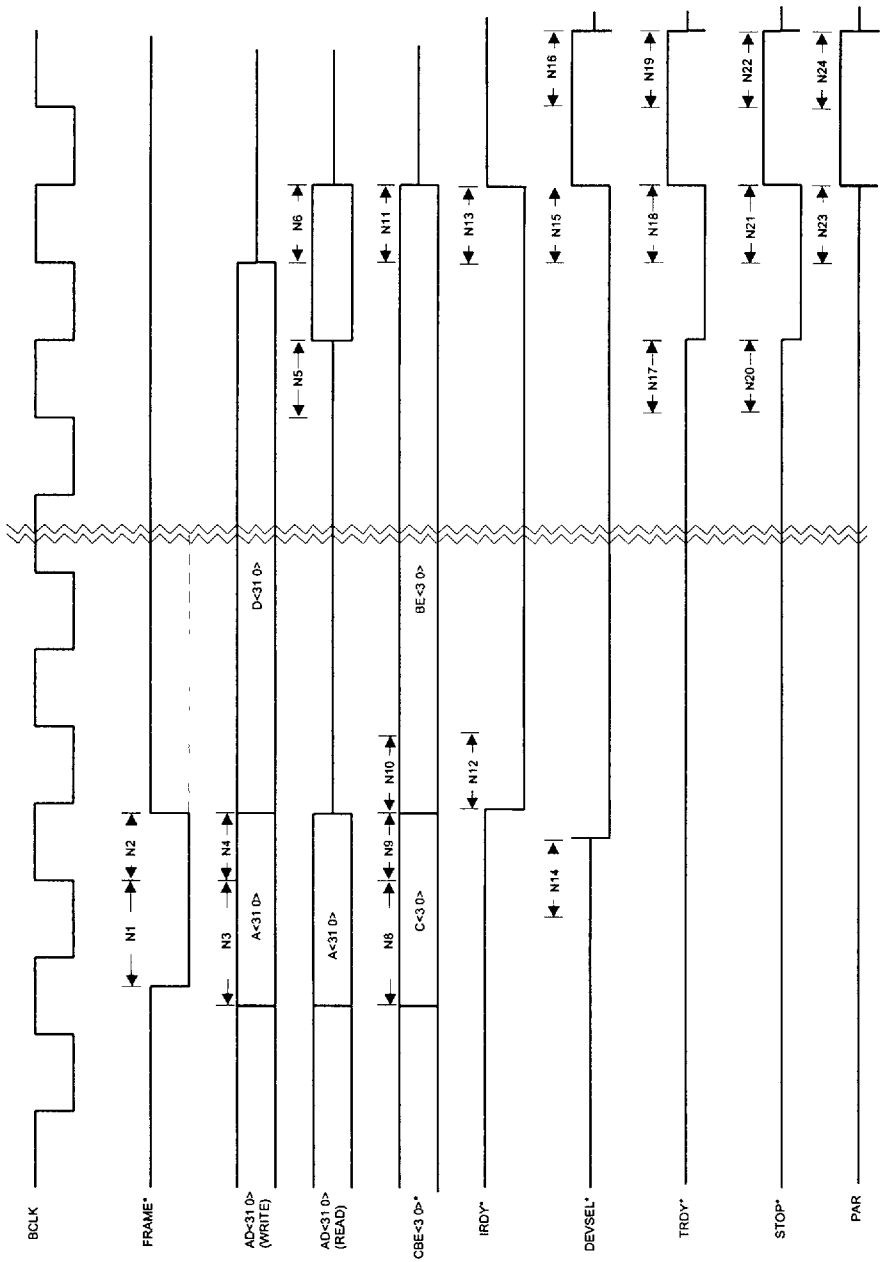


Figure 4.11-2 PCI Bus Timing

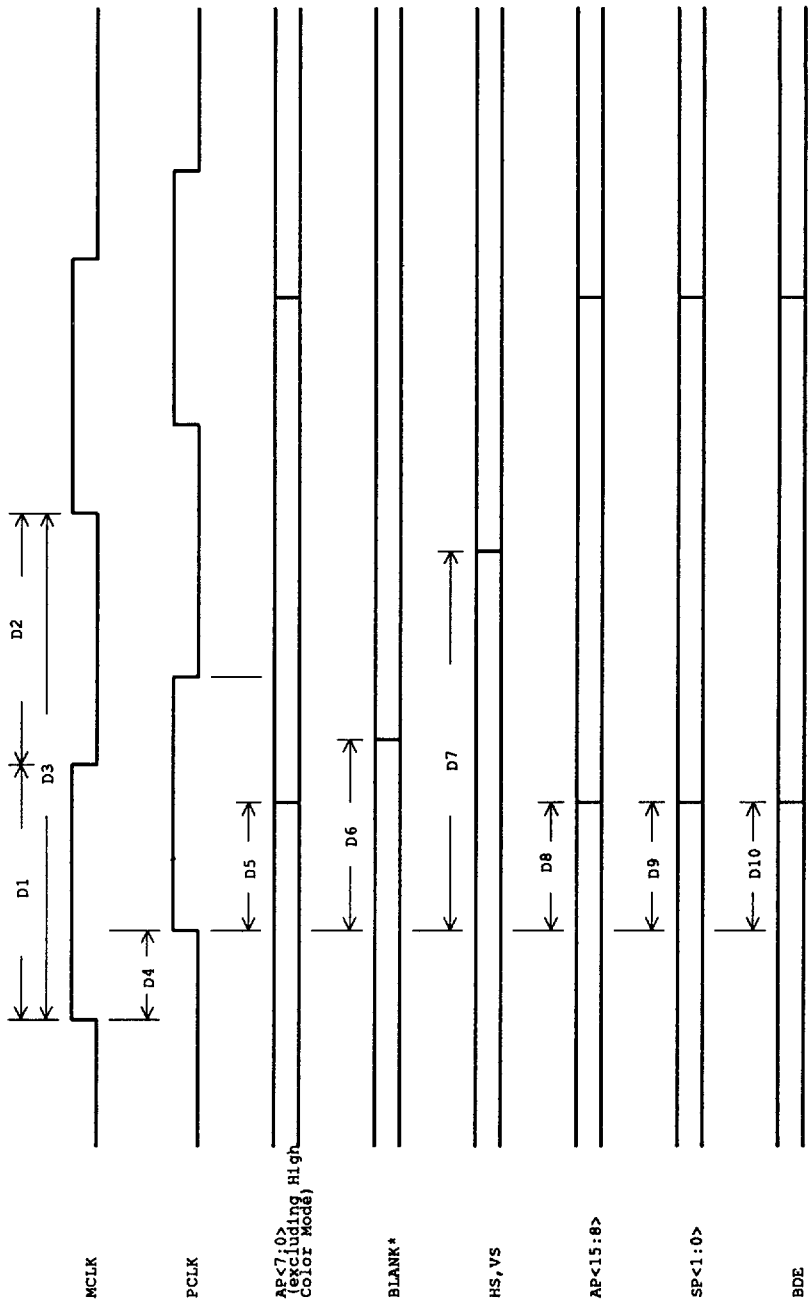


Figure 4.11-3 Video Bus Interface Timing

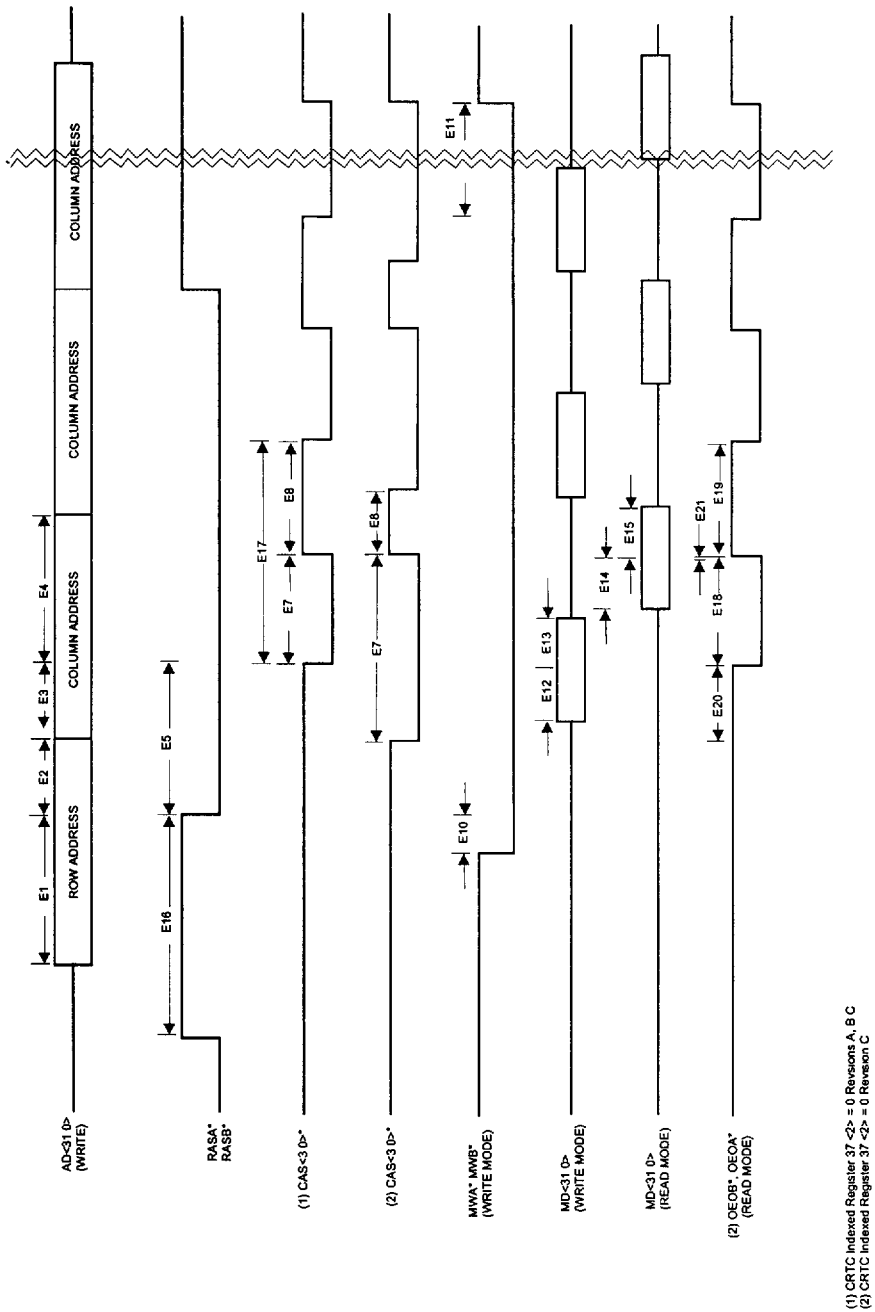


Figure 4.11-4 Display Memory Read/Write Timing - Non-interleaved

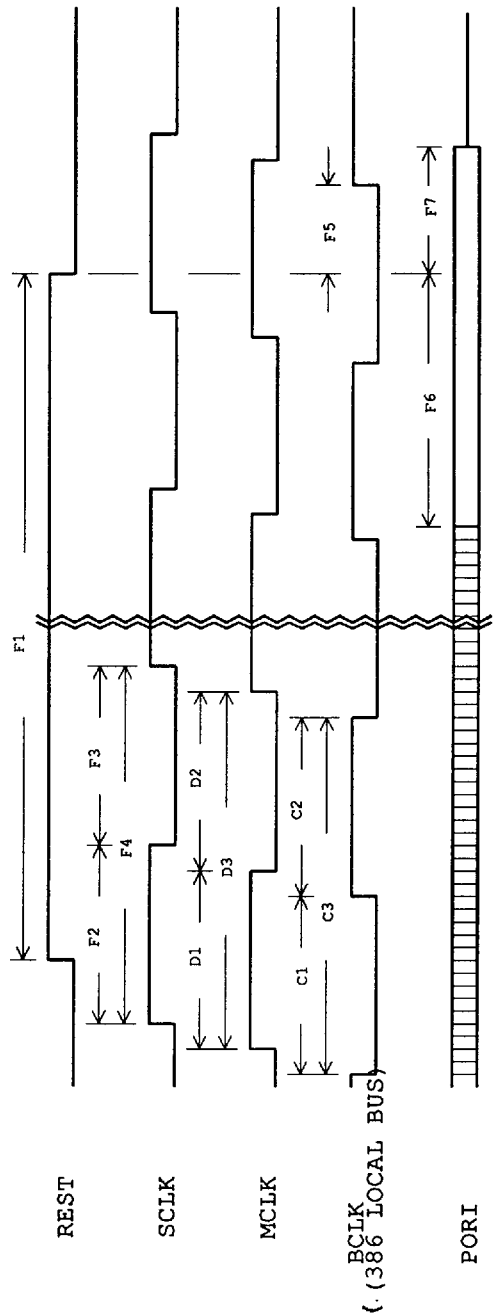


Figure 4.11-5 Reset Initialize and Clock Timing

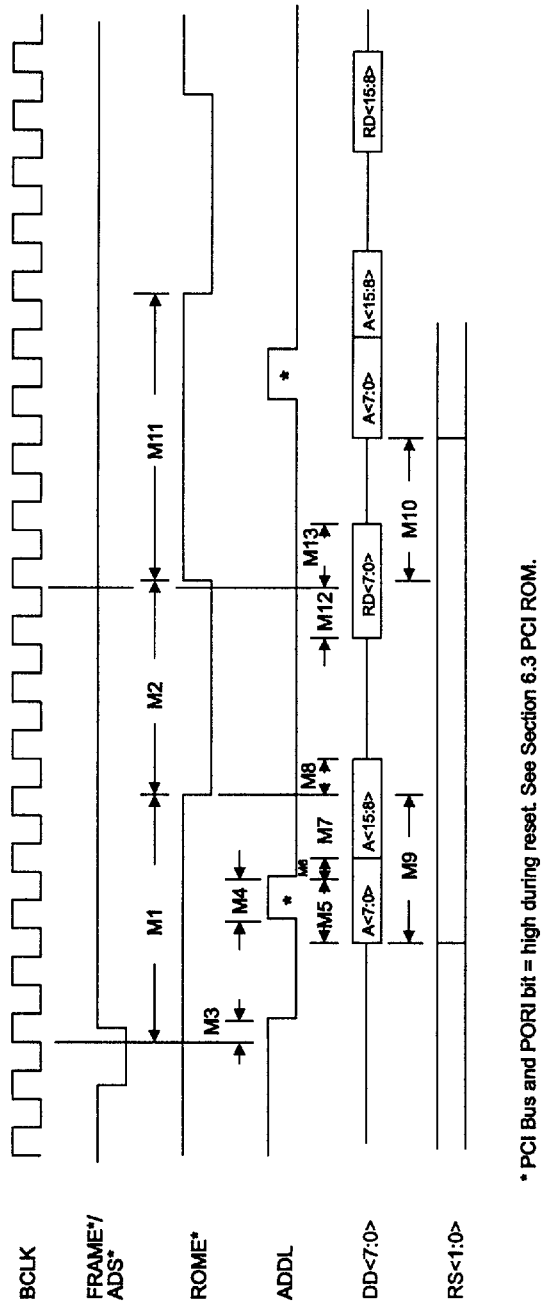


Figure 4.11-6 ROM BIOS Timing

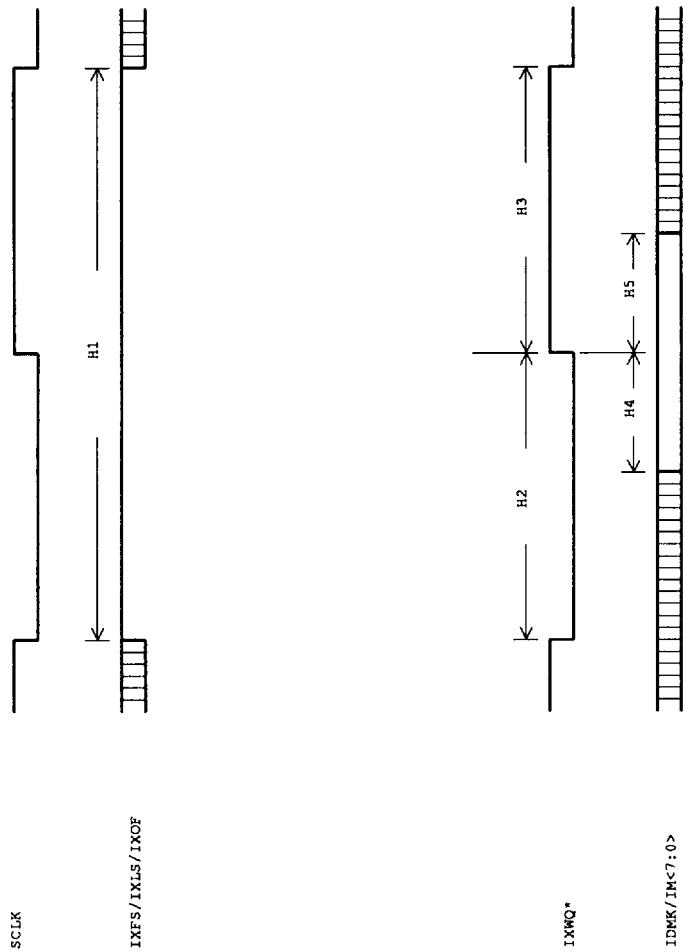


Figure 4.11-7 Image Port Command and Write Timing

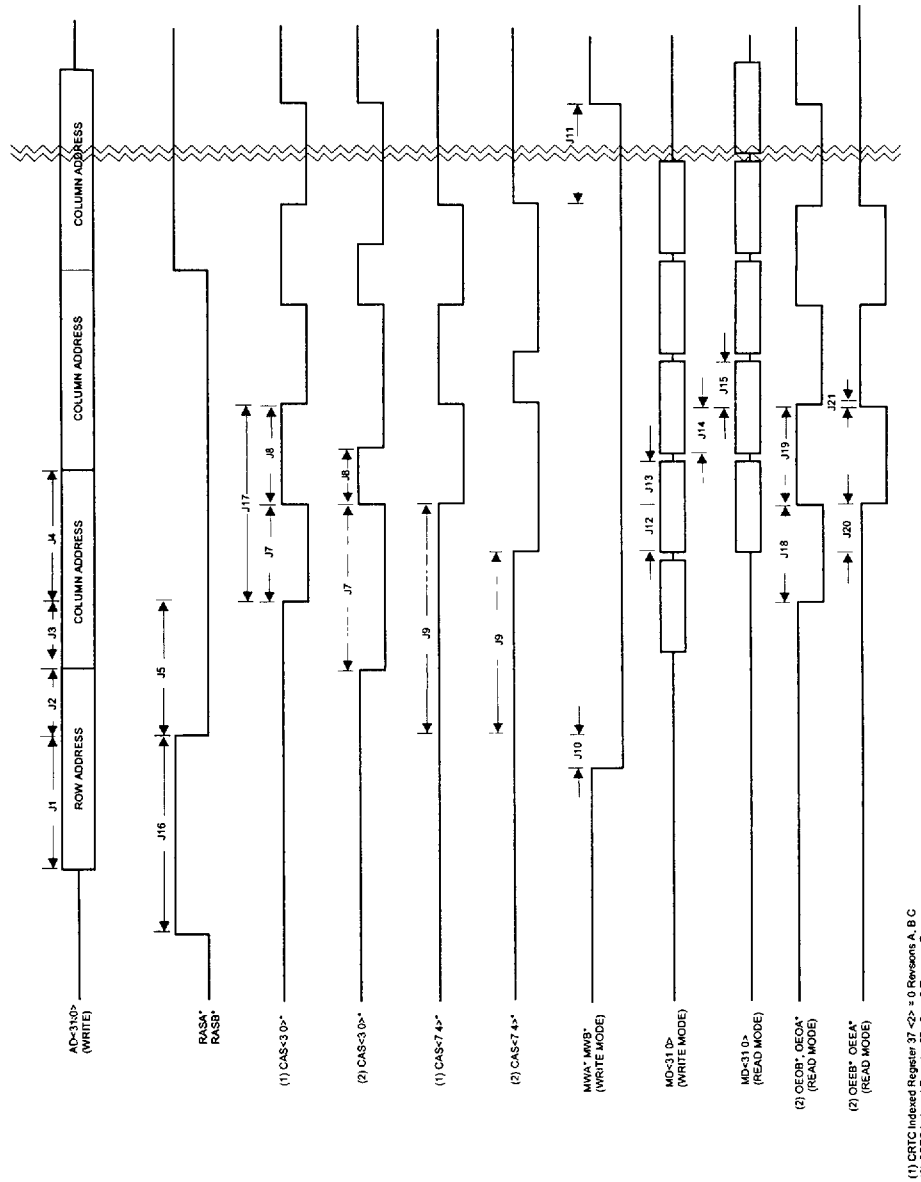


Figure 4.11-8 Display Memory Read/Write Timing - Interleaved

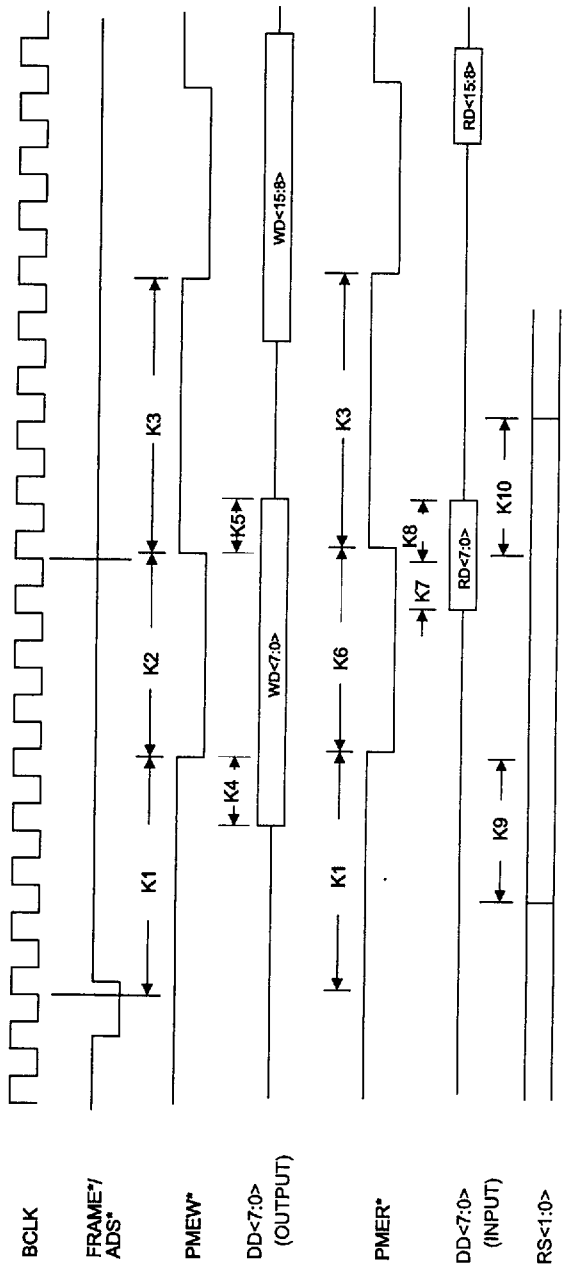


Figure 4.11-9 DAC Read/Write Timing

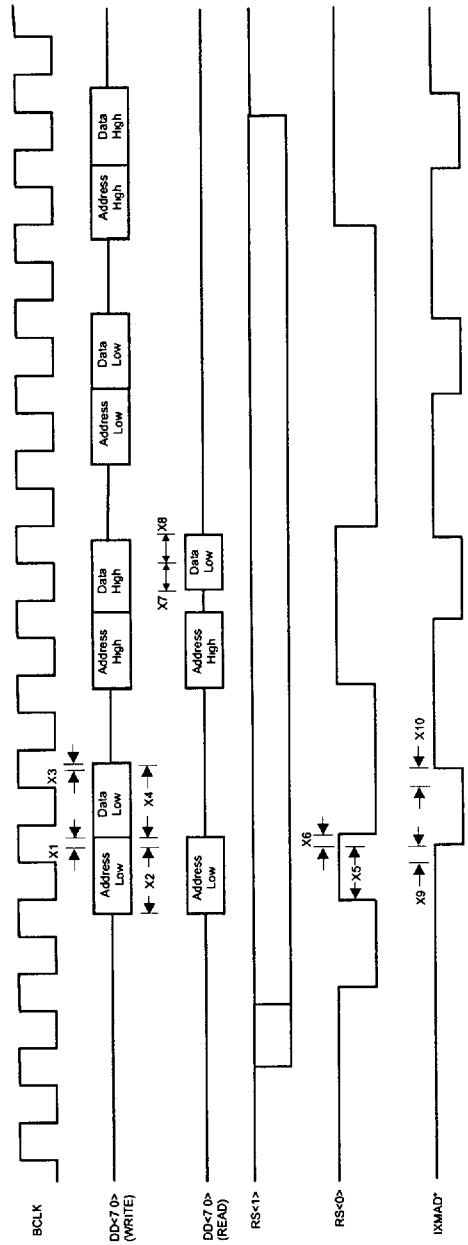


Figure 4.11-10 Extended Memory-Mapped Register Access

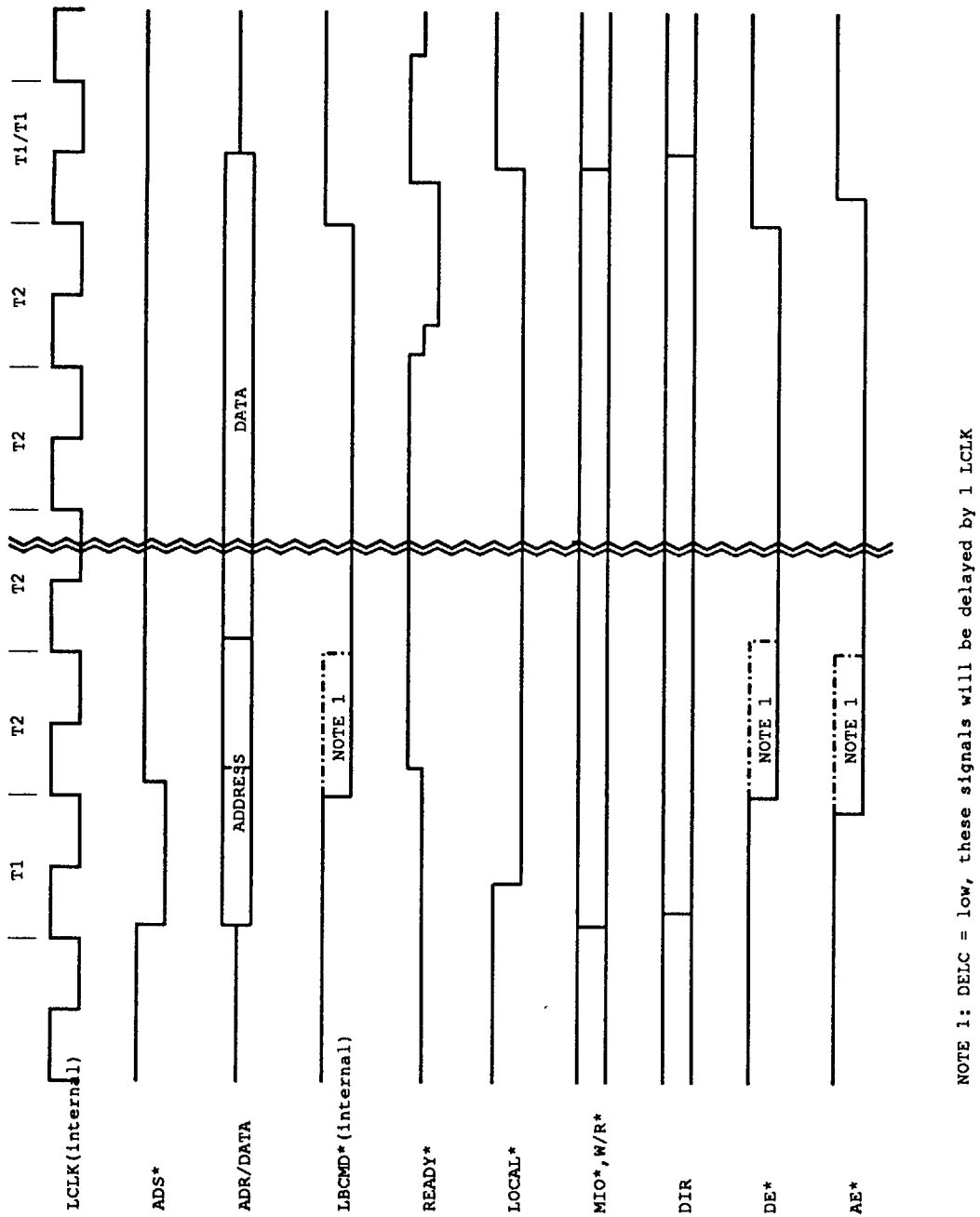


Figure 4.12-1 Normal Local Bus Cycle

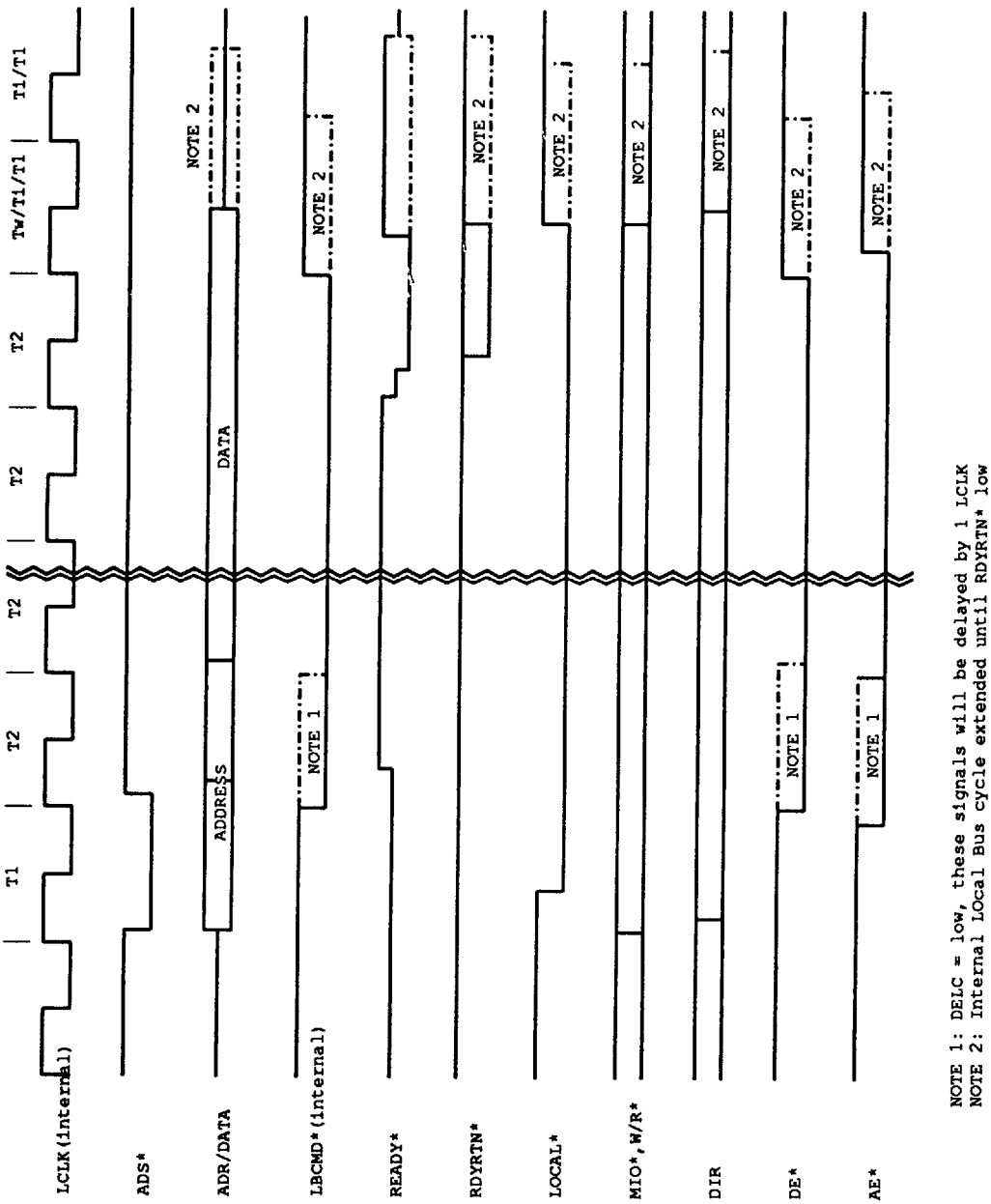


Figure 4.12-2 Local Bus Cycle Ready Return

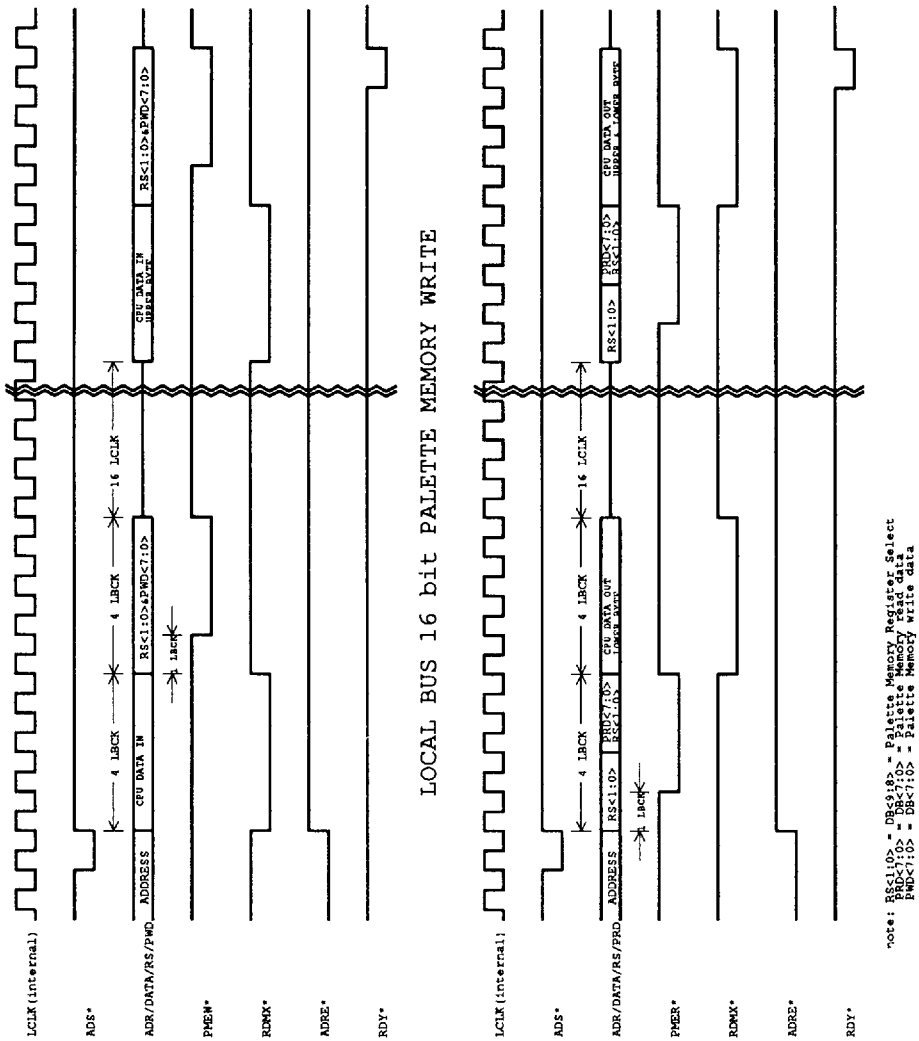


Figure 4.12-3 Local Bus 16-bit Palette Memory Write/Read

5. ET4000/W32p Register Descriptions

Table 5.0-1 ET4000/W32p Registers, R/W Operation, Port Addresses, Size

Register	R/W	I/O	
General	**Operation**	**Port Address**	**Bits**
Misc. Output	W	x3C2	<7:0>
	R	x3CC	<7:0>
Input Status 0	R	x3C2	<7:0>
Input Status 1	R	x3#A	<7:0>
Feature Control	W	x3#A	<7:0>
	R	x3CA	<7:0>
Video Subsystem Enable	RW	x3C3/46E8	<7:0>
Display Mode Control	RW	x3#8	<7:0>
External Palette RAM			
Pixel Mask	RW	x3C6	<7:0>
Pixel Write Address	RW	x3C8	<7:0>
Pixel Color Value	RW	x3C9	<7:0>
Pixel Read Address	W	x3C7	<7:0>
DAC State	R	x3C7	<7:0>
Attribute Controller (ATC)			
Address/Index	W	x3C0 (Index)	<7:0>
	R	x3C0	<7:0>
Indexed registers	W	x3C0 (Data)	<7:0>
	R	x3C1	<7:0>
Primary CRT Controller (CRTC)			
Address/Index	RW	x3#4	<7:0>
Indexed registers	RW	x3#5	<7:0>
Timing Sequencer (TS)			
Address/Index	RW	x3C4	<7:0>
Indexed registers	RW	x3C5	<7:0>
Graphics Display Controller (GDC)			
Segment Select 1	RW	x3CD	<7:0>
Segment Select 2	RW	x3CB	<7:0>
Address/Index	RW	x3CE	<7:0>
Indexed registers	RW	x3CF	<7:0>
Secondary CRT Controller (CRTCB/Sprite)			
Address/Index	RW	21xA (Index)	<7:0>
Indexed registers	RW	21xB (Data)	<7:0>
Image Port (IMA)			
Address/Index	RW	21xA (Index)	<7:0>
Indexed registers	RW	21xB (Data)	<7:0>

# = B in monochrome emulation modes; D in color emulation modes, controlled by bit 0 in the Miscellaneous Output Register.
x = IOD<2:0>. See Section 3.3.1, IOD<2:0>; 6.2.1 Local Bus PORI; and following table.

IOD<2:0> also defines VGA I/O address mapping as follows:

IOD<2:0>	CRTCB	CRTC	GDC	ATC	TS	Ext. Palette RAM
000	210A/B	73D4/5	73CE/F	73C0	73C4/5	73C6-73C9
001	211A/B	63D4/5	63CE/F	63C0	63C4/5	63C6-63C9
010	212A/B	53D4/5	53CE/F	53C0	53C4/5	53C6-53C9
011	213A/B	43D4/5	43CE/F	43C0	43C4/5	43C6-43C9
100	214A/B	33D4/5	33CE/F	33C0	33C4/5	33C6-33C9
101	215A/B	23D4/5	23CE/F	23C0	23C4/5	23C6-23C9
110	216A/B	13D4/5	13CE/F	13C0	13C4/5	13C6-13C9
111	217A/B	03D4/5	03CE/F	03C0	03C4/5	03C6-03C9

Table 5.0-2 ET4000/W32p Mapped Registers, R/W Operation, Size

See Section 7.3 for the memory base address for the MMU and ACL registers. The offset in the table below is added to the base address to calculate the actual address of the register.

Register	R/W Operation*	Memory Offset	Bits
Memory Management Unit (MMU)			
MMU Base Pointer 0	RW	00	<21:0>
MMU Base Pointer 1	RW	04	<21:0>
MMU Base Pointer 2 ***Revision A ONLY***	RW	08	<21:0>
MMU Control Register	RW	13	<7:0>
Graphics Accelerator (ACL)			
Suspend/Terminate	RW	30	<7:0>
Operation State	WO	31	<7:0>
Sync Enable	RW	32	<7:0>
Interrupt Mask	RW	34	<7:0>
Interrupt Status	RW	35	<7:0>
Accelerator Status	RW	36	<7:0>
X Position	RW	38	<11:0>
Y Position	RW	3A	<11:0>
Pattern Address	RW	80	<21:0>
Source Address	RW	84	<21:0>
Pattern Y Offset	RW	88	<11:0>
Source Y Offset	RW	8A	<11:0>
Destination Y Offset	RW	8C	<11:0>
Pixel Depth	RW	8E	<7:0>
Direction	RW	8F	<7:0>
Pattern Wrap	RW	90	<7:0>
Source Wrap	RW	92	<7:0>
X Count	RW	98	<11:0>
Y Count	RW	9A	<11:0>
Routing Control	RW	9C	<7:0>
Background Raster Operation	RW	9E	<7:0>
Foreground Raster Operation	RW	9F	<7:0>
Destination Address	RW	A0	<21:0>
Mix Address	RW	A4	<24:0>
Mix Y Offset	RW	A8	<11:0>
Error Term	RW	AA	<15:0>
Delta Minor	RW	AC	<11:0>
Delta Major	RW	AE	<11:0>

* See Section 2.12 on reading and writing accelerator registers.

5.1 General Registers

The ET4000/W32p has five General Registers, each with its own port address allowing direct programming access, and requiring no pairing of index and data registers. The Input Status #1 and Feature Control registers have separate addresses for monochrome and color modes.

5.1.1 Miscellaneous Output Register

I/O address = 3CC read; 3C2 write

Bit	Description	Access
7	Vertical Retrace Polarity.	RW
6	Horizontal Retrace Polarity.	RW
5	Page Select for Odd/Even.	RW
4	Reserved.	
3	Clock Select 1.	RW
2	Clock Select 0.	RW
1	Enable RAM.	RW
0	I/O Address Select.	RW

Hardware resets return all bits to zero.

