W32i

ET4000/W32i Graphics Accelerator

Data Book

GRAPHICS BY

TSENG

LABS

I

TSENG LABS ET4000/W32i DATA BOOK ERRATA

Release Date: 10/1/93

10/1/93

Section 2.11.6.3 has been changed to the following:

2.11.6.3 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.

II

The following sections, 2.11.6.5 and 2.11.6.6, have been added:

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.

Contents

Figures

Tables

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.

1. INTRODUCTION

The Tseng Labs ET4000/W32i video controller is an ISA/EISA/MCA- 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/W32i 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/W32i provides 32 multiplexed address/data lines, and can be connected to the processor via the local bus. ISA/EISA/MCA designs can use a 16-bit design, and allow the additional pins to be reassigned to other features during power-up. 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 W32i is designed to optimize GUI application functionality and performance, speeding the operation of applications like Microsoft Windows significantly. Additionally, the W32i 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/W32i, 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/W32i. 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/W32i is packaged in a 160-pin PFP (Plastic Flat Package) configuration.

1.1 ET4000/W32i Specifications

The following is an outline of the ET4000/W32i 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: ISA, EISA, MCA, 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. 8-bit pixel output AP<7:0>
  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/W32i 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 1024 x 768 in 65,536 colors interlaced or non-interlaced in graphics mode
      2. Pixel Clock Rate: graphics mode, 86MHz; text mode, 56MHz

V. Compatibility
   A. Register level: CGA/MDA/HERC/EGA/VGA
   B. Display level: 8514A
   C. Monitor: IBM's 8503, 8512, 8513, 8514, 5154, 5153, 5151, NEC's MultiSync, MultiSync Plus, XL, and other monitors with up to 1024 x 1024-pixel resolution

The ET4000/W32i offers performance starting with all of the IBM VGA/EGA features and provides enhanced performance with features summarized below.

1.2 ET4000/W32i Overview

- IBM Video Graphics Array (VGA) and Enhanced Graphics Adapter (EGA) register-level compatibility.
- Pin-to-pin compatible with ET4000/W32.
- Supports interface to 386/486 SX/DX local bus, ISA/EISA/MCA bus, with data bus width up to 32 bits.
- Local Bus has programmable number of wait states down to 0-wait state, with no external PAL chips required.
- 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.
- 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.

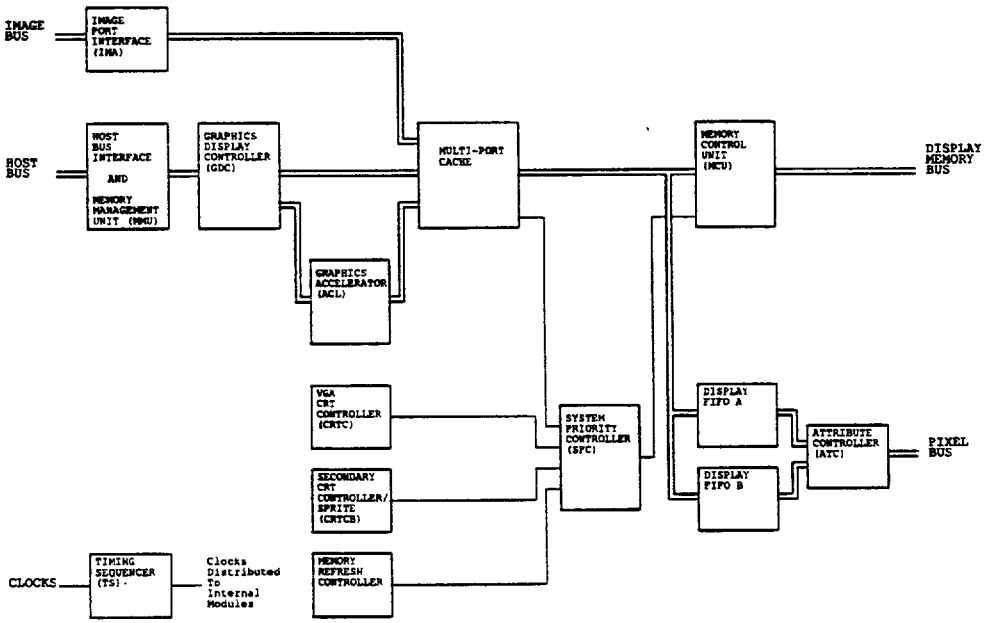
1.3 Notational Conventions

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
MCA	Micro Channel Architecture
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/W32i FUNCTIONAL DESCRIPTION

All major elements of the ET4000/W32i are contained within a single 160-pin Plastic Flat Package. The block diagram below shows the internal architecture of the ET4000/W32i. The following sections provide a breakdown of the major elements of the chip.



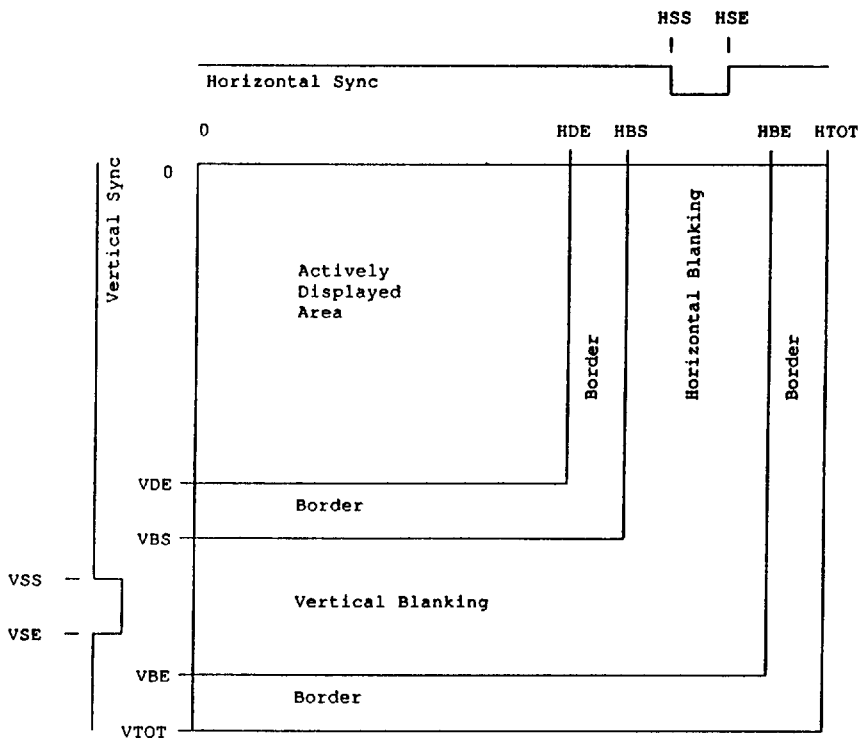
ET4000/W32i Block Diagram

2.1 VGA CRT Controller (CRTC)

The ET4000/W32i 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/W32i'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.

The diagram below displays the role that the CRTC registers play to effect the horizontal and vertical timings of the CRT.

Figure 2.1-1: CRTC Video Timing Control 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

2.2 Secondary CRT Controller (CRTCB)/Sprite

This module of the ET4000/W32i 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/W32i 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.

2.2.1 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.

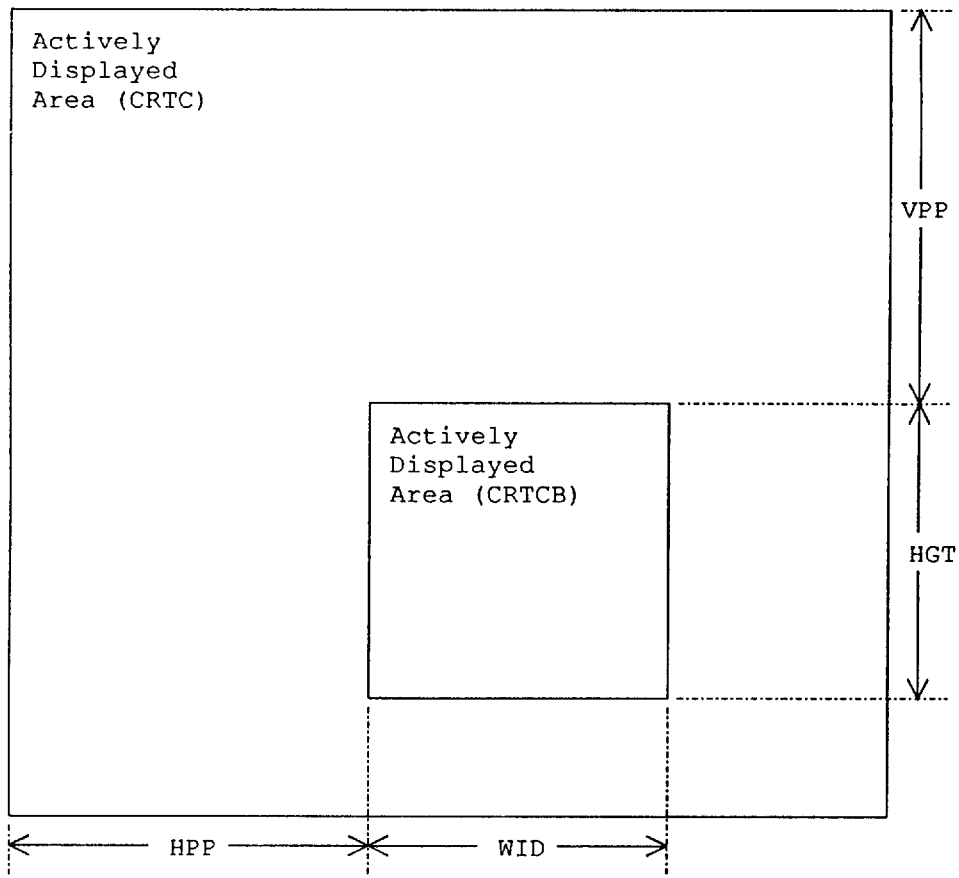
2. The color attributes displayed are not subject to internal "ATC" processing; i.e., the CRTCB display is packed pixel (linear graphics) format only.

3. The CRTCB display must be overlayed 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.

2.2.2 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 2.2.2-1: CRTCB Window Positioning



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

2.2.3 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

2.2.4 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:

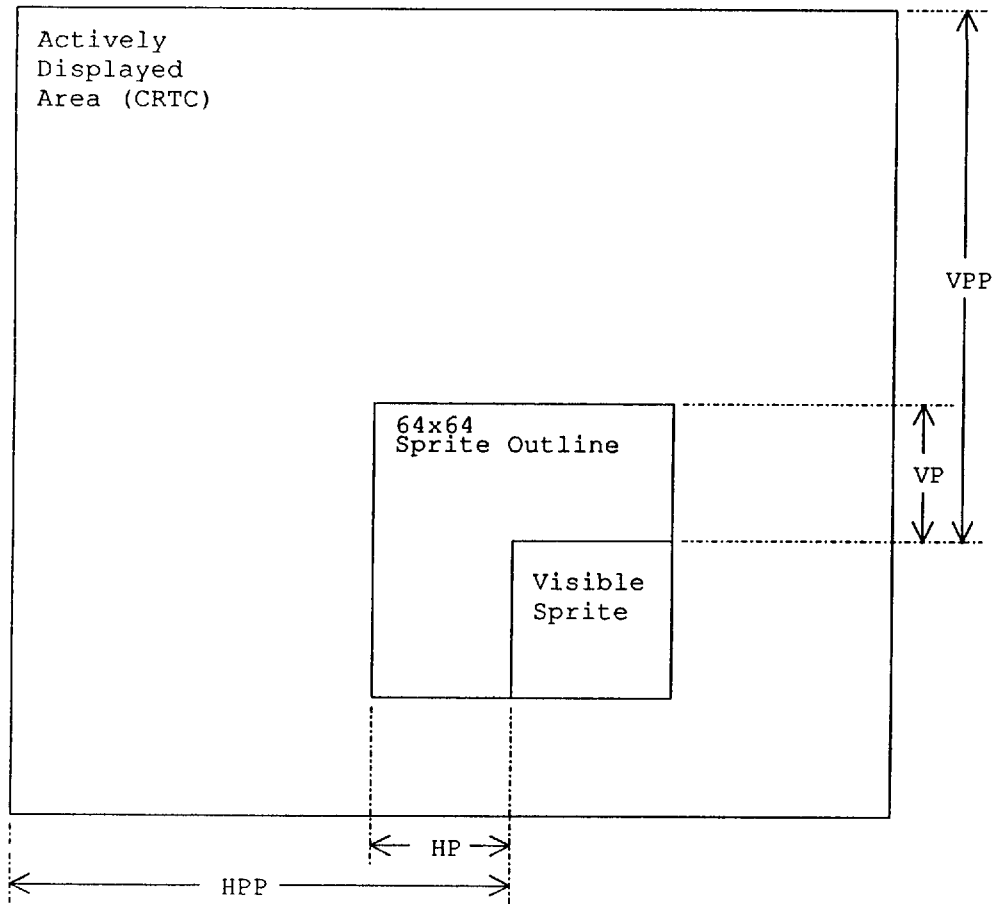
Bits<1:0>	Sprite Effect
0 0	Sprite Color 0 (defined as 00)
0 1	Sprite Color 1 (defined as FF)
1 0	Transparent (allow CRTC pixel pass through)
1 1	Invert (allow CRTC pixel invert through)

2.2.5 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 2.2.5-1: Sprite Positioning



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

2.2.6 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.

2.3 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.

2.3.1 Memory Interleave

Memory interleaving is done to increase DRAM bandwidth without doubling the DRAM data bus pin count. The ET4000/W32i 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 W32i 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 W32i 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 W32i, however, can dynamically allocate its resource to the CPU, the graphics accelerator, the CRTC, or data being input through the IMA port.

2.4 System Priority Controller (SPC)

The SPC's main task is to maximize performance by orchestrating the ET4000/W32i'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, 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 6.3.

2.5 Multiport Cache

The internal Multiport Cache™, an exclusive feature of the ET4000/W32i, allows access of multiple masters to the display memory cache simultaneously. Multiport Cache provides the ET4000/W32i 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/W32i will allow all three masters to read/write into the display memory, while operating concurrently.

2.6 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)

2.7 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/W32i 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).

2.8 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.

2.9 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/W32i 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 W32i 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/W32i 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/W32i 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/W32i's frame buffer. The masked data transferred can be used to reduce the bandwidth requirement.

2.9.1 Image Port Interface Protocol

1. External Image Processor produces IXFS and IXLS pulses signaling initialization of linear address.

2. ET4000/W32i 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 + 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/W32i 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.

2.10 Memory Management Unit (MMU)

The ET4000/W32i 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 byte boundary in the 4MB display memory space.

2.10.1 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 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 2.10.1-1 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.

When the APT bit of an aperture is set to "0," the MBP for that aperture must be doubleword-aligned. This alignment restriction arises from the fact that the internal GDC cannot perform a multiple-cycle operation to memory, which would be required if an access crossed a doubleword boundary. The accelerator does not have any such restriction; so when the APT bit is set to "1," the MBP can point to any byte boundary.

Important Note: A read through an aperture with the APT bit set to "1" will return an undefined result. The chip does not allow data to be read from the accelerator.

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. If the Aperture Type bit is "1" and the host is supplying Mix Map data to the Graphics Accelerator, then the 13-bit host address is multiplied by 8 before being added to the Memory Base Pointer. This is done to compensate for the 8-to-1 ratio of "bytes processed" to "bits in the Mix Map." Figure 2.10.1-2 depicts the address translation process.

Figure 2.10.1-1 3 MMU Aperture Mapping

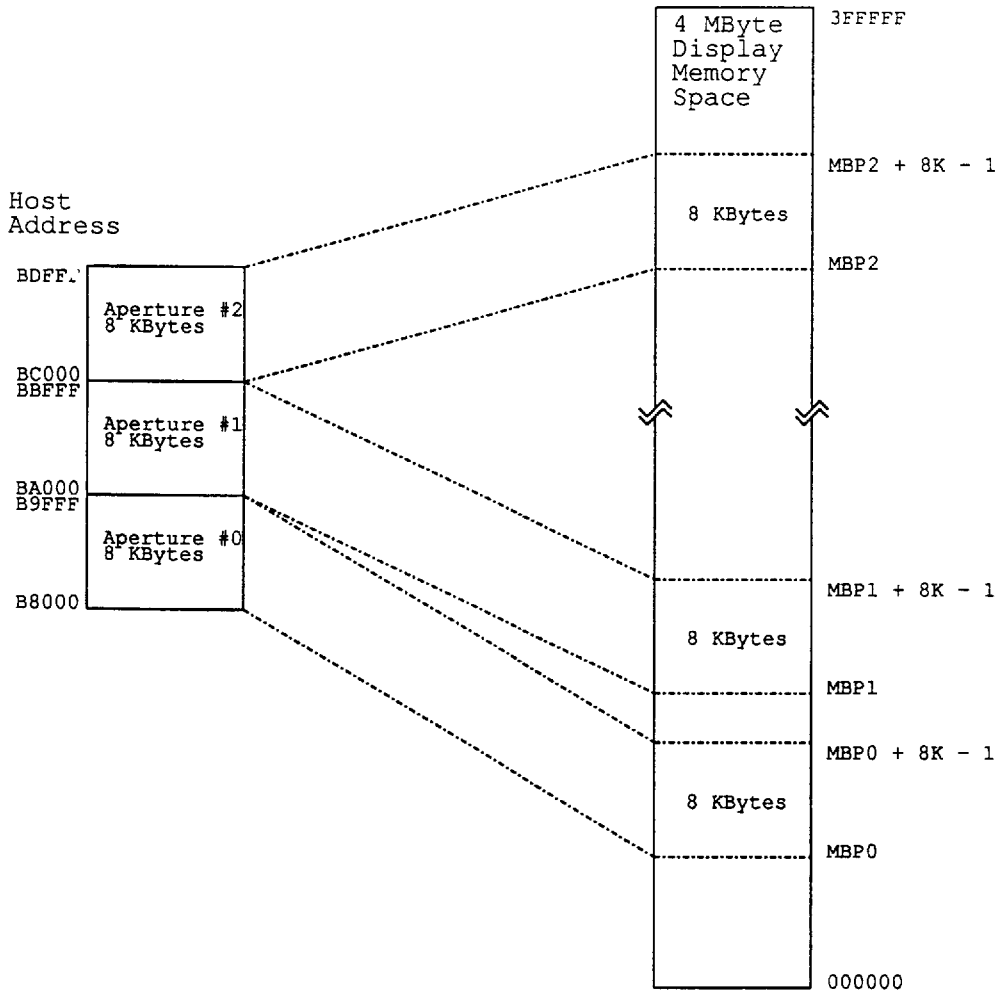
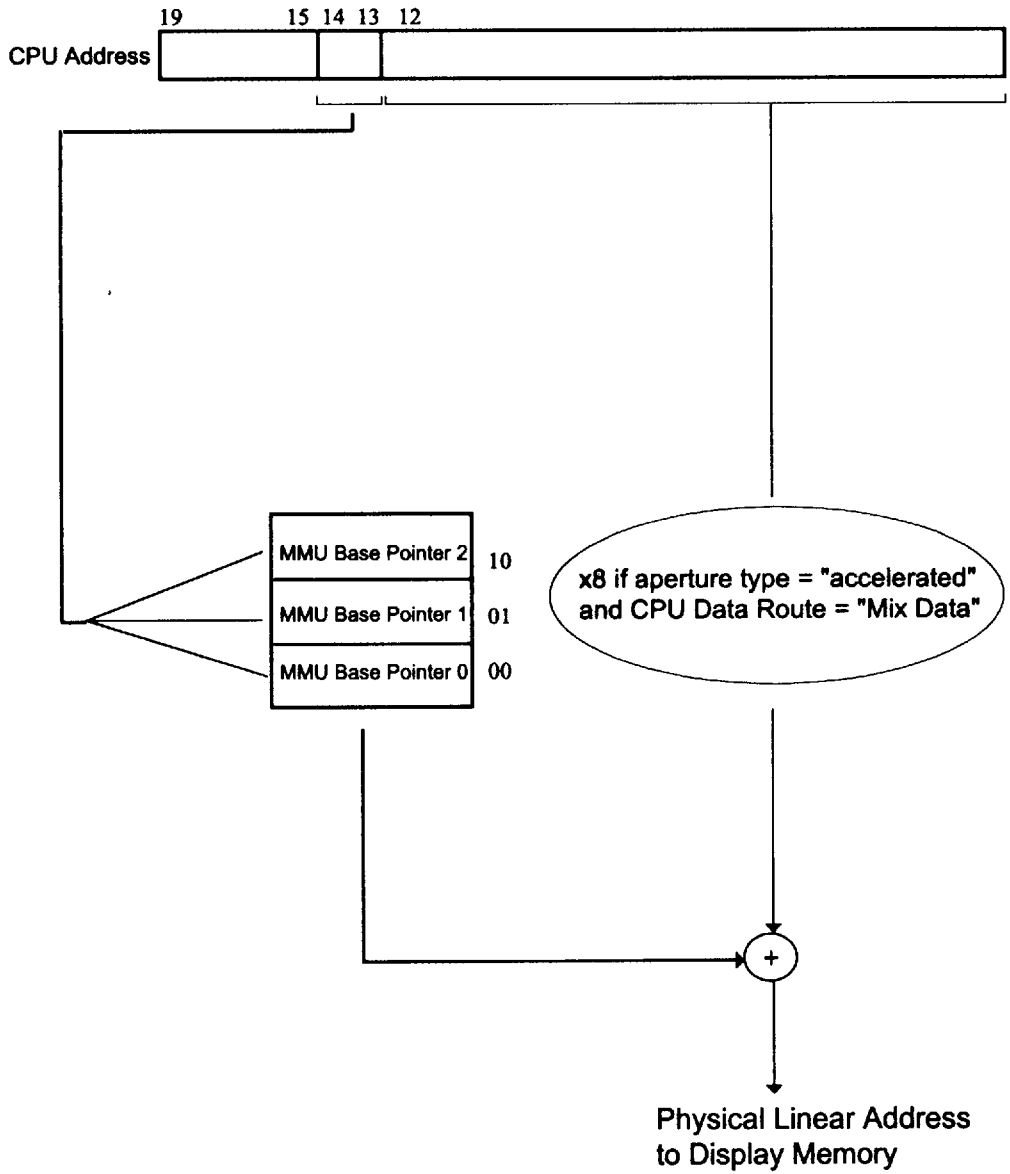


Figure 2.10.1-2: MMU Address Translation



2.10.2 Hardware Considerations

As mentioned earlier, the size of the aperture depends upon the system configuration. For example, if the image port is being used in a system linear-mapped configuration, only 20 bits of the host address can be wired to the chip (since the Image Port uses some pins normally used by the host address bus). In this configuration, bits 19 and 18 are used to select the aperture, while bits <17:0> are added to the Memory Base Pointer. The Video Memory Map in section 7.3 summarizes the effects of system configuration on the aperture size.

2.11 Graphics Accelerator (ACL)

The ET4000/W32i 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 or Raster Operations. The ET4000/W32i allows the CPU to distribute these tasks to its Graphics Accelerator. The Accelerator is mapped to unused areas of the display memory address space, and reconfigures itself automatically when multiple adapters are present, or if the VGA switches from graphics into text mode.

The ET4000/W32i 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") 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 Source Map, a Pattern Map, and a Destination Map, and writes it 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 ET4000/W32i and the host processor.
- The CPU need not perform any combinatorial functions on the graphics data; all Raster Operations are performed by the Graphics Accelerator.
- For most rectangular BitBLT's, the CPU need not maintain address pointers.
- Accesses to the display memory are localized within the ET4000/W32i, allowing the best possible utilization of display memory bandwidth.
- The CPU always has the option to maintain more control over a graphics operation, but still use the accelerator to assist in certain aspects of the operation. For example, the programmer may wish to have the CPU provide control of addressing the memory, but allow the accelerator to take care of applying the Raster Operation to the data.

2.11.1 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.

The accelerator operates on four "Pixel Maps":

1. Source Map
2. Pattern Map
3. Destination Map
4. Mix Map

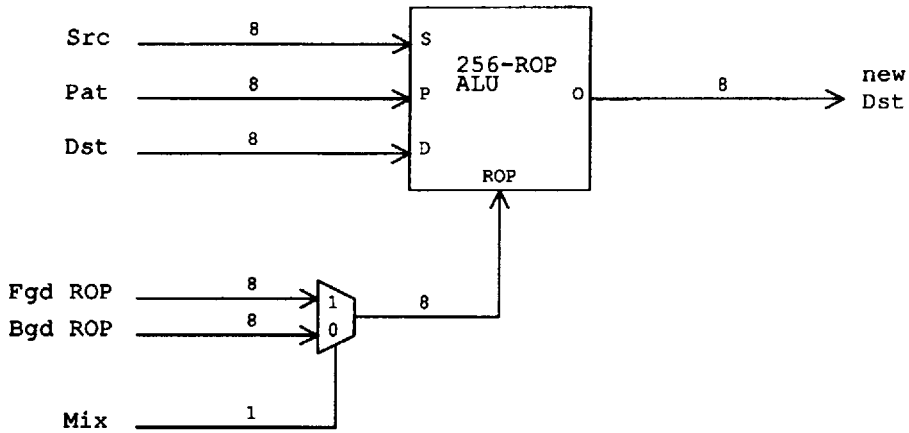
Data from these maps are combined according to the following rule:

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 ROP's is 100% compatible with the Microsoft Windows specification.

