GRiDCase 1500 Series Hardware Technical Reference Manual

GRiDCase 1500 Series Hardware Technical Reference Manual

October 1988

47211 Lakeview Blvd.
P.O. Box 5003 Fremont, CA 94537-5003
(415) 656-4700
Telex No. 296226

Manual Name: GRiDCase 1500 Series Hardware Technical Reference Manual
Order Number: 1500-50
Issue date: October 1988

Limited Warranty and Limitation of Liability

You should carefully read the following terms and conditions. You will be deemed to have accepted these terms and conditions if you have not returned the product with your proof of purchase to your GRiD representative, within 5 days of the date of payment.

The term "Products," as used herein, means hardware, hardware parts, and documentation sold by GRiD.

1. Warranty

GRiD warrants that the Products sold hereunder shall be free from defects in materials and workmanship for a period of 90 days after date of shipment. Defective Products will be repaired or replaced, at GRiD's option, by GRiD during the warranty period. You must notify GRiD of the defective Product within the 90-day period, obtain a Product Repair Order number and return the Product to GRiD along with proof of purchase data. If this Product is delivered by mail, you agree to insure the Product or assume the risk of loss or damage in transit, to prepay shipping charges to the warranty service location and to use the original shipping container or equivalent. Contact an authorized GRiD Systems representative or write to: GRiD Systems Corporation, GRiD Resource Center, 47211 Lakeview Boulevard, P.O. Box 5003, Fremont, CA 94537-5003 for further information.

You are responsible for removing any data or programs and keeping backup copies prior to returning Products to GRiD. GRiD shall not be liable for loss of data contained in any returned equipment.

This warranty is not valid if the Products have been abused, misused, altered, neglected, or if repairs are attempted by unauthorized personnel. **Except as provided above, GRiD makes no warranties, expressed or implied, including any warranty of merchantability or fitness for a particular purpose.**

2. Limitation of Liability

In no event will GRiD be liable for special, incidental, or consequential damages, however caused. GRiD's liability under or for breach of this agreement shall be limited to refund of the purchase price.

3. General

You shall not assign this Agreement nor any rights hereunder without the prior written consent of GRiD. GRiD reserves the right to withhold such consent for any reason whatsoever. Subject to the foregoing, this Agreement shall bind and inure to the benefit of the respective parties hereto and their heirs, personal representatives, successors, and assigns.

The Terms and Conditions of this Limited Warranty and Limitation of Liability and of the Class A Software License Agreement represent the entire Agreement between GRiD and you relating to Product purchases and software licensing and may only be modified in writing, signed by both parties.

By reading this agreement, you acknowledge that you understand it, and agree to be bound by its terms and conditions.

TABLE OF CONTENTS

CHAPTER 3: ROM-BIOS AND I/O REGISTERS

CHAPTER 4: PROGRAMMABLE DMA CONTROLLER

CHAPTER 5: PROGRAMMABLE INTERRUPT CONTROLLER

CHAPTER 6: TIMER, SPEAKER & REAL-TIME CLOCK

CHAPTER 7: DISPLAY SUBSYSTEM

CHAPTER 7: DISPLAY SUBSYSTEM (CONTINUED)

CHAPTER 8: KEYBOARD SUBSYSTEM

CHAPTER 8: KEYBOARD SUBSYSTEM (CONTINUED)

CHAPTER 9: FLOPPY DISK DRIVE SUBSYSTEM

CHAPTER 10: COMPATIBLE HARD DISK DRIVE SUBSYSTEM

CHAPTER 11: PARALLEL PORT SUBSYSTEM

CHAPTER 12: SERIAL I/O PORT & MODEM SUBSYSTEM

CHAPTER 13: I/O EXPANSION CAPABILITY

APPENDIX A: IBM AT COMPATIBILITY

List of Illustrations

List of Tables

List of Tables (Continued)

CHAPTER 1: INTRODUCTION

This manual provides hardware related technical reference information required for understanding operation of the GRiDCase 1500 Series Computer and for connecting external devices to its input/output connectors. The information in this manual was carefully prepared to provide a complete operational guide for the peripheral design engineers, programmers, and others who require technical information about the computer functions.

HOW THIS MANUAL IS ORGANIZED

This technical reference manual is organized in chapters with each chapter describing features or a subsystem of the GRiDCase 1500 Series Computer. The balance of Chapter 1 contains an overview of the computer. Chapters 2 through 6 provide more detailed information about the computer functions as follows:

Chapter 2:	Describes system operation and memory configuration including performance characteristics with 16-bit and 32-bit microprocessors, system operating modes, and memory mapping.
Chapter 3:	Describes input/output features including an overview of the I/O register assignments and the service routines provided by the ROM-contained Basic Input/Output System (ROM-BIOS) driver and device handler. Detailed information about the I/O registers and ROM-BIOS routines is contained in the chapters provided for each subsystem of the computer.
Chapter 4:	Describes Direct Memory Access (DMA) operation.
Chapter 5:	Describes operation of programmable interrupts
Chapter 6:	Describes operation of the Timer, Speaker, and Real-Time Clock

Chapters 7 through 13 describe subsystems of the GRiDCase 1500 Series Computer. Each subsystem chapter begins with a block

diagram and an overview of the subsystem functions. Next, descriptions of the ROM-BIOS routines that service the subsystem are provided. Each chapter also contains a description of the hardware-level interface that is needed if the ROM-BIOS routines are bypassed. The subsystems described in chapters 7 through 13 are as follows:

Chapter 7. Display Subsystem
Chapter 8. Keyboard Subsystem
Chapter 9. Floppy Disk Drive Subsystem
Chapter 10. Compatible Hard Disk Drive Subsystem
Chapter 11. Parallel Port Subsystem
Chapter 12. Serial I/O Port and Modem Subsystem
Chapter 13. I/O Expansion Capability

GRIDCASE 1500 SERIES COMPUTER SYSTEM OVERVIEW

The GRiDCase 1500 Series Computer is an easily-transported, briefcase size portable computer. The computer architecture is compatible with the IBM AT personal computer and it runs the MS-DOS Operating System. The computer is also compatible with other operating systems including Xenix, OS/2, etc., and it supports a variety of "off the shelf" application programs.

In addition, The GRiDCase 1500 Series Computer is battery operable for use in areas remote to an ac power source, and is equipped with floppy disk drives and/or an internal hard disk. Figure 1-1 provides a block diagram of the GRiDCase 1500 Series Computer.

The GRiDCase 1500 Series Computer is a fully-functional computer with a self-contained operating system, memory, display, standard-size keyboard, input/output control, and power supply. The computer uses a low-power, high-performance version of a 16-bit or a 32-bit microprocessor and CMOS components where practical for low power consumption. A Numeric Coprocessor is available as an option to enhance the microprocessor performance for arithmetic functions.

Depending upon its model number, the GRiDCase 1500 Series Computer is equipped with a 16-bit 80286 type microprocessor or a 32-bit 80386 type microprocessor. The optional Numeric Coprocessor is the 80287 type or 80387 type, respectively. The microprocessors and coprocessors are described in Chapter 2.

The computer architecture includes special highly integrated circuits that greatly reduce the number of Integrated Circuits (ICs) and other components required to perform the computer functions. Reducing the number of components also reduces the overall size and increases the reliability of the computer.

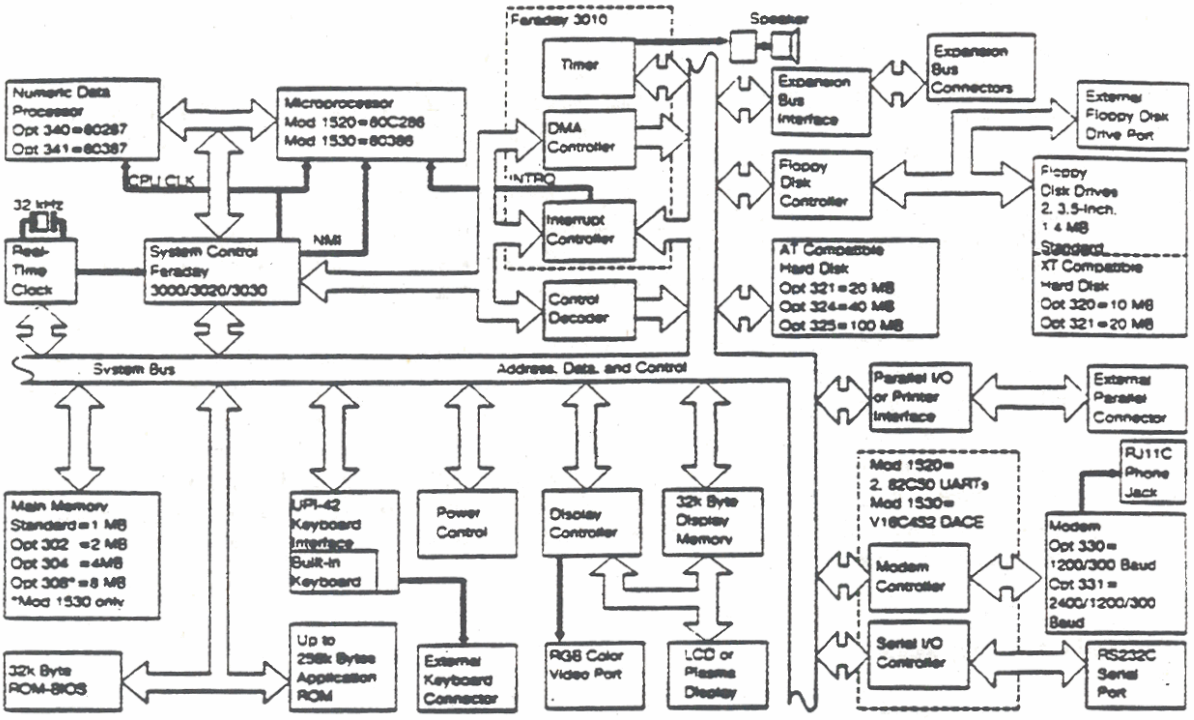


Figure 1-1. GRiDCase 1500 Series Computer System Block Diagram

The highly integrated circuits include a four-component set manufactured by Faraday Electronics, Inc., Sunnyvale, CA. This set includes the following components:

Component Type	Name	Description
FE3000A	CPU Controller	Provides control for the address and data buses, I/O parity checks, and clock control functions.
FE3010	Peripheral Controller	Provides the functional equivalent of two 8237 DMA Controllers connected in cascade mode with DMA page registers, two 8259 Programmable Interrupt Controllers connected in cascade mode, an 8254 Programmable Interval Timer, Speaker and Timer Channel control port, and the refresh address counter.
FE3020	Address Buffer	Provides buffering for the address lines and the associated control lines.
FE3030	Data Buffer	Provides buffering for the data lines and their associated control lines.

System Memory

The standard GRiDCase 1500 Series computer is equipped with 1024k (1M) bytes of Dynamic Random Access Memory (DRAM). The DRAM can be optionally increased to 2M bytes or 4M bytes, and the Model 1530 memory can be increased up to 8M bytes. The system also supports up to 256k bytes of Read Only Memory (ROM) for application programs, 32k bytes of Static RAM (SRAM) for the Display Subsystem, 200k bytes for peripheral drivers and user ROMs, and 32k bytes of ROM contained Basic Input/Output System (BIOS).

Both microprocessors employ virtual memory addressing to access memory locations beyond the installed RAM/ROM capacity. The 16-bit microprocessor automatically maps up to one gigabyte (1G byte) of virtual addresses per task into a 16 megabyte (16M byte) real address space. The 32-bit microprocessor directly addresses up to 4G bytes of real address space and addresses up to 64 terabytes (64T bytes) of virtual memory. System memory is described in more detail in Chapter 2.

Display Subsystem

The GRiDCase 1500 Series Computer display subsystem contains a special display controller that supports a choice of internal flat panel displays or an external video display. The internal flat panel displays provide a 10-inch diagonal, 80 character by 25 line, 640 x 400 pixel resolution, bit-mapped graphic display capability, with one of the following three screen types:

1. Transreflective, backlit, supertwist LCD (standard)

2. Gas plasma (Option 282)

3. Transmissive, backlit, supertwist, LCD (Option 283)

A color video display capability is also available through an external connector to drive a video monitor. Additional Display Subsystem capability is described in Chapter 7.

Keyboard Subsystem

The GRiDCase 1500 Series Computer provides a standard-size, 72-key keyboard that emulates all 84 keys of the standard IBM AT keyboard. The keyboard keys are color coded to aid in learning and using the keystrokes required for emulation of the IBM AT keyboard. For example, the "Fn" key has blue lettering, and is depressed to enable the functions indicated by blue lettering on other keys. The blue lettered functions include Print Screen (PrtSc), Scroll Lock (ScrLk), Number Lock (NmLk), etc. The keyboard also includes a row of 12 function keys (F1 through F12).

In addition to the built-in keyboard, the Keyboard Subsystem supports an external IBM AT compatible 84-key keyboard, an enhanced keyboard with 101/102 keys, or an external keypad for numbers and mathmetic functions. The external keyboard or keypad is connected via a 5-pin DIN type receptacle that is located on the back panel of the computer. A description of the Keyboard Subsystem capability is provided in Chapter 8.

Floppy Disk Drive Subsystem

The standard GRiDCase 1500 Series computer is equipped with two internal floppy-disk drives, which are located on the side of the computer (see Figure 1-2). The internal floppy-disk drives accept double-sided, 3.5-inch diskettes with a formatted capacity of 1.4M bytes or 720k bytes. Support for an external 3.5-inch or 5.25-inch floppy disk drive is provided via an external disk drive port located on the back of the computer. A more detailed Floppy Disk Drive Subsystem description is contained in Chapter 9.

Compatible Hard Disk Drive Subsystem

The GRiDCase 1500 Series Computer is optionally available with several internal hard disk drive configurations. The optional hard disk drives include a 20M byte drive that is IBM XT compatible but not IBM AT compatible (Option 321). A different 20M byte drive (Conners) that is IBM AT compatible (also Option 321) is also available. Either 20M byte drive is installed in place of one 3.5-inch floppy disk drive. The IBM XT compatible disk drive provides 8-bit data transfers using DMA. The IBM AT compatible disk drive provides 16-bit data transfers using Programmed Input/Output (PIO).

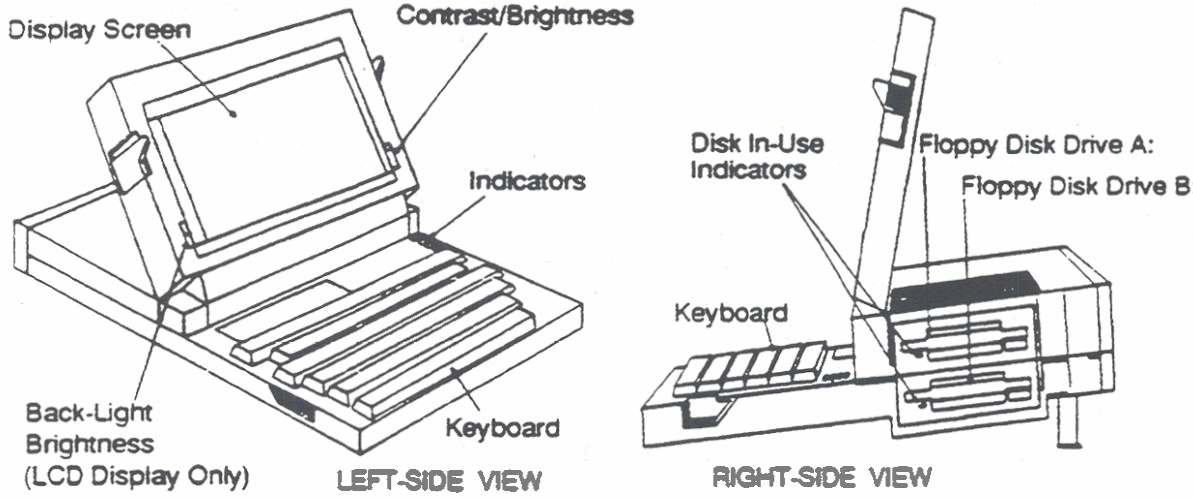


Figure 1-2. GRiDCase 1500 Series Computer
Left and Right Side Views

Internal hard disk drives with formatted capacities of 40M byte and 100M byte are also available (options 324 and 325, respectively). Both of these larger capacity disk drives are IBM AT compatible and are installed in place of both internal floppy disk drives.

In any of the disk drive configurations, an external disk drive port located on the back of the computer is available. The disk drive port can be used to connect an external 3.5-inch or 5.25-inch floppy disk drive. Also, the port can be used to connect a 40M byte backup tape unit. A detailed description of the Compatible Hard Disk Drive Subsystem is provided in Chapter 10.

Parallel Port Subsystem

The Parallel Port Subsystem is functionally equivalent to the IBM AT printer adapter with additional features for the GRiDCase 1500 Series Computer applications. The port supports Centronics-type (parallel) printers and also can be used as a general purpose, parallel I/O port. As a bidirectional port, the parallel printer interface can be connected to devices that require a parallel data port. The interface connection is provided through a 25-Pin, D-shell, female connector located on the rear panel. The Parallel Port Subsystem is described further ib Chapter 11.

Serial I/O Port

The GRiDCase 1500 Series Computer provides a serial interface I/O port, which uses a UART type device to support asynchronous communications. The serial I/O port is RS-232-C compatible and it allows user programmable control of parameters such as transmission rate, character size, stop bits, and parity. The interface connector is a 9-pin D-shell male connector located on the rear panel (see Figure 1-3). The Serial I/O Port is described in Chapter 12.

NOTE: In standard configurations, COM 2 provides the Serial Port Interface and COM 1 provides the Modem Interface. This configuration may be swappable depending upon the revision level of the MS-DOS operating system.

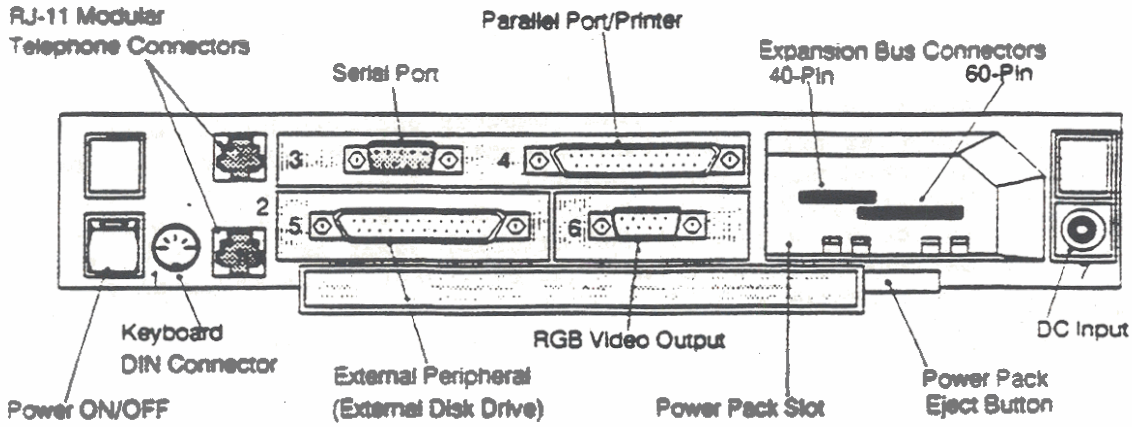


Figure 1-3. GRiDCase 1500 Series Computer Rear View

Modem (Optional)

A choice of two factory installed internal Hayes-compatible modems is optionally available for the GRiDCase 1500 Series computer. Model 330 provides Baud rates of 1200 and 300 and Model 331 provides Baud rates of 2400, 1200, and 300. Both models are Bell 212A and 103 compatible, and the Model 231 is also V.22bis-compatible. The installed modem connects to a commercial telephone network by direct attachment through either of two RJ-11 female connectors on the rear of the computer. The modem is further described in Chapter 12.

I/O Expansion Bus

Within the power pack slot on the rear of the computer, there is one 40-pin and one 60-pin edge connector. These connectors interface with specially designed I/O expansion cartridges that can be inserted in the power pack slot whenever an external power pack is being used. Additional information on the I/O Expansion Bus is provided in Chapter 13.

Power

The GRiDCase 1500 Series Computer operates on dc power. The computer operates on an integral battery pack that is installed at the back panel (see Figure 1-3). A power receptacle near the battery pack allows the computer to be connected to an external source of dc power. The external power source can be another battery pack or an ac-to-dc converter. The external power source must provide +10.5V to +18V dc.

The standard Internal/External Power Supply provided with the computer is an autosensing ac-to-dc converter. The converter connects to a 110/220 Volt ac power source and supplies +16V dc to the computer. This standard power supply can be inserted in the battery pack slot for direct connection to the computer, or may be connected through the dc input connector using a dc-to-dc cable.

Computers with 32-bit microprocessors combined with a gas plasma display and 40M byte hard disk, require more power than is provided by the internal battery pack or standard ac-to-dc converter. For these computers, a heavier duty external autosensing ac-to-dc converter is supplied (Model No. 34170). This external power supply is connected to the computer through the dc input connector using a dc-to-dc cable.

Controls and Indicators

The GRiDCase 1500 Series Computer provides controls and indicators that are accessible to the computer user. The controls consist of the Power ON/OFF switch on the back panel of the computer and two brightness controls. When operating the computer from battery power, the brightness controls should be set to the lowest level possible that is consistent with good visibility. Less power is consumed at lower brightness levels and therefore the battery will retain its charge longer.

The computer indicators consist of a column of four Light Emitting Diodes (LEDs) located just above the Keyboard Back Space (BkSp) key. The LEDs glow red or green to provide specific indications, and in one case, when both the red and green are on, the LED appears to glow orange.

Table 1-1 lists the controls and indicators and provides a brief description of their purpose. See Figure 1-2 for location information.

Table 1-1. GRiDCase 1500 Series Computer Controls and Indicators

Name	Description
Power ON/OFF	Switches dc power to the computer. Switch must be ON for operation.
Back-Light Brightness (LCD)	Controls back-light brightness on LCD display screens. The brighter the back-light, the greater the power consumption by the display.
Contrast/Brightness	Controls brightness of the displayed image. The brighter the image, the greater the power consummed.
Battery Low/Charging	Indicates battery pack condition: Red indicates that the battery pack power is low and requires a recharge. A power failure is imminent. Green indicates that the battery is being recharged. When the indicator is not lit, the battery output is within operating range or no battery is installed.
Lower/External Disk In Use	Green indicates that a disk access is taking place on the lower disk drive if two drives are installed. The lower disk drive can be either a floppy or a hard disk drive. Red indicates that a disk access is taking place on an external drive. Orange indicates that a disk access is taking place on both the lower and the external disk drives.
Processor Low	Red indicates that the slow processor speed has been selected: 5 MHz for the 16-bit microprocessor and 6.25 MHz for the 32-bit microprocessor.
Upper Disk In Use	Green indicates that a disk access is taking place on the upper disk drive when two drives are installed. Not used if only one drive is installed.

CHAPTER 2: SYSTEM OPERATION AND MEMORY CONFIGURATION

This chapter describes the GRiDCase 1500 Series computer system operation and performance characteristics with a 16-bit or 32-bit microprocessor, optional numeric coprocessor operation, and system operating modes. Also, this chapter contains information about the system memory configuration and memory options, memory mapping, and system power mode control. The system Input/Output (I/O) information including assignments for all of the I/O Registers, is summarized in Chapter 3.

MICROPROCESSORS

The GRiDCase 1500 Series computers use either of two microprocessors depending upon the system model number. The two microprocessors are as follows:

Model Number	Micro-Processor	Description
1520	80C286	16-bit High-Performance Microprocessor with Memory Management and Protection
1530	80386	32-bit High-Performance Microprocessor with Integrated Memory Management and Protection

80C286 Microprocessor

The 80C286 Microprocessor is a low-power (CMOS) version of the 80286 Microprocessor. The 80C286 Microprocessor provides an advanced, 16-bit, 10 MHz (100 nsec), high-performance processor that integrates multiuser/multitasking support, memory management, and four levels of memory protection on one printed circuit assembly. Depending upon the application, the 80C286 Microprocessor is up to six times faster than the standard 8086 Microprocessor, while providing complete upward software compatibility.

The 80C286 Microprocessor contains the same basic set of registers, instructions, and addressing modes as the 8086 and 8088 Microprocessors and the 80C286 is object code compatible with existing 8086/8088 software.

In addition, the 80C286 Microprocessor can map one gigabyte (1G byte) of virtual addresses per task into a 16 megabyte (16M byte) real address space. The additional capability also provides memory protection to isolate the operating system and ensure the privacy of programs and data for each task.

Performance characteristics for 80C286 Microprocessor running in the Model 1520 Computer are as follows:

1. The microprocessor runs at 10 MHz (100 nsec) clock rate with 1 wait state.

2. The microprocessor executes one 16-bit wide transfer in 300 nsec for a maximum bus bandwidth of 6.67M Bytes/sec.

. The following list provides data transfer rates for different conditions on the data bus.

Microprocessor Length	Destination Length	Number of CPU Clocks
16	16	3
8	8	6

These values assume word or double word alignment. Misaligned transfers can double the number of clock cycles required to perform the operation.

4. When transferring data between an external peripheral device and the I/O address space, the hardware slows down the timing so that the device has time to respond, unless the device asserts the ZEROWS- signal on its own device interface.

5. Data transfer rates for external devices that do not assert ZEROWS- are provided in the following list.

Microprocessor Length	Destination Length	Number of CPU Clocks
16	16	7
16	8	7
8	8	7

80386 Microprocessor

The 80386 Microprocessor provides a 32-bit, 12.5 MHz (80 nsec), high-performance processor that integrates multiuser/multitasking support, memory management, four levels of memory protection, address translation caches, and a high-speed bus interface on one printer circuit assembly.

The 80386 Microprocessor provides the same base architecture as the 8086/8088 Microprocessors and uses a similar addressing mechanism, memory size, and interrupt handling. This base architecture allows the 80386 to be object code compatible with existing 8086/8088/80C286 software.

In addition, the 80386 Microprocessor contains 8 general purpose 32-bit registers and supports 32-bit addresses and data types. This processor addresses up to four gigabytes of physical memory, and 64 terabytes (2^{46}) of virtual memory. The integrated memory management and protection architecture includes address translation registers, advanced multitasking hardware, and a protection mechanism to allow simultaneous running of multiple operating systems.