Bit **Description**

Bit 7 When set to 1, selects negative vertical retrace.

When set to 0, selects positive vertical retrace. The relationship between vertical screen size and polarities is as follows:

Vsync polarity	Hsync polarity	Vertical size
+	+	768 lines
+	-	400 lines
-	+	350 lines
-	-	480 lines

Bit 6 When set to 1, selects negative horizontal retrace polarity.

When set to 0, selects positive horizontal retrace polarity.

Bit 5 Selects between two 64K pages of memory when in the Odd/Even display modes (0,1,2,3,7).

When set to 1, it is the default for operation of the HiRes text mode.

When set to 0, selects the high page of memory.

5.1.1 Miscellaneous Output Register (cont'd)

Bit	Description
Bits 3:2	Used to select the clock rate according to the following table: Bits 3 2 0 0- Selects MCLK clock 1 0 1- Selects MCLK clock 2 1 0- Selects MCLK clock 3 1 1- Selects MCLK clock 4

See Section 5.2.29, CRTC Indexed Register 31 for more information regarding clock selects.

Bits <3:2> of the MISCOUT register (CS<1:0>) can be translated to provide compatibility between the EGA mode and the EGA monitor when the external clock select circuit is connected as follows:

CS1	CS0	Clock Frequency
1	1	—
1	0	32.514MHz
0	1	28.322MHz
0	0	25.175MHz

Bit	Description
Bit 1	When set to 1, enables access to display memory. When set to 0, disables display memory access from the host.
Bit 0	When set to 1, sets CRTC addresses to 3DX and Input Status Register 1's address to 3DA for Color/Graphics Monitor Adapter emulation. When set to 0, sets CRTC addresses to 3BX and Input Status Register 1's address to 3BA for monochrome emulation.

5.1.2 Input Status Register Zero

I/O address = 3C2

Bit	Description	Access
7	CRT Interrupt.	RO
6	Feature code 1.	RO
5	Feature code 0.	RO
4	Switch Sense.	RO
3	Reserved.	
2	Reserved.	
1	Reserved.	
0	Reserved.	

NOTE: the "KEY" must be set in order to read bits 5 and 6.

To set the KEY:

- Write 03 to Hercules Compatibility Register (3BF);
- Set bits 7 and 5 of the Mode Control Register (3#8) to 1,1, e.g. A0 (other bits are don't care).

Example:

```
mov     dx,3BFh
mov     al,3
out     dx,al
mov     dx,3D8h  ;3B8h in mono mode
mov     al,0A0h
out     dx,al
```

To turn OFF the KEY:

- Set 3D8 (or 3B8) bits 7 and 5 to a value not equal to 1,1

Example: set 3D8 (or 3B8) = 29h.
Also 3BF should be set = 1 to restore to normal.

Bit	Description
Bit 7	A value of 1 indicates a pending vertical retrace interrupt. A value of 0 means that the vertical retrace interrupt has been cleared.
Bit 6:5	Inputs can be used to determine the type of monitor connected to the system. Input status is from external feature input. NOTE: The external feature input bits 6 & 5 are DD<1:0> bus status at the last REST low to high transition. If the DD<1:0> are not "pull-down" by a 1K resistor, then a "11" status will be the default value.
Bit 4	Input can be used to determine the default video mode upon power-up, or the type of monitor connected to the system. The Clock Select field setting (bits 2,3 in the Miscellaneous Output Register) determines the switch to read.

5.1.3 Input Status Register One

I/O address = 3BA (mono)/3DA (color)

Bit	Description	Access
7	Vertical retrace complement.	RO
6	CRTCB vertical display enable.	RO
5:4	Video display feedback test.	RO
3	Vertical retrace.	RO
2	CRTCB display enable.	RO
1	CRTC Horizontal Display Enable.	RO
0	Display enable complement.	RO

Bit Description

Bit 7 A value of 1 indicates that video data is currently being displayed.

A value of 0 indicates the vertical blanking or vertical border time. (See Figure 2.1-1)

Bit 6 A value of 1 indicates that the CRTCB window is active within the current scan line.

Bits 5:4 Used for diagnostic purposes. They are selectively connected to two of the eight color outputs of the Attribute Controller. The Color Plane Enable register (ATC Indexed Register 12) controls the multiplexer for the video wiring. Available combinations are:

Color Plane Register Bits		Input Status Register One Bits	
5	4	5	4
0	0	P2	AP0
0	1	P5	AP4
1	0	P3	AP1
1	1	P7	AP6

Bit 3 A value of 0 indicates that video data is currently being displayed.

A value of 1 indicates a vertical retrace interval during the vertical sync pulse.

Bit 2 A value of 1 indicates that the CRTCB window is active.

Bit 1 A value of 1 indicates CRTC horizontal display enabled.

Bit 0 A value of 1 indicates a vertical or horizontal retrace interval and is the real-time status of the inverted display enable signal.

5.1.4 Feature Control Register

I/O address = 3CA read; 3BA/3DA write

Bit	Description	Access
7:4	Reserved.	
3:2	Monitor ID.	RO
1	Feat (1).	RW
0	Feat (0).	RW

NOTE: The "KEY" must be set in order to read bits 5:2, and 7. See Section 5.1.2, Input Status Register Zero for definition of "KEY".

Bit	Description
Bits 3:2	Used to read back the MONID<3:2> pins for monitor identification. See also Section 5.1.2, Input Status Register Zero, bits <6:5> for monitor ID.
Bits 1:0	General purpose read/write bits. In previous designs these bits were output to the Feature Connector.

5.1.5 Video Subsystem Enable Register

I/O address = 3C3/46E8

The Video Subsystem Enable Register is accessible via one of two locations (03C3 or 046E8), selected by bit 3 of CRTC Indexed Register 34. If the Video Subsystem Enable Register is at 03C3, then bit 0 is the "Enable Video Subsystem" bit. The power-up default has this register at 03C3.

When the video subsystem is disabled, the chip does not respond to any host read/writes, except to the Video Subsystem Enable Register.

Bit	Description	Access
7:4	Reserved (=0).	
3	Enable video subsystem (address 46E8).	RW
2:1	Reserved.	
0	Enable video subsystem (address 03C3).	RW

Bit	Description
Bit 3	When set to 1, enables the video subsystem when the port address is configured for 46E8.
Bit 0	When set to 1, enables the video subsystem when the port address is configured for address 3C3.

5.1.6 Display Mode Control Register

I/O address = 3B8 (monochrome); 3D8 (color)

Bit	Description	Access
7	Key bit.	RW
6	Enable second page.	RO
5	Key bit.	RW
4:0	Reserved.	

Bit	Description
Bits 7, 5	Bits used to set the KEY (See Input Status Register Zero for instructions to set KEY).
Bit 6	Enable second page (bit 1 of the Hercules Compatibility Register (see Section 5.1.7).

5.1.7 Hercules Compatibility Register

I/O address = 3BF

Bit	Description	Access
7:2	Reserved.	
1	Enable second page.	WO
0	Reserved.	

Bit	Description
Bit 1	When set to 1 when in monochrome mode with the display memory set to start at B0000, enables the second page of display memory, starting at B8000, providing 64KB of display memory. This bit must be set in order to set the KEY. This bit can be read from bit 6 of the Display Mode Control Register (3#8).

5.2 CRTC Register Description

The CPU interface to the ET4000/W32p internal primary CRT Controller (CRTC) consists of 33 read/write registers. Of these registers, one Register, the CRTC Index Register, is accessed by a separate independent I/O address (3#4, where # = B in monochrome emulation modes; D in color emulation modes, as controlled by bit 0 in the Miscellaneous Output Register.) The remaining 32 registers are internally indexed, which means that they are accessed via a common I/O address (3#5) with one of the 32 registers that is actually accessed selected by the CRTC Index Register. NOTE: The "KEY" must be set in order to write CRTC indices above 18, except indices 33 and 35, (CRTC 35 is protected by bit 7 of CRTC 11). See Section 5.1.2, Input Status Register Zero for definition of "KEY".

All values are in hexadecimal unless otherwise noted.

Table 5.2-1 CRTC Index Register

Register Name		Address
CRTC Index Register	(Read/Write)	3#4

Table 5.2-2 CRTC Indexed Registers

CRTC Indexed Register Name	CRTC Indexed Address		Port Address
Horizontal Total	0	(Read/Write)	3#5
Horizontal Display End	1	(Read/Write)	3#5
Horizontal Blank Start	2	(Read/Write)	3#5
Horizontal Blank End	3	(Read/Write)	3#5
Horizontal Sync Start	4	(Read/Write)	3#5
Horizontal Sync End	5	(Read/Write)	3#5
Vertical Total	6	(Read/Write)	3#5
Overflow Low	7	(Read/Write)	3#5
Initial Row Addr (Raster Counter)	8	(Read/Write)	3#5
Maximum Row Address	9	(Read/Write)	3#5
Cursor Start Row Address	A	(Read/Write)	3#5
Cursor End Row Address	B	(Read/Write)	3#5
Linear Starting Address Middle	C	(Read/Write)	3#5
Linear Starting Address Low	D	(Read/Write)	3#5
Cursor Address Middle	E	(Read/Write)	3#5
Cursor Address Low	F	(Read/Write)	3#5
Vertical Sync Start	10	(Read/Write)	3#5
Vertical Sync End	11	(Read/Write)	3#5
Vertical Display End	12	(Read/Write)	3#5
Row Offset	13	(Read/Write)	3#5
Underline Row Address	14	(Read/Write)	3#5
Vertical Blank Start	15	(Read/Write)	3#5
Vertical Blank End	16	(Read/Write)	3#5
CRTC Mode	17	(Read/Write)	3#5
Split Scr Start Low (Line Compare)	18	(Read/Write)	3#5
System Segment Map Comparator	30	(Read/Write)	3#5
General Purpose	31	(Read/Write)	3#5
RAS/CAS Configuration	32	(Read/Write)	3#5
Extended Start Address	33	(Read/Write)	3#5
Overflow High	35	(Read/Write)	3#5
Video System configuration 1	36	(Read/Write)	3#5
Video System configuration 2	37	(Read/Write)	3#5
Horizontal Overflow	3F	(Read/Write)	3#5

Many of the CRTC values, such as the Linear Starting Address and the Vertical Sync Start, are broken up into numerous non-adjacent registers. This is because of the need to maintain IBM VGA and EGA compatibility. For example, Vertical Sync Start bits 7:0 are in Register 10 hex, Vertical Sync Start bits 9:8 are in Register 7, Overflow Low. These two registers provide the 10-bit vertical sync start value in IBM's VGA. The ET4000/W32p chip supports 11-bit vertical values, so Register 35 hex, Overflow High, contains bit 10 of the vertical sync start value. Although this can sometimes be awkward, it is the only way to provide both IBM VGA and EGA compatibility and the extended functionality of the ET4000/W32p chip.

Because there are so many ET4000/W32p registers and because many CRTC values are spread over numerous registers, the following table lists many of the registers arranged according to general function.

Table 5.2-3 CRTC Registers By Function

Primary Function	Sub Function	CRTC Index	Indexed Register Name
Horizontal timings	Scan line length	0	Horizontal Total (bit 7:0)
		3F	Horizontal Overflow (bit 8)
	Display enable	1	Horizontal Display End (bit 7:0)
		3	Horizontal Blank End (bit 6:5) (Horizontal Display Enable Skew)
	Blanking	2	Horizontal Blank Start (bit 7:0)
		3F	Horizontal Overflow (bit 8)
		3	Horizontal Blank End (HBE bit 0:4)
		5	Horizontal Sync End (HBE bit 5)
	Sync	4	Horizontal Sync Start (bit 7:0)
		3F	Horizontal Overflow (bit 8)
		5	Horizontal Sync End (bit 4:0)
Vertical timings	Frame height	6	Vertical Total (bit 7:0)
		7	Overflow Low (VT bit 8,9)
		35	Overflow High (VT bit 10)
	Display enable	12	Vertical Display End (bit 7:0)
		7	Overflow Low (VDE bit 8,9)
		35	Overflow High (VDE bit 10)
	Blanking	15	Vertical Blank Start (bit 7:0)
		7	Overflow Low (VBS bit 8)
		9	Maximum Row Address (VBS bit 9)
		35	Overflow High (VBS bit 10)
		16	Vertical Blank End (bit 7:0)
	Sync	10	Vertical Sync Start (bit 7:0)
		7	Overflow Low (VSS bit 8,9)
		35	Overflow High (VSS bit 10)
		11	Vertical Sync End (bit 3:0)

Table 5.2-3 CRTC Registers By Function (continued)

Primary Function	Sub Function	CRTC Index	Indexed Register Name
Cursor	Address	F	Cursor Address Low (bit 7:0)
		E	Cursor Address Middle (bit 15:8)
		33	Extended Start Address (CURA bit 19:16)
	Row Address	A	Cursor Start Row Address (bit 4:0)
		B	Cursor Stop Row Address (bit 4:0)
	Skew	B	Cursor Stop Row Address (bit 6,5)
Memory address	Linear address	D	Linear Start Addr Low (bit 7:0)
		C	Linear Start Addr Middle (bit 15:8)
		33	Extended Start Address (LA bit 19:16)
	Row offset	13	Row Offset (bit 7:0)
		3F	Row Offset (bit 8)
Split screen	Start scan line	9	Maximum Row Addr (Split Scr bit 9)
		18	Line Compare (bit 7:0)
		7	Overflow Low (Line Compare bit 8)
		35	Overflow High (Line Compare bit 10)

For the rest of the descriptions in this section: # = B in monochrome emulation modes; D in color emulation modes, controlled by bit 0 in the Miscellaneous Output Register.

5.2.1 CRTC Index

I/O address = 3#4

Bit	Description	Access
7:6	Reserved.	
5:0	Current CRTC index.	RW

Bit **Description**

Bits 5:0 These bits provide the index of the currently selected internally indexed register. The CRTC Index register determines which CRTC indexed register will be accessed when a read/write is performed using port address 3#5.

5.2.2 CRTC Indexed Registers

The following registers are CRTC indexed registers. These registers are accessed by first writing the index of the desired register to the CRTC Index register and then accessing the register using the address 3#5.

5.2.3 CRTC Indexed Register 0: Horizontal Total

I/O address = 3#5 (write-protectable—see Section 5.2.20)

Bit	Description	Access
7:0	Total character times per horizontal scan line (-5 VGA, -2 for EGA mode).	RW

Bit	Description
Bits 7:0	The Horizontal Total register defines the horizontal scan line time by controlling the length of the scan line in character times units. The character time unit is defined by TS Indexed Register 1, bit 0.

5.2.4 CRTC Indexed Register 1: Horizontal Display End

I/O address = 3#5 (write-protectable—see Section 5.2.20)

Bit	Description	Access
7:0	Character count of horizontal display enable end -1.	RW

Bit	Description
Bits 7:0	The Horizontal Display End register contains the 8-bit value of the internal horizontal character counter after which the horizontal display enable period is to end. The total number of characters displayed per horizontal scan line is one greater than the contents of the Horizontal Display End register.

5.2.5 CRTC Indexed Register 2: Horizontal Blank Start

I/O address = 3#5 (write-protectable—see Section 5.2.20)

Bit	Description	Access
7:0	Character count of horizontal blanking start.	RW

Bit	Description
Bits 7:0	The Horizontal Blank Start register contains the 8-bit value of the internal horizontal character counter at which horizontal blanking is to start.

5.2.6 CRTC Indexed Register 3: Horizontal Blank End

I/O address = 3#5 (write-protectable—see Section 5.2.20)

Bit	Description	Access
7	Test bit.	RW
6:5	Display enable skew.	RW
4:0	Character count of horizontal blanking end modulo 32 (EGA); 5 least significant bits of character count of horizontal blanking end modulo 64 (VGA mode).	RW

Bit Description

Bit 7 When set to 1, sets normal mode of operation.

Bits 6:5 These bits form a 2-bit integer that defines the skew of the horizontal display enable in character clocks as follows:

Bit 6	Bit 5	Skew
0	0	0 character clocks.
0	1	1 character clock.
1	0	2 character clocks.
1	1	3 character clocks.

Bits 4:0 EGA mode: Provides the 5-bit value of the internal horizontal character counter at which horizontal blanking is to end. Since the character counter is an 8-bit counter and the Horizontal Blank End is a 5-bit register, the upper 3 bits of the character counter are ignored in making this comparison. This means that the horizontal blanking end position is defined relative to the horizontal blanking start position; the first time after the start of horizontal blanking that the Horizontal Blank End register matches the lower 5 bits of the character counter, horizontal blanking will end.

VGA mode: The Horizontal Blank End register value is increased to six bits; the five bits will provide the least significant five bits of this value, while the most significant bit is found in CRTC Indexed Register 5 (Horizontal Sync End register) bit 7.

5.2.7 CRTC Indexed Register 4: Horizontal Sync Start

I/O address = 3#5 (write-protectable—see Section 5.2.20)

Bit	Description	Access
7:0	Character count of horizontal sync start.	RW

Bit Description

Bits 7:0 The Horizontal Sync Start register contains the 8-bit value of the internal horizontal character counter at which horizontal sync (the horizontal retrace pulse) is to start.

5.2.8 CRTC Indexed Register 5: Horizontal Sync End

I/O address = 3#5 (write-protectable—see Section 5.2.20)

Bit	Description	Access
7	Bit 5 of Horizontal Blank End for VGA modes.	RW
6:5	Horizontal sync skew.	RW
4:0	Character count of horizontal sync end modulo 32.	RW

Bit **Description**

Bit 7 Provides bit 5 of the Horizontal Blank End value for VGA modes.

Bits 6:5 These bits form a 2-bit integer that defines the skew of the horizontal sync signal in character clocks as follows:

Bit 6	Bit 5	Skew
0	0	0 character clocks.
0	1	1 character clock.
1	0	2 character clocks.
1	1	3 character clocks.

Bits 4:0 These bits make up the 5-bit value of the internal horizontal character counter at which horizontal sync is to end. Since the character counter is an 8-bit counter and horizontal sync end is a 5-bit value, the upper three bits of the character counter are ignored in making this comparison. This means that the horizontal sync end position is defined relative to the horizontal sync start position; the first time after the start of horizontal sync that the Horizontal Sync End register matches the lower 5 bits of the character counter, horizontal sync will end.

5.2.9 CRTC Indexed Register 6: Vertical Total

I/O address = 3#5 (write-protectable—see Section 5.2.20)

Bit	Description	Access
7:0	VGA mode: Horizontal scan lines per vertical frame -2 (bits 7:0).	RW
	EGA Mode: Horizontal scan lines per vertical frame -1 (bits 7:0).	RW

Bit **Description**

Bits 7:0 The Vertical Total register contains the lower eight bits of the 11-bit vertical total value, which defines the number of horizontal scan lines per vertical frame.

Note that bits 9:8 of the vertical total value are in the Overflow Low register, and bit 10 is in the Overflow High register.

5.2.10 CRTC Indexed Register 7: Overflow Low

I/O address = 3#5 (write-protectable—see Section 5.2.20)

Bit	Description	Access
7	Vertical Sync Start (bit 9).	RW
6	Vertical Display Enable End (bit 9).	RW
5	Vertical Total (bit 9).	RW
4	Line Compare (Split Screen) (bit 8).	RW
3	Vertical Blank Start (bit 8).	RW
2	Vertical Sync Start (bit 8).	RW
1	Vertical Display Enable End (bit 8).	RW
0	Vertical Total (bit 8).	RW

Bit **Description**

Bits 7:0 The Overflow register contains one extra bit for each of five values that cannot fit in a single byte. Bits 9:8 of the Vertical Total, Vertical Display Enable End, and Vertical Sync Start are contained in the Overflow register, as is bit 8 for Vertical Blank Start and Line Compare.

5.2.11 CRTC Indexed Register 8: Preset Row Scan/Initial Row Address

I/O address = 3#5

Bit	Description	Access
7	Reserved.	
6:5	Byte Panning.	RW
4:0	Initial row address after vertical sync.	RW

Bit **Description**

Bits 6:5 Control horizontal byte panning in modes programmed as multiple shift modes.

Bits 4:0 Define the row address of the first scan line following vertical sync.

5.2.12 CRTC Indexed Register 9: Maximum Row Address

I/O address = 3#5

Bit	Description	Access
7	Double Scan Enable: 200-to-400 scan line conversion.	RW
6	Line Compare (Split Screen bit 9).	RW
5	Vertical Blank Start (bit 9) .	RW
4:0	Number of scan lines per character row -1.	RW

Bit **Description**

Bit 7 When set to 1, sets scan lines to 400 from 200. This divides the clock in the row scan counter by 2, effectively doubling the lines displayed by displaying every line twice.

When set to 0, returns the row scan counter clock equal to the horizontal scan rate.

Bit 6 Bit 9 of the Line Compare (Split Screen) register.

Bit 5 Bit 9 of the Vertical Blank register.

Bits 4:0 These bits define the height in scan lines of each character row. It is used to select the desired scan line from the font character being displayed.

5.2.13 CRTC Indexed Register A: Cursor Start Row Address

I/O address = 3#5

Bit	Description	Access
7	Reserved.	
6	Reserved.	
5	Used to turn the cursor off (=1) or on (=0).	RW
4:0	The row address at which the cursor starts being enabled.	RW

Bit **Description**

Bit 5 When set to 1, turns the cursor off.

When set to 0, turns the cursor on.

Bits 4:0 These bits contain the value of the internal row address counter at which the cursor is to begin to be enabled.

5.2.14 CRTC Indexed Register B: Cursor End Row Address

I/O address = 3#5

Bit	Description	Access
7	Reserved.	
6:5	Cursor skew.	RW
4:0	The row address at which the cursor stops being enabled.	RW

Bit Description

Bits 4:0 These bits contain the row address at which the cursor is to stop being enabled. That is, Cursor End Row Address register equals last cursor row address displayed plus 1.

Bits 6:5 These bits form a 2-bit integer that defines the skew of the cursor signal in character clocks as follows:

Bit 6	Bit 5	Skew
0	0	0 character clocks.
0	1	1 character clock.
1	0	2 character clocks.
1	1	3 character clocks.

In general, the cursor location must maintain a relationship with the display enable signal such that a cursor positioned at both the extreme left and extreme right of the screen will always appear.

5.2.15 CRTC Indexed Register C: Linear Starting Address Middle

I/O address = 3#5

Bit	Description	Access
7:0	Linear starting address (<15:8>).	RW

Bit Description

Bits 7:0 This register contains bits 15:8 of the 20-bit linear starting address. The linear starting address is the display memory address at which the regen buffer (the area of memory scanned by the linear counter for video data) begins; the linear counter is set to this value at the start of the vertical frame. The linear starting address can be incremented or decremented to perform horizontal character panning; the ATC's horizontal pixel panning feature can be used for finer horizontal panning. In graphics modes, the linear starting address can be incremented or decremented by the value of the Row Offset register to perform smooth (scan line) vertical scrolling. In text modes, the linear starting address can be used to perform character vertical scrolling; in this case, the Initial Row Address register can be used to adjust, on a scan line basis, to smooth-scroll the text.

Note that bits 19:16 of the linear starting address are in the Extended Start Address register and bits 7:0 are in the Linear Starting Address Low register.

5.2.16 CRTC Indexed Register D: Linear Starting Address Low

I/O address = 3#5

Bit	Description	Access
7:0	Linear starting address (<7:0>).	RW

Bit Description

Bits 7:0 This register contains bits 7:0 of the 20-bit linear starting address. See Section 5.2.15, Linear Starting Address Middle register for details on the linear starting address.

5.2.17 CRTC Indexed Register E: Cursor Address Middle

I/O address = 3#5

Bit	Description	Access
7:0	Cursor start address (<15:8>).	RW

Bit Description

Bits 7:0 This register contains bits 15:8 of the 20-bit cursor address. The cursor address is the display memory address at which the cursor is located in text mode.

Note that bits 7:0 of the cursor address are in the Cursor Address Low register, and bits 19:16 are in the Extended Start Address Register (See Section 5.2.31).

5.2.18 CRTC Indexed Register F: Cursor Address Low

I/O address = 3#5

Bit	Description	Access
7:0	Cursor start address (<7:0>).	RW

Bit Description

Bits 7:0 The Cursor Address Low register contains bits 7:0 of the 20-bit cursor address. See Section 5.2.17, Cursor Address Middle register for details on the cursor address.

5.2.19 CRTC Indexed Register 10: Vertical Sync Start

I/O address = 3#5

Bit	Description	Access
7:0	Scan line at which vertical sync starts.	RW

Bit Description

Bits 7:0 This register contains the lower eight bits of the 11-bit vertical sync start value. The vertical sync start value specifies the value of the internal line counter at which vertical sync (the vertical retrace pulse) is to start.

Note that bits 9:8 of the vertical sync start value are in the Overflow Low register, and bit 10 is in the Overflow High register.

5.2.20 CRTC Indexed Register 11: Vertical Sync End

I/O address = 3#5

Bit	Description	Access
7	Protection bit.	RW
6	Reserved.	
5	Enable vertical interrupt when low.	RW
4	Clear vertical interrupt when low.	RW
3:0	Scan line at which vertical sync ends modulo 16.	RW

Bit Description

Bit 7 When set to 1, prevents CRTC registers 0-7 and 35, from being written to, with the exception of bit 4 of the Overflow register (CRTC Register 7) and bits 4,7 of CRTC Indexed Register 35.

Bit 5 When set to 0 and bit <4> is set to 1, enables the vertical interrupt to occur. If bit 5 is set to 0 and bit <4> is set to 1 and the vertical interrupt is cleared, then IRQ will be asserted at the end opf the last display line.

When set to 1, vertical interrupts cannot occur.

Bit 4 When set to 0, clears the vertical interrupt.

When set to 1 and bit 5 is low and the vertical interrupt is cleared, then output pin IRQ will be asserted at the end of the last display line. The vertical interrupt should be cleared whenever a vertical interrupt occurs, before re-enabling interrupts.

Bits 3:0 These bits contain the 4-bit value of the internal line counter at which the vertical sync signal is to end. Since the line counter is an 11-bit counter and vertical sync end is a 4-bit value, the upper 7 bits of the line counter are ignored in making this comparison. This means that the vertical sync end position is defined relative to the vertical sync start position; the first time after the start of vertical sync that the Vertical Sync End register matches the lower 4 bits of the line counter, vertical sync will end.

5.2.21 CRTC Indexed Register 12: Vertical Display End

I/O address = 3#5

Bit	Description	Access
7:0	Number of last scan line displayed vertically.	RW

Bit Description

Bits 7:0 This register contains the lower eight bits of the 11-bit vertical display end value.

Note that bits 9:8 of the vertical display end value are in the Overflow Low register, while bit 10 is in the Overflow High register.

5.2.22 CRTC Indexed Register 13: Row Offset

I/O address = 3#5

Bit	Description	Access
7:0	Word memory address offset between the start of one displayed row and the next.	RW

Bit Description

Bits 7:0 This register specifies the amount to be added to the internal linear counter when advancing from one screen row to the next. The addition is performed whenever the internal row address counter advances past the maximum row address value, indicating that all the scan lines in the present row have been displayed. The Row Offset register is programmed in terms of CPU-addressed words per scan line, counted as either words or doublewords, depending on whether byte or word mode is in effect. If the CRTC Mode register is set to select byte mode, the Row Offset register is programmed with a word value. For a 640-pixel (80-byte) wide graphics display then, a value of 80/2 = 40 (28 hex) would normally be programmed, where 80 is the number of bytes per scan line. If the CRTC Mode register is set to select word mode, then the Row Offset register is programmed with a doubleword, rather than a word, value. For instance, in 80-column text mode, a value of 160/4=40 (28 hex) would be programmed, because from the CPU-addressing side, each character requires two linear bytes (character code byte and attribute byte), for a total of 160 (A0 hex) bytes per row.

In effect, the Row Offset register defines a virtual screen width, so that the physical screen area could be considered a window onto a virtual screen that has a width defined by the Row Offset register. The horizontal pixel panning feature of the ATC can be used with the linear start address to move horizontally around a virtual screen larger than the actual screen size, and the linear start address and the Initial Row Address register can be used to move vertically.

5.2.23 CRTC Indexed Register 14: Underline Row Address

I/O address = 3#5

Bit	Description	Access
7	Reserved (=0).	
6	Doubleword addressing.	RW
5	Linear address count by 4.	RW
4:0	Row address at which underline signal is to be asserted.	RW

Bit	Description
Bit 6	When set to 1, indicates that memory addresses being used are doubleword addresses.
Bit 5	When set to 1, clocks the memory address counter with the character clock divided by 4, used when doubleword addressing is used. NOTE: When bit 3 of the CRTC Mode Register also equals 1, the linear counter will increment twice per character.
Bits 4:0	These bits contain the value of the row address counter at which the underline is to be enabled. The ATC enables underline attribute decoding and displays the underline whenever the underline attribute is true during that scan line. The underline may be disabled by setting the Underline Row Address register to a value greater than the setting of the Maximum Row Address register. The value set is equal to the scan line number requested minus one.

5.2.24 CRTC Indexed Register 15: Vertical Blank Start

I/O address = 3#5

Bit	Description	Access
7:0	Scan line at which vertical blanking begins -1.	RW

Bit	Description
Bits 7:0	This register contains bits 7:0 of the 11-bit Vertical Blank Start value. The Vertical Blank Start specifies the value of the internal line counter at which vertical blanking is to start minus 1. Note that bit 8 of the Vertical Blank Start value is in the Overflow Low register, and bit 9 of the Vertical Blank Start value is in the Maximum Row Address register, while bit 10 is in the Overflow High register.

5.2.25 CRTC Indexed Register 16: Vertical Blank End

I/O address = 3#5

Bit	Description	Access
7:0	Scan line at which vertical blanking ends.	RW

Bit	Description
Bits 7:0	This register contain the 8-bit value of the internal line counter at which vertical blanking is to end. Since the line counter is an 11-bit counter and the Vertical Blank End is an 8- bit register, the upper three bits of the line counter are ignored in making this comparison. This means that the vertical blanking end position is defined relative to the Vertical Blanking Start position; the first time after the start of vertical blanking that the Vertical Blank End register matches the lower 8 bits (In EGA mode only bits 4:0 are used in the comparison) of the line counter, vertical blanking will end.

5.2.26 CRTC Indexed Register 17: CRTC Mode

I/O address = 3#5

Bit	Description	Access
7	Hold control.	RW
6	Word/byte mode select.	RW
5	Alternate address line +MA00 output.	RW
4	Reserved.	
3	Linear counters count by 2.	RW
2	Line counter count by 2.	RW
1	Alternate address line LA14 output.	RW
0	Alternate address line LA13 output.	RW

Bit	Description
Bit 7	When set to 0, places all horizontal and vertical timing control circuitry into a hold state.
Bit 6	When set to 0, selects word mode. When set to 1, selects byte mode.

5.2.26 CRTC Indexed Register 17: CRTC Mode (cont'd)

Bit	Description
Bit 5	Provides an alternate value for LA00 output during the display enable period; that is, the display memory address line LA00 is multiplexed. In word mode, when this bit is set to 0, the LA00 output line is equal to linear counter bit 13. When this bit is set to 1, the LA00 output line is equal to linear counter bit 15. In byte mode, bit 5 has no effect, and linear counter bit 0 is always multiplexed to LA00. Word mode is typically used in text mode. The reason for selecting this alternate value for LA00 is so that the CRTC display memory mapping matches the CPU display memory mapping. In text mode, even/odd mode (See Section 5.3.7, TS Memory Mode register) is active to allow CPU memory addressing to match the CRTC organization of display memory. In even/odd mode, the CPU A<0> line is used to select between plane 0 and plane 1, with planes 2 and 3 storing the soft character font. The CRTC matches this by shifting the linear address counter up one bit before placing it on the LA(17:00) lines (refer to the discussion of bit 6, word/byte mode select, below), and then the full 16 bits of the character code and attribute for a given character are accessed in parallel to generate the character. Consequently, the linear counter provides no direct value for the LA00 line. The highest useful linear address counter value should be wrapped to LA00, to provide the maximum addressable memory in text modes. When 16KB per plane is installed, bit 5 should be set to 0 to wrap linear address bit 13 to LA00, providing the CRTC with 16KB addressing. When more than 16KB of memory per plane is installed, bit 5 should be set to 1 to wrap linear address bit 15 to LA00, providing the CRTC with 64KB of addressing. Externally, the CPU address line A<14> or A<16> or a page select bit, should correspond to the LA00 line in even/odd mode. In non-even/odd mode, the CPU address line A<0> should correspond to the LA00 line.
Bit 3	When set to 1, causes the linear counter to increment on every other character clock, rather than incrementing on every character clock. When set to 0, the linear counter is incremented on every character clock. This is typically associated with situations where DOTCLK is not divided by two but VLOAD is divided by two and word mode addressing is selected; the linear counting is divided by two to synchronize the linear counters with the ATC video data rate. If VLOAD and DOTCLK are both divided by two, then bit 3 should not be set to 1. NOTE: When this bit equals 1 and bit 5 of the Underline Row Address Register also equals 1, then the linear counter will increment twice per character.
Bit 2	When set to 1, causes the line counter to increment on every other scan line, rather than incrementing on every scan line. This has the effect of doubling all vertical timings without affecting any horizontal timings. When set to 0, the line counter increments with every scan line.
Bit 1	Provides an alternate value for LA14 output during the display enable period; that is, the display memory address line LA14 is multiplexed. When set to 1, linear counter bit 14 or bit 13, in byte or word mode, respectively, is multiplexed to LA14. When set to 0, the LA14 output line is equal to row address bit 1, so that out of each group of four scan lines, scan lines 2 and 3 are addressed 16KB after the corresponding even scan lines 0 and 1.
Bit 0	Provides an alternate value for LA13 output during the display enable period; that is, the display memory address line, LA13, is multiplexed. When set to 1, linear counter bit 13 or bit 12, in byte or word mode, respectively, is multiplexed to LA13.

5.2.27 CRTC Indexed Register 18: Line Compare (Split Screen)

I/O address = 3#5

Bit	Description	Access
7:0	Line Compare.	RW

Bit Description

Bits 7:0 This register contains bits 7:0 of the compare target. The line compare target value specifies the value of the internal line counter at which the internal linear counter is to be reset to 0. This means that at the scan line after the scan line specified by the line compare target value the display will reflect the contents of display memory starting at address 0. This split screen section will continue to the bottom of the screen, and will remain unchanged even if the linear starting address is changed.

Note that bit 8 of the line compare value is contained in the Overflow Low register, bit 9 is in the Maximum Row Address register, while bit 10 is in the Overflow High register.

The following CRTC registers are TLI's extended registers. To write to these register(s) (except indices 33 and 35), the "KEY" must be set. (CRTC Indexed Register 35 is protected by bit 7 of CRTC 11.) See Section 5.1.2, Input Status Register Zero for definition of "KEY."

5.2.28 CRTC Indexed Register 30: Address Map Comparator

I/O address = 3#5

Bit	Description	Access
7:0	Address comparator.	RW

Bit Description

Bits 7:0 The value set in this register is relative to the space in system memory addressing to which the W32p will respond. This value is compared against A<29:22>, which must compare true in order for memory operations to occur.

The power-up condition is 00h. This register is primarily used for linear frame buffer operation. When in non-linear frame buffer mode (VGA modes), this register must contain a value of 00h.

NOTE: ***Except Revisions A & B*** In PCI mode, AD<31:30> are compared to bits <31:30> of the PCI configuration space Base Address Register. See Section 2.1.7.

5.2.29 CRTC Indexed Register 31: General Purpose

I/O address = 3#5

Bit	Description	Access
7	Clock Select 4.	RW
6	Clock Select 3.	RW
5:4	Reserved.	
3:0	General purpose.	RW

Bit **Description**

Bits 6:7 The values in these bits are driven out on the CS<4> and CS<3> pins (see Section 3.3.5). Clock Select bit 2 is in the CRTC Indexed Register 34, bit 6, and Clock Select bits <1:0> are in Miscellaneous Output Register, bits <3:2>. These five clock select lines provide selection of up to 32 different video clock frequencies.

Bits 3:0 These bits are provided to the programmer as a general storage location. An example of its use would be to maintain configuration information about the video system.

5.2.30 CRTC Indexed Register 32: RAS/CAS Configuration (RCCONF; protected by KEY)

I/O address = 3#5

Bit	Description	Access
7	Memory interleave enable.	RW
6:5	RCD<1:0> delay.	RW
4:3	RSP<1:0> ($+1)*SCLK, RAS pre-charge time.	RW
2	Reserved.	
1:0	CSW<1:0> ($+1)*SCLK, CAS low pulse-width.	RW

Bit Description

Bit 7 When set to 1 (memory interleave enabled), the chip operates properly only if CSW<0> = 0, CSP<0> = 0, and CRTC Indexed Register 37 bit 0 = 1. When interleave is enabled, the CAS<7:4> signals are applied to the "odd" bank and CAS<3:0> signals are applied to the "even" bank. The W32p memory controller can perform a memory access every SCLK. This bit is set to 0 at reset.

Bits 6:5 These two bits are used to control row/column delay. They are used as follows:

Bit 6	Bit 5	trcd # clocks
1	1	3
1	0	2
0	1	3
0	0	1

These bits are set to (<6:5> = 0,1) at reset.

Bits 4:3 RSP<1:0>, plus 1, form a programmed value for RASB, RASA pre-charge control (Trsp). The actual pulse width high is equal to the programmed value plus 1 of SCLK clock period. These bits are set to (<4:3> = 0,1) at reset.

Bits 1:0 CSW<1:0>, plus 1 ($1+1), form the programmed value for low CAS pulse width control (Tcas). These bits are set to (<1:0> = 0,0) at reset. The actual pulse width value in SCLK periods is determined by the following table:

CRTC 32

Mode CSW <1:0>	Graphics CAS<7:0> CAS Low Pulse Width	Text CAS<7:6;3:2> CAS Low Pulse Width	Text CAS<5:4;1:0> CAS Low Pulse Width
0 0	1	1	1
0 1	2	2	2
1 0	1	3	1
1 1	1	4	1

NOTE: If bit 7 is set to 1, CSW<0> always = 0.

5.2.31 CRTC Indexed Register 33: Extended Start Address

I/O address = 3#5

Bit	Description	Access
7:4	Cursor address bits (<19:16>).	RW
3:0	Linear start address bits (<19:16>).	RW

Bit	Description
Bits 7:4	These are bits 19:16 of the 20-bit Cursor Address value.
Bits 3:0	These are bits 19:16 of the 20-bit Linear Starting Address value.

5.2.32 CRTC Indexed Register 34: Auxiliary Control Register (protected by KEY)

I/O address = 3#5

Bit	Description	Access
7	Memory write time select	RW
6	Memory address setup time select.	RW
5	Vertical sync disable. ***Reserved - Revisions A & B.***	RW
4	Burst enable (PCI only).	RW
3	ENVS VSE register port address (1=46E8, 0=3C3).	RW
2	TRIS 1=tri-state the ET4000/W32p's output tri-state pins.	RW
1	CS2 MCLK clock select 2*	RW
0	Horizontal sync disable ***Reserved - Revisions A & B.***	RW

NOTE: All bits are set to 0 at reset.

Bit	Description
Bit 7	MWA, MWB setup time select from RAS. When set to 1, selects 0ns setup from RAS. When set to 0, selects ½ SCLK clock period (i.e., 10ns @ SCLK = 50MHz).
Bit 6	AA<9:0>, AB<9:0> setup time select from CAS. When set to 1, selects 4ns setup from CAS. When set to 0, selects ½ SCLK clock period.
Bit 5	When set to 1, vertical sync is disabled [in order to comply with the VESA Display Power Management Support (DPMS) standard.] When set to 0 (default setting), vertical sync is enabled.
Bit 4	When set to 1, W32p will perform PCI memory burst transfers. When set to 0, PCI memory burst transfers are disabled.
Bit 3	When set to 1, will set the Video Subsystem Enable register port address to 46E8; 0 = 3C3.
Bit 2	When set to 1 (Output tri-state control), causes all output pins to go to a tri-state condition. The symbols are as follows; see Section 3.2 for pin numbers. RASB*, RASA*, CAS<7:0>*, MWB*, MWA*, MD<31:0>, AB<9:0>, AA<9:0>, VS, HS, AP<15:0>, PCLK, BLANK*.

continued on next page

5.2.32 CRTC Indexed Register 34: Auxiliary Control Register (con'td)

Bit **Description**

Bit 1 Clock select 2 (CS2), in conjunction with the MISCOUT<3:2> clock select lines (CS1, CS0) and in combination with CRTC Indexed Register 31<7:6>, provides up to 32 video clocks to be selected. See CRTC Indexed Register 31 for more information regarding clock selects.

Bit 0 When set to 1, horizontal sync is disabled [in order to comply with the VESA Display Power Management Support (DPMS) standard.]

When set to 0 (default setting), horizontal sync is enabled.