The Destination and Pattern maps must reside in the display memory. The Source map data may reside in display memory or be supplied by the CPU during an accelerated graphics operation. The Mix Map data may be supplied by the CPU, or it may be fixed to "1" (always use 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 following figure shows the path of one byte through the accelerator's Graphics Data Processor; in reality more than one byte is processed at a time.



ET4000/W32i Graphics Accelerator Data Path

2.11.2 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 in one of two ways:

1. The host performs a write to the display memory, using the MMU Buffer Space. If the APT bit for the MMU aperture is "1," then an accelerator operation will begin. The address which is generated by the MMU translation is implicitly loaded into the Destination Address Register in the Accelerator Queue, and the accelerator operation is begun. Thus, the act of writing to the display memory using an MMU aperture which is in "accelerated mode" implicitly specifies that area of memory to be the Destination Map. ...OR...

2. The starting address of the Destination Map can be explicitly loaded into the Destination Address Register, and then a write to the Accelerator Operation State Register can be performed with both the "Resume" and "Restore" control bits set to "1."

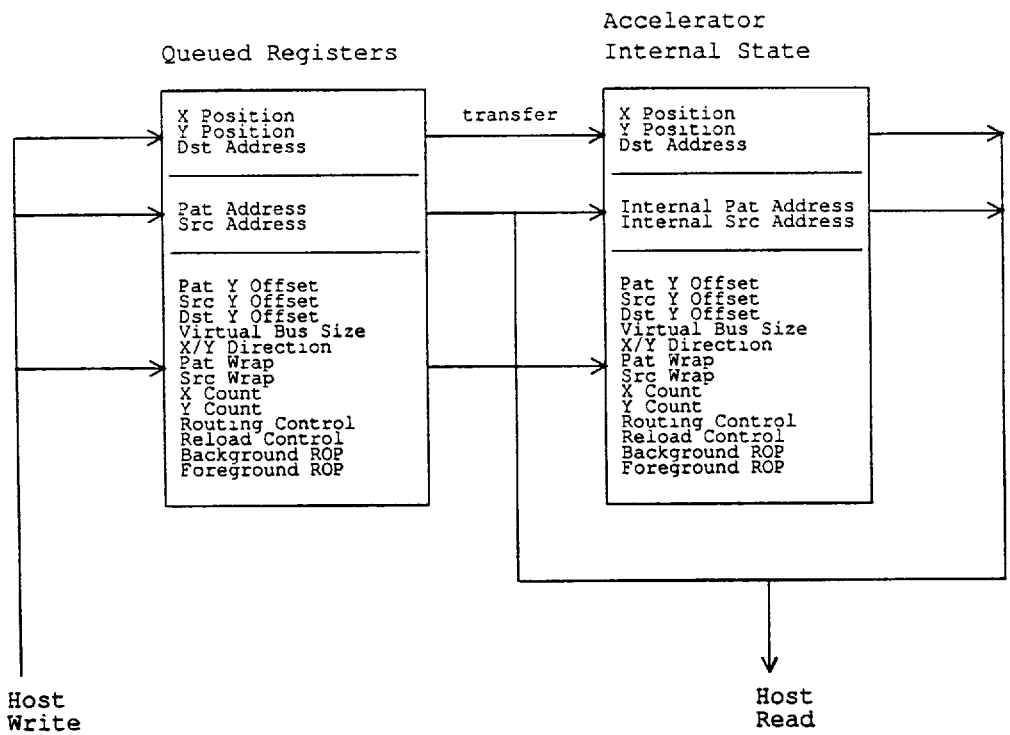
An accelerated graphics operation is defined as a two-dimensional "walk" through display memory. The X Count Register and Y Count Register specify the limits of each dimension.

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.

The accelerator always operates on pixel maps as if they are in linear format in display memory. The accelerator uses the MMU **only** to translate the starting Destination Address. The accelerator **does not** use the MMU to perform any address translations while it is performing a graphics operation. In other words, the MMU only operates on addresses from the host, and never on internal addresses generated by the accelerator. This means that the programmer has complete freedom to alter any MMU-related registers while the accelerator is in the middle of a graphics operation.

2.11.3 The Accelerator's Queue

The ET4000/W32i 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 one of the two methods 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.



ET4000/W32i 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. There is a slight difference in the storage of the Source and Pattern Address Registers; the Restore operation shifts the data from the queue into the Initial Source, Pattern Registers, while the data in the Initial Source, Pattern Registers is moved into the internal Source, Pattern Address Registers of the accelerator. In other words, the Source and Pattern Address Registers are configured as a three-stage shift register, with the Restore operation triggering a shift by one. Conversely, all of the other registers are configured as a two-stage shift register.

2.11.4 Accelerator Operation

The accelerator maintains internal copies of the Source Address, Pattern Address, and Destination Address Registers which it increments as an accelerator operation progresses. The CPU may use the internal Source and Pattern Address Registers as the starting point of a subsequent accelerator operation by setting the appropriate bits in the ACL Reload Control Register (see Section 5.9.21). After an accelerator operation has completed, the internal Source Address Register (and/or Pattern Address Register) 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 should be initialized to zero when the system is powered-up, and if they need to be loaded with non-zero values for a State-Save, they should subsequently be loaded with zero before resuming operation. In other words, the programmer should ensure that the X Position and Y Position Registers in the queue are zero before any graphics operation is begun; but it is not necessary to spend time loading the registers before every operation.

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 Internal Source Address Register (ISA) will **not** be altered. Basically, by using the Address-Route (ADRO) and Data-Route (DARO) control fields in the ACL Routing Control Register (see Section 5.9.20), the programmer can perform the following types of operations:

ADRO	DARO	Data Movement	Address Step	Notes
00	000	Screen-to-Screen	BLT	No host involvement, all maps in display mem.
00	001	Host-to-Screen	BLT	Src data from host
00	010	Host-to-Screen	BLT	Mix data from host (e.g., Color Expansion)
01	000	Screen-to-Screen	host controls	e.g. draw circle, line, etc., pixel-by-pixel
01	001	Host-to-Screen	host controls	host provides Src data
01	010	Host-to-Screen	host controls	host provides Mix data
0x	100	Screen-to-Screen BLT	host provides	X Count (e.g., draw line segment)
0x	101	Screen-to-Screen BLT	host provides	Y Count (e.g., draw line segment)

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 can be fixed to "1", or be provided by the host; it cannot originate from the display memory.

2.11.5 Passing Map Data to the Accelerator

2.11.5.1 Virtual Bus Size

To specify the passing of data from the host to the accelerator, the concept of "Virtual Bus Size" has been developed. The Virtual Bus Size is used to eliminate the effects of different physical system buses on the way data is transferred to the accelerator. The Virtual Bus Size allows the chip to be programmed and to function in the exact same manner whether it is connected to an 8-bit system bus or a 32-bit system bus, for example.

The Virtual Bus Size can be programmed to 1-byte, 2-bytes, or 4-bytes. Internally, the Host Interface of the ET4000/W32i "waits" for the specified number of bytes, then releases the data to the accelerator. The Virtual Bus Size is only enforced when the CPU is passing Source or Mix Map data to an accelerated operation; all other writes to the chip operate normally. If the Virtual Bus Size is programmed to match the size of data transfer that is being done in the host assembly language, the accelerator will always give correct results no matter how many bus cycles are required to transfer the data, and no matter what address-order the bus cycles are in.
Several important restrictions arise from this approach:

1. All CPU double-word writes to the accelerator must be doubleword aligned.
2. All CPU word writes to the accelerator must be word aligned.
3. No restrictions on byte writes.

Once again, it is important to understand that these restrictions are only for writes of Source or Mix Map data to the accelerator; there are no restrictions on ordinary reads/writes to registers or to the display memory.

To handle word or doubleword writes to an unaligned Destination map, the MMU Memory Base Pointer Register can be set to any byte boundary. This permits the ET4000/W32i to manage all unaligned data, while the CPU acts as if the map is aligned. This also maximizes performance by eliminating double or triple bus cycles on the host bus.

If the host is supplying Source or Mix data and the XCNT is not a multiple of the number of bytes specified by the Virtual Bus Size, then the "extra" bytes at the end of the line will be ignored.

Note that the Address Registers contain **byte** addresses, so if a BLT is moving in the "Decreasing-X" and "Decreasing-Y" direction, the initial Pattern Address, for example, should point to the **last byte** of the Pattern map. This is independent of the programmed Virtual Bus Size. The Destination Address, however, **is** dependent on Virtual Bus Size. If a write through an MMU aperture is initiating the operation, the write should address the last "Virtual-Bus-Sized" element of the Destination Map. For example, if VBS is 4-bytes, the initiating write should address the last doubleword of the Destination Map. The Accelerator internally takes care of pointing to the correct starting byte.

2.11.5.2 Synchronization

Another important issue when passing data from the host to the accelerator is synchronization. What we have here is 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/W32i 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, since 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 set the Virtual Bus Size to the largest value possible (usually 4 bytes) when passing Source Map data; and set to one byte when passing Mix Map data.

2.11.5.3 Data Alignment

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.

2.11.6 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.

2.11.6.1 Line Drawing

The Graphics Accelerator can draw vertical, horizontal, and diagonal (slope of +1 or -1) lines by appropriate programming of certain registers:

XCNT	YCNT	Y OFFSET	TYPE OF LINE
n-1	0	don't care	Horizontal
0	n-1	w-1	Vertical
0	n-1	(w-1) ± 1	Diagonal

where: n = length of line (in bytes)
w = width of pixel map (in bytes)

Any line drawn using Bresenham's algorithm is made up of smaller line segments which are all horizontal, all vertical, or all diagonal. The only difference among these smaller line segments is the **length** of the segment. The accelerator allows the programmer to specify a destination starting address **and** the length (XCNT or YCNT) with a single bus write cycle to the chip. Thus, the inner loop of a line drawing routine need only contain a single write to the chip (plus whatever algorithmic processing must be done), thereby minimizing the number of bus transfers required to the chip. It is important to note that these write cycles **must** be a size of **one byte**, so the upper 4 bits (bits <11:8>) of the Count register must be previously loaded with the necessary value. This type of accelerator operation is achieved by programming the DARO field in the ACL Routing Control Register (see Section 5.9.20) to be 4 (to load the X Count Register), or 5 (to load the Y Count Register).

For simplicity, the table shown above is for 1 byte per pixel. It is a simple task to adapt the table to other pixel depths. Below is the same table, generalized to allow different pixel depths:

XCNT	YCNT	Y OFFSET	TYPE OF LINE
n * b - 1	0	don't care	Horizontal
b-1	n - 1	(w x b)-1	Vertical
b-1	n - 1	(w x b)-1 ± b	Diagonal

where: n = length of line (in pixels)
w = width of pixel map (in pixels)
b = bytes per pixel

2.11.6.2 Tiled and Fixed-Color Fills

Both the Source and Pattern Maps can be programmed to "wrap" or "tile" as the graphics operation progresses. The programmed starting address for the map indicates the **corner** of the tile which will be repeated through the operation.

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.

2.11.6.3 Color Expansion

The accelerator is capable of expanding a 1 bit-per-pixel (monochrome) pixel map into an 8 bit-per-pixel map. This is accomplished 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 Mix Map will draw the Source color into the Destination. Of course, it is also possible to make the Foreground ROP a function of the Source and Destination, and the Background ROP a function of the Pattern and Destination if this is required.

2.11.6.4 Clipping

For the most part, rectangular clipping 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.

2.11.7 Accelerator Interrupts

The ET4000/W32i is capable of generating a system interrupt on three 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).

2.11.8 Accelerator State Save/Restore

The ET4000/W32i 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. This type of 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 below outlines the steps required to save and restore the state of the Graphics Accelerator:

2.11.8.1 State-Save Interrupt Handler

while (STAT.WRST==1) /*wait for queue to be emptied */
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 not doing anything */
Read all accelerator registers from chip (including STAT) and save into local array, called SAVE1.
Write "1" to bit 0 (RSO) of ACL Operation State Register. /* "shifts" state from queue */
Read Source Address and Pattern Address Registers from chip and save into variables called ISA and IPA.
Write "1" to bit 0 (RSO) of ACL Operation State Register. /* "shifts" state from queue */
Read all accelerator registers from chip and save into local array, called SAVE2.
Write "1" to bit 4 (TO) of ACL Suspend/Terminate Register. /* terminates operation and resets accelerator */
while (STAT.RDST==1) /* wait for Read-status OK */
;
Write "0" to bit 4 (TO) of ACL Suspend/Terminate Register.
Done with State Save.

2.11.8.2 State-Restore Interrupt Handler

Load the array SAVE1 back into chip (including STAT).
Write "1" to bit 0 (RSO) of ACL Operation State Register. /* "shifts" state from queue */
Load ISA and IPA into Source and Pattern Address Registers.
Write "1" to bit 0 (RSO) of ACL Operation State Register. /* "shifts" state from queue */
Load the array SAVE2 back into chip.
Write (SAVE1.STAT & 8) to ACL Operation State Register. /* resume screen-to-screen op if necessary */
Done with State Restore.

3. ET4000/W32i Pin Descriptions

The ET4000/W32i 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:

- ISA 8/16 bits
- Micro Channel 8/16 bits
- Local Bus 386/486 SX/DX 16/32 bits
- VESA LBUS

A specific host bus type is selected by pulling the UCPC and AEN* input pins to appropriate levels at reset. A Power On Reset Initialize (PORI) scheme is used to determine different configurations for each bus type to ensure full hardware compatibility.

3.1 Power On Reset Initialize (PORI)

During the high-to-low transition of the REST signal, DB<15:0> is latched internally. These latched data bits are used to determine the host interface configuration. DB<15:0> are normally pulled high internally; these signals can be pulled down via a series resistor to ground, or driven low with a tri-state buffer during reset.

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

3.2 Bus Configurations

SYMBOL	PIN#	I/O	TYPE	FUNCTION
UCPC	16	I	TTL,PU, TS, B4	Bus type selection, latched during PORI. Pin is shared with Image Port IXCM* signal. Low: ISA bus interface protocol. High: MCA or Local Bus interface protocol, depending on the level on the AEN* input.
AEN*	32	I	TTL,PU	MCA or Local Bus selection. Pin is shared with ISA bus AEN* signal. Low : MCA bus interface protocol. High: Local Bus interface protocol.

UCPC	AEN*	Bus Type
L	x	ISA
H	L	MCA
H	H	Local Bus

Additional PORI configuration bits are described in each individual bus interface section.

3.3 Host Interface

3.3.1 ISA Bus

3.3.1.1 ISA PORI

SYMBOL	PIN#	I/O	TYPE	FUNCTION
IOD<2:0>	70,68 67	I	TTL,PU, TS,B4	CRTCB/Sprite I/O register map for I/O addresses 21xA, 21XB where x = IOD<2:0>. Pins are shared with DB<15:13>.
MONID<5:0>	57-52	I	TTL,PU, TS,B4	Bits 0 and 1 can be read back via Status Register 0. Bits 5 through 2 can be read back via the Feature Control Register. Pins are shared with DB<5:0>.
SYSW	65	I	TTL,PU	System Linear Wiring bit. See Table 6.2.1.2-1 in Section 6.2.1.2 to determine upper address bus configuration. Pin is shared with DB<11>.

3.3.1.2 ISA Bus Interface

SYMBOL	PIN#	I/O	TYPE	FUNCTION
SEGE*	50	I	TTL,PU, TS	Segment Enable Input for memory access.This is used to decode LA<23:20>. This input is latched internally during the high-to-low transition of the BUSALE input. 8/16-bit design : Decode LA<23:20> 16-bit Image Port design : Decode LA<23:20> 16-bit Only/System Linear design: Connect A<23>

3.3.1.2 ISA Bus Interface (cont'd)

SYMBOL	PIN#	I/O	TYPE	FUNCTION
A<22:20>	49-47	I	TTL,PU, TS,B4	These are the LA<22:20> address lines. They are latched internally on the high-to-low transition of the BUSALE input. Pins are shared with Image Port IXFS, IXLS, and IXOF signals, respectively. IMA Indexed Register F7 selects this function onto these pins.
A<19:17>	46-44	I	TTL,PU, TS,B4	These are the LA<19:17> address lines. They are latched internally on the high-to-low transition of the BUSALE input.
A<16>	43	I	TTL,PU, TS,B4	This is the SA<16> address line.
DB<15:0>	70, 68-62, 59-52	IO	TTL,PU, B4,TS	Data/Address time-multiplexed bus, A<15:0> input while ADRE* active and DB<15:0> input/ output while RDML*/RDMH* is active. SA<15:2> are latched internally by ET4000/W32i during the high-to-low transition of the BUSALE input and A<1:0> are latched during high-to-low transition of the read or write commands (MRDC*, MWTC*, IORC*, IOWC*).
RDMH*	73	O	TTL,PD, B4,TS	Active low signal used to enable the external tri-state DB<15:8> data bus driver.
RDML*	72	O	TTL,B8	Active low signal used to enable the external tri-state DB<7:0> data bus driver.
DIR	71	O	TTL,B8	A signal used to control the direction of bi-directional bus drivers connected to DB<15:0> between ET4000/W32i and ISA 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.
ADRE*	74	O	TTL,B8	Active low signal, used to enable the external tri-state A<15:0> address bus driver to ET4000W32i.
SBHE*	35	I	TTL,PU	Active low input signal, along with A<0>, identifies current bus cycle as an 8- or 16-bit access.

3.3.1.2 ISA Bus Interface (cont'd)

SYMBOL	PIN#	I/O	TYPE	FUNCTION
EBIO*	25	I	TTL,PU	This pin must be tied low. Low: BIOS is decoded from A<23:15> into 0C0000-0C7FFF. High: BIOS is decoded from A<23:17> into 0C0000-0DFFFF.
AEN*	32	I	TTL,PU	Active low I/O address valid signal.
MRDC*	30	I	CMOS,S	Active low command pulse for memory read access.
MWTC*	31	I	CMOS,S	Active low command pulse for memory write access.
IORC*	28	I	CMOS,S	Active low command pulse for I/O read access.
IOWC*	29	I	CMOS,S	Active low command pulse for I/O write access.
IOWW*	17	I	TTL,PU	Active low input signal to ET4000/W32i for 16-bit I/O write access. This signal is generated by delaying IOWC*. Note: This input is used only if 16-bit I/O is enabled via CRTC Indexed Register 36, bit 7, and DB<15:8> meets the setup time as specified. Otherwise, this input must be pulled low.
BUSALE	38	I	TTL,PU	A high-to-low transition is used to internally latch A<23:2> of the ISA address bus.
MCS16*	37	O	TTL,OC,B8	Active low output signal to identify ET4000/W32i memory as a 16-bit device.
XR16*	75	O	TTL,OC,B4	Active low output signal to identify ET4000/W32i I/O as a 16-bit device. If translation ROM is enabled (8-bit I/O, when CRTC Indexed Register 37 <3> = 1), this output becomes the external translation ROM enable signal. If CRTC Indexed Register 37 <3> = 1 and XROM is disabled, the output becomes the IO16 signal. Otherwise, AA<9> is enabled when CRTC Indexed Register 37 <3> = 0.
WAIT*	33	O	TTL,OC,B8	Active low signal, generated to asynchronously extend an ISA bus command cycle.
SRDY*	97	O	TTL,OC,B8	Active low output signal to identify current memory write cycle as a zero wait state cycle.

3.3.1.2 ISA Bus Interface (cont'd)

SYMBOL	PIN#	I/O	TYPE	FUNCTION
REST	34	I	CMOS,S	Active high reset signal to initialize the ET4000/W32i. When high, the chip is held in the reset state.

3.3.2 MCA Bus

3.3.2.1 MCA PORI

SYMBOL	PIN#	I/O	TYPE	FUNCTION
IOD<2:0>	70,68, 67	I	TTL,PU, TS,B4	CRTCB/Sprite I/O register map for I/O addresses 21xA, 21xB where x = IOD<2:0>. Pins are shared with D<15:13>.
PID<7:0>	59-52	I	TTL,PU, TS,B4	POS register ID. Pins are shared with D<7:0>. If PID<7:0>=00h-FEh then POS 100 =00h-FEh POS 101 =80h If PID<7:0>=FFh then POS 100 =FFh POS 101 =FFh

3.3.2.2 MCA Bus Interface

SYMBOL	PIN#	I/O	TYPE	FUNCTION
A<23:16>	50-43	I	TTL,PU, TS,B4	Upper 8 bits of address interface for MCA bus, these inputs are latched internally during the high-to-low transition of the CMD* input.
D<15:0>	70, 68-62, 59-52	IO	TTL,PU, B4,TS	Data/Address time-multiplexed bus, A<15:0> input while ADRE* active and D<15:0> input/output while RDML*/RDMH* is active. A<15:0> are latched internally by ET4000/W32i during the high-to-low transition of the CMD* input.
RDMH*	73	O	TTL,PD,B4	Active low signal used to enable external tri-state DB<15:8> data bus driver.
RDML*	72	O	TTL,B8,TS	Active low signal used to enable external tri-state DB<7:0> data bus driver.

3.3.2.2 MCA Bus Interface (cont'd)

SYMBOL	PIN#	I/O	TYPE	FUNCTION
DIR	71	O	TTL,B8	A signal used to control the direction of bi-directional bus drivers connected to DB<15:0> between ET4000/W32i and MCA 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.
ADRE*	74	O	TTL,B8	Active low signal, used to enable external tri-state A<15:0> address bus driver to ET4000/W32i.
SBHE*	35	I	TTL,PU	System Byte High Enable. Active low input signal, along with A<0>, identifies the current bus cycle as an 8- or 16-bit access.
MADE24	38	I	TTL,PU	Memory Address Enable 24. Active high input signal to indicate A<31:24> are undefined. This input must be driven high during all memory accesses.
SETUP*	28	I	CMOS,S	Card Setup. Active low input signal to ET4000/W32i for channel setup.
MIO*	29	I	CMOS,S	Memory or I/O status input. Low : I/O High: Memory
S<1:0>*	25,26	I	TTL,PU	Status bits. These inputs define the current bus cycle as read or write.
CMD*	31	I	CMOS,S	Command. Active low input signal to indicate a read/write command cycle.
IOWW*	17	I	TTL,PU	Active low input signal to ET4000/W32i for 16-bit I/O write access. This signal is generated by delaying CMD*. Note: This input is used only if 16-bit I/O is enabled via CRTC Indexed Register 36, bit 7, and DB<15:8> meets the setup time as specified. Otherwise, this input must be pulled low.
SFDBK*	97	O	TTL,TS,B8	Card Selected Feedback. Active low signal to indicate the availability of ET4000/W32i's memory or I/O resources to the Micro Channel host.

3.3.2.2 MCA Bus Interface (cont'd)

SYMBOL	PIN#	I/O	TYPE	FUNCTION
CHRDY	33	O	TTL,OC,B8	Channel Ready. Active tri-state indicates termination of a command cycle.
DS16*	37	O	OC,B8	Card Data Size 16. Active low output signal to identify ET4000/W32i memory as a 16-bit device.
XROM*	75	O	TS,B8,PU	Active low output signal to enable external translation ROM.
CHRESET	34	I	CMOS,S	Channel Reset. Active high reset signal to initialize the ET4000/W32i. When high, the chip is held in the reset state.

3.3.3 Local Bus

3.3.3.1 Local Bus PORI

SYMBOL	PIN#	I/O	TYPE	FUNCTION
SNPE	62	I	TTL,PU, TS,B4	Snoop Enable. When snoop feature is enabled, the W32i will notr respond with LOCAL* and RDY* during the DAC write operation. Pin is shared with D<8>. Low : External RAMDAC snoop enable. High: External RAMDAC snoop disable.
IOD<2:0>	70,68, 67	I	TTL,PU, B4,TS,	CRTCB/Sprite I/O register map for I/O addresses 21xA, 21xB where x = IOD<2:0>. Pins are shared with DB<15:13>.
WAT<1:0>	59,58	I	TTL,PU, TS,B4	Local Bus programmed wait state. Pins are shared with D<7:6>.

MIO*	WAT<1:0>	# CPU T2 States
H	LL	4
H	LH	3
H	HL	2
H	HH	1 (zero wait state)
L	LL	4
L	LH	3
L	HL	4
L	HH	3

3.3.3.1 Local Bus PORI (cont'd)

SYMBOL	PIN#	I/O	TYPE	FUNCTION
SXDX	63	I	TTL,PU, B4,TS,	SX/DX selection. Pin is shared with D<9>. Low : BE3*, BE2*, BE1*, BE0* decoding on lower 4 bytes of address. High: A<1>, BHE*, BLE* decoding on lower 4 bytes of address.
DVCK	64	I	TTL,PU, TS,B4	Internal local bus (LCLK) clock divide. Pin is shared with D<10>. Low : LCLK = BCLK High: LCLK = BCLK/2
DELC	65	I	TTL,PU, TS,B4	Command delay. Pin is shared with D<11>. Low : Internal command starts at first T2 after the low-to-high transition of ADS*. High: Internal command starts at first T2.
BU32*	66	I	TTL,PU, TS,B4	32-bit data bus enable. Pin is shared with D<12>. Low : Enables 32-bit data bus operation. High: Enables 16-bit data bus operation.
MONID<3:0>	55-52	IO	TTL,PU, TS,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 D<3:0>.
RDYS	57	I	TTL,PU, TS,B4	RDY* type select. Pin is shared with D<5>. Low : RDY* ouput = tri-state High: RDY* ouput = totem pole
DISB	56	I	TTL,PU, TS,B4	Disable BIOS. Pin is shared with D<4>. Low : Enable BIOS decode. High: Disable BIOS decode.