Performance characteristics for 80386 Microprocessor running in the Model 1530 Computer are as follows:

1. The microprocessor runs at 12.5 MHz (80 nsec) clock rate with 1 wait state for interleaved RAM.

   - The microprocessor executes one 32-bit wide transfer in 240 nsec for a maximum bus bandwidth of 16.7M Bytes/sec.

   - The following list provides data transfer rates for different conditions on the data bus.

Microprocessor Length	Destination Length	Number of CPU Clocks Interleaved	Not Interleaved
32	32	3	4
16	16	3	4
8	8	3	4

These values assume word or double word alignment. Misaligned transfers can double the number of clock cycles required to perform the operation.

4. When transferring data between an AT-compatible device and the I/O address space, the hardware slows down the timing so that the device has time to respond.

5. The following list provides data transfer rates for AT compatible devices.

Microprocessor Length	Destination Length	Number of CPU Clocks
32	16	8-9
32	8	18-20
16	16	7
16	8	8-9
8	8	7-14

COPROCESSOR (OPTIONAL)

The GRiDCase 1500 Series computers are available with an optional numeric coprocessor. The numeric coprocessor enhances the performance of the primary microprocessor by performing mathematical instructions in parallel with other operations of the microprocessor. The microprocessor automatically passes the mathematical instructions to the coprocessor whenever they are encountered. Coprocessors used with the GRiDCase 1500 Series computers depend upon the system model number and options as follows:

Model Number	Micro-Processor	Option Number	Co-Processor	Remarks
1520	80C286	340	80287	Uses a 16-bit wide bus
1530	80386	341	80387	Uses a 32-bit wide bus

80287 Coprocessor

The 80287 Coprocessor uses the same clock generator as the 80C286 Microprocessor, but works at one-third the frequency. The coprocessor functions as an I/O device through I/O addresses 0F8h, 0FAh, and 0FCh. The microprocessor sends OP codes and operands through these I/O ports and receives and stores results through the same ports.

The coprocessor detects six different exception conditions that can occur during instruction execution. If an exception occurs, and

its exception mask is not set, the coprocessor generates an error signal. The error signal in turn, generates interrupt INT 75h and sets the coprocessor BUSY flag. The Coprocessor BUSY flag is cleared by a WRITE command to the 8-bit I/O register 0F0h, with all bits (D0-D7) set to "0."

The coprocessor has two operating modes similar to the operating modes of the microprocessor (refer to the subsequent paragraphs in this chapter). The coprocessor enters its default Real Address Mode following a system reset or power turn ON and also following a WRITE command to the I/O register at address 0F1h. This Real Address Mode is compatible with software that uses the 8087 Coprocessor.

The 80287 Coprocessor enters Protected Virtual Address Mode by executing the SETPM ESC instruction. To return to the Real Address Mode, a WRITE command is issued to the I/O register at address 0F1h with all bits in the 8-bit register (D0-D7) set to "0."

During a system reset or power turn ON, the system enables interrupt 75h and sets the interrupt vector to point to a routine in the ROM-BIOS. The routine in ROM-BIOS first clears the BUSY flag and then transfers control to the address pointed to by the NMI interrupt vector. The NMI interrupt handler must read the coprocessor status to determine if it caused the interrupt. If the coprocessor did not cause the interrupt, control must be returned to the original NMI interrupt handler. Refer to the NMI Interrupt description in Chapter 3.

80387 Coprocessor

The 80387 Coprocessor is directly coupled to the 80386 Microprocessor through a 32-bit synchronous data bus. When the 80386 encounters a coprocessor instruction, it automatically generates one or more I/O cycles to addresses 800000F8h and 800000FCh. Address 800000F8h is the write command address and 800000FCh is the read and write data address.

The microprocessor performs all necessary bus cycles to memory and transfers data to and from the coprocessor. Read cycles, which transfer data from the coprocessor to the microprocessor, require at least one wait state. Write cycles to the coprocessor do not require any wait states.

The microprocessor initiates coprocessor operations under program control and during execution of a coprocessor (ESC) instruction. The coprocessor uses a PEREQ signal to request operand transfers to or from system memory. Since these operand transfers occur when the coprocessor requests them, they are synchronous to the instruction execution of the microprocessor.

When the microprocessor executes an ESC instruction that requires the transfer of operands to or from the coprocessor, the microprocessor automatically performs the following operations:

1. Sets an internal Memory Address Base register
2. Sets a Memory Address Limit Register
3. Sets an I/O Direction flag

The coprocessor then requests operand transfers by driving its PEREQ output to the active state while it is executing an instructions (BUSY- is active). Each operand transfer requires from two to five bus cycles to complete depending upon the type of operation being performed as follows:

1. Transfers with aligned operands require one coprocessor cycle and one memory cycle.
2. Transfers with misaligned operands require one coprocessor cycle and two memory cycles.
3. Transfers of 32-bit operands to 16-bit memory requires one coprocessor cycle and either two or three memory cycles.
4. Transfers of 64-bit operands to 16-bit memory requires one coprocessor cycle and four memory cycles.

Coprocessor data transfers have the same bus priority as microprocessor programmed data transfers.

OPERATING MODES

Operation of a GRiDCase 1500 Series computer is controlled by its operating mode. The operating mode determines the system addressing mechanism, addressable memory size, interrupt handling, and instruction execution. The two operating modes are called:

1. Real Address Mode (Default)
2. Protected Virtual Address Mode

Real Address Mode

When the system is reset or power is turned ON, the microprocessor is initialized in Real Address Mode, which is the default mode. In Real Address Mode, both the 80C286 and the 80386 Microprocessors

have the same base architecture as an 8086 Microprocessor. All of the 8086 instructions can be performed and programs use real addresses with up to 1M byte of addressable space. Physical memory is limited to 640k bytes. However, EMS type expandable memory can be used to increase physical memory space up to 4M bytes.

Protected Virtual Address Mode

In Protected Virtual Address Mode, the full capabilities of the advanced microprocessors are available to the system. In this mode, operation of the computer is dependent upon which microprocessor is being used. Operation in the Protected Virtual Address Mode is described separately for each microprocessor in the following paragraphs.

80C286 Microprocessor Operation

The 80C286 Microprocessor enters Protected Virtual Address Mode from Real Address Mode by setting the Protect Enable (PE) bit of the machine status word with the Load Machine Status Word (LMSW) instruction. Protected mode offers extended physical and virtual address space, memory protection mechanisms, and support for the operating system and virtual memory.

All of the registers, instructions, and addressing used in Real Address Mode remain the same. However, the maximum addressable space is increased from 1M byte to 16M bytes. Up to one gigabyte of virtual address space per task can be mapped into the 16M bytes of space. The virtual address space can be larger than the physical address space because any address that does not map to a physical memory location causes a restartable exception.

80386 Microprocessor Operation

The 80386 Microcomputer enters Protected Virtual Address Mode from Real Address Mode by loading Control Register 0 with the Protection Enable (PE) bit set. After enabling the Protected Mode, the next instruction should execute an intersegment jump (JMP) to load the Code Segment (CS) register and flush the instruction decode queue. Finally, all of the Data Segment registers are loaded with the initial selector values.

Within Protected Virtual Address Mode the linear address space is increased to four gigabytes (4G bytes) and virtual memory is increased to 64 terabytes (64T bytes). Protected Mode allows the 80386 to run all of the 8086/80C286 programs while providing a more advanced memory management and a hardware-assisted protection

mechanism. Protected Mode also allows the use of additional instructions that support multitasking operating systems. The main differences between the Real Mode and Protected Mode for the 80386 are the increased address space and a different addressing mechanism.

Also, within Protected Mode the software can perform a task switch to enter Virtual 8086 mode. In Virtual 8086 mode, each task is performed as 8086 instructions, which allows all 8086 software to be executed. Virtual 8086 tasks can be isolated from one another and from the 80386 operating system by the use of paging and the I/O Permission bit map.

SYSTEM MEMORY

Main memory for the standard GRiDCase 1500 Series computer system is contained in 1024k (1M) bytes of 120 ns dynamic RAM. The dynamic RAM main memory can be optionally increased to 2M bytes (Option 302) or 4M bytes (Option 304). The Model 1530 memory can be optionally increased to 8M bytes (Option 308). The system also supports up to 512k bytes of application ROM, 32k bytes of static RAM to support the display subsystem, a 200k byte address space for peripheral drivers and option ROMs, and 64k bytes of ROM-BIOS. Each of these addressable segments of system memory are described in the following paragraphs. Refer to the system block diagram in Figure 1-1.

Memory Mapping

The GRiDCase 1500 Series computers can contain up to 8M bytes of internal memory (RAM) and have the capability of addressing up to 16M bytes (80C286) or 4G Bytes (80386) of system memory. Three factors determine how the system memory is mapped. The first factor is the physical memory size, and the second factor is the size of the RAM ICs used to make up the physical memory. The third factor that determines how the main system memory is mapped is the microprocessor type.

For the 80C286 Microprocessor, the internal main system memory is 16-bits wide and each 1M byte of memory is configured as 512k x 16 bits. This memory runs at the speed of the 80C286 cycle (10 MHz) with 1 wait state. If additional memory (up to 4M bytes total) is attached via the AT bus, its performance will be slower.

The 80386 Microprocessor internal main system memory is 32-bits wide and each 1M byte of memory is configured as 256k x 32 bits. This memory runs at the speed of the 80386 cycle (12.5 MHz) with 1

wait state and pipelined accesses. If additional memory (up to 8M bytes total) is attached via the AT bus, its performance will be slower.

Additional information on memory mapping for each microprocessor type is provided in the following paragraphs:

80C286 Microprocessor Memory Mapping

The 16-bit 80C286 Microprocessor (Model 1520 Computer) addresses up to 16M bytes of main system memory. The lowest 640k bytes are directly addressable memory locations, and all memory above the 640k byte boundary is treated as either Extended RAM or Expanded RAM. Extended RAM is the IBM AT compatible (EXT) memory configuration. Whereas, the Expanded RAM requires an Expanded Memory Specification (EMS) software driver that meets the specification developed by Lotus/Intel/Microsoft (LIM). The Expanded memory (EMS) operation is described in subsequent paragraphs of this chapter.

The user selects how memory above 640k bytes is treated via an interrupt (INT 15) as described in Chapter 3, or via the MODE command. The command MODE MEM=EXT specifies Extended (AT compatible) RAM and the MODE MEM=EMS command specifies Expanded (EMS compatible) RAM. Refer to the *MS-DOS Reference: Release 3.2* and subsequent.

For MEM=EXT, the low order bit of Real-Time Clock memory location 3Fh is reset to "0." For MEM=EMS, the same bit is set to "1." At system initialization or reset, the system checks the same bit to determine the hardware configuration of the system. Table 2-1 lists a typical 80286 Microprocessor-based memory map to address up to 16M bytes of memory using both AT compatible Extended (EXT) RAM and EMS compatible RAM. The actual memory map is listed in Table 2-3. Memory locations are given in hexadecimal (h).

80386 Microprocessor Memory Mapping

The 32-bit 80386 Microprocessor (Model 1530 Computer) directly addresses up to 4G bytes of memory. However, mapping of the memory depends upon how much memory is actually installed. The Model 1530 computer provides main memory in increments of 1, 2, 4, and 8M bytes depending upon the installed options, and addressable system memory locations up to 16M bytes. Memory locations above 640k bytes are accessed as AT compatible Extended Memory (AT). Table 2-2 lists a typical 80386 Microprocessor-based memory map to address up to 16M bytes of system memory using AT compatible Extended (EXT) RAM. The actual memory map is listed in Table 2-3. Memory locations are given in hexadecimal (h).

Table 2-1. Typical 80C286 Microprocessor-Based Memory Map

MODE MEM = EXT		Memory Configuration			
Memory Location		1M Byte	2M Byte	4M Byte	8M Byte
Hex (h)	Decimal	256k RAMs	256k RAMs	1M RAMs	1M RAMs
000000h	00k				
080000h	512k	User	User	User	User
0A0000h	640k	Video	Video	Video	Video
0B0000h	704k	CGA Display RAM in all configurations			
0D0000h	832k	Peripheral Drivers and Option ROMs			
0F0000h	960k	ROM-BIOS in all configurations			
100000h	1M	Extended (EXP) RAM Start			
160000h	1.384M	User	User	User	User
1D0000h	1.832M	AT/PC	User	User	User
200000h	2M	AT/PC	User	User	User
260000h	2.384M	AT/PC	User	User	User
300000h	3M	AT/PC	AT/PC	User	User
400000h	4M	AT/PC	AT/PC	AT/PC	User
800000h	8M	NA	NA	AT/PC	AT/PC
F00000h	15M	FF8000h through FFFFFFh is			
FFFFFFh	16M	ROM-BIOS in all configurations			

MODE MEM = EMS		Memory Configuration			
Memory Location		1M Byte	2M Byte	4M Byte	8M Byte
Hex (h)	Decimal	256k RAMs	256k RAMs	1M RAMs	1M RAMs
000000h	0k				
080000h	512k	User	User	User	User
0A0000h	640k	User	User	User	User
0B0000h	704k	CGA Display RAM in all configurations			
0D0000h	832k	Peripheral Drivers and Option ROMs			
0F0000h	960k	EMS (24)	EMS (88)	EMS (128)	EMS (128)
100000h	1M	ROM-BIOS in all configurations			

NOTES:

1. User = User or Main memory
2. NA = Not applicable for the listed RAM IC size.
3. AT/PC = Memory locations usable by IBM AT style (16-bit wide) or IBM PC style (8-bits wide) RAM.
4. Application ROMs are bank switched into the Display RAM area between A0000h and BFFFFh. Refer to the description in the following paragraphs.
5. EMS (xx) = Extended Memory Specification RAM is mapped in four 16k byte pages to physical memory locations 0E0000h-0EFFFFh. The (xx) represents the maximum number of 16k byte pages supported by each EMS RAM configuration.

Table 2-2. Typical 80386 Microprocessor-Based Memory Map

Memory Location		Memory Configuration			
Hex (h)	Decimal	1M Byte	2M Byte	4M Byte	8M Byte
000000h	00k	User	User	User	User
0A0000h	640k	Video	Video	Video	Video
0B0000h	704k	CGA Display RAM in all configurations			
0D0000h	832k	Peripheral Drivers and User ROMs			
0F0000h	960k	ROM-BIOS in all configurations			
100000h	1M	Extended (EXP) RAM Start			
160000h	1.384M	User*	User	User	User
200000h	2M	Not Used	User	User	User
260000h	2.384M	AT	User*	User	User
300000h	3M	AT	Not Used	User	User
400000h	4M	AT	AT	User	User
460000h	4.384M	AT	AT	User	User
500000h	5M	AT	AT	Not Used	User
600000h	6M	AT	AT	AT	User
660000h	6.384M	AT	AT	AT	User
700000h	7M	AT	AT	AT	User
800000h	8M	AT	AT	AT	User
860000h	8.384M	AT	AT	AT	User*
900000h	9M	AT	AT	AT	AT
A00000h	10M	AT	AT	AT	AT
B00000h	11M	AT	AT	AT	AT
C00000h	12M	AT	AT	AT	AT
D00000h	13M	AT	AT	AT	AT
E00000h	14M	AT	AT	AT	AT
F00000h	15M	AT	AT	AT	AT
FF0000h	15.960M	FF8000h through FFFFFFh (32k)is high			
FFFFFFH	16M	memory ROM-BIOS in all configurations			

NOTES:

1. User = User or Main memory
2. User* = User or Main memory that can be disabled. Disabling causes a loss of 384k of Main memory but allows the contiguous 1M byte of address space to be used by Extended (EXT) memory.
3. Not Used = Memory locations that are not accessible.
4. AT = Memory usable by IBM AT style (16-bit wide) RAM.
5. Application ROMs are bank switched into the Display RAM area between A0000h and BFFFFh. Refer to the description in the following paragraphs.

Main Memory Location

Main memory is allocated starting at the low end of the available address space (address 000000h). The standard 1M bytes of user memory are split between low and high areas. The low area is located between address 000000h and 0A0000h (640k). The high area of user memory is located between address 100000h and 160000h (384k). The low memory between 640k and 1M bytes is allocated to display RAM, peripheral drivers, and the low portion of the ROM-BIOS. The first 1M byte portion of memory is allocated in the same way for all configurations (refer to Tables 2-1 and 2-2.

The high area of the available address space contains the EMS or EXT RAM and the high portion of the ROM-BIOS. A more detailed listing of the main memory locations is provided in Table 2-3.

Table 2-3. GRiDCase 1500 Series Computer Main Memory Map

Memory Location From/To	Decimal From/To	Purpose
000000h/0003FFh	0k/1k	System Interrupt Vectors
000400h/00047Fh	1k/1.15k	ROM-BIOS RAM Check
000480h/09FFFFh	1.15k/640k	MS-DOS and User Programs
0A0000h/0AFFFFh	640k/704k	EGA/VGA Display RAM (Optional)
0B0000h/0B7FFFh	704k/736k	Monochrome Display RAM (Not Used)
0B8000h/0BFFFFh	736k/768k	CGA Display RAM (32k Bytes)[1]
0C0000h/0EFFFFh	768k/960k	Peripheral Drivers and Option ROMs[2]
0F0000h/0FFFFFh	960k/1M	ROM-BIOS (Low Memory)
100000h/F00000h	1M/15M	Extended (EXT) RAM/EMS RAM
FF0000h/FFFFFFh	15M/16M	ROM-BIOS (High Memory)

NOTES:

1. Up to 128k bytes of application ROM can be bank switched from either of two ROM sockets into the 128k byte Display RAM area. The ROM is configured 8-bits wide and its contents are biased toward the top of the available area between addresses A0000h and B8FFFh.

2. Expanded (EMS) memory for the Model 1520 Computer is bank switched into a portion of the area reserved for peripheral drivers and option ROMs. The 64k byte portion used for EMS is located from address E0000h through EFFFFh.

Application ROM Mapping

The computer supports up to 256k bytes of application ROM, which is contained in up to two ROM or EPROM Integrated Circuits. The ROMs

can be installed in sockets under a special cover, which is just above the keyboard Function keys (F1 - F10). Figure 2-1 shows the location of the ROM sockets. The ROMs are user installable.

The maximum application ROM size is supported with one 128k byte EPROM installed in each of the two sockets. The application ROMs are activated through a bank switching structure that maps one or more of the ROMs into a 512k byte block of memory address space that is normally used for Display RAM. The portion of system memory that is used for application ROM is located at address A0000h through BFFFFh. The ROM(s) are always biased toward the top of the available address space so that an 8k byte ROM would appear at addresses A0000h through A1F40h.

Only one ROM socket is mapped (active) at a time. The ROM is accessed by writing to the I/O Register at address 0440h. This register is eight bits wide and capable of selecting any one of 256 different ROMs.

When either ROM socket contains a ROM, the ROM Enable Register at I/O Address 405h must be set to "1" to enable the system address space for use by the ROM. When the ROM sockets are empty, the ROM Enable Register should be reset to "0" so that the 512k byte address space is available for use by system memory. The ROM Enable Register is set or reset though software commands. Refer to the paragraphs on ROM Sockets later in this chapter.

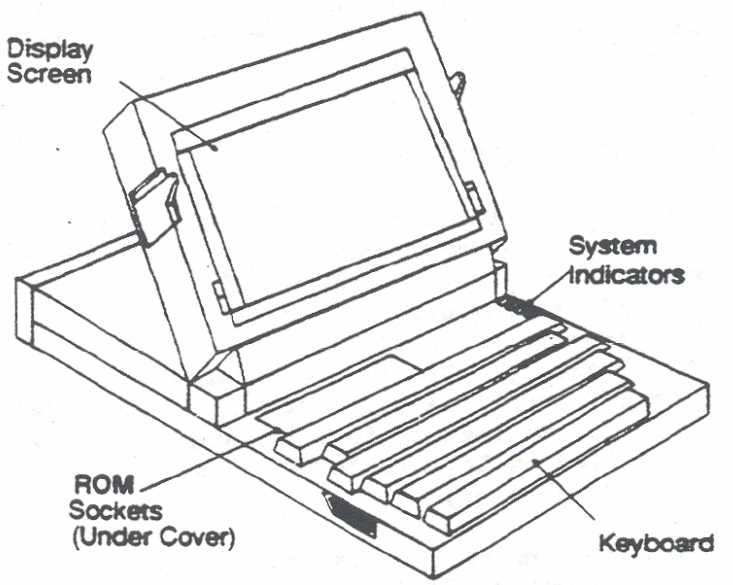


Figure 2-1. ROM Socket Locations

Display Subsystem Buffer Location

The 32k byte display subsystem buffer is separate from and in addition to the system memory, but is addressed by the same microprocessor as the system memory. The display subsystem buffer for CGA displays is located from address B8000h through BC000h. Additional address space is available for the optional EGA/VGA display, which can be connected through the computer back panel. This additional address space is located in the user memory area from address 0A0000h through 0AFFFFh. Detailed information on the Display Subsystem is contained in Chapter 6.

Peripheral Drivers and Option PROM Mapping

A block of up to 128k bytes in the microprocessor addressable space is reserved for peripheral drivers and option ROMs. Option ROMs are required by some external devices for programming control functions. Typical devices that require option ROMs are printer spoolers and networking interfaces. The address space is located from address D0000h through EFFFFh. A portion of this addressable space (E0000-EFFFF) can be used by EMS type memory also.

ROM-BIOS Memory Location

The 64k byte ROM-BIOS, bootstrap instructions, and other start-up data are contained in the addressable space from address F0000h through address FFFFFh. The ROM-BIOS contains service routines to support the MS-DOS Operating System and GRiD Developement software. A summary of the ROM-BIOS service routines is contained in Chapter 3 and more detailed descriptions are provided in the following chapters where the specific routines are described.

Expanded Memory Specification (EMS) Operation

The Expanded Memory Specification (EMS) operation provides a method for expanding the system memory addressing capability of GRiDCase 1500 Series computers that use the MS-DOS Operating System, since MS-DOS limits the addressing capacity to 640k bytes. In computers that use expanded memory, the system memory from 0 to 640k bytes is called conventional memory. System memory above 640k bytes is addressed via the EMS and is called expanded memory. For the Model 1520 Computer, EMS is implemented in a hardware/software configuration specified by Lotus/Intel/Microsoft (LIM). EMS is also referred to as LIM memory.

The Model 1530 Computer normally uses the AT type Extended memory (EXT) and not EMS. However, EMS can be emulated by the Model 1530 computer using available application software. The following paragraphs refer specifically to the implementation of EMS for the Model 1520 Computer.

EMS uses a bank switching structure where a portion of the system memory beyond the directly addressable 640k bytes is switched into a 64k byte addressable space that is reserved for peripheral drivers and special ROMs. The 64k byte portion used for bank switching Expanded Memory is located from address E0000h through EFFFFh.

Four I/O registers control bank switching of expanded memory pages into the addressable space of the microprocessor. The four registers each control one page (any 16k bytes) of expanded memory, and together they control four physical pages. The four physical pages form a 64k byte block, which is bank switched through the microprocessor addressable space from E0000h through EFFFFh. By bank switching the 16k byte pages of expanded memory, the conventional memory appears to be larger than 640k bytes.

The EMS I/O registers are 8 bits wide. Bit seven (MSB) of each register is an indicator for Page Enable (1) or Page Disable (0). The remaining seven bits (0 through 6) of each register represent any 16k byte page of Expanded Memory. The registers also handle both read and write operations. The I/O addresses and memory addresses of the registers are as follows:

Physical Page	I/O Address	Memory Address
0	0258h	E0000h - E3FFFh
1	4258h	E4000h - E7FFFh
2	8258h	E8000h - EBFFFh
3	C258h	EC000h - EFFFFh

SYSTEM POWER MODE CONTROL

The system power mode control provides a unique method of power conservation. Writing a "1" to I/O register 416h causes the microprocessor to run at a reduced speed, which reduces the power consumption. Also, the microprocessor clock rate is reduced by a power of 2. When I/O register 416h is reset to "0," the default state, the microprocessor runs at its normal speed. During operation, the microprocessor speed is normally controlled by a ROM-BIOS service routine through interrupt INT 15. The BIOS subsystem functions invoked by interrupt INT 15 are described in Chapter 3.

NOTE: Copy protected and older versions of some application programs may require a slower clock speed in order to operate effectively. Newer releases of the same programs will usually operate properly at the faster clock speed.

CHAPTER 3: ROM-BIOS AND I/O REGISTERS

The GRiDCase 1500 Series computer provides 32k bytes of System ROM. The System ROM contains the ROM-BIOS service routines that support the MS-DOS Operating System. The System ROM also provides low-level routines that support the InteGRiD Application Environment and GRiD Development software. System ROM is located in the 80286/80386 Microprocessor addressable space from address F8000h through address FFFFFh (low memory) and from address FF8000h through address FFFFFFh (high memory).

This chapter provides a summary of the ROM-BIOS service routines that support MS-DOS for the GRiDCase 1500 Series computer subsystems, and also provides a summary of the I/O registers used by the subsystems. The service routines and registers summarized in this chapter are described in more detail in the subsequent chapters that describe the subsystems.

The ROM-BIOS ensures compatibility with the IBM AT and also includes some extended service routines that are unique to the GRiDCase 1500 Series computer. The ROM-BIOS routines allow assembly language programmers to perform block or character level operations without knowing the programmed device address characteristics.

The I/O registers summarized in this chapter are used in the GRiDCase 1500 Series computer to interface with and control the operation of the computer subsystems.

In addition to the standard ROM-BIOS, there are GRiD Subsystem Functions, which provide a logical interface for manipulation of specific features of the GRiD Subsystem hardware. The GRiD Subsystem Functions are described later in this chapter.

ROM-BIOS SERVICE ROUTINES

Access to the ROM-BIOS service routines is provided via the 80286/80386 Microprocessor software interrupts in real-address mode. Each of the entry points to the ROM-BIOS is available through a specific interrupt vector. The interrupt service_routine to be executed is specified by the programmed interrupt number, and subroutines within the service routine are specified by the function number placed in register AH. (Register AH is the highest eight bits of the 16-bit 80286/80386 accumulator register AX.) The ROM-BIOS service routines save the contents of all microprocessor

registers except AX and the flags. The microprocessor register contents are modified only if they return a value to the caller.