5.2.33 CRTC Indexed Register 35: Overflow High

I/O address = 3#5

Bit	Description	Access
7	Vertical interlace mode (1=enable).	RW
6	CRTCB or CRTC interrupt select.	RW
5	External sync reset (gen-lock) the line/chr counter (1=enable).	RW
4	Line Compare (Split Screen) Bit 10.	RW
3	Vertical Sync Start Bit 10[1].	RW
2	Vertical Display End Bit 10[1].	RW
1	Vertical Total Bit 10[1].	RW
0	Vertical Blank Start Bit 10[1].	RW

Bit **Description**

Bit 7 When set to 1, will enable the vertical interlace mode where the odd-numbered lines will be displayed, followed by the even-numbered lines, thus doubling the effective vertical resolution with the same vertical timing.

Bit 6 When set to 1, will select the CRTCB or Sprite as the vertical interrupt.

When set to 0, will select the CRTC as the vertical interrupt.

NOTE: To enable/clear vertical interrupt, see Section 5.2.20, CRTC Indexed Register 11, bits 5:4.

Bit 5 When set to 1, will enable the SYNR input to reset the ET4000/W32p's internal line and character counter asynchronously. Also, the TKN<1:0> outputs are redefined as TKN<1>= interlace mode active, TKN<0>=EVEN field. For additional details see Section 3, Pin Descriptions.

NOTE: SYNR is redefined as Image Port Data Byte Mask if IMAE (IMA Indexed Register F7: Image Port Enable), CRTCB/Sprite Enable, bit 0 is set to 1. See Section 3, Pin Descriptions for an explanation of SYNR.

Bits 4:0 These are bit 10 of the Line Compare (Split Screen Start), Vertical Sync Start, Vertical Display End, Vertical Total, and Vertical Blank Start values, respectively.

[1] These bits are write-protectable using the protection bit found in CRTC Indexed Register 11<7>. See Section 5.2.20 for a description of this bit.

5.2.34 CRTC Indexed Register 36: Video System Configuration 1 (VSCONF1) (protected by KEY)

I/O address = 3#5

Bit	Description	Access
7:6	Reserved.	
5	Enable memory mapped registers.	RW
4	Enable system linear map.	RW
3	Enable memory management buffers.	RW
2:0	Refresh count per line.	RW

Bit **Description**

Bit 5 When set to 1 (and bit 3 also set to 1), enables the memory mapped registers (MMU and Accelerator). See Sections 2.11 - 2.12 for more information on memory mapped registers, and Section 7.3 for the effect this bit has on the Video Memory Map.

Bit 4 When set to 1, enables the system linear map, i.e., the video memory is accessed directly as flat CPU addresses in an up to 4 megabyte area of physical memory, rather than via the 64K segments at physical address A0000/B0000.

Bit 3 When set to 1, enables the three memory management buffers. Accessing one of these buffers will indirectly access the video memory at an offset determined by the corresponding MMU Base Pointer register. See Sections 2.11 - 2.12 for more information on memory mapped registers, and Section 7.3 for the effect this bit has on the Video Memory Map.

Bits 2:0 These bits form a 3-bit value equal to the number of refresh cycles to the DRAM per scan line, i.e., if programmed to 000 then no refresh operation occurs.

NOTE: Bits <5:3> are reset to 0 when a synchronous reset is done (by setting TS Indexed Register 0, bit 1 equal to 0).

5.2.35 CRTC Indexed Register 37: Video System Configuration 2 (VSCONF2)

(protected by KEY)
I/O address = 3#5

Bit	Description	Access
7	FIFO low threshold control.	RW
6	Test: 1=TLI internal test mode.	RW
5	FIFO high threshold control.	RW
4	Reserved.	
3	Effective Row/Column memory address (AB<9:0>, AA<9:0>).	RW
2	Memory data latch delay.	RW
1	Reserved.	
0	Display Memory data bus width.	RW

Bit Description

Bit 7 When set to 1, increases the FIFO low threshold control. For example, if the CRTC or CRTCB (primary or secondary window) is 16 bits/pixel and the MCLK to SCLK ratio is > 1.25 (this is the recommended setting in this case); or, if the 128x128-pixel Sprite is enabled.

Bit 6 When set to 1, directs the ET4000/W32p to internal test mode set-up. This bit must be set to 0 at all other times for normal operation. Default power-up condition is 0.

Bit 5 When set to 0, will increase the utilization of the display memory's bandwidth. However, the memory's response time to the host might be increased. This bit should normally be set to 0 for better performance. Default power-up condition is 0.

Bit 3 Determines the effective row/column memory address, as illustrated in the following table:

Bit 3 DRAM Type	Programmed value	Row Address	Column Address
256K x 4,8,16	1	AB<8:0>	AB<8:0>
		AA<8:0>	AA<8:0>
1MB x 4	0	AB<9:0>	AB<9:0>
		AA<9:0>	AA<9:0>

Bit 2 ***Except Revisions A & B:*** When set to 0, enables OE controlled interleave operation. This enables the four OE controlled signals on the MUX<1>, EDCK. ESYC, and EVID pins. This can only be used on configurations where these pins are defined. See Section 6.4.2.1 OE Controlled Interleave DRAM Interface.

Revisions A & B ONLY: When set to 0, delays sampling (3ns delay typical) of the MD bus from the DRAM, allowing use of DRAMs with slower access time (slower tcac or taa).

When set to 1 (power-up default), provides normal sampling of MD bus. It is recommended that this bit be set to 1 for interleave memory configurations.

NOTE: The ET4000/W32p utilizes the interleave capability and uses the 4 CAS* and 2 WRITE ENABLE* signal configuration exclusively. Reference sample schematics for memory design methods. The W32p interleave DRAM requires 4 additional signals: CAS<7:4>, which are pins 95, 96, 3, and 123, respectively.

Bit 0 Determines the width (from 16-bit to 32-bit) of the MD<31:0> bi-directional display memory data bus:

Bit 0	MD<31:24>	MD<23:16>	MD<15:8>	MD<7:0>	Bus Width
1	MD<31:24>	MD<23:16>	MD<15:8>	MD<7:0>	32
0	—	MD<15:8>	—	MD<7:0>	16

5.2.36 CRTC Indexed Register 3F: Horizontal Overflow

I/O address = 3#5

Bit	Description	Access
7	Row Offset Bit 8.	RW
6	Reserved.	
5	Reserved.	
4	Horizontal Sync Start Bit 8.	RW
3	Reserved.	
2	Horizontal Blank Start Bit 8.	RW
1	Reserved (always set to 0).	
0	Horizontal Total Bit 8.	RW

Bit **Description**

Bit 7 Provides a ninth bit to specify the amount to be added to the internal linear counter when advancing from one screen row to the next. See Section 5.2.22, CRTC Indexed Register 13: Row Offset.

Bit 4 Provides a ninth bit for the value of the internal horizontal character counter at which horizontal sync is to start. See Section 5.2.7, CRTC Indexed Register 4: Horizontal Sync Start.

Bit 2 Provides a ninth bit for the value of the internal horizontal character counter at which horizontal blanking is to start. See Section 5.2.5, CRTC Indexed Register 2: Horizontal Blank Start.

Bit 0 Provides a ninth bit to define the horizontal scan line time. See Section 5.2.3, CRTC Indexed Register 0: Horizontal Total.

5.3 TS Register Descriptions

The CPU interface to the ET4000/W32p internal Timing Sequencer (TS) consists of eight read/write registers. Of these registers, one register, the TS Index Register, is accessed by a separate independent I/O address (3C4). The remaining seven registers are internally indexed, which means that they are accessed via a common I/O address (3C5), with one of the seven registers that is actually selected by the TS Index Register.

Table 5.3-1 TS Index Register

Port Register Name		Address
TS Index Register	(Read/Write)	3C4

Table 5.3-2 TS Indexed Registers

TS Indexed Register	TS Indexed Name	Port Address	Address
Synchronous Reset	0	(Read/Write)	3C5
TS Mode	1	(Read/Write)	3C5
Write Plane Mask	2	(Read/Write)	3C5
Font Select	3	(Read/Write)	3C5
Memory Mode	4	(Read/Write)	3C5
Reserved	5		
TS State Control	6	(Read/Write)	3C5
TS Auxiliary Mode	7	(Read/Write)	3C5

5.3.1 TS Index

I/O address = 3C4

Bit	Description	Access
7:3	Reserved.	
2:0	Current TS index.	RW

Bit	Description
Bits 2:0	Provide the index of the currently selected internally indexed register. The TS Index register determines which TS indexed register will be accessed when a read/write is performed using port address 3C5.

5.3.2 TS Indexed Registers

The following registers are TS indexed registers. These registers are accessed by first writing the index of the desired register to the TS Index Register and then accessing the register using address 3C5.

5.3.3 TS Indexed Register 0: Synchronous Reset

I/O address = 3C5

Bit	Description	Access
7:2	Reserved.	
1	Synchronous reset control.	RW
0	Asynchronous reset control.	RW

Bit	Description
Bit 1	When set to 0, commands the timing sequencer to synchronously clear and halt. Both bits 0 and 1 must be set to 1 for the timing sequencer to run. For compatibility, a synchronous reset should be in effect whenever changing the Timing Sequencer or clock state. In general, synchronous reset periods should be kept as short as possible to prevent possible loss of display memory data.
Bit 0	When set to 0, commands the timing sequencer to synchronously clear and halt. When set to 1, the sequencer will run unless bit 1 is set to 0.

5.3.4 TS Indexed Register 1: TS Mode

I/O address = 3C5

Bit	Description	Access
7:6	Reserved.	
5	Screen off (fast mode).	RW
4	Shift 4.	RW
3	Dot clocks/2.	RW
2	Video load/2.	RW
1	Reserved.	
0	Timing sequencer state (bit 0).	RW

Bit	Description
Bit 5	When set to 1, will force blanking on the screen, allowing CPU access of video memory to go into a fast mode.
Bit 4	When set to 1, will allow the video shifter input latches to be loaded at quarter rate.
Bit 3	When set to 1, provides sequencer clocking at half the MCLK rate, known as dot clock/2 mode. This generates the dot clock signal at half the normal rate, effectively halving the pixel rate provided by the master clock. In VGA/EGA compatible operation, dot clock/2 mode is used in all display modes that have 320, rather than 640, pixels per scan line.
Bit 2	When set to 1, loads the video shifter (such as the ATC) input latches at half the video load rate.
Bit 0	This bit is used to set the timing sequencer state value. When set to a 0, the TS is set to State 0, or the 9-dot character clock; when set to a 1, the TS is set to State 1, or the 8-dot character clock.

5.3.5 TS Indexed Register 2: Write Plane Mask

I/O address = 3C5

Bit	Description	Access
7:4	Reserved.	
3	Write enable display memory plane 3.	RW
2	Write enable display memory plane 2.	RW
1	Write enable display memory plane 1.	RW
0	Write enable display memory plane 0.	RW

Bit **Description**

Bits 3:0 The Write Plane Mask register enables or disables CPU write access to display memory planes on a plane-by-plane basis, and is only useful for 16-color (plane) systems. In 256 color mode, this register should be set to "0F" hex.

5.3.6 TS Indexed Register 3: Font Select

I/O address = 3C5

Bit	Description	Access
7:6	Reserved.	
5,3,2	Font Select B (FSB<2:0>).	RW
4,1,0	Font Select A (FSA<2:0>).	RW

Bit **Description**

Bits 5:0 FSA or FSB (as selected by Attribute bit 3) is used to select one of eight possible soft fonts, providing two simultaneous character sets for display.

Based on the Selection bits derived, the font memories are selected as follows:

Selection Bits (SEL<2:0>)	Selected Segment	Offset in Font Memory
0 0 0	0	0
0 0 1	1	16K
0 1 0	2	32K
0 1 1	3	48K
1 0 0	4	8K
1 0 1	5	24K
1 1 0	6	40K
1 1 1	7	56K

NOTE: When ATC Indexed Register, bit 7 is set to 1, this register is not used. By using FS<2:0>, CC<7:0>, and RA<4:0> as listed in Table 7.3-1, 8 simultaneous fonts and character sets are available (B/W), and 4 and 4 respectively for Color. A total of 2048 character codes are available from which one could define 8 sets of 256 cc's each. 256 is a standard, albeit arbitrary number. The FS, CC, and RA pointers define which of the fonts and character sets are being used at a given time. If ATC Indexed Register 7, bit 7 is 0, this register is used as is.

5.3.7 TS Indexed Register 4: Memory Mode

I/O address = 3C5

Bit	Description	Access
7:4	Reserved.	
3	Enable Chain 4.	RW
2	Odd/even mode.	RW
1	Extended memory.	RW
0	Reserved.	

Bit Description

Bit 3 When set to a 1, will enable Chain 4 (linear graphics) mode, where all four memories are chained linearly into a byte-oriented memory array whereby each byte will provide the eight bits (256-color) for each pixel. When set to 1, causes the two low-order bits of the address (A1 and A0) to select the plane that is accessed:

A<1:0>	Plane
0 0	0
0 1	1
1 0	2
1 1	3

When set to 0, the processor will access data sequentially in the bit plane.

Bit 2 When set to 0, selects odd/even mode, in which even display memory planes (0 and 2) are active on display CPU accesses to even memory addresses (A0=0), while odd memory planes (1 and 3) are active on accesses to odd memory addresses (A0=1). When set to 1, causes the processor addresses to write to display memory planes according to the Write Map mask register.

Bit 1 When set to 1, enables selection among multiple fonts, where one of up to eight fonts can be selected (See Section 5.3.6, Font Select Register).

5.3.8 TS Indexed Register 6: TS State Control (protected by KEY)

I/O address = 3C5

Bit	Description	Access
Bits <7:4> are Reserved for Revision A.		
7	Enable synchronous host interface.	RW
6	Enable zero read/write wait state.	RW
5	Enable fast read command.	RW
4	Enable fast write command.	RW
3	Reserved.	
2:1	Timing sequencer state (bits 1 & 2).	RW
0	Reserved.	

Bit Description

Bit 7 When set to 1, defines the system clock (SCLK) to run synchronously with the PCI bus clock (BCLK). The power-up default setting is 0.

Bit 6 When set to 0, enables zero read/write wait state. The default setting is 0.

Bit 5 When set to 1, enables faster internal operation of the read command based on having early address space information (A<31:2>, BE<3:0>). In order to use this, the address bus set-up time to LCLK must be greater than 18ns.

Bit 4 When set to 1, enables faster internal operation of the write command based on having early address spaceinformation (A<31:2>, BE<3:0>). In order to use this, the address bus set-up time to LCLK must be greater than 18ns and write data is available at the 2nd LCLK (command/data phase).

NOTE: If bits <5:4> are set to 1 and burst (CRTC Indexed Register 34 <4> = 1) is enabled, the minimum zero wait state read/write operation is 2 LCLKs, otherwise it is 3 LCLKs.

NOTE: ***Except Revisions A & B*** If either bit<5> or bit<4> is set to 1 then there will be no turn-around cycle for DEVSEL in PCI mode.

Bits 2:1 These bits are used to set the extended timing sequencer state value. In conjunction with bit 0 of the TS Mode register, the additional states are used to define the number of dots per character in text mode:

TS Bit <2:1>	Mode 0	dots/char
1 1	1	16
1 0	0	12
0 1	1	11
0 1	0	10
0 0	1	8
0 0	0	9
1 0	1	7
1 1	0	6

IMPORTANT: All CRTC "character" timing calculations are based on programmed number dots/char.

5.3.9 TS Indexed Register 7: TS Auxiliary Mode (protected by KEY)

I/O address = 3C5

Bit	Description	Access
7	VGA mode.	RW
6	Select MCLK/2 (if bit 0 is set to 0).	RW
5	BIOS ROM Address Map 2.	RW
4	Reserved (Set to 1 always).	RW
3	BIOS ROM Address Map 1.	RW
2	Reserved (Set to 1 always).	
1	Reserved.	
0	Select MCLK/4.	RW

Bit **Description**

Bit 7 When set to 1, enables VGA compatibility. A value of 0 will enable EGA compatibility. NOTE: The ET4000/W32p is set to default on power-up to VGA mode.

Bit 6 When set to 1, will divide the MCLK input clock frequency by two if bit 0 is equal to 0.

Bits 5,3 These bits are used for selection of ROM BIOS address space.

Bit 3	Bit 5	ROM BIOS Address Map Space Allocation	Total Memory Used
0	0	C0000-C3FFF	16KB
0	1	disabled	0KB
1	0	C0000-C5FFF; C6800-C7FFF	30KB
1	1	C0000-C7FFF*	32KB

* Power-up default

Bit 0 When set to 1, will divide the MCLK input clock frequency by 4.

5.4 GDC Register Descriptions

The CPU interface to the ET4000/W32p internal Graphics Data Controller (GDC) consists of 11 read/write registers. Of these registers, two are accessed by separate independent I/O addresses. The remaining 9 registers are internally indexed, which means that they are accessed via a common I/O address (3CF), with one of the 9 registers that is actually accessed selected by the GDC Index register.

Table 5.4-1 GDC Registers and Addresses

Register Name	Port Address	Indexed Address
Segment Select 1	R/W : 3CD	
Segment Select 2	R/W : 3CB	
GDC Index register	R/W : 3CE	
Indexed Register Name		
Set/Reset	R/W : 3CF	0
Enable Set/Reset	R/W : 3CF	1
Color Compare	R/W : 3CF	2
Data Rotate	R/W : 3CF	3
Read Plane Select	R/W : 3CF	4
GDC Mode	R/W : 3CF	5
Miscellaneous	R/W : 3CF	6
Color Care	R/W : 3CF	7
Bit Mask	R/W : 3CF	8

5.4.1 GDC Segment Select

I/O address = 3CB,3CD

Bit 3CB	Bit 3CD	Description	Access
5:4	7:4	Read segment pointer (RSP<5:0>).	RW
1:0	3:0	Write segment pointer (WSP<5:0>).	RW

Bit Description

When CRTC Indexed Register 36 (Video System Configuration 1) bit 4 is set to 0, then:

Bits
5:4 3CB A 6-bit segment pointer selects one of 64 segments
7:4 3CD (segment 0 to F) for CPU read operations.

Bits
1:0 3CB A 6-bit segment pointer selects one of 64 segments
3:0 3CD (segment 0 to F) for CPU write operations.

NOTE: This register is reset to 0 when a synchronous reset is done (by setting TS Indexed Register 0, bit 1 equal to 0).

5.4.2 GDC Index

I/O address = 3CE

Bit	Description	Access
7:4	Reserved.	
3:0	Current index.	RW

Bit	Description
Bits 3:0	Provide the index of the currently selected internally indexed register. The GDC Index register determines which GDC indexed register will be accessed when a read/write is performed using port address 3CF.

5.4.3 GDC Indexed Registers

The remaining GDC registers are indexed registers, accessed by first writing the index value into the GDC Index register, and then accessing the indexed register using port address 3CF.

5.4.4 GDC Indexed Register 0: Set/Reset

I/O address = 3CF

Bit	Description	Access
7:4	Reserved.	
3	Set/reset value for map 3.	RW
2	Set/reset value for map 2.	RW
1	Set/reset value for map 1.	RW
0	Set/reset value for map 0.	RW

Bit	Description
Bits 3:0	Each set/reset bit specifies the value to be written to all bits of the addressed byte of the corresponding memory map (or plane), 0 through 3, when the set/reset function is enabled for that map. (See Section 5.4.5, GDC Indexed Register 1.)

5.4.5 GDC Indexed Register 1: Enable Set/Reset

I/O address = 3CF

Bit	Description	Access
7:4	Reserved.	
3	Enable set/reset value for map 3.	RW
2	Enable set/reset value for map 2.	RW
1	Enable set/reset value for map 1.	RW
0	Enable set/reset value for map 0.	RW

Bit **Description**

Bits 3:0 Each enable set/reset bit enables or disables the set/reset function for the corresponding memory map (or plane), 0-3. When any of bits 3:0 are set to 0, the set/reset function in the corresponding plane will be disabled. When set to 1, the set/reset function will be enabled. When enabled, the set/reset function stores either a 0 or FF value in the addressed byte of a given plane, depending on the set/reset value (see Section 5.4.4, GDC Indexed Register 0). When set/reset is enabled for a plane, the logical functions (see Section 5.4.7, GDC Indexed Register 3) operate on the set/reset value for each plane and the latched data for that plane; the bit mask (see Section 5.4.12, GDC Indexed Register 8) is also in effect. When the set/reset function is disabled, the addressed byte in a given plane is written as a combination of latched and CPU data, according to the write mode in effect and the bit mask, and the set/reset value has no effect. The set/reset function has no effect in write mode 1.

5.4.6 GDC Indexed Register 2: Color Compare

I/O address = 3CF

Bit	Description	Access
7:4	Reserved.	
3	Color compare value for plane 3 bits.	RW
2	Color compare value for plane 2 bits.	RW
1	Color compare value for plane 1 bits.	RW
0	Color compare value for plane 0 bits.	RW

Bit **Description**

Bits 3:0 The Color Compare register is used in read mode 1 to determine which pixels from the display memory location, read by the CPU, match a specified color. The 4-bit color value in the color compare register is compared to the 4-bit color value of each of the eight pixels, spread across the four planes.

From this comparison, a bit value of 1 is returned in the data byte to the CPU, at the position corresponding to each pixel that matches the Color Compare register, and 0 is returned for each pixel that does not match the Color Compare register. In other words, an 8-bit value is returned to identify the comparison for all eight pixels.

NOTE: Both the Color Compare and Color Care registers are useful only in the "PLANE" (16 colors) systems. In the "LINEAR BYTE" (256 colors) systems, the color compare operation should be performed at the CPU level.

5.4.7 GDC Indexed Register 3: Data Rotate and Function Select

I/O address = 3CF

Bit	Description	Access
7:5	Reserved.	
4:3	Function select.	RW
2:0	Rotate count.	RW

Bit Description

Bits 4:3 Select the logical operation to be performed by the ALU on incoming CPU data and latched data. The logical operation is performed on only those bits that are enabled by the bit mask register; mask-disabled bits are written as the latched value (resulting from previous memory reads) only. For those bits that are mask-enabled, one of four logical operations may be performed between CPU data and latched data by setting the function select as follows:

Bit 4	Bit 3	Logical operation
0	0	MOVE CPU data through unchanged.
0	1	AND CPU data with latched data.
1	0	OR CPU data with latched data.
1	1	XOR CPU data with latched data.

Note that write mode 1 may be used to write latched data unmodified; the same effect could be obtained by ANDing a CPU data byte of FF, ORing or XORing a CPU data byte of 0, or by setting the bit mask register to 0. The logical functions operate in write modes 0, 2 and 3 only; they are ignored in write mode 1.

Bits 2:0 These bits set the number of bits (0-7) by which CPU data should be rotated to the right before it is sent to the ALU for bit masking and logical functions. Rotation is circular, with bit 0 feeding back to bit 7.

5.4.8 GDC Indexed Register 4: Read Plane Select

I/O address = 3CF

Bit	Description	Access
7:2	Reserved.	
1:0	Plane select.	RW

Bit Description

Bits 1:0 Select the memory plane 0-3 from which the addressed byte is to be read and returned on the CPU data bus, in the "PLANE" (16 colors) configurations. Only one plane can be read at any one time.

5.4.9 GDC Indexed Register 5: GDC Mode

I/O address = 3CF

Bit	Description	Access
7	Reserved.	
6	Enable 256 color mode.	RW
5	Reserved.	
4	Odd/even mode.	RW
3	Read mode.	RW
2	Reserved.	
1:0	Write mode.	RW

Bit **Description**

Bit 6 When set to 0, permits the loading of the ATC's shift registers to be controlled by bit 5. When set to 1, the registers are loaded to support the 256-color mode.

Bit 4 When set to 1, selects odd/even addressing mode, in which even maps are accessed with even addresses and odd maps are accessed with odd addresses. The function of this bit is to determine from which display map data is to be routed to the CPU data bus on a CPU read in odd/even mode. If bit 4 is 1 and the Read Map Select register selects either of two maps in a given pair, then the even map is selected if address line 0 is 0, and the odd map is selected if address line 0 is 1. Bit 2 of CRTC Timing Sequencer Indexed Register 4 should be set to 0 to select odd/even mode, to generate all address control other than the read data selection in odd/even addressing mode. Odd/even addressing mode is useful for text modes.

Bit 3 Selects the read mode. When bit 3 is 0 (Read Mode 0), the data read from the map indicated by the read map select register (see Section 5.4.8, GDC Indexed Register 4) is returned on the CPU data bus. This is the normal read mode of operation. When bit 3 is 1, the color compare operation is enabled on a CPU read. (See Section 5.4.6, Color Compare Register).

Bits 1:0 These bits select the mode in which data bytes are to be written to screen memory. The write modes are:

Bit (1:0)	Write Mode Selected
0 0	Write mode 0. Each CPU data byte written to display memory, as modified by the current rotation setting (see Section 5.4.7, GDC Indexed Register 3), is combined with the latched data for each map according to the current logical function (see Section 5.4.7, GDC Indexed Register 3) and written to each memory map. The byte written by the CPU is passed identically to the ALU for each map; differences in the byte actually written to the screen may occur due to differences in the latch contents for different maps. If the set/reset function is enabled for any map (see Section 5.4.5, GDC Indexed Register 1), then the set/reset bit value for that map (see Section 5.4.4, GDC Indexed Register 0) is written to every bit of the addressed byte of that map regardless of the CPU data. The bit mask (see Section 5.4.12, GDC Indexed Register 8) applies in write mode 0, and causes the latch data alone to be written to each bit that is mask-disabled.

(continued on next page)

5.4.9 GDC Indexed Register 5: GDC Mode (cont'd)

Bit
(1:0)

0 1 Write mode 1. The data contained in the latches is written unmodified to the addressed byte in screen memory. All maps are written. This is useful for rapid data movement from display memory to display memory, as all maps can be latched with a single read and then written with a single write mode 1 operation. The bit mask is ignored,

Bit	Description

Bit
(1:0) Write Mode Selected (cont'd)

1 0 Write mode 2. Each bit, 0-3, of the data written by the CPU is extended to a byte and written to the four corresponding planes. Bit 0 of the data byte is extended to a byte and written to the addressed byte of map 0, bit 1 is extended to a byte and written to map 1, and so on up to bit 3, which is extended to a byte and written to map 3. The bit mask applies to the data byte for each map; that is, after the bit for each map from the CPU data written is extended to a byte, the byte for each map is masked as if it were the CPU data byte. The selected logical function operates normally on the byte for each map and the latched data for that map. The set/reset operation functions normally, overriding the write mode 2 bit for a given map when enabled. The data rotate register has no effect in write mode 2.

1 1 Write mode 3. Eight bits of the value contained in the Set/Reset register are written for each map. Rotated CPU data are ANDed with data from the bit mask register (see Section 5.4.12, GDC Indexed Register 8) to produce an 8-bit value that functions as the bit mask register does in write modes 0 and 2.

5.4.10 GDC Indexed Register 6: Miscellaneous

I/O address = 3CF

Bit	Description	Access
7:4	Reserved (= 0).	
3:2	Memory map.	RW
1	Enable odd/even mode.	RW
0	Graphics mode enable.	RW

Bit	Description

Bits 3:2 Memory Map—Control mapping of the Frame Buffer into CPU address space. NOTE: Bits 2&3 should be set to 0 when bit 4 of CRTC Indexed Register 36 is set to 1 (linear system).

See Section 7.3 for the effect this bit has on the Video Memory Map.

Bit 1 When set to 1, enables odd/even mode, will cause the replacement of the CPU address bit 0 with a high-order bit, and the odd/even maps are "chained" via the CPU A0 bit.

Bit 0 When set to 1, enables graphics mode.

5.4.11 GDC Indexed Register 7: Color Care

I/O address = 3CF

Bit	Description	Access
7:4	Reserved.	
3	Enable color compare color output 3.	RW
2	Enable color compare color output 2.	RW
1	Enable color compare color output 1.	RW
0	Enable color compare color output 0.	RW

Bit	Description
Bits 3:0	Each bit enables or disables the participation of the corresponding plane in a read mode 1 color comparison. When set to 1, the color compare is enabled for that plane (see Section 5.4.6, GDC Indexed Register 2: Color Compare). When set to 0, then the value in that plane has no effect on the value returned by the color comparison.

5.4.12 GDC Indexed Register 8: Bit Mask

I/O address = 3CF

Bit	Description	Access
7:0	Controls CPU data routing for corresponding bits of addressed screen map byte.	RW

Bit	Description
Bits 7:0	Each bit of the Bit Mask register either blocks the corresponding CPU data bit from affecting the value written to the screen or allows the CPU data bit through. A zero (0) value blocks and a 1 value passes CPU data. If a given bit is blocked, the value stored in that bit of each data latch (one for each plane) is sent to the corresponding screen plane. If a given bit is enabled, the value in that bit position of the CPU data is passed to the ALU, where it can be mixed with latched data via the selected logical function. The data will be rotated (see Section 5.4.7, GDC Indexed Register 3) before it is masked.

5.5 ATC Register Descriptions

The CPU interface to the ET4000/W32p internal Attribute Controller (ATC) consists of 23 read/write registers, and a separate flip-flop (1-bit register) which can be toggled between index/data mode. Two I/O addresses are used in conjunction with the index/data mode flip-flop to access the 23 registers as follows: An I/O read to the Input Status 1 register (3BA or 3DA depending on monochrome or color mode respectively, as controlled by bit 0 in the Miscellaneous Output Register) will reset the index/data flip-flop to index mode. Every I/O write with port address 3C0 will also toggle the index/data flip-flop between index and data mode. The index value in the ATC index register can be read with I/O address 3C0. While in index mode, the index value can be written to the ATC index register with I/O address 3C0, with the index/data mode flip-flop toggled to the data mode.

If the 16-bit I/O is enabled, an I/O WORD access to port 3C0 will automatically reset the index/data flip-flop. All of the 23 indexed registers can be read with I/O address 3C1. While in data mode, all of these indexed registers can be written to with I/O address 3C0, with the index/data mode flip-flop toggled to the index mode.

Table 5.5-1 ATC Index Register

Register Name	Port Address	Indexed Address
ATC Index register	R : 3C0 W : 3C0 (INDEX)	

Table 5.5-2 ATC Indexed Registers

Indexed Register Name	Port Address	Indexed Address
Palette	R : 3C1 W : 3C0 (DATA)	0-F
ATC Mode	R : 3C1 W : 3C0 (DATA)	10
Overscan	R : 3C1 W : 3C0 (DATA)	11
Color Plane Enable	R : 3C1 W : 3C0 (DATA)	12
Horizontal Pixel Panning	R : 3C1 W : 3C0 (DATA)	13
Color Reset	R : 3C1 W : 3C0 (DATA)	14
Miscellaneous	R : 3C1 W : 3C0 (DATA)	16
Miscellaneous 1	R : 3C1 W : 3C0 (DATA)	17

5.5.1 ATC Index

R : Port address = 3C0
W : Port address = 3C0 (index/data flip-flop in INDEX mode)

Bit	Description	Access
7:6	Reserved.	
5	Palette RAM address source.	RW
4:0	Current ATC index.	RW

Bit	Description
Bit 5	When set to 1, disables CPU write access to palette RAM and allows ATC access of RAM. This bit must be set to 0 before the CPU can update any palette RAM location. After the palette RAM is updated, this bit must be set to 1 so the ATC can access the palette RAM for video information. When set to 0, enables CPU write access to palette RAM, and replaces all video outputs with the contents of the overscan register.
Bits 4:0	Provide the index of the currently selected internally indexed register.

5.5.2 ATC Indexed Registers

The following registers are the ATC indexed registers. These registers are accessed by writing the index of the desired register to the ATC Index register when the index/data flip-flop is in INDEX mode. They are then accessed using the index value in the ATC Index Register. See details under previous paragraphs under ATC Register Descriptions.

5.5.3 ATC Indexed Registers 0-F: Palette RAM

R : Port address = 3C1
W : Port address = 3C0 (index/data flip-flop in DATA mode)

Bit	Description	Access
7	Reserved.	
6	Reserved.	
5	Secondary red video.	RW
4	Secondary green/intensity video.	RW
3	Secondary blue/mono video.	RW
2	Primary red video.	RW
1	Primary green video.	RW
0	Primary blue video.	RW

These 16 internal palette registers define a dynamic remapping between colors as defined by text attributes and graphics bit maps and the colors actually generated by the video circuitry. Each palette register 0-15 corresponds to an attribute, 0-15, in the "PLANE" (16 colors) configuration. Four bits (1 bit from each plane) of video data for a given pixel enters the palette RAM and addresses one of the 16 palette registers. The 6-bit value stored in the corresponding palette register is then transferred to the output latch of the ATC to provide the actual pixel data. In "linear byte" (256 colors) configuration, these

Bit	Description
Bits 5:0	When set to 1, select the appropriate color attribute. When set to 0, indicate the appropriate color is not present.

5.5.4 ATC Indexed Register 10: Mode Control

R : Port address = 3C1
W : Port address = 3C0 (index/data flip-flop in DATA mode)

Bit	Description	Access
7	SB/SG select.	RW
6	PELCLOCK/2.	RW
5	Enable pixel panning.	RW
4	Reserved.	
3	Background intensity/blink.	RW
2	Line graphics enable.	RW
1	Mono/color select.	RW
0	Graphics/text select.	RW

Bit Description

Bit 7 Used to select for the SB and SG video bits. When set to 1, SB and SG are bits 0 and 1, respectively, of the Color Select Register.

When set to 0, SB and SG are bits 4 and 5 of the internal palette register. This is not applicable to linear graphics (256-color) modes, for which SB and SG always come from memory data.

Bit 6 When set to 1, halves the rate of pixel output to the screen such that only 4, as opposed to the usual 8, pixels are output in a character clock time. This is normally used only for the 320x200 256-color graphics mode. For all other 256-color modes, this bit should be set to 0.

Bit 5 When set to 1, disables pixel panning while in split screen.

When set to 0, enables panning. NOTE: pixel panning is not supported in the A window if the B window (CRTCB) overlays the A window.

Bit 3 When set to 1, enables blinking in both text and graphics modes. When enabled in text mode, blinking occurs whenever bit 7 of the attribute byte for a given character is 1; when enabled in graphics mode, blinking occurs for all bits that have a 1 in the intensity plane. Blinking is performed by toggling the most significant address line (bit 3) into the palette RAM, thus toggling the video data between the lower eight and upper eight palette RAM registers. This means that the effect of the blink (for example, reverse video to video, video to high-intensity video, dark to dark) is completely programmable. Bit-mapped graphics modes can be programmed to support all the attributes of text modes, for instance.

NOTE: The non-blinking bits will use the upper eight palette registers.

When set to 0, disables blinking; in this case bit 3 of the palette RAM address is multiplexed directly from the video data to the palette RAM. When bit 3 is 0, all 16 simultaneous colors are enabled in graphics mode; in text mode, all 16 background colors are available simultaneously.

(continued on next page)

5.5.4 ATC Indexed Register 10: Mode Control (cont'd)

Bit	Description
Bit 2	When set to 1, specifies that in the 9 dots/character state (controlled by the CRTC), the ninth (and last) dot produced horizontally per character should replicate the eighth dot for character codes C0 hex through DF hex. The ninth dot of all other character codes is always 0 when line graphics is enabled. This is normally used to allow the text mode line graphics characters supported on the IBM Monochrome Display, which are 8-dot- wide characters in a 9-dot-wide character box, to connect. If this bit is 0 and the CRTC is set to the 9 dots/character state, then the 9th dot will display bit 7 of Intensity Memory plane (plane 3). For states greater than 9 dots, this bit should be set to 0. The 9th and subsequent dots are taken from bits 7,6, etc. of Intensity Memory plane (plane 3).
Bit 1	When set to 1, selects a monochrome display attribute; when set to 0, enables a color display attribute.
Bit 0	When set to 1, enables the ATC to process the pixel data in graphics mode; when set to 0, enables the ATC to process the pixel data in text mode.

5.5.5 ATC Indexed Register 11: Overscan Color

R : Port address = 3C1
W : Port address = 3C0 (index/data flip-flop in DATA mode)

Bit	Description	Access
7	Secondary Intensity border color.	RW
6	Secondary Red border color.	RW
5	Secondary Green border color.	RW
4	Secondary Blue border color.	RW
3	Intensity border color.	RW
2	Red border color.	RW
1	Green border color.	RW
0	Blue border color.	RW

This register defines the color to be displayed around the perimeter of the working screen area (the border or overscan color).

Bit	Description
Bits 7:0	When set to 1, select the appropriate border color/attribute, each bit corresponding to one of the output pins. This value is a 0 for the monochrome display.

5.5.6 ATC Indexed Register 12: Color Plane Enable

R : Port address = 3C1
W : Port address = 3C0 (index/data flip-flop in DATA mode)

Bit	Description	Access
7:6	Reserved.	
5:4	Video status select.	RW
3:0	Enable plane.	RW

Bit Description

Bits 5:4 These bits select two of eight color outputs to be returned by the Status register, as follows:

Input Status Register 1

Bit 5	Bit 4	Bit 5	Bit 4
0	0	PR	PB
0	1	SG	SB
1	0	PI	PG
1	1	SI	SR

Bits 3:0 In "PLANE" (16 colors) configuration, the color plane relative to each of bits 0-3 is enabled when the appropriate individual bits are set to one. Bits 0,1,2,3 control the enabling of planes 0,1,2,3, respectively.

5.5.7 ATC Indexed Register 13: Horizontal Pixel Panning

R : Port address = 3C1
W : Port address = 3C0 (index/data flip-flop in DATA mode)

Bit	Description	Access
7:4	Reserved.	
3:0	Horizontal pixel panning.	RW

Bit Description

Bits 3:0 These bits specify the number of pixels by which the video data should be shifted to the left. Shifts of up to nine pixels are supported. Note that in 9-dot modes, a value of 8 signifies no shift, and the values of 0-7 signify shifts of 1-8 pixels, respectively.

NOTE: In 6- and 7-dot modes, values of 2 and 1, respectively, signify no shift.

5.5.8 ATC Indexed Register 14: Color Select

R : Port address = 3C1
W : Port address = 3C0 (index/data flip-flop in DATA mode)

Bit	Description	Access
7:4	Reserved.	
3	S_color 7.	RW
2	S_color 6.	RW
1	S_color 5.	RW
0	S_color 4.	RW

Bit	Description
Bits 3:2	Provide the two high-order bits of the exported digital color value in plane systems. With 256-color graphic modes, the 8-bit attribute value becomes the 8-bit digital value exported from the chip.
Bits 1:0	Available for replacement use of bits 5 and 4 of the attribute palette registers, forming an 8-bit value for color to be exported from the chip. When bit 7 of the ATC Mode register is set to 1, bits 1 and 0 are selected as SG and SB outputs of the plane system.

5.5.9 ATC Indexed Register 16: Miscellaneous (protected by KEY)

R : Port address = 3C1
W : Port address = 3C0 (index/data flip-flop in DATA mode)

Bit	Description	Access
7	Bypass the internal palette.	RW
6	Token Status Enable.	RW
5:4	Select High Resolution/color mode.	RW
3:2	Reserved.	
1	Protect external DAC.	RW
0	Protect border.	RW

Bit	Description
Bit 7	When set to 1, causes the internal palette to be bypassed (effectively, the output value equals the input value).
Bit 6	This bit is used in conjunction with IMA Indexed Register F7 <6:5> and CRTCB/Sprite Control Register (Index EF)<1:0> to determine the definition of the MUX<1:0> output pins. See Section 3.3.5 MUX<1:0>.

5.5.9 ATC Indexed Register 16: Miscellaneous (cont'd)

Bit **Description**

Bits 5:4 These bits, in combination, select normal power-up 8-bit per PCLK or else 16-bit per PCLK (AP<15:0>) output. Note that to support ET4000 Rev. G's high color mode, configure CRTC and ATC to 8-bit per pixel (256 color mode) and clock AP<7:0> with double-clocking.

Bit 5	Bit 4	
0	0	Normal power-up default (8-bit/clock)
1	0	16 bit per clock mode
0	1	Reserved
1	1	Reserved

Bit 1 When set to 1, disables I/O writes to External/Internal Palette RAM. Normal power-up default is set to 0.

Bit 0 When set to 1, disables I/O writes to Overscan Color register bits <3:0>. Normal power-up default is set to 0.

5.5.10 ATC Indexed Register 17: Miscellaneous 1 (protected by KEY)

R : Port address = 3C1
W : Port address = 3C0 (index/data flip-flop in DATA mode)

Bit	Description	Access
7	Redefine attribute (SMAE).	RW
6:0	Reserved.	

Bit **Description**

Bit 7 When set to 1, protects the internal palette RAM and is used to redefine the attribute bits as follows:

Monochrome

Attribute Bit	Description
7	Normal/reverse video
6	Full/half intensity
5	Character visible/invisible
4	Underline off/on
3	Blinking off/on
2:0	Font select

Bit **Description**

Bit 7 When set to 1, enables the reverse video attribute. When set to 0, displays normal video.

Bit 6 When set to 1, changes the character intensity to half. When set to 0, displays full intensity.

(continued on next page)

5.5.10 ATC Indexed Register 17: Miscellaneous 1 (cont'd)

Bit **Description**

Bit 5 When set to 1, disables characters from being displayed. When set to 0, enables normal character display.

Bit 4 When set to 1, turns the underline attribute on. When set to 0, enables normal character display.

Bit 3 When set to 1, turns the blinking attribute on. When set to 0, enables normal character display.