3.3.3.1 Local Bus PORI (cont'd)

SYMBOL	PIN#	I/O	TYPE	FUNCTION
BLOK	15	IO	TTL,PU, TS,B2	I/O space decode. Determines the I/O space that the W32i decodes. When I/O space is decoded, then the W32ii will generate LOCAL* to claim the cycle, and RDY* to terminate the cycle. When BLOK is low the following space is decoded:

03B0-03BE (if Miscellaneous Output Register bit 0 = 0)
03BF
03C0-03CF
03D0-03DF (if Miscellaneous Output Register bit 0 = 1)
21xA-21xB (x=IOD<2:0> section 3.3.3.1)
46E8

If BLOK is high, the following space is decoded:

03B0-03BB (if Miscellaneous Output Register bit 0 = 0)
03BF-03C5
03C6-03C9
03CA-03CF
03D0-03DC, 03DE (if Miscellaneous Output Register bit 0 = 1)
21xA-21xB (x=IOD<2:0> section 3.3.3.1)
46E8

Note: When BLOK is enabled, the DAC cannot be disabled. 03BF is unaffected by BLOK decode.

This function is used to facilitate the use of motherboard controllers that have the capability to internally generate LOCAL* for the VGA controller space.

3.3.3.2 Local Bus SX/DX 16-bit Interface

SYMBOL	PIN#	I/O	TYPE	FUNCTION
A<23:16>	50-43	I	TTL,PU, TS,B4	Upper 8-bit address interface for local bus, these inputs are latched internally during the low to high transition of LCLK at the start of internal command.
D<15:0>	70, 68-62, 59-52	IO	TTL,PU, B4,TS	Data/Address time-multiplexed bus. A<15:1>, BLE* input for SX Bus or A<15:2>, BE1*, and BE0* for DX bus while ADRE* active and DB<15:0> input/output while RDMX* is active. A<15:0>, BE1*, and BE0* are latched internally during the low to high transition of LCLK at the start of internal command. If External Palette Memory is enabled, DB<9:8> also outputs RS<1:0> (RAMDAC Register Select bits) at the proper time (see Section 4.5, Local Bus Timing specifications)

	Local Bus
16-bit	32-bit
D<15>	A<15> A<15>
D<14>	A<14> A<14>
.	. .
.	. .
D<2>	A<2> A<2>
D<1>	A<1> BE1*
D<0>	BLE*BE0*

RDMX*	72	O	TTL,B8	Active low signal used to enable DB<15:0> data bus driver.
DIR	71	O	TTL,B8	A signal used to control the direction of bi-directional bus drivers connected to DB<15:0> between ET4000/W32i and 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.
ADRE*	74	O	TTL,B8	Active low signal, used to enable A<15:0> of Local Bus address to ET4000/W32i.
ADS*	30	I	CMOS,S	Active low input signal indicates address and status valid.
SEG1	28	I	CMOS,S	Active high input signal from upper address decode. This input must be high during memory access cycle to ET4000/W32i.

3.3.3.2 Local Bus SX/DX 16-bit Interface (cont'd)

SYMBOL	PIN#	I/O	TYPE	FUNCTION
SEG0	38	I	TTL,PU	Active high input signal from upper address decode. This input must be high during memory access cycle to ET4000/W32i.
SEG2	26	I	TTL,PU	Active high input signal from upper address decode. This input must be high during memory access cycle to ET4000/W32i.
BE3*	31	I	CMOS,S	Active low input signal in DX mode to indicate byte 3 data is enabled.
BHE*	35	I	TTL,PU	In SX mode, this is an active low input signal which, along with A<0>, identifies the current bus cycle as an 8- or 16-bit access. In DX mode, this is tied to the CPU's BE2* to indicate that byte 2 is enabled.
MIO*	29	I	CMOS,S	Memory or I/O status input. Low : I/O High: Memory
W/R*	25	I	TTL,PU	This signal defines the current bus cycle as read or write.
LOCAL*	97	O	TTL,B8,TS	Active low signal to indicate the availability of ET4000/W32i's memory or I/O resources to the Local Bus.
BS16*	37	O	TTL,OC,B8	Active low output signal to identify ET4000/ W32i memory as a 16-bit device.
RDY*	33	O	TTL,OC, B8,TS	Active low output indicates termination of a command cycle. Open collector or totem pole output selected by RDYS PORI bit.
RDYR*	73	I	TTL,PD, TS,B4	Ready Return. Active low signal used to extend the current Local Bus cycle.
REST	34	I	CMOS,S	Active high reset signal to initialize the ET4000/W32i. When high, the chip is held in the reset state.
BCLK	17	I	TTL,PU	Local Bus system clock input.

3.3.3.3 Local Bus DX 32-bit Interface

SYMBOL	PIN#	I/O	TYPE	FUNCTION
D<31:24>	159,122, 119,82, 79,42, 39,2	IO	TTL,TS,B4	Upper 8-bit data bus interface.
D<23:0>	50-43, 70, 68-62, 59-52	IO	TTL,PU, B4,TS	Data/Address time multiplexed bus, A<23:2>, BE1*, and BE0* while ADRE* active and DB<23:0> input/output while RDMX* is active. A<23:2>, BE1*, and BE0* are latched internally during the low to high transition of LCLK at the start of internal command. If External Palette Memory is enabled, DB<9:8> also outputs RS<1:0> (RAMDAC Register Select bits) at the proper time (see Section 4.5, Local Bus Timing specifications).
RDMX*	72	O	TTL,B8	Active low signal used to enable DB<31:0> data bus driver.
DIR	71	O	TTL,B8	A signal used to control the direction of bi-directional bus drivers connected to DB<31:0> between ET4000/W32i and 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.
ADRE*	74	O	TTL,B8	Active low signal, used to enable A<23:2>, BE1* and BE0* of Local Bus address to ET4000/W32i.
ADS*	30	I	CMOS,S	Active low input signal indicates address and status valid.
SEG1	28	I	CMOS,S	Active high input signal from upper address decode. This input must be high during memory access cycle to ET4000/W32i.

Local Bus	32-bit
D<23>	A<23>
D<22>	A<22>
.	.
.	.
D<2>	A<2>
D<1>	BE1*
D<0>	BE0*

3.3.3.3 Local Bus DX 32-bit Interface (cont'd)

SYMBOL	PIN#	I/O	TYPE	FUNCTION
SEG0	38	I	TTL,PU	Active high input signal from upper address decode. This input must be high during memory access cycle to ET4000/W32i.
SEG2	26	I	TTL,PU	Active high input signal from upper address decode. This input must be high during memory access cycle to ET4000/W32i.
BE3*	31	I	CMOS,S	Active low input signal to indicate byte 3 data is enabled.
BE2*	35	I	TTL,PU	Active low input signal to indicate byte 2 data is enabled.
MIO*	29	I	CMOS,S	Memory or I/O status input. Low : I/O High: Memory
W/R*	25	I	TTL,PU	This signal defines the current bus cycle as read or write.
LOCAL*	97	O	TTL,B8,TS	Active low signal to indicate the availability of ET4000/W32i's memory or I/O resources to the Local Bus.
BS16*	37	O	TTL,OC,B8	Active low output signal to identify ET4000/W32i memory as a 16-bit device.
RDY*	33	O	TTL,OC, B8,TS	Active low output indicates termination of a command cycle. Open collector or totem pole output selected by RDYS PORI bit.
RDYR*	73	I	TTL,PD, TS,B4	Ready Return. Active low signal used to extend the current Local Bus cycle.
REST	34	I	CMOS,S	Active high reset signal to initialize the ET4000/W32i. When high, the chip is held in the reset state.
BCLK	17	I	TTL,PU	Local Bus system clock input.

3.4 Clock Interface

SYMBOL	PIN#	I/O	TYPE	FUNCTION
SCLK	22	I	TTL	System Clock used to generate all display memory timing and ET4000/W32i internal sequencing.
MCLK	19	I	TTL	A variable frequency clock used to generate the necessary video, vertical, and horizontal timing for the ET4000/W32i. 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>	15,23, 24	O	TTL,B2, PU,TS	Clock Select. Used to select 1 of 8 possible MCLK frequencies.
CS<3>	36	O	TTL,B4	Clock Select. Additional MCLK select signal. Pin is shared with Image Port IXRD signal. IMA Indexed Register F7, bit 0 selects the function of this pin.
CS<4>	81	O	TTL,B4	Clock Select. Additional MCLK select signal. Pin is shared with CRTCB/Sprite BDE signal. IMA Indexed Register F7, bit 7 selects the function of this pin.

3.5 General Interface

SYMBOL	PIN#	I/O	TYPE	FUNCTION
SYNR	98	IO	TTL	Active high synchronous reset signal. When active, it indicates a request to reset the ET4000/W32i's internal LINE and CHARACTER counters. Pin is shared with the Image Port IDMK signal and TKN/SP <0>. IMA Indexed Register F7, bits 0 and 6 select the function of this pin. (Bits 0 and 6 must = 0 for this to be active.)
SWSE	18	IO	TTL	Input status, can be read back via Input Status Register 0, bit 4. Pin is shared with the Image Port IXWQ* signal and TKN/SP <1>. IMA Indexed Register F7, bit 0 selects the function of this pin. (Bits 0 and 6 must = 0 for this to be active.)
ROME*	76	O	TTL,B2	Active low signal, used to enable external BIOS ROM during ROM read operation.
IRQ	27	O	TTL,OC,B8	Interrupt Request. Active tri-state signal to indicate an interrupting condition.

3.6 Display Memory Interface

3.6.1 Bank A (MD<15:0>)

SYMBOL	PIN#	I/O	TYPE	FUNCTION
AA<9:0>	75, 4-12	O	TTL,TS,B4	Row/column time multiplexed address bus outputs. AA<9> is available on pin 75 when CRTC Indexed Register 37, bit 3 = 0.
MWA*	13	O	TTL,TS,B8	Active low write command output.
RASA*	135	O	TTL,TS,B4	Active low row address strobe.
CAS<0>*	139	O	TTL,TS,B8	Active low column address strobe for MD<7:0>.
CAS<1>*	138	O	TTL,TS,B8	Active low column address strobe for MD<15:8>.
CAS<4>*	123	O	TTL,TS,B8	Active low column address strobe for MD<7:0> when interleave is enabled (CRTC Indexed Register 32, bit 7 = 1).
CAS<5>*	3	O	TTL,TS,B8	Active low column address strobe for MD<15:8> when interleave is enabled (CRTC Indexed Register 32, bit 7 = 1).
MD<15:0>	142-148 150-158	IO	TTL,PU,B4, TS	Display memory data bus.

3.6.2 Bank B (MD<31:16>)

SYMBOL	PIN#	I/O	TYPE	FUNCTION
AB<9:0>	81, 124-130, 132-133	O	TTL,TS,B4	Row/column time multiplexed address bus outputs. AB<9> is available on pin 81 when CRTC Indexed Register 37, bit 3 = 0.
MWB*	117	O	TTL,TS,B8	Active low write command output.
RASB*	134	O	TTL,TS,B4	Active low row address strobe.
CAS<2>*	137	O	TTL,TS,B8	Active low column address strobe for MD<23:16>.
CAS<3>*	136	O	TTL,TS,B8	Active low column address strobe for MD<31:24>.
CAS<6>*	96	O	TTL,TS,B8	Active low column address strobe for MD<23:16> when interleave is enabled (CRTC Indexed Register 32, bit 7 = 1).
CAS<7>*	95	O	TTL,TS,B8	Active low column address strobe for MD<31:24> when interleave is enabled (CRTC Indexed Register 32, bit 7 = 1).
MD<31:16>	99, 102-116	IO	TTL,PU, B4,TS	Display memory data bus.

3.7 Display Interface

SYMBOL	PIN#	I/O	TYPE	FUNCTION
PCLK	92	O	TTL,TS,B8	Used to sample pixel data AP<15:0>, and SP<1:0>.
AP<7:0>	91-84	O	TTL,TS,B8	Display pixel data bus connects to bits <7:0> of external Digital to Analog Converter (DAC) pixel data inputs.
MBS	83	O	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.
PMER*	78	O	TTL,B2	Active low external Digital to Analog Converter (DAC) register read command.
PMEW*	77	O	TTL,B2	Active low external Digital to Analog Converter (DAC) register write command.
VS	93	O	TTL,TS,B4	Vertical retrace synchronization, supplied to the CRT monitor.
HS	94	O	TTL,TS,B4	Horizontal retrace synchronization, supplied to the CRT monitor.

3.7.1 Hardware Sprite

SYMBOL	PIN#	I/O	TYPE	FUNCTION
BDE	81	O	TTL,B8	CRTCB/Sprite display enable timing, synchronized to first and last pixel of display data. Pin is shared with Clock Select CS<4>. IMA Indexed Register F7, bit 7 selects the function of this pin. This pin is used as AB<9> when CRTC Indexed Register 37, bit 3 = 0).

3.7.2 Token Status

SYMBOL	PIN#	I/O	TYPE	FUNCTION

TKN<1:0>	18,98	O	TTL,B2	Token Status output. Pins are shared with Hardware Sprite SP<1:0> signals, and SWSE (18) and SYNR (98).Token Status output is enabled when IMA Indexed Register F7<6> = 1 and F7<0> = 0, or, CRTCB/Sprite Control Indexed Register EF<1:0> ≠ <0,0>.
				
				If CRTC Index 35, bit 5 = 0, then
				TKN<1> = ET4000/W32i's MCU is processing FONT cycle.
				TKN<0> = ET4000/W32i'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>	18,98	O	TTL,B2	Sprite data bus to external XGA-compatible DAC. Pins are shared with TKN<1:0> signals, and SWSE (18) and SYNR (98). Sprite data output is enabled when IMA Indexed Register 7<6> = 1 and F7<7> = 1, and CRTCB/Sprite Control Indexed Register EF<1:0> = <0,0>. IMA Indexed Register F7, bit 7, and CRTCB/Sprite Control Register Indexed Register EF<1> select the function of this pin.

3.8 Image Port Interface

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
IM<7:0>	159,122, 119,82, 79,42, 39,2	I	TTL,TS,B4	Image Data bus from external image processing device. Pins are shared with bits <31:24> of host interface data bus.
IDMK	98	I	TTL	Image Data Mask. Pin is shared with SYNR signal. Low : ET4000/W32i ignores the current data byte; it only steps the internal address sequencer. High: ET4000/W32i accepts the current data byte and writes it to display memory.
IXWQ*	18	I	TTL	External Write Request. Pin is shared with SWSE signal. IM<7:0> and IDMK are strobed into the ET4000/W32i during the low to high transition of this signal.
IXRD	36	O	TTL,B4	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. Pin is shared with Clock Select pin CS<3>.
IXFS	49	I	TTL,PU, TS,B4	External Frame Synchronization. Low to high edge indicates the start of a new frame of image data. Pin is shared with bit 22 of the host address bus.
IXLS	48	I	TTL,PU, TS,B4	External Line Synchronization. Low to high edge indicates the start of a new line of image data. Pin is shared with bit 21 of the host address bus.
IXOF	47	I	TTL,PU, TS,B4	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. Pin is shared with bit 20 of the host address bus.

3.8 Image Port Interface (cont'd)

SYMBOL	PIN#	I/O	TYPE	FUNCTION
IXCM*	16	O	TTL,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. The ET4000/W32i will provide control for data/address bus steering at on-board host interface logic. Pin is shared with Bus Configuration UCPC signal.

3.9 Power Source Interface

SYMBOL	PIN#	I/O	TYPE	FUNCTION
VSS	14,21, 40,51, 61,69, 80,101, 120,131, 141,149	I	PWR	Ground.
VDD	1,20, 41,60, 100,118, 121,140, 160	I	PWR	+5V

3.10 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.

3.10.1 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					
	TTL		2.0			V
	CMOS SCHMITT trigger		4.0			
V_{IL}	Low level input voltage					
	TTL				0.8	V
	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					
	BUFFER B2	I_{OH} = -2 mA	2.4			V
	BUFFER B4	I_{OH} = -4 mA				
	BUFFER 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	.4			
	BUFFER B4	I_{OL} = 4 mA				
	BUFFER 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/W32i Timing Specifications

4.1 ISA Bus Timing

NO.	DESCRIPTION	MIN(ns)	MAX(ns)
A1	BUSALE min. pulse width	2.8	—
A2	AEN setup time to BUSALE low	4.4	—
A3	AEN hold time from BUSALE low	2.5	—
A4	AEN hold time from IORC*, IOWC* high	5.0	—
A5	SBHE* setup time to IORC*,IOWC*, MWTC*,MRDC* low	0.0	—
A6	SBHE* hold time from IORC*,IOWC*, MWTC*,MRDC* low	10.0	—
A7	A<1:0> setup time to IORC*,IOWC*, MWTC*,MRDC* low	3.0	—
A8	A<1:0> hold time from IORC*,IOWC*, MWTC*,MRDC* low	3.3	—
A9	A<23:2> setup time to BUSALE low	4.4	—
A10	A<23:2> hold time from BUSALE low	4.0	—
A11	DB<15:0> setup time to IOWC* high	0.0	—
A12	DB<15:0> input delay from MWTC* low	—	1.5s+0.3
A13	DB<15:0> hold time from IOWC*, MWTC* high	8.0	—
A14	DB<15:0> delay from IORC* low	—	21.8
A15	DB<15:0> setup to IOCHRDY Z	-4.2	—
A16	DB<15:0> float delay from IORC*, MRDC* high	—	18.0
A17	IORC*,IOWC* min. high pulse width	30.0	—
A18	IORC*,IOWC* min. low pulse width	30.0	—
A19	MWTC*,MRDC* min. high pulse width	30.0	—
A20	MWTC*,MRDC* min. low pulse width	1.5s+3	—
A21	ADRE* high delay from IORC*,IOWC*, MWTC*,MRDC* low	—	14.0
A22	ADRE* low delay from IORC*,IOWC*, MWTC*,MRDC* high	—	18.0
A23	RDML*,RDMH* low delay from IORC*, IOWC*,MWTC*,MRDC* low	—	21.0
A24	RDML*,RDMH* high delay from IORC*, IOWC*,MWTC*,MRDC* high	—	20.0
A25	DIR delay from IORC*,MRDC* low	—	13.5
A26	DIR delay from IORC*,MRDC* high	—	18.0
A27	IOCS16* low delay from A<15:0>, AEN low	—	18.0
A28	MCS16* low delay from A<23:17>	—	11.6
A29	IOWW* low delay from DB<7:0> valid	25.0	—
A30	IOWW* low min pulse width	25.0	—
A32	IOCHRDY* low delay from MWTC* MRDC*	—	22.7
A33	SRDY* low delay from MWTC* low	—	9.2
A34	ROME* low delay from MRDC* low	—	17.5
A35	ROME* high delay from MRDC* high	—	13.7
A36	IXCM* low delay from MRDC*, MWTC* low	—	17.5
A37	IXCM* high delay from MRDC*, MWTC* high	—	13.7

s = SCLK clock period

4.2 Micro Channel Bus Timing

NO.	DESCRIPTION	MIN(ns)	MAX(ns)
B1	S<1:0>*,M/IO*,SETUP* setup time to CMD* low	0.0	—
B2	S<1:0>*,M/IO*,SETUP* hold time from CMD* low	8.0	—
B3	CMD* high min. pulse width	30.0	—
B4	CMD* low min. pulse width	1.5s+3.05	—
B5	A<23:16>, SBHE*,MADE24 setup time to CMD* low	-0.15	—
B6	A<23:16>, SBHE*,MADE24 hold time from CMD* low	8.18	—
B7	A<15:0> setup time to CMD*	0.0	—
B8	A<15:0> hold time from CMD*	7.0	—
B9	DB<15:0> IO setup time to CMD* high	-3.3	—
B10	DB<15:0> memory input delay from CMD* low	—	1.5s+.3
B11	DB<15:0> hold time from CMD* high	12.0	—
B12	DB<15:0> delay from CMD* low	21.8	—
B13	DB<15:0> setup to CHRDY Z	67.0	—
B14	DB<15:0> float delay from read CMD* high	—	12.0
B15	ADRE* high delay from CMD* low	—	15.0
B16	ADRE* low delay from CMD* high	—	26.4
B17	RDMH*,RDML* low delay from CMD* low	—	23.4
B18	RDMH*,RDML* high delay from CMD* high	—	22.0
B19	DIR to (read) to CMD* low	—	18.3
B20	DIR to (read) to CMD* high	—	15.0
B21	CS16* low delay from A<15:0>, S<1:0> (IO)	—	22.2
B22	CS16* low delay from A<23:15>, S<1:0> (memory)	—	11.6
B23	CS16* float delay from CMD* high	—	10.6
B24	IOWW* delay from DB<15:0> valid	25.0	—
B25	IOWW* min. pulse width	25.0	—
B26	SFDB* low delay from A<23:0>	—	21.1
B27	SFDB* high delay from CMD* high	—	15.5
B28	CHRDY low delay from A<23:0> (early wait enable)	—	14.1
B29	CHRDY high delay from CMD* low (early wait enable)	—	25.3
B30	CHRDY low delay from CMD* low	—	19.6
B31	ROME* low delay from CMD* low	—	15.4
B32	ROME* high delay from CMD* high	—	15.2
B33	IXCM* low delay from CMD* low	—	15.4
B34	IXCM* high delay from CMD* high	—	15.2

s = SCLK clock period

4.3 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	DB<31:0> input delay	—	1.5s+.3
C13	DB<31:0> setup time to BCLK high	-1.7	—
C14	DB<31:0> hold time from BCLK high	4.0	—
C15	DB<31:0> delay from BCLK high	—	20.8
C16	DB<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	RDY* low delay from BCLK high	—	12.0
C26	RDY* 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	RDYR* setup time to BCLK high	11.0	—
C30	RDYR* 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	RDY* tri-state delay	—	8.5

s = SCLK clock period

4.4 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	SP<1:0> delay time from PCLK	1.4	5.8
D9	BDE delay time from PCLK	3.5	9.1

4.5 Display Memory Timing - Non-interleaved

NO.	DESCRIPTION	MIN(ns)	MAX(ns)
E1	AA,AB address setup time to RAS	32.30	36.12
E2	AA,AB address hold time to RAS	19.01	26.37
E3	AA,AB address setup time to CAS	11.81	29.20
E4	AA,AB address hold time to CAS	25.93	28.19
E5	RAS low to CAS low delay	(3*SCLK)-6.92	(3*SCLK)-1.82
E6			
E7	CAS low pulse width	SCLK-3.13	SCLK-0.82
E8	CAS high pulse width	SCLK+0.82	SCLK-3.13
E9			
E10	MWA*,MWB* setup time to CAS	9.88	10.70
E11	CAS low to MWA*,MWB* high	SCLK-10.34	SCLK-9.30
E12	MD out setup time to CAS	10.26	10.81
E13	MD out hold time	24.95	27.93
E14	MD in setup time[1]	1.63	—
	MD in setup time[2]	0.209	—
E15	MD in hold time[1]	6.374	—
	MD in hold time[2]	11.204	—
E16	RAS high pulse width	(3*SCLK)+1.45	(3*SCLK)+5.5
E17	CAS cycle time	SCLK	SCLK

[1] = Indexed Register 37<2> = 1

[2] = Indexed Register 37<2> = 0

4.6 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	UCPC,DB<15:0> setup time to REST low	-0.9	—
F7	UCPC,DB<15:0> hold time from REST low	5.4	—

c = greatest of SCLK, MCLK, or BCLK period

4.7 Translation ROM Interface

NO.	DESCRIPTION	MIN(ns)	MAX(ns)
G1	XROM* low delay from commands	—	19.4
G2	XROM* high delay from commands	—	18.3
G3	Translation address delay from CMD*,IORC*,IOWC* low	—	2s+20.1
G4	DB<7:0> setup time to commands	0.3	—
G5	DB<7:0> hold time from commands	12.0	—

4.8 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

4.9 Display Memory Timing - Interleaved

NO.	DESCRIPTION	MIN(ns)	MAX(ns)
J1	AA,AB address setup time to RAS	9.55	30.60
J2	AA,AB address hold time to RAS	5.58	8.12
J3	AA,AB address setup time to CAS	7.71	10.11
J4	AA,AB address hold time to CAS	9.89	12.29
J5	RAS low to CAS low delay	(3*SCLK)-6.71	(3*SCLK)-1.77
J6			
J7	CAS low pulse width	SCLK-3.13	SCLK-0.82
J8	CAS high pulse width	SCLK+0.82	SCLK-3.13
J9	RAS to second CAS delay	(4*SCLK)-6.71	(4*SCLK)-1.77
J10	MWA*,MWB* setup time to CAS	0.72	2.68
J11	CAS low to MWA*,MWB* high	SCLK-1.32	SCLK-0.36
J12	MD out setup time to CAS	3.90	9.11
J13	MD out hold time	9.63	11.31
J14	MD in setup time[1]	3.583	—
	MD in setup time[2]	2.162	—
J15	MD in hold time[1]	6.374	—
	MD in hold time[2]	11.204	—
J16	RAS high pulse width	(3*SCLK)+1.45	(3*SCLK)+5.5
J17	CAS cycle time	SCLK	SCLK