The ROM-BIOS service routine interrupts for the GRiDCase 1500 Series computer are listed in Tables 3-1 and 3-2. Table 3-1 contains a list of the service routines by interrupt number and the applicable four-byte vector address. Other chapters that contain additional information are referenced where applicable. When the interrupt number is followed by an asterisk (*), the interrupt function numbers for specifying subroutines are listed in Table 3-2 for quick reference. For completeness, the non-BIOS software interrupts are listed in Table 3-3, but are not described further in this manual. In the tables, all the numerical values are given in hexadecimal (h).

Table 3-1. ROM-BIOS Service Routine Interrupts and Vector Addresses

Interrupt Number	Vector Address	Service Routine Description
00	00:03	Divide by Zero
01	04:07	Single Step
02	08:0B	Non-Maskable Interrupt (Chapter 3)
03	0C:0F	Breakpoint
04	10:13	Overflow
05	14:17	Print Screen (PrtSc) (Chapter 8)
06, 07	18:1F	Reserved
08	20:23	Time of Day, IRQ0, (Chapter 6)
09	24:27	Keyboard Interrupts, IRQ1, (Chapter 8)
0A	28:2B	Cascaded Interrupt Cont, IRQ2, (Chapter 5)
0B	2C:2F	Serial Port, COM2, IRQ3, (Chapter 12)
0C	30:33	Modem, COM1, IRQ4, (Chapter 12)
0D	34:37	Reserved, IRQ5
0E	38:3B	Floppy Disk Drive Cont, IRQ6, (Chapter 9)
0F	3C:3F	Parallel Printer Port, IRQ7, (Chapter 11)
10*	40:43	Display (Chapter 6)
11	44:47	Equipment Check (Chapter 3)
12	48:4B	Memory Size (Chapter 3)
13*	4C:4F	Floppy Disk/Hard Disk (Chapter 9/10)
14*	50:53	Serial I/O Port (Chapter 12)
15*	54:57	ROM-BIOS Extensions (Chapter 3)
16*	58:5B	Keyboard I/O (Chapter 8)
17*	5C:5F	Parallel Printer Port (Chapter 11)
18	60:63	Reserved
19	64:67	Bootstrap
1A*	68:6B	Time of Day (Chapter 6)
1B	6C:6F	Keyboard Break/Abort Handler (Chapter 3)
1C	70:73	Timer Tick Interrupt Handler (Chapter 3)
1D	74:77	Display Parameters (not used)
1E	78:7B	Floppy Disk Parameters (not used)
1F	7C:7F	Graphic Character Extensions (not used)
20-3F	80:FF	Non-BIOS Interrupt Functions (Table 3-3)

Table 3-2. ROM-BIOS Service Routine Interrupts and Subroutine Function Numbers

Interrupt Number	Function No. (AH Reg)	Service Routine Subfunction Description
Display Service Routines (Chapter 7)		
10	00	Set Display Mode
10	01	Set Cursor Type
10	02	Set Cursor Position
10	03	Read Cursor Position
10	05	Select Active Display Window (Alpha Mode)
10	06	Scroll Active Window Up
10	07	Scroll Active Window Down
10	08	Read Attribute/Character at Cursor
10	09	Write Attribute/Character at Cursor
10	0A	Write Character Only at Cursor
10	0B	Set Color Palette
10	0C	Write Pixel
10	0D	Read Pixel
10	0E	Write TTY to Active Window
10	0F	Return Current Display State
10	10-12	Reserved
10	13	Write Character String
Disk Drive Service Routines (Chapters 9 and 10)		
13	00	Reset Disk Drive Controller
13	01	Read Disk Drive Status from Last Operation
13	02	Read Data from Desired Sectors Into Memory
13	03	Write Data from Memory to Desired Sectors
13	04	Verify that Disk Sectors are Valid
13	05	Format Desired Track on Disk
13	06	Unused
13	07	Unused
13	08	Read Current Disk Drive Parameters
13	09	* Initialize Drive Pair Characteristics
13	0A	* Read Long
13	0B	* Write Long (* = for Hard Disk Only)
13	0C	* Seek
13	0D	* Alternate Disk Reset
13	0E	* Unused
13	0F	* Unused
13	10	* Test Drive Ready
13	11	* Recalibrate
13	12, 13	* Unused
13	14	* Controller Internal Diagnostic
13	15	Read DASD Type (@ = for Floppy Disk Only)
13	16	@ Disk Change Line Status
13	17	@ Set DASD Type for Format
13	18	@ Set Media Type for Format

Table 3-2. ROM-BIOS Service Routine Interrupts and Subroutine Function Numbers

Interrupt Number	Function No. (AH Reg)	Service Routine Subfunction Description
Serial I/O Port Service Routines (Chapter 12)		
14	00	Initialize Serial I/O Port
14	01	Send a Character to Serial I/O Port
14	02	Receive a Character from Serial I/O Port
14	03	Read Status of Serial I/O Port
BIOS Extensions (Chapter 3)		
15	80	Device Open
15	81	Device Closed
15	82	Program Termination
15	83	Event Wait
15	84	Joystick Support
15	85	System Request Key Pressed
15	86	Wait
15	87	Block Move
15	88	Determine Size of Extended Memory
15	89	Processor in Virtual Mode
15	90	Device Busy Loop
15	91	Set Flag for Interrupt Complete
15	C0	Return Configuration Parameters Pointer
15	E4	BIOS Subsystem Functions
Keyboard Service Routines (Chapter 8)		
16	00	Read an ASCII Character from Keyboard
16	01	Set Flag if ASCII Character Ready to Read
16	02	Read Shift Status from Keyboard
16	05	Place ASCII Character and Scan Code in Keyboard Buffer as if key was pressed.
16	10	Extended Read (00h) for Enhanced Keyboard
16	11	Extended Status for Enhanced Keyboard, (otherwise like 02h)
16	12	Read Extended Shift Status for Enhanced Keyboard
Printer (Parallel Port) Service Routines (Chapter 11)		
17	00	Send a Character to Printer
17	01	Initialize Printer Port
17	02	Store Printer Status in Register

Table 3-2. ROM-BIOS Service Routine Interrupts and Subroutine Function Numbers

Interrupt Number	Function No. (AH Reg)	Service Routine Subfunction Description
Time of Day Clock/Calendar Service Routines (Chapter 6)		
1A	00	Read Time of Day Clock Setting
1A	01	Set Time of Day Clock
1A	02	Read Real Time Clock
1A	03	Set Real Time Clock
1A	04	Read Date from Real Time Clock
1A	05	Set Date in Real Time Clock
1A	06	Set Alarm to Interrupt at Specified Time
1A	07	Reset the Alarm Interrupt Function

Table 3-3. Software (non-BIOS) Interrupts and Vector Addresses

Interrupt Number	Vector Address	Interrupt Function
20	80:83	DOS Program Terminate
21	84:87	DOS Function Call
22	99:8B	DOS Terminate Address
23	8C:8F	DOS Control Break Exit Address
24	90:93	DOS Fatal Error Vector
25	94:97	DOS Absolute Disk Read
26	98:9B	DOS Absolute Disk Write
27	9C:9F	DOS Terminate, Fix in Storage

NOTE: Throughout this chapter, numerical values are given in hexadecimal. The hexadecimal values have an "h" suffix unless otherwise specified.

ROM-BIOS SPECIAL FUNCTION INTERRUPTS

The ROM-BIOS service routine interrupts listed in Table 3-1 include several interrupts with special functions. These special function interrupts are given in the following list. The list is then followed by brief descriptions of the special functions. The special function interrupts are:

Interrupt No. (Hex)	Function
11, 12	System Configuration
15	BIOS Extensions/GRiD Subsystem
1B	Keyboard Break/Abort Handler
1C	Timer Tick Interrupt Handler
1D	Display Parameters
1E	Floppy Disk Parameters
1F	Graphic Character Extensions
40	Reserved (Floppy and Hard Disk)
41, 46	Hard Disk Parameters

System Configuration Service Routines (INT 11, 12)

Interrupt service routines 11h and 12h return information that describes the equipment and memory available in the system.

The equipment check service routine vector (Int 11h, AH = 00) returns the following information in register AX:

Bit(s)	Description
15, 14	Number of Parallel Printers Attached
13	Unused (always 0)
12	Unused (always 0)
11, 10, 9	Number of Asynchronous Communication Ports
8	Unused (always 0)
7, 6	Number of Floppy Disk Drives Attached to System: 00 = 1 Drive 01 = 2 Drives 10 = 3 Drives 11 = Reserved
5, 4	Video Mode: 00 = Unused 01 = 40x25 Black and White with Color Mode 10 = 80x25 Black and White with Color Mode 11 = 80x25 Monochrome Mode
3, 2	Unused (always 0)
1	Math Coprocessor 0 = No Coprocessor, 1 = Coprocessor installed
0	Floppy Disk Drive (FDD) identifier: 0 = No FDD, 1 = FDD(s) installed

The memory size service routine vector (Int 12h, AH = 00) determines the number of contiguous 1k blocks of memory (up to 640k) that are in the system. The number is based on the Power On Self Test (POST) performed during the power turn ON. The number of contiguous 1k blocks of memory is returned in register AX.

BIOS Extensions/GRiD Subsystem (INT 15)

Interrupt service routine 15h was previously used for a cassette interface. However, in the GRiDCase 1500 Series computers, interrupt 15h is used for two sets of special functions. When interrupt 15h is used with function numbers 80-91h or C0h in the AH register, it invokes IBM AT compatible ROM-BIOS extensions. The ROM-BIOS extensions are listed in the Table 3-4. Interrupt 15h with function E4h in the AH register invokes special GRiD subsystem functions. The special GRiD Subsystem functions are described separately under BIOS Subsystem Functions later in this chapter. In Table 3-4, all numerical values are given in hexadecimal.

Table 3-4. ROM-BIOS Extensions with Interrupt 15h

Function No. (AH Reg)	Extension/Description	
80	Device Open	
	BX = Device ID	
	CX = Process ID	
81	Device Close	
	BX = Device ID	
	CX = Process ID	
82	Program Termination	
	BX = Device ID	
83	Event Wait	
	AL = 00h	Set Interval
	CX,DX = ms	Number of microseconds to lapse before posting.
	ES:DX = p	Pointer to a byte in user memory that has the high order bit set soon after the interval expires.
	AL = 01h	Cancel wait
84	Joystick Support	
	Not Used	
85	Emulate System Request (Sys Req) Key	
	AL = 00h	Key Pressed (make)
	AL = 01h	Key Released (break)

Table 3-4. ROM-BIOS Extensions with Interrupt 15h (Continued)

Function No. (AH Reg)	Extension/Description
86	Wait CX,DX = us Number of microseconds to lapse before return to user.
87	Block Move Refer to the following paragraph on block move.
88	Determine Extended Memory Size Refer to the following paragraph on Extended Memory Size.
89	Processor to Virtual Mode Refer to the following paragraph on Processor to Virtual Mode.
90	Device Busy Loop AL = Device Code*
91	Set Interrupt Complete Flag AL = Device Code*

*Device Codes for Functions 90 and 91.

Device Code	Description
00-7F	Serial reusable devices where access is serialized by the operating system. Examples: 00 = Hard Disk (Requires Timeout) 01 = Floppy Disk (Requires Timeout) 02 = Keyboard (No Timeout)
80-BF	Reentrant devices. Where simultaneous I/O calls are allowed, ES:BX is used to distinguish the different calls. Example: 80 = Network (No Timeout) ES:BX = NCB
C0-FF	Wait only calls. These are timeouts only and times are function dependent. Examples: FD = Floppy Disk Motor Start FE = Printer

C0	Configuration Parameters Pointer AH = 00h and Carry Flag = 0 if valid ES:BX = Parameter table address pointer AH = 86h and Carry Flag = 1 if not valid

Block Move (AH = 87h)

Block move allows the program in real mode to move data to or from memory locations above the 1M byte address range by switching to protected mode. A Global Descriptor Table (GDT) must be built by the user before issuing interrupt 15h with function 87h. The descriptors are used to perform the block move in protected mode. The segment length for the source and target descriptors must be two times the word count minus one (2xCX-1) or greater. The 24-bit address (byte high, word low) is set to the target/source. No interrupts are allowed during transfer. Upon completion all registers are restored except AH.

NOTE: To avoid timing conflicts, do not perform block moves during serial communications.

AH = 87h Block Move
CX = Word count for block to be moved. The maximum count is 8000h for 32k words (64k bytes).
ES:SI = Location of GDT

On Exit:
AH = 00h Successful Block Move
AH = 01h Parity Error Registers Cleared (Memory Parity)
AH = 02h Another Exception Interrupt Error Occurred
AH = 03h Gate Address Line 20 failed

Carry Flag = 0 for successful block move
Carry Flag = 1 for error during block move

Definition of Block Move GDT

The following paragraphs describe the block move Global Descriptor Table (GDT) that is shown in Figure 3-1. The first four descriptors are initialized by the user.

1. The first descriptor is a required dummy, which is initialized at 0.

2. The second descriptor points to the GDT as a data segment. It is initialized to 0 and then modified by the BIOS.

3. The third descriptor is initialized to point to the source to be moved (from).

4. The fourth descriptor is initialized to point to the destination (to).

5. The fifth descriptor is used by the BIOS to create a Protected Mode code segment. It is initialized to 0 by the user and then modified by the BIOS.

6. The sixth descriptor is used by the BIOS to create a Protected Mode stack segment. It is initialized to 0 and then modified by the BIOS to point to the user stack.

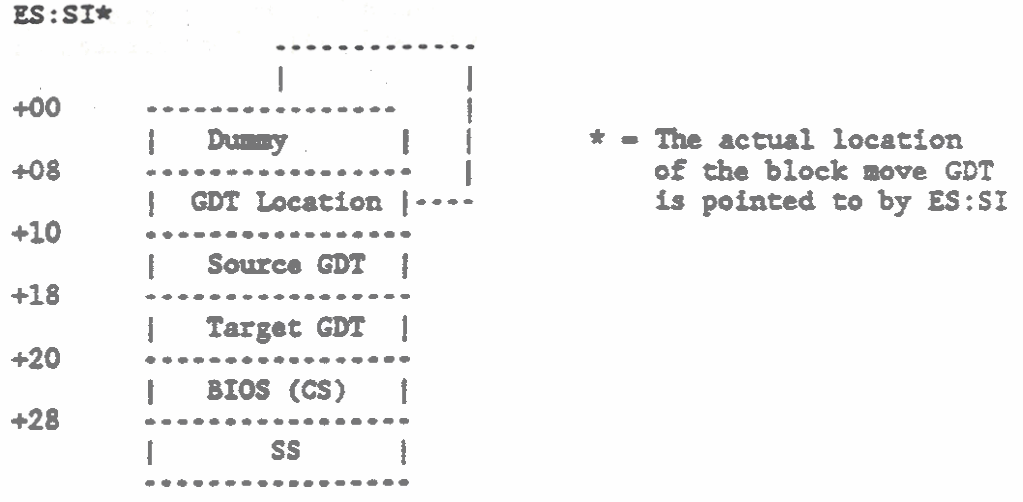


Figure 3-1. Block Move Global Descriptor Table (GDT)

Procedure:

The following steps describe the operations required to perform a block move function.

1. Save entry registers and setup for shutdown exit.

2. Build required entries in GDT at ES:SI.

3. Set address line 20 active, Clear Interrupts (CLI), and set shutdown codes.

4. Load an Interrupt Descriptor Table Register (IDTR) to point to ROM resident table.

5. Load Global Descriptor Table Register (GDTR) from the offset pointer (ES:SI).

6. Set microprocessor to protected mode (refer to the following procedure for function 89h)

7. Load DS and ES registers with selectors for the source and target, respectively.

8. Execute REP MOVSW DS:SI (Source) ES:DI (target)

9. Check for parity errors.

10. Restore Real Mode when Shutdown (09h) is executed.

11. Check for errors and set return codes for AH.

12. Clear address line 20 active (disable).

13. Return with registers restored and status return code. (ZF = 1 if ok, ZF = 0 if error).

Extended Memory Size (AH = 88h)

Extended memory size allows the user to determine the size of memory available starting at the 1M byte address range. This size is stored at memory locations 40:30h and 40:31h during the Power On Self Test (POST). The size is based on the following assumptions:

1. All installed memory is functional.

2. All memory from 0 to 640k is contiguous.

AH = 88h Determine Memory Size

On Exit:
AX = nn The number of contiguous 1k blocks of available memory starting at 1M byte.

Processor to Virtual Mode (AH = 89h)

Processor to virtual mode allows the user to switch into virtual (protected) mode. When the function is complete, the microprocessor is in virtual (protected) mode and control is transferred to the code segment specified by the user. A Global Descriptor Table (GDT) must be built by the user before issuing interrupt 15h with function 89h. The descriptors are used to initialize the IDTR, GDTR, and Stack Segment selector. The Data Segment (DS) and Extra Segment (ES) selectors are also initialized to descriptors built by the user.

Register BH contains the offset into the Interrupt Descriptor Table (IDT) to where the first group of eight interrupts start. Register BL contains the offset into the IDT to where the second group of eight interrupts start.

AH = 89h Processor To Virtual (Protected) Mode
ES:SI = Location of GDT
On Exit:
AH = 00h Successful switch to Protected Mode

NOTE: The contents of all segment registers are changed and the contents of registers AX and BP are destroyed.

Definition of Processor to Virtual Mode GDT

The following paragraphs describe the Processor to Virtual Mode Global Descriptor Table (GDT) that is shown in Figure 3-2. The seven descriptors initialized by the user must contain all necessary data including the limit, base address, and the access rights byte.

1. The first descriptor is a required dummy, which is initialized at 0.

2. The second descriptor is initialized to point to the GDT as a data segment.

3. The third descriptor is initialized to point to the user defined Interrupt Descriptor Table (IDT).

4. The fourth descriptor is initialized to point to the user's Data Segment (DS).

5. The fifth descriptor is initialized to point to the user's Extra Segment (ES).

6. The sixth descriptor is initialized to point to the user's Stack Segment (SS).

7. The seventh descriptor is initialized to point to the user's Code Segment (CS), which is the Code Segment this function returns to.

8. The eighth descriptor is used by the BIOS to establish a Code Segment for its own use. It is required so that this function can complete its execution while in protected mode.

Considerations:

The following steps provide information that should be considered before performing the procedure to switch the processor from Real Mode to Virtual (Protected) Mode.

1. The ROM-BIOS is not available to the user. Therefore, the user must handle all I/O commands.

2. The hardware interrupt controller must be reinitialized to define new locations that are outside the processor reserved locations.

3. The Exception Interrupt Table and Handler must be initialized by the user.

4. The interrupt descriptor table cannot overlap the Real Mode BIOS Interrupt Descriptor Table.

5. The following list provides an example of the user's code to invoke the Processor to Virtual (Protected) Mode function:

```
"Real Mode User's Code"
   MOV   AX,GDT Segment
   MOV   ES,AX
   MOV   SI,GDT Offset
   MOV   BH, Hardware Interrupt Level 1 Offset
   MOV   BL, Hardware Interrupt Level 2 Offset
   MOV   AH,89h
   INT   15h
"Virtual Mode User's Code"
```

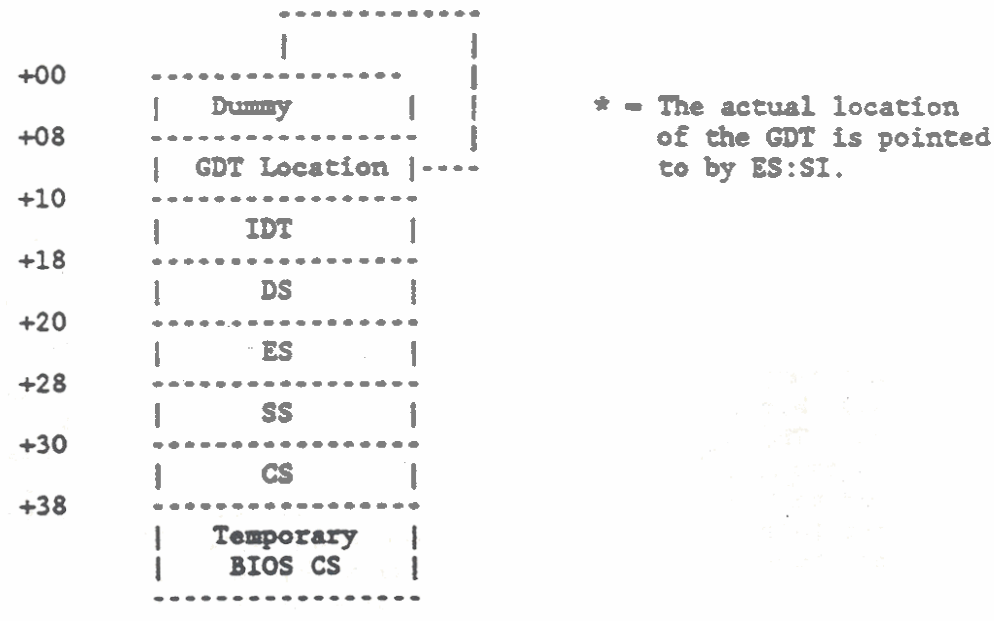


Figure 3-2. Processor to Virtual (Protected) Mode Global Descriptor Table (GDT)

Procedure:

The following steps describe the operations required to perform a Processor to Virtual (Protected) Mode function. When the function is successfully completed, the user's Code Segment (CS) descriptor value is substituted on the stack for a return to the user. Control is then transferred to the user with interrupts disabled.

1. Build required entries in GDT at ES:SI.

2. Set address line 20 active.

3. Clear Interrupts (CLI). No interrupts are allowed during execution of this function.

4. Load GDTR with the GDT base address.

5. Load IDTR with IDT base address.

6. Reinitialize the hardware interrupt controller with the new interrupt offsets.

7. Initialize the current user Stack Segment (SS) descriptor.

8. Put the processor in Virtual Mode with the Code Segment (CS) designated for this function.

9. Load the Data Segment (DC) with the user defined selector for the DS register.

10. Load the Extra Segment (ES) with the user defined selector for the ES register.

11. Load the Stack Segment (SS) with the user defined selector for the SS register.

Keyboard Break/Abort Handler (INT 1B)

The interrupt service routine pointed to by interrupt 1Bh is executed when the Ctrl, Break, and -Fn- keys are simultaneously pressed. The microprocessor services interrupt 1B while responding to the normal keyboard interrupt (09h), and control is then returned to the user through an IRET instruction. The power-on routines initialize interrupt 1Bh to point to an IRET instruction. The operating system (MS-DOS) then uses this interrupt vector to abort programs when the Ctrl and Break keys are pressed.

To regain control following a Break, consider the following possibilities:

1. The Break may have occurred during interrupt processing, which requires that one or more End of Interrupt commands be sent to the hardware interrupt controller.

2. The Break may have occurred during execution of an I/O operation. Therefore, all I/O devices should be reset.

Timer Tick Interrupt Handler (INT 1Ch)

The interrupt service routine pointed to by interrupt 1Ch is executed at every tick of the system clock. The microprocessor services interrupt 1Ch while responding to the normal timer interrupt (08h), and control is then returned to the user through an IRET instruction. The power-on routines initialize interrupt 1Ch to point to an IRET instruction. Therefore, no operations occur unless the application program modifies the pointer. The application program must also save and restore all registers that are modified. When this interrupt passes control to an application program, all interrupts from the hardware interrupt controller are disabled.

Display Parameters (INT 1Dh)

The power-on routines initialize interrupt 1Dh to point to display parameters contained in the Display RAM. These parameters are not used by the GRiDCase 1500 Series computers.

Floppy Disk Parameters (INT 1Eh)

The power-on routines initialize interrupt 1Eh to point to Floppy Disk parameters contained in System ROM. These default parameters provide specific values for any IBM-compatible drive connected to the system. The parameters are not used by the GRiDCase 1500 Series computers.

Graphic Character Extensions (INT 1Fh)

The power-on routines initialize interrupt 1Fh to 000:0. This vector can be changed by the user to point to a table of up to 1k bytes. The table is then used to hold graphic character extension information, which consists of up to 128 code points with 8 bytes of graphic information for each point.

Floppy Disk Drive Pointer (INT 40h)

Interrupt 40h provides a service routine to revector the floppy disk drive pointer. This revectoring is required in systems that use both hard disk and floppy disk drives.

Hard Disk Parameters (INT 41h, 46h)

The interrupt service routines pointed to by interrupts 41h and 46h are executed during the hard disk drive interrupt service routine used for initializing the drive characteristics (INT 13, Function 09h). Interrupt 41h points to a data block for drive 0 (DL register contains 80h) and interrupt 46h points to a data block for drive 1 (DL register contains 81h). Chapter 10 provides additional information on hard disk drive operation.

BIOS SUBSYSTEM FUNCTIONS

BIOS subsystem functions are a special case of service routines for Interrupt 15h. These subsystem functions were added to the ROM-BIOS to provide a logical interface for manipulation of GRiD hardware specific features. The subsystem functions are divided into eight groups. However, only the first three groups are currently defined as follows:

1. ROM Subsystem Functions (00h - 1Fh)

2. Display Subsystem Functions (20h - 3Fh)

3. Configuration Subsystem Functions (40h - 5Fh)

To perform a subsystem function, an interrupt 15h is performed with the subsystem function identifier, 0E4h, in the AH register and the function number in the AL register. The combination of interrupt 15h and the subsystem function identifier 0E4h provides access to the BIOS subsystem functions. Once inside the subsystem, the specific function to be performed is selected via the function number in the AL register.

After the selected function is performed, the carry flag is cleared (0) or set (1). If the carry flag is cleared, the operation was successful. When the carry flag is set, an error has occurred and an error number is returned to the AH register. If an error number 86h is returned, the selected subsystem function is not supported by the system hardware.

Use of the other registers and additional error numbers are described in the following paragraphs.

ROM Subsystem Functions (00h - 1Fh)

The ROM Subsystem Functions are identified by numbers 00h through 1Fh. Functions 00h through 04h are currently defined as given in

the following list and described in the subsequent paragraphs. Functions 05h through 1Fh are reserved.