Bits 2:0 Used to select up to eight simultaneous soft fonts and up to eight simultaneous character sets for display. See Table 7.3-1 CPU/CRTC Addressing Modes, note 4; bits <2:0> here correspond to FS<1:0,2> in the table. (When ATC Indexed Register 17 bit 7 is set to 1 TS Indexed Register 3: Font Select is not used.)

NOTE: To get the attributes indicated here the ATC Palette RAM registers 0-7 must be programmed to 0,0,0,0,18,0,8,0 where 0=off, 8=half intensity, and 18=full intensity, and blinking should be enabled (via ATC Indexed Register 10 bit 3).

Color

Attribute bit	Description
7	Background red
6	Background green
5	Background blue
4	Foreground red
3	Foreground green
2	Foreground blue
1:0	Font select

Bits 7:5 When set to 1, are used to select background colors of red, green, and blue, respectively.

Bits 4:2 When set to 1, are used to select foreground colors of red, green, and blue, respectively.

Bits 1:0 Used to select up to four simultaneous soft fonts and up to four simultaneous character sets for display. Bit 2 is not used for font select for color operation. See Table 7.3-1 CPU/CRTC Addressing Modes, note 4. (When ATC I<7> is set to 1, TS Indexed Register 3: Font Select is not used.)

NOTE: To get the attributes indicated here, the ATC Palette RAM registers 8-F should be set equal to register 0-7 (registers 0-7 containing the normal range of colors), and blinking should be disabled (via ATC Indexed Register 10 bit 3). Also, the underline register (CRTC Indexed Register 14) should be set equal to or greater than the character height.

5.6 CRTCB/Sprite Register Descriptions

Index	Register
E0	CRTCB/Sprite Horizontal Pixel Position Low
E1	CRTCB/Sprite Horizontal Pixel Position High
E2	CRTCB Width Low/Sprite Horizontal Preset
E3	CRTCB Width High
E4	CRTCB/Sprite Vertical Pixel Position Low
E5	CRTCB/Sprite Vertical Pixel Position High
E6	CRTCB Height Low/Sprite Vertical Preset
E7	CRTCB Height High
E8	CRTCB/Sprite Starting Address Low
E9	CRTCB/Sprite Starting Address Middle
EA	CRTCB/Sprite Starting Address High
EB	CRTCB/Sprite Row Offset Low
EC	CRTCB/Sprite Row Offset High
ED	CRTCB Pixel Panning
EE	CRTCB Color Depth
EF	CRTCB/Sprite Control

The CRTCB/Sprite registers are accessed using an indexed addressing scheme whereby a number selecting a register is first written to address 21xA (Index) and then the register can be read from or written to at address 21xB. The CRTCB/Sprite registers use indices E0 through EF.

The Index Register at address 21xA is also used to address the IMA Indexed Registers, see Section 5.7 for details.

The value of 'x' in the addresses 21xA and 21xB is determined by the logical value on the IOD<2:0> pins of the chip at power-up reset; see Section 3 for details.

5.6.1 CRTCB Index Register

I/O address = 21xA

Bit	Description	Access
7:0	Indexed register select.	RW

Bit	Description
Bits 7:0	This register is used to select the CRTCB/Sprite indexed register that is accessed when address 21xB is read or written.

5.6.2 CRTCB/Sprite Horizontal Pixel Position Low (Index: E0)

I/O address = 21xB

Bit	Description	Access
7:0	Horizontal pixel position, bits <7:0>.	RW

5.6.3 CRTCB/Sprite Horizontal Pixel Position High (Index: E1)

I/O address = 21xB

Bit	Description	Access
7:3	Reserved.	
2:0	Horizontal pixel position, bits <10:8>.	RW

Bit	Description
Bits 11:0	The Horizontal Pixel Position is the position in pixels of the leftmost edge of the actively displayed portion of the CRTCB window or Sprite, relative to the leftmost edge of the CRTC active picture area.

5.6.4 CRTCB Width Low/Sprite Horizontal Preset (Index: E2)

I/O address = 21xB

Bit	Description	Access
7:0	CRTCB width, bits <7:0>/Sprite horizontal preset.	RW

Bit	Description
Bits 7:0	When the CRTCB is selected (with CRTCB/Sprite Indexed Register EF, bit 0), this register contains bits <7:0> of the 12-bit value defining the width of the CRTCB window in pixels. The value loaded should be 1 less than the desired width. When the Sprite is selected, bits <5:0> specify the Horizontal Pixel Preset. That is, the horizontal position relative to the beginning of the sprite area at which display of the sprite starts. The sprite does not wrap and always ends at position 63. See also Section 2.3 Secondary CRTC/Sprite (CRTCB).

5.6.5 CRTCB Width High (Index: E3)

I/O address = 21xB

Bit	Description	Access
7:3	Reserved.	
2:0	CRTCB width, bits <10:8>.	RW

Bit Description

Bits 2:0 When the CRTCB is selected (with CRTCB/Sprite Indexed Register EF, bit 0), this register contains bits <10:8> of the 12-bit value defining the width of the CRTCB window in pixels. The value loaded should be 1 less than the desired width.

NOTE: Programming of the CRTCB display area must not exceed the boundaries of the CRTC (primary) display area. This restriction does not apply when the Sprite is enabled.

5.6.6 CRTCB/Sprite Vertical Pixel Position Low (Index: E4)

I/O address = 21xB

Bit	Description	Access
7:0	Vertical pixel position, bits <7:0>.	RW

5.6.7 CRTCB/Sprite Vertical Pixel Position High (Index: E5)

I/O address = 21xB

Bit	Description	Access
7:3	Reserved	
2:0	Vertical pixel position, bits <10:8>.	RW

Bit Description

Bits 11:0 The Vertical Pixel Position is the position in scan lines of the topmost edge of the actively displayed portion of the CRTCB window or Sprite, relative to the topmost edge of CRTC active picture area.

5.6.8 CRTCB Height Low/Sprite Vertical Preset (Index: E6)

I/O address = 21xB

Bit	Description	Access
7:0	CRTCB height, bits<7:0>/Sprite Vertical Preset.	RW

Bit Description

Bits 7:0 When the CRTCB is selected (with CRTCB/Sprite Indexed Register EF, bit 0), this register contains bits <7:0> of the 12-bit value defining the height of the CRTCB window in scan lines. The value loaded should be 1 less than the desired height.

When the Sprite is selected, bits <5:0> specify the Vertical Pixel Preset. That is, the vertical position relative to the beginning of the 64x64 pixel sprite area at which display of the sprite starts. The sprite does not wrap and always ends at position 63. See also Section 2.3 Secondary CRTC/Sprite (CRTCB).

NOTE: Programming of the CRTCB display area must not exceed the boundaries of the CRTC (primary) display area. This restriction does not apply when the Sprite is enabled.

5.6.9 CRTCB Height High (Index: E7)

I/O address = 21xB

Bit	Description	Access
7:3	Reserved.	
2:0	CRTCB height, bits <10:8>.	RW

Bit Description

Bits 2:0 When the CRTCB is selected (with CRTCB/Sprite Indexed Register EF, bit 0), this register contains bits <10:8> of the 12-bit value defining the height of the CRTCB window in scan lines. The value loaded should be 1 less than the desired height.

NOTE: Programming of the CRTCB display area must not exceed the boundaries of the CRTC (primary) display area. This restriction does not apply when the Sprite is enabled.

5.6.10 CRTCB/Sprite Starting Address Low Register (Index: E8)

I/O address = 21xB

Bit	Description	Access
7:0	CRTCB/Sprite starting address, bits <7:0>.	RW

5.6.11 CRTCB/Sprite Starting Address Middle Register (Index: E9)

I/O address = 21xB

Bit	Description	Access
7:0	CRTCB/Sprite starting address, bits <15:8>.	RW

5.6.12 CRTCB/Sprite Starting Address High Register (Index: EA)

I/O address = 21xB

Bit	Description	Access
7:4	Reserved.	
3:0	CRTCB/Sprite starting address, bits <19:16>.	RW

Bit	Description
Bits 19:0	These three registers define a 20-bit offset into display memory at which the CRTCB/Sprite pixel data is located. The starting address is measured in doublewords, so if the data resides at byte address 256, then a value of 64 (i.e., 256/4) should be programmed into these registers.

5.6.13 CRTCB/Sprite Row Offset Low Register (Index: EB)

I/O address = 21xB

Bit	Description	Access
7:0	Memory address offset, bits <7:0>.	RW

Bit Description

Bits 7:0 This and the following register comprise the CRTCB/Sprite row offset, which specifies the number of quadwords from the start of one row of the CRTCB pixel map to the start of the next row. This register MUST be programmed to "2" when Sprite is enabled.

5.6.14 CRTCB/Sprite Row Offset High Register (Index: EC)

I/O address = 21xB

Bit	Description	Access
7:4	Revision ID.	RO
3:1	Reserved.	
0	Memory address offset bit <8>.	RW

Bit Description

Bits 7:4 These bits are used to indicate the chip and revision level. The values are defined as follows:

Bit 7 6 5 4	Chip/Rev.
0 0 0 0	W32
0 0 0 1	W32i
0 0 1 0	W32p, Revision A
0 0 1 1	W32i, Revision B
0 1 0 1	W32p, ***Revision B***
0 1 1 1	W32p, ***Revision C***
↓ ↓ ↓ ↓	
1 1 1 1	Reserved

Bit 0 This is the ninth bit of these two registers, CRTCB/Sprite Row Offset Low and CRTCB/Sprite Row Offset High, which specify the number of quadwords between the start of one row of the CRTCB pixel map to the start of the next row. This field MUST be programmed to "2" when the Sprite is enabled.

5.6.15 CRTCB Pixel Panning (Index: ED)

I/O address = 21xB

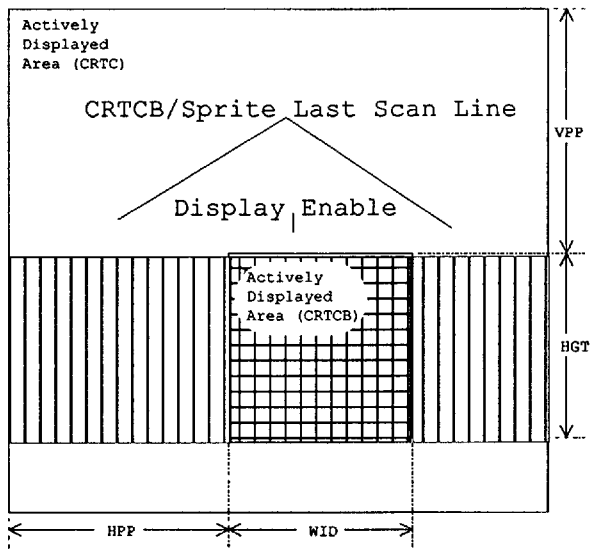
Bit	Description	Access
7	CRTCB/Sprite last scan line.	RO
6	Display enable.	RO
5:4	Reserved.	
3:0	CRTCB pixel panning.	RW

Bit **Description**

Bit 7 When high indicates is in the horizontal blank period prior to the first scan line of the CRTCB/Sprite.

Bit 6 This bit reflects real time status of the BDE pin (See Sections 3.2.1.1, 3.2.2.1, 3.2.3.1, 3.2.4.1, 3.2.5.1, or 3.2.6.1) for CRTCB/Sprite.

Bits 3:0 This value specifies the number of pixels by which the video data should be shifted to the left. Shifts of up to nine pixels are supported. Note that in 9-dot modes, a value of 8 signifies no shift, and the values of 0-7 signify shifts of 1-8 pixels, respectively.



HPP: Horizontal Pixel Position VPP: Vertical Pixel Position
WID: Width HGT: Height

CRTCB Window Positioning

5.6.16 CRTCB Color Depth Register (Index: EE)

I/O address = 21xB

Bit	Description	Access
7:6	Vertical zoom factor.	RW
5:4	Reserved (always set to 0).	
3:0	CRTCB bits/pixel select.	RW

Bit Description

Bits 7:6 The value of these two bits determine the number of times the line is repeated.

Bit 7 6	Times line repeated (no. of times each line is replicated)
0 0	1
0 1	2
1 0	3
1 1	4

Bits 3:0 This value selects the color depth for the CRTCB. Combinations for color depth beyond 16 bits/pixel are currently reserved.

Bit 3 2 1 0	Bit/pixel
0 0 0 0	1
0 0 0 1	2
0 0 1 0	4
0 0 1 1	8
0 1 0 0	16
↓↓↓↓	
1 1 1 1	Reserved.

5.6.17 CRTCB/Sprite Control (Index: EF)

I/O address = 21xB

Bit	Description	Access
7:3	Configuration read-back.	RO
2	Sprite size control.	RW
1	B pixel overlay/B pixel data output.	RW
0	CRTCB/Sprite select.	RW

Bit **Description**

7:3 These bits are used to read back the configuration status of the W32p as determined by FCG<4:0>.

Bit 2 When set to 1, sets sprite size to 128x128.

When set to 0, specifies sprite size at 64x64.

Bit 1 When set to 1, specifies that the CRTCB pixels overlay the CRTC (i.e., normally displayed) pixels.

When set to 0, the CRTCB pixel data (2-bit sprite) is output to the SP<1:0> pins.

Bit 0 When set to 1, selects the CRTCB functioning.

When set to 0, selects the sprite. See also Section 5.7.9, IMA Indexed Register F7, and TKN, SP pin descriptions in section 3.3.5.

5.7 IMA Register Descriptions

IMA Indexed Registers

Index	Register
F0	Image Starting Address Low
F1	Image Starting Address Middle
F2	Image Starting Address High
F3	Image Transfer Length Low
F4	Image Transfer Length High
F5	Image Row Offset Low
F6	Image Row Offset High
F7	Image Port Control

5.7.1 Indexed Addressing

The IMA registers are accessed using an indexed addressing scheme whereby a number selecting a register is first written to address 21xA (Index) and then the register can be read from or written to at address 21xB. The IMA registers use indices F0 through F7.

The Index Register at address 21xA is also used to address the CRTCB/Sprite Indexed Registers, see Section 5.6 for details.

The value of 'x' in the addresses 21xA and 21xB is determined by the logical value on the IOD<2:0> pins of the chip at power-up reset; see Section 3 for details.

5.7.2 IMA Indexed Register F0: Image Starting Address Low

I/O address = 21xB

Bit	Description	Access
7:0	Image starting address, bits <7:0>.	RW

5.7.3 IMA Indexed Register F1: Image Starting Address Middle

I/O address = 21xB

Bit	Description	Access
7:0	Image starting address, bits <15:8>.	RW

5.7.4 IMA Indexed Register F2: Image Starting Address High

I/O address = 21xB

Bit	Description	Access
7:4	Reserved.	
3:0	Image starting address, bits <19:16>.	RW

Bit	Description
Bits 19:0	These three registers define a 20-bit offset into display memory at which the image data is to be stored. The starting address is measured in doublewords, so if the data is to be stored at byte address 256, then a value of 64 (i.e., 256/4) should be programmed into these registers.

5.7.5 IMA Indexed Register F3: Image Transfer Length Low

I/O address = 21xB

Bit	Description	Access
7:0	Image width, bits <7:0>.	RW

5.7.6 IMA Indexed Register F4: Image Transfer Length High

I/O address = 21xB

Bit	Description	Access
7:4	Reserved.	
3:0	Image width, bits <11:8>.	RW

Bit	Description
Bits 11:0	These two registers define a 12-bit value which is the number of doublewords to be transferred per scan line. The value loaded should be one less than the desired number of doublewords.

5.7.7 IMA Indexed Register F5: Image Row Offset Low

I/O address = 21xB

Bit	Description	Access
7:0	Memory address offset, bits <7:0>.	RW

5.7.8 IMA Indexed Register F6: Image Row Offset High

I/O address = 21xB

Bit	Description	Access
7:4	Reserved.	
3:0	Memory address offset, bits <11:8>.	RW

Bit	Description
Bits 11:0	These two registers specify the number of doublewords between the start of one row of the stored image data and the start of the next row.

5.7.9 IMA Indexed Register F7: Image Port Control

I/O Address = 21xB

Bit	Description	Access
7	CRTCB/Sprite enable.	RW
6:5	MUX<1:0> output enable.	RW
4:2	Reserved.	
1	Image Port interlace transfer mode.	RW
0	Image Port enable.	RW

Bit **Description**

Bit 7 When set to 1, enables the CRTCB or the Sprite, whichever is selected in CRTCB/Sprite Indexed Register EF, bit 0.

Bits 6:5 These bits select signals to be output on MUX<1:0> pins. See Section 3.3.5 for a description of these shared pins.

Bit 1 When set to 1, enables odd/even interlace transfer.

When set to 0, enables linear interlace transfer. At reset, this bit has a value of 0.

Bit 0 When set to 1, enables the Image Port.

When set to 0, disables the Image Port. At reset, this bit has a value of 0.

5.8 MMU Register Descriptions

See Section 7.3 for the memory base address for the Memory-Mapped Registers. The offset in the table below is added to the base address to calculate the actual address of the register.

Memory Offset	Register
00	MMU Memory Base Pointer Register 0
04	MMU Memory Base Pointer Register 1
08	MMU Memory Base Pointer Register 2 ***Revision A ONLY***
13	MMU Control Register

5.8.1 MMU Memory Base Pointer Register 0

Memory offset = 00

Bit	Description	Access
31:22	Reserved.	
21:0	Memory base pointer.	RW

Bit Description

Bits 21:0 The base pointer defines the starting address in display memory of MMU aperture number 0. Since the Memory Base Pointers MUST be aligned to the doubleword, the least significant two bits are ignored, when writing and always return a value of 00, when reading.

5.8.2 MMU Memory Base Pointer Register 1

Memory offset = 04

Bit	Description	Access
31:22	Reserved.	
21:0	Memory base pointer.	RW

Bit Description

Bits 21:0 The base pointer defines the starting address in display memory of MMU aperture number 1. Since the Memory Base Pointers MUST be aligned to the doubleword, the least significant two bits are ignored and are always set to 0.

5.8.3 MMU Memory Base Pointer Register 2

Memory offset = 08 ***Revision A Register ONLY***

Bit	Description	Access
31:22	Reserved.	
21:0	Memory base pointer.	RW

Bit Description

Bits 21:0 The base pointer defines the starting address in display memory of MMU aperture number 2. Since the Memory Base Pointers MUST be aligned to the doubleword, the least significant two bits are ignored and are always set to 0. See Section 2.11.1 for a description of MMU Memory Base Pointer Register 2.

5.8.4 MMU Control Register

Memory offset = 13

Bit	Description	Access
7	Reserved.	
6:4	Linear Address Control (LAC).	RW
3	Reserved.	
2:0	Aperture type (APT).	RW

Bit **Description**

Bits 6:4 There is one Linear Address Control bit for each MMU aperture. Bit 6 of this register corresponds to MMU aperture 2, bit 5 to aperture 1, and bit 4 to aperture 0.

When set to 0, the memory is organized according to the current display mode.

When set to 1, the memory is organized in linear fashion. The effect is as if the following register modifications were made:

```
TS2<3:0>=1111
TS4<3>=1
GDC1<3:0>=0000
GDC3<4:0>=00000
GDC5<3>=0
GDC5<1:0>=00
GDC6<1>=0
GDC8<7:0>=11111111
```

Bits 2:0 There is one Aperture Type bit for each MMU aperture. Bit 2 of this register corresponds to MMU aperture 2, bit 1 to aperture 1, and bit 0 to aperture 0. This bit indicates whether an aperture is in "accelerated mode" or not.

When set to 0, access through this MMU aperture is routed through the GDC to display memory.

When set to 1, access through this MMU aperture is routed to the accelerator.

5.9 ACL Register Descriptions

See Section 7.3 for the memory base address for the Memory-Mapped Registers. The offset in the table below is added to the base address to calculate the actual address of the register. See also Table 5.0-2 for R/W information and register size.

Memory Offset	Register
	Non-Queued Registers
30	ACL Suspend/Terminate Register
31	ACL Operation State Register
32	ACL Sync Enable Register
33	ACL Write Interface Valid Bits **Except Revision A**
34	ACL Interrupt Mask Register
35	ACL Interrupt Status Register
36	ACL Accelerator Status Register
38	ACL X Position Register
3A	ACL Y Position Register
	Queued Registers
80	ACL Pattern Address Register
84	ACL Source Address Register
88	ACL Pattern Y Offset Register
8A	ACL Source Y Offset Register
8C	ACL Destination Y Offset Register
8E	ACL Pixel Depth Register
8F	ACL Direction Register
90	ACL Pattern Wrap Register
92	ACL Source Wrap Register
98	ACL X Count Register
9A	ACL Y Count Register
9C	ACL Routing Control Register
9E	ACL Background Raster Operation Register
9F	ACL Foreground Raster Operation Register
A0	ACL Destination Address Register
A4	ACL Mix Address Register
A8	ACL Mix Y Offset Register
AA	ACL Error Term
AC	ACL Delta Minor
AE	ACL Delta Major

5.9.1 ACL Suspend/Terminate Register

This is a non-queued register
Memory offset = 30

Bit	Description	Access
7:5	Reserved.	
4	Terminate Accelerator Operation (TO).	RW
3	Host Write Enable (HWE).	RW
2	Host Read Enable (HRE).	RW
1	Flush Accelerator Operation (FLO).	RW
0	Suspend Accelerator Operation (SO).	RW

Bits 3:1 are Reserved in Revision A

Bit Description

Bit 4 Used to terminate an Accelerator operation. To terminate an Accelerator operation, the programmer should write a 1 to this bit, wait for RDST (ACL Status Register, bit 1) to be 0, then write a 0 to this bit.

Termination returns the accelerator to its initial power-on state, with the ACL registers returned to the values that they contain after a reset of the chip. The programmer is advised to treat all accelerator registers as having undefined values after a termination, and reprogram all registers before initiating another accelerator operation.

Bit 3 When set to 1, allows the host to write data to the accelerator, independent of the programming of the ACL Routing Control Register. This bit is generally used during State Restore operations to restore queued data to the accelerator. During normal operation this bit should be set to 0.

At power-up reset or ACL termination (bit 4 of ACL Suspend/Terminate Register), this bit is set to 0.

Bit 2 When set to 1, allows the host to read data from the accelerator, independent of the programming of the ACL Routing Control Register. This bit is generally used during Suspend and State Save operations to extract queued data from the accelerator. During normal operation this bit should be set to 0.

At power-up reset or ACL termination (bit 4 of ACL Suspend/Terminate Register), this bit is set to 0.

Bit 1 Used to Flush the data from the accelerator pipeline when a Memory-to-Screen operation is suspended. To flush the pipeline, the programmer should toggle this bit from 0 to 1 to 0.

At power-up reset, this bit is set to 0.

Bit 0 Used to suspend an Accelerator operation. To suspend an Accelerator operation, the programmer should write a 1 to this bit, wait for ASUS (ACL Status Register, bit 6) [***Revision A*** - wait for RDST (ACL Status Register, bit 1) instead] to be 0, then write a 0 to this bit.

5.9.2 ACL Operation State Register

This is a non-queued register
Memory offset = 31

Bit	Description	Access
7:5	Reserved.	
4	Address Start Enable (ASEN).	RW
3	Resume Accelerator Operation (RMO).	WO
2:1	Reserved.	
0	Restore Accelerator Operation State (RSO).	WO

Bit	Description
Bit 4	When set to 1, a write to byte 3 (bits 31:24) of the ACL Destination Address Register will initiate an accelerator operation. When set to 0, a write to byte 3 (bits 31:24) of the ACL Destination Address Register will NOT initiate an accelerator operation. At power-up reset, this bit is set to 0.
Bit 3	When set to 1, a paused screen-to-screen accelerator operation is resumed. When set to 0, no action is taken.
Bit 0	When set to 1, the state in the accelerator's queue is transferred to the internal registers of the accelerator. When set to 0, no action is taken. It is not possible to set both of the above bits to 1 in a single write access in order to transfer the queued state into the accelerator and resume an accelerator operation; this must be done in two separate write accesses.

5.9.3 ACL Sync Enable Register

This is a non-queued register
Memory offset = 32

Bit	Description	Access
Bits <7:2> are Reserved in Revision A		
7	Sub-cycle Status. *Except Revisions A & B*	RO
6:5	Reserved.	
4	Force read/write OK.	RW
3	Reserved.	
2	Zero Wait-State Disable for Write.	RW
1	Zero Wait-State Enable for Read.	RW
0	Sync Enable.	RW

Bit Description

Bit 7 A value of 1 indicates that the accelerator has completed all internal sub-cycle operations for the current data read cycle.

A value of 0 indicates that the accelerator has been suspended in the middle of a sub-cycled read operation.

This bit is used by State Save/Restore software to aid in the computation of the backtrack amount when resuming an accelerator operation.

Bit 4 When set to 1, the accelerator DOES NOT ignore host accesses while the accelerator is idle.

When set to 0, the accelerator ignores host accesses to an accelerated MMU aperture when the accelerator is not programmed to be interacting with the host. This setting is provided as a safety feature to prevent a hung bus cycle (and therefore, system crash) if the host inadvertently accesses an accelerated MMU address space while the accelerator is idle.

It is recommended that this bit be set to 0 at all times. At power-up reset, this bit is set to 0.

Bit 2 When set to 1, the chip internally adds one extra SCLK clock cycle to writes to an accelerated MMU aperture and to the MMU and ACL Registers.

It is recommended that this bit be set to 0 at all times. At power-up reset, this bit is set to 0.

Bit 1 When set to 1, allows the host to perform zero wait-state reads from an accelerated MMU aperture and from the MMU and ACL Registers. The ACRO bit (ACL Routing Control Register, bit 6) is used to program an accelerated operation which returns data to the host.

When set to 0, a host read from these areas will have at least one wait-state. At power-up reset, this bit is set to 0.

Bit 0 When set to 1, indicates that a write to a full queue will be "wait-stated" until the queue becomes not full. If the current accelerator operation requires the host to read data, such a read will be "wait-stated" if the accelerator does not yet have any data available.

When set to 0, indicates that a write to a full queue will be ignored. If the current accelerator operation requires the host to read data, such a read will be ignored if the accelerator does not yet have any data available. It is possible to generate an interrupt when such a write or read is ignored; see Section 2.12.9.

5.9.4 ACL Write Interface Valid Bits

Except Revision A

This is a non-queued register
Memory offset = 33

Bit	Description	Access
7:4	Reserved.	
3:0	Write Interface Valid bits.	RW

Bit Description

Bits 3:0 These bits indicate the status of each byte in the write interface between the host and the accelerator. Bit 0 corresponds to byte 0 (data bus 7:0), bit 1 to byte 1 (data bus 15:8), etc.

When the bit value is set to 1, it indicates that the host has written to that byte, and the accelerator has not yet consumed that byte.

When the bit value is set to 0, it indicates that the host may write to that byte without incurring a large number of wait-states on that bus cycle.

This register should only be written to for State Save/Restore operation. In order to write to this register, either Host Write Enable (ACL Suspend/Terminate Register, bit 3) or Force Read/Write Ok (ACL Sync Enable Register, bit 4) should be set.

At power-up reset or ACL termination (bit 4 of ACL Suspend/Terminate Register), these bits are set to 0.

5.9.5 ACL Interrupt Mask Register

This is a non-queued register
Memory offset = 34

Bit	Description	Access
7	Master Interrupt Enable.	RW
6:4	Reserved.	
3	Read Fault Interrupt Enable.	RW
2	Write Fault Interrupt Enable.	RW
1	Read Interrupt Enable.	RW
0	Write Interrupt Enable.	RW

Bit Description

Bit 7 When set to 1, allows the cases below to generate interrupts.

When set to 0, the accelerator will not generate an interrupt under any circumstance, regardless of the setting of the remaining bits in this register. This allows the Interrupt Status Register to be used as a polling status register.

At power-up reset or ACL termination (bit 4 of ACL Suspend/Terminate Register), this bit is set to 0.

5.9.5 ACL Interrupt Mask Register (cont'd)

Bit **Description**

Bit 3 When set to 1, enables a Read Fault Interrupt when the host reads data from the accelerator when the accelerator has no data available (and ACL Sync Enable Register, bit 0 is 0). This is an EVENT-triggered interrupt; i.e., the interrupt line asserts when the faulting read occurs. The interrupt is cleared by a write of 1 to the Read Fault Interrupt Status bit (ACL Interrupt Status Register, bit 3).

When set to 0, this interrupt is disabled.

At power-up reset or ACL termination (bit 4 of ACL Suspend/Terminate Register), this bit is set to 0.

Bit 2 When set to 1, enables a Write Fault Interrupt to occur under two possible circumstances:
1. Write to a full queue.
2. A byte of data is written to the accelerator "out-of-order" in such a manner to cause a "deadlock" in the passing of data from the host to the accelerator. In order for any interrupt to be generated the "Sync Enable" bit must be off (ACL Sync Enable Register, bit 0 is 0). If the Sync Enable bit is on, a case 1 write will be "wait-stated", and a case 2 write will be ignored. This is an EVENT-triggered interrupt; i.e., the interrupt line asserts when the faulting write occurs. The interrupt is cleared by a write of 1 to the Write Fault Interrupt Status bit (ACL Interrupt Status Register, bit 2).

When set to 0, this interrupt is disabled.

Bit 2 At power-up reset or ACL termination (bit 4 of ACL Suspend/Terminate Register), this bit is set to 0.

Bit 1 When set to 1, enables a Read Interrupt when the queue is empty and the Accelerator goes from busy to idle. This is an EVENT-triggered interrupt; i.e., the interrupt line asserts when the accelerator goes from busy to idle. The interrupt is cleared by a write of 1 to the Read Interrupt Status bit (ACL Interrupt Status Register, bit 1).

When set to 0, this interrupt is disabled.

At power-up reset or ACL termination (bit 4 of ACL Suspend/Terminate Register), this bit is set to 0.

Bit 0 When set to 1, enables a Write Interrupt when the queue is not full. This is a STATE-triggered interrupt; i.e., the interrupt line is asserted while the queue is in the state of being "not-full." This interrupt is cleared by disabling it.

When set to 0, this interrupt is disabled.

At power-up reset or ACL termination (bit 4 of ACL Suspend/Terminate Register), this bit is set to 0.

5.9.6 ACL Interrupt Status Register

This is a non-queued register
Memory offset = 35

Bit	Description	Access
7:4	Reserved.	
3	Read Fault Interrupt Status.	RW
2	Write Fault Interrupt Status.	RW
1	Read Interrupt Status.	RW
0	Write Interrupt Status.	RO

Bit	Description
Bit 3	A value of 1 indicates that the current interrupt condition was caused by a Read Fault. When set to 1, clears the Read Fault Interrupt condition. When set to 0, the value of this bit is unaffected.
Bit 2	A value of 1 indicates that the current interrupt condition was caused by a Write Fault. When set to 1, clears the Write Fault Interrupt condition. When set to 0, the value of this bit is unaffected.
Bit 1	A value of 1 indicates that the current interrupt condition was caused by a Read Interrupt. When set to 1, clears the Read Interrupt condition. When set to 0, the value of this bit is unaffected.
Bit 0	A value of 1 indicates that the current interrupt condition was caused by a Write Interrupt. To clear the Write Interrupt, disable by setting bit 0 of the ACL Interrupt Mask Register to 0 (See Section 5.9.5).

5.9.7 ACL Accelerator Status Register

This is a non-queued register
Memory offset = 36

Bit	Description	Access
7	Accelerator Data Ready Status (ADST).	RO
6	Accelerator Pipeline Not Empty (APNE).	RO
5	Reserved.	
4	Accelerator Suspended Status (ASUS).	RO
3	Screen-to-Screen Status (SSO).	RO
2	XY Status (XYST).	RW
1	Read Status (RDST).	RO
0	Write Status (WRST).	RW

Bits <6:4> are Reserved in Revision A

Bit Description

Bit 7 A value of 1 indicates that the Accelerator has data available for the host to read. It is used during accelerator operations that have the Accelerator Data Routing bit (ACL Routing Control Register, bit 6) programmed so that the Destination data is routed back to the host.

Bit 6 A value of 1 indicates that the accelerator has new Destination Map data waiting in its pipeline to be written out to frame buffer memory or to the host (depending on the programming of the ACL Routing Control Register).

A value of 0 indicates that the accelerator pipeline is empty.

Bit 4 A value of 1 indicates that the accelerator is suspended. An accelerator operation is suspended by setting the SO bit (ACL Suspend/Terminate Register, bit 0, see section 5.9.1).

Bit 3 A value of 1 indicates that the current Accelerator operation is a screen-to-screen operation. This bit is only valid when bit 2 is 1. It is used by State-Restore software to determine if a write to the RMO bit (ACL Operation State Register, bit 3) is necessary.

Bit 2 When set to 1, indicates that the Accelerator is processing an X/Y block An "X/Y Block" means that the accelerator's internal XPOS, YPOS have not yet reached the terminal XCNT,YCNT for a given operation. Note that this bit must be restored when the state is restored for a suspended operation.

Bit 1 A value of 1 indicates that the Accelerator is busy (i.e., may be modifying display memory) or the queue is not empty.

A value of 0 indicates that the Accelerator is idle and the queue is empty, or the Accelerator is suspended. In other words, when this bit is 0, the host is guaranteed to read correct results from the display memory and from the accelerator's internal registers.

Bit 0 A value of 1 indicates that the accelerator's queue is full, and cannot accept any more host writes.

A value of 0 indicates that the accelerator's queue is not full and hence it is okay to write to a queued register or to an accelerated MMU aperture.

Note that this bit must be restored when the state is restored for a suspended operation.

NOTE: Since the host is reading the real-time status of the accelerator, it is possible for several of the status bits to be changing at the instant that the host reads them. For this reason, it is advisable for the programmer to test only ONE bit of this register each time it is read; if more than one bit is to be tested, multiple reads should be performed until consistent results are read.

5.9.8 ACL X Position Register

This is a non-queued register
Memory offset = 38

Bit	Description	Access
15:12	Reserved.	
11:0	X Position.	RW

Bit	Description
Bits 11:0	Reading this register returns the accelerator's internal X Position, indicating how far it has progressed through a given graphics operation. As the accelerator is running, this register is constantly changing; starting from zero and approaching the terminal value as programmed in the ACL X Count Register. The X and Y Position registers are used only for saving and restoring the Accelerator's current position when it is suspended in the middle of an operation.

5.9.9 ACL Y Position Register

This is a non-queued register
Memory offset = 3A

Bit	Description	Access
15:12	Reserved.	
11:0	Y Position.	RW

Bit	Description
Bits 11:0	See Section 5.9.8, ACL X Position Register for an explanation of this register.

QUEUED REGISTERS

Queued registers are found at memory offsets in the range 80 to FF. These registers are used to set up parameters for Accelerator operations.

5.9.10 ACL Pattern Address Register

This is a queued register
Memory offset = 80

Bit	Description	Access
31:22	Reserved.	
21:0	Pattern Address.	RW

Bit	Description
Bits 21:0	This value is the absolute address in display memory for the Pattern Map. It should be programmed to point to the first byte to be processed by a given accelerated graphics operation. See Section 2.12 for information on storage of the Pattern Map in display memory.

5.9.11 ACL Source Address Register

This is a queued register
Memory offset = 84

Bit	Description	Access
31:22	Reserved.	
21:0	Source Address.	RW

Bit	Description
Bits 21:0	This value is the absolute address in display memory for the Source Map. It should be programmed to point to the first byte to be processed by a given accelerated graphics operation.

5.9.12 ACL Pattern Y Offset Register

This is a queued register
Memory offset = 88

Bit	Description	Access
15:12	Reserved.	
11:0	Pattern Y Offset.	RW

Bit	Description
Bits 11:0	This value is the amount to be added to the accelerator's internal Pattern address pointer when going from one line to the next during Accelerator operations. The actual value programmed is one less than the desired number of bytes to be added. For example, if the Pattern Map is 8 bytes wide, a value of 7 should be programmed into this register.

5.9.13 ACL Source Y Offset Register

This is a queued register
Memory offset = 8A

Bit	Description	Access
15:12	Reserved.	
11:0	Source Y Offset.	RW

Bit	Description
Bits 11:0	This value is the amount to be added to the accelerator's internal Source address pointer when going from one line to the next during Accelerator operations. The actual value programmed is one less than the desired number of bytes to be added. For example, if the Source Map is 640 pixels wide, a value of 639 should be programmed into this register.

5.9.14 ACL Destination Y Offset Register

This is a queued register
Memory offset = 8C

Bit	Description	Access
15:12	Reserved.	
11:0	Destination Y Offset.	RW

Bit **Description**

Bits 11:0 This value is the amount to be added to the accelerator's internal Destination address pointer when going from one line to the next during Accelerator operations. The actual value programmed is one less than the desired number of bytes to be added. For example, if the Destination Map is 640 pixels wide, a value of 639 should be programmed into this register.

5.9.15 ACL Pixel Depth Register

This is a queued register
Memory offset = 8E

Bit	Description	Access
7:6	Reserved.	
5:4	Pixel Depth.	RW
3:0	Reserved.	

Bit **Description**

Bits 5:4 The Pixel Depth is used by the accelerator for LineDraw operations to determine how many bytes are to be processed for each pixel. It is encoded as follows:

Pixel Depth	Bytes Per Pixel
00	1
01	2
10	3
11	4

The Pixel Depth is also used for BitBlt operations to control how many bytes are processed with each step of the accelerator. The accelerator always processes 4 bytes at a time, unless the Pixel Depth is set to 10; in which case 3 bytes are processed at a time. Therefore, for maximum performance, avoid setting the Pixel Depth to 10 for BitBlt operations.

5.9.16 ACL Direction Register

This is a queued register
Memory offset = 8F

Bit	Description	Access
7	Graphics Opcode (OP).	RW
6	Reserved.	
5	Load Error Term From Queue (LETQ).	RW
4	LineDraw Algorithm Control.	RW
3	Reserved.	
2	Axial Direction.	RW
1	Y Direction.	RW
0	X Direction.	RW

Bit **Description**

Bit 7 The Graphics Opcode controls whether the accelerator is to perform a BitBlt or a LineDraw operation.

When set to 0, the accelerator performs a BitBlt operation.

When set to 1, the accelerator performs a LineDraw operation.

Bit 5 This bit is only used when a LineDraw operation is initiated; it is not used by BitBlt operations.

When set to 1, at the initiation of a LineDraw operation the accelerator loads its internal Error Term register from the ACL Error Term Register in the queue.

When set to 0, at the initiation of a LineDraw operation the accelerator loads its internal Error Term register with the value in the ACL Delta Major Register.

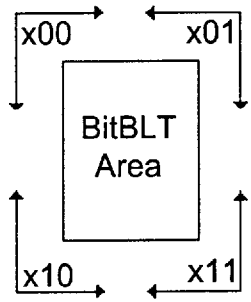
Generally, this bit is only set to 1 if the line is clipped; see Section 2.12.8.1 for a discussion of LineDraw operation.

Bit 4 This bit controls the internal LineDraw algorithm of the accelerator; it is not used by BitBlt operations.

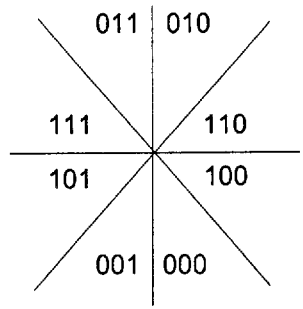
To draw a line compatible with Microsoft's Windows VGA driver, set this bit to be the same as the Y-Direction (bit 1 of this register); to draw a line compatible with IBM's XGA hardware, set this bit to the opposite of the Y-Direction bit.

Bits 2:0 These bits indicate in which direction the accelerated operation will proceed. The following figure summarizes the effect of various programmed values:

Encoding for BitBlt:



Encoding for LineDraw:



5.9.17 ACL Pattern Wrap Register

This is a queued register
Memory offset = 90

Bit	Description	Access
7	Reserved.	
6:4	Pattern Y Wrap (PYWR).	RW
3	Reserved.	
2:0	Pattern X Wrap (PXWR).	RW

Bit Description

Bits 6:4 Bits 2:0 The Pattern X Wrap and Pattern Y Wrap fields define an x-by-y tile size for the Pattern Map. If a Pattern Map has wrapping enabled in either the X or Y dimension, the base of the map must be stored in memory on a mod (X-wrap times Y-wrap) boundary. Naturally, if the X-Wrap is set to "No wrap", the boundary restriction is lifted. The ACL Pattern Address Register can then be loaded with the address of the byte which is to be anchored to the first byte of the BLT (or Line). Three combinations of X- and Y-wrap values are not allowed:

X-wrap	Y-wrap
"No wrap"	2
"No wrap"	4
"No wrap"	8

The Source map has wrap control registers that are identical to the Pattern.

Pattern X Wrap	Horizontal Wrap Length
000	Reserved
001	Reserved
010	4-byte
011	8-byte
100	16-byte
101	32-byte
110	64-byte
111	No wrap

Pattern Y Wrap	Vertical Wrap Length
000	1-line
001	2-line
010	4-line
011	8-line
100	Reserved
101	Reserved
110	Reserved
111	No wrap

5.9.18 ACL Source Wrap Register

This is a queued register
Memory offset = 92

Bit	Description	Access
7	Reserved.	
6:4	Source Y Wrap (SYWR).	RW
3	Reserved.	
2:0	Source X Wrap (SXWR).	RW

Bit	Description
Bits 6:4 Bits 2:0	See Section 5.9.17, ACL Pattern Wrap Register for an explanation of this register.