[1] = Indexed Register 37<2> = 1

[2] = Indexed Register 37<2> = 0

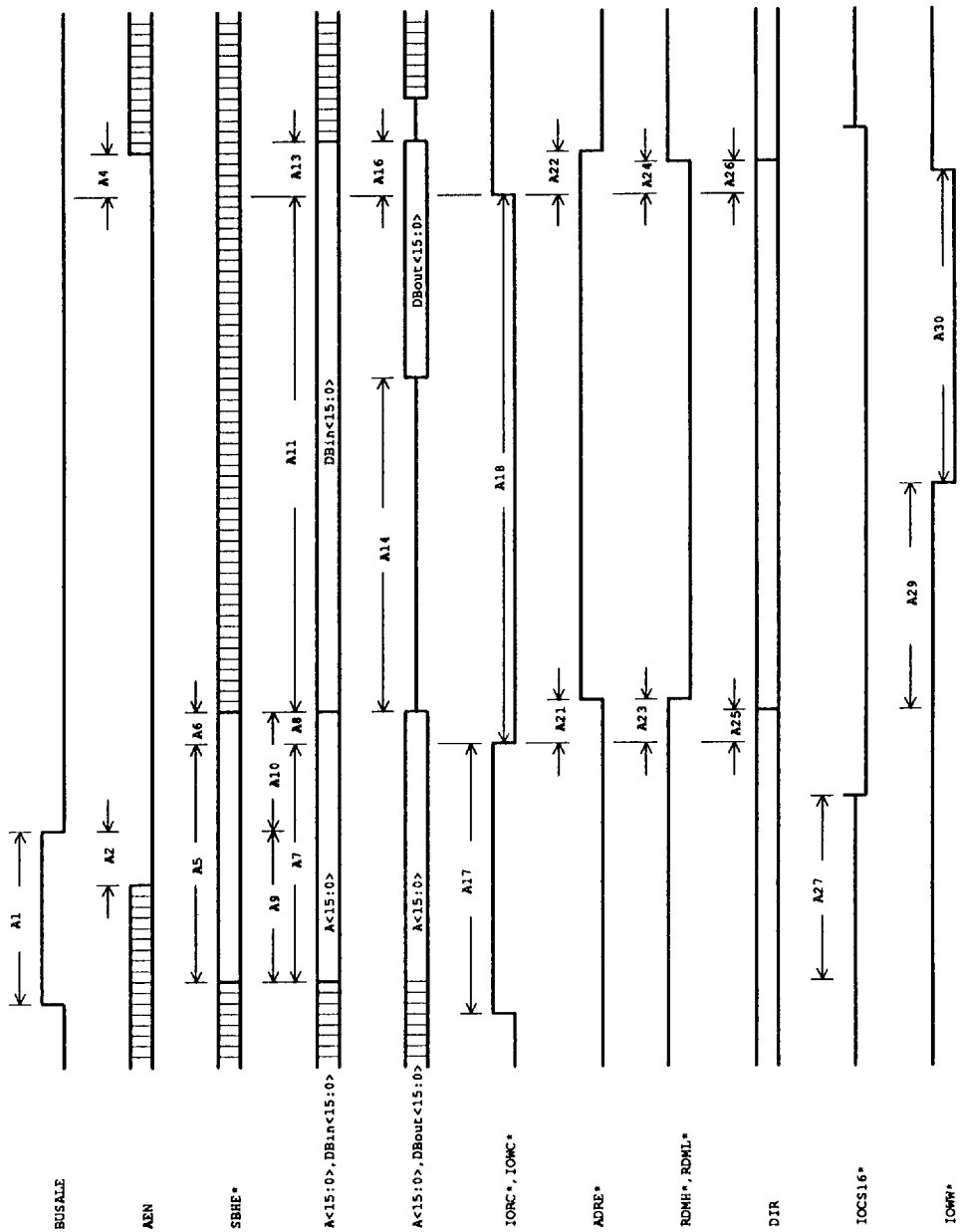


Figure 4.10-1 ISA Bus IO Read/Write Timing

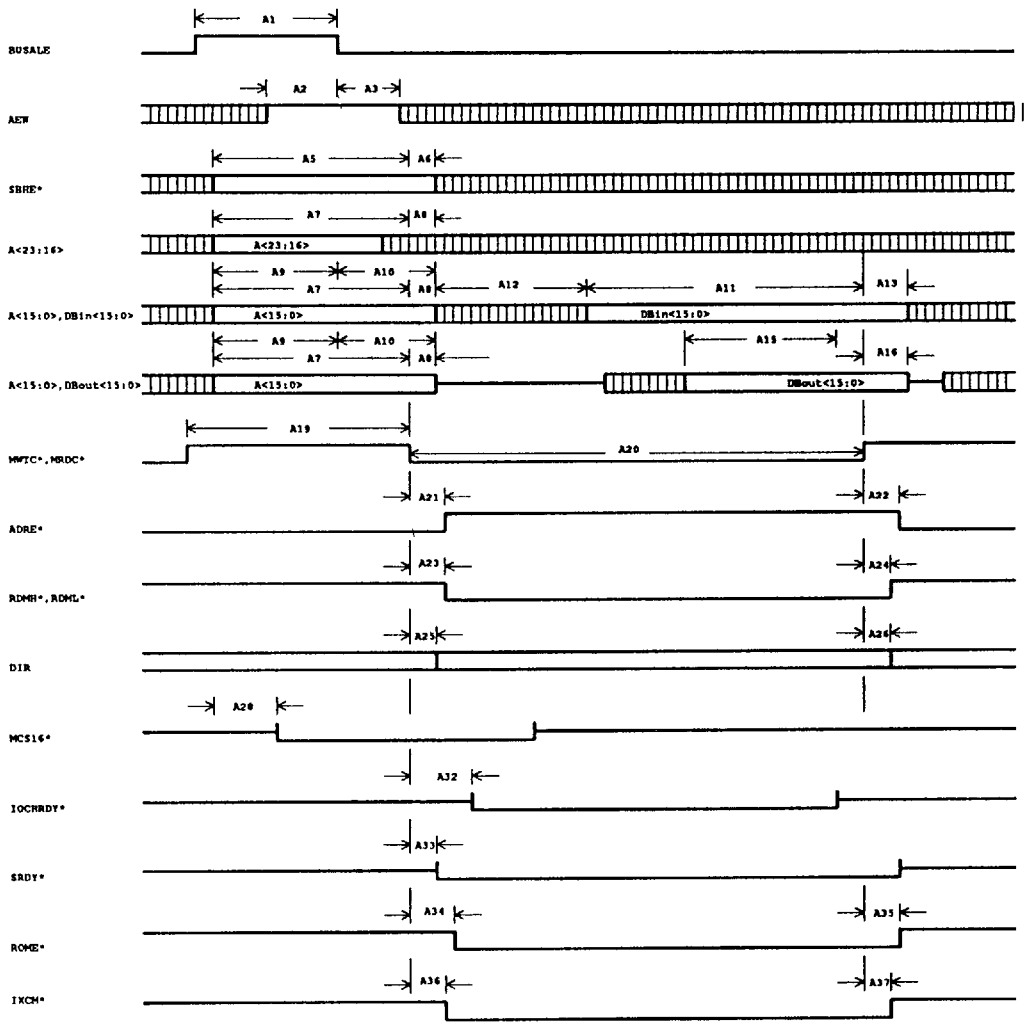


Figure 4.10-2 ISA Bus Memory Read/Write Timing

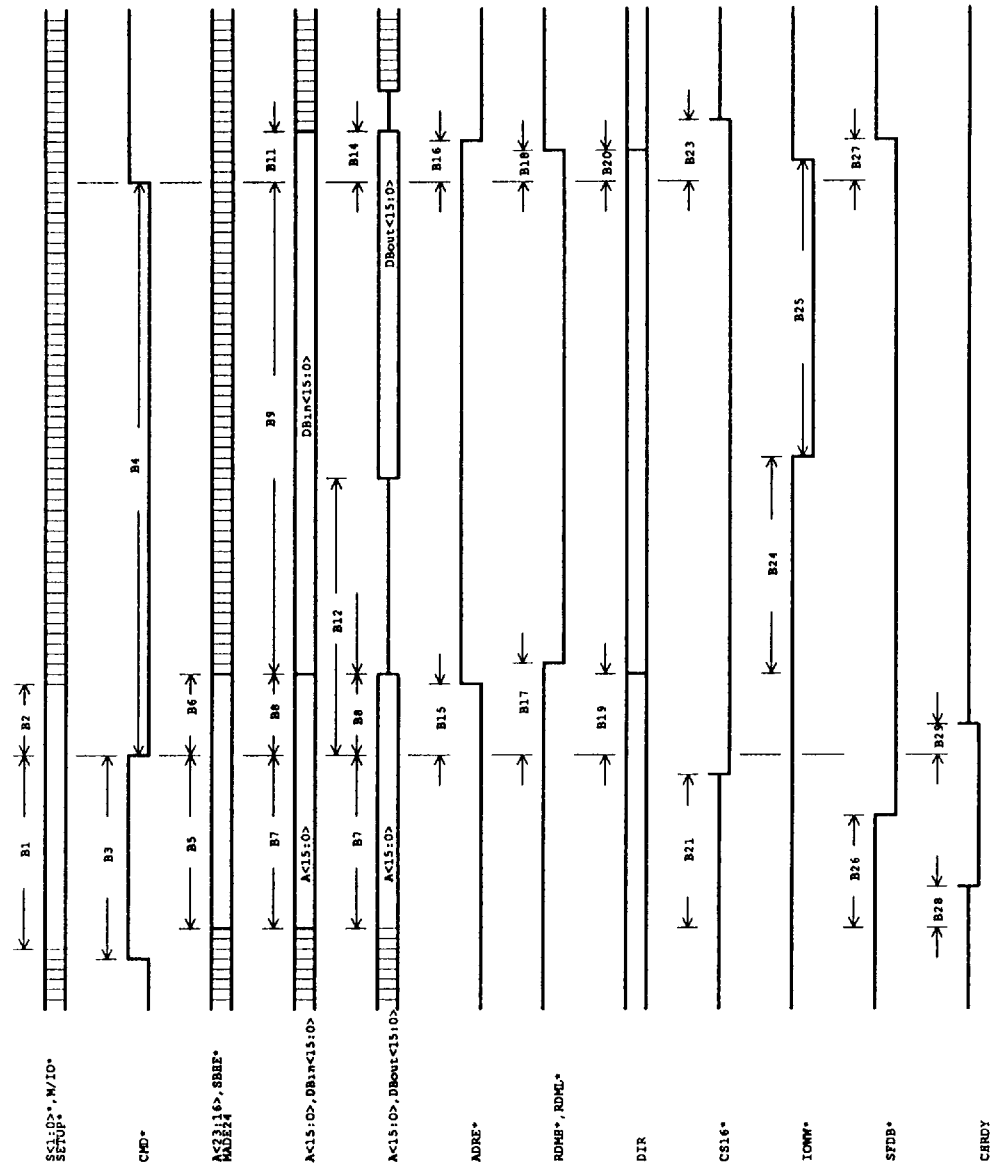


Figure 4.10-3 Micro Channel Bus IO Read/Write Timing

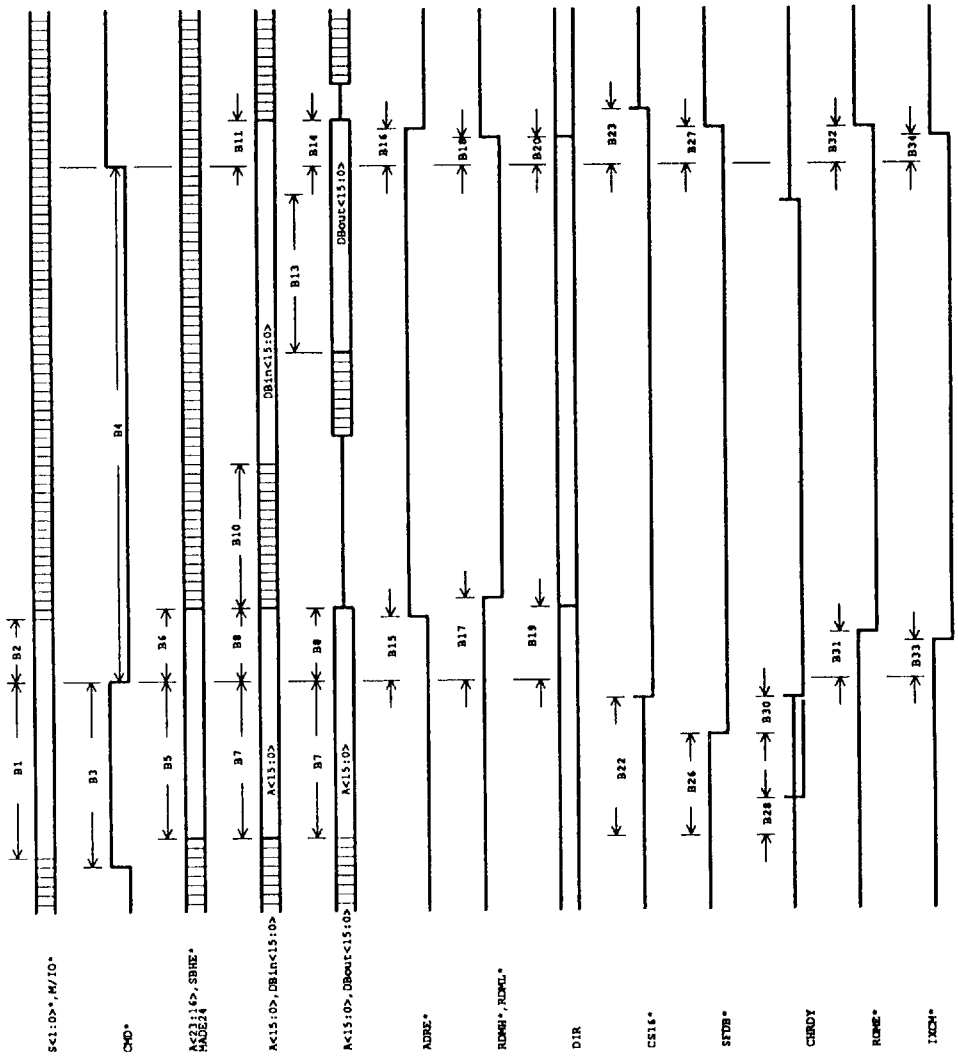


Figure 4.10-4 Micro Channel Bus Memory Read/Write Timing

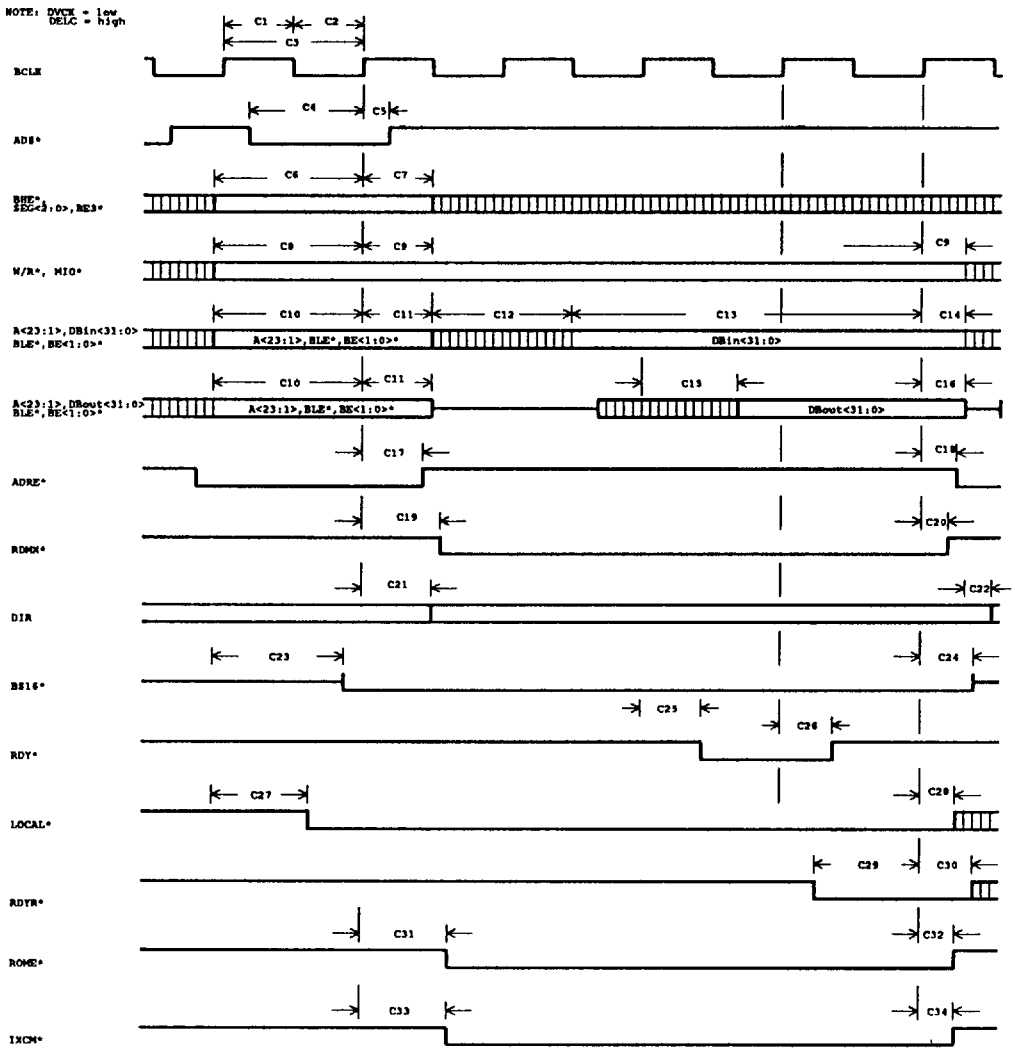


Figure 4.10-5 Local Bus Read/Write Timing

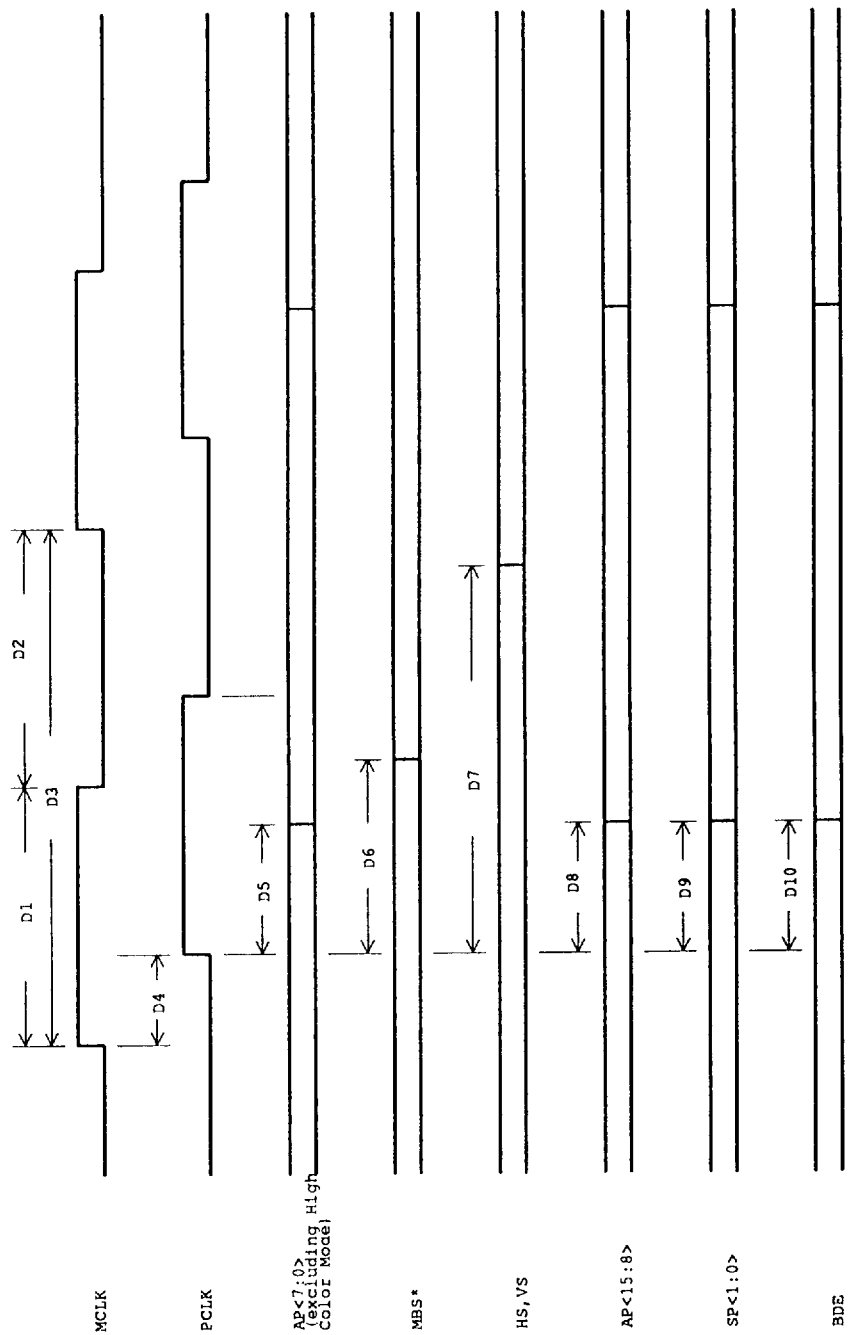


Figure 4.10-6 Video Bus Interface Timing

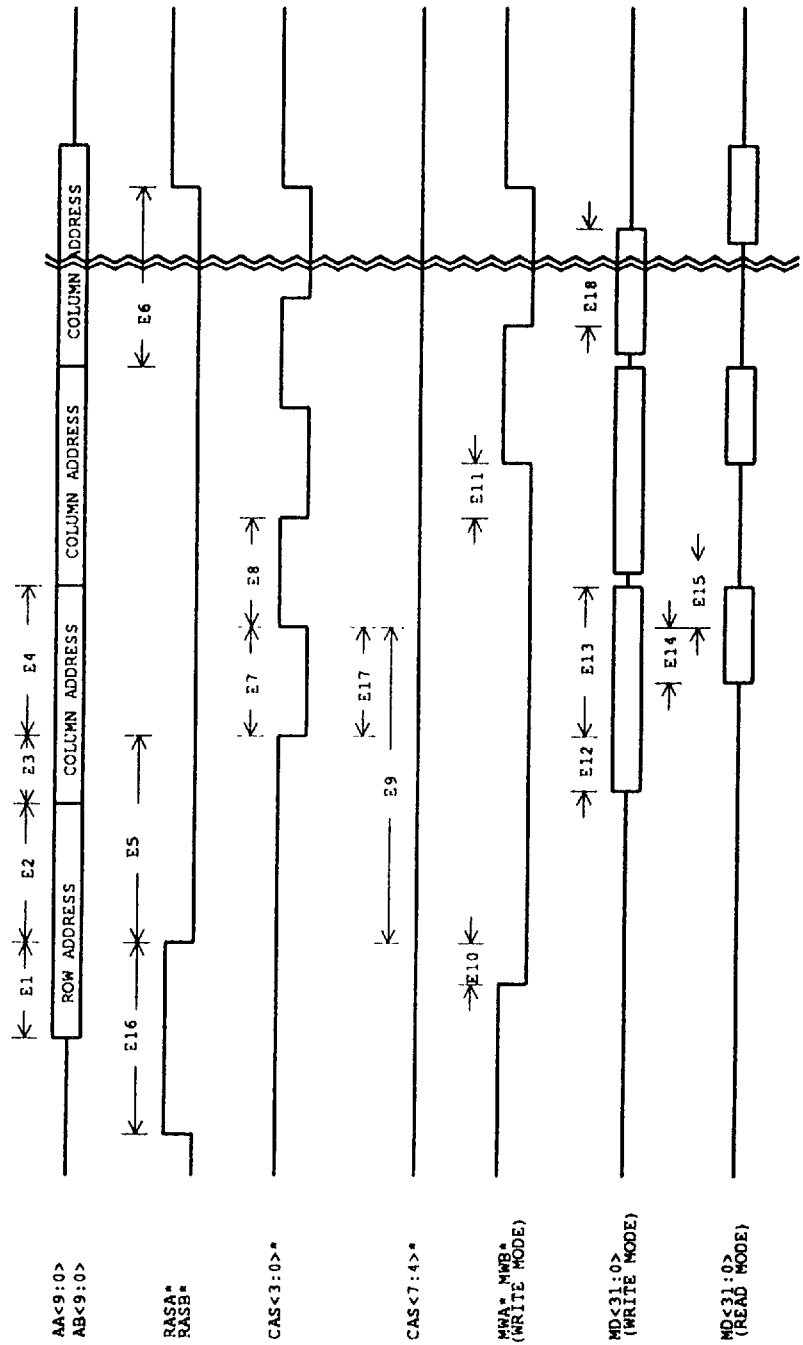


Figure 4.10-7 Display Memory Read/Write Timing - Non-interleaved

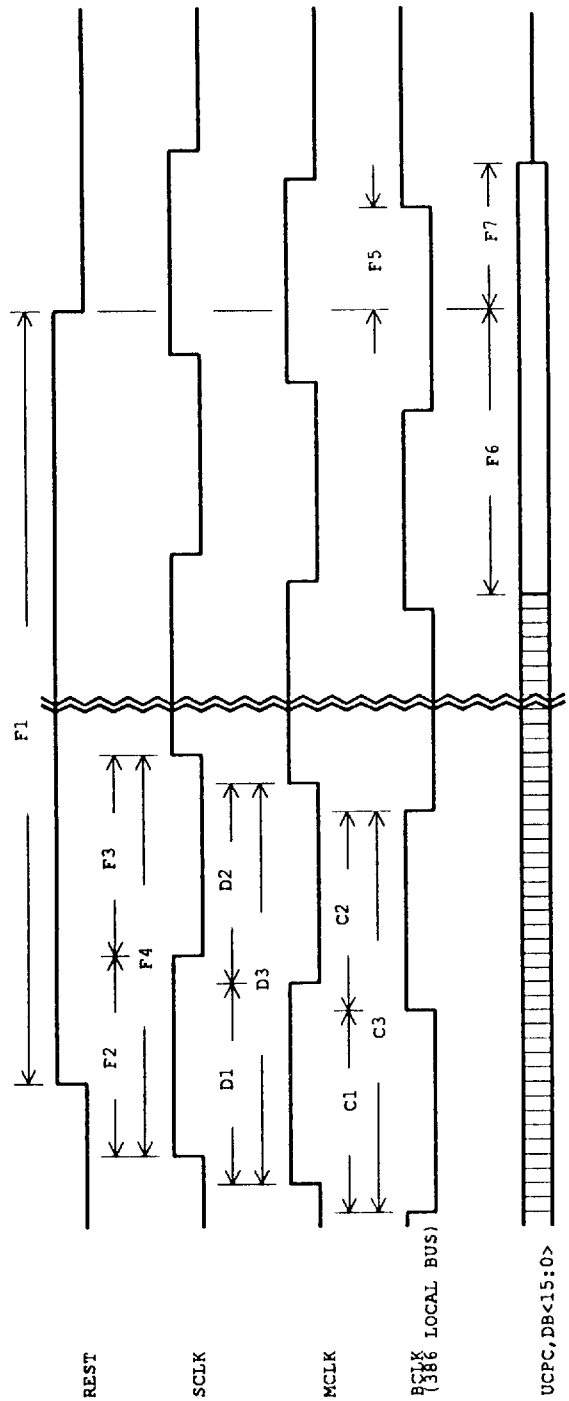


Figure 4.10-8 Reset Initialize and Clock Timing

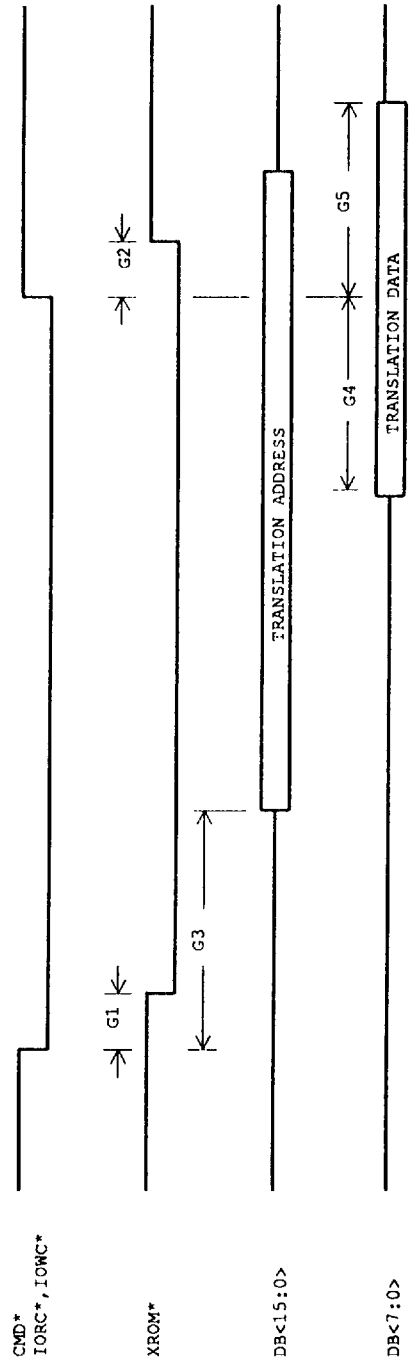


Figure 4.10-9 Translation ROM Read/Write Timing

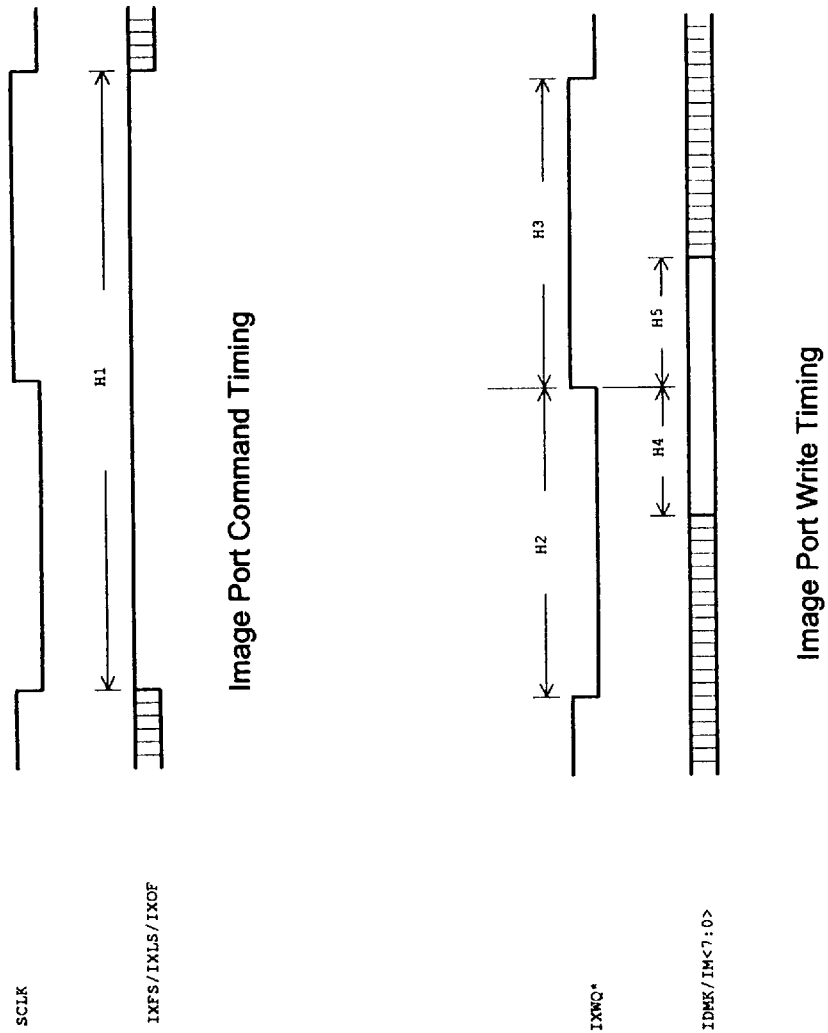


Figure 4.10-10 Image Port Command / Write Timing

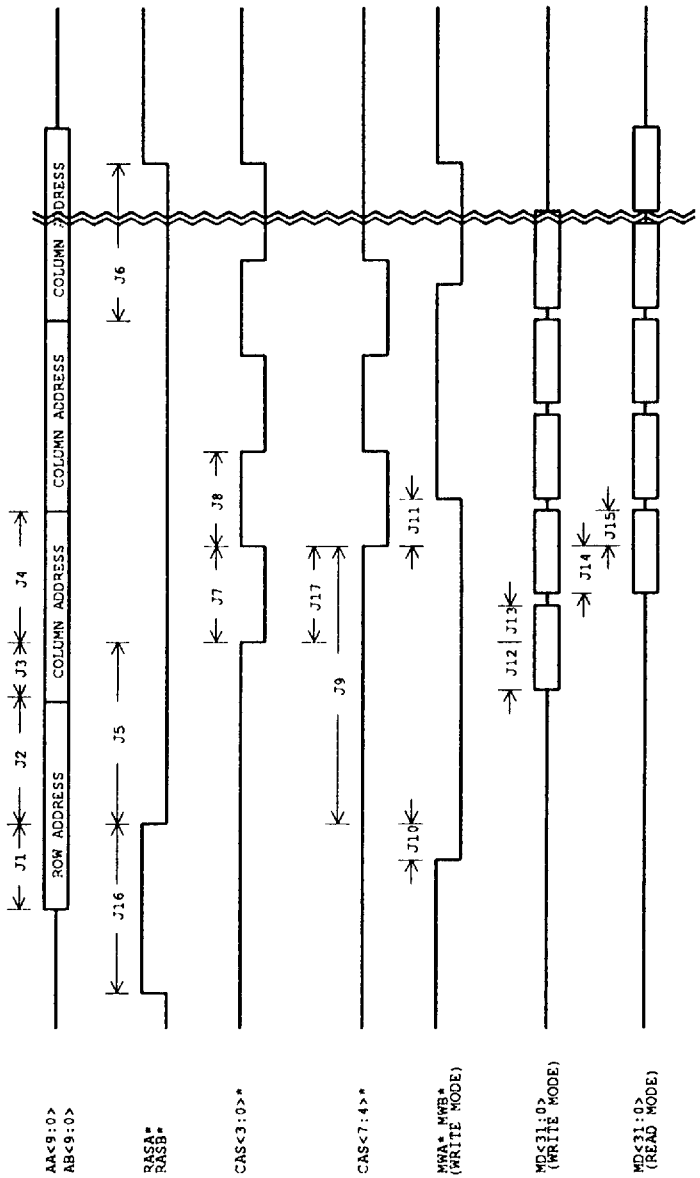


Figure 4.10-11 Display Memory Read/Write Timing - Interleaved

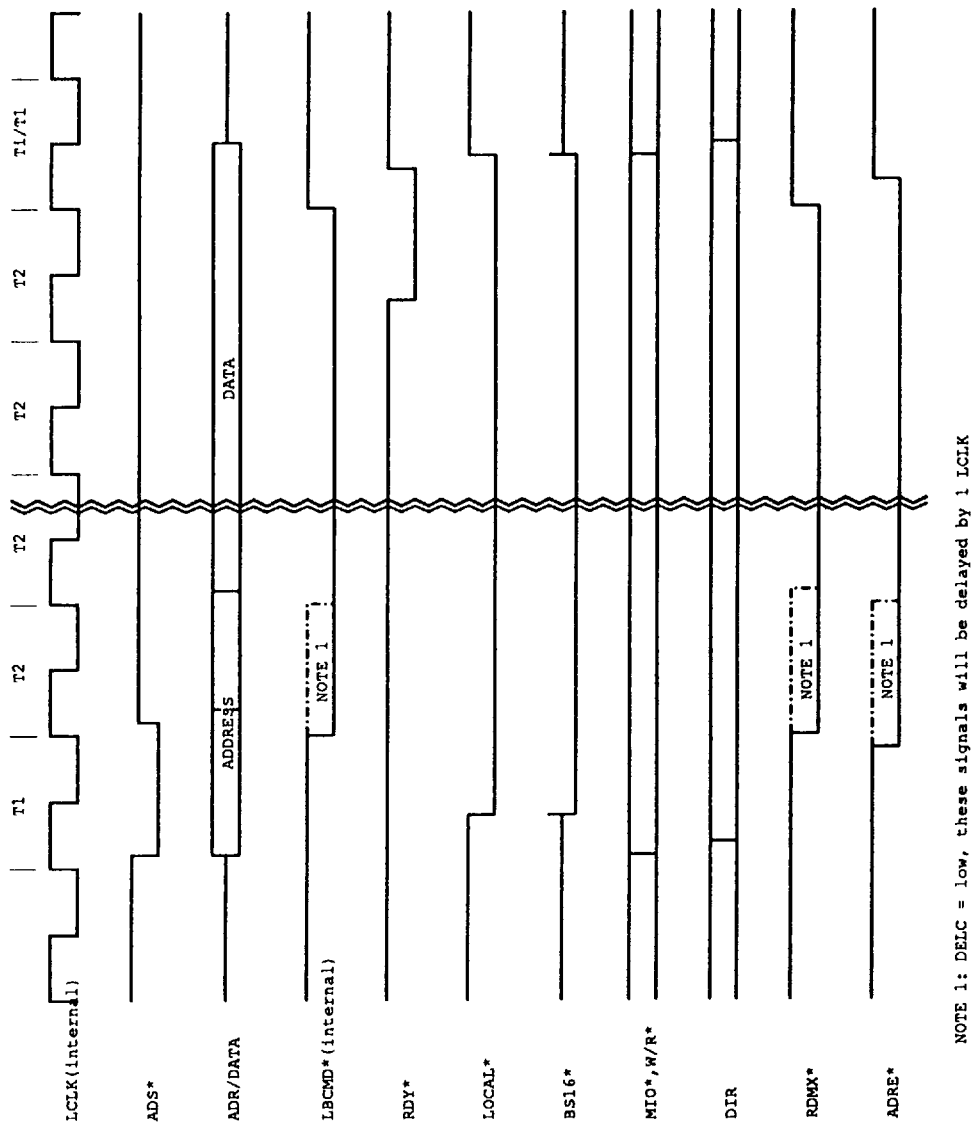


Figure 4.11-1 Normal Local Bus Cycle

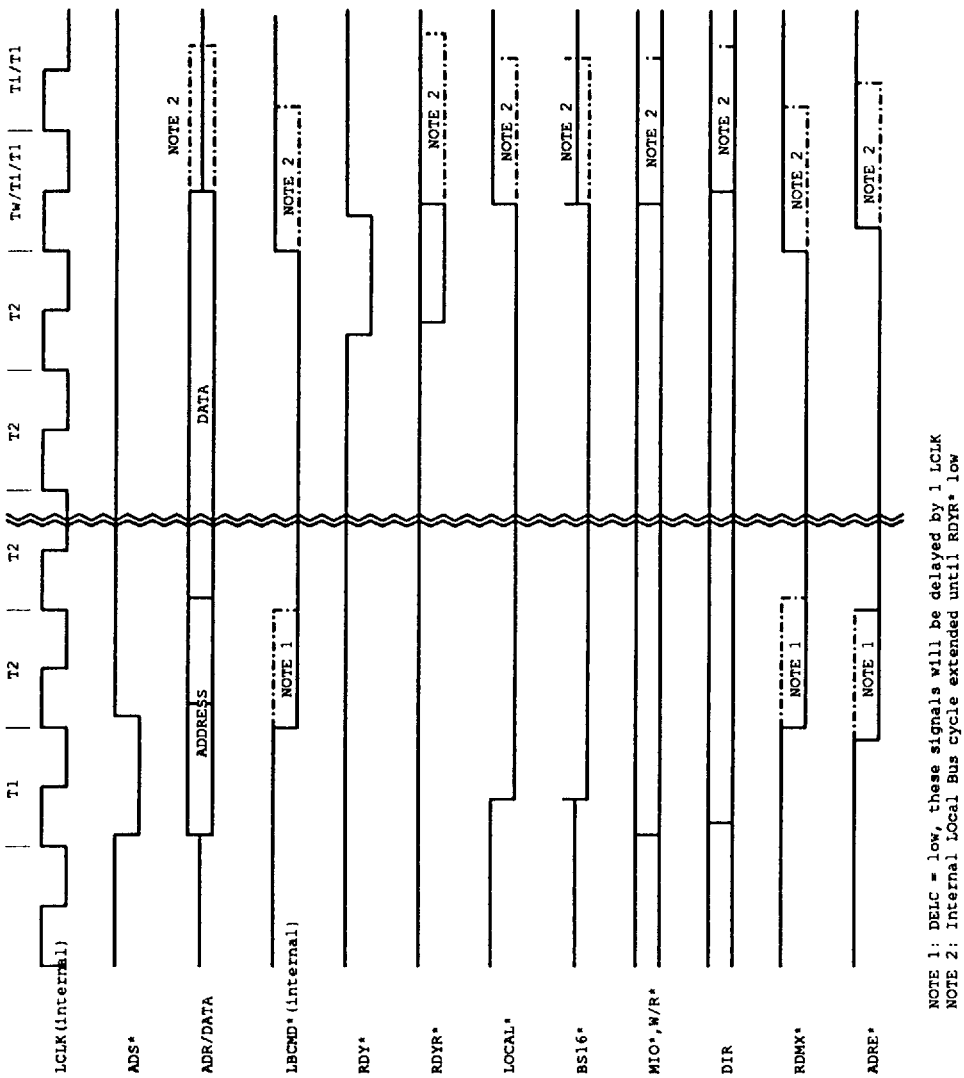


Figure 4.11-2 Local Bus Cycle Ready Return

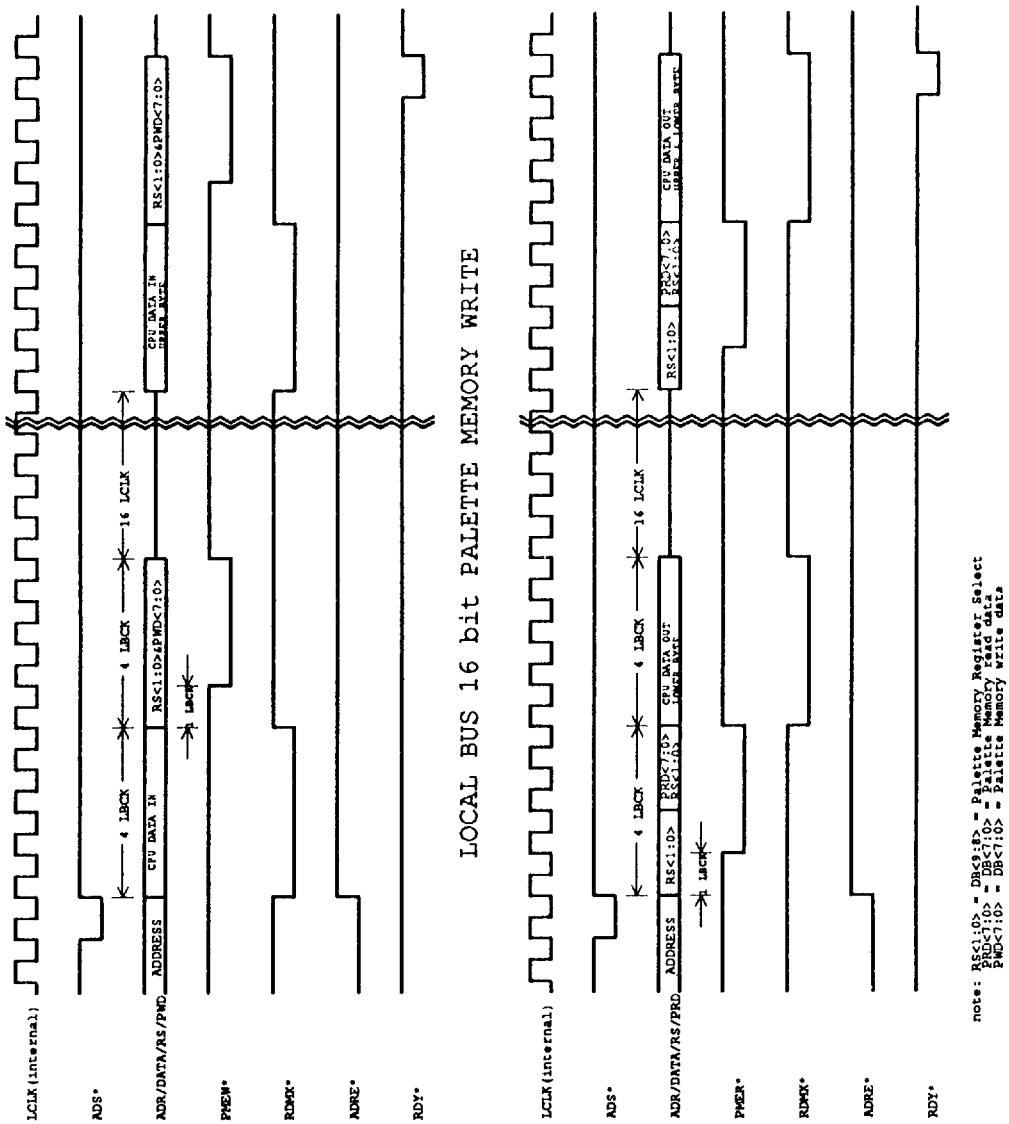


Figure 4.11-3 Local Bus 16-bit Palette Memory Read

5. ET4000/W32i Register Descriptions

Table 5.0-1 ET4000/W32i Registers, R/W Operation, Port Addresses, Size

Register	R/W Operation	I/O Port Address	Bits
General			
Misc. Output	W	03C2	<7:0>
	R	03CC	<7:0>
Input Status 0	R	03C2	<7:0>
Input Status 1	R	03#A	<7:0>
Feature Control	W	03#A	<7:0>
	R	03CA	<7:0>
Video Subsystem Enable	RW	03C3/46E8	<7:0>
External Palette RAM			
Pixel Mask	RW	003C6	<7:0>
Pixel Write Address	RW	003C8	<7:0>
Pixel Color Value	RW	003C9	<7:0>
Pixel Read Address	W	003C7	<7:0>
DAC State	R	003C7	<7:0>
Attribute Controller (ATC)			
Address/Index	W	03C0 (Index)	<7:0>
	R	03C0	<7:0>
Indexed registers	W	03C0 (Data)	<7:0>
	R	03C1	<7:0>
Primary CRT Controller (CRTC)			
Address/Index	RW	03#4	<7:0>
Indexed registers	RW	03#5	<7:0>
Timing Sequencer (TS)			
Address/Index	RW	03C4	<7:0>
Indexed registers	RW	03C5	<7:0>
Graphics Display Controller (GDC)			
Segment Select 1	RW	03CD	<7:0>
Segment Select 2	RW	03CB	<7:0>
Address/Index	RW	03CE	<7:0>
Indexed registers	RW	03CF	<7:0>
Secondary CRT Controller (CRTCB/Sprite)			
Address/Index	RW	21xA (Index)[1]	<7:0>
Indexed registers	RW	21xB (Data)[1]	<7:0>
Image Port (IMA)			
Address/Index	RW	21xA (Index)[1]	<7:0>
Indexed registers	RW	21xB (Data)[1]	<7:0>

= B in monochrome emulation modes; D in color emulation modes, controlled by bit 0 in the Miscellaneous Output Register.

[1] x = IOD<2:0>. See Section 3.3.1.1, IOD<2:0>.

Table 5.0-2 ET4000/W32i 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	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>
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>
Virtual Bus Size	RW	8E	<7:0>
X/Y Direction	RW	8F	<7:0>
Pattern Wrap	RW	90	<7:0>
Source Wrap	RW	92	<7:0>
X Position	RW	94	<11:0>
Y Position	RW	96	<11:0>
X Count	RW	98	<11:0>
Y Count	RW	9A	<11:0>
Routing Control	RW	9C	<7:0>
Reload Control	RW	9D	<7:0>
Background Raster Operation	RW	9E	<7:0>
Foreground Raster Operation	RW	9F	<7:0>
Destination Address	RW	A0	<21:0>

* See Section 2.11.3 on reading and writing accelerator registers.

5.1 General Registers

The ET4000/W32i 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 5.3.28, 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

The clock select bits CS<1:0> can be translated by the ET4000/W32i according to the following conditions:

NOTES: 1. EMCK = CRTC Index 34 bit 0 and ENXL = CRTC Index bit 5.
2. If CS<1:0> are used to select the external switch setting, care must be taken to ensure proper selection of the switch setting after the translation of CS<1:0>.