Function No.	Function Name
00h	Get ROM Subsystem Information
01h	Initialize ROM Subsystem
02h	Get ROM Image Information
03h	Enable a ROM Image
04h	Disable ROM Subsystem
05h-1Fh	Reserved

Get ROM Subsystem Information (Function 00h)

Function 00h returns preliminary ROM information, which includes the number of hardware ROM slots and the size in paragraphs of the data space that must be provided when calling all other functions.

```
Entry:
     AH = 0E4h      Selects the ROM Subsystem
     AL = 00h       ROM Subsystem Function number
Exit:
Carry Flag = 0 (no error)
     AL = Number of hardware ROM slots
     BX = Number of paragraphs needed for data area
Carry Flag = 1 (error returned)
     AH = 86h       Subsystem not supported
```

Initialize ROM Subsystem (Function 01h)

Function 01h initializes the ROM subsystem. Each physical ROM slot is tested for application ROM ICs. When two or more ROM ICs make up one logical ROM image, the driver enters this data in its ROM table. The function then returns the number of logical ROM images.

```
Entry:
     AH = 0E4h      Selects the ROM Subsystem
     AL = 01h       ROM Subsystem Function number
     DS = xx        The data segment for the ROM data area
                    is provided by the calling application.
                    The data segment must be as large as the
                    value specified by function 00h.
Exit:
Carry Flag = 0 (no error)
     AL = Number of logical ROM images
Carry Flag = 1 (error returned)
     AH = 86h       Subsystem not supported
        = 2         Passed data segment is invalid
        = 3         No ROM images exist
```

Get ROM Image Information (Function 02h)

Function 02h returns information about the specified ROM image. Any ROM image number can be entered or 0FFh can be entered to determine the information about the current ROM.

Entry:

AH = 0E4h	Selects the ROM Subsystem
AL = 02h	ROM Subsystem Function number
DL = xx	Enter 0FFh for the current ROM image or enter any other ROM image number
DS = nn	The data segment for the ROM data area is provided by the calling application. The data segment must be as large as the value specified by function 00h.

Exit:

Carry Flag = 0 (no error)

- AL = Total number of ROM images
- BX = ROM ID
- CX = Number of paragraphs in ROM image
- DH = System type
- DL = ROM image number
- ES = ROM base segment

Carry Flag = 1 (error returned)

AH = 86h	Subsystem not supported
= 2	Passed data segment is invalid
= 3	No ROM image exists
= 4	Image number out of range

Enable a ROM Image (Function 03h)

Function 03h enables the selected ROM image and returns a pointer to the beginning of the image. This function also returns the size of the ROM image in paragraphs.

Entry:

AH = 0E4h	Selects the ROM Subsystem
AL = 03h	ROM Subsystem Function number
DL = xx	Enter 0FFh for the current ROM image or enter any other ROM image number
DS = nn	The data segment for the ROM data area is provided by the calling application. The data segment must be as large as the value specified by function 00h.

Exit:

Carry Flag = 0 (no error)

ES:BX =	Pointer to the start of the ROM image
CX =	Number of paragraphs in ROM image

Carry Flag = 1 (error returned)

AH = 86h	Subsystem not supported
= 2	Passed data segment is invalid
= 3	No ROM image exists
= 4	Image number out of range

Disable ROM Subsystem (Function 04h)

Function 04h disables the ROM subsystem by clearing the current ROM image.

NOTE: This function does not turn on interrupts after clearing the current ROM image. The calling procedure must determine whether or not to turn ON the interrupts.

Entry:

AH = 0E4h	Selects the ROM Subsystem
AL = 04h	ROM Subsystem Function number
DS = nn	The data segment for the ROM data area is provided by the calling application. The data segment must be as large as the value specified by function 00h.

Exit:

Carry Flag = 0 (no error)

No other significant return

Carry Flag = 1 (error returned)

AH = 86h	Subsystem not supported
= 2	Passed data segment is invalid
= 3	No ROM image exists

Display Subsystem Functions (20h - 3Fh)

The Display Subsystem Functions are identified by numbers 20h through 3Fh. Functions 20h through 23h are currently defined as given in the following list and described in the subsequent paragraphs. Functions 24h through 3Fh are reserved.

Function No.	Function Name
20h	Select or get current display
21h	Select or get color map
22h	Select current display font
23h	Set backlight time out
24h-3F	Reserved

Select or Get Current Display (Function 20h)

Function 20h selects either the internal display panel or the external RGB video port, or returns the currently active display. The value entered in DL register determines the display to be selected. When value 0FFh is entered in the DL register, the current display is returned. The value returned for the internal panel is the panel type.

Entry:

AH = 0E4h		Selects the ROM Subsystem
AL = 20h		Display Subsystem Function number
DL = 0		Selects the external RGB video port
= 1		Selects the internal panel
= 0FFh		Returns the display type

Exit:

Carry Flag = 0 (no error)

If 0FFh was entered in DL, then:

DL = 0	External Display port is selected
DL = 1	Internal 640x200 plasma panel selected
DL = 5	Internal 640x400 plasma panel selected
DL = 7	Internal 640x400 LCD panel selected

Carry Flag = 1 (error returned)

AH = 86h	Subsystem not supported

NOTE: The internal display is also controlled by the MS-DOS MODE commands DISPLAY = {ON|OFF}

Select or Get Color Map (Function 21h)

Function 21h is used to select one of six predefined color maps and is also used to activate the currently selected color map. The color map index is stored at absolute address 40:A7h. The value entered in the DL register determines which color map is selected. When the value 0FFh is entered in the DL register, the currently selected color map is activated. The six predefined color maps are as follows:

NOTE: MAP 0 through MAP 5 in the following list correspond to MS-DOS MODE commands COLORMAP = 1-6, respectively.

Color	Map 0	Map 1	Map 2	Map 3	*Map 4	*Map 5
Black	Off	Off	Off	Off	Off	Off
Blue	Off	2/3	2/3	Off	Full	Full
Green	2/3	2/3	1/3	1/3	Full	Full/ul
Cyan	2/3	2/3	1/3	1/3	Full	Full
Red	1/3	1/3	2/3	2/3	Full	Full
Magenta	1/3	1/3	2/3	2/3	Full	Full
Brown	Full	Full	Full	Full	Full	Full
White	Full	Full	Full	Full	Full	Full

NOTES: The color maps are defined in terms of the gray scale as follows:

Off	= nothing displayed
1/3	= one third bright
2/3	= two thirds bright
Full	= full brightness
/ul	= underlined
*	= inverse video is displayed when the background color is white

```
Entry:
     AH = 0E4h       Selects the ROM Subsystem
     AL = 21h        Display Subsystem Function number
     DL = nn         Enter 0 - 5 to select the color map
                     or enter 0FFh to activate the currently
                     selected color map.
Exit:.
Carry Flag = 0 (no error)
     If 0FFh was entered in DL, then:
     DL = Currently active color map
Carry Flag = 1 (error returned)
     AH = 86h        Subsystem not supported
        = 2          Color map value out of range
```

Select Current Display Font (Function 22h)

Function 22h selects one of the four fonts that are available in the hardware font generator. The four available fonts and their selection values are as follows:

Font Name	Value
English	0
French Canadian	1
Norwegian	2
Hebrew	3

NOTE: Font values 0-3 correspond to MS-DOS MODE commands FONT = 1-4, respectively.

```
Entry:
     AH = 0E4h       Selects the ROM Subsystem
     AL = 22h        Display Subsystem Function number
     DL = nn         Enter 0 - 3 to select the font number
Exit:
Carry Flag = 0 (no error)
     No other significant return
Carry Flag = 1 (error returned)
     AH = 86h        Subsystem not supported
        = 2          Font value out of range
```

Set Backlight Time Out (Function 23h)

Function 23h allows the backlight on an internal LCD display panel to be turned ON and OFF directly or turned on and timed to turn off after some period of keyboard inactivity. This function has no effect on a plasma display.

Entry:

AH = 0E4h		Selects the ROM Subsystem
AL = 23h		Display Subsystem Function number
DL = 0		Turns OFF the backlight
= 1-60		Turns ON the backlight and starts the timer to turn it OFF if the keyboard is inactive for a period of 1 to 60 (decimal) minutes.
DL = 0FFh		Turn ON backlight with no time out.

NOTE: The DL register values correspond to MS-DOS MODE commands BACKLITE = {1-60|ON|OFF}, respectively.

Exit:

Carry Flag = 0 (no error)

No significant return

Carry Flag = 1 (error returned)

AH = 86h	Subsystem not supported

Configuration Subsystem Functions (40h - 5Fh)

The Configuration Subsystem Functions are identified by numbers 40h through 5Fh. Functions 40h through 46h and 49h are currently defined as given in the following list and described in the subsequent paragraphs. Functions 47h, 48h, and 4Ah through 5Fh are reserved.

Function No.	Function Name
40h	Get EMS RAM size
41h	Get Extended RAM size
42h	Select EMS RAM or Extented RAM
43h	Return external floppy drive number
44h	Check for Pouch Tape Drive
45h	Select microprocessor speed
46h	Battery low power check
47h, 48h	Reserved
49h	Hard Disk spin-down mode
4Ah-5Fh	Reserved

Get EMS RAM Size (Function 40h)

Function 40h returns the size in kilobytes of the Expanded Memory Specification (EMS) RAM present in the computer.

Entry:

AH = 0E4h		Selects the ROM Subsystem
AL = 40h		Configuration Subsystem Function number

```
Exit:
Carry Flag = 0 (no error)
     CX = nn         Size of EMS RAM in kilobytes
Carry Flag = 1 (error returned)
     AH = 86h        Subsystem not supported
```

Get Extended RAM Size (Function 41h)

Function 41h returns the size in kilobytes of the AT type Extended RAM present in the computer.

```
Entry:
     AH = 0E4h       Selects the ROM Subsystem
     AL = 41h        Configuration Subsystem Function number
Exit:
Carry Flag = 0 (no error)
     CX = nn         Size of Extended RAM in kilobytes
Carry Flag = 1 (error returned)
     AH = 86h        Subsystem not supported
```

Select EMS RAM or Extended RAM (Function 42h)

Function 42h selects either EMS or Extended RAM to be used as extra memory for the Model 1520 (80286 Microprocessor controlled) computer. The computer must be reinitialized (booted) for any change in this configuration to take effect. This function is also used to determine which type of extra memory is currently selected.

```
Entry:
     AH = 0E4h       Selects the ROM Subsystem
     AL = 42h        Configuration Subsystem Function number
     DL = 0          Selects Extended RAM
        = 1          Selects EMS RAM
        = 0FFh       Returns the current extra memory type.
Exit:
Carry Flag = 0 (no error)
     If 0FFh was entered in DL, then:
     DL = 0    Extended RAM selected
     DL = 1    EMS RAM Selected
Carry Flag = 1 (error returned)
     AH = 86h        Subsystem not supported
```

NOTE: The extra RAM type (EMS or Extended) selected via the DL register corresponds to the MS-DOS MODE commands MEM = {EMS|EXT}, respectively.

Return External Floppy Drive Number (Function 43h)

Function 43h returns the ROM-BIOS drive number for the External Floppy Disk Drive if the drive is installed in the system.

```
Entry:
     AH = 0E4h       Selects the ROM Subsystem
     AL = 43h        Configuration Subsystem Function number
Exit:
Carry Flag = 0 (no error)
     DL = nn         External Floppy ROM-BIOS drive number.
Carry Flag = 1 (error returned)
     AH = 86h        Subsystem not supported
        = 2          External Floppy drive not installed
```

Check for Pouch Tape Drive (Function 44h)

Function 44h tests to determine if there is a Pouch Tape Drive installed in the computer.

```
Entry:
     AH = 0E4h       Selects the ROM Subsystem
     AL = 44h        Configuration Subsystem Function number
Exit:
Carry Flag = 0 (no error)
     Pouch Tape Drive is installed
Carry Flag = 1 (error returned)
     AH = 86h        Subsystem not supported
        = 2          Pouch Tape Drive is not installed.
```

Select Microprocessor Speed (Function 45h)

Function 45h allows the Microprocessor speed to be switched between fast and slow speeds.

```
Entry:
     AH = 0E4h       Selects the ROM Subsystem
     AL = 45h        Configuration Subsystem Function number
     DL = 0          Selects slow speed
        = 1          Selects fast speed
Exit:
Carry Flag = 0 (no error)
     No significant return
Carry Flag = 1 (error returned)
     AH = 86h        Subsystem not supported
```

NOTE: The DL register value corresponds to the MS-DOS MODE command SPEED = {SLOW|FAST}, respectively.

Battery Low Power Check (46h)

Function 46h provides a low power check for the battery.

```
Entry:
     AH = 0E4h       Selects the ROM Subsystem
     AL = 46h        Configuration Subsystem Function number
Exit:
Carry Flag = 0 (no error)
     DL = 0          Battery power is within tolerance
        = 1          Batter power is low
Carry Flag = 1 (error returned)
     AH = 86h        Subsystem not supported
```

Hard Disk Spin-Down Mode (49h)

A hard disk spin-down mode is supported for low-power IBM AT compatible hard disk drives only (refer to Chapter 10). This mode is not supported for the IBM XT compatible hard disk drives or the high-power IBM AT compatible hard disk drives. If the spin-down mode is not supported on the installed hard disk drive, an error indication is returned.

When a Hard Disk Drive spins down, it enters standby mode and its power consumption is reduced to 0.5 Watts. Normal power consuption for these drives during operation is 2 Watts for the 20M byte drive and 2.5 Watts for the 40M byte drive. Since extra power is consumed when the drive is subsequently spun up, the drives must remain spun down for some period of time in order to actually conserve power. For example, the 40M byte drive should remain spun down for 90 seconds in order to conserve the same amount of power that will be used when the drive is subsequently spun up.

The 100M byte drive does not support the spin-down command because, due to its mass, it would consume too much power when it was subequently spun up.

```
Entry:
     AH = 0E4h       Selects the ROM Subsystem
     AL = 49h        Configuration Subsystem Function number
     DH = 0FFh       Refer to Exit.
     DH = 0          Turn OFF drive motor immediately if DL is
                     greater than 0 and less then 0FFh.
     DH = 1          Turn ON drive motor immediately if DL is
                     greater than 0 and less then 0FFh.
     DH <> 0FFh      Selects DL register contents
     DL = 0          Turn OFF drive motor immediately
     DL = 0FFh       Turn ON drive motor immediately
     DL = 1-0FEh     Turn OFF drive motor after n seconds of
                     inactivity, where n = DL*5.
```

Exit:	
Carry Flag = 0	Successful operation with no error
DH = 0FFh	On entry, then:
CX = x	Contains the maximum number of 5-second intervals
DX = y	Contains the minimum number of 5-second intervals
BX = z	The z indicates the estimated number of starts/stops in the lifetime of the drive/1000.
DH <> 0FFh	All registers except AX are preserved.
Carry Flag = 1	(error returned)
AH = 86h	Subsystem is not supported for this drive.
AH = Err	Where Err is any standard BIOS hard disk drive error code.

SYSTEM IDENTIFICATION

The GRiDCase 1500 Series Computers provide two locations in the ROM-BIOS that contain system identification information. The first location identifies the computer as a GRiD Systems computer, and the second location provides a value that indicate the system model number. The ROM-BIOS locations for these values are as follows:

Location	Value	Description
0FDFDh:000Ch	2D2Dh	Identifies GRiD Systems as the manufacturer
0F000h:0DFFEh	74h	GRiDCase 1530 (80386 Microprocessor)
	34h	GRiDCase 1520 (80286 Microprocessor)
	14h	GRiDLite Plus (8086 Microprocessor)
	8	GRiDLite
	4	GRiDCase Plus (1200 Series)
	3	Tempest GRiDCase (1300 Series)
	0	GRiDCase (1200 Series)

I/O DMA CHANNEL ASSIGNMENTS

The GRiDCase 1500 Series computer uses a Faraday 3010 Peripheral Controller that emulates two cascaded 8237A-5 DMA Controllers. The DMA Controller provides up to seven DMA channels. An eighth DMA channel (DMA Channel 4) is used to cascade the master and slave controller channels and therefore cannot be used for DMA data transfers. The eight DMA channels are assigned as follows:

NOTE: For additional DMA Controller information, refer to the Chapter 4.

DMA Channel	I/O Address	(M)aster/ (S)lave	System Function
0	87h	M	Spare (Highest Priority)
1	83h	M	Reserved
2	81h	M	Floppy Disk Drive Controller
3	82h	M	Hard Disk Drive Controller
4	--	/	Cascade to Master Controller
5	8Bh	S	Spare
6	89h	S	Spare
7	8Ah	S	Spare (Lowest Priority)
Refresh	8Fh	/	

Master Controller Channels 0-3 support 8-bit to 8-bit data transfers between I/O Adapters and memory locations. Each channel transfers data in up to 64k byte blocks throughout the 16M byte system address space. Addresses for DMA channels 0-3 are generated by concatenating the I/O (page) address to the address for the master DMA Controller (00h - 1Fh). The total address length is 24 bits.

Slave Controller Channels 5-7 support 16-bit transfers between 16-bit I/O Adapters and memory locations. These channels cannot transfer data on odd byte boundaries. Each channel transfers data in up to 128k byte blocks throughout the 16M byte system address space. Addresses for DMA channels 5-7 are generated by concatenating the high order 7-bits of the I/O (page) address to the address for the slave DMA Controller (C0h - DFh). The total address length is 24 bits.

HARDWARE INTERRUPT ASSIGNMENTS

The GRiDCase 1500 Series computer uses the Non-Maskable Interrupt (NMI) and a Faraday 3010 Peripheral Controller. The 3010 Peripheral Controller provides logic that emulates two cascaded 8259A Programmable Interrupt Controllers (PICs) to provide 15 levels of vectored priority interrupts for the 80286/80386 Microprocessor. A sixteenth interrupt level (IRQ2) is used to cascade the Master and Slave interrupt levels and therefore cannot be used to provide a separate interrupt. Along with NMI, 16 interrupt levels are provided.

NMI Interrupt (INT 02h)

When power is initially applied to the computer, the non-maskable interrupt (NMI) is masked off. The system program then sets or resets the NMI mask bit as follows:

1. To enable NMI, MASK ON is set by writing to I/O address 070h, with bit 7 set to 0.

2. To subsequently disable NMI, MASK OFF is set by writing to I/O address 070h, with bit 7 set to 1.

At the completion of the Power On Self-Test (POST), the system program enables NMI.

The NMI provides a Real Mode parity and I/O Channel check routine, which is enabled via interrupt INT 02h. Interrupt INT 02h accesses the ROM-BIOS service routine at vector address 0000:0008h. This routine checks the base 640k memory locations in an attempt to find the memory location with bad parity. When the routine is first accessed it displays the message "Parity Check 1" or "Parity Check 2" as follows:

> Parity Check 1 indicates that parity is being checked within the first 256k bytes or 512k bytes of memory, depending upon the size of the RAM ICs that are being used.
>
> Parity Check 2 indicates that parity is being checked in memory location above the initial 256k or 512k bytes of system memory.

When the address location containing the parity error is found, the service routine displays the address segment. If no parity error is found, the service routine displays the following message in place of the address segment.

> (Intermittent Read Problem) ?????

The I/O Channel Check reports non-correctable errors on the I/O channel. The I/O Channel Check is initialized at power turn ON when the Power On Self-Test (POST) performs the following operations:

1. Data is written to all I/O RAM adapter memory locations to establish good parity at all locations.

2. The I/O CH CK line is enabled.

3. NMI is enabled as previously described.

If a non-correctable error occurs after the POST operations are complete, an NMI is initiated. The system status bits are read to determine the source of the NMI. To determine the location of the error, write to any memory location within a given adapter. If the error was from the adapter that was written to, the I/O CH CK- line is reset to inactive.

Programmable Hardware Interrupts

The programmable hardware interrupts are handled as if there are two 8-channel interrupt controllers connected in cascade mode. The master controller handles interrupt requests IRQ0 through IRQ7 and is located at I/O address 20h. The slave controller handles interrupt Requests IRQ8 through IRQ15 and is located at I/O address A0h. One of the Interrupt Request lines, IRQ2, in the master controller is used to cascade the two controllers. Therefore, fifteen separate interrupting devices or functions can be connected to request interrupts.

Inputs on the interrupt request lines are provided by the system hardware. Interrupt request lines IRQ0 through IRQ15 also have corresponding BIOS interrupts (INT). The interrupt requests, interrupting hardware devices, and the ROM-BIOS interrupt number (INT) and the vector address are given in the following list from the highest to the lowest priority.

NOTE: For additional interrupt controller information refer to Chapter 5

Interrupt Controller	Request	Interrupting Device	ROM-BIOS INT	Vector
Master	IRQ0	Timer (8254) Output	08h	0000:0020
Master	IRQ1	Keyboard (Output Buffer Full)	09h	0000:0024
Master	IRQ2	Cascade to Slave Controller	0Ah	0000:0028
Slave	IRQ8	Real Time Clock	70h	0000:01C0
Slave	IRQ9	Software redirected to INT 0Ah	71h	0000:01C4
Slave	IRQ10	Reserved	72h	0000:01C8
Slave	IRQ11	Reserved	73h	0000:01CC
Slave	IRQ12	Reserved	74h	0000:01D0
Slave	IRQ13	Math Coprocessor (80287)	75h	0000:01D4
Slave	IRQ14	Hard Disk Controller	76h	0000:01D8
Slave	IRQ15	Reserved	77h	0000:01DC
Master	IRQ3	COM 2	0Bh	0000:002C
Master	IRQ4	COM 1	0Ch	0000:0030
Master	IRQ5	Reserved	0Dh	0000:0034
Master	IRQ6	Floppy Disk Drive Controller	0Eh	0000:0038
Master	IRQ7	Parallel Printer Port	0Fh	0000:003C

NOTE: In standard configurations, COM 2 provides the Serial Port Interface and COM 1 provides the Modem Interface. This configuration may be swappable depending upon the revision level of the MS-DOS operating system.

I/O REGISTER SUMMARY

In addition to the ROM-BIOS service routines, GRiDCase 1500 Series computer uses I/O registers to configure the system (including memory) and to control the computer input/output ports. The I/O registers can be used to bypass the ROM-BIOS and provide direct access to the I/O device interfaces. Subsequent chapters of this manual contain detailed information on the use of the I/O registers as well as the ROM-BIOS service routines.

A summary of the I/O registers is provided in Table 3-5. The I/O registers are identified by their memory addresses, which are given in hexadecimal (h). The first part of the table lists all I/O registers that are assigned in groups to system level devices. The remaining portions of the table lists specific uses of I/O register addresses by system level devices. Chapter references are provided to indicate where additional information is available.

Table 3-5. I/O Register Summary

I/O Register Address (Hex)	Register Name
General System Devices	
000 - 01F	DMA Controller, Channels 0-3 (Chapter 4)
020 - 03F	Interrupt Controller, IRQ0-IRQ7 (Chapter 5)
040 - 05F	Programmable Interval Timer (Channel 6)
060, 064	Keyboard Controller (Chapter 8)
061	Read Parity and NMI/Write Parity Enable (Chapter 3)
070	Write NMI Register (Chapter 3)
070 - 07F	Real Time Clock (Chapter 6)
080 - 09F	DMA Page Registers (Chapter 4)
0A0 - 0BF	Interrupt Controller, IRQ8-IRQ15 (Chapter 5)
0C0 - 0DF	DMA Controller, Channels 5-7 (Chapter 4)
416	Write Microprocessor Speed: Fast = 0, Slow = 1
6F8 - 6FF	Write Subsystem Select, Read Subsystem Status
EF8 - EFF	80386 RAM Chip Select
FF5, FF6	Memory Size in megabytes (M): (Not Defined)
FF7	EMS RAM = 0, EXT RAM = 1
Keyboard Interface (Chapter 8)	
060h Read	Keyboard Response Codes, scan codes, data, etc.
060h Write	Keyboard Commands and Data
064h Write	Keyboard Controller Commands
064h Read	Keyboard Subsystem Status

Table 3-5. I/O Register Summary (Continued)

I/O Register Address (Hex)	Register Name
80287/80387 Math Coprocessor (Chapter 2)	
0F0	Clear Math Coprocessor Busy
0F1	Reset Math Coprocessor
0F8, 0FA, 0FC	Math Coprocessor Data
Expanded Memory Mapping (EMS) Control Registers (Chapter 2)	
0258	Page 0 Register, E000:0000 to E000:3FFF
4258	Page 1 Register, E000:4000 to E000:7FFF
8258	Page 2 Register, E000:8000 to E000:BFFF
C258	Page 3 Register, E000:C000 to E000:FFFF
Disk Drive Interface (Chapters 9 and 10)	
1F0h-1F7h	Task File Registers (Chapter 10)
320	Hard Disk Data Register
321	Hard Disk Hardware Status/Reset Register
322	Hard Disk Configuration/Control Select Register
323	Interrupt and DMA Enable Register
379 (Bit 0)	External Disk Drive Address, BOBA- (Chapter 11)
379 (Bit 1)	Main Power Level, MAINPWRLO- (Chapter 11)
379 (Bit 2)	External Disk Drive Type, 3ORS- (Chapter 11)
37A (Bit 5)	External Disk Drive Attached, EFLOPPY- (Chapter 11)
37A (Bit 6)	External Tape Drive Attached, TAPEDETECT (Chap 11)
37A (Bit 7)	High Density Floppy Disk, HDDRIVE (Chapter 11)
3F0, 3F1	Reserved
3F2	Digital Output Register (Write Only)
3F3	Reserved
3F4	Main Status Register (Read Only)
3F5	Data Register (Read/Write)
3F6	Alternate Status/Digital Output Register (Chap 10)
3F7	Digital Input/Control Register (Chap 9)/Drive Address Register (Chap 10)
Display Interface (Chapter 7)	
3D0 - 3DF	Display Controller
7F8 - 7FF	Write Video ROM, Read Video Status
COM 2/COM 1 (Chapter 12)	
2F8/3F8	Transmitter Holding Register (Write Only)
2F8/3F8	Receiver Buffer Register (Read Only)
2F8/3F8	Baud Divisor Latch LSB (When DLAB=1)
2F9/3F9	Baud Divisor Latch MSB (When DLAB=1)
2F9/3F9	Interrupt Enable Register
2FA/3FA	Interrupt Identification Register (Read Only)
2FB/3FB	Line Control Register

Table 3-5. I/O Register Summary (Continued)

I/O Register Address (Hex)	Register Name
COM 2/COM 1 (Continued)	
2FC/3FC	Modem Control Register
2FD/3FD	Line Status Register
2FE/3FD	Modem Status Register
2FF/3FF	Scratch Register

NOTE: In standard configurations, COM 2 provides the Serial Port Interface and COM 1 provides the Modem Interface. This configuration may be swappable depending upon the revision level of the MS-DOS operating system.