5.9.19 ACL X Count Register

This is a queued register
Memory offset = 98

Bit	Description	Access
15:12	Reserved.	
11:0	X Count.	RW

Bit	Description
Bits 11:0	This value specifies the number of bytes in the X dimension on which the accelerator should operate. The value programmed into this register is dependent upon the value in the ACL Pixel Depth Register (see Section 5.9.15). The formula for converting from "number of pixels" to "X Count" is as follows: X Count = (Number_Of_Pixels - 1) * Bytes_Per_Pixel For example, if the Pixel Depth field of the ACL Pixel Depth Register is programmed to be 01 (2 bytes per pixel), and the programmer wishes to operate on 6 pixels in the X dimension, then the X Count should be loaded with (6 - 1) * 2 = 10. Note that the accelerator will actually operate on 12 bytes in this case.

5.9.20 ACL Y Count Register

This is a queued register
Memory offset = 9A

Bit	Description	Access
15:12	Reserved.	
11:0	Y Count.	RW

Bit	Description
Bits 11:0	This value specifies the number of lines in the Y dimension on which the accelerator should operate. The Y Count should be programmed to one less than the desired number of lines to be operated on.

5.9.21 ACL Routing Control Register

This is a queued register
Memory offset = 9C

Bit	Description	Access
7	Invalidate disable.	RW
6	Routing of Accelerator Data (ACRO).	RW
5	Reserved.	
4	Enable Transparency (ENXP). ***Reserved - Revision A***	RW
3	MixMap Enable (MXEN).	RW
2	Reserved.	
1:0	Routing of CPU data (DARO).	RW

Bit	Description
Bit 7	When set to 1, the multiport cache is not invalidated at the end of the accelerator operation. When set to 0, the multiport cache is invalidated at the end of the accelerator operation. It is recommended that this bit always be programmed to 0.
Bit 6	This bit controls whether the resulting data from the accelerator operation is routed to the display memory or to the host CPU. When set to 1, accelerator data is routed to the host CPU. When set to 0, accelerator data is routed to display memory.

(continued on next page)

5.9.21 ACL Routing Control Register (cont'd)

Bit **Description**

Bit 4 When set to 1, the MixMap is used as a "byte-enable" mask to control which bytes of the Destination Map will be modified. The MixMap is used in this way only if the Background Raster Operation is set to AA ("no-operation") and the Foreground Raster Operation is not a function of the Destination Map. This method of operation allows for faster drawing of transparent BitBlt's since it avoids reading the Destination Map data from the frame buffer.

This bit may be set to 1 at all times; the only need to set it to 0 would be if the host wished to read Destination

When set to 1, the Mix Map data is taken from the CPU if the DARO field is set to 10. Otherwise, the Mix Map data is taken from the display memory, using the ACL Mix Address Register to determine the location of the map in display memory.

When set to 0, the Mix Map data is taken from the CPU if the DARO field is set to 10. Otherwise, the Mix Map is not used (as if Mix data were fixed to "1"), and the accelerator applies the Foreground Raster Operation to all bytes of the operation.

Bits 1:0 Routing of CPU data:

DARO	
00	CPU data not used
01	CPU data is Source data
10	CPU data is Mix Data
11	Reserved

The table below summarizes the data routing capabilities of the accelerator, under control of the MXEN and DARO fields:

MXEN	DARO	CPU Data Routing	Mix Data Routing
0	00	CPU data not used	Mix Data fixed to "1"
0	01	CPU data is Source data	Mix Data fixed to "1"
0	10	CPU data is Mix data	Mix Data from CPU
0	11	Reserved	Mix Data fixed to "1"
1	00	CPU data not used	Mix Data from display memory
1	01	CPU data is Source data	Mix Data from display memory
1	10	CPU data is Mix Data	Mix Data from CPU
1	11	Reserved	Mix Data from display memory

5.9.22 ACL Background Raster Operation Register

This is a queued register
Memory offset = 9E

Bit	Description	Access
7:0	Background Raster Operation (BGR).	RW

Bit	Description
Bits 7:0	This is the logical operation between Source, Pattern, and Destination bytes used when the Mix data has a value of 0. Each bit of the Mix Map selects whether the Background (0) or Foreground (1) Raster Operation is applied to the corresponding byte of the graphics operation. See Appendix A, ET4000/W32p (Microsoft) Raster Operations Codes and Definitions.

5.9.23 ACL Foreground Raster Operation Register

This is a queued register
Memory offset = 9F

Bit	Description	Access
7:0	Foreground Raster Operation (FGR).	RW

Bit	Description
Bits 7:0	This is the logical operation between Source, Pattern, and Destination bytes used when the Mix data has a value of 1. Each bit of the Mix Map selects whether the Background (0) or Foreground (1) Raster Operation is applied to the corresponding byte of the graphics operation. See Appendix A, ET4000/W32p (Microsoft) Raster Operations Codes and Definitions.

5.9.24 ACL Destination Address Register

This is a queued register
Memory offset = A0

Bit	Description	Access
31:22	Reserved.	
21:0	Destination Address (DA).	RW

Bit	Description
Bits 21:0	This value is the absolute address in display memory for the Destination Map. It should be programmed to point to the first byte to be processed by a given accelerated graphics operation.

5.9.25 ACL Mix Address Register

This is a queued register
Memory offset = A4

Bit	Description	Access
31:25	Reserved.	
24:0	Mix Address (MA).	RW

Bit	Description
Bits 24:0	This value is the absolute address in display memory for the Mix Map. Since the Mix Map has one bit for each byte processed in any other map, this register should be programmed to point to the first *bit* to be processed by a given accelerated graphics operation.

5.9.26 ACL Mix Y Offset Register

This is a queued register
Memory offset = A8

Bit	Description	Access
15:12	Reserved.	
11:0	Mix Y Offset.	RW

Bit	Description
Bits 11:0	This value is the amount to be added to the accelerator's internal Mix address pointer when going from one line to the next during Accelerator operations. The actual value programmed is one less than the desired number of bits to be added. For example, if the Mix Map is 14 bits wide, a value of 13 should be programmed into this register.

5.9.27 ACL Error Term Register

This is a queued register
Memory offset = AA

Bit	Description	Access
15:0	Error Term (ET).	RW

Bit	Description
Bits 15:0	This register is used to initialize or restore the Bresenham Error Term for the accelerator's internal LineDraw Engine. The LETQ bit of the ACL Direction Register (See Section 5.9.16) controls whether the accelerator's LineDraw engine initializes its internal error term from this register, or calculates it automatically from the ACL Delta Major Register. This register is ignored by BitBlt operations.

5.9.28 ACL Delta Minor Register

This is a queued register
Memory offset = AC

Bit	Description	Access
15:12	Reserved.	
11:0	Delta Minor (DMIN).	RW

Bit	Description
Bits 11:0	This register is loaded with the distance between endpoints on a line along the minor axis of the line. The minor axis of the line is defined as the dimension along which the endpoints are the closest together. In other words, if delta-X has a higher magnitude than delta-Y, then Y is the minor axis for that line, and this register is loaded with the absolute value of delta-Y (i.e., |y2 - y1|).

5.9.29 ACL Delta Major Register

This is a queued register
Memory offset = AE

Bit	Description	Access
15:12	Reserved.	
11:0	Delta Major (DMAJ).	RW

Bit	Description
Bits 11:0	This register is loaded with the distance between endpoints on a line along the major axis of the line. The major axis of the line is defined as the dimension along which the endpoints are the farthest apart. In other words, if delta-X has a higher magnitude than delta-Y, then X is the major axis for that line, and this register is loaded with the absolute value of delta-X (i.e., |x2 - x1|).

6. Board-Level Design Considerations

6.1 Bus Design Configurations

The ET4000/W32p is capable of having its function and features changed by a series of definition pins, FCG<4:0>, that are checked during initialization. While 32 possibilities exist, functionally only five are practical to the designer. The following defines options, the value of these options, and the design consideration in choosing configurations.

Feature Choices

Bus type. The chip will be a PCI design, or other local bus. If defined as PCI, the chip will assume the look and actions of a PCI device. Otherwise, the device will become a generic local bus peripheral similar and may be used in VL-bus or proprietary local bus designs. Bus type is determined by the status of the W32p's PCI/LOC pin (LBPI).

Address/data buffers. The W32p was designed to reduce or eliminate the need for external address/data buffer chips, to minimize cost, board space and inventory requirements.

Pixel data bus. The bus between the W32p and the RAMDAC can be defined as the standard 8-bit bus, or new 16-bit bus. The latter will allow 1280x1024 non-interlaced 72Hz display at 8 bits per pixel ; 1024x768, 16 bits per pixel; and 800x600, 24 bits per pixel. Additionally, the 16-bit path will allow high-resolution graphics display modes with mixed 8-bit and 16-bit windows. Some degree of pixel mixing will be possible with the 8-bit path, but resolution will be limited.

Feature connector control. The W32p will allow a feature connector to be implemented without external buffers on the pixel bus by including feature connector control pins. This reduces logic and cost from devices that include a feature connector.

Image Memory Access (IMA) port. The W32p has a functional 8-bit IMA port, which is essentially an asynchronous direct memory access port into the W32p's video DRAM. The bandwidth of the IMA is 40MB/sec.

Linear address internal/external decode. The PCI/386/486 32-bit data bus can be fully decoded by the W32p. In some configurations, the designer may opt to not decode all 32 bits (4 gigabyte space). To use the linear addressing modes of the W32p, special drivers are required to define and utilize the flat memory space used for video. In cases that do not provide full 32-bit linear addressing, the W32p can have external decode circuitry to increase the space to 32 bits.

CS3/CS4. Tseng graphics chips include a number of chip select pins that can be programmed via the internal registers. In some configurations the CS3 and CS4 pins may be shared with other functions. CS3 may be shared with the SYNR input or TKN<0> output. CS4 may be shared with the SWSE (switch sense) input or TKN<1> output.

Generic Configurations

PCI. 32-bit linear addressing to the CPU, 8/16-bit pixel bus defined by the BIOS during initialization. An 8-bit IMA port and the feature connector control are included. CS3/CS4 are always available.

LBB (Local Buffered Bus). A local bus design with all of the above features, but using the same configuration pin ID may be created. To use all of the above features on a non-PCI local bus, address/data buffers must be added. CS3/CS4 are shared.

Local Bus option 1. The W32p decodes a 29-bit (512MB) space on the local bus, with no address/data buffers. A programmable 8/16-bit pixel data bus, and feature connector control are included.

No IMA port is available. CS3 and CS4 are shared.

Local Bus Option 2. Full 32-bit linear addressing (4 gigabyte), local bus with no address/data buffers. Feature connector control is included. The pixel data bus is 8 bits.

6.2 Local Bus

A series of definition pins (see Section 3.3.1 FCG<4:0> determine the function and features of the W32p. The following five are the most practical:

F10011 = Local bus with external decode; no buffers; 16-bit pixel bus; no image port; feature connector control.

This configuration is for Local Bus with up to 512Mbytes of direct Address decode. If a full 4 gigabyte decode is required, then external Address decode into SEGI* is required. This configuration does not require any external Address or Data buffers, and has a 16-bit DAC interface as well as support for direct attachment of the W32p pixel port to the feature connector. This configuration is intended for designs created for lowest cost with extended SuperVGA[1] video modes when a feature connector is required.

F01000 = Local bus; no buffers; 8-bit pixel bus; no image port; feature connector control.

This configuration is for Local Bus with full 4GB Address decoding. It does not require external buffers for Address or Data. It has control for direct attachment of the W32p pixel port to the feature connector. This configuration is intended for designs created for lowest cost if extended SuperVGA[1] video is not required.

F01011 = Local bus; no buffers; 16-bit pixel bus; no image port; no feature connector control.

This configuration is for Local Bus with full 4GB Address decoding. It does not require external buffers for Address or Data and has a 16-bit DAC interface. This configuration is intended for designs created for lowest cost with extended SuperVGA[1] video modes when a feature connector is not required.

F10001 = Local bus with external decode; no buffers; 8-bit pixel bus; 8-bit image port; feature control.

This configuration is for Local Bus and requires external decoding of the Address into SEGI*. This configuration does not require any external Address or Data buffers, and has a 8-bit DAC interface as well as support for direct attachment of the 8-bit W32p Image Port. This configuration is intended for designs created for lowest cost when an IMA port is required and extended SuperVGA modes and feature connector are not required.

F11100 = LBB pin configuration; all functions; multiplexed host address bus.

This is the highest cost of the Local Bus configurations because it requires external multiplexing of the Address and Data buses. This configuration should be used when IMA and extended SuperVGA[1] video modes are required on the same design. Additionally, this configuration includes support for feature connector control. Address Data multiplexing typically requires eight F245 devices.

[1] = Extended SuperVGA modes. The following modes are available with a 16-bit DAC interface only:

1280x1024	256K colors	non-interlaced
1024x768	32K/65K colors	non-interlaced
800x600	16.8 million colors	non-interlaced

Feature Choices

See Section 3.1.2 Configuration-Specific Definitions.

6.2.1 Local Bus PORI

The following inputs provide the general Local Bus configuration:

LBPI = low PCI Bus
= high Local Bus

SNPE = low External RAMDAC snoop enabled
= high External RAMDAC snoop disabled

IOD<2:0> = x CRTCB/Sprite I/O register map for I/O addresses 21xA, 21xB where x = 0,1,2..,7. IOD<2:0> also defines VGA I/O address mapping as follows:

IOD<2:0>	CRTCB	CRTC	GDC	ATC	TS	Ext. Palette RAM
000	210A/B	73D4/5	73CE/F	73C0	73C4/5	73C6-73C9
001	211A/B	63D4/5	63CE/F	63C0	63C4/5	63C6-63C9
010	212A/B	53D4/5	53CE/F	53C0	53C4/5	53C6-53C9
011	213A/B	43D4/5	43CE/F	43C0	43C4/5	43C6-43C9
100	214A/B	33D4/5	33CE/F	33C0	33C4/5	33C6-33C9
101	215A/B	23D4/5	23CE/F	23C0	23C4/5	23C6-23C9
110	216A/B	13D4/5	13CE/F	13C0	13C4/5	13C6-13C9
111	217A/B	03D4/5	03CE/F	03C0	03C4/5	03C6-03C9

DVCK Divided clock/PCI ROM select: ***Except for Revisions A & B in PCI mode*** this signal is used to enable the PCI ROM address bus function of DD<7:0>. In local bus mode, this PORI bit controls DVCK *only* in all revisions.
Low : LCLK = BCLK
High: LCLK = BCLK/2

Except Revisions A & B Low (PCI bus mode): Disables PCI ROM address function of DD<7:0>. PCI ROM address derived from AD<14:2>.
High (PCI bus mode: Enables PCI ROM address function of DD<7:0>.

DELC Command delay. = low W32p internal command starts at first T2 after ADS* goes from low to high.
= high W32p internal command starts at first T2 of the CPU state.

DISB = low Enables BIOS decode.
= high Disables BIOS decode.

DISB = low ROME* is generated for BIOS ROM residing on the W32p's DD<7:0> data bus. ROME* will be generated multiple times per CPU access cycle depending on the BE<3:0>* signals. Also, LOCAL* and READY* will be generated.
= high ROME* is generated based on address decode only. LOCAL* and READY* will not be generated. ROME* is intended to be used as the chip select for the BIOS ROM which resides on an external bus (such as ISA bus).

6.2.2 Local Bus Special Note

Clock Synchronization

All ET4000/W32p bus interface timings are synchronized with LCLK which is derived from the BCLK input. If BCLK is twice the frequency of LCLK (the internal Local Bus Clock) proper RESET to BCLK setup timing and phasing is required to ensure that both the processor and the ET4000/W32p clocks are in phase.

Local Bus Cycle

All ET4000/W32p internal bus cycles are initiated by ADS* low while LCLK changes from low to high. If an appropriate address is decoded by the ET4000/W32p, it will assert the LOCAL* signal and an internal Local Bus Command (LBCMD) will be generated. The READY* signal is asserted by the ET4000/W32p at the last LCLK cycle of LBCMD. If external ready re-synchronization is required as in VESA VL-Bus, neither the ET4000/W32p nor the processor will terminate the command cycle until RDYRTN* is asserted by an external device.

External Palette Memory DAC

The ET4000/W32p provides byte read or write support for the external palette RAMDAC. If a 16-bit I/O instruction is detected to the external palette RAMDAC, the ET4000/W32p will generate two separate 8-bit I/O PMEW* or PMER* cycles to the external palette RAMDAC. The RS<1:0> register select address to the RAMDAC is provided by the ET4000/W32p on RS<1:0>. Palette RAMDAC read or write data are interfaced via DD<7:0>.

Write Cycle: During a RAMDAC write cycle, all 32 bits of CPU write data are latched internally by the ET4000/W32p. The ET4000/W32p then uses the address to determine the lowest byte within that doubleword. The lower two bits of that byte address are output onto RS<1:0>. Simultaneously, the data corresponding to that byte address is output onto DD<7:0> and PMEW* is activated for 3 LCLKs. The ET4000/W32p goes idle for 14 LCLKs and repeats the previous operation with the byte address incremented by one. This continues until the last byte of valid data is written to the RAMDAC. The ET4000/W32p asserts READY* low after the last byte of valid data is transferred.

Read Cycle: During a RAMDAC read cycle, the W32p first uses the address to determine the lowest byte within that doubleword. The lower two bits of that byte address are output onto RS<1:0>, at which time PMER* is asserted for 4 LCLKs. The RAMDAC will output the data corresponding to that byte address onto DD<7:0>. The W32p will de-assert PMER* for 13 LCLKs and will simultaneously latch the data that is on DD<7:0>. This continues until the last byte of valid data has been read from the RAMDAC. The W32p will then drive the latched data onto AD<31:0>, issue READY*, and continue the drive until RDYRTN* is received.

Local Bus ROM

Read Cycle: During a ROM read cycle, the W32p first uses the address to determine the lowest byte within that doubleword. The lower two bits of that byte address are output onto RS<1:0>, at which time the W32p will assert ROME* for 8 clocks. The ROM will output the data corresponding to that byte address onto DD<7:0>. The W32p will de-assert ROME* and will simultaneously latch data that is on DD<7:0>. After 5 LCLKs it will re-assert ROME* if more data transfers are required. This continues until the last byte of valid data has been read from the ROM. The W32p will then drive the latched data onto AD<31:0>, issue READY*, and continue the drive until RDYRTN* is received.

NOTE: Problems will occur if there is an attempt to shadow the local bus BIOS. It is therefore recommended that the ROM be placed on the ISA bus with external address decoding to avoid difficulties that occur when the LOCAL* signal is generated during shadowed cycles.

External Memory-Mapped Registers

These registers include the capability to write to all 16 bits, though only the lower 8 will be valid during read cycles. All operations to the registers must be 16-bit using *only* word reads and writes. An operation to the external memory-mapped registers takes place when IXMAD goes active while RS<1> will indicate whether the operation is read or write. See section 7.3 for external memory-mapped registers to determine the addresses during which IXMAD is active.

Write: Data being written is latched internally to the W32p. Address and data are multiplexed onto DD<7:0>. The format is address low, data low; address high, data high. During each of these four phases IXMAD will go active for 1 LCLK and then inactive for 2 LCLKs. During the data phases RS<0> is high; during address phases RS<0> is low. RS<1> is low during this entire cycle.

Read: The data is driven by and external device. Address and data are multiplexed onto DD<7:0>. The format is address low, dummy cycle, address high, data low. During each of these four phases IXMAD will go active for 1 LCLK and then inactive for 2 LCLKs. During the data phases RS<0> is high; during address phases RS<0> is low. RS<1> is high during this entire cycle.

Segment Address Comparator

CRTC Indexed Register 30 bits <7:0> are compared with A<29:22> inputs to allow memory access. In configurations where A<31:30> exist, they are always decoded when low. Note that A<31:22> are not available in every configuration. In these cases as many of the addresses that exist on the W32p in these configurations are decoded as illustrated in the following table.

CRTC 30	7	6	5	4	3	2	1	0
F10011	0	SEGI	A<27>	A<26>	A<25>	A<24>	A<23>	A<22>
F01000	A<29>	A<28>	A<27>	A<26>	A<25>	A<24>	A<23>	A<22>
F01011	A<29>	A<28>	A<27>	A<26>	A<25>	A<24>	A<23>	A<22>
F10001	0	0	0	0	0	0	SEGI	A<22>
F11100	AD<29>	AD<28>	AD<27>	AD<26>	AD<25>	AD<24>	AD<23>	AD<22>

6.3 PCI Bus

A series of definition pins (see Section 3.3.1 FCG<4:0> determine the function and features of the W32p. The following is the only configuration necessary:

F11100 = PCI pin configuration (see Section 3.2 Pin Descriptions).
Features 16-bit pixel bus, image port, feature connector control.

Only one configuration exists for PCI which supports all available features without the addition of external glue logic.

External Palette Memory DAC

The ET4000/W32p provides byte read or write support for the external palette RAMDAC. If a 16-bit I/O instruction is detected to the external palette RAMDAC, the ET4000/W32p will generate two separate 8-bit I/O PMEW* or PMER* cycles to the external palette RAMDAC. The RS<1:0> register select address to the RAMDAC is provided by the ET4000/W32p on RS<1:0>. Palette RAMDAC read or write data are interfaced via DD<7:0>.

Write Cycle: During a RAMDAC write cycle, all 32 bits of CPU write data are latched internally by the ET4000/W32p. The ET4000/W32p then uses the address to determine the lowest byte within that doubleword. The lower two bits of that byte address are output onto RS<1:0>. Simultaneously, the data corresponding to that byte address is output onto DD<7:0> and PMEW* is activated for 3 LCLKs. The ET4000/W32p goes idle for 14 LCLKs and repeats the previous operation with the byte address incremented by one. This continues until the last byte of valid data is written to the RAMDAC. The ET4000/W32p asserts TRDY* low after the last byte of valid data is transferred.

Read Cycle: During a RAMDAC read cycle, the W32p first uses the address to determine the lowest byte within that doubleword. The lower two bits of that byte address are output onto RS<1:0>, at which time PMER* is asserted for 4 LCLKs. The RAMDAC will output the data corresponding to that byte address onto DD<7:0>. The W32p will de-assert PMER* for 13 LCLKs and will simultaneously latch the data that is on DD<7:0>. This continues until the last byte of valid data has been read from the RAMDAC. The W32p will then drive the latched data onto AD<31:0>, issue TRDY*.

PCI ROM

Read Cycle: During a ROM read cycle, the W32p first uses the address to determine the lowest byte within that doubleword. The lower two bits of that byte address are output onto RS<1:0>, at which time the W32p will assert ROME* for 8 clocks. The ROM will output the data corresponding to that byte address onto DD<7:0>. The W32p will de-assert ROME* and will simultaneously latch data that is on DD<7:0>. After 5 LCLKs it will re-assert ROME* if more data transfers are required. This continues until the last byte of valid data has been read from the ROM. The W32p will then drive the latched data onto AD<31:0>, issue READY*, and continue the drive until RDYRTN* is received.

NOTE: Because the PCI address is not available through the entire cycle, it must be externally latched with transparent latches using the ADDL signal.

Except Revisions A & B: In order to remove the extra input load due to external latches on the PCI's AD bus, the DD<7:0> bus may be used to generate the ROM address. For this to occur the DVCK PORI must be high. If the DVCK PORI is low, ADDL latches the PCI's AD bus as described above. When the DD bus is used to generate ROM address, the DD<7:2> is latched into ROM address <7:2> on the high-to-low transition of ADDL. DD<6:0> is latched into ROM address <14:8> on the high-to-low transition of ROME*, which also enables the ROM. RS<1:0> are directly connected to ROM address <1:0>.

External Memory-Mapped Registers

These registers include the capability to write to all 16 bits, though only the lower 8 will be valid during read cycles. All operations to the registers must be 16-bit using *only* word reads and writes. An operation to the external memory-mapped registers takes place when IXMAD goes active while RS<1> will indicate whether the operation is read or write. See section 7.3 for external memory-mapped registers to determine the addresses during which IXMAD is active.

Write: Data being written is latched internally to the W32p. Address and data are multiplexed onto DD<7:0>. The format is address high, data high; address low, data low. During each of these four phases IXMAD will go active for 1 LCLK and then inactive for 2 LCLKs. During the data phases RS<0> is high; during address phases RS<0> is low. RS<1> is low during this entire cycle.

Read: The data is driven by and external device. Address and data are multiplexed onto DD<7:0>. The format is address low, dummy cycle, address high, data low. During each of these four phases IXMAD will go active for 1 LCLK and then inactive for 2 LCLKs. During the data phases RS<0> is high; during address phases RS<0> is low. RS<1> is high during this entire cycle.

Segment Address Comparator

CRTC Indexed Register 30 bits <7:0> are compared with A<29:22> inputs to allow memory access. As illustrated in the following table, A<31:30> are always decoded when low. ***Except Revisions A & B:*** A<31:30> are compared with bits <31:30> of the configuration space Base Address Register. When the system linear bit is enabled (CRTC Indexed Register 36 bit <4> = 1).

CRTC 30	7	6	5	4	3	2	1	0
F11100	AD<29>	AD<28>	AD<27>	AD<26>	AD<25>	AD<24>	AD<23>	AD<22>

6.4 Display Memory Design Considerations

The ET4000/W32p provides a flexible interface to standard dynamic RAMs (DRAMs) for Display Memory. The Display Memory is organized into two separately-addressable banks, with a data bus width of one or two bytes per bank.

Functionally, the memory interface consists of:

1. Address:	Two multiplexed address buses (one for each bank)	AA<9:0>, AB<9:0>
2. Data:	One 32-bit data bus	MD<31:0>
3. Control:	Two Row Address Strobes (one for each bank)	RASA*, RASB*
	Eight Column Address Strobes (one for each byte)	CAS<7:0>*
	Two Memory Writes (one for each bank)	MWA*, MWB*
Except Revisions A & B		
	Four OE control signals (optional)	OEOB*, OEOA*, OEEB*, OEEA*

6.4.1 Memory Type and Upgrade Considerations

The table below shows the DRAM connection and the resultant total display memory size:

Display Memory Configuration Table

SIZE	CONFIGURATION	ADDRESS	DATA	RAS	CAS	WE	OE(1)
BYTE 3							
512KB	Non-interleave	NC	NC	NC	NC	NC	NC
1MB	Non-interleave	AB<8:0>	MD<31:24>	RASB	CAS<3>	MWB	NC
2MB	Non-interleave	AB<9:0>	MD<31:24>	RASB	CAS<3>	MWB	NC
4MB	Non-interleave	AB<9:0>	MD<31:24>	RASB	CAS<3>	MWB	NC
2MB	Interleave	AB<8:0>	MD<31:24>	RASB	CAS<3>	MWB	OEEB
					CAS<7>		OEEB
BYTE 2							
512KB	Non-interleave	AB<8:0>	MD<23:16>	RASB	CAS<2>	MWB	OEEB
1MB	Non-interleave	AB<8:0>	MD<23:16>	RASB	CAS<2>	MWB	OEEB
2MB	Non-interleave	AB<9:0>	MD<23:16>	RASB	CAS<2>	MWB	OEEB
4MB	Non-interleave	AB<9:0>	MD<23:16>	RASB	CAS<2>	MWB	OEEB
2MB	Interleave	AB<8:0>	MD<23:16>	RASB	CAS<2>	MWB	OEEB
					CAS<6>		OEOB
BYTE 1							
512KB	Non-interleave	NC	NC	NC	NC	NC	NC
1MB	Non-interleave	AA<8:0>	MD<15:8>	RASA	CAS<1>	MWA	OEEA
2MB	Non-interleave	AA<9:0>	MD<15:8>	RASA	CAS<1>	MWA	OEEA
4MB	Non-interleave	AA<9:0>	MD<15:8>	RASA	CAS<1>	MWA	OEEA
2MB	Interleave	AA<8:0>	MD<15:8>	RASA	CAS<1>	MWA	OEEA
					CAS<5>		OEOA
BYTE 0							
512KB	Non-interleave	AA<8:0>	MD<7:0>	RASA	CAS<0>	MWA	OEEA
1MB	Non-interleave	AA<8:0>	MD<7:0>	RASA	CAS<0>	MWA	OEEA
2MB	Non-interleave	AA<9:0>	MD<7:0>	RASA	CAS<0>	MWA	OEEA
4MB	Non-interleave	AA<9:0>	MD<7:0>	RASA	CAS<0>	MWA	OEEA
2MB	Interleave	AA<8:0>	MD<7:0>	RASA	CAS<0>	MWA	OEEA
					CAS<4>		OEOA

(1) OE is only necessary when using OE controlled interleave DRAM Interface. See Section 6.4.2.1 OE Controlled Interleave DRAM Interface.

An example of the minimum configuration consists of four 256Kx4 devices for a total display memory size of 512KB. An example of the maximum non-interleaved configuration consists of eight 1MBx4 devices for a total display memory size of 4MB. An example of the interleaved configuration consists of 16 256Kx4 devices for a total display memory size of 2MB.

6.4.2 Memory Resource Considerations

The system designer must consider several parameters when designing the display memory subsystem. The primary consideration is the types of display modes the video subsystem will support. This choice affects the amount of display memory required as well as the operating speed of the memory.

Other considerations involve the other chip features that will be utilized. For example, if the Image Port is used to display full-motion video, there are definite criteria that must be met to provide satisfactory performance. Also, certain memory configurations will allow maximum performance of the Graphics Accelerator.

This section presents guidelines for evaluating different memory configurations to meet required system specifications.

There are four primary requesters of the memory:

1. CRT Controllers (CRTC and CRTCB)
2. Host
3. Accelerator
4. Image Port

Technically, the Memory Refresh Controller is also a requester, but since it requires so little memory bandwidth, it is omitted from this discussion.

Fundamentally, the memory subsystem must be designed to meet the demands of all requesters at all times. First we will discuss the "supply-side" of the memory subsystem, made up of the display memory and its connection to the ET4000/W32p. The amount of memory bandwidth that can be supplied varies with two things: the memory configuration, and the SCLK rate.

The primary aspect of the memory configuration that affects performance is the width of the display memory data bus (MD bus). This parameter defines the number of bits that the ET4000/W32p will be able to access in a given SCLK period. The SCLK rate is chosen to match the RAS/CAS timing requirements of the DRAM.

The primary goal of the ET4000/W32p is to maximize the number of page-mode cycles to the DRAMs while servicing the memory requests of the various modules in the video subsystem. To emphasize the importance of page-mode DRAM cycles, the following graph (Figure 6.3.2-1) of memory bandwidth versus number of page-mode transfers for a DRAM cycle is provided.

The graph is for a memory configuration of:

- SCLK is 50MHz. (T_{cyc} = 20 ns)
- Random access time is 7 SCLK cycles (T_{ran} = 140 ns)
- Page-mode access time is 2 SCLK cycles (T_{pg} = 40 ns)
- Data Bus is 32 bits wide. (W_{db} = 4 bytes)

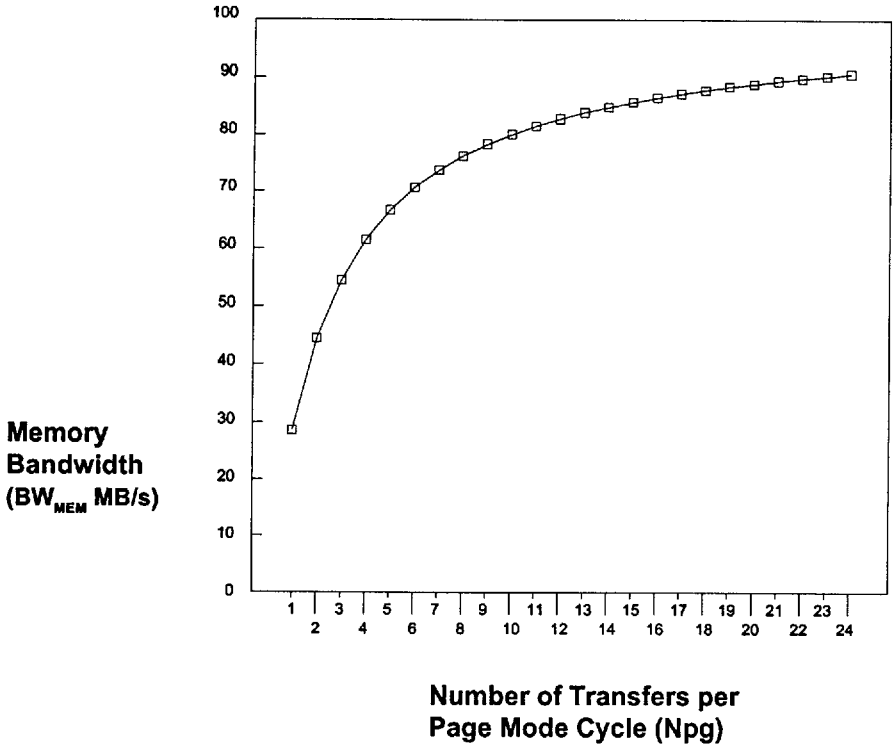
The plotted function is:

Y-Axis: Memory Bandwidth (BW_{mem})
X-Axis: Number of transfers per page mode cycle (N_{pg})

$$BW_{mem} = \frac{N_{pg} \times W_{db}}{T_{ran} + ((N_{pg}-1) \times T_{pg})}$$

Similar graphs can be plotted for different memory configurations.

Figure 6.4.2-1 Memory Bandwidth (MB/s) as a Function of Transfers per Page Mode Access



Depending on the system requirements at any given time, the ET4000/W32p will operate at a certain point on the curve. The operation point on the curve indicates how much bandwidth the ET4000/W32p can **supply**, not how much it is **using**; so if the CRT Controller requires 20 MB/s and the ET4000/W32p can supply 60 MB/s, then there is 40 MB/s (60-20=40) remaining that can be supplied to another requester.

For example, if the CRT Controller is the only requester at a certain point in time, the ET4000/W32p will operate well to the right of the curve because the sequential nature of the display scanning is a good match to the DRAM page-mode capability. However, if the Host attempts to read or write display memory, two things happen: the number of accesses per page-mode cycle on behalf of the CRTC gets lower, and there are additional memory cycles required to service the demands of the Host. These additional cycles may or may not be well-suited to DRAM page-mode, depending on the Host's pattern of addressing. These two factors have the net effect of moving the operating point to the left along the curve. If the Host is drawing a vertical line it will of course move the operating point to the left by more than if it was drawing a horizontal line.

Clearly, as more requesters get involved, the ET4000/W32p's management of the DRAM becomes rather complex and very dynamic, making it difficult to project with any accuracy the operating point on the curve. For a typical workload supporting a single CRTC and ACL BLT activity, the ET4000/W32p can be expected to operate around 8 to 16 transfers per page-mode cycle.

6.4.2.1 Interleave DRAM Interface

Memory interleaving is done to increase DRAM bandwidth without doubling the DRAM data bus pin count. The DRAM bandwidth is increased by 1.7 times by using memory interleaving.

The ET4000/W32p provides the capability to interleave data from two banks of DRAM which share common RAS, address, write enable, and data signals. They are unique via the CAS signals.

Interleaving is defined by specifying that the CAS pre-charge time of one DRAM bank is concurrent with the CAS active time of the other bank.

DRAM Interleave Design Considerations

When in an interleaved configuration, any DRAM device that shares a data line with another DRAM device should be as identical as possible to ensure complete compatibility in the DRAM data output's source and sink current. This degree of compatibility is necessary due to the nature of interleaving which causes one DRAM to turn on while the other is turning off during DRAM read cycles. Another consideration is the Taa (access time from CAS) timing parameters £ 27ns when the SCLK = 50MHz. Every effort should be made to minimize capacitive loading on the DRAM control, address, and data PCB traces.

OE Controlled Interleave DRAM Interface (Except Revisions A & B)

This feature decouples the DRAM column address latching from the DRAM read data output enable. This will add 1/2 SCLK period to the DRAM's column address access time with respect to previous CAS-only controlled interleave designs (i.e., W32i, W32p Revisions A & B). This is done by shortening the CAS precharge time by 1/2 SCLK and adding that time to the CAS low time. In doing this, the CAS low times of the even and odd interleave banks will overlap, necessitating the use of OE data control to eliminate data contention. This allows the use of slower DRAMs or higher SCLKs with respect to the interlave operation. This function is enabled by setting CRTC Indexed Register 37 <2> = 0.

Using this function requires the addition of four output enable pins which are shared as follows:

OEOB* (odd interleave phase "B" bank)	shared with	MUX<1>
OEOA* (odd interleave phase "A" bank)	shared with	EDCK
OEEB* (even interleave phase "B" bank)	shared with	ESYC
OEEA* (even interleave phase "A" bank)	shared with	EVID

EDCK, ESYC, and EVID are only available in the following configurations: F11100
F10011
F01000

Also, in order to supprt this function, MUX<1> must be in input mode by setting IMA Indexed Register F7 <6> = 0. The two remaining configurations, F01011 and F10001 do not support this function.

The input pins that the OE signals are shared with continue to function as inputs. This is accomplished by latching them at times when the state of the OE signals are "don't care." In order to decouple the input and output functions, 4.7Kohm series resistors between the shared pins and the input signals are necessary, and 1Kohm pullup resistors on the input signals (not the shared pins) must also be provided. In order to reduce capacitance on the OE signals, keep the 4.7Kohm resistors close to their respective pins.

The wiring for this function is identical to the CAS-only interleave DRAM wiring with the exception of the added output enables. The output enables are wired as follows:

CAS connection	OE connection
CAS<7:6>	OEOB*
CAS<5:4>	OEOA*
CAS<3:2>	OEEB*
CAS<1:0>	OEEA*

6.4.3 CRTC Bandwidth Requirements

The amount of memory bandwidth required by the CRTC depends upon the display resolution and display refresh frequency. Below is the computation for CRTC bandwidth:

Let HDE = Horizontal Display Enable.
Let VDE = Vertical Display Enable.
Let BPP = Bytes per pixel.
Let V_{rate} = Vertical (Screen) Refresh Rate.

Then CRTC bandwidth is given by:

$$BW_{CRTC} = HDE \times VDE \times BPP \times V_{rate}$$

For example, for display mode 2E:

$$BW_{CRTC} = 640 \times 480 \times 1 \times 60 = 18.4 \text{ MB/s}$$

The table below offers a summary of how the CRTC bandwidth requirement varies with display mode:

Display Mode	BWcrtc(MB/s)	Notes
2E	18.4	640x480x8, 60Hz
2E 72h	22.1	640x480x8, 72Hz
30 38k	28.8	800x600x8, 60Hz
30 48k	34.6	800x600x8, 72Hz
38n	47.2	1024x768x8, 60Hz
38 72m	54.6	1024x768x8, 72Hz

Servicing the requirement of the CRTC is the top priority of the ET4000/W32p. The ET4000/W32p will meet the demands of the CRTC first, and any remaining memory bandwidth will be allotted to other requesters.

6.4.4 Accelerator Bandwidth Requirements

The internal Graphics Accelerator is designed to have very high bandwidth requirements, so that generally the speed of an accelerated operation is limited by the memory bandwidth made available by the ET4000/W32p's memory controller (MCU). This implies that the accelerator performance will increase in direct proportion to faster memory chips and a wider MD bus.

6.4.5 Image Port Bandwidth Requirements

The bandwidth requirement of the Image Port is dependent upon the nature of the data being transferred and the amount of data per frame. Below is the computation for Image Port bandwidth:

Let H_{size} = Number of pixels per scan line.
Let V_{size} = Number of scan lines per frame.
Let BPP = Bytes per pixel.
Let V_{rate} = Vertical (Screen) Refresh Rate.

Then Image Port bandwidth is given by:

BW_{ima} = Hsize x Vsize x BPP x Vrate

For example, for a window of size 400x200 pixels, with 24 bits per pixel, at 30 frames/sec:

BW_{ima} = 400 x 200 x 3 x 30 = 7.2 MB/s

Generally, if a system is being designed to capture and display full-motion, True-Color video, only a 32-bit MD bus should be considered.

6.5 Image Bus Interface Design Considerations

Some system design limitations must be observed:

1. General limitation: System linear map limited to 1MB flat address space instead of 4MB.

2. The IXRD signal should be synchronized to the external image processor's clock.

3. If IMA Indexed Register 7, bit 1 is programmed to 0 (non-interlaced mode), the IXOF input pin should be low.

6.6 Clock Generator Design Considerations

6.6.1 Master Clock Select

A variable Master Clock (MCLK), is used internally by the ET4000/W32p to derive the video, vertical, and horizontal timing for the various video modes. Depending on the video monitor and display timing desired, up to 8 different frequencies, and optionally up to 32, can be selected. The clock source is controlled by clock select signals (CS<2:0> and optionally C3IR and C4BD—see CRTC Indexed Register 31) generated by the ET4000/W32p as programmed. (See also Section 5.2.32, CRTC Indexed Register 34: Auxiliary Control Register.)

6.6.2 System Clock Select

The System Clock (SCLK) is required to sequence the ET4000/W32p's internal control logic. In addition, the SCLK is used to produce the memory interface control timing: RAS, CAS, and MW, etc.