In VGA mode:

a. If EMCK bit is set to 0: CS<1:0> are equal to the programmed value.
b. If EMCK bit is set to 1: CS<1> is equal to the programmed value, and CS<0> is equal to inversion of programmed value.

In EGA mode:

a. if ENXL = 0 then:

programmed CS<1:0>;	output CS<1:0>
1 1	0 0
1 0	1 1
0 1	1 0
0 0	0 1

b. if ENXL = 1 then:
1. If EMCK bit is set to 0: CS<1:0> are equal to the programmed value.
2. If EMCK bit is set to 1: CS<1> is equal to the programmed value, and CS<0> is equal to inversion of programmed value.

5.1.1 Miscellaneous Output Register (cont'd)

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 DB<1:0> bus status at the last REST low-to-high transition. If the DB<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 complement.	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 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.

5.1.4 Feature Control Register

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

Bit	Description	Access
7	Reserved.	
6	Reserved.	
5:2	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 5:2	Used to read back the MONID<5:2> pins (MONID<3:2> for Local Bus) for monitor identification. See also Section 5.1.2, Input Status Register Zero, bit 4 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 below).

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, enables the second page of display memory, starting at B8000, providing 64KB of display memory. This bit can be read from bit 6 of the Display Mode Control Register (3#8). NOTE: Bits <1:0> are used to set the KEY. See Input Status Register Zero for instructions to set the KEY.

5.2 CRTC Register Description

The CPU interface to the ET4000/W32i 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.

All values are in hexadecimal unless otherwise noted.

Table 5.2-1 CRTC Index Register

Register Name		Port 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

= B in monochrome emulation modes; D in color emulation modes, controlled by bit 0 in the Miscellaneous Output 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".

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/W32i 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/W32i chip.

Because there are so many ET4000/W32i 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

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

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

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

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

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, indicates 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

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

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	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

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

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 the 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, enables the vertical interrupt to occur. If bit 5 is set to 0 and the vertical interrupt is cleared, then IRQ will be asserted when "VS" becomes true. When set to 1, vertical interrupts cannot occur.
Bit 4	When set to 0, clears the vertical interrupt. If bit 5 is low and the vertical interrupt is cleared, then output pin IRQ will be asserted when output line "VS" becomes true. 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. So for a 640-pixel (80-byte) wide graphics display, 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 2 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 -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 is equal to 1 and bit 5 of the Underline Row Address Register is also equal to 1, then the linear counter will increment twice per character.