Line Printer and Parallel Port (Chapter 11)	
378	Read Printer Data
379(Bits 3-7)	Read Status Register B
37A(Bits 0-4)	Read Status Register C
37B	Undefined
37C	Write Printer Data
37D	No Operation
37E	Printer Control Register
37F	Undefined
423	Parallel I/O Port: 0 = OFF, 1 = ON
FF8	Read Port A
FF9	Read Port B
FFA	Read Port C
FFB	Undefined
FFC	No Operation
FFD	Write Port B
Application ROM Mapping (Chapter 2)	
405	ROM Sockets Enable: 0 = Disabled, 1 = Enabled
440	ROM Socket Select
Special Application Registers	
410-415	Not Used
416	Write Microprocessor Speed: 0 = Fast, 1 = Slow
417	Not Used
420	Analog Loop Back: Modem Diagnostic (Model 1520)
421	Modem Mode: 0 = Smart Terminal, 1 = RS232-C Handshake Mode (Model 1520)
422	Not Used
424	Not Used
425	Not Used

CHAPTER 4: PROGRAMMABLE DMA CONTROLLER

The GRiDCase 1500 Series Computer uses an FE3010 Peripheral Control Logic circuit manufactured by Faraday Electronics, Inc. The FE3010 contains the functional equivalent of two Intel 8237 Programmable Direct Memory Access (DMA) Controllers connected in cascade mode. These DMA Controller circuits allow devices on the data bus to directly transfer data to and from system memory (see Figure 4-1).

The two DMA Controller circuits provide up to seven DMA channels for data transfer operations. An eighth DMA channel is used to cascade the two controller circuits and therefore cannot be used for DMA data transfers. Cascading the controller circuits makes them appear as one seven channel DMA Controller (DMAC).

The DMAC is a dynamic device that is clocked at the system clock rate to provide the following operations:

Address Increment and Decrement.

Independent autoinitialization of all channels.

Enable/disable control of independent DMA Requests.

DMA requests via hardware or software.

In addition, the FE3010 contains a dual ported RAM that provides the DMA Page Register. The page register is used to generate address bits for DMA transfers and refresh cycles.

DMA CHANNEL ASSIGNMENTS

The GRiDCase 1500 Series computer supports seven DMA channels, by using the functional equivalent of two 8237 DMA Controllers connected in cascade mode. An eighth DMA channel (Channel 4) is used to cascade the master and slave controller circuits and therefore cannot be used for DMA data transfers. The eight DMA channels are assigned as follows:

NOTE: Throughout this chapter, numbers given in hexadecimal notation have a letter "h" suffix.

DMA Channel	Page Register	System Function
0	87h	Spare (Highest Priority)
1	83h	Reserved
2	81h	Floppy Disk Drive Controller
3	82h	Hard Disk Drive Controller
4	--	Cascade Channel (No DMA Transfers)
5	8Bh	Spare
6	89h	Spare
7	8Ah	Spare (Lowest Priority)
Refresh	8Fh	RAM Refresh Address Register

Master Controller Channels 0-3 support 8-bit to 8-bit data transfers between I/O Adapters and system memory locations. Each channel transfers data in 64k byte blocks throughout the 16M byte system address space.

Slave Controller Channels 5-7 support 16-bit transfers between 16-bit I/O Adapters and system memory locations. These channels cannot transfer data on odd byte boundaries. Each channel transfers data in 128k byte blocks throughout the 16M byte system address space.

DMA PAGE REGISTER

The DMA Page Register is an 8-bit by 16-byte dual-ported RAM. One port of the page register is a read-only port used to generate address bits for both 8-bit and 16-bit DMA transfers, and to generate memory refresh cycles. For 8-bit DMA transfers, the page register generates address bits 16 through 23. For 16-bit DMA transfers, the page register generates address bits 17 through 23.

The second port of the page register RAM is a read/write port used by the microprocessor.

Depending upon the selected channel, the DMAC addresses blocks of data that are either 64k bytes wide (0-3) or 128k bytes wide (5-7). Refer to the previous list of DMA Channel assignments. In order to access the full 16M bytes of system address space, the page register addresses are combined with the DMA channel address as follows:

Addresses for DMA channels 0-3 are generated by concatenating the Page Register (I/O) address to the address for the master DMA Controller (00h - 1Fh). The total address length is 24 bits.

Addresses for DMA channels 5-7 are generated by concatenating the high order 7-bits of the Page Register (I/O) address to the address for the slave DMA Controller (C0h - DFh). The total address length is 24 bits.

Page addresses for all DMA channels do not increase or decrease through page boundaries.

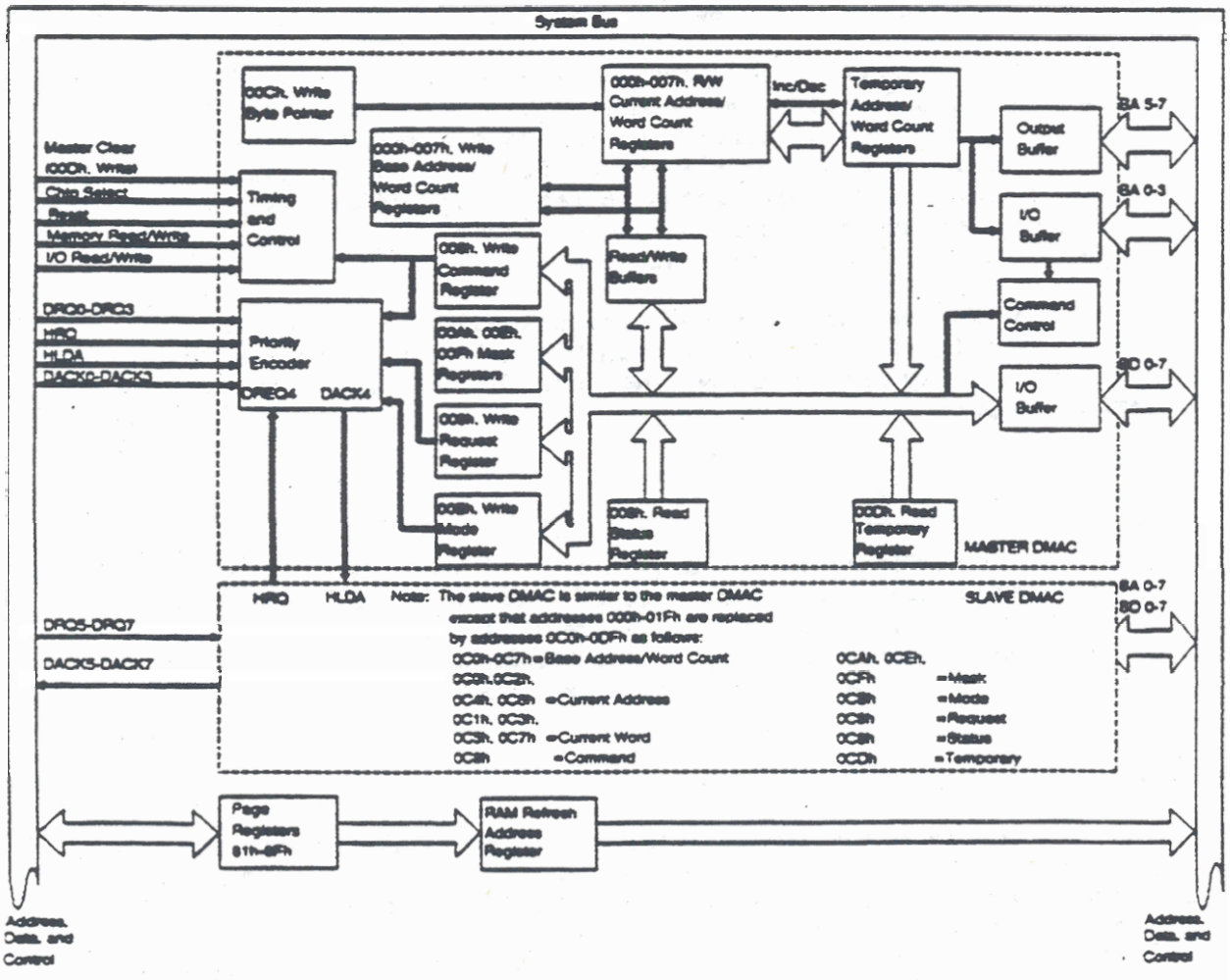


Figure 4-1. DMA Controller Block Diagram

DMAC OPERATIONS

Operations of the DMAC are performed in two device cycles, called the Idle cycle and the Active cycle. If none of the seven channels are requesting service, the DMAC is in the Idle cycle. While in the Idle cycle, the DMAC samples the DMA Request (DRQ0 thru DRQ7) lines on every clock cycle to determine if any channel is requesting service. Also, the Chip Select (CS) line is sampled to determine if the Microprocessor is accessing the DMAC.

The DMAC enters an Active cycle when it is not being programmed and a DMA Request is detected from one of the seven channels. When a DMA Request is received from one of the channels, the DMAC returns the applicable DMA Acknowledge (DACK0 thru DACK2) signals. The DACK0 thru DACK2 lines are decoded by a binary one-of-eight decoder circuit to provide one acknowledge line for each of the seven DMA channels (DACK0 - DACK7). The DMAC then takes control of the address and data buses to begin the data transfer operation.

Data transfers to and from a device require a minimum of four states (S1 thru S4) with each state requiring one full clock cycle. State S1 is the initial state when the DACK signal is true. State S4 is the final state when the End of Process (EOP) signal is true or the word count is decremented to zero. If additional time is needed to complete the transfer, wait states are inserted between states S2 and S3 or between states S3 and S4.

When a channel requests service from the DMAC, the data transfer takes place in the DMAC Single Transfer mode. The Single Transfer mode is the only DMAC transfer mode used by the GRiDCase 1500 Series computer.

Single Transfer Mode

In Single Transfer Mode, the device that requests a data transfer is programmed to transfer one byte of data for each DMA request. As a result of the transfer, the address is incremented or decremented and the word count is decremented toward zero. When the word count is zero, DMA service is terminated.

Cascade Mode

Cascade mode is not a data transfer mode, but a mode used when two four-channel DMA controller circuits are connected together. By connecting the circuits in cascade mode, DMA requests from the slave device are allowed to propagate through the priority network of the master device. One channel (Channel 0) of the slave device

is used to control the cascade operation and does not output any address or control information. In this way, the two DMA controller circuits function independently as two four-channel controllers on the same priority string. A DMA request on any of the seven lines (DRQ 0-3 and DRQ5-7) is responded to in the order of its priority from DRQ0 (highest) to DRQ7 (lowest).

DMAC REGISTER DESCRIPTIONS

The two cascaded DMAC circuits are identical in all respects except for their I/O register addresses. The master DMAC registers are assigned to I/O addresses 00h through 1Fh, and the slave DMAC registers are assigned to I/O addresses C0h through DFh. The following DMAC register descriptions apply separately to each four-channel controller unless it is specifically stated otherwise.

Each four-channel controller has 27 internal registers that vary in size from 16 bits down to 3 bits. Seven of the registers are common to all four channels, and five registers are unique to each of the four channels. The register names, their size in number of bits, and the quantity of registers of each type are listed in Table 4-1. The registers are described in the subsequent paragraphs. For a more complete description of the controller, refer to the information supplied for the 8239A Programmable DMA Controller in the Intel *Microprocessor and Peripheral Handbook, Volume 1*.

Table 4-1 also lists the I/O Register address for each of the controller internal registers and indicates if the controller register is a read only (R) or write only (W). The last column in the table indicates whether the addresses on that line are for the Master controller (M) or the Slave controller (S). The information given for the master controller registers applies to the slave controller registers, except for the register addresses.

The Mask register has three I/O addresses (refer to the note at the end of the table). The Temporary Address and Temporary Word Registers cannot be read or written by the microprocessor.

Base Address and Base Word Count Registers (000h-007h or 0C0h-0C7h)

Each channel has a pair of write-only registers, called Base Address and Base Word Count, which store the original address and word count values. The values can be then be restored in the Current Address and Current Word registers whenever required. The values contained in the Base Address and Base Word Count registers are written into the registers by the microprocessor during the programming operations. The value is written in two successive 8-bit bytes at the same time that the Current Address and Current

Table 4-1. DMAC Internal Registers and Special Commands

Name	Size	Qty	I/O Address Channel No. 0	1	2	3	Type	M/S
DMAC Internal Registers								
Base Address	16	4	000	002	004	006	W	M
			0C0	0C2	0C4	0C6	W	S
Base Word Count	16	4	001	003	005	007	W	M
			0C1	0C3	0C5	0C7	W	S
Current Address	16	4	000	002	004	006	R	M
			0C0	0C2	0C4	0C6	R	S
Current Word	16	4	001	003	005	007	R	M
			0C1	0C3	0C5	0C7	R	S
Temporary Address	16	1	N/A	N/A	N/A	N/A	-	M/S
Temporary Word Count	16	1		N/A	N/A	N/A	-	M/S
Command	8	1	008	008	008	008	W	M
			0C8	0C8	0C8	0C8	W	S
Mask	4	1		See Note				
Mode	8	4	00B	00B	00B	00B	W	M
			0CB	0CB	0CB	0CB	W	S
Request	3	1	009	009	009	009	W	M
			0C9	0C9	0C9	0C9	W	S
Status	8	1	008	008	008	008	R	M
			0C8	0C8	0C8	0C8	R	S
Temporary	8	1	00D	00D	00D	00D	R	M
			0CD	0CD	0CD	0CD	R	S
Special Commands								
Clear First/Last Flip Flop			00C	00C	00C	00C	W	M
(Byte Pointer)			0CC	0CC	0CC	0CC	W	S
Master Clear			00D	00D	00D	00D	W	M
			0CD	0CD	0CD	0CD	W	S
Clear Mask Register			00E	00E	00E	00E	W	M
			0CE	0CE	0CE	0CE	W	S

NOTE:
The Mask register has three separate I/O register addresses. The address to use depends upon the required operation as follows:

1. To set/reset individual mask bits use address 00Ah for the master or 0CAh for the slave.
2. To set or reset all four mask bits with one instruction use address 00Fh for the master and 0CFh for the slave.
3. To clear the four mask bits and allow DMA on all four channels, use address 00Eh for the master and 0CEh for the slave.

Word Count registers are written. An internal Byte Pointer flip-flop determines if the high byte or low byte is written. (Refer to the Special Commands at the end of this chapter.) The values in the Base Address and Base Word Count registers cannot be read by the microprocessor. The I/O address for each register is listed in Table 4-1.

Current Address Register (000h,002h,004h,006h, or 0C0h,0C2h,0C4h,0C6h)

The four channels each have a Current Address Register that holds the address value used during DMA transfers. The address value is automatically incremented or decremented after each data transfer. The new value is retained in the register until the next data transfer or until the register is Autoinitialized following a true EOP signal. If Autoinitialized, the Current Address Register is restored to its original value from the value stored in the applicable Base Address Register. The Current Address Register is written or read by the microprocessor in two successive 8-bit bytes. An internal Byte Pointer flip-flop determines if the high byte or low byte is written. (Refer to the Special Commands at the end of this chapter.) The original address value is written into the Current Address Register and the Base Address Register at the same time during programming operations. The I/O address for each register is listed in Table 4-1.

Current Word Register (001h,003h,005h,007h or 0C1h,0C3h,0C5h,0C7h)

The value contained in the Word Count Register determines how many data transfers are performed. Each channel has a Word Count Register that keeps track of how many transfers have occurred. The number of transfers performed is always one greater than the number contained in the register. The word count value is automatically decremented after each data transfer. The new value is then retained in the register until the next data transfer, or until the register is Autoinitialized following a true EOP signal. If Autoinitialized, the Current Word Register is restored to its original value from the value stored in the applicable Base Word Count Register.

When the word count value in the Current Word Count Register is decremented through zero to FFFFh, a Terminal Count (TC) is generated to indicate that the programmed number of transfers has occurred. The TC is used to generate the true EOP signal and also to enable Autoinitialize if the channel is programmed for that operation. If not Autoinitialized, the Current Word Register has a word count of FFFFh following a TC.

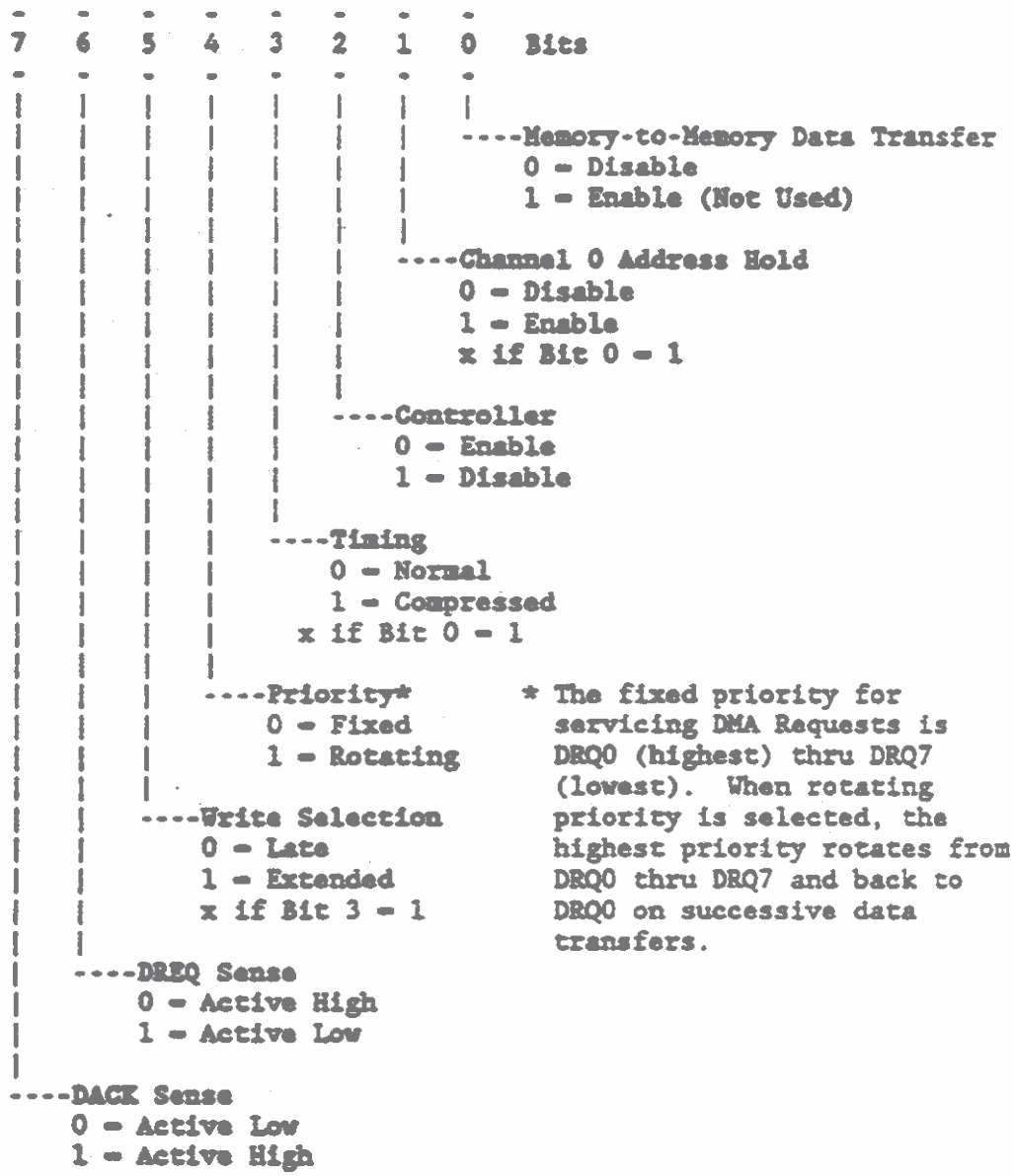
The Current Word Count Register is written and read by the microprocessor in two successive 8-bit bytes. An internal Byte Pointer flip-flop determines if the high byte or low byte is written. (Refer to the Special Commands at the end of this chapter.) The original word count value is written into the Current Word Count Register and the Base Word Count Register at the same time during programming operations. The I/O address for each register is listed in Table 4-1.

Temporary Address and Word Count Registers

There is one pair of registers called the Temporary Address and Temporary Word Count Registers. These registers are used as temporary buffers for incrementing or decrementing the current address value and for decrementing the current word count value. Incrementing and decrementing are done automatically when data transfers occur. Therefore, the Temporary Address and Word Count Registers cannot be written or read by the microprocessor.

Command Register (008h or 0C8h)

The Command Register is an 8-bit, write-only buffer. The Command register is used with the Mode Register to select the operating parameters of the controller. The Command register is loaded by the microprocessor during programming operations, and is cleared by a Master Clear Command or a System Reset. The register bits are programmed as follows:



Mask Register (00A,00Eh,00Fh or 0CAh,0CEh,0CFh)

The Mask Register is an 8-bit, write-only buffer that uses only three or four bits depending upon how it is programmed. To set or reset one mask bit at a time without affecting the other three bits, I/O address 00Ah is used for the master controller and address 0CAh is used for the slave controller. To set or reset all four mask bits with one instruction, I/O address 00Fh is used for the master controller and address 0CFh is used for the slave controller. I/O address 00Eh is used for the master controller and 0CEh is used for the slave controller to clear the four mask bits so that all four channels can request an interrupt. Also, the

Master Reset command (I/O address 00Dh for the master controller and 0CDh for the slave controller) sets the mask bits so that DMA requests are inhibited on all four channels.

The Mask register is used with the Request Register to control the incoming DMA Requests (DRQ0 thru DRQ7) signals for all seven DMA channels. The mask bit associated with each channel is set to disable the DMA Requests and cleared to enable the DMA Requests. If the channel is not programmed to Autoinitialize, its mask bit is set when EOP goes true. Also, a System Reset sets the mask bits for all seven DMA channels and inhibits all DMA requests. When set, the Mask bits remain set until cleared by addressing the appropriate I/O register. Depending upon how they are programmed during programming operations, each mask bit is set and reset separately or all together as shown in the following bit definitions.

I/O Address = 00Ah or 0CAh

```
    ----------------   -   -----
    7   6   5   4   3   2   1   0   Bits
    ----------------   -   -----
            |              |   |
        Don't Care         |   ----Selected Mask Bit
                           |       0 0 = Channel 0 (DRQ0)
                           |       0 1 = Channel 1 (DRQ1 or DRQ5)
                           |       1 0 = Channel 2 (DRQ2 or DRQ6)
                           |       1 1 = Channel 3 (DRQ3 or DRQ7)
                           |
                           ----Mask Bit
                               0 = Clear
                               1 = Set
```

I/O Address = 00Fh or 0CFh

```
    ------------   -   -   -   -
    7   6   5   4   3   2   1   0   Bits
    ------------   -   -   -   -
          |        |   |   |   |
      Don't Care   |   |   |   ----Channel 0 Mask Bit
                   |   |   |       0 = Clear
                   |   |   |       1 = Set
                   |   |   |
                   |   |   ----Channel 1 Mask Bit
                   |   |       0 = Clear
                   |   |       1 = Set
                   |   |
                   |   ----Channel 2 Mask Bit
                   |       0 = Clear
                   |       1 = Set
                   |
                   ----Channel 3 Mask Bit
                       0 = Clear
                       1 = Set
```

Mode Register (00Bh or 0CBh)

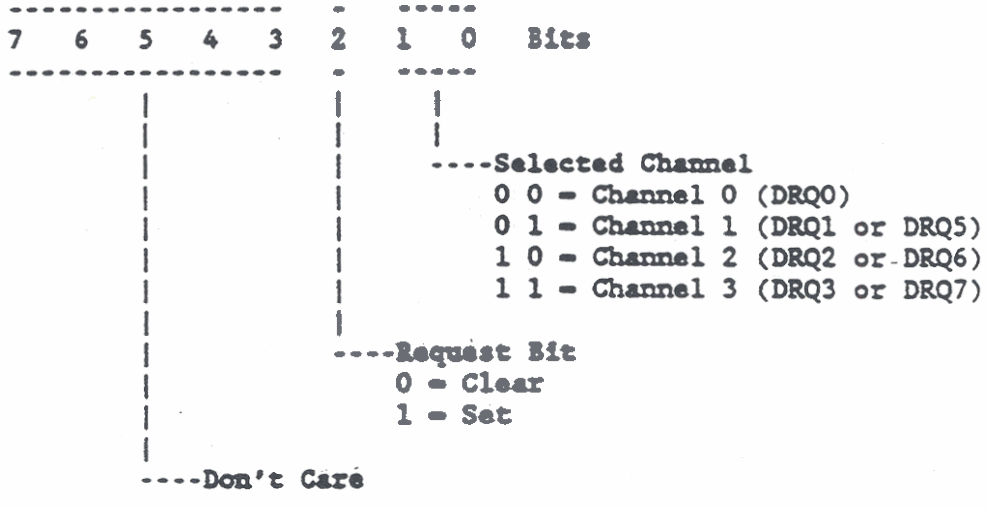
A Mode Register is used with the Command Register to select operating parameters for each channel of the four-channel controllers (I/O address 00Bh for the master and 0CBh for the slave). A separate 8-bit, write-only Mode Register is required for each of the four channels. The registers are loaded by the microprocessor during programming operations. The Mode Registers select the data transfer type and transfer mode, enable Autoinitialization, and the address direction (increment or decrement). Only one I/O address is required for all four channels of each controller because bits 1 and 0 of the register determine which of the four Mode Registers is written. The Mode Register bits are programmed as follows:

```
-----   -   -   -----   -----
7   6   5   4   3   2   1   0   Bits
-----   -   -   -----   -----
  |     |   |     |       |
  |     |   |     |       --Channel Select
  |     |   |     |         0 0 = Channel 0
  |     |   |     |         0 1 = Channel 1
  |     |   |     |         1 0 = Channel 2
  |     |   |     |         1 1 = Channel 3
  |     |   |     |
  |     |   |     --Transfer Type
  |     |   |       0 0 = Verify
  |     |   |       0 1 = Write
  |     |   |       1 0 = Read
  |     |   |       1 1 = Illegal Value
  |     |   |       x x if Bits 6 and 7 = 11
  |     |   |
  |     |   ----Autoinitialization
  |     |         0 = Disable
  |     |         1 = Enable
  |     |
  |     ----Address Direction Select
  |           0 = Increment
  |           1 = Decrement
  |
  ----Data Transfer Mode Select
      0 0 = Demand Mode
      0 1 = Single Mode*
      1 0 = Block Mode
      1 1 = Cascade Mode*
```

* The Single Mode and Cascade Mode are the only Data Transfer modes used by the GRiDCase 1500 Series computer.