The SCLK also affects the overall "balance" of the ET4000/W32p's performance. Therefore, designers must fully understand the effect of SCLK when cost/performance trade-offs are considered. In general, the SCLK's cycle time should be equal to the CAS low pulse width and less than 25ns. (See also Section 5.2.30, CRTC Indexed Register 32: RAS/CAS Configuration.)

6.6.3 Display Support and Video Timing

VGA-compatible video subsystems are used as an example here to discuss display support and video timing for an ET4000/W32p-based video design. The VGA-compatible video subsystem supports attachment of 31.5kHz horizontal sweep frequency direct-drive analog displays. These displays have a vertical sweep frequency capability of 50 to 70 cycles per second, providing extended color and sharpness and reduced flicker in most modes. The following table summarizes the VGA-compatible analog display and high-resolution interlaced monitor characteristics.

Parameter	Color	Monochrome	HiRes Color	(Interlaced)
Horizontal Scan Rate	31.5kHz3	1.5kHz	35.5kHz	
Vertical Scan Rate	50 to 70Hz	50 to 70Hz	43.5 Hz	
Video Dot Clock	28MHz	28MHz	44.9MHz	
Displayable Colors*	256/256K Max.	64/64 Shades Gray	256/256K Max.	65,536
Max. Horiz. Resolution	720 PELs	720 PELs	1024 PELs	1024 PELs
Max. Vert. Resolution	480 PELs	480 PELs	768 PELs	768 PELs

* Controlled by Video Circuit

All IBM-compatible VGA/EGA modes have the same horizontal sweep rate. The vertical height of the display is controlled by the polarity of the vertical and horizontal pulses. This is done so that 350, 400, or 480 lines can be displayed without adjusting the height of the display.

The BIOS sets the ET4000/W32p registers to generate the video modes. The video modes are shown in table 8.1-1. All of these modes are 70 Hz vertical retrace except for modes 11 and 12. These two modes are 60 Hz vertical retrace. The ET4000/W32p generates timings that are within the specifications for the supported displays using these modes.

The VGA-compatible analog displays operate from 50 to 70 Hz vertical retrace frequency. The following timing diagrams represent only the vertical frequencies set by the BIOS.

VSYNC Polarity	HSYNC Polarity	Vertical Size
+	+	768 lines
+	-	400 lines
-	+	350 lines
-	-	480 lines

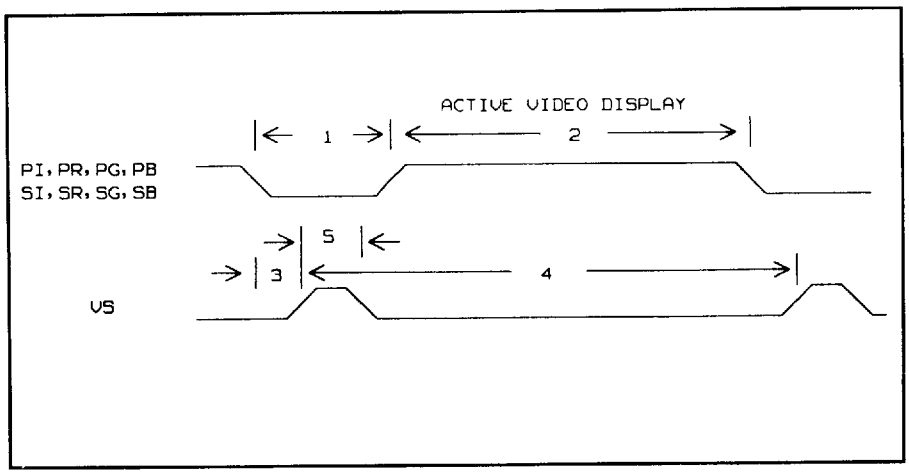


Figure 6.6.3-1 Display Vertical Sync, 350 lines

Signal Time	Typical
1	2.765 milliseconds
2	11.504 milliseconds
3	0.985 milliseconds
4	14.268 milliseconds
5	0.064 milliseconds

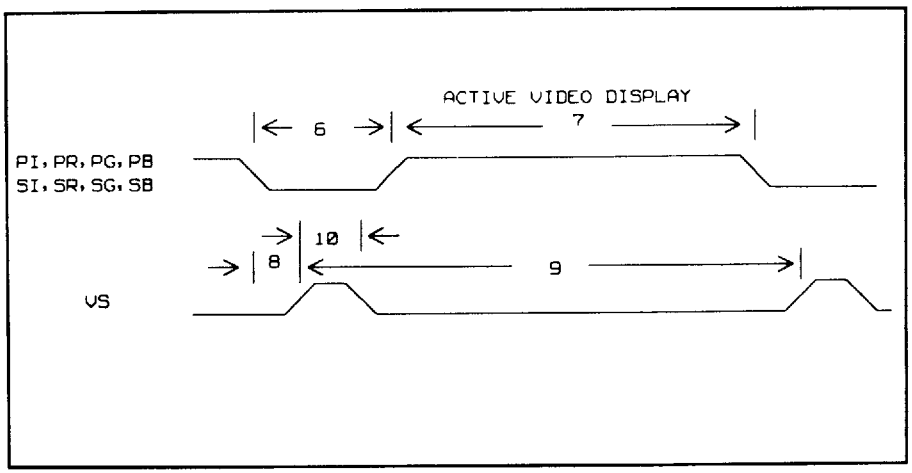


Figure 6.6.3-2 Display Vertical Sync, 400 lines

Signal Time	Typical
6	1.112 milliseconds
7	13.156 milliseconds
8	0.159 milliseconds
9	14.268 milliseconds
10	0.064 milliseconds

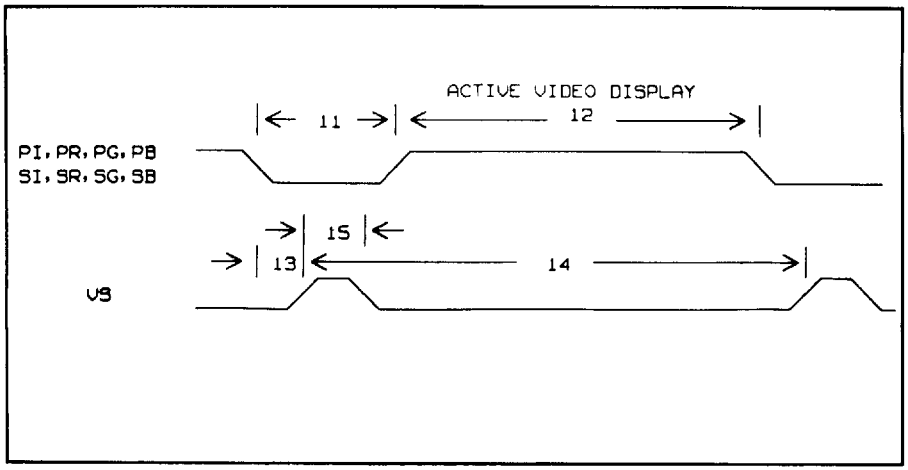


Figure 6.6.3-3 Display Vertical Sync, 480 lines

Signal Time	Typical
11	0.922 milliseconds
12	15.762 milliseconds
13	0.064 milliseconds
14	16.683 milliseconds
15	0.064 milliseconds

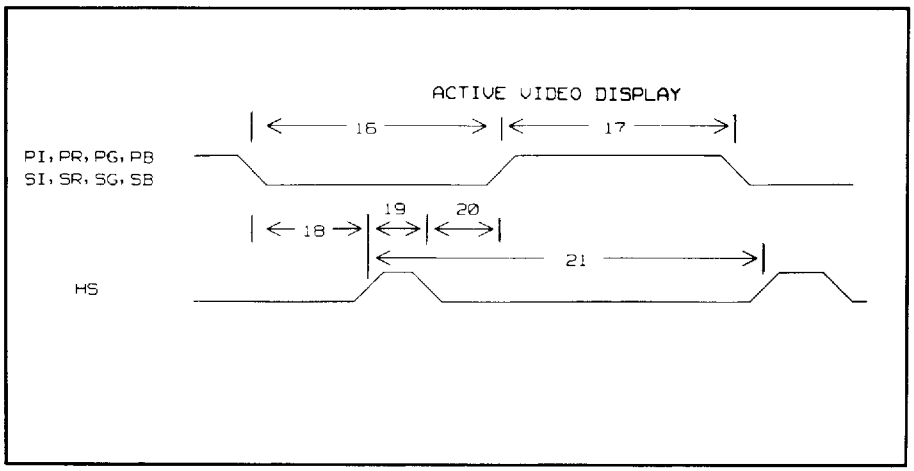


Figure 6.6.3-4 Display Horizontal Timing, 80 Column with Border

Signal Time	Typical
16	5.720 microseconds
17	26.058 microseconds
18	0.318 microseconds
19	3.813 microseconds
20	1.589 microseconds
21	31.778 microseconds

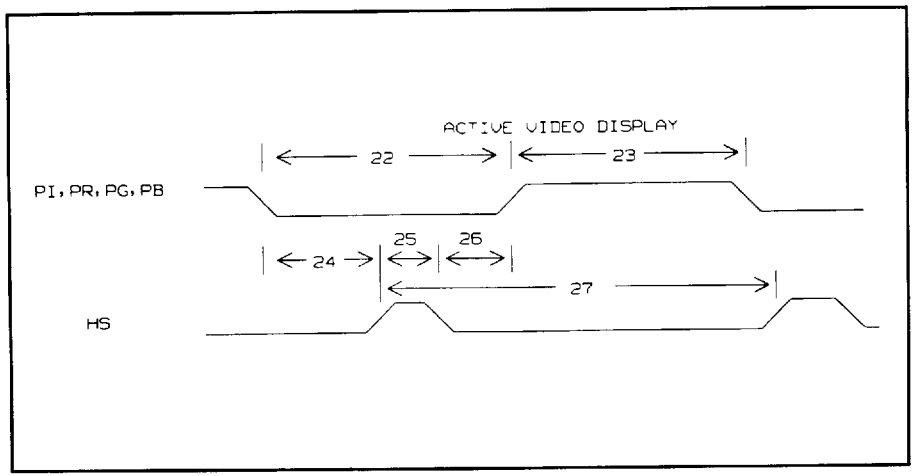


Figure 6.6.3-5 Display Horizontal Timing, 40/80 Column, no Border

Signal Time	Typical
22	6.356 microseconds
23	25.422 microseconds
24	0.636 microseconds
25	3.813 microseconds
26	1.907 microseconds
27	31.778 microseconds

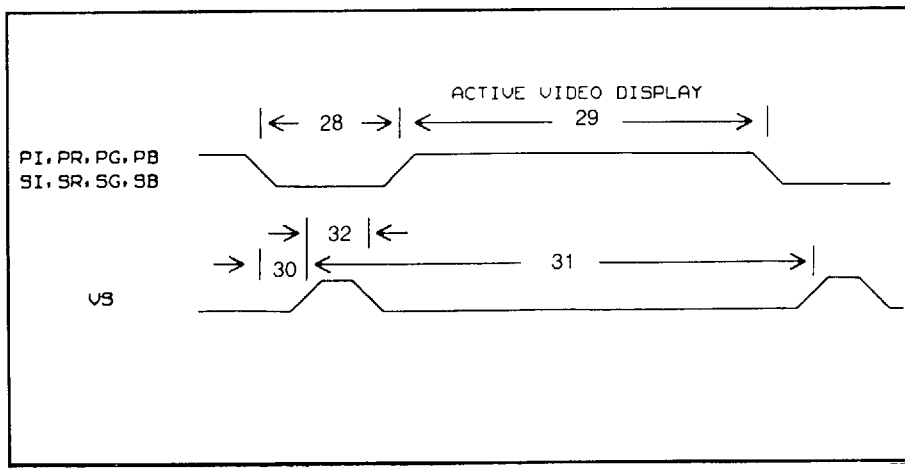


Figure 6.6.3-6 Display Vertical Sync Timing, 768 Lines

Signal Time	Typical
28	1.38 microseconds
29	21.62 microseconds
30	0.014 microseconds
31	23.0 microseconds
32	0.112 microseconds

6.7 Video Clock Design Considerations

The MCLK can be selected by programming the Miscellaneous Output Register (I/O address = 3C2) bit <3,2> = Clock select CS<1,0> (Refer to Fig. 2.1-1). The recommended hardware connection and programming of the CS<1,0> bits are shown below:

clock inputs	selected by	type
CK1 = 25.175MHz	CS<1,0> = 00	VGA mode
CK2 = 28.322MHz*	= 01	VGA mode/CGA*
CK3 = 32.514MHz	= 10	EGA* mode
CK4 = 40.0MHz	= 11	Extended mode

* Use MCLK/2 internally to yield half MCLK.

The ET4000/W32p provides 3 additional clock selects CS<4:2> for up to 32 clock sources; CRTC Indexed Register 34 Bit 1, clock select CS<2> and CRTC Indexed Register 31, bits 7:6, clock select CS<4:3>, when used in conjunction with CS<1:0>.

6.8 RAMDAC

An external Palette RAM with Digital-to-Analog Converter (DAC) is used to translate 8-, 16- or 24 bits of digital video signal into the three Analog outputs (R, G, B).

For complete Palette RAM interface, the following output pins are available from the ET4000/W32p: PMERL, PMEWL, PCLK, MBSL, AP<15:0>, BDE.

6.9 ET4000/W32p BIOS ROM

The ET4000/W32p BIOS ROM contains modules that provide generic video BIOS functions to support both VGA- and EGA-compatible modes. (See Section 7.4 for operation modes.) When the BIOS ROM option is employed, the CPU reads the BIOS ROM during the bootstrap operation, and the ROM Enable signal (ROMEL) will be activated to enable the ROM data onto the DD bus.

ROME* can be made to respond to address decode only, allowing for external bus-based (ISA bus) ROM to be decoded from the local bus addresses without the necessity for external decode glue logic. See Sections 3.3.1, 6.2.2 (Local Bus ROM), and 6.3 (PCI ROM).

6.9.1 Local Bus ROM

In Local Bus configurations, the ROM can be placed either on the local bus (via the W32p's DD bus), external (ISA) bus, or embedded in the system ROM. Placement on the local bus is problematic when video BIOS shadowing is enabled in the core logic. Therefore, it is either recommended that the local bus ROM be placed either on the external bus or embedded in the system ROM. For the W32p Rev.A, an external address decoder is required to generate ROME*. On the W32p Rev. B the signal can be generated using a ROME* pin if the DISB PORI bit is set high.

6.9.2 PCI Bus ROM

In PCI Bus configurations, the ROM can be placed either on the PCI bus (via the W32p's DD bus) or embedded in the system ROM. Placement on the PCI bus requires external latches to latch the PCI's address/data bus.

Except Revisions A & B The PCI BIOS may be placed enitrely on the DD bus by latching its address using DD<7:0>, ADDL, and ROME* as described in Section 6.3 PCI ROM.

6.9.3 VGA ROM versus System ROM trade-offs

The BIOS ROM is either on the ISA bus or it is embedded in the System ROM. The benefit of embedding the BIOS into the system ROM is that you can save an EPROM, thus saving the cost of a component and some real estate of the system board. The configuration is that the system BIOS instructions are located in one area and the video BIOS is in another, typically located at E0000. The BIOS would need to be intelligent enough to detect the presence of a bus-installed video adapter should a video card be introduced to the system, or a jumper or switch would need to be utilized in order to disable the on-board video to avoid conflict.

6.9.4 Specifying BIOS ROM Address Space

The ET4000/W32p is designed to decode C0000-C7FFF (hex) as the EROM address space on power-up, providing 32KB code size for the ET4000/W32p BIOS ROM modules. This address space can be redefined by programming TS Index Register 7 (TS Auxiliary Register) bits 5 and 3.

7. Programming Considerations

7.1 Text Modes

Four bit planes are provided to allow more efficient use of video memory. The use of these planes depends on the mode being selected.

When an alphanumeric mode is selected, four bit planes are divided into two pairs of odd/even planes. The BIOS transfers character patterns from the ROM to the upper plane pair—bit planes 2 and 3. The system microprocessor stores the character and attribute data in the lower plane pair—bit planes 0 and 1. The programmer can view bit planes 0 and 1 as a single buffer in alphanumeric modes. The CRTC generates sequential word addresses and fetches one character/attribute word at a time. This allows the execution of programs having the character code in even byte addresses and the attribute data for that character in the odd byte address that follows.

Every display character position in the alphanumeric mode is defined by two bytes in the display buffer. Both the color/ graphics and the monochrome emulation modes use the following 2-byte character/attribute format.

Bit	Attributes
7	Blink/background intensity
6	Background color, red
5	Background color, green
4	Background color, blue
3	Foreground intensity/character select
2	Foreground color, red
1	Foreground color, green
0	Foreground color, blue

NOTES:
1. Bit 7 can be used for either Blink or Background Intensity characters. See function "AH=10h, AL=3 (Toggle Intensity/Blinking Bit)" in section 5.
2. Bit 3 can be used for either Foreground Intensity or Character Set Select.

VGA/TLI compatible text modes supported by the ET4000/W32p's GENERIC BIOS are described on the following pages.

7.1.1 40x25 Text (Modes 0 and 1):

Mode	Char/ Row	No. Rows	Buffer Start	Max. Pages	Memory Required
0,1	40	25	B8000	8	2KB

Memory allocations for display pages:

Page	Start Address
1	B8000
2	B8800
3	B9000
4	B9800
5	BA000
6	BA800
7	BB000
8	BB800

7.1.2 80x25 Text (Modes 2, 3, and 7):

Mode	Char/ Row	No. Rows	Buffer Start	Max. Pages	Memory Required
2,3	80	25	B8000	8	4KB
7	80	25	B0000	8	4KB

Memory allocations for display pages:

Page	Modes 2 and 3 Start Address	Mode 7 Start Address
1	B8000	B0000
2	B9000	B1000
3	BA000	B2000
4	BB000	B3000
5	BC000	B4000
6	BD000	B5000
7	BE000	B6000
8	BF000	B7000

7.1.3 132x44 Text (Modes 18 and 22):

Mode	Char/ Row	No. Rows	Buffer Start	Max. Pages	Memory Required
18	132	44	B0000	2	11.6KB
22	132	44	B8000	2	11.6KB

Memory allocations for display pages:

Page	Modes 18 Start Address	Mode 22 Start Address
1	B0000	B8000
2	B4000	BC000

7.1.4 132x25 Text (Modes 19 and 23):

Mode	Char/ Row	No. Rows	Buffer Start	Max. Pages	Memory Required
19	132	25	B0000	4	6.6KB
23	132	25	B8000	4	6.6KB

Memory allocations for display pages:

Page	Modes 19 Start Address	Mode 23 Start Address
1	B0000	B8000
2	B2000	BA000
3	B4000	BC000
4	B6000	BE000

7.1.5 132x28 Text (Modes 1A and 24):

Mode	Char/ Row	No. Rows	Buffer Start	Max. Pages	Memory Required
1A	132	28	B0000	4	7.4KB
24	132	28	B8000	4	7.4KB

Memory allocations for display pages:

Page	Modes 1A Start Address	Mode 24 Start Address
1	B0000	B8000
2	B2000	BA000
3	B4000	BC000
4	B6000	BE000

7.1.6 80x60 Text (Mode 26):

Mode	Char/ Row	No. Rows	Buffer Start	Max. Pages	Memory Required
26	80	60	B8000	2	9.6KB

Memory allocations for display pages:

Page	Start Address
1	B8000
2	BC000

7.1.7 100x40 Text (Mode 2A):

Mode	Char/ Row	No. Rows	Buffer Start	Max. Pages	Memory Required
2A	100	40	B8000	4	8KB

Memory allocations for display pages:

Page	Start Address
1	B8000
2	BA000
3	BC000
4	BE000

7.2 Graphic Modes

VGA/TLI compatible graphics modes supported by the ET4000/W32p's GENERIC BIOS are described below:

7.2.1 320x200 Four Color (Modes 4 and 5):

Mode	Colors/ Pixel	Pixels/ Byte	Pixels/ Row	No. Rows	Buffer Start	Max. Pages	Memory Required
4,5	4	4	320	200	B8000	1	16KB

The ET4000/W32p should be programmed to be in IBM VGA mode in modes 4 and 5.

Each data byte contains two color bits for four PELs:

Bit	
7,6	C1,C0 of first PEL
5,4	C1,C0 of second PEL
3,2	C1,C0 of third PEL
1,0	C1,C0 of fourth PEL

where the color bits combine to give:

C1	C0	
0	0	Black
0	1	Green or cyan
1	0	Red or magenta
1	1	Brown or intense white

C0 is stored in bit plane 0; C1 is stored in bit plane 1.

The display buffer is partitioned into two sections of 8000 bytes each. One section contains PELs for display on even scan lines (lines 0, 2, 4 through 198), the other contains PELs for the odd scan lines:

Section	Lines	Start Address	End Address
0	Even	B8000	B9F3F
1	Odd	BA000	BBF3F

NOTES:

1. Hex B8000 contains PEL data for the upper-left corner of the display area.
2. Odd scan lines are offset from even scan lines by 8K.

7.2.2 640x200 Two Color (Mode 6):

Mode	Colors/ Pixel	Pixels/ Byte	Pixels/ Row	No. Rows	Buffer Start	Max. Pages	Memory Required
6	2	8	640	200	B8000	1	16KB

Each data byte defines a row of eight PELs on the screen. Each bit defines 2 colors for each PEL as follows:

0	Black
1	Intensified white

7.2.3 16/256K Colors (Modes D, E, 10 and 12):

Mode	Colors/ Pixel	Pixels/ Byte	Pixels/ Row	No. Rows	Buffer Start	Max. Pages	Memory Required
D	16	1*	320	200	A0000	8	32KB
E	16	1*	640	200	A0000	4	64KB
10	16	1*	640	350	A0000	2	112KB
12	16	1*	640	480	A0000	1	153.6KB

Starting addresses for the display pages are:

	Mode D	Mode E	Mode 10	Mode 12
Number of display pages:	8	4	2	1
Starting addresses for each page:				
Page				
1	A0000	A0000	A0000	A0000
2	A2000	A4000	A8000	
3	A4000	A8000		
4	A6000	AC000		
5	A8000			
6	AA000			
7	AC000			
8	AE000			

* Four bit planes, each starting at location hex A0000, provide the four color bits required for each pixel as follows:

Plane 3	C3	Intensity
Plane 2	C2	Red
Plane 1	C1	Green
Plane 0	C0	Blue

Within each bit plane, each data byte defines a row of eight pixels on the screen, for a specific color bit. The planes are accessed simultaneously and the color bits together address one register in a table of 16 color registers.

If the values in the registers are the supplied default values, the colors will be:

C3	C2	C1	C0	
0	0	0	0	Black
0	0	0	1	Blue
0	0	1	0	Green
0	0	1	1	Cyan
0	1	0	0	Red
0	1	0	1	Magenta
0	1	1	0	Brown
0	1	1	1	White
1	0	0	0	Gray
1	0	0	1	Light Blue
1	0	1	0	Light Green
1	0	1	1	Light Cyan
1	1	0	0	Light Red
1	1	0	1	Light Magenta
1	1	1	0	Yellow
1	1	1	1	Intense White

The 16 colors are mapped to the supported monochrome monitor as 16 shades ranging from black to intense white.

The graphics program interface to the bit planes is through the READ DOT and WRITE DOT functions; C0 through C3 are bits 0 through 3 of the color data.

7.2.4 640x350 Monochrome (Mode F):

Mode	Colors/ Pixel	Pixels/ Byte	Pixels/ Row	No. Rows	Buffer Start	Max. Pages	Memory Required
F	4*	1**	640	350	A0000	2	56K

* 4 monochrome attributes per pixel
** 1 bit from each of plane 0 and 2 per pixel

Memory allocation for the display pages are:

Page	Start Address
1	A0000
2	A8000

Two bytes (one from plane 0 and one from plane 2) together define a row of eight PELs on the screen. Bit planes 2 and 0 are accessed simultaneously, to support graphics on displays that use the following attributes: black, video, blinking video, and intensified video. Bit plane 2 provides color bit C2 and plane 0 provides color bit C0, combining to give the attributes as follows:

C2	C0	
0	0	Black
0	1	Video
1	0	Blink video
1	1	Intense video

The graphics program interface to the bit planes is through the READ DOT and WRITE DOT functions; C0 and C2 are bits 0 and 2 of the color data.

7.2.5 640x480 Two Color (Mode 11):

Mode	Colors/ Pixel	Pixels/ Byte	Pixels/ Row	No. Rows	Buffer Start	Max. Pages	Memory Required
11	2	8	640	480	A0000	*	38.4KB

Each data byte defines a row of eight pixels on the screen. Each bit defines 2 colors for each PEL as follows:

0	Black
1	Intensified white

* The data are stored in a linear display buffer (containing both odd and even rows) starting at address hex A0000.

7.2.6 256/256K Colors (Mode 13):

Mode	Colors/ Pixel	Pixels/ Byte	Pixels/ Row	No. Rows	Buffer Start	Max. Pages	Memory Required
13	256	1	320	200	A0000	1	64KB

The four memory planes are chained together to form a linear display buffer 256KB deep starting at address hex A0000. Each byte defines the 8-bit color data for one PEL on the screen.

Color data read from the display buffer is used as a pointer to address one register in a table of 256 color registers (the External Palette RAM). The default color mapping is:

Registers 0 through 15 map to the 16 EGA colors;
Registers 16 through 31 map to evenly spaced shades of gray;
Registers 32 through 247 map to a range of color shades
based on a Hue/Saturation/Intensity model that provides a
usable set of colors.

NOTE: Changing the internal palette (that which is compatible with the Enhanced Graphics Adapter) from the default setting will produce unpredictable results. If you want to change the colors, change the values in the individual color registers.

7.2.7 16/256K Colors (Modes 25, 29, 37, 3D):

These modes share the same 16-color data format as modes D, E, 10 and 12 described previously. For a detailed description of the color format and access to Palette RAM, refer to the prior description. The resolution and display buffer map are, however, different and are described below:

Mode	Colors/ Pixel	Pixels/ Byte	Pixels/ Row	No. Rows	Buffer Start	Max. Pages	Memory Required
25	16	1*	640	480	A0000	1	153.6KB
29	16	1*	800	600	A0000	1	240KB
37	16	1*	1024	768	A0000	1	393.2KB
3D	16	1*	1280	1024	A0000	1	655.4KB

* One bit from each bit plane per pixel.

7.2.8 256/256K Colors (Modes 2D, 2E, 2F, 30, 38):

These modes share the same 256-color data format as mode 13. For a detailed description of the color format and access to External Palette RAM, refer to the description following "256/256K Colors (Mode 13):"

The resolution and display buffer map are different, however, and are described below:

Mode	Colors/ Pixel	Pixels/ Byte	Pixels/ Row	No. Rows	Buffer Start	Max. Pages	Memory Required
2D	256	1	640	350	A0000	1	224KB
2E	256	1	640	480	A0000	1	307.2KB
2F	256	1	640	400	A0000	1	256KB
30	256	1	800	600	A0000	1	480KB
38	256	1	1024	768	A0000	1	786.4KB

7.3 Video Memory Map

AMC <2:0>	MAP <1:0>	VGA memory (no MMU)	MMU buffer memory	MMU Aperture Size	Memory Mapped Registers	External Mapped Registers
000	00	xA0000-xBFFFF	—	—	—	—
000	01	xA0000-xAFFFF	—	—	—	—
000	10	xB0000-xB7FFF	—	—	—	—
000	11	xB8000-xBFFFF	—	—	—	—
100	01	xA0000-xAFFFF	xB8000-xBDFFF	8KB	—	—
100	10	xB0000-xB7FFF	xA8000-xADFFF	8KB	—	—
100	11	xB8000-xBFFFF	xA8000-xADFFF	8KB	—	—
110	01	xA0000-xAFFFF	xB8000-xBDFFF	8KB	xBFF00-xBFFFF	xBE000-xBEFFF
110	10	xB0000-xB7FFF	xA8000-xADFFF	8KB	xAFF00-xAFFFF	xAE000-xAEFFF
110	11	xB8000-xBFFFF	xA8000-xADFFF	8KB	xAFF00-xAFFFF	xAE000-xAEFFF
001	xx	000000-3FFFFF	—	—	—	—
*111	xx	000000-1FFFFF	200000-37FFFF	512KB	3FFF00-3FFFFF	380000-3BFFFF
**111	xx	—	000000-2FFFFF	1MB	3FFF00-3FFFFF	37F000 - 37FFFF

Notes:
* = *Revisions A&B ONLY*
** = *Revision C*
xx = don't care

MAP<1:0> = GDC 6, bits <3:2> (VGA Memory Map)
AMC<2> = CRTC 36, bit <3> (1 enables MMU)
AMC<1> = CRTC 36, bit <5> (1 enables memory-mapped registers)
AMC<0> = CRTC 36, bit <4> (1 enables system linear memory mapping)

The following table describes the external memory address mapping relationship between the CRTC address and CPU address lines.

Table 7.3-1 CPU/CRTC Addressing Modes

MA	CPU PG	OE	LG	S1	CRTC S0	SF
-	RW0	A0	A0	##	##	#
-	RW1	RW1	A1	##	##	#
A<0>	A<0>	PGS	A<2>	L<0>	L<15>	RA<0>
A<1>	A<1>	A<1>	A<3>	L<1>	L<0>	RA<1>
A<2>	A<2>	A<2>	A<4>	L<2>	L<1>	RA<2>
A<3>	A<3>	A<3>	A<5>	L<3>	L<2>	RA<3>
A<4>	A<4>	A<4>	A<6>	L<4>	L<3>	RA<4>
A<5>	A<5>	A<5>	A<7>	L<5>	L<4>	CC<0>
A<6>	A<6>	A<6>	A<8>	L<6>	L<5>	CC<1>
A<7>	A<7>	A<7>	A<9>	L<7>	L<6>	CC<2>
A<8>	A<8>	A<8>	A<10>	L<8>	L<7>	CC<3>
A<9>	A<9>	A<9>	A<11>	L<9>	L<8>	CC<4>
A<10>	A<10>	A<10>	A<12>	L<10>	L<9>	CC<5>
A<11>	A<11>	A<11>	A<13>	L<11>	L<10>	CC<6>
A<12>	A<12>	A<12>	A<14>	L<12>	L<11>	CC<7>
A<13>	A<13>	A<13>	A<15>	L<13>	L<12>	FS<2>
A<14>	A<14>	A<14>	SP<0>	L<14>	L<13>	FS<0>
A<15>	A<15>	A<15>	SP<1>	L<15>	L<14>	FS<1>
A<16>	SP<0>	SP<0>	SP<2>	L<16>	L<16>	-
A<17>	SP<1>	SP<1>	SP<3>	L<17>	L<17>	-
A<18>	SP<2>	SP<2>	SP<4>	L<18>	L<18>	-
A<19>	SP<3>	SP<3>	SP<5>	L<19>	L<19>	-

NOTES:

1. A<21:0> = CPU Byte Address
2. MA<19:0> = Video Memory Doubleword Address
3. SP<7:0> = 8-Bit Read/Write Segment Pointer (GDC Segment Select)
4. L<19:0> = Linear Counter Doubleword Address
   RA <4:0> = Character Row Scan
   CC <7:0> = Character Code
   FS <2:0> = Character Font Select
5. SO = word CRTC address mode S1 = byte or doubleword CRTC mode
6. # = In text mode, I/R lanes are font planes and G/B planes are attribute/character code planes. In graphics modes all four planes are used as pixel data.
7. RW<1:0> = read plane select (GDC Indexed Register 4) and write plane mask (TS Indexed Register 2).
8. PGS = /PSEL*CHAN*(CDS<1>|CDS<0>)
   | A<16>*CHAN*/(CDS<1>|CDS<0>)
   | A<0>*/CHAN

7.3.1 Two Major Types of Memory Organization

From a programming viewpoint, the display data can be structured into video bit-planes (under a Planar Organization), or into memory arrays (under a Linear Byte Organization), depending on the particular video mode to be supported. A discussion of these two types of systems along with their addressing scheme and typical modes follows.

7.3.1.1 Planar Organization

In VGA-compatible 16-color modes, a "Plane" configuration is used, where four independently addressable bit planes (I,R,G,B) are accessed in parallel, and each pixel is represented by up to four bits from the four planes, selecting up to 16 colors. The size of each plane depends on the resolution supported.

The CPU shall access the display buffer using the Read Plane Select (RPS) (GDC Index Register 4) and Write Plane Mask (WPM) (TS Index register 2), applicable only for Planar Organizations, with 16-bit address lines addressing up to 64KB on each plane.

For planes greater than 64KB, the Segment Select register is used.

The major advantages of a Planar Organization include:

1. Allows parallel access of all four color planes (32 bits or eight Pixels), through one CPU I/O operation, therefore minimizing the frequency of CPU accesses.

2. By spreading four bits per pixel over four bit-planes, the total address space per plane is reduced by a factor of 4, therefore minimizing the need of crossing 64KB segment boundaries.

3. The ET4000/W32p provides two types of read and four types of write operations, during which the display data can be processed, provided they are structured as four planes. These operations include "color compare" to expedite color-fill functions, block move operations, set/reset functions to facilitate initialization of the display buffer, and bit-masking facilities to allow modification of up to 32 pixels by a single CPU memory read/write operation.

7.3.1.2 Linear Byte Organization

To support either 256 or 65,536 colors, a "Linear Byte" organization is used, whereby all memory planes are chained together as a linear byte-oriented memory. Each pixel is represented by 1 single byte in 256 colors or 2 adjacent bytes in 65,536 colors. The depth of the linear array depends on the resolution or number of colors supported, and therefore the amount of display buffer required.

The CPU shall access the display buffer using address lines and the segment select register, or in conjunction with the MMU's logical to physical address translation, while the Read Plane Select (RPS) and Write Plane Mask (WPM) are **ignored**.

The major advantages of a Linear Byte organization include:

1. Each pixel is represented by adjacent bytes (packed data), which eliminates the need to manipulate data across byte boundaries associated with Plane systems.

2. Simplifies direct data manipulation such as the color shading function which adjusts the color of each pixel, since addressing a pixel does not require any I/O port access as does a plane system.

3. W32p Accelerator can be used to greatly increase performance for many operations.

7.4 Operation Mode Tables

The following tables summarize the values to be programmed into the ET4000/W32p registers for the various modes of operation.

VGA MODES

Table 7.4.4-1 General Registers

Register Name	Port	Index	0|0*|0+	1|1*|1+	2|2*|2+	3|3[*]|3+	4	5	6	7|7*	D	E	F	10	11	12	13	21	22	23	24	25	26	29	2A	2D	2E	30	37i	37n	2F	38i	38n	3D
Misc Output	3C2	-	63|A3|63	63|A3|63	63|A3|63	63|A3|63	63	63	63	A2|62	63	63	A2	A3	E3	E3	63	E3	A7	A7	A7	E3	E3	EF	EF	A3	E3	EF	2F	EF	63	2F	EF	2F
Clock			25|25|28	25|25|28	25|25|28	25|25|28	25	25	25	28|28	25	25	25	25	25	25	25	40	40	40	25	28	40	40		25	25	40	45	65	25	45	65	80

Table 7.4.4-2 Timing Sequencer Registers

Register Name	Port	Index	0|0*|0+	1|1*	2|2*|2+	3|3*	4	5	6	7|7*	D	E	F	10	11	12	13	21	22	23	24	25	26	29	2A	2D	2E	30	37i	37n	2F	38i	38n	3D
TS Index	3C4	-																																
Synch Reset	3C5	00	03|03|03	03|03	03|03|03	03|03	03	03	03	03|03	03	03	03	03	03	03	03	03	03	03	03	03	03	03	03	03	03	03	03	03	03	03	03
TS Mode	3C5	01	09|09|08	09|09	01|01|00	01|01	09	09	01	00|00	09	01	01	01	01	01	01	01	01	01	01	01	01	01	01	01	01	01	01	01	01	01	01	10
Write Plane Mask	3C5	02	03|03|03	03|03	03|03|03	03|03	03	03	01	03|03	0F	0F	0F	0F	0F	0F	0F	03	03	03	03	0F	03	0F	03	0F	0F	0F	0F	0F	0F	0F	0F	0F
Font Select	3C5	03	00|00|00	00|00	00|00|00	00|00	00	00	00	00|00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00
Memory Mode	3C5	04	02|02|02	02|02	02|02|02	02|02	02	02	06	03|02	06	06	06	06	06	06	0E	02	02	03	03	06	02	06	03	0E	0E	0E	06	06	0E	0E	0E	06
Reserved		05																																
TS STAT	3C5	06	00|00|00	00|00	00|00|00	00|00	00	00	00	00|00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00
TS Aux Mode	3C5	07	BC|BC|BC	BC|BC	BC|BC|BC	BC|BC	BC	BC	BC	BC|BC	BC	BC	BC	BC	BC	BC	BC	BC	BC	BC	BC	BC	BC	BC	BC	BC	BC	BC	BC	BC	BC	BC	BC	BC

Table 7.4.4-3 CRT Controller Registers

Register Name	Port	Index	0|0*|0+	1|1*	2|2*|2+	3|3*	4	5	6	7|7*	D	E	F	10	11	12	13	21	22	23	24	25	26	29	2A	2D	2E	30	37i	37n	2F	38i	38n	3D
CRTC Index	3D4																																		
Horiz Tot	3D5	00	2D|2D|2D	2D|2D	5F|5F|5F	5F|5F	2D	2D	5F	5F|5F	2D	5F	5F	5F	5F	5F	5F	9F	9F	97	97	5F	5F	7F	7F	5F	5F	7F	99	A1	5F	99	A1	C8
Hor Dis End	3D5	01	27|27|27	27|27	4F|4F|4F	4F|4F	27	27	4F	4F|4F	27	4F	4F	4F	4F	4F	4F	83	83	83	83	4F	4F	63	63	4F	4F	63	7F	7F	4F	7F	7F	9F
Hor Blnk Strt	3D5	02	28|28|28	28|28	50|50|50	50|50	28	28	50	50|50	28	50	50	50	50	50	50	84	84	84	84	50	50	64	64	50	50	64	7F	80	50	7F	80	AD
Hor Blnk End	3D5	03	90|90|90	90|90	82|82|82	82|82	90	90	82	82|82	90	82	82	82	82	82	82	82	82	1A	1A	82	82	02	02	82	82	02	1D	04	82	1D	04	8E
Hor Sync Strt	3D5	04	2B|2B|2B	2B|2B	55|55|55	55|55	2B	2B	54	55|55	2B	54	54	54	54	54	54	8C	8B	8D	8D	54	55	64	65	54	54	64	83	88	54	83	88	A9
Hor Sync End	3D5	05	A0|A0|A0	A0|A0	81|81|81	81|81	80	80	80	81|81	80	80	80	80	80	80	80	9F	80	16	16	80	81	17	18	80	80	17	17	9E	80	17	9E	04
Vert Tot	3D5	06	BF|BF|BF	BF|BF	BF|BF|BF	BF|BF	BF	BF	BF	BF|BF	BF	BF	BF	0B	0B	BF	FA	B2	77	7A	0B	0B	77	77	C0	0B	77	2F	26	BF	2F	26	4F
Ovrflow Low	3D5	07	1F|1F|1F	1F|1F	1F|1F|1F	1F|1F	1F	1F	1F	1F|1F	1F	1F	1F	3E	3E	1F	1F	1F	1F	1F	3E	3E	F0	F0	1F	3E	F0	F5	FD	1F	F5	FD	52
Init Row Addr	3D5	08	00|00|00	00|00	00|00|00	00|00	00	00	00	00|00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00
Max Row Addr	3D5	09	C7|4D|4F	C7|4D	C7|4D|4F	C7|4D	C1	C1	C1	4D|4F	C0	C0	40	40	40	40	41	47	48	4D	4C	40	47	60	6E	40	40	60	60	60	40	60	60	40
Cursor Strt	3D5	0A	06|0B|0D	06|0B	06|0B|0D	06|0B	00	00	00	0B|0D	00	00	00	00	00	00	00	06	07	0B	0A	00	06	00	0C	00	00	00	00	00	00	00	00	00
Cursor End	3D5	0B	07|0C|0E	07|0C	07|0C|0E	07|0C	00	00	00	0C|0E	00	00	00	00	00	00	00	07	08	0C	0B	00	07	00	0D	00	00	00	00	00	00	00	00	00
Lin Strt Mid	3D5	0C	00|00|00	00|00	00|00|00	00|00	00	00	00	00|00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00
Lin Strt Low	3D5	0D	00|00|00	00|00	00|00|00	00|00	00	00	00	00|00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00
Cursor Mid	3D5	0E	00|00|00	00|00	00|00|00	00|00	00	00	00	00|00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00
Cursor Low	3D5	0F	00|00|00	00|00	00|00|00	00|00	00	00	00	00|00	00	00	00	00	00	00	00	8E	8E	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00
Vrt Sync Strt	3D5	10	9C|83|9C	9C|83	9C|83|9C	9C|83	9C	9C	9C	83|9C	9C	9C	83	83	EA	EA	9C	E3	94	64	6E	EA	EA	60	60	83	EA	60	08	08	9C	08	08	11
Vrt Sync End	3D5	11	8E|85|8E	8E|85	8E|85|8E	83|85	8E	8E	8E	85|8E	8E	8E	85	85	8C	8C	8E	85	86	86	80	8C	8C	82	82	85	8C	82	80	8A	8E	80	8A	89
Vrt Dis End	3D5	12	8F|5D|8F	8F|5D	8F|5D|8F	8F|5D	8F	8F	8F	5F|8F	8F	8F	5D	5D	DF	DF	8F	DF	8B	5D	6B	DF	DF	57	57	5D	DF	57	FF	FF	8F	FF	FF	FF
Row Offset	3D5	13	14|14|14	14|14	28|28|28	28|28	14	14	28	28|28	14	28	28	28	28	28	28	42	42	42	28	28	32	32	50	50	64	40	40	50	80	80	50
Underline Row	3D5	14	1F|1F|1F	1F|1F	1F|1F|1F	1F|1F	00	00	00	0D|0F	00	00	0F	0F	00	00	40	1F	1F	1F	1F	00	1F	00	1F	60	60	60	00	00	60	60	60	00
Vrt Blnk Strt	3D5	15	96|63|96	96|63	96|63|96	96|63	96	96	96	63|96	96	96	63	63	E7	E7	96	E2	8F	60	6C	E7	E7	5B	5B	63	E7	5B	FF	04	96	FF	04	09
Vrt Blnk End	3D5	16	B9|BA|B9	B9|BA	B9|BA|B9	B9|BA	B9	B9	B9	BA|B9	B9	B9	BA	BA	04	04	B9	F8	AF	75	7A	04	04	75	75	BA	04	75	30	22	B9	30	22	46
CRTC Mode	3D5	17	A3|A3|A3	A3|A3	A3|A3|A3	A3|A3	A2	A2	C2	A3|A3	E3	E3	E3	E3	C3	E3	A3	A3	A3	A3	A3	E3	A3	C3	A3	AB	AB	AB	C3	C3	AB	AB	AB	C3
Line Compare	3D5	18	FF|FF|FF	FF|FF	FF|FF|FF	FF|FF	FF	FF	FF	FF|FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF
General Purps	3D5	31	00|00|00	00|00	00|00|00	00|00	00	00	00	00|00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00
Display Memory	3D5	32																28	28																
Ext'd Strt Add	3D5	33	00|00|00	00|00	00|00|00	00|00	00	00	00	00|00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00
Compat Ctrl	3D5	34																0A	0A																
Overflow Hi	3D5	35	00|00|00	00|00	00|00|00	00|00	00	00	00	00|00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	80	00	00	80	00	9B
Video Sys. 1	3D5	36																43	43																
Video Sys. 2	3D5	37																1F	1F																
Hor Overflow	3D5	3D	00|00|00	00|00	00|00|00	00|00	00	00	00	00|00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00