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

Bit	Description
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: System Segment Map Comparator

I/O address = 3#5

Bit	Description	Access
7:5	Reserved.	
4:0	Addressing mode.	RW

Bit Description

Bits 4:0 These bits are compared to the host's upper address bus or decode and are used to select the video memory segments within the system memory map. The default value on power-up for bits 4:0 is

Bits
4 3 2 1 0
1 1 1 0 0

ISA bus bits 1:0 compare to inputs SEGE and A<22>, respectively. If System Linear Mode is disabled, this comparison is ANDed with inputs A<21> and A<20>, which must be low. If System Linear Mode is enabled, A<21> and A<20> are address inputs. Bits 4:2 are always ignored.

MCA bus bits 2:0 compare to inputs MADE24, A<23> and A<22>, respectively. If System Linear Mode is disabled, this comparison is ANDed with inputs A<21> and A<20>, which must be low. If System Linear Mode is enabled, A<21> and A<20> are address inputs. Bits 4:3 are always ignored.

Local Bus bits 5:0 compare to inputs SEG2, SEG1, SEG0, A<23>, and A<22>, respectively. If System Linear Mode is disabled, this comparison is ANDed with inputs A<21> and A<20>, which must be low. If System Linear Mode is enabled, A<21> and A<20> are address inputs.

When the Image Port is enabled, A<22:20> are ignored (see Section 5.7, IMA Register Descriptions, and 3.8, Image Port Interface). Consequently, maximum system linear space is 1MB.

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 3.4). 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+1)*SCLK, RAS pre-charge time.	RW
2	Reserved.	
1:0	CSW<1:0> ($1+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 W32i memory controller can perform a memory access every SCLK.

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

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.

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

CRTC 32

	Graphics	Text	
Mode CSW <1:0>	CAS<7:0> CAS Low Pulse Width	CAS<7:6;3:2> CAS Low Pulse Width	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 bit (<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	ENXL 1=enable translation ROM when writing CRTC/MISCOUT.	RW
4	ENXR 1=enable translation ROM when reading CRTC/MISCOUT.	RW
3	ENVS VSE register port address (1=46E8, 0=3C3).	RW
2	TRIS 1=tri-state the ET4000/W32i's output tri-state pins.	RW
1	CS2 MCLK clock select 2*	RW
0	EMCK 1=enable translation of CS0 bit.	RW

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.

Bits 5:4 (Bit 5 for write, bit 4 for read) when set to 1, disable the RDMEL output when an I/O read/write to the CRTC Data register 3#5 or MISCOUT register is performed. This allows the external translation ROM to be enabled for the CRTC register. To use the translation, external ROM must be incorporated.

When set to 0, the Translation Mode is 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 for pin numbers. RASB*, RASA*, CAS<3:0>*, MWB*, MWA*, MD<31:0>, AB<9:0>, AA<9:0>, VS, HS, AP<7:0>, PCLK, MBS*

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, is used to enable the ET4000/W32i to translate the clock select bits (CS1,CS0). (See Section 5.1.1, Miscellaneous Output Register.) Also, during ENXL set to 1, EMCK is used to select the external translation ROM's map while in 6845 mode (see Miscellaneous Output Register).

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.	RW
2	Vertical Display End Bit 10.	RW
1	Vertical Total Bit 10.	RW
0	Vertical Blank Start Bit 10.	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/W32i'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, I/O 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 I/O Pin Description 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.

5.2.34 CRTC Indexed Register 36: Video System Configuration 1

(VSCONF1) (protected by key)
I/O address = 3#5

Bit	Description	Access
7	16-bit I/O read/write.	RW
6	16-bit display memory read/write.	RW
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 7 When set to 1 in ISA, Micro Channel, and Local Bus implementations, will enable the 16-bit CPU I/O read/write data bus interface at the DB<15:0> input.

When set to 0 (power-up default condition) in ISA or Micro Channel implementations, will enable the 8-bit CPU I/O read/write data bus interface. For Local Bus implementations, the power-up default is 1 and should not be programmed to 0.

Bit 6 When set to 1 in ISA, Micro Channel, and Local Bus implementations, will enable the 16-bit CPU memory read/write data bus interface at the DB<15:0> input. This is the power-up default condition in local bus configurations.

When set to 0 in ISA or Micro Channel implementations, will enable the 8-bit CPU memory read/write data bus interface at the DB<15:0> input. For Local Bus implementations, 0 will enable the 32-bit CPU memory read/write data bus interface. This is the power-up default condition in ISA and Micro Channel configurations.

CPU Memory Data Bus Width

Bit 6	ISA/MCA	Local Bus
0	8	32
1	16	16

Bit 5 When set to 1(and bit 3 also set to 1), enables the memory mapped registers (MMU and Accelerator). See Sections 2.10 - 2.11 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.

5.2.34 CRTC Indexed Register 36: Video System Configuration 1 (cont'd)

Bit	Description
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 2.10 - 2.11 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.

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	16-bit ROM enable.	RW
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/W32i 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 4 When set to 1, enables the 16-bit ROM configuration for ISA, Micro Channel, and Local Bus implementations (power-up default for Local Bus). When set to 0 (default power-up condition), enables the 8-bit ROM configuration for ISA and Micro Channel implementations, and 32-bit for Local Bus. The following table illustrates the configurations:

	16-bit ISA/MCA Bus	32-bit Local Bus
32-bit enable	-	0
16-bit enable	1	1
8-bit enable	0	-
Power-up	0	1

NOTE: If 16-bit ROM is selected with the ISA bus, then the ROM size should be set to 32K. Bits 3 & 5 of TS Indexed Register 7: TS Auxiliary Mode must be set to 1,1 so that the ROM size is set to 32K.

5.2.35 CRTC Indexed Register 37: Video System Configuration 2 (cont'd)

Bit **Description**

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>

When set to 0, pin 81 is redefined as AB<9>, and pin 75 is redefined as AA<9>. See section 3.6 Display Memory Interface pin descriptions for more information. See also section 3.3.1.2 ISA Bus Interface, XR16, and 3.7.1 Hardware Sprite, BDE, for additional redefinitions effected by the setting of this bit.

Bit 2 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/W32i utilizes the interleave capability and uses the 4 CAS* and 2 WRITE ENABLE* signal configuration exclusively. Reference sample schmematics for memory design methods. The W32i 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/W32i 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

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

Table 5.3-2 TS Indexed Registers

TS Indexed Register Name	TS Indexed Address		Port 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
7:3	Reserved.	
2:1	Timing sequencer state (bits 1 & 2).	RW
0	Reserved.	

Bit Description

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).	RW
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/W32i 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 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/W32i 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 3F) for CPU read operations.

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

Note: This register is reset to 0 when a synchronous reset is done (by setting TS Indexed Register 1, 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 CPUdata 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 one, 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.

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

Bit **Description**

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.

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, as is the selected logical function. The set/reset function is also ignored.

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/W32i 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. A value of 1
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 registers should be programmed to have contents the same as the indexed address to "pass through" the internal display data.

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.

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		Bit	
5	4	5	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	Reserved.	
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).

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

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

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.

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 2.2 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 <11:8> 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 2.2 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

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

I/O address = 21xB

Bit	Description	Access
7:4	Revision ID.	RO
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
0 0 1 1	W32i Rev. B
↓ ↓ ↓ ↓	
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

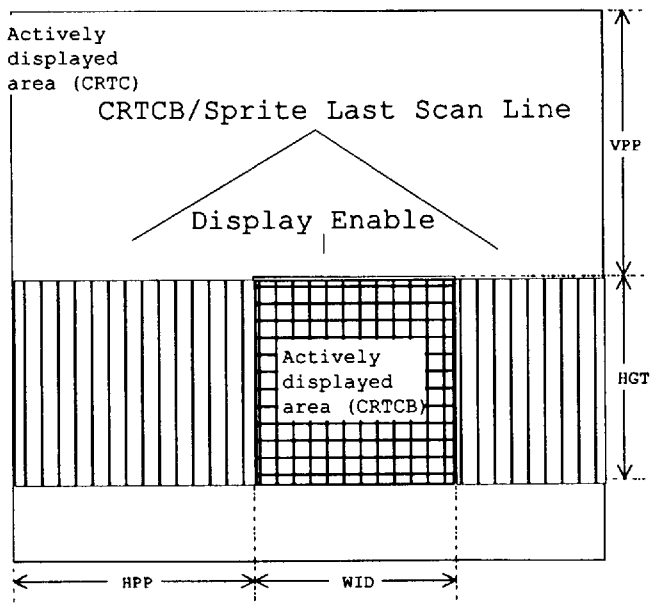
Bit	Description	Access
7	CRTCB/Sprite last scan line.	RO
6	Display enable.	RO
5:3	Reserved.	
2:0	CRTCB pixel panning.	RW

Bit **Description**

Bit 7 When high indicates the last scan line has been displayed for the CRTCB/Sprite.

Bit 6 This bit reflects real time status of the BDE pin (See Section 3.7.2) for CRTCB/Sprite.

Bits 2:0 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.	
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	Reserved.	
6:5	Bus select.	RO
4:3	Reserved.	
2	Sprite size control.	RW
1	B pixel overlay/B pixel data output.	RW
0	CRTCB/Sprite select.	RW

Bit Description

Bits 6:5 These bits are the PORI value of UCPC, and AEN* pins, respectively (see Section 3.2) and determine bus type.

Bits 6 5	Bus
0 0	ISA
0 1	ISA
1 0	MCA
1 1	Local Bus

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 3.7.2.

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	Token outputs/External Sprite enable.	RW
5:2	Reserved.	
1	Interlace Image Port address.	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. This bit, in combination with CRTC Indexed Register 37 <3> redefines the BDE output. This is illustrated as follows:

CRTC I37 <3>	IMA F7 <7>	BDE
1	1	BDE
1	0	CS4
0	x	AB<9>

At reset, this bit has a value of 0. See section 3, Pin Descriptions (BDE, XR16, CS<4>, AA<9:0>, AB<9:0> for more information.

Bit 6 When set to 1, enables the token status outputs or external sprite on pins 98 and 18.

When set to 0, token status bits or external sprite are disabled, and enables pins 98 and 18 for either SYNR and SWSE signals, or IDMK and IXWQ* signals depending on bit 0 of this register.

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

When set to 0, enables linear interlace transfer. See Section 2.2.7 for additional information. At reset, this bit has a value of 0.

Bit 0 When set to 1, enables the Image Port. (See Section 3.8 Image Port Interface for shared pin information.) This bit is valid only if IMA Indexed Register F7 <6> = 0.

When set to 0, disables the Image Port, meaning that all the inputs to the Image Port are ignored and the pins assume their alternate functions. 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 MMU 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
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.

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.

5.8.3 MMU Memory Base Pointer Register 2

Memory offset = 08

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.

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 MMU registers. The offset in the table below is added to the base address to calculate the actual address of the register.

Memory Offset	Register
	Non-Queued Registers
30	ACL Suspend/Terminate Register
31	ACL Operation State Register
32	ACL Sync Enable Register
34	ACL Interrupt Mask Register
35	ACL Interrupt Status Register
36	ACL Accelerator Status 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 Virtual Bus Size Register
8F	ACL X/Y Direction Register
90	ACL Pattern Wrap Register
92	ACL Source Wrap Register
94	ACL X Position Register
96	ACL Y Position Register
98	ACL X Count Register
9A	ACL Y Count Register
9C	ACL Routing Control Register
9D	ACL Reload Control Register
9E	ACL Background Raster Operation Register
9F	ACL Foreground Raster Operation Register
A0	ACL Destination Address Register

5.9.1 ACL Suspend/Terminate Register

This is a non-queued register
Memory offset = 30

Bit	Description	Access
7:4	Reserved.	
4	Terminate Accelerator Operation (TO).	RW
3:1	Reserved.	
0	Suspend Accelerator Operation (SO).	RW

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 0) 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 0	Used to suspend an Accelerator operation. To suspend an Accelerator operation, the programmer should write a 1 to this bit, wait for RDST (ACL Status Register, bit 0) 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
7:4	Reserved.	
3	Resume Accelerator Operation (RMO).	WO
2:1	Reserved.	
0	Restore Accelerator Operation State (RSO).	WO

Bit	Description
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 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 (or initiate) an accelerator operation.

5.9.3 ACL Sync Enable Register

This is a non-queued register
Memory offset = 32

Bit	Description	Access
7:2	Reserved.	
0	Sync Enable.	RW

Bit	Description
Bit 0	When set to 1, indicates that a write to a full queue will be "wait-stated" until the queue becomes not full. When set to 0, indicates that a write to a full queue will be ignored. It is possible to generate an interrupt when such a write is ignored, see Section 2.11.7.

5.9.4 ACL Interrupt Mask Register

This is a non-queued register
Memory offset = 34

Bit	Description	Access
7:3	Reserved.	
2	Write Fault Interrupt Enable.	RW
1	Read Interrupt Enable.	RW
0	Write Interrupt Enable.	RW

Bit	Description
Bit 2	When set to 1, enables a Write Fault Interrupt when a write to a full queue occurs (and ACL Sync Enable Register, bit 0 is 0). 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 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.
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.

5.9.5 ACL Interrupt Status Register

This is a non-queued register
Memory offset = 35

Bit	Description	Access
7:3	Reserved.	
2	Write Fault Interrupt Status.	RW
1	Read Interrupt Status.	RW
0	Write Interrupt Status.	RO

Bit	Description
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 5.9.4).

5.9.6 ACL Accelerator Status Register

This is a non-queued register
Memory offset = 36

Bit	Description	Access
7:4	Reserved.	
3	Screen-to-Screen Status (SSO).	RO
2	XY Status (XYST).	RW
1	Read Status (RDST).	RO
0	Write Status (WRST).	RO

Bit	Description
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.

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.

5.9.7 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.

5.9.8 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. If the host is providing data for the Source Map, the value in this register is not used by the accelerator.

5.9.9 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 pixels wide, a value of 7 should be programmed into this register.

5.9.10 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.

5.9.11 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.

5.9.12 ACL Virtual Bus Size Register

This is a queued register
Memory offset = 8E

Bit	Description	Access
7:2	Reserved.	
1:0	Virtual Bus Size (VBS).	RW

Bit Description

Bits 1:0 The Virtual Bus Size is only enforced when the host is providing Source Map data or Mix Map data to an accelerated operation. It is encoded as follows:

VBS	
00	1-byte
01	2-byte
10	4-byte
11	Reserved

The Host Interface of the ET4000/W32i waits for this many bytes and then releases the data to the accelerator. The Virtual Bus Size also controls the amount addresses are to be incremented for each host data transfer to the accelerator. The increment value also depends on the ADRO and DARO values (see Section 5.9.20, ACL Routing Control Register):

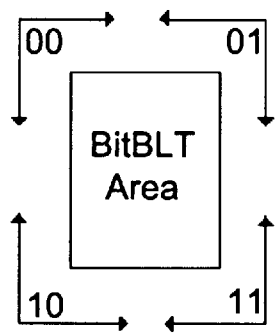
VBS	ADRO	DARO	Increment
xx	00	000	N/A (No CPU data transfer)
xx	01	000	1-byte
00	xx	001	1-byte
01	xx	001	2-bytes
10	xx	001	4-bytes
00	xx	010	8-bytes
01	xx	010	16-bytes
10	xx	010	32-bytes
11	xx	xxx	Reserved

5.9.13 ACL X/Y Direction Register

This is a queued register
Memory offset = 8F

Bit	Description	Access
7:2	Reserved.	
1	Y Direction.	RW
0	X Direction.	RW

Bit	Description
Bits 1:0	These bits indicate in which direction the accelerated operation will proceed. When set to 0, the Accelerator operates from the lowest address to highest address (increasing direction). When set to 1, the Accelerator operates from highest address to lowest address (decreasing direction). The figure below summarizes the effect of various programmed values:



5.9.14 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. After the Accelerator operates on the wrap-length number of bytes (horizontally) or lines (vertically), the Pattern pointer is set back wrap-length number of bytes or lines. 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

5.9.15 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	See Section 5.9.14, ACL Pattern Wrap Register for an explanation of this register.
Bits 2:0	

5.9.16 ACL X Position Register

This is a queued register
Memory offset = 94

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 the value written into this register and approaching the terminal value as programmed in the ACL X Count Register. Writing to this register will load the X Position Register in the queue, but will not affect the accelerator's internal copy of this register; thus if the host attempts to read back a value just written, the values will not match. 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. They should normally be initialized to zero at power-up, the software should ensure that they are zero when any accelerated operation is initiated.

5.9.17 ACL Y Position Register

This is a queued register
Memory offset = 96

Bit	Description	Access
15:12	Reserved.	
11:0	Y Position.	RW

Bit	Description
Bits 11:0	See Section 5.9.16, ACL X Position Register for an explanation of this register.

5.9.18 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 X Count should be programmed to one less than the desired number of bytes to be operated on.

5.9.19 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.

5.9.20 ACL Routing Control Register

This is a queued register
Memory offset = 9C

Bit	Description	Access
7	Invalidate disable.	RW
6	Reserved.	
5:4	Routing of CPU address (ADRO).	RW
3	Reserved.	
2:0	Routing of CPU data (DARO).	RW

Bit **Description**

Bit 7 When set to 1, multiport cache is not invalidated at the end of the accelerator operation.

When set to 0, multiport cache is invalidated at the end of the accelerator operation. It is recommended that this bit always be programmed to 0.

Bits 5:4 Routing of CPU address;

ADRO	
00	CPU address not used
01	CPU address is Destination address
10	Reserved
11	Reserved

"CPU address not used" means only the first write to an accelerated MMU aperture is used to determine the destination address. Then, as the accelerated operation progresses, the destination pointer is updated automatically.

Bits 2:0 Routing of CPU data:

DARO	
000	CPU data not used
001	CPU data is Source data
010	CPU data is Mix Data
011	Reserved
100	CPU data is X Count
101	CPU data is Y Count
11x	Reserved

When the CPU data is X Count or Y Count, the initial write to start the accelerator operation stores the low-order 8 bits of the corresponding internal X Count or Y Count register (the high-order bits come from the X Count or Y Count in the queue as normal). Note that the Virtual Bus size must be 1 byte in this case.

5.9.21 ACL Reload Control Register

This is a queued register
Memory offset = 9D

Bit	Description	Access
7:2	Reserved.	
1	Enable Reload of Pattern Address.	RW
0	Enable Reload of Source Address.	RW

Bit **Description**

Bit 1 When set to 1, the accelerator's Internal Pattern Address value (resulting from the previous accelerator operation) will be the starting Pattern address for the next accelerator operation.

When set to 0, the programmed ACL Pattern Address value will be used as the starting Pattern address.

Bit 0 When set to 1, the accelerator's Internal Source Address value (resulting from the previous accelerator operation) will be the starting Source address for the next accelerator operation.

When set to 0, the programmed ACL Source Address value will be used as the starting Source address.

NOTE: Undefined results occur if either of these bits is set to 1 when the **very first** accelerator operation is initiated, since the accelerator's internal address pointers are not initialized.

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 Maps used when CPU data routing is Mix Data, and the Mix Data bit is a 0 (see Section 5.9.20, ACL Routing Control Register). See also Appendix A, ET4000/W32i (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 Maps used when CPU data routing is Mix Data, and the Mix data bit is a 1, or else when the CPU is not providing Mix Map data (see ACL Routing Control Register). See also Appendix A, ET4000/W32i (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. There are two methods of loading this register, explicit and implicit. An explicit load is accomplished by simply writing to this memory address, similar to any other ACL queued register. An implicit load occurs when a write is performed through an accelerated MMU aperture. When such a write is performed, the address after MMU translation (which is an absolute address into display memory) is loaded into this register.

6. Board-Level Design Considerations

6.1 Typical Board-level Configuration

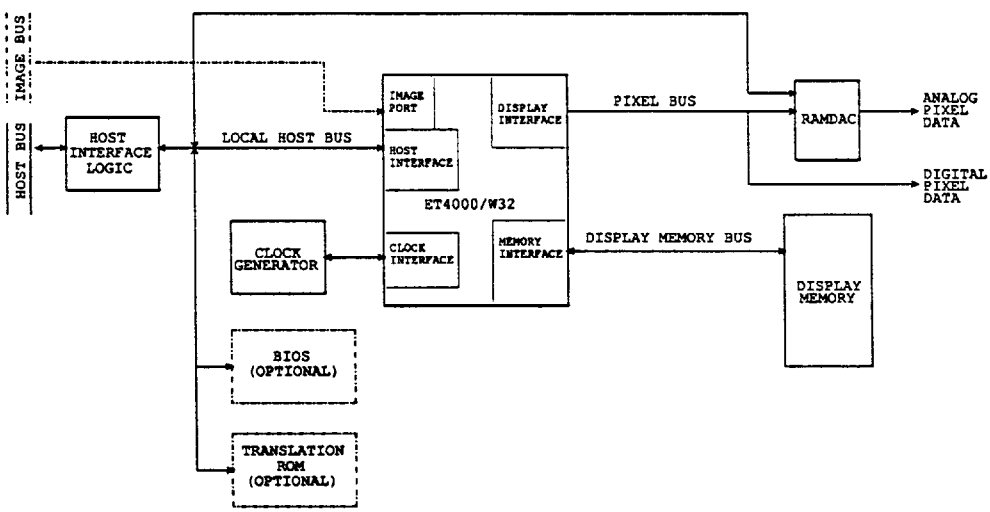
A complete ET4000/W32i VGA design can use as few as 9 chips configured with only two 1MB DRAMs. The ET4000/W32i requires no additional support chips to interface to the ISA and MCA buses. The chip delivers resolutions up to 1024x768 with 65,536 colors.

The ET4000/W32i can be easily programmed and configured to function as a VGA/EGA controller with only 512KB of DRAM.

As shown in Figure 6.1-1, a basic configuration can be interfaced with a host bus, including local bus, and an optional image bus. The configuration consists of a memory bank as display buffer, the ET4000/W32i VLSI chip, an external color look-up DAC (Digital to Analog Converter), and data/address buffers/multiplexers as support logic.

Configurations can be programmed, via a single register, to be 100% register-level compatible with a VGA or EGA controller. All of the VGA/EGA text and graphic modes are supported, and all of the VGA/EGA hardware assist features including data latching, bit mask, rotation, logical functions, and plane-selects are provided at a register-compatible level.

Figure 6.1-1 Board Level Block Diagram



6.2 Host Interface Design Considerations

Address buffers are needed to multiplex the host processor's lower 16- bit address, A<15:0>, so that they can be time-multiplexed over the 16-bit address/data bus (DB<15:0>) into the ET4000/W32i chip. The address input enable signal is generated by the ET4000/W32i to enable the 16-bit addresses onto ET4000/W32i's DB<15:0> bi-directional data bus at the beginning of each memory or I/O operation.

Two bi-directional bus transceivers (LS245) are required to interface the upper DB<15:8> and lower DB<7:0> data bus from the host processor. Three separate control signals, RDMH*, RDML*, and DIR, are provided to enable the transceivers and determine their direction, respectively.

6.2.1 ISA Bus

The ET4000/W32i interfaces to both 8/16-bit or 16-bit-only ISA buses. The 8 or 16-bit data bus width for memory, I/O, or BIOS can be controlled independently by programming CRTC Indexed Registers 36 and 37.

6.2.1.1 ISA Bus PORI

The following inputs provide the general ISA bus configuration:

UCPC	= low	Selects ISA bus
IOD<2:0>	= x	CRTCB/Sprite I/O register map for I/O addresses 21xA, 21xB where x = 0,1,2..,7
SYSW	= high	Disable System Linear Wiring
	= low	Enable System Linear Wiring

6.2.1.2 ISA Bus Special Note

Table 6.2.1.2-1 A<22:17> Configuration

SYSW = 0
IMAE = 0

ET4000/W32i	ISA Bus (16-bit slot only)
A<22>	LA<22>
A<21>	LA<21>
A<20>	LA<20>
A<19>	LA<19>
A<18>	LA<18>
A<17>	LA<17>

NOTE: LA<23> connects directly to ET4000/W32i SEGE input.

SYSW = 1
IMAE = 1 (16-bit slot only)

ET4000/W32i	ISA Bus	IMA Port
A<22>	Not used	IXFS
A<21>	Not used	IXLS
A<20>	Not used	IXOF
A<19>	LA<19>	Not used
A<18>	LA<18>	Not used
A<17>	LA<17>	Not used

NOTE: LA<23:20> must be decoded into ET4000/W32i SEGE input.

Data and Address Bus Multiplex

ADRE*, RDML*, RDMH*, and DIR provide control to the external data/address multiplexer as well as the direction of data flow.

16-Bit Only—System Linear Address Mode

In this configuration, LA<22:17> should be connected to the A<22:17> pins, and LA<23> must be connected to the SEGE pin. MEMW and MEMR from the 36-pin AT connector must be used instead of SMEMW and SMEMR from the 62-pin PC connector. The SYSW PORI bit must be low in this configuration.

16-Bit Only with Image Port Enabled

In this configuration, LA<19:17> must be connected to A<19:17> pins. A<22:20> inputs are redefined for the Image Port, and LA<31:20> must be externally decoded as segment enable input.

16-Bit Only with Image Port Enabled / System Linear Address Mode

Same as 16-bit Only with Image Port Enabled except that LA<23:21> are externally decoded to the desired 1M bytes of memory (see section 7.3 Video Memory Map) and that MEMW and MEMR from the 36-pin AT connector must be used instead of SMEMW and SMEMR from the 62-pin PC connector.

Early MCS16 Decode

When CRTC Indexed Register 36 <6> = 1, the W32i is enabled for 16-bit video memory transfers on the ISA bus. If system linear mode is disabled (CRTC Indexed Register 36, bit 4 = 0) MCS16 is decoded for the address space A0000-BFFFF. If system linear mode is enabled (CRTC Indexed Register 36, bit 4 = 1) MCS16 is decoded when SEGE and A<22> inputs are equal to CRTC Indexed Register 30, bits 1:0, respectively.