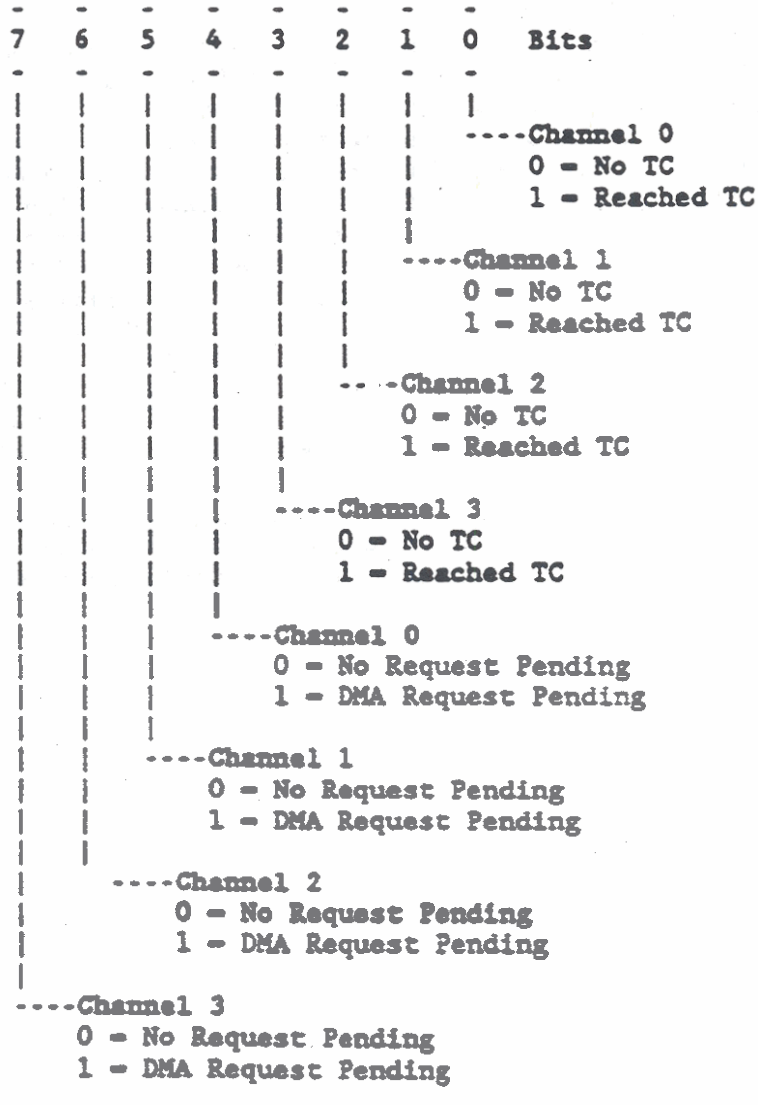
Request Register (009h or 0C9h)

The Request Register is an 8-bit, write-only buffer, that is accessed via I/O address 009h for the master controller or 0C9h for the slave controller. The Request Register allows the software to initiate a DMA request for service. To make a software request, the Single Transfer Mode must be selected via the Mode register. Then, one of the four channels is selected by three nonmaskable bits of the Request Register. Software DMA requests are subject to the same priority encoding via the Command register as the hardware requests. To set or reset each DMA service request bit separately, the Request Register is loaded with the desired data word by the microprocessor during programming operations. The bit that was set is cleared when Terminal Clear (TC) is generated, an external EOP is received, or the register is cleared by a Master Clear command or a System Reset. The bit definitions for the Request Register are as follows:



Status Register (008h or 0C8h)

The Status Register is an 8-bit, read-only buffer that provides two bits of status information for each of the four channels in the selected controller. The status information consists of one bit to indicate when the channel has a DMA Service Request pending and the other bit to indicate when the channel has reached a Terminal Count (TC). The DMA request pending bit is set to "1" when the channel is requesting service. The DMA request pending bit is reset to "0" by a Terminal Count (TC), when an external EOP is received, or when a System Reset occurs. The TC bit is set to "1" each time the channel reaches a terminal count. The TC bits are reset to "0" each time the Status Register is read and also by a Master Clear command or a System Reset. The Status register is read via I/O address 008h for the master controller or 0C8h for the slave controller. Bit definitions for the Status Register are as follows:



Temporary Register (00Dh or 0CDh)

The Temporary Register is an 8-bit, read-only buffer that holds data during memory-to-memory data transfers. The Temporary Register is not used because memory-to-memory data transfers are not used by the GridCase 1500 Series computer.

Special Commands

There are three special commands in each controller that control hardware registers. These three special commands are accessed via their I/O register addresses and do not depend upon any specific bit pattern on the data lines. The three commands are:

Clear First/Last Flip-Flop (00Ch or 0CCh)

The First/Last Flip-Flop special command is accessed via a write to I/O address 00Ch for the master controller or 0CCh for the slave controller. The command clears the Byte Pointer flip-flop that keeps the controller upper and lower address bytes in the proper sequence. Before writing or reading new address or word count information, the program executes the Clear First/Last Flip-Flop command to initialize the Byte Pointer flip-flop to a known state. The flip-flop is also cleared by a Master Clear command. Subsequent accesses to the Address and Word Count registers are then made in the correct sequence.

Master Clear (00Dh or 0CDh)

The Master Clear special command is accessed via a write to I/O address 00Dh for the master controller or 0CDh for the slave controller. The command forces the controller into the Idle state. When the command is executed, it has the same effect on the controller as a System Reset. Executing a Master Clear command or issuing a System Reset sets the Mask Register to inhibit DMA requests on all channels and clear the following five registers in each controller:

- Command
- Request
- Status
- Temporary
- First/Last Flip-Flop

Clear Mask Register (00Eh or 0CEh)

The Clear Mask Register command is accessed via a write to I/O address 00Eh for the master controller or 0CEh for the slave controller. The command clears the Mask register to allow all four channels in the respective controller to accept DMA requests for service.

CHAPTER 5: PROGRAMMABLE INTERRUPT CONTROLLER

The GRiDCase 1500 Series Computer uses an FE3010 Peripheral Control Logic circuit manufactured by Faraday Electronics, Inc. The FE3010 contains the functional equivalent of two Intel 8259 Programmable Interrupt Controller circuits connected in cascade mode. These two interrupt controller circuits, and the microprocessor NonMaskable Interrupt (NMI), provide 16 levels of vectored priority interrupts for the 80286/80386 Microprocessor. The NMI provides either a parity or an I/O Channel check function.

Connecting the two interrupt controller circuits in cascade mode makes them appear to the microprocessor as one Programmable Interrupt Controller (PIC) with 15 levels of prioritized interrupts (see Figure 5-1). The PIC manages the interrupt-driven activities of the computer so that the microprocessor has more time to run other programmed operations. Interrupt requests are received by the PIC from input/output devices such as the disk drives, printer, and keyboard. The PIC screens the incoming interrupt requests for the highest priority, and also checks the priority of the incoming

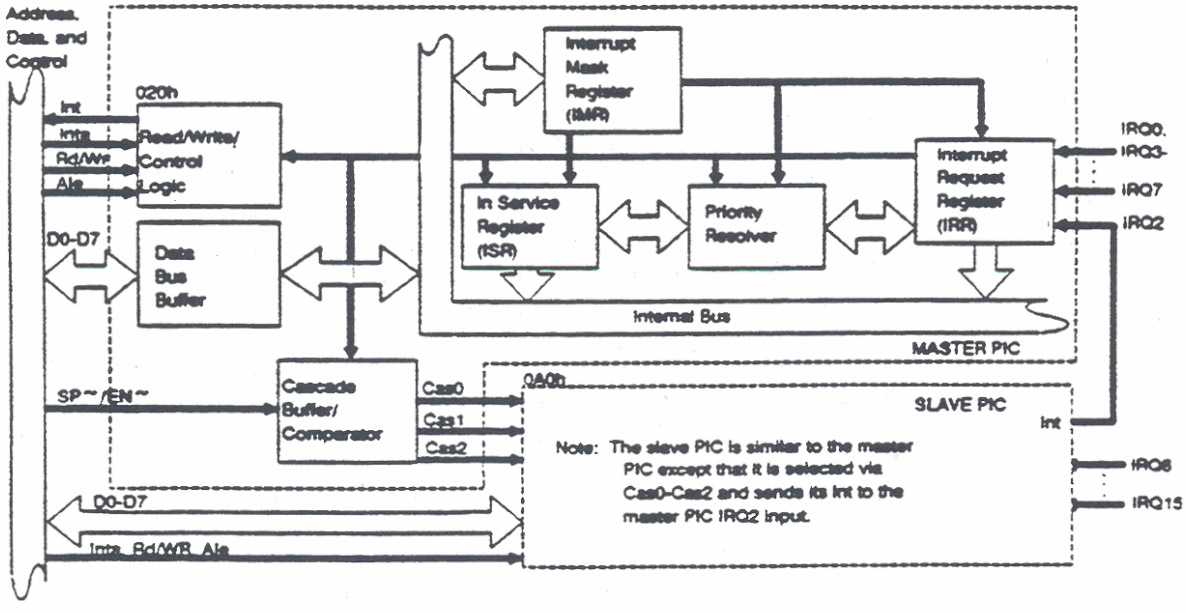


Figure 5-1. Programmable Interrupt Controller Block Diagram

request against the priority of the device currently being serviced. If the incoming interrupt request has a higher priority than the request being serviced, the microprocessor is allowed to complete the current instruction before servicing the higher priority interrupt. When the interrupting routine is complete, the microprocessor resumes execution of the previous instruction from the point where it was interrupted.

Each device that issues interrupt requests is connected to one of the PIC inputs labeled IRQ0 thru IRQ15. When active, the interrupt requests generate a corresponding interrupt (INT) that points to an interrupt vector address. The interrupt vector address directs the microprocessor program counter to the starting address of an interrupt service routine. When interrupted by the PIC, the microprocessor stores its current address and then jumps to the starting address of the service routine pointed to by the interrupt vector address. The service routine is a special program, associated with the requesting device, that defines how the device is serviced by the microprocessor. The last instruction in the service routine is a jump to the address location that the microprocessor stored before it began servicing the interrupt request.

If an interrupt service routine is divided into separate operations, the interrupt request is further defined by a function number. When requesting an interrupt, the function number is placed in the microprocessor AH register. The function number points to subroutine within the service routine for the requesting device. The requesting device may require different subroutines for read data, write data, read status, and write control operations. The service routine branches to the subroutine to perform the requested operation (refer to Table 3-2).

For the GRiDCase 1500 Series computers, the interrupt vector table is located in the first 1k bytes of system memory. This area provides room for 256 distinct interrupts, which is more than are routinely used. The vector for interrupt 0 is located at memory location 0. Since each vector address is four bytes wide, the vector for any interrupt number (n) is located at address n times 4. The addresses in the interrupt vector table point to an area of the ROM-BIOS or to an address in main memory where the interrupt service routine is located. Generally, interrupts invoked by hardware have their interrupt routines located in the ROM-BIOS, and interrupts invoked by software have their interrupt routines located in system memory.

HARDWARE INTERRUPT ASSIGNMENTS

The GRiDCase 1500 Series computer uses the functional equivalent of two cascaded 8259 Programmable Interrupt Controllers to provide up to 15 interrupts, including one interrupt from the counter-timer. A master interrupt controller circuit controls interrupts INT0 through INT7 and is located at I/O address 20h. A slave interrupt controller circuit controls interrupts INT8 through INT15 and is located at I/O address A0h. One interrupt request line (IRQ2) is used for cascading the slave controller interrupt requests to the master controller.

NOTE: Throughout this chapter, numbers given in hexadecimal notation have a letter "h" suffix.

Interrupt requests to the interrupt controllers are provided by the system hardware and peripheral devices. The interrupt requests (IRQ0-IRQ15), the interrupting hardware devices, and the ROM-BIOS controlled interrupt address (INT) and vector start address designations are given in the following list from the highest to the lowest priority.

Interrupt Controller	Request	Interrupting Device	ROM-BIOS INT	ROM-BIOS Vector
Master	IRQ0	Timer (8254) Output	08h	0000:0020
Master	IRQ1	Keyboard (Output Buffer Full)	09h	0000:0024
Master	IRQ2	Cascade to Slave Controller	0Ah	0000:0028
Slave	IRQ8	Real Time Clock	70h	0000:01C0
Slave	IRQ9	Software redirected to INT 0Ah	71h	0000:01C4
Slave	IRQ10	Reserved	72h	0000:01C8
Slave	IRQ11	Reserved	73h	0000:01CC
Slave	IRQ12	Reserved	74h	0000:01D0
Slave	IRQ13	Math Coprocessor (80287)	75h	0000:01D4
Slave	IRQ14	Hard Disk Controller	76h	0000:01D8
Slave	IRQ15	Reserved	77h	0000:01DC
Master	IRQ3	COM 2	0Bh	0000:002C
Master		COM 1	0Ch	0000:0030
Master	IRQ5	Reserved	0Dh	0000:0034
Master	IRQ6	Floppy Disk Drive Controller	0Eh	0000:0038
Master	IRQ7	Parallel Printer Port	0Fh	0000:003C

NOTE: In standard configurations, COM 2 provides the Serial Port Interface and COM 1 provides the Modem Interface. This configuration may be swappable depending upon the revision level of the MS-DOS operating system.

PIC OPERATIONS

The PIC provides two independent interrupt controller circuits that are connected in cascade mode to manage 15 levels or requests for interrupts. The interrupt output (INT) of the slave controller circuit is connected to an interrupt request input (IRQ2) of the master controller circuit. In this way, the two interrupt controller circuits function independently as two eight-channel Programmable Interrupt Controllers on the same priority string.

The two PIC circuits are functionally identical in all respects except for their I/O addresses. The master PIC circuits are assigned to I/O addresses 00h through 1Fh and the slave PIC circuits are assigned to I/O addresses A0h through BFh. The following PIC register descriptions apply separately to each of the eight-channel controller circuits unless it is specifically stated otherwise.

The PIC is programmed by the system software as an I/O device. Priority modes of the PIC are programmable so that the way requests are processed is configured to match the system requirements. The priority modes can be changed at any time while running the MS-DOS Operating System, which allows the interrupt structure to be redefined as required.

For the GRiDCase 1500 Series Computer, the PIC operations are divided into the following functions:

1. Data Bus Buffer
2. Read/Write/Control Logic
3. Interrupt Request Register (IRR)
4. In Service Register (ISR)
5. Priority Resolver
6. Interrupt Mask Register (IMR)
7. Cascade Buffer and Comparator (CB/C)

Data Bus Buffer

The Data Bus Buffer is a 3-state, bidirectional 8-bit register. The register provides the interface between the PIC and the system data bus. The command words used to program the PIC are transferred through the Data Bus Buffer.

Read/Write/Control Logic

The Read/Write/Control logic receives the microprocessor control signals and provides the registers used to program the PIC. The programmable registers store the priority modes and control formats required for device operation. Also, the PIC status information is stored in the registers until it is transferred through the Data Bus Buffer to the microprocessor.

Interrupt Request Register (IRR)

The Interrupt Request Register (IRR) receives the Interrupt Request signals (IRQ0-IRQ7 or IRQ8-IRQ15) from the interrupting devices. The IRR stores all of the interrupt requests for resolution by the Priority resolver.

In Service Register (ISR)

The In Service Register (ISR) works in cascade with the IRR. The ISR stores the Interrupt Requests that are currently being serviced. Input to the ISR comes from the Priority Resolver, which puts the highest priority request into the ISR during the interrupt acknowledge (INTA) pulse from the microprocessor.

Priority Resolver

The Priority Resolver determines the priorities of the bits stored in the IRR. The highest priority line is then selected and strobed into the corresponding bit position in the ISR.

Interrupt Mask Register (IMR)

The Interrupt Mask Register (IMR) screens the inputs to the IRR. Any interrupt request can be masked to prevent the associated device from requesting an interrupt. Masking interrupts from high priority devices does not affect the interrupts requested by lower priority devices.

Cascade Buffer and Comparator (CB/C)

The Cascade Buffer and Comparator (CB/C) is used to store and compare the identification of all PICs in the system. The three CB/C lines called CAS0 through CAS2 are output lines on the master PIC and input lines on the slave PIC. The CAS lines are used by the master PIC to enable the slave PIC operations.

Interrupt Sequence

The PIC programmability and interrupt service routine addressing capability allow it to jump directly or indirectly to the specific routine that is requested.

Normal Operation

The sequence of events to service a normal interrupt is given in the following steps.

1. After the PIC has been programmed, one or more devices request an interrupt. The corresponding Interrupt Request lines (IRQ0-IRQ7 or IRQ8-IRQ15) then go to a 'high' state and the related IRR bits are set.

2. The PIC evaluates the requests and sends an Interrupt (INT) signal to the microprocessor.

3. The microprocessor acknowledges the INT signal by returning the first of two Interrupt Acknowledge (INTA) pulses to the PIC.

4. When the first INTA pulse is received, the PIC freezes the state of the interrupts for priority resolution, sets the highest priority bit in the ISR, and resets the corresponding bit in the IRR. The PIC does not drive the data bus on this cycle.

5. The microprocessor then sends the second INTA pulse. During the second INTA pulse the PIC sends the following 8-bit byte of data to the microprocessor.

```
-----------------   ---------
7   6   5   4   3   2   1   0   Bits
-----------------   ---------
        |               |
        |               ----Interrupt Request Number
Interrupt Address           0 0 0 - IRQ0 or IRQ8
  Bits T7-T3,               0 0 1 - IRQ1 or IRQ9
 respectively               0 1 0 - IRQ2 or IRQ10
                            0 1 1 - IRQ3 or IRQ11
                            1 0 0 - IRQ4 or IRQ12
                            1 0 1 - IRQ5 or IRQ13
                            1 1 0 - IRQ6 or IRQ14
                            1 1 1 - IRQ7 or IRQ15
```

6. If Automatic End Of Interrupt (AEOI) was programmed, the ISR bit is reset at the end if the second INTA pulse and the interrupt request is complete. If AEOI was not programmed, the ISR bit remains set until an appropriate End Of Interrupt (EOI) is issued at the end of an interrupt service routine.

Cascade Mode Operation

The sequence of events to service an interrupt in cascade mode is similar to normal operation, except as follows:

1. When the slave PIC has selected its highest priority interrupt, it notifies the master PIC by asserting its interrupt request (INT) line at the IRQ2 input of the master PIC.

2. The master PIC is ready to service the interrupt from the slave PIC, when it sends the slave PIC identification bits over the CAS0-CAS2 lines.

3. The slave PIC then outputs the I/O address for the interrupt request to be serviced. The I/O address is output onto the data bus during the subsequent INTA pulses from the microprocessor.

4. At completion of the interrupt service, two End Of Interrupt (EOI) commands are issued: one for the slave PIC and one for the master PIC.

PIC REGISTER DESCRIPTIONS

The PIC accepts two types of command words from the microprocessor. Both types of command words use the same I/O Address because the operation being performed is determined by bits within the command word. The two types of command words are:

1. Initialization Command Words (ICWs).

   ICWs must be issued through I/O address 020h for the master PIC or 0A0h for the slave PIC before normal operation can begin. By issuing a series of two to four bytes timed by a write (WR) pulse, the PIC is initialized to a known starting point.

   The master PIC and the slave PIC must be programmed separately. The only difference in the way they are programmed occurs during initialization (ICW3) as described in the following paragraphs.

2. Operation Command Words (OCWs).

   OCWs are issued through I/O address 020h for the master PIC and 0A0h for the slave PIC anytime following initialization, and are used to change the interrupt priority mode of the PIC. Priority modes selectable by the OCWs are as follows:

   a. Fully Nested Mode
   b. Rotating Priority Mode
   c. Special Mask Mode
   d. Polled Mode

Both types of command words require nine bits for their bit definition. Bit 0 of the address bus (A0) is borrowed and used with data bus bits D7 thru D0 to provide the nine bits. Bit definitions for the command words are provided in the following paragraphs. For additional information on the PIC operation, refer to the information on the 8259A Programmable Interrupt Controller in the Intel *Microprocessor and Peripheral Handbook, Volume 1*.

Initialization Command Words (020h or 0A0h)

A write to I/O register address 020h for the master controller or 0A0h for the slave controller starts the PIC initialization sequence. During initialization, the following operations automatically occur:

1. The interrupt request sensing circuit is reset so that the input signal (IRQ0 - IRQ7 or IRQ8 - IRQ15) must go through a low-to-high transition to generate an interrupt.

2. The Interrupt Mask Register (IMR) is cleared.

3. Interrupt Request 7 (IRQ7) is assigned priority level 7. This value and the value in step 4 are changed during the subsequent initialization.

4. The slave mode address is set to 7.

5. The Special Mask Mode is cleared and Status Read is set to IRR.

6. When bit 0 of the first ICW (ICW1) is set to "0," all functions selected by ICW4 are also set to "0".

Bit definitions for the Initialization Command Words (ICW1-ICW4) are as follows:

A0	D7	D6	D5	D4	D3	D2	D1	D0	Bits
0	A7	A6	A5	1	LTM	ADI	SNGL	ICW4	ICW1 Command Word

- Not Used
- A0 = 0, D4 = 1 Enables ICW1 and Starts Initialization
- ICW4 Required: 0 = ICW4 is needed; 1 = ICW4 not needed
- Number of PICs: 0 = Cascade Mode; 1 = Single PIC
- Call Address Interval: Call Opcode is not used
- IRQ0-IRQ7 or IRQ8-IRQ15 Trigger Mode: 0 = Edge Triggered Mode; 1 = Level Triggered Mode

A0	D7	D6	D5	D4	D3	D2	D1	D0	Bits
1	T7	T6	T5	T4	T3	A10	A9	A8	ICW2 Command Word (Interrupt Vector Address)

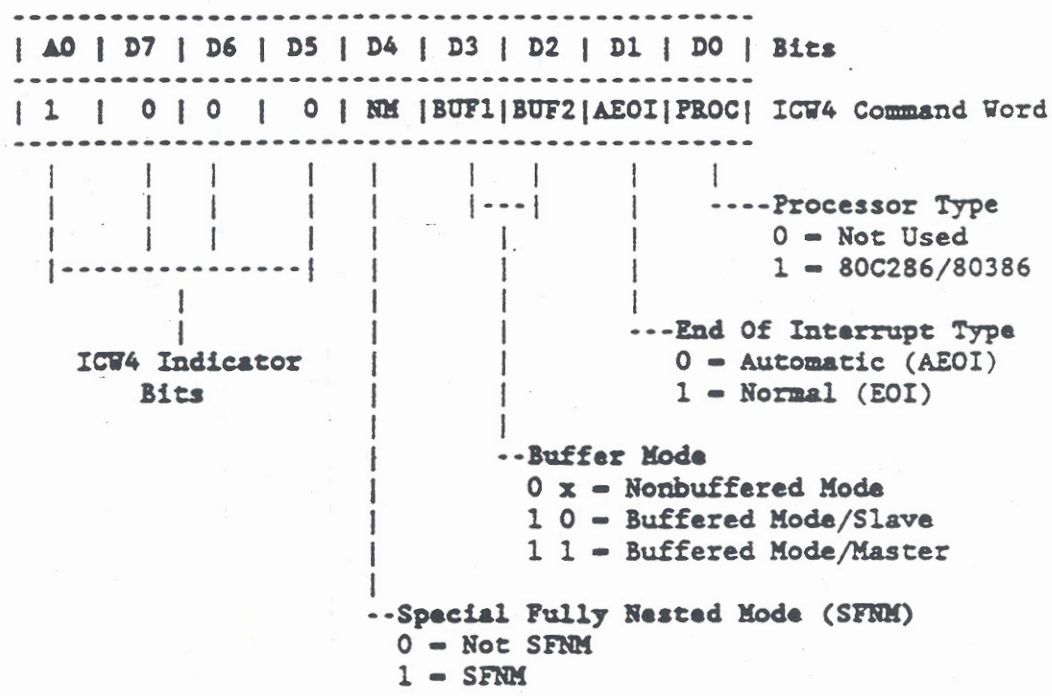
Bits T7 through T3 are inserted in the five most significant bits of the vectoring byte. The PIC then sets the three least significant bits (A10-8) according to the selected interrupt level. A0 = 1 is a program indicator.

A0	D7	D6	D5	D4	D3	D2	D1	D0	Bits
1	S7	S6	S5	S4	S3	S2	S1	S0	ICW3 Command Word (For Master PIC)

When SNGL (bit 1 of ICW1) = 0, S7 - S0 indicate which inputs (IRQ7 - IRQ0, respectively) are connected to a slave PIC. For the GRiDCase 1500 Series computers, S2 = 1. See ICW3 for slave PIC.

A0	D7	D6	D5	D4	D3	D2	D1	D0	Bits
1	0	0	0	0	0	ID2	ID1	ID0	ICW3 Command Word (For Slave PIC)

The identification bits (ID2 - ID0) must match the cascade lines CAS2 - CAS0 from the master PIC. Since the slave INT output is connected to IRQ2 on the master, ID2 - ID0 = 010. See ICW3 for master PIC.



Operation Command Words (020h or 0A0h)

When the Initialization Command Words (ICWs) are programmed, the PIC is ready to accept interrupt requests (IRQ0-IRQ7 or IRQ8-IRQ15) at its input lines. During subsequent operation, Operation Command Words (OCWs) are written (via I/O address 020h for the master PIC or 0A0h for the slave PIC) to change the interrupt mask, interrupt request level, and operating mode. Bit definitions for the Operation Command Words (OCW1-OCW3) are as follows:

A0	D7	D6	D5	D4	D3	D2	D1	D0	Bits
1	M7	M6	M5	M4	M3	M2	M1	M0	OCW1 Command Word

Bits M7 through M0 are the Interrupt Mask Bits for Interrupt Requests IRQ7 through IRQ0 or IRQ15 through IRQ8, respectively.