Table 7.4.4-4 GDC Registers

Register Name	Port	Index	0|0*|0+	1|1|1*	2|2*|2+	3|3|3*	4	5	6	7|7*	D	E	F	10	11	12	13	21	22	23	24	25	26	29	2A	2D	2E	30	37|	37n	2F	38|	38n	3D
Segment Select	3CB																																	
Segment Select	3CD																																	
GDC Index	3CE																																	
Set/Reset	3CF	00	00|00|00	00|00|00	00|00|00	00|00|00	00	00	00	00|00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00
Enabl Set/Res	3CF	01	00|00|00	00|00|00	00|00|00	00|00|00	00	00	00	00|00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00
Color Compare	3CF	02	00|00|00	00|00|00	00|00|00	00|00|00	00	00	00	00|00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00
Data Rotate	3CF	03	00|00|00	00|00|00	00|00|00	00|00|00	00	00	00	00|00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00
Read Plane Sel	3CF	04	00|00|00	00|00|00	00|00|00	00|00|00	00	00	00	00|00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00
GDC Mode	3CF	05	10|10|10	10|10|10	10|10|10	10|10|10	30	30	00	10|10	00	00	00	00	00	00	00	00	00	00	00	00	00	00	10	40	40	40	00	00	40	40	40	00
Miscellaneous	3CF	06	0E|0E|0E	0E|0E|0E	0E|0E|0E	0E|0E|0E	0F	0F	0D	0A|0A	05	05	05	05	05	05	0E	0E	0E	0E	0E	05	0E	05	0E	05	05	05	05	05	05	05	05	05
Color Care	3CF	07	00|00|00	00|00|00	00|00|00	00|00|00	00	00	00	00|00	0F	0F	05	0F	01	0F	0F	00	00	00	00	0F	0F	0F	00	0F	0F	0F	0F	0F	0F	0F	0F	0F
Bit Mask	3CF	08	FF|FF|FF	FF|FF|FF	FF|FF|FF	FF|FF|FF	FF	FF	FF	FF|FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF

Table 7.4.4-5 ATC Indexed Registers

Register Name	Port	Index	0|0*|0+	1|1*	2|2*|2+	3|3*	4	5	6	7|7*	D	E	F	10	11	12	13	21	22	23	24	25	26	29	2A	2D	2E	30	37|	37n	2F	38|	38n	3D
ATC Index	R/W 3C0																																	
All =	R:3C1/W:3C0																																	
Palette		00	00|00|00	00|00	00|00|00	00|00	00	00	00	00|00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00
Palette		01	01|01|01	01|01	01|01|01	01|01	13	13	17	08|08	01	01	08	01	3F	01	01	01	01	01	01	01	01	01	01	01	01	01	01	01	01	01	01	01
Palette		02	02|02|02	02|02	02|02|02	02|02	15	15	17	08|08	02	02	00	02	3F	02	02	02	02	02	02	02	02	02	02	02	02	02	02	02	02	02	02	02
Palette		03	03|03|03	03|03	03|03|03	03|03	17	17	17	08|08	03	03	00	03	3F	03	03	03	03	03	03	03	03	03	03	03	03	03	03	03	03	03	03	03
Palette		04	04|04|04	04|04	04|04|04	04|04	02	02	17	08|08	04	04	18	04	3F	04	04	04	04	04	04	04	04	04	04	04	04	04	04	04	04	04	04	04
Palette		05	05|05|05	05|05	05|05|05	05|05	04	04	17	08|08	05	05	18	05	3F	05	05	05	05	05	05	05	05	05	05	05	05	05	05	05	05	05	05	05
Palette		06	06|14|14	06|14	06|14|14	06|14	06	06	17	08|08	06	06	00	14	3F	14	06	14	14	14	14	14	14	14	14	06	06	06	14	14	06	06	06	14
Palette		07	07|07|07	07|07	07|07|07	07|07	07	07	17	08|08	07	07	00	07	3F	07	07	07	07	07	07	07	07	07	07	07	07	07	07	07	07	07	07	07
Palette		08	10|38|38	10|38	10|38|38	10|38	10	10	17	10|10	10	10	00	38	3F	38	08	38	38	38	38	38	38	38	38	08	08	08	38	38	08	08	08	38
Palette		09	11|39|39	11|39	11|39|39	11|39	11	11	17	18|18	11	11	08	39	3F	39	09	39	39	39	39	39	39	39	39	09	09	09	39	39	09	09	09	39
Palette		0A	12|3A|3A	12|3A	12|3A|3A	12|3A	12	12	17	18|18	12	12	00	3A	3F	3A	0A	3A	3A	3A	3A	3A	3A	3A	3A	0A	0A	0A	3A	3A	0A	0A	0A	3A
Palette		0B	13|3B|3B	13|3B	13|3B|3B	13|3B	13	13	17	18|18	13	13	00	3B	3F	3B	0B	3B	3B	3B	3B	3B	3B	3B	3B	0B	0B	0B	3B	3B	0B	0B	0B	3B
Palette		0C	14|3C|3C	14|3C	14|3C|3C	14|3C	14	14	17	18|18	14	14	00	3C	3F	3C	0C	3C	3C	3C	3C	3C	3C	3C	3C	0C	0C	0C	3C	3C	0C	0C	0C	3C
Palette		0D	15|3D|3D	15|3D	15|3D|3D	15|3D	15	15	17	18|18	15	15	18	3D	3F	3D	0D	3D	3D	3D	3D	3D	3D	3D	3D	0D	0D	0D	3D	3D	0D	0D	0D	3D
Palette		0E	16|3E|3E	16|3E	16|3E|3E	16|3E	16	16	17	18|18	16	16	00	3E	3F	3E	0E	3E	3E	3E	3E	3E	3E	3E	3E	0E	0E	0E	3E	3E	0E	0E	0E	3E
Palette		0F	17|3F|3F	17|3F	17|3F|3F	17|3F	17	17	17	18|18	17	17	00	3F	3F	3F	0F	3F	3F	3F	3F	3F	3F	3F	3F	0F	0F	0F	3F	3F	0F	0F	0F	3F
Mode Ctrl		10	08|08|0C	08|08	08|08|0C	08|08	01	01	01	0E|0E	01	01	0B	01	01	01	41	08	08	08	08	01	0C	01	08	01	01	01	01	01	01	01	01	01	01
Overscan Clr		11	00|00|00	00|00	00|00|00	00|00	00	00	00	00|00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00
Clr Plan En		12	0F|0F|0F	0F|0F	0F|0F|0F	0F|0F	03	03	01	0F|0F	0F	0F	05	0F	0F	0F	0F	0F	0F	0F	0F	0F	0F	0F	0F	0F	0F	0F	0F	0F	0F	0F	0F	0F
Hor Pix Pan		13	00|00|08	00|00	00|00|08	00|00	00	00	00	08|08	00	00	00	00	00	00	00	00	00	00	00	00	08	00	00	00	00	00	00	00	00	00	00	00
Color Select		14	00|00|00	00|00	00|00|00	00|00	00	00	00	00|00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00
Miscellaneous		16	00|00|00	00|00	00|00|00	00|00	00	00	00	00|00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00
Miscellaneous 1		17	00|00|00	00|00	00|00|00	00|00	00	00	00	00|00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00

Table 7.4.4-6 General Registers

Register Name	Port	Index	0|0	1|1	2|2	3|3	4	5	6	7*	D	E	F	10
Misc Output	3C2	-	23|A7	23|A7	23|A7	23|A7	23	23	23	A6	23	23	A2	A7
Input Stat 0	3C2	00	--|--	--|--	--|--	--|--	-	-	-	-	-	-	-	-
Input Stat 1	3CA	01	--|--	--|--	--|--	--|--	-	-	-	-	-	-	-	-
Feature Ctlr	3CA	02	00|00	00|00	00|00	00|00	0	0	0	0	0	0	0	0

Table 7.4.4-7 Timing Sequencer Registers

Register Name	Port	Index	0|0	1|1	2|2	3|3	4	5	6	7*	D	E	F	10
TS Index	3C4	-	--|--	--|--	--|--	--|--	-	-	-	-	-	-	-	-
Sync Reset	3C5	00	03|03	03|03	03|03	03|03	03	03	03	03	03	03	03	03
TS Mode	3C5	01	0B|0B	0B|0B	01|01	01|01	0B	0B	01	00	0B	01	01	01
Wrte Plane Mask	3C5	02	03|03	03|03	03|03	03|03	03	03	01	03	0F	0F	0F	0F
Font Select	3C5	03	00|00	00|00	00|00	00|00	00	00	00	00	00	00	00	00
Memory Mode	3C5	04	03|03	03|03	03|03	03|03	02	02	06	03	06	06	06	06

Table 7.4.4-8 CRT Controller Registers

Register Name	Port	Index	0|0	1|1	2|2	3|3	4	5	6	7*	D	E	F	10
CRTC Index	3C4	-	--|--	--|--	--|--	--|--	-	-	-	-	-	-	-	-
Hor Total	3C5	00	37/2D	37/2D	70/5B	70/5B	37	37	70	60	37	70	60	5B
Hor Display End	3C5	01	27/27	27/27	4F/4F	4F/4F	27	27	4F	4F	27	4F	4F	4F
Hor Blnk Start	3C5	02	2D/2B	2D/2B	5C/53	5C/53	2D	2D	59	56	2D	59	56	53
Hor Blnk End	3C5	03	37/2D	37/2D	2F/37	2F/37	37	37	2D	3A	37	2D	3A	37
Hor Sync Strt	3C5	04	31/28	31/28	5F/51	5F/51	30	30	5E	51	30	5E	50	52
Hor Sync End	3C5	05	15/6D	15/6D	07/5B	07/5B	14	14	06	60	14	06	60	00
Vert Total	3C5	06	04|6C	04|6C	04|6C	04|6C	04	04	04	70	04	04	70	6C
Ovrlflw Low	3C5	07	11|1F	11|1F	11|1F	11|1F	11	11	11	1F	11	11	1F	1F
Init Row Addr	3C5	08	00|00	00|00	00|00	00|00	00	00	00	00	00	00	00	00
Max Row Addr	3C5	09	07|0D	07|0D	07|0D	07|0D	01	01	01	0D	00	00	00	00
Cursor Strt	3C5	0A	06|06	06|06	06|06	06|06	00	00	00	0B	00	00	00	00
Cursor End	3C5	0B	07|07	07|07	07|07	07|07	00	00	00	0C	00	00	00	00
Lin Strt Mid	3C5	0C	00|00	00|00	00|00	00|00	00	00	00	00	00	00	00	00
Lin Strt Low	3C5	0D	00|00	00|00	00|00	00|00	00	00	00	00	00	00	00	00
Cursor Mid	3C5	0E	00|00	00|00	00|00	00|00	00	00	00	00	00	00	00	00
Cursor Low	3C5	0F	00|00	00|00	00|00	00|00	00	00	00	00	00	00	00	00
Vert Sync Strt	3C5	10	E1|5E	E1|5E	E1|5E	E1|5E	E1	E1	E0	5E	E1	E0	5E	5E
Vert Sync End	3C5	11	24|2B	24|2B	24|2B	24|2B	24	24	23	2E	24	23	2E	2B
Vert Display End	3C5	12	C7|5D	C7|5D	C7|5D	C7|5D	C7	C7	C7	5D	C7	C7	5D	5D
Row Offset	3C5	13	14|14	14|14	28|28	28|28	14	14	28	28	14	28	28	28
Underline Row	3C5	14	08|0F	08|0F	08|0F	08|0F	00	00	00	00	00	00	0D	0F
Vert Blnk Strt	3C5	15	E0|5E	E0|5E	E0|5E	E0|5E	E0	E0	DF	5E	E0	DF	5E	5F
Vert Blnk End	3C5	16	F0|0A	F0|0A	F0|0A	F0|0A	F0	F0	EF	0E	F0	EF	0E	0A
CRTC Mode	3C5	17	A3|A3	A3|A3	A3|A3	A3|A3	A2	A2	C2	A3	E3	E3	E3	E3
Line Compare	3C5	18	FF|FF	FF|FF	FF|FF	FF|FF	FF	FF	FF	FF	FF	FF	FF	FF

Table 7.4.4-9 GDC Registers

Register Name	Port	Index	0|0	1|1	2|2	3|3	4	5	6	7	D	E	F	10
GDC Index	3CE													
Set/Reset	3CF	00	00|00	00|00	00|00	00|00	00	00	00	00	00	00	00	00
Enabl Set/Res	3CF	01	00|00	00|00	00|00	00|00	00	00	00	00	00	00	00	00
Color Compare	3CF	02	00|00	00|00	00|00	00|00	00	00	00	00	00	00	00	00
Data Rotate	3CF	03	00|00	00|00	00|00	00|00	00	00	00	00	00	00	00	00
Read Plane Sel	3CF	04	00|00	00|00	00|00	00|00	00	00	00	00	00	00	00	00
GDC Mode	3CF	05	10|10	10|10	10|10	10|10	30	30	00	10	00	00	00	00
Miscellaneous	3CF	06	0E|0E	0E|0E	0E|0E	0E|0E	0F	0F	0D	0A	05	05	05	05
Color Care	3CF	07	00|00	00|00	00|00	00|00	00	00	00	00	0F	0F	0F	0F
Bit Mask	3CF	08	FF|FF	FF|FF	FF|FF	FF|FF	FF	FF	FF	FF	FF	FF	FF	FF

Two versions (CGA/EGA) of modes 0-3 are for 200, and 350 scan lines, respectively.

Table 7.4.4-10 ATC Registers

Register Name	Port	Index	0|0	1|1	2|2	3|3	4	5	6	7*	D	E	F	10*
ATC Index	R/W 3C0	-	--|--	--|--	--|--	--|--	-	-	-	-	-	-	-	-
All =	R:3C1\W:3C0													
Palette		00	00|00	00|00	00|00	00|00	00	00	00	00	00	00	00	00
Palette		01	01|01	01|01	01|01	01|01	13	13	17	08	01	01	08	01
Palette		02	02|02	02|02	02|02	02|02	15	15	17	08	02	02	00	02
Palette		03	03|03	03|03	03|03	03|03	17	17	17	08	03	03	00	03
Palette		04	04|04	04|04	04|04	04|04	02	02	17	08	04	04	18	04
Palette		05	05|05	05|05	05|05	05|05	04	04	17	08	05	05	18	05
Palette		06	06|14	06|14	06|14	06|14	06	06	17	08	06	06	00	14
Palette		07	07|07	07|07	07|07	07|07	07	07	17	08	07	07	00	07
Palette		08	10|38	10|38	10|38	10|38	10	10	17	10	10	10	00	38
Palette		09	11|39	11|39	11|39	11|39	11	11	17	18	11	11	08	39
Palette		0A	12|3A	12|3A	12|3A	12|3A	12	12	17	18	12	12	00	3A
Palette		0B	13|3B	13|3B	13|3B	13|3B	13	13	17	18	13	13	00	3B
Palette		0C	14|3C	14|3C	14|3C	14|3C	14	14	17	18	14	14	00	3C
Palette		0D	15|3D	15|3D	15|3D	15|3D	15	15	17	18	15	15	18	3D
Palette		0E	16|3E	16|3E	16|3E	16|3E	16	16	17	18	16	16	00	3E
Palette		0F	17|3F	17|3F	17|3F	17|3F	17	17	17	18	17	17	00	3F
Mode Control		10	08|08	08|08	08|08	08|08	01	01	01	0E	01	01	0B	01
Overscan Color		11	00|00	00|00	00|00	00|00	00	00	00	00	00	00	00	00
Color Plane Enble		12	0F|0F	0F|0F	0F|0F	0F|0F	03	03	01	0F	0F	0F	05	0F
Hor Pixel Pan		13	00|00	00|00	00|00	00|00	00	00	00	08	00	00	00	00

8. Programming Interface

The functions that are supported as program calls to the adapter are listed in this section. These calls are made through software interrupt 10H (INT 10H).

Functions are identified by the content of the AH register at the time of the call; in some cases, the AH register identifies a group of similar functions and the AL register identifies the specific function. The primary functions are:

Interrupt 10 Functions

(AH)	Function
00H	Mode Set
01H	Set Cursor Type
02H	Set Cursor Position
03H	Read Cursor Position
04H	Read Light Pen Position (not supported)
05H	Select Active Display Page
06H	Scroll Active Page Up
07H	Scroll Active Page Down
08H	Read Character(s) at Current Cursor Position
09H	Write Character(s) at Current Cursor Position
0AH	Write Character(s) Only at Current Cursor Position
0BH	Set Color Palette
0CH	Write Dot
0DH	Read Dot
0EH	Write Teletypewriter to Active Page
0FH	Return Current Video State
10H	Set Palette Registers
11H	Character Generator Routine
12H	Alternate Select
13H	Write String
1AH	Display Combination Code
1BH	Return Functionality/State Information
1CH	Save/Restore
14H	Reserved
15H	Reserved
16H	Reserved
17H	Reserved
18H	Reserved
19H	Reserved

8.1 BIOS Function Calls

All values in hexadecimal unless otherwise noted.

AH=0 Set video mode.

Input:
AL=mode to set (see table below).
Output:
none.

Table 8.1-1 ET4000/W32p Modes

	Mode	Type	Colors/ Shades	Alpha Format	Buffer Start	Box Size	Max. Pag.	Display Size	Vid Clk (MHZ)	H Freq (KHz)	V Freq (Hz)
	0	A/N	16/256K	40x25	B8000	8x8	8	320x200	25.175	31.50	70.00
	0*	A/N	16/256K	40x25	B8000	8x14	8	320x350	25.175	31.50	70.00
	0+	A/N	16/256K	40x25	B8000	9x16	8	360x400	28.322	31.50	70.00
	1	A/N	16/256K	40x25	B8000	8x8	8	320x200	25.175	31.50	70.00
	1*	A/N	16/256K	40x25	B8000	8x14	8	320x350	25.175	31.50	70.00
	1+	A/N	16/256K	40x25	B8000	9x16	8	360x400	28.322	31.50	70.00
	2	A/N	16/256K	80x25	B8000	8x8	8	640x200	25.175	31.50	70.00
	2*	A/N	16/256K	80x25	B8000	8x14	8	640x350	25.175	31.50	70.00
	2+	A/N	16/256K	80x25	B8000	9x16	8	720x400	28.322	31.50	70.00
	3	A/N	16/256K	80x25	B8000	8x8	8	640x200	25.175	31.50	70.00
	3*	A/N	16/256K	80x25	B8000	8x14	8	640x350	25.175	31.50	70.00
	3+	A/N	16/256K	80x25	B8000	9x16	8	720x400	28.322	31.50	70.00
	4	APA	4/256K	40x25	B8000	8x8	1	320x200	25.175	31.50	70.00
	5	APA	4/256K	40x25	B8000	8x8	1	320x200	25.175	31.50	70.00
	6	APA	2/256K	80x25	B8000	8x8	1	640x200	25.175	31.50	70.00
	7	A/N	Mono	80x25	B0000	9x14	8	720x350	28.322	31.50	70.00
	7+	A/N	Mono	80x25	B0000	9x16	8	720x400	28.322	31.50	70.00
	D	APA	16/256K	40x25	A0000	8x8	8	320x200	25.175	31.50	70.00
	E	APA	16/256K	80x25	A0000	8x8	4	640x200	25.175	31.50	70.00
	F	APA	Mono	80x25	A0000	8x14	2	640x350	25.175	31.50	70.00
	10	APA	16/256K	80x25	A0000	8x14	2	640x350	25.175	31.50	70.00
	11	APA	2/256K	80x30	A0000	8x16	1	640x480	25.175	31.50	60.00
	11 72h	APA	2/256K	80x30	A0000	8x16	1	640x480	31.500	37.86	72.81
	12	APA	16/256K	80x30	A0000	8x16	1	640x480	25.175	31.50	60.00
	12 72h	APA	16/256K	80x30	A0000	8x16	1	640x480	31.500	37.86	72.81
	13	APA	256/256K	40x25	A0000	8x8	1	320x200	25.175	31.50	70.00
#	13	APA	32K or 64K	40x25	A0000	8x8	1	320x200	50.350	31.50	70.00
	21	A/N	16/256K	132x60	B8000	8x8	2	1056x480	40.000	31.06	60.00
	22	A/N	16/256K	132x44	B8000	8x9	2	1056x396	40.000	31.06	70.00
	23	A/N	16/256K	132x25	B8000	8x16	4	1056x400	40.000	31.06	70.00
	24	A/N	16/256K	132x28	B8000	8x14	4	1056x392	40.000	31.06	70.00
	25	APA	16/256K	80x60	A0000	8x8	1	640x480	25.175	31.50	60.00
	25 72h	APA	16/256K	80x60	A0000	8x8	1	640x480	31.500	37.86	72.81
	26	A/N	16/256K	80x60	B8000	9x8	2	720x480	28.322	31.50	60.00
	29 35k	APA	16/256K	100x37	A0000	8x16	1	800x600	36.000	35.50	56.00
	29 38k	APA	16/256K	100x37	A0000	8x16	1	800x600	40.000	38.00	60.00
	29 48k	APA	16/256K	100x37	A0000	8x16	1	800x600	50.350	48.40	72.70
	2A 35k	A/N	16/256K	100x40	B8000	8x15	4	800x600	36.000	35.50	56.00
	2A 38k	A/N	16/256K	100x40	B8000	8x15	4	800x600	40.000	38.00	60.00
	2A 48k	A/N	16/256K	100x40	B8000	8x15	4	800x600	50.350	48.40	72.70
	2D	APA	256/256	80x25	A0000	8x14	1	640x350	25.175	31.50	70.00
#	2D	APA	32K or 64K	80x25	A0000	8x14	1	640x350	50.350	31.50	70.00
	2E	APA	256/256K	80x30	A0000	8x16	1	640x480	25.175	31.50	60.00
	2E 72h	APA	256/256K	80x30	A0000	8x16	1	640x480	31.500	37.86	72.81
**#	2E	APA	32K or 64K	80x30	A0000	8x16	1	640x480	50.350	31.50	60.00
**%	2E	APA	16.8mil	80x30	A0000	8x16	1	640x480	75.000	31.50	60.00
	2F	APA	256/256K	80x25	A0000	8x16	1	640x400	25.175	31.50	70.00
#	2F	APA	32K or 64K	80x25	A0000	8x16	1	640x400	50.350	31.50	70.00
	30 35k	APA	256/256K	100x37	A0000	8x16	1	800x600	36.000	35.50	56.00
	30 38k	APA	256/256K	100x37	A0000	8x16	1	800x600	40.000	38.00	60.00
	30 48k	APA	256/256K	100x37	A0000	8x16	1	800x600	50.350	48.40	72.70
**#	30 35k	APA	32K or 64K	100x37	A0000	8x16	1	800x600	72.000	35.50	56.00

	Mode	Type	Colors/ Shades	Alpha Format	Buffer Start	Box Size	Max. Pag.	Display Size	Vid Clk (MHZ)	H Freq (KHz)	V Freq (Hz)
**#	30 38k	APA	32K or 64K	100x37	A0000	8x16	1	800x600	80.000	38.00	60.00
**#&	30 48k	APA	32K or 64K	100x37	A0000	8x16	1	800x600	50.350	48.40	72.70
^%&	30 35k	APA	16.8mil	100x37	A0000	8x16	1	800x600	56.644	35.40	55.84
^%&	30 38k	APA	16.8mil	100x37	A0000	8x16	1	800x600	63.000	37.86	59.81
^%&	30 48k	APA	16.8mil	100x37	A0000	8x16	1	800x600	75.000	48.08	72.19
	37i 45m	APA	16/256K	128x48	A0000	8x16	1	1024x768	44.900	35.50	87.00
	37n 65m	APA	16/256K	128x48	A0000	8x16	1	1024x768	65.000	49.00	60.50
	37n 72m	APA	16/256K	128x48	A0000	8x16	1	1024x768	75.000	56.48	70.07
**	38i 45m	APA	256/256K	128x48	A0000	8x16	1	1024x768	44.900	35.50	87.00
**	38n 65m	APA	256/256K	128x48	A0000	8x16	1	1024x768	65.000	49.00	60.50
**	38 72m	APA	256/256K	128x48	A0000	8x16	1	1024x768	75.000	56.48	70.07
^#	38i 45m	APA	32K or 64K	128x48	A0000	8x16	1	1024x768	89.800	35.50	87.00
^#&	38i 45m	APA	32K or 64K	128x48	A0000	8x16	1	1024x768	44.900	35.50	87.00
^#&	38n 65m	APA	32K or 64K	128x48	A0000	8x16	1	1024x768	65.000	49.00	60.50
^#&	38 72m	APA	32K or 64K	128x48	A0000	8x16	1	1024x768	75.000	56.48	70.07
**	3Di	APA	16/256K	160x64	A0000	8x16	1	1280x1024	80.000	48.10	87.00
^	3Fi	APA	256/256K	160x64	A0000	8x16	1	1280x1024	80.000	48.10	87.00
^&	3Fn 60h	APA	256/256K	160x64	A0000	8x16	1	1280x1024	56.644	64.37	60.33
^&	3Fn 70h	APA	256/256K	160x64	A0000	8x16	1	1280x1024	63.000	75.00	70.03

NOTE: ALL BIOS MODES MAY NOT BE AVAILABLE ON All ET4000/W32p BASED DESIGNS.

A/N = Alphanumeric modes (text)
APA = All Points Addressable modes (graphics)
i = interlaced modesn = noninterlaced modes
* Extended Graphics Adapter text modes with 350 scan lines.
+ 9x16 character cell enhanced text modes with 400 scan lines.
** = modes require 1MB display memory.
^ = modes require more than 1MB display memory.
# = 32,768 or 65536 colors with capable DAC.
& = modes require a W32p and 16-bit pixel port DAC capable of double 8-bit indexed pixel output.
256-color modes require a DAC capable of double 8-bit indexed pixel output.
16.8-million color modes require a DAC capable of packed 24-bit per pixel output.
% = 16.8 million colors with capable TrueColor DAC.

Note that there are a number of distinct text modes available including 132-column monochrome text modes.

NOTES:

1. AL bit 7 can be 0 or 1. When set to 1, the MODE SET function does not clear the display buffer.
2. Default modes are 3+ for color monitor and 7+ for monochrome monitor.
3. Modes 0 through 6 emulate IBM Color Graphics Adapter support.
4. Modes 0, 2, and 5 are identical to modes 1, 3, and 4 respectively.
5. There is no hardware cursor in graphics (APA) modes. Altering the hardware cursor type has no effect in these modes.
6. Selecting the number of scan lines in alphanumeric modes is detailed under "(BL) = 30H, Select Scan Lines for Alphanumeric Modes."
7. Use of the equipment flags variable at address 0:410 (applicable bits are <5,4>):

   - Binary XX11 XXXX = monochrome
   - Binary XX10 XXXX = color

If there is more than one video adapter in the system, the equipment flag setting at the time of the set mode call determines if the mode should be set in the color or monochrome adapter. If necessary, color modes will be converted to monochrome mode 7 and monochrome modes to color mode 3. If there is only one adapter, then in EGA mode, the equipment flag forces a color or monochrome mode to be set, with conversion if necessary. In VGA mode, the equipment flag automatically gets changed to agree with the mode being set.

AH=1 Set cursor type (start and stop scan lines)

Input:

CH=start scan line for cursor.
CL=end scan line for cursor.

Output:

none.

NOTE: Only bits 0 through 4 should be set.

AH=2 Set cursor position.

Input:

BH=page for which cursor is to be set.
DH=row position cursor is to be set to.
DL=column position cursor is to be set to.

Output:

none.

NOTE: (0,0) is upper left of screen.

AH=3 Read cursor position.

Input:

BH=page for which cursor is to be read.

Output:

CH=current start scan line for cursor.
CL=current stop scan line for cursor.
DH=row position of cursor in selected page.
DL=column position of cursor in selected page.

AH=4 Read light pen position

Input:

none.

Output:

AH=0 then light pen switch not activated, return values invalid.
1 then light pen switch activated, valid values returned.
BX=pixel column.
CH=raster line.
CX=raster line (new graphics modes).
DH=row of character light pen position.
DL=column of character light pen position.

AH=5 Select active page.

Input:

AL=page to select as active page.

Output:

none.

AH=6 Scroll up active page.

Input:

AL=number of lines rows are to move up.
0 means blank window.
BH=attribute used to fill blank line or lines at bottom.
CH=row of upper left corner of scroll window.
CL=column of upper left corner of scroll window.
DH=row of lower right corner of scroll window.
DL=column of lower right corner of scroll window.

Output:

none.

AH=7 Scroll down active page.

Input:

AL=number of lines rows are to move down.
0 means blank window.
BH=attribute used to fill blank line or lines at top.
CH=row of upper left corner of scroll window.
CL=column of upper left corner of scroll window.
DH=row of lower right corner of scroll window.
DL=column of lower right corner of scroll window.

Output:

none.

AH=8 Read character and attribute at cursor position.

Input:

BH=page to read from.

Output:

AH=attribute of character at cursor position.
AL=character read from cursor position.

NOTE: Attribute valid in text modes only. Only characters drawn in white matched in graphics modes.

AH=9 Write character and attribute at cursor position.

Input:

AL=character to write at cursor position.
BH=page to write character and attribute to.
BL=attribute to write character with in text mode.
=foreground color in graphics mode.
CX=number of times to write character and attribute.

Output:

none.

Note: If bit 7 of BL is 1 in graphics mode, then the character is XOR'd into video memory, else the character displaces the previous contents of video memory. (XOR not valid in 256 color modes.)

Note: In 256 color modes, the value passed in BH is used as the background color.

AH=0A Write character only at cursor position.

Input:

AL=character to write at cursor position.
BH=page to write character and attribute to.
BL=(in graphics modes only) foreground color for character.
CX=number of times to write character and attribute.

Output:

none.

Note: See previous notes for function AH=9.

AH=0B Color select for color/graphics adapter compatible modes.

Input:

BH=0 means set the background color specified by BL.
<>0 means set the palette specified by BL.
BL=color value to be used:

- When setting the background color, BL selects any of the 16 colors with a value of 0-15 with bits 0-3.
- When selecting the palette, BL operates as follows:
  bit 0=0 selects palette 0 (green/red/brown).
  bit 0=1 selects palette 1 (cyan/magenta/white).

Output:

none.

Note: In text modes, the set background function sets the border color only. In graphics modes, the set background function sets both the border and background colors.

Note: This function is implemented via emulation, since the EGA does not have the same color registers as the color/graphics adapter.

Note: Actual operation is to set palette register 0 for background, palette register 11h for overscan, and palette registers 1-3 for palette colors 1-3. Palette registers are set in any graphics mode, although this was valid only in 320x200 graphics mode on the color/graphics adapter.

AH=0C Draw graphics pixel.

Input:

AL=color (actually attribute that goes to the palette RAM) to draw pixel in.
BH=page to draw pixel in.
CX=screen column to write pixel at.
DX=screen row to write pixel at.

Output:

none.

Note: If bit 7 of AL is 1, then the pixel is XOR'd with the contents of video memory (except in 256 color modes).

AH=0D Read graphics pixel color (actually attribute that goes to the palette RAM).

Input:

BH=page to read pixel from.
CX=screen column to read pixel from.
DX=screen row to read pixel from.

Output:

AL=pixel value read (attribute of pixel).

NOTE: Interpretation of value returned depends on graphics mode in effect.

AH=0E Write TTY.

Input:

AL=character to write.
BL=color to draw character in graphics mode.

Output:

none.

NOTE: Carriage return, backspace, line feed, and bell are commands, not displayed characters. Cursor is moved to the right after character is displayed, with wrap and scroll at right margin of screen.

AH=0F Return video information.

Input:

none.

Output:

AL=video mode in effect.
AH=text columns supported in current mode.
BH=active display page.

NOTE: Bit 7 of AL is set to 1 if the regen buffer was not cleared when the mode was set.

AH=10 Set EGA palette registers.

AL=0 set color for a single palette register.

Input:

BH=color to set palette register to.
BL=palette register to set color of.

Output:

none.

AL=1 set color for overscan (border color) register.

Input:

BH=color to set overscan register to.

Output:

none.

AL=2 set colors for all 16 palette and the overscan registers.

Input:

ES:DX=address of table organized as follows:

bytes 0-15=colors for palette registers 0-15.

byte 16=color for overscan register.

Output:

none.

AL=3 select interpretation of intensity/blink attribute bit.

Input:

BL=0 select high intensity background.

1 select blinking.

Output:

none.

AL=4 reserved.

AL=5 reserved.

AL=6 reserved.

AL=7 read individual palette register.

Input:

BL=palette register to read (range 0 to 15).

Output:

BH=value read.

AL=8 read overscan register.

Input:

none.

Output:

BH=value read.

AL=9 read all palette registers and overscan.

Input:

ES:DX points to 17 byte table area.

Output:

bytes 0-15 = palette values.

byte 16 = overscan value.

AL=10h set individual color register (external palette).

Input:

BX=color register to set.

DH=red value to set.

CH=green value to set.

CL=blue value to set.

Output:

none.

AL=11h reserved.

AL=12h set block of color registers.

Input:

ES:DX=pointer to table of color values in RGB format (i.e..3 bytes for each entry).
BX=starting index.
CX=number of color registers to set.

Output:

none.

AL=13h select color page.
BL=00 select paging mode.

Input:

BH=paging mode.
0 - selects 4 register pages of 64 registers.
1 - selects 16 register pages of 16 registers.

Output:

none.

BL=01 select page.

Input:

BH=page value (0 to nn, where nn = 3 in page mode 0 and nn = 15 in page mode 1).

AL=14h reserved.

AL=15h read individual color register.

Input:

BX=color register to read.

Output:

DH=red value read.
CH=green value read.
CL=blue value read.

AL=16h reserved.

AL=17h read block of color registers.

Input:

ES:DX=pointer to destination for RGB table (3 bytes/entry).
BX=starting index.
CX=number of color registers read.

Output:

(ES:DX)=table.

AL=18h reserved.

AL=19h reserved.

AL=1Ah read color page state.

Input:

none

Output:

BL=current paging mode.
BH=current page.

AL=1Bh sum colors to gray shades (VGA only).
(This call reads R, G, and B values found in external palette ram and performs a weighted sum (30% red, 59% green, and 11% blue), then writes the result into each R, G, and B component of color register (original data is overwritten).

Input:

BX=starting index.
CX=number of color registers to sum.

AH=10, AL=F0 Set HiColor mode.

This call will attempt to set a 16- or 24-bit/pixel (HiColor) mode with the same X and Y dimensions as the specified 256-color mode. The call will fail if there is not a HiColor DAC present, if the specified mode is invalid, or there are memory or other hardware limitations. Note: 16-bit/pixel defaults to 5/5/5 format*.

Input:

BL=FF, BH=mode no.: Set RGB 24-bit/pixel mode (format 3); valid (256-color) mode numbers are: 2E, 30.
BL=FE, BH=mode no.: Set BGR 24-bit/pixel mode (format 4); valid (256-color) mode numbers are: 2E, 30.
BL=mode no.: Set 16-bit/pixel mode; valid (256-color) mode numbers are: 13, 2D; 2E; 2F; 30, 38 (set bit 7 of mode no.=1 to not clear memory).

Output:

AL=10
AH=0 if succeeded.
< > 0 if failed (not equal to 0).

AH=10, AL=F1 Get DAC type

Input:

none.

Output:

AL =10
AH =0
BL =0 normal DAC.
=1 Sierra SC11481, SC11486, SC11488 high-color DACs[1].
=2 Sierra SC11485, SC11487, SC11489 high-color DACs[2].
=3 AT&T ATT20C491 high-color DACs[3].
=4 Cirrus CL-GD5200 (ACUMOS ADAC1) high-color DACs[3].
=5 Sierra SC15025, SC15026 high-color DACs[4].
=6 INMOS IM5G174 high-color DACs[3].
=7 Music MU9C1880 high-color DACs[5].

AH=10, AL=F2 Get/Set HiColor format.

NOTE: Must be in 16-bit/pixel HiColor mode in order to set format.

Input:

BL =Code for HiColor format.
0=Get format.
1=Set format (5/5/5*).
2=Set format (5/6/5*).

Output:

AL=10
AH =0 if succeeded.
<> 0 if failed (not equal to 0).
BL =0 if wasn't in HiColor mode.
=1 if now in format (5/5/5*).
=2 if now in format (5/6/5*).
=3 if now in format (8R/8G/8B*).
=4 if now in format (8B/8G/8R*).

* Formats:

(1)	Bit 15 reserved	<14:10> RED (5)	<9:5> GREEN (5)	<4:0> BLUE (5)
(2)		<15:11> RED (5)	<10:5> GREEN (6)	<4:0> BLUE (5)
(3)		<23:16> RED (8)	<15:8> GREEN (8)	<7:0> BLUE (8)
(4)		<23:16> BLUE (8)	<15:8> GREEN (8)	<7:0> RED (8)

[1]—supports Format (1)
[2]—supports Formats (1) and (2)
[3]—supports Formats (1), (2), and (3)
[4]—supports Formats (1), (2), (3), and (4)
[5]—supports Formats (1), (2), and (4)

AH=11 Font interface.

AL=0 load user font into soft font (text mode).

Input:

BH=# of bytes per character.
BL=# of soft font to load font into.
CX=# of characters to store.
DX=offset into table of first character to store.
ES:BP=pointer to font to load.

Output:

none.

AL=1 load ROM monochrome font into soft font (text mode).

Input:

BL=# of soft font to load font into.

Output:

none.

AL=2 load ROM 8x8 double dot font into soft font (text mode).

Input:

BL=# of soft font to load font into.

Output:

none.

AL=3 select fonts displayed (text mode).

Input:

BL=specification for high/low attribute bit 3: bits 4,1,0=soft font # selected when attr bit 3 is 0. Bits 5,3,2=soft font # selected when attr bit 3 is 1.

Output:

none.

AL=4 load ROM 8x16 font into soft font (text mode).

Input:

BL=# of soft font to load font into.

Output:

none.

Note: The following functions AL=1X are the same as AL=0X, except:

- The active page must be zero.
- The char_height variable will be recalculated.
- The crt_rows variable will be recalculated as: INT((200 | 350 | 400) / char_height) - 1.
- regen_length will be recalculated as: (crt_rows + 1) * crt_columns * 2.
- The CRTC will be reprogrammed as: Max scan line = char_height - 1.

Cursor start = char_height - 2.
Cursor end = char_height - 1 (cursor_type set via set_cursor_type BIOS function).
Vert disp end = ((crt_rows + 1) * char_height) - 1 [char_height*2 above if double scan].
Underline = char_height - 1 (mono. modes only).

AL=10 load user font into soft font (text mode).

Input:

BH=# of bytes per character.
BL=# of soft font to load font into.
CX=# of characters to store.
DX=offset into table of first character to store.
ES:BP=pointer to font to load.

Output:

none.