When CRTC Indexed Register 37<4> = 1, the W32i is enabled for 16-bit video BIOS transfers on the ISA bus. Note that EBIO pin (see section 3.3.1.2) should be low so that the W32i decodes A<23:15> into 0C0000-0C7FFF. If the EBIO pin is high the W32i decodes A<23:17> into 0C0000-0DFFFF for both MCS16 and data transfers.

Zero Wait State

During display memory write operations, if internal cache memory is available, SRDY* becomes active low during the high to low transition of MWTC* input to indicate a zero wait state access cycle.

Segment Address Comparator

CRTC Indexed Register 30 bits 1:0 are compared with SEGE and A<22> inputs to allow memory access. If System Linear Mode is disabled, the A<21> and A<20> inputs must also be low to allow memory access. If System Linear Mode is enabled, A<21> and A<20> are address inputs. If the Image Port is enabled, A<22:20> are ignored for address or comparison purposes.

6.2.2 Micro Channel Bus

ET4000/W32i interfaces to both 8/16-bit or 16-bit only Micro Channel buses. Eight or 16-bit data bus widths for memory, I/O, or BIOS can be controlled independently by programing CRTC Indexed Register 36.

6.2.2.1 Micro Channel Bus PORI

The following inputs provide the general Micro Channel bus configuration:

UCPC and AEN*	= high = low	Selects Micro Channel bus, AEN* must stay low at all times
IOD<2:0>	= x	CRTCB/Sprite I/O register map for I/O addresses 21xA, 21xB where x = 0,1,2..,7
PID<7:0>		POS register ID: If PID<7:0>=00h-FEh then POS 100 = 00h-FEh POS 101 = 80h If PID<7:0>=FFh then POS 100 = FFh POS 101 = FFh

6.2.2.2 Micro Channel Bus Special Note

Table 6.2.2.2-1 A<22:20> Configuration

IMAE = 0

ET4000/W32i	MCA	IMA Port
A<22>	A<22>	Not used
A<21>	A<21>	Not used
A<20>	A<20>	Not used

IMAE = 1

ET4000/W32i	MCA	IMA Port
A<22>	Not used	IXFS
A<21>	Not used	IXLS
A<20>	Not used	IXOF

NOTE: Host A<23:20> must be decoded and connected to the ET4000/W32i's A<23> input when IMAE = 1. ET4000/W32i A<22:20> must be held low by the external interface until IMAE is set to 1.

Data and Address Bus Multiplex

ADRE*, RDML*, RDMH*, and DIR provide control to the external data address multiplexer as well as the direction of data flow. Both SFDBK* and DS16* are generated by the ET4000/W32i and are internally latched during high to low transition of the CMD* input and remain latched until the low to high transition of CMD*.

Segment Address Comparator

CRTC Indexed Register 30 bits 2:0 are compared with MADE24, A<23>, and A<22> inputs to allow memory access. If System Linear Mode is disabled, the A<21> and A<20> inputs must also be low to allow memory access. If System Linear Mode is enabled, A<21> and A<20> are address inputs. If the Image Port is enabled A<22:20> are ignored for address or comparison purposes.

Pull-up Requirement

DS16* and CHRDY* must be externally pulled up with a 330 ohm resistor.

6.2.3 Local Bus

ET4000/W32i interfaces to both DX or SX Local buses in 16 or 32-bit configurations. To provide backward compatibility with the ET4000/AX Rev. G VGA chip, the 16-bit DX bus is also supported. The ET4000/W32i provides flexible configurations to support different microprocessor clock rates.

6.2.3.1 Local Bus PORI

The following inputs provide the general Local Bus configuration:

UCPC and AEN*	= high = high	Selects Local Bus, AEN* must stay high at all times.
SNPE	= low = high	External RAMDAC snoop enabled 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
WAT<1:0>		These 2 inputs determine the minimum internal command pulse width. The actual internal command pulse width depends on the present internal state of the ET4000/W32i.

The following is a table to determine minimum command pulse width for I/O or memory operations.

MIO*	WAT<1:0>	LCLKs (T2 states)
1	00	4
1	01	3
1	10	2
1	11	1
0	00	4
0	01	3
0	10	4
0	11	3

If external palette RAMDAC is enabled, the I/O instructions to the RAMDAC I/O ports result in a minimum of 8 LCLK cycles per command.

SX/DX selection:
Low = DX
High = SX

SX mode selects A<1>, BHE*, BLE* as the lower 4 bytes of address.
DX mode selects BE3*, BE2*, BE1*, and BE0* as the lower 4 bytes of address.

DVCK Divided clock:
Low LCLK = BCLK
High LCLK = BCLK/2

DELC Command delay. When high, ET4000/W32i internal command starts at first T2 of the CPU state.

When low, ET4000/W32i internal command starts at first T2 after ADS* goes from low to high.

BU32 This input, when ANDed with SXDX = low, configures the ET4000/W32i in the 32-bit CPU data bus configuration.

RDYS This input, when high, selects the RDY* output to be totem pole. When low, selects the RDY* to be tri-state drive.

DISB This input, when high, disables BIOS decode. When low, enables BIOS decode. This BIOS disable overrides TS Indexed Register 7 (See Section 5.3.9).

6.2.3.2 Local Bus Special Note

See Table 6.2.2.2-1

Clock Synchronization

All ET4000/W32i 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/W32i clocks are in phase.

Wait-State Considerations

Special care must be taken while WAT<1:0> is set to 1,1 for Local Bus Zero Wait State operation. In this mode, minimum LCLK low pulse widths must be 18ns and minimum LCLK high pulse widths must be 10ns to ensure proper setup time for RDY*.

When the ET4000/W32i is configured for zero wait-state operation (See section 3.3.3.1, Local Bus PORI), it will generate the RDY* output based on **either** the rising or falling edge of the LCLK. This fact places an additional timing constraint on the duty cycle of the BCLK. It is important that the BCLK signal fed to the ET4000/W32i have minimum skew with the system BCLK.

During a zero-wait state cycle the RDY* signal is generated by the falling edge of LCLK during the first T2 and is sampled by the system on the next rising edge of LCLK. Therefore, the minimum low pulse width of LCLK must be long enough to guarantee that the maximum delay of RDY*, plus any system delays on RDY*, plus the setup time to the CPU, can be met.

If the ET4000/W32i is **not** configured for zero wait-state, the RDY* output will only change state as a result of the rising edge of LCLK.

Local Bus Cycle

All ET4000/W32i internal bus cycles are initiated by ADS* low while LCLK changes from low to high. The ET4000/W32i decodes address and SEGn inputs, if an appropriate address is decoded, ET4000/W32i will assert the LOCAL* signal and an internal Local Bus Command (LBCMD) will be generated. The READY* signal is asserted by the ET4000/W32i at the last LCLK cycle of LBCMD. If external ready re-synchronization is required as in VESA LBUS, neither the ET4000/W32i nor the processor will terminate the command cycle until RDYR* is asserted by an external device.

16-Bit Local Bus

ET4000/W32i supports 16-bit buses. On the 486 bus, an external multiplexer is required for the upper 2 bytes of the CPU data bus and RDMX* must be externally decoded into RDML* and RDMH* depending on the state of BE3*, BE2*, BE1*, BE0*. BS16* is always generated in this configuration.

32-bit Local Bus

ET4000/W32i supports 32-bit DX buses. In this configuration, neither a 16-bit pixel bus nor the Image Port are supported. BS16* is generated during I/O cycles and BIOS cycles if CRTC Indexed Register 37 has 16-bit ROM enabled.

External Palette Memory DAC

The ET4000/W32i 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/W32i 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/W32i on DB<9:8>. Palette RAMDAC read or write data are interfaced via DB<7:0>.

Write Cycle: During the first 8-bit RAMDAC write cycle, the lower 8 bits of CPU write data are latched internally by the ET4000/W32i in the first half of this cycle. The ET4000/W32i then drives the DB<7:0> data bus with the latched data in the second half of this cycle. The ET4000/W32i goes idle for 16 LCLKs and repeats the previous operation with the upper 8-bit CPU data. The ET4000/W32i asserts RDY* low at the end of the second cycle. RS<1:0> will increment by 1 during the second cycle.

Read Cycle: During the first 8-bit RAMDAC read cycle, 8 bits of RAMDAC read data are latched by the ET4000/W32i in the first half of this cycle. The ET4000/W32i then drives this latched data with RDMX* active low in the second half of this cycle. The ET4000/W32i goes idle for 16 LCLKs and repeats the previous operation with all 16 bits of CPU data out to the appropriate bytes. The ET4000/W32i asserts RDY* low at the end of the second cycle. RS<1:0> will increment by 1 during the second cycle.

If only byte operation is required, ET4000/W32i asserts RDY* low at the end of the first cycle.

Segment Address Comparator

CRTC Indexed Register 30 bits 4:0 are compared with SEG<2:0>, A<23>, and A<22> inputs to allow memory access. If System Linear Mode is disabled, the A<21> and A<20> inputs must also be low to allow memory access. If System Linear Mode is enabled, A<21> and A<20> are address inputs. If the Image Port is enabled A<22:20> are ignored for address or comparison purposes.

Note that since CRTC Indexed Register 30 bit 4:2 default on power up to a value of 1, external inverters must be added to the address signals connected to these inputs.

Example:

> A<26:24> signals from the local bus connect to SEG<2:0> inputs of the W32i through inverters. A<23:20> signals from the local bus connect to A<23:20> inputs of the W32i. When in System Linear Mode, this allows the W32i to be decoded into any 4MB area of 128MB space.

For full 32-bit address decode, SEG<2:0>: The ET4000/W32i provides address interface for the lower 16MB of address space. Therefore, A<31:24> must be decoded by an external 74F27 into SEG<2:0> inputs.

Table 6.2.2.2-2 16-bit Local Bus A<22:20> Configuration

IMAE = 0

ET4000/W32i	Local Bus	IMA Port
A<22>	A<22>	Not used
A<21>	A<21>	Not used
A<20>	A<20>	Not used

IMAE = 1

ET4000/W32i	Local Bus	IMA Port
A<22>	Not used	IXFS
A<21>	Not used	IXLS
A<20>	Not used	IXOF

NOTE: Host A<23:20> must be decoded and connected to the ET4000/W32i's A<23> input when IMAE = 1. ET4000/W32i A<22:20> must be held low by the external interface until IMAE is set to 1.

6.3 Display Memory Design Considerations

The ET4000/W32i 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*

6.3.1 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
			BYTE 3			
512KB	Non-interleave	NC	NC	NC	NC	NC
1MB	Non-interleave	AB<8:0>	MD<31:24>	RASB	CAS<3>	MWB
2MB	Non-interleave	AB<9:0>	MD<31 24>	RASB	CAS<3>	MWB
4MB	Non-interleave	AB<9:0>	MD<31:24>	RASB	CAS<3>	MWB
2MB	Interleave	AB<8:0>	MD<31:24>	RASB	CAS<3>	MWB
					CAS<7>	
			BYTE 2			
512KB	Non-interleave	AB<8.0>	MD<23:16>	RASB	CAS<2>	MWB
1MB	Non-interleave	AB<8.0>	MD<23:16>	RASB	CAS<2>	MWB
2MB	Non-interleave	AB<9.0>	MD<23:16>	RASB	CAS<2>	MWB
4MB	Non-interleave	AB<9:0>	MD<23.16>	RASB	CAS<2>	MWB
2MB	Interleave	AB<8.0>	MD<23:16>	RASB	CAS<2>	MWB
					CAS<6>	
			BYTE 1			
512KB	Non-interleave	NC	NC	NC	NC	NC
1MB	Non-interleave	AA<8:0>	MD<15:8>	RASA	CAS<1>	MWA
2MB	Non-interleave	AA<9:0>	MD<15:8>	RASA	CAS<1>	MWA
4MB	Non-interleave	AA<9:0>	MD<15:8>	RASA	CAS<1>	MWA
2MB	Interleave	AA<8:0>	MD<15:8>	RASA	CAS<1>	MWA
					CAS<5>	
			BYTE 0			
512KB	Non-interleave	AA<8:0>	MD<7:0>	RASA	CAS<0>	MWA
1MB	Non-interleave	AA<8:0>	MD<7:0>	RASA	CAS<0>	MWA
2MB	Non-interleave	AA<9:0>	MD<7.0>	RASA	CAS<0>	MWA
4MB	Non-interleave	AA<9.0>	MD<7:0>	RASA	CAS<0>	MWA
2MB	Interleave	AA<8:0>	MD<7:0>	RASA	CAS<0>	MWA
					CAS<4>	

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.

6.3.2 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/W32i. 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/W32i 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/W32i 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 50 MHz. (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)

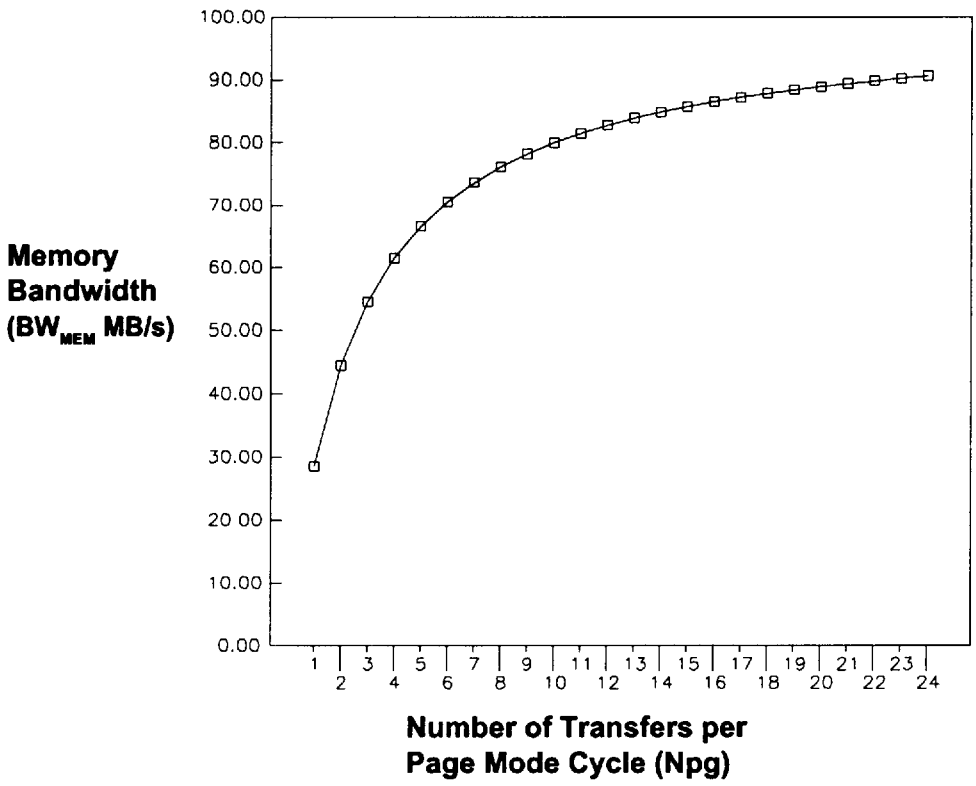
The plotted function is:

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 6.3.2-1 **Memory Bandwidth (MB/s) as a Function of Transfers per Page Mode Access**



Depending on the system requirements at any given time, the ET4000/W32i will operate at a certain point on the curve. The operation point on the curve indicates how much bandwidth the ET4000/W32i can **supply**, not how much it is **using**; so if the CRT Controller requires 20 MB/s and the ET4000/W32i 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/W32i 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/W32i'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/W32i can be expected to operate around 8 to 16 transfers per page-mode cycle.

6.3.2.1 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/W32i 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. The DRAMs should be of the same manufacturer, part number, revision level, and speed. DRAMs should have 50ns or better access time. 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.

Field upgradability is impractical because of the stringent controls necessary to ensure reliable interleave operation. For this reason it is strongly recommended that field upgrades not be considered for interleave designs.

6.3.3 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/W32i. The ET4000/W32i will meet the demands of the CRTC first, and any remaining memory bandwidth will be allotted to other requesters.

6.3.4 Host Bandwidth Requirements

Depending upon the system design (ISA, MCA, Local Bus), the Host Interface will exhibit certain bandwidth requirements. For example, on the ISA bus, the CPU can perform a 16-bit write in as little as 200ns, resulting in a demand of 10MB/s. The ET4000/W32i must have at least this much bandwidth available after servicing the CRTC's requirements in order to deliver zero-wait-state performance to the CPU. It is also important to realize that read cycles to display memory are inherently slower than write cycles.

6.3.5 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/W32i's memory controller (MCU). This implies that the accelerator performance will increase in direct proportion to faster memory chips and a wider MD bus.

6.3.6 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.

6.4 Image Bus Interface Design Considerations

The Image Port can be enabled by programming the IMA Image Port Enable Register (IMA Indexed Register F7, bit 0). When enabled, the ET4000/W32i will configure the I/O pins as described in Section 2.3 (Image Port Interface pin descriptions).

Most of the Image Port Interface I/O pins are shared with other functions so some system design limitations must be observed:

1. ISA bus design limitation: Board design must be either 8-bit slot only or 16-bit slot only (see Section 6.2.1.2 ISA Bus Special Note).

2. Local/EISA/MCA bus design limitations: 16-bit host data bus width only.

3. General limitations:
   - CS<3> select function not used.
   - SYNR function not used.
   - SWSE function not used.
   - System linear map limited to 1MB flat address space instead of 4MB.

4. The IXRD signal should be synchronized to the external image processor's clock.

5. If IMA Indexed Register 7, bit 1 is programmed to 0 (non-interlaced mode), the IXOF input pin should be low.

6.5 Clock Generator Design Considerations

6.5.1 Master Clock Select

A variable Master Clock (MCLK), is used internally by the ET4000/W32i 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/W32i as programmed. (See also Section 5.2.32, CRTC Indexed Register 34: Auxiliary Control Register.)

6.5.2 System Clock Select

The System Clock (SCLK) is required to sequence the ET4000/W32i'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/W32i'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 4.3, CRTC Indexed Register 32: RAS/CAS Configuration.)

6.5.3 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/W32i-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	31.5kHz	31.5kHz	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/W32i 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/W32i 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

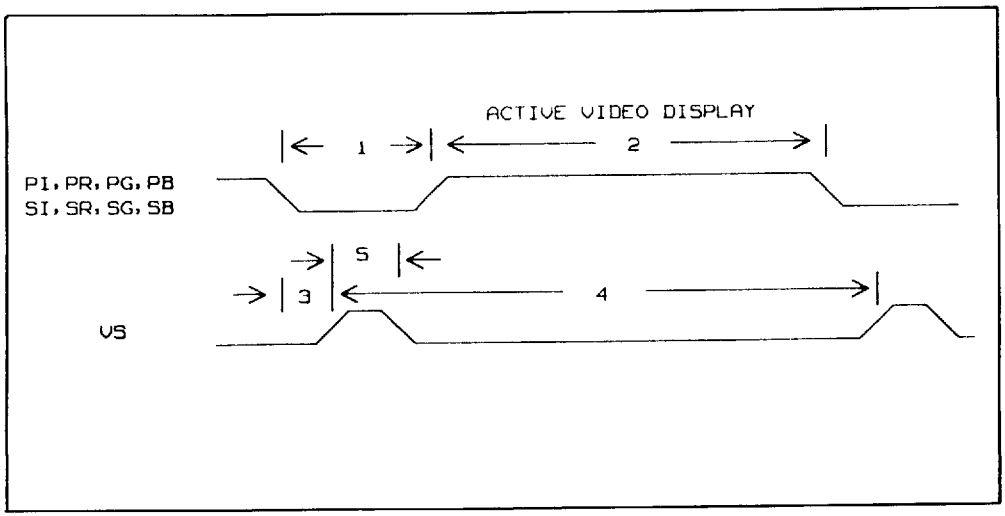


Figure 6.5.3-1 Display Vertical Sync, 350 lines

Signal Time	Typical
1	2.765 milliseconds
2	11.504 milliseconds
3	0.985 milliseconds
4	14.268 milliseconds
5	0.064 milliseconds

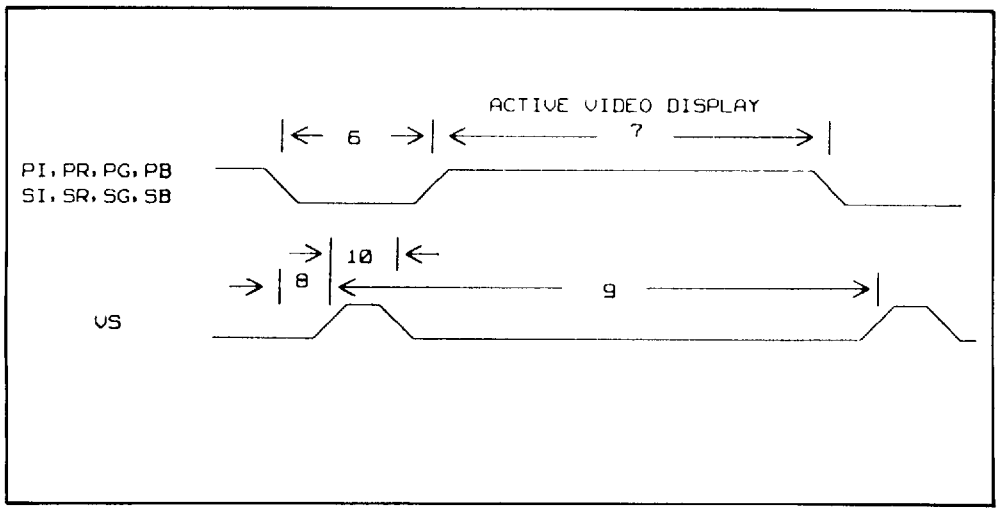


Figure 6.5.3-2 Display Vertical Sync, 400 lines

Signal Time	Typical
6	1.112 milliseconds
7	13.156 milliseconds
8	0.159 milliseconds
9	14.268 milliseconds
10	0.064 milliseconds

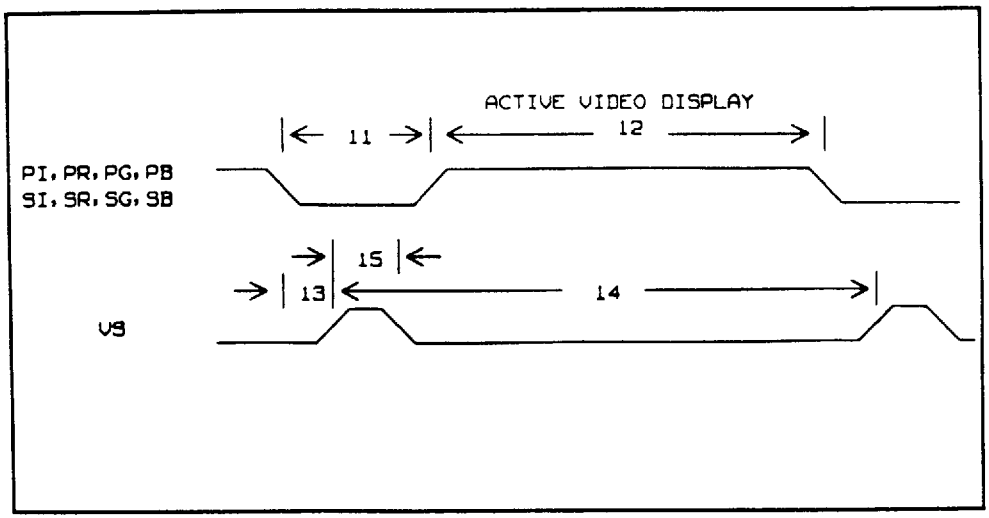


Figure 6.5.3-3 Display Vertical Sync, 480 lines

Signal Time	Typical
11	0.922 milliseconds
12	15.762 milliseconds
13	0.064 milliseconds
14	16.683 milliseconds
15	0.064 milliseconds