0 = Mask Bit is reset
1 = Mask Bit is set

A0	D7	D6	D5	D4	D3	D2	D1	D0	Bits
0	ROT	SEL	EOI	0	0	LEV2	LEV1	LEV0	OCW2 Command Word

A0, D4, and D3 Provide the OCW2 Indicator Bits (000)

See Bit A0

--Interrupt Level (8-Bit)
Refer to Interrupt Level Bit Definition that follows.

--Rotate and EOI Mode Control

0 0 1 - Nonspecific EOI Command[1]
0 1 1 - Specific EOI Command[1]
1 0 1 - Rotate On Nonspecific EOI Command[2]
1 0 0 - Rotate in AEOI Mode (Set)[2]
0 0 0 - Rotate in AEOI Mode (Clear)[2]
1 1 1 - Use LEV2 - LEV0 to Rotate on Specific EOI Command[3]
1 1 0 - Use LEV2 - LEV0 to Set Priority Command[3]
0 1 0 - No Operation

NOTES:
1. End Of Interrupt Operations
2. Automatic Rotation Operations
3. Specific Rotation

A0	D7	D6	D5	D4	D3	D2	D1	D0	Bits
0	0	ESMM	SMM	0	1	POLL	IRR	ISR	OCW3 Command Word

Bits A0, D7, D4, and D3 Provide the OCW3 Indicator Bits (0001)

See Bit A0

--Read PIC Status
Refer to the Reading PIC Status information that follows.

--Poll Command
0 = No Poll Command
1 = Poll Command*

*Poll command inhibits interrupts. Device service is then achieved by software using a poll command.

---Special Mask Mode
0 x = Special Mask Mode Not Enabled
1 0 = Normal Mask Mode
1 1 = Special Mask Mode Enabled

Interrupt Levels

The following lists gives the interrupt levels controlled by bits D0 through D2, which select LEV0 through LEV2 of OCW2. The LEV2 through LEV0 bits select a pattern of OCW2 bits that are acted upon when the SEL bit (D6) is active. Bits 0 through 7 represent IRQ0 through IRQ7 or IRQ8 through IRQ15, respectively.

0	1	2	3	4	5	6	7	Bits
0	1	0	1	0	1	0	1	LEV0 = 1
0	0	1	1	0	0	1	1	LEV1 = 1
0	0	0	0	1	1	1	1	LEV2 = 1

Reading PIC Status

To aid in updating priority levels and modes, the status of the PIC is determined by reading the input registers. The following registers, which where described previously, can be read during PIC operation:

1. Interrupt Request Register (IRR)

2. In Service Register (ISR)

3. Interrupt Mask Register (IMR)

The contents IRR and ISR are read at port 20h or A0h by first setting the associated bits of Operation Command Register 3 (OCW3), and then issuing a Read Register Command. Bits D1 and D0 of OCW3 are used to select which of the two registers is read. Bit definitions for D1 and D0 are as follows:

D1	D0	OCW3
IRR	ISR	
0	x	= Registers are not read
1	0	= IRR is read
1	1	= ISR is read

After the D1 and D0 bits of OCW3 are set, they remain in the same state for subsequent status reads. Following initialization (ICW1), IRR is set to read and remains set until specifically changed by writing a new OCW3.

Reading the contents of IMR requires that port A0h = 1, which is true for OCW1. Then, the output bus contains IMR and is read when the RD pulse is active.

CHAPTER 6: TIMER, SPEAKER, AND REAL-TIME CLOCK

The GRiDCase 1500 Series Computer provides three general-purpose logic circuits that are required for computer operation. The three circuits are a Programmable Timer/Counter, Speaker Control, and a Time-of-Day Clock. This chapter provides descriptions of the three general-purpose logic circuits.

The ROM-BIOS subroutines support Time Of Day (TOD) clock operations, but do not support the timer and speaker operations. Timer and speaker operations are interfaced at the hardware level via I/O register addresses provided by the microprocessor.

PROGRAMMABLE TIMER/COUNTER

The GRiDCase 1500 Series Computer uses an FE3010 Peripheral Control Logic circuit manufactured by Faraday Electronics, Inc. The FE3010 contains the functional equivalent of an Intel 8254 Programmable Interval Timer, which contains three independent 16-bit counters, control logic, and a data buffer. All three counters are driven by a 1.19 MHz clock input. The clock input to each counter is then divided by the 16-bit value that is loaded into it through an 8-bit parallel data bus. The output from the three counters is used for various timing requirements as described in the following list. The I/O addresses are given in hexadecimal (h).

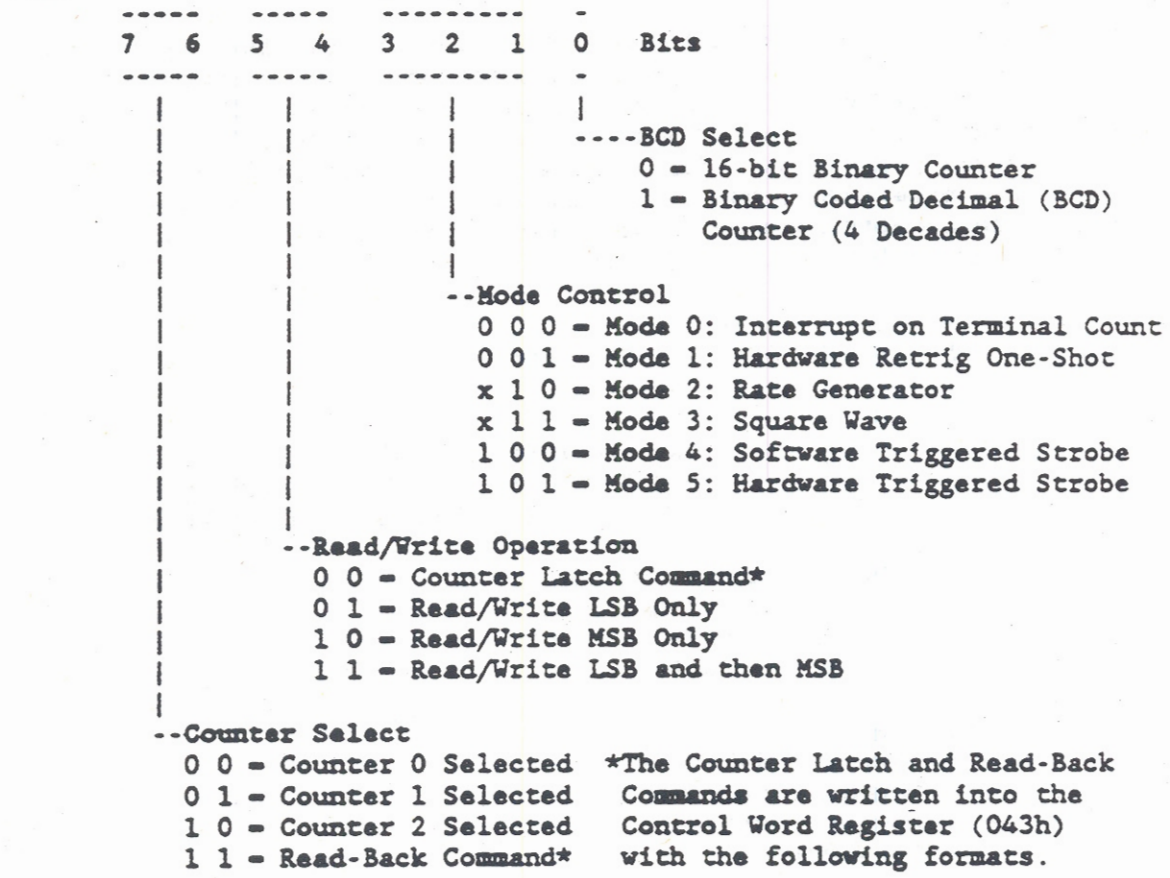
Counter	I/O Address	Purpose
0	040h	Implements the time-of-day clock by sending hardware interrupt IR0 to the Interrupt Controller. The interrupt is accessed through the ROM-BIOS with software interrupt 08h.
1	041h	Routed to the microprocessor control logic to time and request memory refresh cycles.
2	042h	Supports tone generation for the speaker (see Figure 6-1).
-	043h	Counter Control Register

Counter Control Register (043h)

The three counters operate independently, but are programmed via the same Control Word Register, which is accessed through I/O address 043h. Since the register and each of the three counters have separate addresses, and the Control Word specifies which counter it applies to, the required instruction sequence is simplified. Each counter is programmed by first writing the control word and then by loading the initial count. The initial count must be loaded as the least significant byte (LSB) only, the most significant byte (MSB) only, or the LSB followed by the MSB. A new initial count can be loaded into a counter at any time without affecting the programmed mode, but the new count must follow the same format as the initial count. For additional information on the Programmable Timer/Counter, refer the information on 82C54 Programmable Interval Time in the Intel *Microprocessor and Peripheral Handbook, Volume 2*.

The state of the three counters is undefined following a power ON or system reset. Before operating the counters, the operating parameters for the counters are defined by programming the Control Word for each counter. The programming of each counter is identical, but each counter can be programmed to operate in a different mode and format. The programmable counter parameters are as follows:

Counter Control Word Register



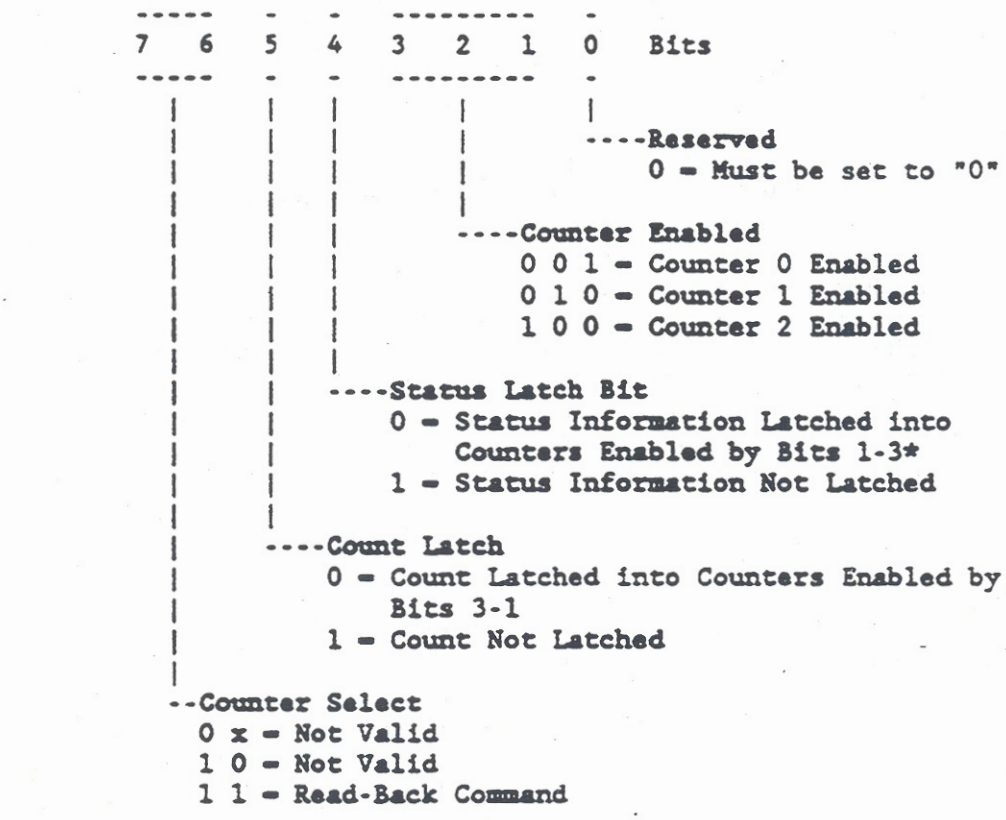
Counter Latch Command

The Counter Latch Command latches the count at the time the command is valid. The count then remains latched until it is read by the microprocessor or until the counter is reprogrammed. The microprocessor must read the count in the same format in which the counter was programmed. For example, if the counter is programmed for a two-byte count, two bytes must be read. After the count is read, the counter returns to programmed operations. The Counter Latch Command format is as follows.

```
-----   -----   -------------
7   6   5   4   3   2   1   0   Bits
-----   -----   -------------
  |       |           |
  |       |           --BCD and Mode Control Bits
  |       |             0 0 0 0 - Not Used but should be "0"
  |       |                       for future compatibility.
  |       |
  |       --Counter Latch Command
  |         0 0 - Latch Command
  |         0 1 - Not Valid
  |         1 x - Not Valid
  |
  |-Counter Select
    0 0 - Counter 0 Selected
    0 1 - Counter 1 Selected
    1 0 - Counter 2 Selected
    1 1 - Not Valid
```

Counter Read-Back Command

The Read-Back Command is used to check the count value, programmed mode, and current state of the counter. The Read-Back Command format is as follows:



*The status information latched by bit 4 is accessed by reading the latched counter. The status information format is read as follows.

Counter Status Information

The Counter Status information must be latched by the Read-Back Command before it is read. Once the status information is latched, it can be read from the Counter Control Word register (043h) as follows.

```
-   -   --------------------
7   6   5   4   3   2   1   0   Bits
-   -   --------------------
|   |             |
|   |             --Counter Programmed Mode
|   |               Data in these bits should appear exactly
|   |               as programmed.
|   |
|   ----Null Count
|       Indicates that the last count written to a Counter
|       Register (CR) was loaded into the Counting Element (CE).
|       If the count is latched or read before it is written
|       into the CE, the count value is not valid.  The null
|       count bit indicates the following.
|       0 = New Count is Loaded into CE
|       1 = Write to Control Word Register or the second byte
|           (MSB) of a two-byte count is written into the CR.
|
|---SPKR
    Indicates the current state of the Counter 2 output pin,
    which is used to drive the speaker.
```

SPEAKER LOGIC

The speaker is controlled at the hardware level by accessing the I/O Register at address 061h and the Programmable Timer/Counter (Counter 2) at I/O address 042h. The speaker logic is shown in Figure 6-1. To determine the frequency applied to the speaker logic, the 1.9 MHz clock input to Counter 2 is divided by the value loaded into the counter. (Refer to the previous paragraphs on programming the counter.) The output of Counter 2 is applied directly to the speaker logic, but has no affect until the speaker is gated on by setting a bit in the I/O Register at 0061h.

The speaker logic is driven in one of three different ways, depending upon bits 0 and 1 of the I/O register at address 0061h, as described in the following list.

Bits 1	0	Operation
0	0	No operation
t	0	Bit 1 is toggled (t) to generate a pulse train to the speaker logic.
1	1	The output of Counter 2 is programmed to generate a waveform to drive the speaker logic.
1	t	Bit 0 is toggled (t) to modulate the input to the PIT and; therefore, the output of Counter 2 is modulated at the speaker logic.

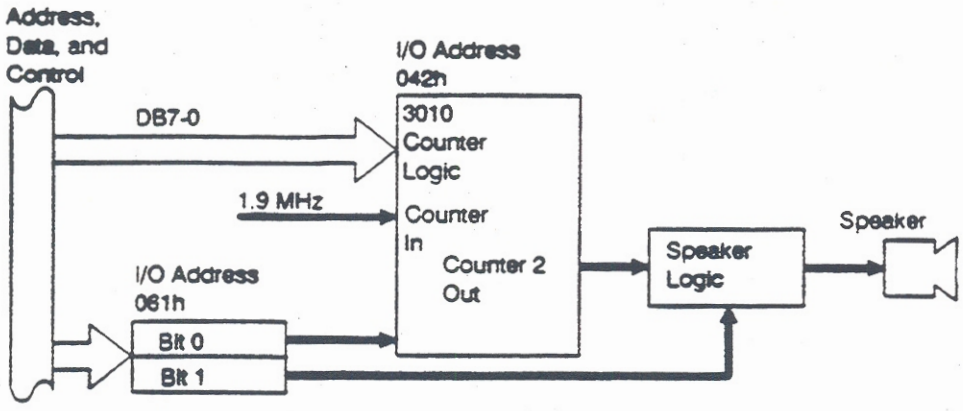


Figure 6-1. GRiDCase 1500 Series Computer Speaker Logic

ROM-BIOS CLOCK SERVICE ROUTINES

The GRiDCase 1500 Series Computer ROM-BIOS provides service routines that support the Time-Of-Day (TOD) Clock. At the ROM-BIOS interface the service routines are compatible with the IBM AT. The ROM-BIOS clock service routines are described in the following paragraphs.

Clock Set or Read (INT 1Ah)

The ROM-BIOS service routine functions at interrupt 1Ah are used to set and read the Time Of Day (TOD) clock and the Real-Time Clock (RTC). The clock set and read routines are summarized in the following list and described in the subsequent paragraphs.

Interrupt (Hex)	Function No. (AH reg)	Description
1A	00	Read current TOD Clock Setting
1A	01	Set TOD Clock
1A	02	Read RTC
1A	03	Set RTC
1A	04	Read Date From RTC
1A	05	Set Date into RTC
1A	06	Set Alarm to Interrupt at Specified Time
1A	07	Reset Alarm Interrupt Function

The microprocessor register input and output operations for the TOD clock service routines are given in the following list. Register AH determines which function within the service routine is enabled, while the other microprocessor registers define the action to be performed. The register contents are specified in hexadecimal (h).

AH = 00h Read Time-Of-Day Clock Setting

On Exit:
- CX = High portion of count
- DX = Low portion of count
- AL = 0 if timer has not passed 24 hours since last read
- AL <> 0 if on another day

AH = 01h Set Time-Of-Day Clock
- CX = High portion of count
- DX = Low portion of count

NOTE: Counts occur at the rate of 1193180/65536 per second or approximately 18.2 counts per second.

AH = 02h Read Real Time Clock

On Exit:
CH = Hours in BCD (00-23)
CL = Minutes in BCD (00-59)
DH = Seconds in BCD (00-60)
DL = Daylight Savings Enable (00-01)

AH = 03h Set Real Time Clock
CH = Hours in BCD (00-23)
CL = Minutes in BCD (00-59)
DH = Seconds in BCD (00-60)
DL = Daylight Savings Enable = 01, Otherwise 00

AH = 04h Read Date From Real Time Clock

On Exit:
CH = Century in BCD (19 or 20)
CL = Year in BCD (00-99)
DH = Month in BCD (01-12)
DL = Date in BCD (01-31)

AH = 05h Set Date Into Real Time Clock
CH = Century in BCD (19 or 20)
CL = Year in BCD (00-99)
DH = Month in BCD (01-12)
DL = Date in BCD (01-31)

AH = 06h Set Alarm to Interrupt at Specified Time
CH = Hours in BCD (00-23 or FFh)
CL = Minutes in BCD (00-59 or FFh)
DH = Seconds in BCD (00-59 or FFh)

AH = 07h Reset the Alarm Interrupt Function

NOTES:

1. For AH = 02h, and 04h, On Exit Carry Flag (CY) = 0 on successful completion, and CY = 1 if the Real Time Clock is not operating.

2. For AH = 06h, the Carry Flag (CY) = 1 if alarm is already enabled. To use the alarm function, a routine must intercept the correct address in the vector table for interrupt 4Ah. Use 0FFh for any "do not care" position for the interval interrupts. Interrupts are disabled during data modification. Refer to the following Alarm Interrupt Handler (INT 70h) description.

3. Registers AH and AL are returned modified and not defined except where indicated otherwise.

Alarm Interrupt Handler (INT 70h)

The ROM-BIOS service routine at interrupt 70h is used to handle the periodic and alarm interrupts for the Real Time Clock (RTC). Interrupt 70h is a hardware interrupt requested through IRQ8. The input frequency is 32 kHz, or approximately 32k interrupts per second for the periodic interrupt. The alarm timing and reset are controlled via interrupt 1Ah.

The periodic and alarm interrupts are enabled when the periodic event or alarm operation is activated. For the periodic event interrupt, the service routine decrements a wait counter. When the count decrements to zero, the service routine sets a designated location to 80h.

For the alarm interrupt, a routine is provided by the device handler to intercept the correct address from the vector table invoked by interrupt 4Ah. The address is intercepted before setting the RTC alarm (refer to Interrupt 1Ah, AH = 06h).

Timer Interrupt (INT 08h)

The ROM-BIOS service routine at interrupt 08h is used to handle the timer interrupt from hardware interrupt request IRQ0. The timer input frequency is 14.3 MHz with a divisor of 65536, which results in approximately 21.8 interrupts per second.

The service routine maintains a count, in location 40:6Ch, of the interrupts since power was turned on. This interrupt count can be used to establish and maintain the time of day.

The service routine also decrements a motor control count in location 40:40h. When the motor control count decrements to zero, the motors in any internal floppy disk drives are turned off and the motor running flags are reset.

The service routine also invokes a routine through interrupt 1Ch. Interrupt 1Ch generates an interrupt at every timer tick. The routine invoked by INT 08h must place the correct address in the vector table. Refer to INT 1Ch in Chapter 3.

REAL-TIME CLOCK

The Real-Time Clock (RTC) is based on the Motorola MC146818 Integrated Circuit (IC) clock, and is functionally identical to the IBM AT clock, but is not compatible with the IBM PC or other models of the GRiDCase computer. The IC Clock circuit is driven by an independent 32.8 kHz oscillator, which provides the time base. The oscillator, the IC Clock circuit, and the associated memory are provided with battery backup so they continue to function when the main power is off. The IC Clock circuit memory consists of 64 bytes of battery backed-up CMOS RAM.

RTC I/O Register Operations

The microprocessor uses two system I/O Registers for access to the registers that are internal to the clock circuit RAM. The two system I/O registers are used to address and to read and write the clock circuit RAM as described in the following paragraphs. The I/O register addresses are given in hexadecimal (h).

Writing Clock Circuit RAM Registers

Writing to the registers in the clock circuit RAM requires two steps as follows:

1. Issue an OUT instruction to I/O address 70h with the address of the RAM register that will be written to.

2. Issue an OUT instruction to I/O address 71h with the data to be written into the previously written address.

Reading Clock Circuit RAM Registers

Reading the clock circuit RAM registers is also a two step process as follows:

1. Issue an OUT instruction to I/O address 70h with the address of the RAM register that will be read from.

2. Issue an IN instruction to I/O address 71h: the data that was read is located in the microprocessor AL register.

Clock Circuit Register Assignments

The following table lists the addresses of the registers that are internal to the clock circuit RAM. Operation of the registers is summarized in following list and the bit definitions for each register are given in Table 6-1.

Register Address	Read/Write	Function
00-0Dh	R/W	Real Time Clock Information
00h	R/W	Seconds in Binary Coded Decimal (BCD)
01h	R/W	Second Alarm in BCD
02h	R/W	Minutes in BCD
03h	R/W	Minute Alarm in BCD
04h	R/W	Hours in BCD
05h	R/W	Hour Alarm in BCD
06h	R/W	Day of Week in BCD
07h	R/W	Date of Month in BCD
08h	R/W	Month in BCD
09h	R/W	Year in BCD
0Ah	R/W	Status Register A
0Bh	R/W	Status Register B
0Ch	R	Status Register C
0Dh	R	Status Register D
0Eh	R/W	Diagnostic Status Byte
0Fh	R/W	Shutdown Status Byte
10h	R/W	Floppy Disk Drive Type Byte Drives A (upper nibble) and B (lower nibble)
11h	R/W	Reserved
12h	R/W	Hard Disk Drive Type Drives C (upper nibble) and D (lower nibble)
13h	R/W	Reserved
14h	R/W	Equipment Byte
15h	R/W	Low Base Memory in kilobytes (kb)
16h	R/W	High Base Memory in kb
17h	R/W	Expanded Memory in kb (low byte)
18h	R/W	Expanded Memory in kb (high byte)
19h	R/W	Drive C Extended Byte
1Ah	R/W	Drive D Extended Byte
1Bh-2Dh	R/W	Reserved
2E-2Fh	R/W	Checksum of bytes 10h through 2Dh
30h	R/W	Low Base Expanded Memory above 1Mb
31h	R/W	High Base Expanded Memory above 1Mb
32h	R/W	Date Century Byte
33h	R/W	Information Flag
34h-3Fh	R/W	Reserved Bytes

Table 6-1. Clock Circuit RAM Register Descriptions

Register (hex)	Bit(s)	Description
00h	7-0	Seconds*
01h	7-0	Second Alarm*
02h	7-0	Minutes*
03h	7-0	Minute Alarm*
04h	7-0	Hours*
05h	7-0	Hour Alarm*
06h	3-0	Day of Week, 1 = Sunday
07h	7-0	Day of Month*
08h	7-0	Month*
09h	7-0	Year(00-99)*

NOTE: * indicates values are in binary or Binary Coded Decimal (BCD). Data mode is selected by Status Register B (0Bh), bit 2.

0Ah		Status Register A
	7	Update in Progress (UIP) 1 = Time is being updated 0 = Time and date are available to be read
	6-4	22 stage divider: Sets the time-base frequency. The system initializes bits 6-4 to 010, which sets the output frequency to 32.768 kHz.
	3-0	Rate selection: Sets the divider output frequency. The system initializes bits 3-0 to 0110, which sets the square-wave output frequency to 1.024 kHz and the periodic interrupt rate to 976.562 usec.

Table 6-1. Clock Circuit RAM Register Descriptions (Continued)

Register (hex)	Bit(s)	Description
0Bh		Status Register B
	7	Set 0 = Update clock once per second 1 = Stop update (for clock setting)
	6	Periodic interrupt enable 1 = Enable
	5	Alarm Interrupt Enable: 1 = Enable
	4	Update Ended Interrupt: 1 = Enable
	3	Square Wave Enabled: 1 = Enable
	2	Data Mode: 0 = BCD, 1 = Binary
	1	12/24 Mode: 0 = 12 Hour,1 = 24 Hour
	0	Daylight Savings Enable 1 = Enable
0Ch		Status Register C
	7	Interrupt Request Bit (IRQF): Read Only 1 = Active
	6	Periodic Interrupt Bit (PF): Read Only 1 = Active
	5	Alarm Interrupt Bit (IRQF): Read Only 1 = Active
	4	Update Ended Bit (IRQF): Read Only 1 = Active
	3-0	Reserved: Should be written as zero
0Dh		Status Register D
	7	Valid RAM Bit (VRB): Read Only 0 = RTC has lost power since this register was last read. A Read sets this bit to 1.
	6-0	Reserved: Should be written as zero.
NOTE:		Addresses 0Eh through 3Fh are configuration information for the CMOS RAM.
0Eh		Diagnostic Status Byte
	7	Real-Time Clock Power 0 = Power On 1 = Power Off
	6	Configuration Record (Checksum Status) 0 = Checksum is good 1 = Checksum is not good
	5	Incorrect configuration information 0 = Valid configuration 1 = Invalid configuration at Power ON.