AL=11 load ROM monochrome font into soft font (text mode).

Input:

BL=# of soft font to load font into.

Output:

none.

AL=12 load ROM 8x8 double dot font into soft font (text mode).

Input:

BL=# of soft font to load font into.

Output:

none.

AL=14 load ROM 8x16 font into soft font (text mode).

Input:

BL=# of soft font to load font into.

Output:

none.

AL=20 set user font chars 128-255 for color/graphics adapter compatible modes (graphics).

Input:

ES:BP=pointer to font to load.

Output:

none.

AL=21 set user font (graphics).

Input:

BL=# of rows on screen, as follows:
0 then DL=user specified # rows.
DL=# rows.
1 then 14 rows.
2 then 25 rows.
3 then 43 rows.
CX=character height.
ES:BP=pointer to font to load.

Output:

none.

AL=22 set ROM 8x14 font (graphics).

Input:

BL=# of rows on screen, as follows:
0 then DL=user specified # rows.
DL=# rows.
1 then 14 rows.
2 then 25 rows.
3 then 43 rows.

AL=23 set ROM 8x8 double dot font (graphics).

Input:

BL=# of rows on screen, as follows:
0 then DL=user specified # rows.
DL=# rows.
1 then 14 rows.
2 then 25 rows.
3 then 43 rows.

AL=24 set ROM 8x16 font (graphics).

Input:

BL=# of rows on screen, as follows:
0 then DL=user specified # rows.
DL=# rows.
1 then 14 rows.
2 then 25 rows.
3 then 43 rows.

AL=30 return font information.

Input:

BH=0 return pointer to upper 128 graphics characters (INT 01Fh pointer-color/graphics adapter compatible modes).
BH=1 return pointer to graphics font (INT 043h pointer).
BH=2 return pointer to ROM 8x14 font.
BH=3 return pointer to ROM 8x8 double dot font.
BH=4 return pointer to top half of ROM 8x8 double dot font.
BH=5 return pointer to ROM font supplement for 9x14 text.
BH=6 return pointer to ROM 8x16 font.
BH=7 return pointer to ROM font supplement for 9x16 text.

Output:

CX=char_height.
DL=crt_rows -1.
ES:BP=pointer to table selected by BH.

AH=12 Return EGA information or select alternate print screen handler.

BL=10 return information.

Input:

none.

Output:

BH =0 color mode, addressed at 03DX.
=1 monochrome mode, addressed at 03BX.
BL =installed video memory as follows:
0 =64K bytes installed.
1 =128K bytes installed.
2 =192K bytes installed.
3 =256K (or more) bytes installed.
CH =feature bits (bits 4-7 of info_1 shifted right).
CL =switches (bits 0-3 of info_1).

BL=20 select this BIOS's print screen routine, which supports all modes of this BIOS.

Input:

none.

Output:

none.

NOTE: This function selects the print screen routine built into this ROM to replace the standard BIOS print screen routine.

BL=30 select scan lines for text modes.

Input:

AL=scan lines to set (takes effect on next mode change).

0 = 200 scan lines.

1 = 350 scan lines.

2 = 400 scan lines.

Output:

AL=12h.

BL=31 set default palette load.

Input:

AL=# enable/disable palette loading.

0=enable palette loading.

1=disable palette loading.

Output:

AL=12h.

BL=32 Enable/disable video.

Input:

AL=# enable/disable video.

0=enable video.

1=disable video.

Output:

AL=12h.

BL=33 Enable/disable gray scale summing.

Input:

AL=# enable/disable gray scale summing.

0=enable summing.

1=disable summing.

Output:

AL=12h.

BL=34 Enable/disable cursor emulation.

Input:

AL=# enable/disable cursor emulation.

0=enable emulation.

1=disable emulation.

Output:

AL=12h.

BL=35 Select/deselect display.

Input:

Buffers for adapter and planar video are initialized then:
AL=# select/deselect adapter/planar video.
0=initial deselect adapter video.
1=initial select planar video.
2=deselect active display.
3=select inactive display.
ES:DX=pointer to 128-byte buffer.

Output:

AL=12h.

BL=36 Enable/disable video output.

Input:

AL=# enable/disable video output.
0=enable video output.
1=disable video output.

Output:

AL=12h.

Extension to INT 10 Bios function AH=012:

BL=0F1 Set/Get Frequency Type

This function sets or gets the currently-selected frequency type (i.e. 60 Hz, 72 Hz, etc) for modes of a particular resolution.

Input:

AL=0: set frequency type
AL=1: get frequency type
BH=code for screen resolution group:
0=640x480
1=800x600
2=1024x768
3=1280x1024
if AL=0, CX=code for frequency type:
640x480: 0=60 Hz, 1=72 Hz
800x600: 0=56 Hz, 1=60 Hz, 2=72 Hz
1024x768: 0=43.5 Hz (interlaced), 1=60 Hz, 2=70 Hz, 3=72 Hz
1280x1024: 0=43.5 Hz (interlaced), 1=60 Hz, 2=70 Hz

Output:

AL=012 (can be used to check whether this function is present)
CX=code for frequency type currently set (at the end of the call)

Notes:

1. Some frequency types are not available on all boards. (Check the returned frequency type to see if a specified frequency type got set or not.)
2. In some cases, a particular mode might get set at a lower frequency than indicated by the frequency type value returned from this function due to a bandwidth limitation (such as with a HiColor mode).
3. A few boards have alternate methods of determining the frequency to use.

AH=13 Write text string.

Input:

AL=0 text string is characters only. Cursor not moved from original position.
BL=attribute to write text string with.

1 text string is characters only. Cursor moved to end of text string.
BL=attribute to write text string with.

2 text string is alternating character/attribute sequence. Cursor not moved from original position.

3 text string is alternating character/attribute sequence. Cursor moved to end of text string.
BH=page to write text string to.
CX=count of characters (not bytes) in string to display.
DH=row position at which to start displaying string.
DL=column position at which to begin displaying string.
ES:BP=pointer to text string to be written.

NOTE: Scroll, backspace, carriage return, if any, will take place in the active page only.

AH=1A Read/write display code function.

Display combination codes:
00 - No display.
01 - Monochrome with 5151.
02 - CGA with 5153/4.
03 - Reserved.
04 - EGA with 5153/4.
05 - EGA with 5151.
06 - Professional Graphics System with 5175.
07 - VGA with analog BW.
08 - VGA with analog color.
09 - Reserved.
0A - System 30 with 5153/4.
0B - System 30 with analog BW.
0C - System 30 with color.
0D to FE - Reserved.
FF - Unknown.

AL=0 Read display code.

Input:

none.

Output:

AL=1Ah.
BL=Active display code.
BH=Alternate display code.

AL=1 Write display code.

Input:

BL=Active display code.
BH=Alternate display code.

Output:

AL=1Ah.

AH=1B Return functionality/state information.

Input:

BX=implementation type.
ES:DI=buffer (40h bytes).

Output:

AL=1Bh.
Buffer, in the following format:

offset	type	description
00	word	Offset to static functionality information.
02	word	Segment to static functionality information.
04	byte	Video mode.
05	word	Number of columns on screen.
07	word	Length of regen buffer.
09	word	Start address of regen buffer (offset).
0B	8*word	Cursor position for 8 pages (row, column).
1B	word	Cursor mode setting (start, end).
1D	byte	Active page.
1E	word	CRTC address.
20	byte	Current setting of 3x8 register (mode register).
21	byte	Current setting of 3x9 register.
22	byte	Rows on screen.
23	word	Character height.
25	byte	Active display combination code.
26	byte	Alternate display combination code.
27	word	Colors supported for current video mode.
29	byte	Display pages supported for current video mode.
2A	byte	Scan lines in current video mode. 0=200 1=350 2=400 3=480 4=Reserved 5=600 (Note: IBM reserves this) 6=768 (Note: IBM reserves this) 7-255=Reserved

2B	byte	Primary character block. 0=block 0 1=block 1 . . . 255=block 255
2C	byte	Secondary character block.
2D	byte	Miscellaneous state information. 0-1=all modes on all monitors active 1-1=summing active 2-1=monochrome active 3-1=mode set default palette loading disabled 4-1=cursor emulation active 5-0=background intensity / 1=blinking 6-7=Reserved
2E	byte	Reserved.
2F	byte	Reserved.
30	byte	Reserved.
31	byte	Video memory available. 0=64KB 1=128KB 2=192KB 3=256KB 4-255=Reserved
32	byte	Save pointer state information. 0=512 character set active 1=dynamic save area active 2=alpha font override active 3=graphics font override active 4=palette override active 5=DCC extension active 6-7=Reserved
33-3F	byte	Reserved.

Format of static functionality table:

bit flags: 0=not supported
1=supported

offset	type	description
00	byte	Bit Video modes. 0 mode 0 1 mode 1 2 mode 2 3 mode 3 4 mode 4 5 mode 5 6 mode 6 7 mode 7

01	byte	Bit Video modes. 0 mode 8 1 mode 9 2 mode A 3 mode B 4 mode C 5 mode D 6 mode E 7 mode F
02	byte	Bit Video modes. 0 mode 10 1 mode 11 2 mode 12 3 mode 13 4-7 Reserved.
03-06	byte	Reserved.
07	byte	Bit Scan lines available in text mode. 0 200 scan lines 1 350 scan lines 2 400 scan lines 3-7 Reserved
08	byte	Character blocks available in text mode.
09	byte	Maximum number of active character blocks in text modes.
0A	byte	Bit Miscellaneous functions. 0 all modes on all monitors 1 summing 2 character font loading 3 mode set default palette loading 4 cursor emulation 5 EGA palette 6 color palette 7 color paging
0B	byte	Bit Miscellaneous functions. 0 light pen 1 save/restore 2 background intensity / blinking control 3 DCC 4-7 Reserved
0C	byte	Reserved.
0D	byte	Reserved.
0E	byte	Bit Save pointer functions. 0 512 character set 1 dynamic save area 2 alpha font override 3 graphics font override 4 palette override 5 DCC extension 6-7 Reserved.
0F	byte	Reserved.

AH=1C Save/restore video state.
AL=0 Return save/restore state buffer size.

Input:

CX=requested states.

Output:

AL=1Ch.
BX=# of 64 byte blocks needed for save buffer.

AL=1 Save state

Input:

CX=requested states.
ES:BX=pointer to save area.

Output:

(ES:BX) area modified.

AL=1Ch

AL=2 Restore state.

Input:

CX=requested states.
ES:BX=pointer to save area.

Output:

AL=1Ch.

Requested states in CX - defined as follows:
bit 0=1 - save/restore video hardware state.
bit 1=1 - save/restore video BIOS data area.
bit 2=1 - save/restore video external palette.
bits 3-F=Reserved.

8.2 ACL Programming Considerations

Mathematical Algorithm that Accelerator Implements:
Stepping of Destination Address:

```
da = Starting Dst Address (from MMU Translation)
for (ypos = 0; ypos <= ycnt; ypos++) {
        for (xpos = 0; xpos < xcnt; xpos++) {
                if (xdir)
                        process(da - xpos);
                else
                        process(da + xpos);
        }
        if (ydir)
                da = da - dyof;
        else
                da = da + dyof;
}
```

Appendix A. ET4000/W32p (Microsoft) Raster Operations Codes and Definitions

The ET4000/W32p supports all Microsoft 256 Raster Operation Codes. These codes define the ways in which BitBLT combines the bits in a source bitmap with the bits in a brush or pattern bitmap, and the bits in the destination bitmap.

Operands used in operations are:

S	Source bitmap
P	Pattern bitmap
D	Destination bitmap

Boolean operators used in operations are:

o	Bitwise OR
x	Bitwise Exclusive OR
a	Bitwise AND
n	Bitwise NOT (invert)

Operations presented in this description are in reverse Polish notation. Example:

DPsoo = logical OR on source and pattern, then
another logical OR with destination.
Result is stored in destination.

This operation can also be stated in this manner: DPoSo. Either form accomplishes the same objective though the latter may be somewhat more clear. Generally, functions are expressed so that they are easily read outward from the place at which they change from upper to lower case.

Example: PSDPSanaxx can be read as:

PSD PSa naxx: 'and' source with pattern.
PSDPSa n axx: complement result.
PS D PSan a xx: 'and' with destination.
P S PDPSana x x: 'xor' with source.
P SDPSanax x: 'xor' with pattern.

Another more complex example is SSPxDSxaxn, which, expanded is:

S SP DSxaxn: 'xor' source and pattern.
SSPx DSx axn: 'xor' destination and source.
S [SPx][DSx]a xn: 'and' the bracketed items.
S SPxDSxa x n: 'xor' result of last step with source.
SSPxDSxax n: complement result, and put into destination.

Raster Operation Codes (ROP)

Each raster operation code is an 8-bit value that represents the result of the Boolean operation on pre-defined pattern (P), source (S), and destination (D) values. For example, the operation indices for the PSo, PSon, and DPSoo operations are:

	P	S	D	PSo	PSon	DPSoo	Arbitrary Function
	0	0	0	0	1	0	1
	0	0	1	0	1	1	0
	0	1	0	1	0	1	0
	0	1	1	1	0	1	1
	1	0	0	1	0	1	1
	1	0	1	1	0	1	0
	1	1	0	1	0	1	1
	1	1	1	1	0	1	0
Hex Opcode:				FC	03	FE	59

Boolean functions can be represented by the string of 1's and 0's on the right side of such a table. In this example, PSon is the string 00000011 (read from bottom to top), which is 03h. Note the PSon function in line 4 of the table.

In general, arbitrary functions, like the one on the far right in the table, have a unique hexadecimal number associated with them (in this instance, 0x59). By referring to the table, you can find the appropriate ROP and a function that evaluates it (DPSnox).

The value of each raster operation code determines the location of the raster operation in the table: the PSo operation is in line 252 (FCh) of the table; DPSoo is in line 254 (FEh), and so on.

Operation Code List

The list that follows is of the Boolean functions in hexadecimal (the ROP value) and reverse Polish notation, along with common names for same.

Boolean function in HEX	Boolean function in R Polish	Common name
00	0	Blackness
01	DPSoon	-
02	DPSona	-
03	PSon	-
04	SDPona	-
05	DPon	-
06	PDSxnon	-
07	PDSaon	-
08	SDPnaa	-
09	PDSxon	-
0A	DPna	-
0B	PSDnaon	-
0C	SPna	-
0D	PDSnaon	-
0E	PDSonon	-
0F	Pn	-
10	PDSona	-
11	DSon	NOTSRCERASE
12	SDPxnon	-
13	SDPaon	-
14	DPSxnon	-
15	DPSaon	-
16	PSDPSanaxx	-
17	SSPxDSxaxn	-
18	SPxPDxa	-
19	SDPSanaxn	-
1A	PDSPaox	-
1B	SDPSxaxn	-
1C	PSDPaox	-
1D	DSPDxaxn	-
1E	PDSox	-
1F	PDSoan	-
20	DPSnaa	-
21	SDPxon	-
22	DSna	-
23	SPDnaon	-
24	SPxDSxa	-
25	PDSPanaxn	-
26	SDPSaoxxn	-
27	SDPSxnox	-
28	DPSxa	-
29	PSDPSaoxxn	-
2A	DPSana	-
2B	SSPxPDxaxn	-
2C	SPDSoax	-
2D	PSDnox	-
2E	PSDPxox	-
2F	PSDnoan	-
30	PSna	-
31	SDPnaon	-
32	SDPS00x	-
33	Sn	NOTSRCOPY
34	SPDSaox	-

Boolean function in HEX	Boolean function in R Polish	Common name
35	SPDSxnox	-
36	SDPox	-
37	SDPoan	-
38	PSDPoax	-
39	SPDnox	-
3A	SPDSxox	-
3B	SPDnoan	-
3C	PSx	-
3D	SPDSonox	-
3E	SPDSnaox	-
3F	PSan	-
40	PSDnaa	-
41	DPSxon	-
42	SDxPDxa	-
43	SPDSanaxn	-
44	SDna	SRCERASE
45	DPSnaon	-
46	DSPDaox	-
47	PSDPxaxn	-
48	SDPxa	-
49	PDSPDoaxxn	-
4A	DPSDoax	-
4B	PDSnox	-
4C	SDPana	-
4D	SSPxDSxoxn	-
4E	PDSPxox	-
4F	PDSnoan	-
50	PDna	-
51	DSPnaon	-
52	DPSDaox	-
53	SPDSxaxn	-
54	DPSonon	-
55	Dn	DSTINVERT
56	DPSox	-
57	DPSoan	-
58	PDSPoax	-
59	DPSnox	-
5A	DPx	PATINVERT
5B	DPSDonox	-
5C	DPSDxox	-
5D	DPSnoan	-
5E	DPSDnaox	-
5F	DPan	-
60	PDSxa	-
61	DSPDSaoxxn	-
62	DSPDoax	-
63	SDPnox	-
64	SDPSoax	-
65	DSPnox	-
66	DSx	SRCINVERT
67	SDPSonox	-
68	DSPDSonoxxn	-
69	PDSxxn	-
6A	DPSax	-
6B	PSDPSoaxxn	-
6C	SDPax	-
6D	PDSPDoaxx	-
6E	SDPSnoax	-
6F	PDSanan	-
70	PDSana	-

Boolean function in HEX	Boolean function in R Polish	Common name
71	SSDxPDxaxn	-
72	SDPSxox	-
73	SDPnoan	-
74	DSPDxox	-
75	DSPnoan	-
76	SDPSnaox	-
77	DSan	-
78	PDSax	-
79	DSPDSoaxxn	-
7A	DPSDnoax	-
7B	SDPxnan	-
7C	SPDSnoax	-
7D	DPSxnan	-
7E	SPxDSxo	-
7F	DPSaan	-
80	DPSaa	-
81	SPxDSxon	-
82	DPSxna	-
83	SPDSnoaxn	-
84	SDPxna	-
85	PDSPnoaxn	-
86	DSPDSoaxx	-
87	PDSaxn	-
88	DSa	SRCAND
89	SDPSnaoxn	-
8A	DSPnoa	-
8B	DSPSxoxn	-
8C	SDPnoa	-
8D	SDPSxoxn	-
8E	SSDxPDxax	-
8F	PDSanan	-
90	PDSxna	-
91	SDPSnoaxn	-
92	DPSDPoaxx	-
93	SPDaxn	-
94	PSDPSoaxx	-
95	DPSaxn	-
96	DPSxx	-
97	PSDPSonoxx	-
98	SDPSonoxn	-
99	DSxn	-
9A	DPSnax	-
9B	SDPSoaxn	-
9C	SPDnax	-
9D	DSPDoaxn	-
9E	DSPDSaoxx	-
9F	PDSxan	-
A0	DPa	-
A1	PDSPnaoxn	-
A2	DPSnoa	-
A3	DPSDxoxn	-
A4	PDSPonoxn	-
A5	PDxn	-
A6	DSPnax	-
A7	PDSPoaxn	-
A8	DPSoa	-
A9	DPSoxn	-
AA	D	-
AB	DPSono	-
AC	SPDSxax	-

Boolean function in HEX	Boolean function in R Polish	Common name
AD	DPSDaoxn	-
AE	DSPnao	-
AF	DPno	-
B0	PDSnoa	-
B1	PDSPxoxn	-
B2	SSPxDSxox	-
B3	SDPanan	-
B4	PSDnax	-
B5	DPSDoaxn	-
B6	DPSDPaoxx	-
B7	SDPxan	-
B8	PSDPxax	-
B9	DSPDaoxn	-
BA	DPSnao	-
BB	DSno	MERGEPAINT
BC	SPDSanax	-
BD	SDxPDxan	-
BE	DPSxo	-
BF	DPSano	-
C0	PSa	MERGECOPY
C1	SPDSnaoxn	-
C2	SPDSonoxn	-
C3	PSxn	-
C4	SPDnoa	-
C5	SPDSxoxn	-
C6	SDPnax	-
C7	PSDPoaxn	-
C8	SDPoa	-
C9	SPDoxn	-
CA	DPSDxax	-
CB	SPDSaoxn	-
CC	S	SRCCOPY
CD	SDPono	-
CE	SDPnao	-
CF	SPno	-
D0	PSDnoa	-
D1	PSDPxoxn	-
D2	PDSnax	-
D3	SPDSoaxn	-
D4	SSPxPDxax	-
D5	DPSanan	-
D6	PSDPSaoxx	-
D7	DPSxan	-
D8	PDSPxax	-
D9	SDPSaoxn	-
DA	DPSDanax	-
DB	SPxDSxan	-
DC	SPDnao	-
DD	SDno	-
DE	SDPxo	-
DF	SDPano	-
E0	PDSoa	-
E1	PDSoxn	-
E2	DSPDxax	-
E3	PSDPaoxn	-
E4	SDPSxax	-
E5	PDSPaoxn	-
E6	SDPSanax	-
E7	SPxPDxan	-
E8	SSPxDSxax	-

Boolean function in HEX	Boolean function in R Polish	Common name
E9	DSPDSanaxxn	-
EA	DPSao	-
EB	DPSxno	-
EC	SDPao	-
ED	SDPxno	-
EE	DSo	SRCPAINT
EF	SDPnoo	-
F0	P	PATCOPY
F1	PDSono	-
F2	PDSnao	-
F3	PSno	-
F4	PSDnao	-
F5	PDno	-
F6	PDSxo	-
F7	PDSano	-
F8	PDSao	-
F9	PDSxno	-
FA	DPo	-
FB	DPSnoo	PATPAINT
FC	PSo	-
FD	PSDnoo	-
FE	DPSoo	-
FF	1	WHITENESS

Appendix B. Schematics

Sample schematics are available from your Tseng Labs Account Representative. Please contact your Account Representative for current schematic samples.

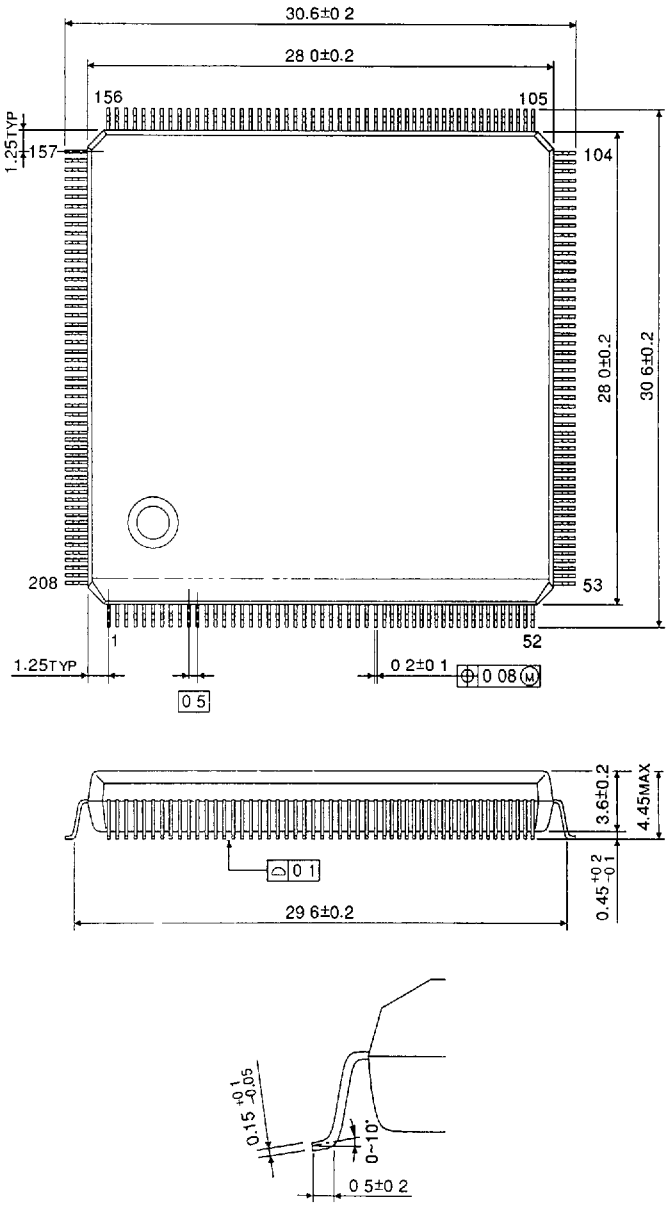
Appendix C. 2-Byte Character Code

When using the 2-byte Character Code (CC) feature, observe the following:

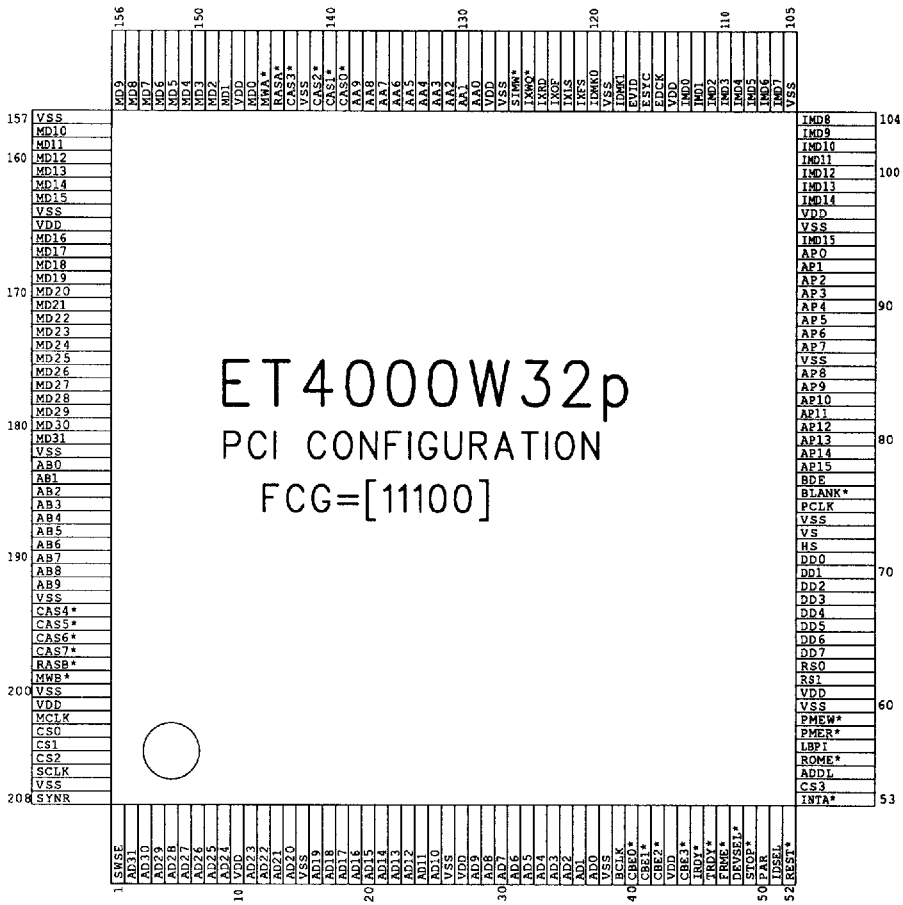
- For the first font fetch (i.e.: RASBL and CASL<3:2> DRAM access cycle) of each scan line, the MD<31:16> font data are ignored.
- The first or subsequent Character Codes (CC0 or CC1, CC2, etc.) should be latched by the external circuit and this Character Code should remain latched until the next font fetch cycle.
- The second, or odd, Character Code (CC1 or CC3, CC5, etc.) and the previously latched even Character Code (CC0 or CC2, CC4, etc.) total of 16-bit Character Codes should be latched at the beginning of every even font fetch cycle via transparent latches.
- If the current CCn is not an English character, then the font data (MD<31:16>) will be fetched according to the 16-bit Character Code (i.e., the external EPROMs are enabled), else the DRAMs font (which contains the VGA's font) are enabled.
- If the last character font of the scan line is a 24-bit wide font and begins at an odd column, then only half of the font (12-dot) will be displayed.

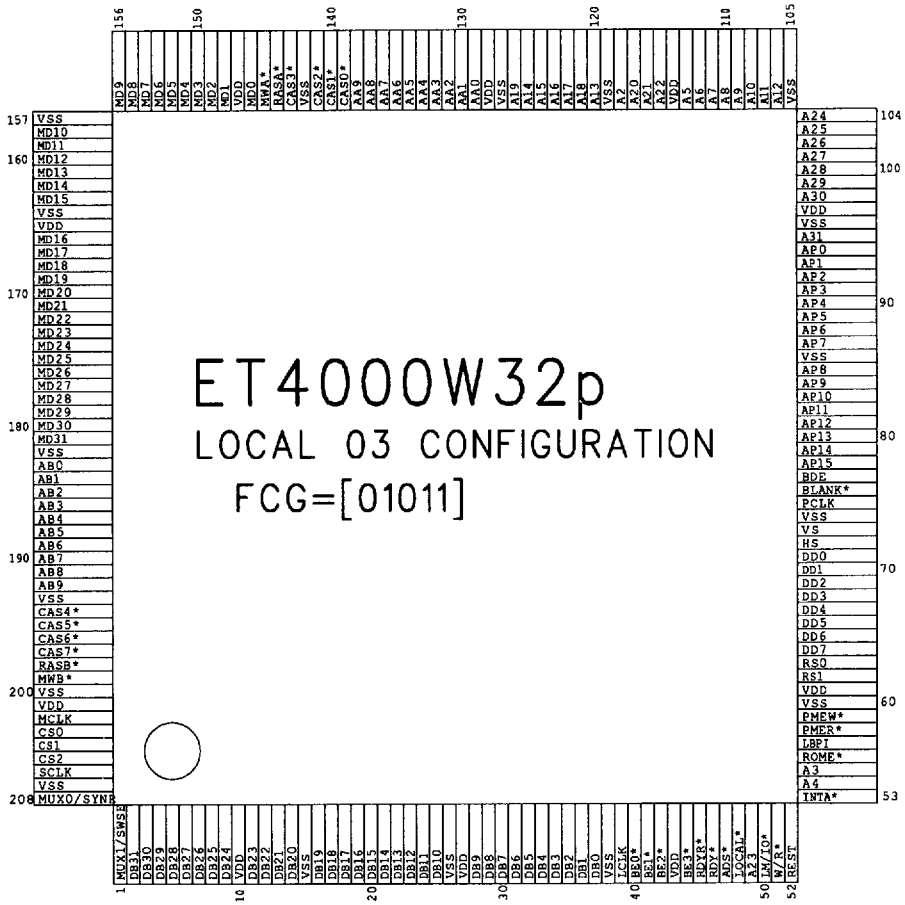
Appendix D. Mechanical Specifications

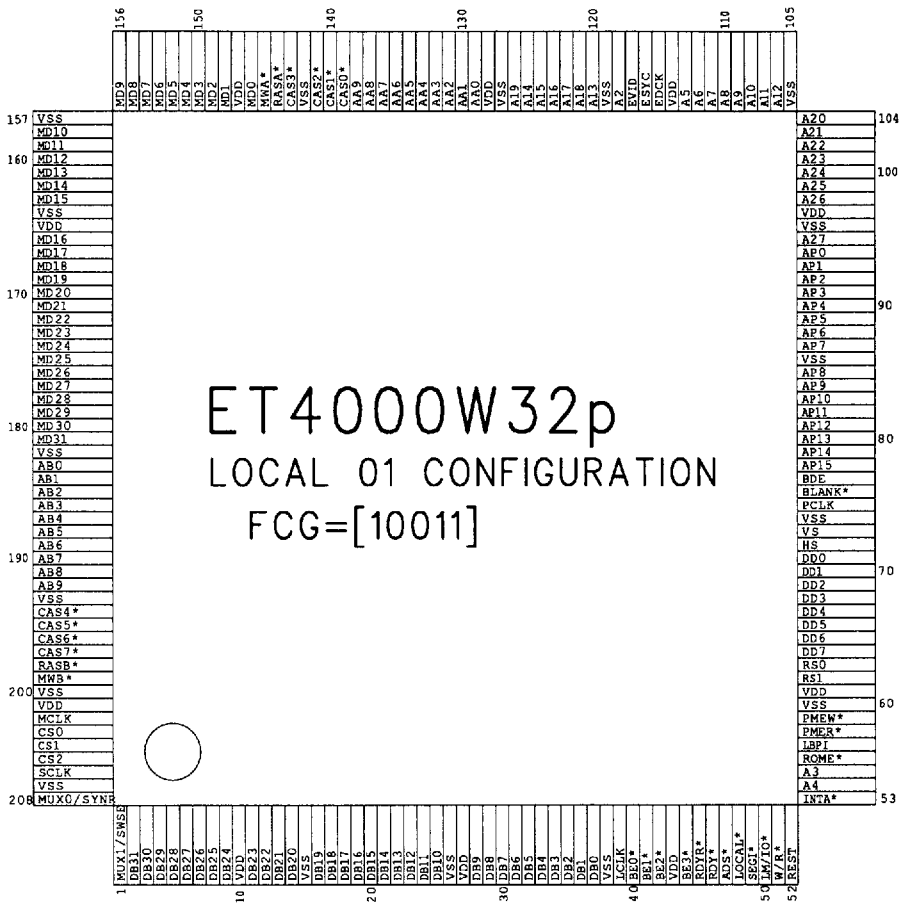
ET4000/W32p 208-pin Quad Flat Package

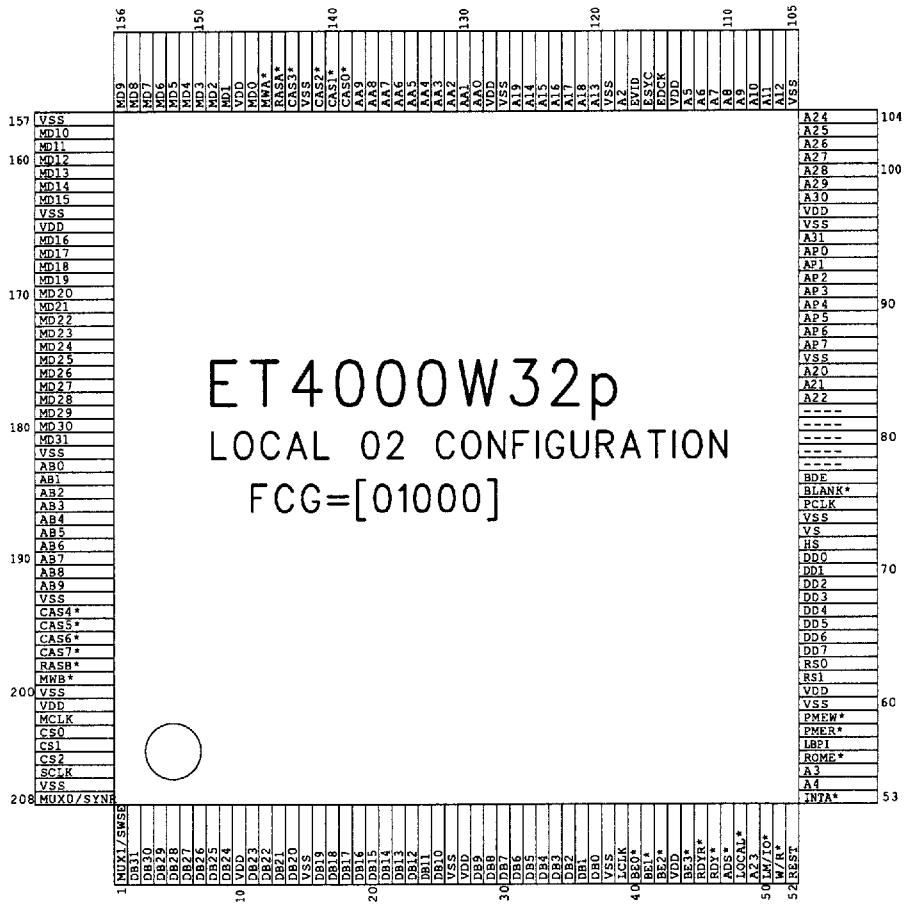


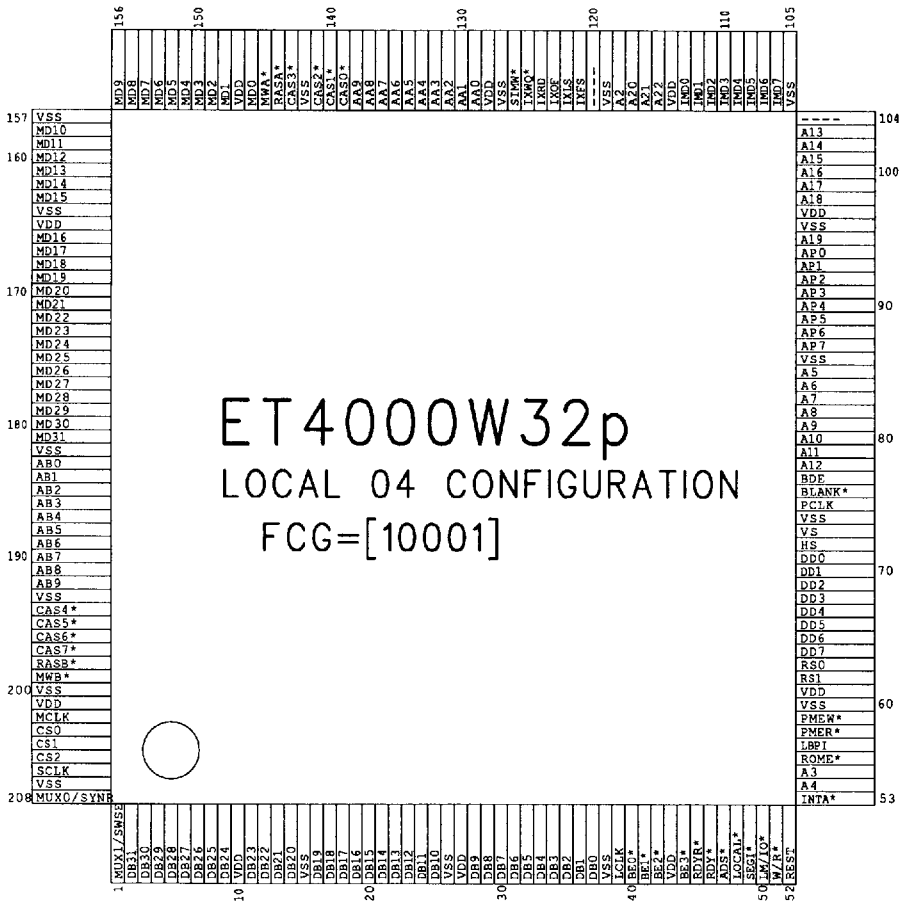
Appendix E. Specific Configuration Pin-outs

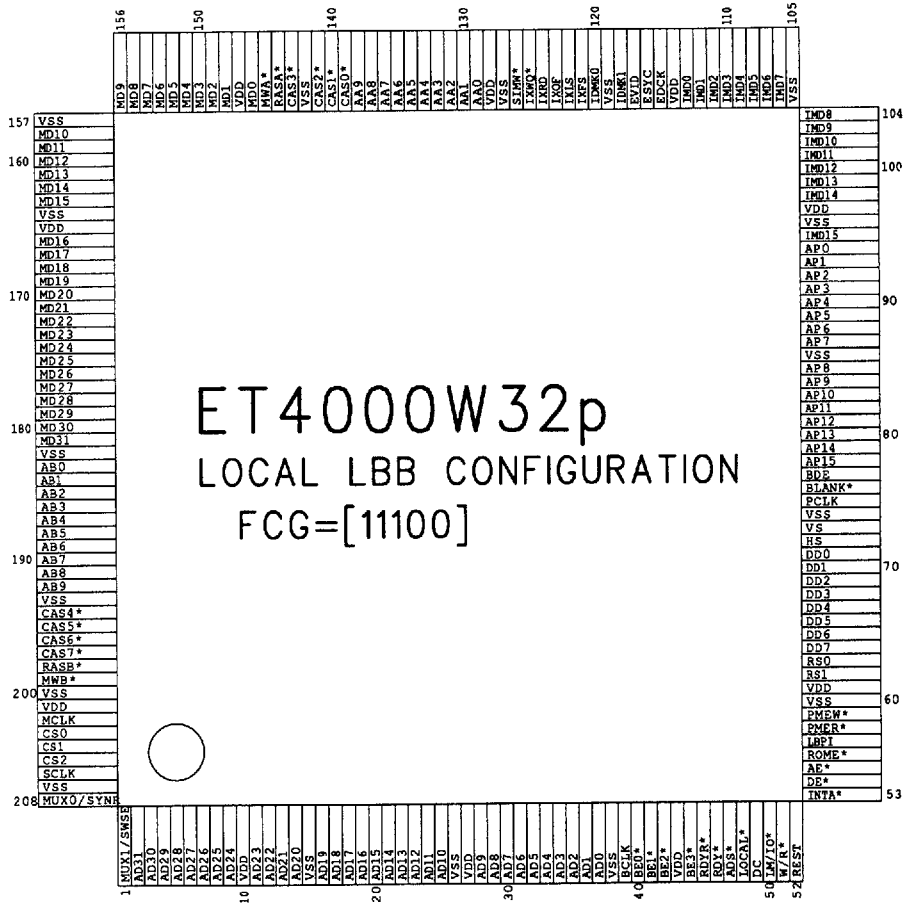












Index

A

B

C

D

E

F

G

H

I

K

L

M

O

P

Q

R

S

T

U

V

W

X

Y

Z

Symbols

Appendix D. Mechanical Specifications

ET4000/W32p 208-pin Quad Flat Package