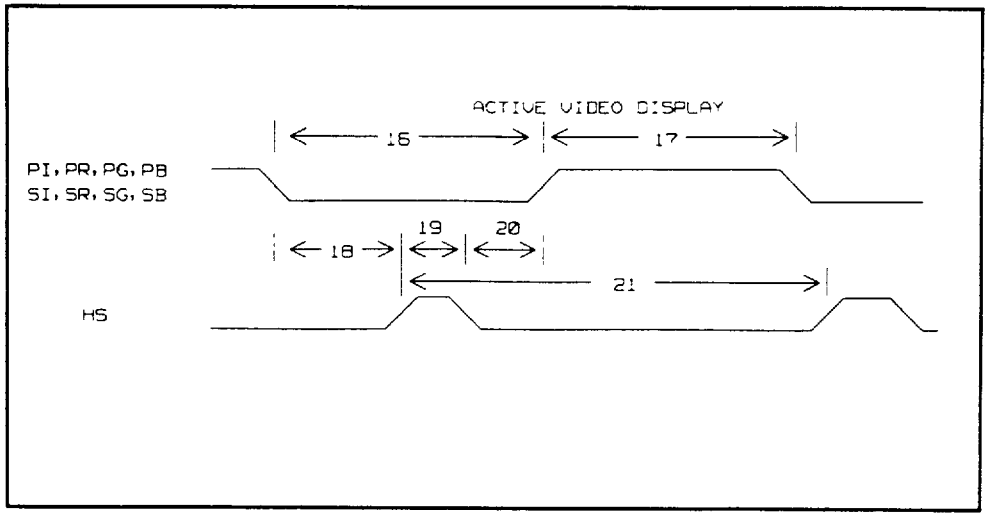


Figure 6.5.3-4 Display Horizontal Timing, 80 Column with Border

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

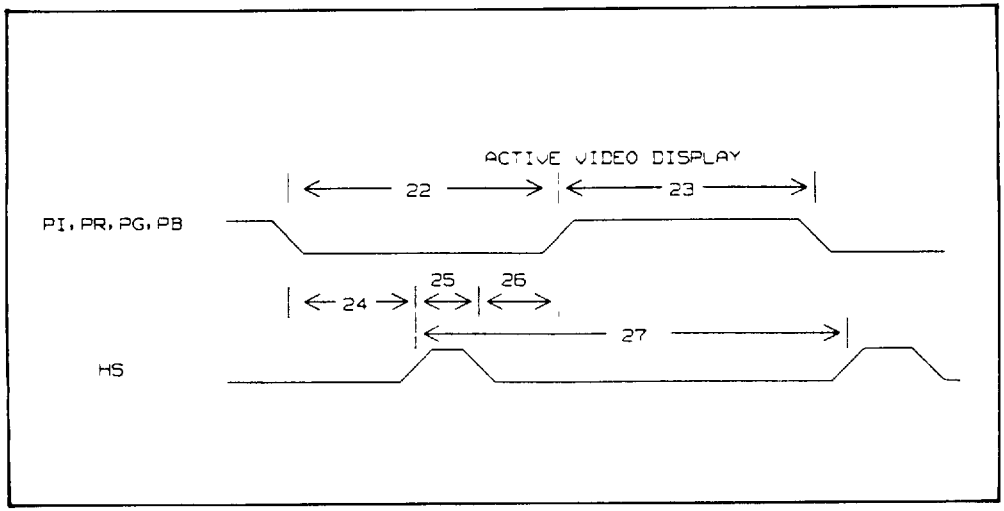


Figure 6.5.3-5 Display Horizontal Timing, 40/80 Column, No Border

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

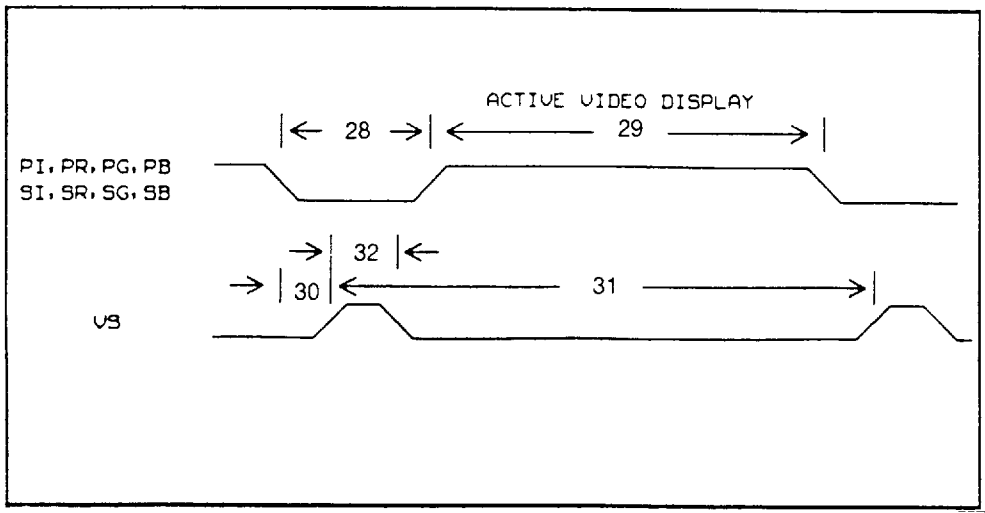


Figure 6.5.3-6 Display Vertical Sync Timing, 768 Lines

Signal Time	Typical
28	1.38 microseconds
29	21.62 microseconds
30	0.014 microseconds
31	23.0 microseconds
32	0.112 microseconds

6.6 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/W32i 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>.

6.7 RAMDAC

An external Palette RAM with Digital-to-Analog Converter (DAC) is used to translate 8-, 16- or 32 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/W32i: PMERL, PMEWL, PCLK, MBSL, AP<7:0>.

6.8 ET4000/W32i BIOS ROM

The ET4000/W32i BIOS ROM contains modules that provide generic video BIOS functions to support both VGA- and EGA-compatible modes. (See section 3.2.4 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 DB bus, with the DIR signal driven high to allow the ROM data to be read by the host processor.

6.8.1 BIOS ROM — 8 versus 16-bit

The choice between using 8 or 16-bit BIOS ROMs relates directly to motherboard or add-in board designs. For motherboard designs, an 8-bit BIOS ROM is more desirable because of the widespread use of "Shadow" RAM. This utility relocates the contents of video ROM into system RAM, so the increased performance afforded by a 16-bit ROM is rendered unnecessary. Using an 8-bit BIOS ROM also reduces cost and saves space on the motherboard design. Local Bus designs use 8-bit BIOS ROMs typically because of the availability of Shadow RAM on virtually all current-technology system board designs. A 16-bit BIOS ROM is only advantageous in systems that do not incorporate shadow RAM.

6.8.2 ROM versus "Shadow" RAM

System Shadow RAM makes 16-bit BIOS unnecessary because the contents of the video BIOS is loaded into system memory on power-up. While this initial load exacts a slight time penalty, the processing performance of the shadowed BIOS instruction far outweigh this initial delay. An 8-bit BIOS then, is more than adequate to provide processing performance with the aid of Shadow RAM. Excluding the extra BIOS ROM for 16-bit BIOS designs saves production expense and system board space.

Additionally, the 16-bit BIOS cannot be fully decoded for the CS16 signal on the ISA bus because the early unlatched LA address lines are needed, of which only 23-17 are available, and 16-15 are needed to fully decode C0000-C7FFFF. This means the late address bus must be used to decode that address space, translating to two points: either the LA lines are decoded which gives you C0000-DFFFF, or the space is fully decoded using the late lines, which decodes it to C0000-C7FFFF.

6.8.3 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.

The 16-bit BIOS configuration is faster when considering add-in video adapters used in older systems that do not incorporate shadowing. When the BIOS is actually getting used because it is not being shadowed, it is about 4 times faster than an 8-bit BIOS because a 16-bit cycle to the ISA bus is 3 bus clocks; an 8-bit cycle can be 6 bus clocks and there are two of them.

6.8.4 Specifying BIOS ROM Address Space

The ET4000/W32i is designed to decode C0000-C7FFF (hex) as the EROM address space on power-up, providing 32KB code size for the ET4000/W32i BIOS ROM modules. This address space can be redefined by programming TS Index Register 7 (TS Auxiliary Register) bits 5 and 3.

If the BIOS ROM is part of the main "motherboard" BIOS, then bits 3 and 5 of TS Indexed Register 7: Auxiliary Mode should be set to 0,1, thus disabling the decoding of ROM BIOS address space.

6.9 Translation ROM

6.9.1 Translation of CRTC Index Registers and Clock Frequency

In order to support various video modes on a variety of monitors, the CRTC data registers and the Miscellaneous Output register need to be programmed accordingly to provide proper timing for the display. The ET4000/W32i is designed to provide such adjustments via the hardware. The mechanism of translating the CRTC registers and Miscellaneous Output register is described in the following pages.

6.9.2 Translation of CRTC data registers

An alternative is provided to allow the CRTC registers to be re-programmed for the use of external translation ROM. This approach is described as follows:

6.9.3 Use of Translation ROM

Hardware configuration - (refer to board schematic figures in APPENDIX C.)

- ET4000/W32i output pin "XR16L" (active low) shall be used to enable both the read/write translation ROMs
- ET4000/W32i data bus bits 14 to 8 (BDB<14:8>) and the lower byte of the CPU data bus (DB<7:0>) (total of 15 bits) shall be used to address 32K bytes of write translation ROM
- ET4000/W32i data bus bits 14 to 8 (BDB<14:8>) and the lower byte of the ET4000/W32i data bus (BDB<7:0>) (total of 15 bits) shall be used to address 32K bytes of read translation ROM
- 8-bit data output of the write translation ROM shall be fed back to source the lower 8 bits of ET4000/W32i data bus (BDB<7:0>)
- 8-bit data output of read translation ROM shall be fed back to source the lower 8 bits of the CPU data bus (DB<7:0>)
- The DIR output from the ET4000/W32i should be used to enable the write translation ROM
- The DIR output from the ET4000/W32i with logical inversion should be used to enable the read translation ROM

Programming requirements -

- Auxiliary Control Register (CRTC Index 34) bit 5=1, to enable write translation mode
- Auxiliary Control Register (CRTC Index 34) bit 4=1, to enable read translation mode
- Video System Configuration Register (CRTC Index 36) bit 7=0, to disable 16-bit I/O operation

Theory of operation -

After the above programming, translation via use of read/write translation ROM is enabled:

Any I/O read/writes to the CRTC Index registers 0 - 1F, (port address = 3#5) will cause translation to occur as follows:

- BDB<15:8> will be sourced by ET4000/W32i (during XR16 XROML active) as follows:
  BDB<15> = TS Index 7 bit 7
  BDB<14> = Misc. Output Register bit 1
  BDB<13> = Misc. Output Register bit 2
  BDB<12:8> = CRTC Index value to be written

- The translation ROM enabled by XR16 (XROML) goes from 1 to 0, BDB<15:8> (sourced from ET4000/W32i) and DB<7:0> (sourced from CPU) will address the translation ROM for the desired value to be written to the CRTC Index Register. See Fig. 2.2.5-1.

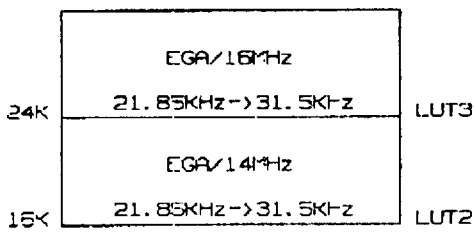
Similarly, an I/O write to the Miscellaneous Output register (port address 3C2) will cause the ET4000/W32i to activate XR16L (XROML) and source BDB<15:8>, with the exception that BDB<12:8> will be sourced as "11010," to access a desired value for the Miscellaneous Output register (MISCOUT).

NOTE: Translation ROM is not available when using Local Bus Mode.

Note that the CS(0) bit (contained in the MISCOUT register) is also an address input to the translation ROM. The MISCOUT register should be set prior to setting the CRTC values in order to select the proper EGA-to-PS/2 translation table (see Figure 2.2.5-1). Since the table to be used is unknown when the MISCOUT register is set, identical values should be in both tables at index 1A (from which the MISCOUT register is translated).

The translation of the MISCOUT allows the clock select bits (bits 2,3) and horizontal and vertical sync polarity bits (bits 6,7) to be properly adjusted.

Figure 6.9.3-1 Translation ROM



Translation ROM LUT mapping

EGA -> PS/2 analog monitor:
Translate ROM mapping = LUT2 or 3
Index 34= 2D or 21

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/W32i'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/W32i'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/W32i 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	00	xA0000-xBFFFF	xB8000-xBDFFF	—	—	—
100	01	xA0000-xAFFFF	xB8000-xBDFFF	8KB	—	—
100	10	xB0000-xB7FFF	xA8000-xADFFF	8KB	—	—
100	11	xB8000-xBFFFF	xA8000-xADFFF	8KB	—	—
110	00	xA0000-xBFFFF	xB8000-xBDFFF	—	—	—
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	x00000-xFFFFF*	—	—	—	—
001	xx	000000-3FFFFF**	—	—	—	—
111	xx	x00000-x7FFFF*	x80000-xDFFFF	128KB	xFFF00-xFFFFF	xFE000-xFEFFF
111	xx	000000-1FFFFF**	200000-37FFFF	512KB	3FFF00-3FFFFF	380000-3BFFFF

Notes:

* if IMA port enabled
** if IMA port disabled

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/W32i 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. W32i 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/W32i 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	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	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	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	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	37i	37n	2F	38i	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	37i	37n	2F	38i	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
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 7.4.4-7Timing 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 Notes

A/N = Alphanumeric modes (text).
APA = All Points Addressable modes (graphics).
* Extended Graphics Adapter text modes with 350 scan lines.
+ 9x16 character cell enhanced text modes with 400 scan lines.
** = modes require 512kb display memory.
*** = modes require 1mb display memory.
i = interlaced modes n = noninterlaced modes
# = Capable of 32,768 or 65,536 colors with Sierra HiColor DAC.
Memory requirements for HiColor modes: 13=256k, 2D and 2F=512kb, 2E and 30=1mb
% = Capable of 16.8 million colors with AT&T DAC and 1mb display memory.

NOTE: ALL BIOS MODES MAY NOT BE AVAILABLE ON ALL ET4000/W32i-BASED DESIGNS.

Table 8.1-1 ET4000/W32i 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	28.322	31.50	70.00
0*	A/N	16/256K	40x25	B8000	8x14	8	320x350	28.322	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	28.322	31.50	70.00
1*	A/N	16/256K	40x25	B8000	8x14	8	320x350	28.322	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	28.322	31.50	70.00
2*	A/N	16/256K	80x25	B8000	8x14	8	640x350	28.322	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	28.322	31.50	70.00
3*	A/N	16/256K	80x25	B8000	8x14	8	640x350	28.322	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	Monochrome	80x25	B0000	9x14	8	720x350	28.322	31.50	70.00
7+	A/N	Monochrome	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	Monochrome	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	32.514	38.70	72.70
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	32.514	38.70	72.70
13	APA	256/256K	40x25	A0000	8x8	1	320x200	25.175	31.50	70.00
21	A/N	16/256K	132x60	B8000	8x8	2	1056x480	40.000	30.50	60.00
22	A/N	16/256K	132x44	B8000	8x9	2	1056x396	40.000	30.50	70.00
23	A/N	16/256K	132x25	B8000	8x16	4	1056x400	40.000	30.50	70.00
24	A/N	16/256K	132x28	B8000	8x14	4	1056x392	40.000	30.50	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	32.514	38.70	72.70
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

Table 8.1-1 ET4000/W32i Modes (con'd)

Mode	Type	Colors/ Shades	Alpha Format	Buffer Start	Box Size	Max. Pag.	Display Size	Vid Clk (MHZ)	H Freq (KHz)	V Freq (Hz)
**2D	APA	256/256	80x25	A0000	8x14	1	640x350	25.175	31.50	70.00
**#2D	APA	32k/65k	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	32.514	38.70	72.70
**#2E	APA	35k/65k	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	35k/65k	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 35k	APA	35k/65k	100x37	A0000	8x16	1	800x600	72.000	35.50	56.00
**30 38k	APA	256/256K	100x37	A0000	8x16	1	800x600	40.000	38.00	60.00
**#30 38k	APA	35k/65k	100x37	A0000	8x16	1	800x600	80.000	38.00	60.00
**30 48k	APA	256/256K	100x37	A0000	8x16	1	800x600	50.350	48.40	72.70
**37i	APA	16/256K	128x48	A0000	8x16	1	1024x768	44.900	35.50	87.00
**37n	APA	16/256K	128x48	A0000	8x16	1	1024x768	65.000	49.00	60.50
**37 72m	APA	16/256K	128x48	A0000	8x16	1	1024x768	72.000	56.30	69.80
***38i	APA	256/256K	128x48	A0000	8x16	1	1024x768	44.900	35.50	87.00
***38n	APA	256/256K	128x48	A0000	8x16	1	1024x768	65.000	49.00	60.50
***38 72m	APA	256/256K	128x48	A0000	8x16	1	1024x768	72.000	56.30	69.80
***3Di	APA	16/256K	160x64	A0000	8x16	1	1280x1024	80.000	48.10	87.00

NOTE: ALL BIOS MODES MAY NOT BE AVAILABLE, ON All ET4000-BASED DESIGNS.

A/N = Alphanumeric modes (text)

APA = All Points Addressable modes (graphics)

* Extended Graphics Adapter text modes with 350 scan lines.

+ 9x16 character cell enhanced text modes with 400 scan lines.

** = modes require 512kb display memory.

*** = modes require 1mb display memory.

i = interlaced modes

n = noninterlaced modes

= Capable of 32,768 or 65,535 colors with Sierra HiColor DAC

% = Capable of 16.8 million colors with AT&T DAC and 1MB display memory

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: 2D, 2E, 2F.
BL=FE, BH=mode no.: Set BGR 24-bit/pixel mode (format 4); valid (256-color) mode numbers are: 2D, 2E, 2F.
BL=mode no.: Set 16-bit/pixel mode; valid (256-color) mode numbers are: 2D; 2E; 2F; 30 (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.

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/W32i (Microsoft) Raster Operations Codes and Definitions

The ET4000/W32i 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	-

Boolean function in HEX	Boolean function in R Polish	Common name
2D	PSDnox	-
2E	PSDPxox	-
2F	PSDnoan	-
30	PSna	-
31	SDPnaon	-
32	SDPS00x	-
33	Sn	NOTSRCOPY
34	SPDSaox	-
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	-

Boolean function in HEX	Boolean function in R Polish	Common name
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	-
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	-

Boolean function in HEX	Boolean function in R Polish	Common name
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	-
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	-

Boolean function in HEX	Boolean function in R Polish	Common name
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	-
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	-

Boolean function in HEX	Boolean function in R Polish	Common name
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. Micro Channel Registers

Programmable Option Select (POS)—To maintain compatibility with the Micro Channel standard, adapter cards used in PS/2 and compatible computers must contain POS registers 100, 101, and 102.

POS 100, 101: Card ID

I/O address 100-101

Bit	Description	Access
7:0	Value derived from DB<7:0> when latched.	RO

Bit	Description
7:0	This register is accessible only when -CD SETUP is active. Values for bits 7:0 are determined during Power-up/Reset. POS 100 is defined as POS I.D. low; POS 101 as POS I.D. high.

Examples: If DB<7:0> = 00h-FEh then POS 100 = 000h-0FEh
POS 101 = 080h

If DB<7:0> = FFh then POS 100 = 0FFh
POS 101 = 0FFh

POS 102: Card Enable

I/O address 102

Bit	Description	Access
7:4	Reserved.	
3	RAM wait enable.	RW
2	ROM wait enable.	RW
1	I/O wait enable.	RW
0	Card Enable.	RW

Bit	Description
Bits 3:0	Set to 0 during power-up.
Bit 3	When set to 1, indicates the implementation of a minimum of one asynchronous extended cycle (wait state) for video memory.
Bit 2	When set to 1, indicates the implementation of a minimum of one asynchronous extended cycle (wait state), for the ROM BIOS.
Bit 1	When set to 1, indicates the implementation of a minimum of one asynchronous extended cycle (wait state) for the ET4000/W32i I/O .
Bit 0	Set to 1 by the system during power-up when an adapter in the system is detected in the system.

APPENDIX C. Schematics

Sample schematics are available from your Tseng Labs Account Representative. Please contact your Account Representative for current schematic samples.

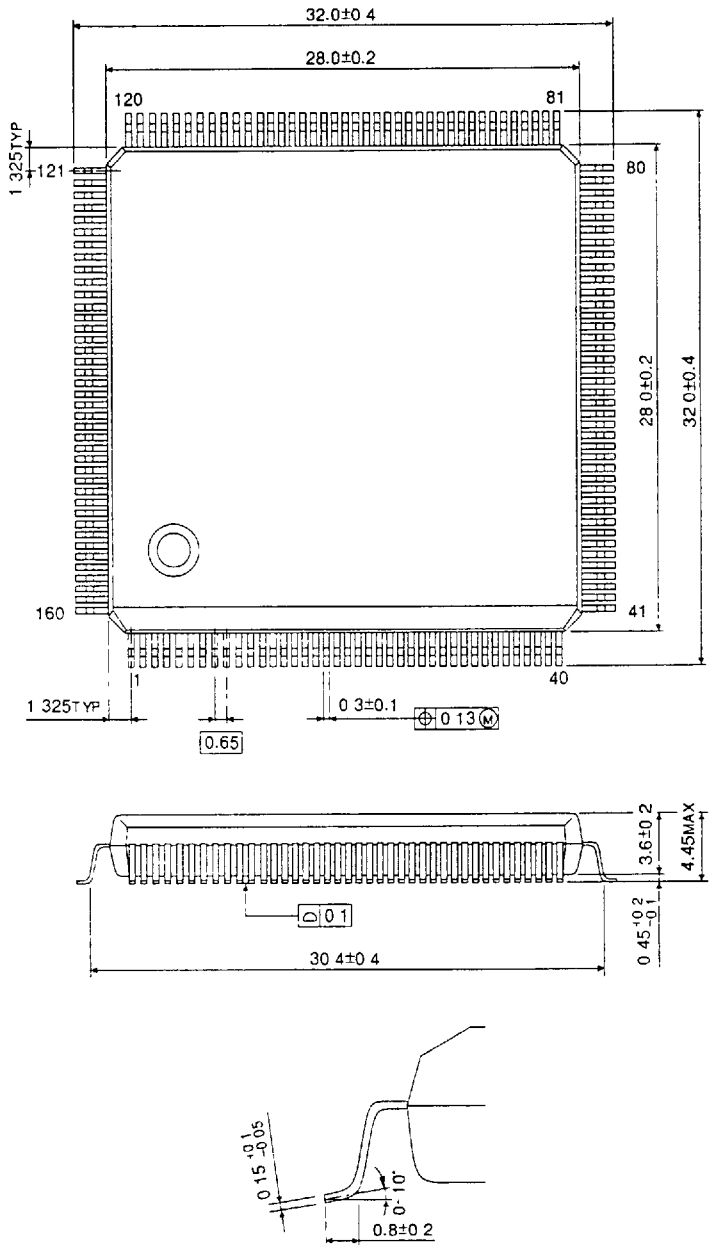
APPENDIX D. 2-Byte Character Code

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.

MECHANICAL SPECIFICATIONS

ET4000/W32i 160 Pin Plastic Flat Package



References

The following is provided to assist in acquiring support materials for host bus interfacing and VGA programming.

Host Bus Interfacing

AT Bus Design
by Edward Scolari
Annabooks
12145 Alta Caramel Court, Suite 250
San Diego, CA 92128

Micro Channel Architecture: Revolution in Personal Computing
by Pat A. Bowlds, Ph.D.
Van Nostrand Reinhold
115 Fifth Avenue
New York, New York 10003

IBM Personal System/2 Hardware Interface Technical Reference - Architectures
IBM document reference number - 84F8933

EISA Specification Version 3.12, BCPR Services, Inc.

VGA Programming

Programmer's Guide to the EGA and VGA Cards
by Richard F. Ferraro
Addison-Wesley Publishing Company, Inc.

Advanced Programmer's Guide to SuperVGAs
by George Sutty and Steve Blair
Simon & Schuster, Inc.
Brady Books

Tseng Labs Authorized Manufacturer's Representatives

MAINE • MASSACHUSETTS • NEW HAMPSHIRE • RHODE ISLAND • VERMONT • CONNECTICUT • NORTHERN NEW JERSEY • NEW YORK

New England Technical Sales
128 Warren Street
Lowell, MA 01852
Attn: Keith Caveney
TJ Caveney
(508) 458-9788
(508) 441-2737 - FAX

PENNSYLVANIA • DELAWARE • SOUTH JERSEY • MARYLAND

Omega Electronic Sales, Inc.
Four Neshaminy Interplex
Suite 101
Trevose, PA 19053
Attn: Walt Kuster
Jack Doman
Bill Kenney
Ken Kidd
(215) 244-4000
(215) 244-4104 - FAX

TEXAS • OKLAHOMA

T. L. Marketing
14850 Quorum Drive
Suite 100
Dallas, TX 75240
Attn: Bob Hamby
Michael Roso
(214) 490-9300
(214) 960-6075 - FAX

T. L. Marketing
8100 Shoal Creek, Suite 250
Austin, TX 78758
Attn: Steve Stites
Chip Waldroup
(512) 371-7272
(512) 371-0727 - FAX

T. L. Marketing
14343 Torrey Chase Blvd.
Suite I
Houston, TX 77014
Attn: John Veith
Mark Malcolm
(713) 587-8100
(713) 580-7517 - FAX

ALABAMA • TENNESSEE • MISSISSIPPI • KENTUCKY

The Novus Group, Inc.
(Main Office)
2905 Westcorp. Blvd.
Suite 120
Huntsville, AL 35805
Attn: Peter Lami
Oliver Tann
John Shiels
(205) 534-0044
(205) 534-0186 - FAX

NORTH CAROLINA • SOUTH CAROLINA

The Novus Group, Inc.
102-L Commonwealth Court
Cary, NC 27511
Attn: Andy Tribolini
Kim Best
(919) 460-7771
(919) 460-5703 - FAX

GEORGIA • EASTERN TENNESSEE

The Novus Group, Inc.
6115-A Oakbrook Pkwy.
Norcross, GA 30093
Attn: Chip Peagler
Michele Mack
(404) 263-0320
(404) 263-8946 - FAX

SOUTHERN CALIFORNIA

TNT & Associates, Inc.
22865 Lake Forest Drive
Lake Forest, CA 92630
Attn: Tom Thornton
Todd Muradian
(714) 830-6096
(714) 830-0154 - FAX

Index

A

B

C

D

E

F

G

H

I

N

O

P

Index

A

B

C

D

E

F

G

H

I

N

O

P

Q

R

S

T

U

V

W

X

Y

Z

Symbols

MECHANICAL SPECIFICATIONS

ET4000/W32i 160 Pin Plastic Flat Package