Table 6-1. Clock Circuit RAM Register Descriptions (Continued)

Register (hex)	Bit(s)	Description
0Eh		Diagnostic Status Byte (Continued)
	4	Memory Size Comparison 0 = Memory size is same size contained in the configuration record. 1 = Memory size is not the same.
	3	Hard Disk/Drive C Initialization Status 0 = Normal Operation 1 = Hard Disk Controller or Drive C failed initialization
	2	Time Status Indicator 0 = Time is Valid 1 = Time is not valid
	1, 0	Reserved
0Fh	7-0	Shutdown Status Byte
10h		Floppy Disk Drive Type
	7-4	First Floppy Disk Drive Type (Drive A) 0000 No Drive Present 0001 Double Sided Drive (48 TPI) 0010 High Capacity Drive (96 TPI) All others reserved
	3-0	Second Floppy Disk Drive Type (Drive B) 0000 No Drive Present 0001 Double Sided Drive (48 TPI) 0010 High Capacity Drive (96 TPI) All others reserved
11h	7-0	Reserved Byte
12h		Hard Disk Drive Type
	7-4	First Hard Disk Drive Type (Drive C) 0000 = No Hard Disk Drive 0001-1110 = Not Supported 1111 = Drive C Extended Byte at 19h
	3-0	Second Hard Disk Drive Type (Drive D) 0000 = No Hard Disk Drive 0001-1110 = Not Supported 1111 = Drive D Extended Byte at 1Ah
13h	7-0	Reserved Byte

Table 6-1. Clock Circuit RAM Register Descriptions (Continued)

Register (hex)	Bit(s)	Description
14h		Equipment Byte
	7-6	Number of Floppy Disk Drives Installed 00 = 1 drive NOTE: Bits 7-6 are 01 = 2 drives valid only when 10 = 3 drives bit 0 = 1. 11 = reserved
	5-4	Primary display 00 = No primary display 01 = Color Graphics in 40 column mode 10 = Color Graphics in 80 column Mode 11 = Monochrome adapter
	3-2	Not used
	1	Coprocessor: 1 = Installed
	0	Floppy Disk: 1 = 1 or 2 installed
15h	7-0	Low Base Memory Byte
16h	7-0	High Base Memory Byte Valid sizes for Low and High Base Memory 0100h = 256k 0200h = 512k 0280h = 640k
17h	7-0	Low Base Expansion Memory Byte
18h	7-0	High Base Expansion Memory Byte Valid sizes for Low and High Base Expansion Memory 0200h = 512k NOTE: Refer also to 0400h = 1024k 30h and 31h. 0600h = 1536k through 3C00h = 15360k (15M maximum)
19h		Drive C Extended Byte
	7-0	First Hard Disk Drive Type (Drive C) 00000000 through 00001111 are reserved 00010000 through 11111111 define drive types 16 through 255 (refer to Table 6-2)

Table 6-1. Clock Circuit RAM Register Descriptions (Continued)

Register (hex)	Bit(s)	Description
1Ah		Drive D Extended Byte
	7-0	Second Hard Disk Drive Type (Drive D) 00000000 through 00001111 are reserved 00010000 through 11111111 define drive types 16 through 255 (refer to Table 6-2)
1Bh-2Dh	7-0	Reserved Bytes
2Eh	7-0	High Byte of Checksum
2Fh	7-0	Low Byte of Checksum
NOTE:	The checksum is calculated on addresses 10h-2Dh.	
30h	7-0	Low Base Expansion Memory Byte
31h	7-0	High Base Expansion Memory Byte Valid sizes for Low and High Base Expansion Memory above 1M byte 0200h = 512k NOTE: Refer also to 0400h = 1024k 17h and 18h. 0600h = 1536k through 3C00h = 15360k (15M maximum)
32h		Date Century Byte
	7-0	BCD value for the century
33h		Information Flag
	7	1 = Top 128k of base memory is installed
	6	Used by setup utility to put out a first user message.
	5-0	Reserved
34h-3Fh		Reserved Bytes

Table 6-2. ROM-BIOS Extended Hard Disk Parameters

Drive Type	No. of Cylinders	No. of Heads	Write Pre-Comp	Landing Zone
16	612	4	All Cyl	663
17	977	5	300	977
18	977	7	None	977
19	1024	7	512	1023
20	733	5	300	732
21	733	7	300	732
22	733	5	300	733
23	306	4	None	336
24-255	Reserved			

CHAPTER 7: DISPLAY SUBSYSTEM

The GRiDCase 1500 Series Computer display subsystem contains a special display controller, display buffer, and control logic that support a choice of internal flat panel displays or an external video display. The internal flat panel displays provide a 10-inch diagonal, full-screen, 80 character by 25 line, 640x400 resolution, bit-mapped graphic display capability in one of the following three formats:

1. Transflective, backlit, supertwist LCD with 11:1 contrast ratio (standard)

2. Gas plasma display with 20:1 contrast ratio (Option 282)

3. Transmissive, backlit, supertwist, Blue LCD display with 12:1 contrast ratio (Option 283)

Similar display capability is available through an external connector to drive a video monitor. The display subsystem uses a Yamaha V6366 Display Controller that emulates the MC6845 CRT controller used by the IBM AT. All of the MC6845 functions are emulated except for the Skew function and for Interlacing in the Video mode. A block diagram of the Display subsystem is provided in Figure 7-1.

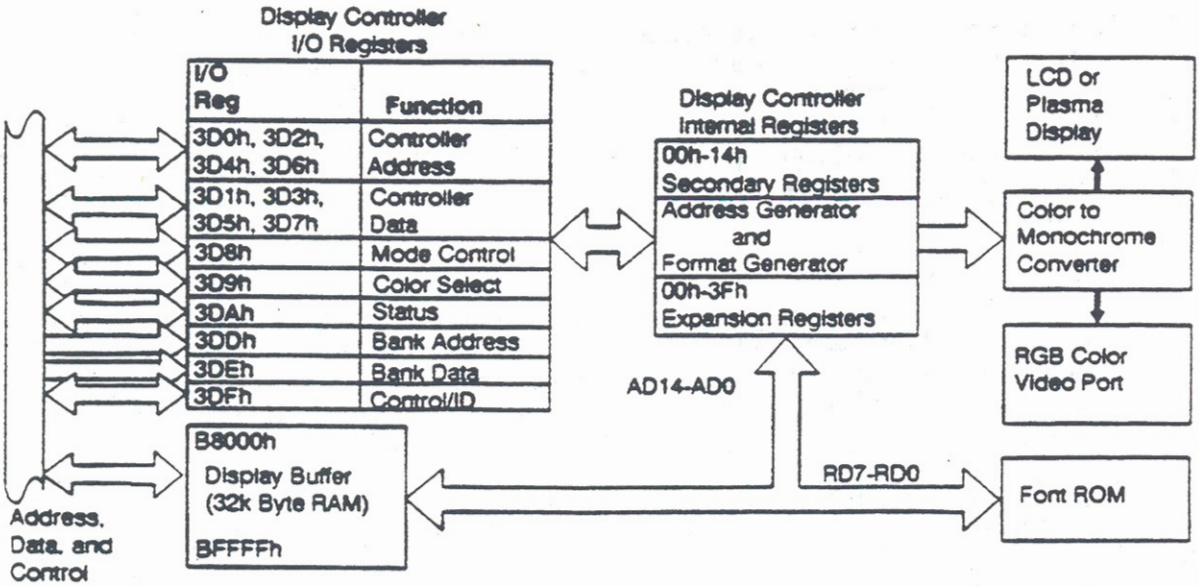


Figure 7-1. GRiDCase 1500 Series Computer Display Subsystem Block Diagram

The display subsystem for the GRiDCase 1500 Series Computer is fully compatible with the IBM Color Graphics Adapter (CGA). To support the CGA video display, the display subsystem contains a 32k byte Static RAM (SRAM) display buffer . The 32k byte SRAM is in addition to, and separate from the RAM that is used for system memory. The separate SRAM display buffer allows the full complement of system memory to be available to the operating systems and application software. The starting address of the CGA video display SRAM is at B8000H and it can be accessed by the microprocessor at any time without disturbing the video signal. An additional 96k byte address space from A0000h through B7FFFh is reserved for EGA and VGA display formats.

With an IBM-type monochrome adapter, the video display begins at address B0000h. However, the GRiDCase 1500 Series computers do not support the IBM monochrome adapter so the starting address is B8000h for all display modes.

Within the display subsystem, the video is converted from the normal RGB format to a monochrome format for driving the built-in flat panel display. The monochrome conversion uses gray scaling and hatching from within the display controller. The same video data in an RGB format is optionally supplied to a 9-pin "D" type connector to support an external CGA compatible video monitor. Gray scaling and hatching are not available at the external connector.

Connections are also provided through the Expansion Bus Interface to connect an optional display controller. The optional display controller supports VGA, EGA, CGA, MDA, and Hercules display modes. The interface is designed to allow a video controller on the expansion bus to coexist with the internal display controller.

The internal display controller supports combinations of display modes for three different screen resolutions, two alphanumeric (text) modes, three graphics modes, and monochrome or color displays. Each combination of the display modes is described briefly in the following paragraphs under display operating modes. Subsequent paragraphs describe each display operating mode in more detail.

For the graphics modes, screen resolution is determined by the number of pixels (light dots) on the display as expressed in horizontal (hor) and vertical (ver) quantities (hor x ver). The three resolutions supported by the controller are 320x200, 640x200, and 640x400. In general, greater resolutions provide clearer displays.

The two alphanumeric modes use either the 40x25 or the 80x25 resolution. In alphanumeric modes, the resolution is determined by the number of horizontal text lines multiplied by the number of character positions on the line. For alphanumeric display modes, the controller also allows for blinking, underline, and boldface characters (intensity).

Mode and color selection for the display controller are accomplished through the Mode Control and Color Select I/O registers. These registers are described in detail in the subsequent paragraphs on programming the display controller.

DISPLAY OPERATING MODES

The display controller for the GRiDCase 1500 Series Computer operates in the following modes:

Alphanumeric Mode (40 x 25 or 80 x 25)

In alphanumeric mode, the data bytes in display memory are interpreted as characters. Each character is made up from a field of 8x16 pixels on the flat-panel display or a field of 8x8 pixels on an external monitor. On the screen, the characters are arranged in 40 columns by 25 lines or in 80 columns by 25 lines. Character 0 is at the top left of the screen and the last character (2,000 for 80 column mode, 1,000 for 40 column mode) is at the lower right corner of the screen. Each character (data byte) has an attribute byte associated with it. The attribute byte specifies foreground color, background color, blinking, and intensity.

320x200 Graphics Mode

In the 320x200 graphics mode, the display memory is arranged as two 8000 byte blocks. The first block contains the even scan lines and the second block contains the odd scan lines. Each pixel on a scan line occupies two bits, and four pixels are represented in a byte.

640x200 Graphics Mode

In the 640x200 Graphics mode the display memory is arranged as two 8000 byte blocks. The first block contains the even scan lines while the second block contains the odd scan lines. Each pixel on a scan line occupies one bit, and eight pixels are represented in a byte.

640x400 Graphics Mode

In the 640x400 Graphics mode the display memory is arranged in four 8000 byte blocks. Each of the blocks contains scan lines in multiples of four. The first block contains even scan lines 0, 4, 8, 12, etc. The second block contains odd scan lines 1, 5, 9, 13, etc. The third block contains even scan lines 2, 6, 10, 14, etc. And, the fourth block contains odd scan lines 3, 7, 11, 15, etc. Each pixel on a scan line occupies one bit.

ALPHANUMERIC MODES

When the display controller is in alphanumeric mode and where character 0 is at the top left of the screen and character *n* is at the lower right corner of the screen, the display memory is interpreted as shown is Figure 7-2. The value of *n* is 1999 for 80 column mode and 999 for 40 column mode. Therefore, in the 80 column mode, each page requires 4000 bytes of display memory. For 40 column mode, each page requires 2000 bytes of display memory. Since only the first 16000 bytes of the 32000 byte display memory are used for alphanumeric characters, it allows for the display of four pages in 80 column mode and eight pages in 40 column mode. The pages are selected by manipulation of the Start Address register. The Start Address register is a secondary register as described in subsequent paragraphs on programming the display controller.

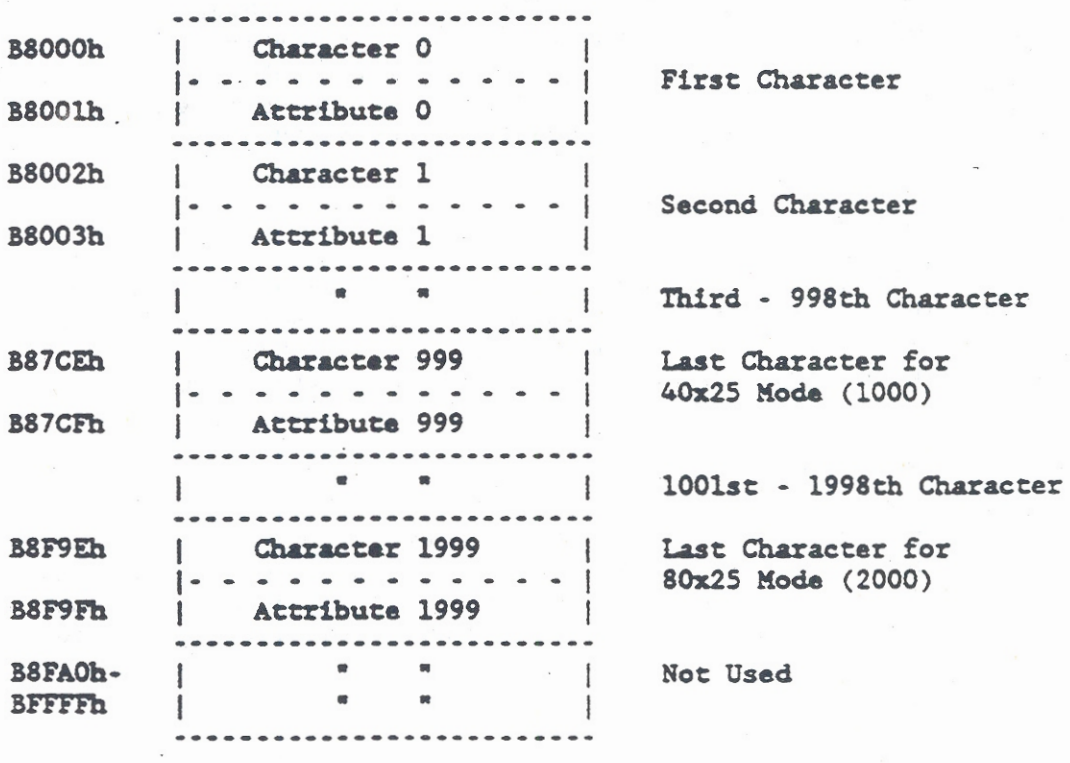
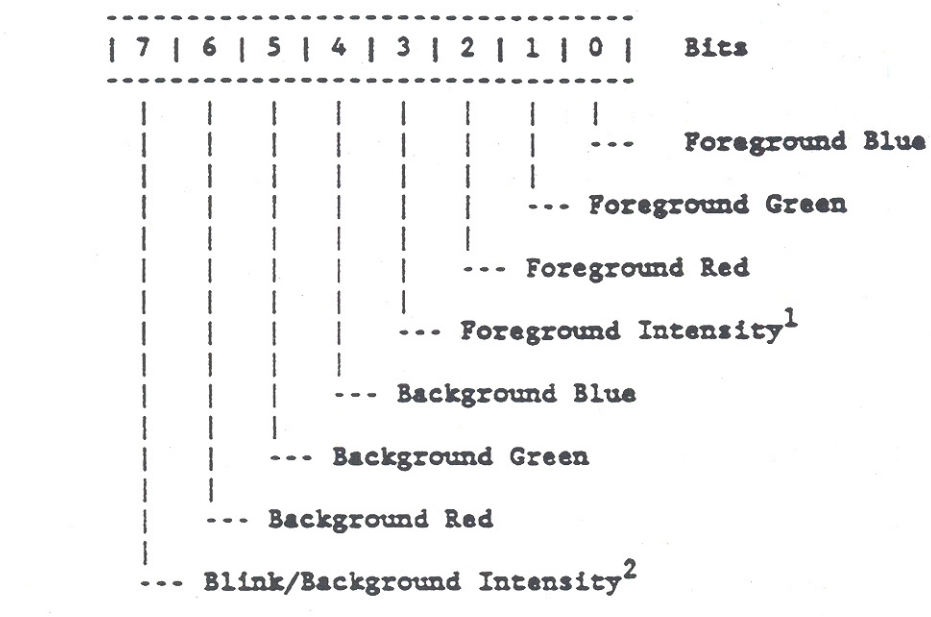


Figure 7-2. Alphanumeric Mode Display Memory Map

Attribute Byte Bit Definitions

When an alphanumeric mode display is selected, each data byte in display memory is interpreted as a character. Each character in display memory has an 8-bit attribute byte associated with it as shown in Figure 7-2.. The attribute byte is defined as follows.



NOTES:

1. Foreground intensity controls normal or bold font. Normal intensity ("0")is displayed as a thin font, and bold ("1") is displayed as a thick font (three pixels wide).

2. Bit 7 controls blinking when Mode Control register bit 5 is set to "1;" otherwise it controls background intensity.

Color-To-Monochrome Converter

For the GRiDCase 1500 Series computers, the color-to-monochrome conversion is controlled through a ROM-BIOS service routine. The service routine allows the selection of one of six color maps. The monochrome color maps are defined in terms of gray scaling at full brightness, two-thirds brightness, one-third brightness, and off. The ROM-BIOS service routines for the display subsystem, and the display subsystem functions for interrupt 15h (INT 15h) are described in subsequent paragraphs.

Also, the display controller registers that control gray scaling and hatching are described in subsequent paragraphs under Programming the Display Controller.

320X200 GRAPHICS MODE

In the 320x200 Graphics mode, 16000 bytes of display memory are used. The bytes are arranged as two 8000 byte blocks (see Figure 7-3). The first block contains the even scan lines and the second block contains the odd scan lines. The byte displayed at the upper left corner on the first scan line (line 0) is located at the start address (B8000h), and successive bytes are displayed left to right, top to bottom on even scan lines. The byte pointed to by the start address plus 8192 is displayed at the upper left corner on the second scan line (line 1) and successive bytes are displayed top to bottom, left to right on odd scan lines.

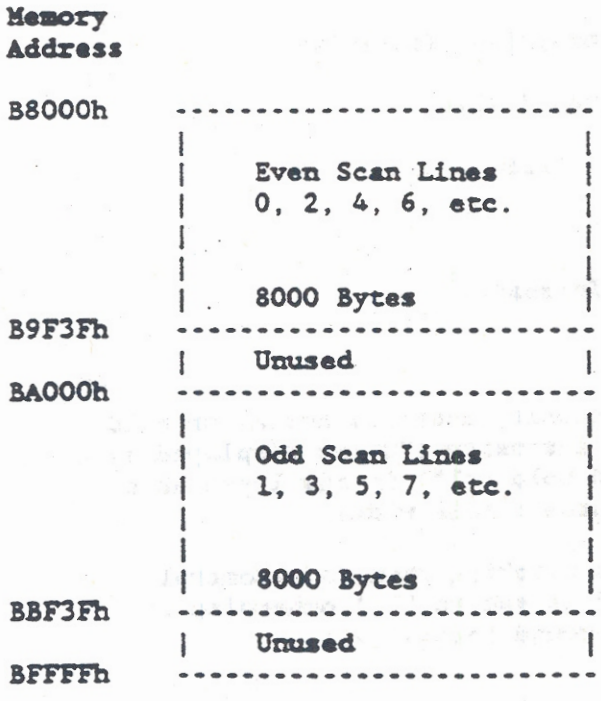


Figure 7-3. 320x200 and 640x200 Graphics Mode Memory Map

For the 320x200 graphics mode, each byte contains four pairs of bits and each pair (C1 and C0) corresponds to one pixel on the screen. Each byte has the following format with the most significant pair being the leftmost pixel on the screen (first displayed of the byte).

7 6	5 4	3 2	1 0	Bits
C1 C0	C1 C0	C1 C0	C1 C0	Bit Definition

Bits C1 and C0 are attribute bits that select one of four colors for display. The colors selected for display are preselected by the Color Select register. Color Select register bits 0-3 select the background colors, bit 4 selects foreground intensity, and bit 5 selects the one of two foreground color palettes as follows:

C1	C0	Color Palette = 0	Color Palette = 1	Monochrome = 1
0	0	Background color	Background color	Background color
0	1	Green	Cyan	Cyan
1	0	Red	Magenta	Red
1	1	Yellow	White	White

The Mode Control register, bit 2, selects the monochrome or color display as shown.

The Color Select and Mode Control registers are described in greater detail in the following paragraphs under Programming the Display Controller.

640X200 GRAPHICS MODE

In the 640x200 Graphics mode, 16000 bytes of the display memory are used. The bytes are arranged in two 8000 byte blocks similar to the 320x200 Graphics mode (see Figure 7-3). The first block contains the even scan lines and the second block contains the odd scan lines. The byte displayed at the upper left corner on the first scan line (line 0) is located at the start address (B8000h), and successive bytes are displayed left to right, top to bottom on even scan lines. The byte pointed to by the start address plus 8192 is displayed at the upper left corner on the second scan line (line 1) and successive bytes are displayed top to bottom, left to right on odd scan lines.

Each bit in a byte corresponds to one pixel on the screen. The most significant bit of each byte is the leftmost bit.

In 640X200 graphics mode, the display pixels have a 1:1 correspondence with each bit of the bytes read from the display memory. When data exists, the color selected by Color Select register bits 0-3 (16 possible colors) is displayed. If no data exists, Color Select register bits 0-3 are set to "0" (0000) and the displayed color is black. The border is also displayed in black. Operation of the Color Select register is described in subsequent paragraphs on programming the display controller.

640X400 GRAPHICS MODE

In the 640x400 Graphics mode, all 32000 bytes of the display memory are used. The bytes are arranged in four 8000 byte blocks (see Figure 7-4). Each of the four blocks contains scan lines in multiples of four.

The byte displayed at the upper left corner on the first scan line (line 0) is located at the start address (B8000h), and successive bytes are displayed left to right, top to bottom on every other even scan line.

The byte pointed to by the start address plus 8192 is displayed at the upper left corner on the second scan line (line 1) and successive bytes are displayed top to bottom, left to right on every other odd scan line.

The byte pointed to by the start address plus 16384 is displayed at the upper left corner on the third scan line (line 2) and successive bytes are displayed top to bottom, left to right on every other even scan line.

The byte pointed to by the start address plus 24576 is displayed at the upper left corner on the fourth scan line (line 3) and successive bytes are displayed top to bottom, left to right on every other odd scan line.

The 640x400 Graphics mode is similar to the 640x200 Graphics mode in that each bit in a byte corresponds to one pixel on the screen. The most significant bit of each byte is the leftmost bit.

The display pixels have a 1:1 correspondence with each bit of the bytes read from the display memory. When data exists, the color selected by Color Select register bits 0-3 (16 possible colors) is displayed. If no data exists, Color Select register bits 0-3 are set to "0" (0000) and the displayed color is black. The border is also displayed in black. Operation of the Color Select register is described in subsequent paragraphs on programming the display controller.

NOTE: The 640x400 graphics mode display is compatible at the ROM-BIOS level with both the AT&T 6300 and Toshiba T3100 computers.

Page 7-9
is missing

ROM-BIOS DISPLAY SERVICE ROUTINES

The GRiDCase 1500 Series Computer ROM-BIOS provides the following IBM AT compatible software interrupt service routines to handle the display.

Interrupt (Hex)	Function No. (AH reg)	Description
10	00	Set Display Mode
10	01	Set Cursor Type
10	02	Set Cursor Position
10	03	Read Cursor Position
10	04	Read Light Pen Position (Not Supported)
10	05	Select Active Display Window (Alpha Mode)
10	06	Scroll Active Window Up
10	07	Scroll Active Window Down
10	08	Read Character and Attribute at Cursor
10	09	Write a Character and Attribute at Cursor
10	0A	Write a Character (No Attribute) at Cursor
10	0B	Set Color Palette
10	0C	Write a Pixel to the Display
10	0D	Read a Pixel from the Display
10	0E	Write TTY to Active Window
10	0F	Read Current Display State
10	10-12	Reserved
10	13	Write Character String

Display Service Routine Descriptions

The 80286/80386 Microprocessor register inputs and outputs for the display interrupt service routines are listed in the paragraphs that follow. In all of the routines, the contents of the following registers are preserved: SP, BP, DI, SI, BX, CX, DX, SS, DS, and ES. In all other registers, the contents are destroyed. Register AH determines the function that is invoked within the service routine, while the other microprocessor registers further define the action that is performed. All register contents are given in hexadecimal (h). For details on modes and controller internal register settings that result from the routines, refer to the Controller Register Descriptions provided later in this chapter.

NOTE: The 640x400 Graphics mode display is compatible at the ROM-BIOS level with both the AT&T 6300 and the Toshiba T3100 computers.

Page 7-11
is missing

AH = 07h Scroll Active Window Down
AL = Number of lines, input lines blanked at window top
AL = 00h specifies blank entire window
BH = Attribute to be used on blank lines
CH, CL = Row, column of upper left corner of scroll
DH, DL = Row, column of lower right corner of scroll

AH = 08h Read Attribute/Character at Current Cursor Position*
BH = Display window (valid for alpha modes only)
On Exit:
AL = Character read
AH = Attribute of character read (alpha modes only)

AH = 09h Write Attribute/Character at Current Cursor Position*
AL = Character to write
BH = Display window (valid for alpha modes only)
BL = Attribute of character for alpha mode or
Color of character for graphics mode[1]
If Bit 7=1, attribute is XORed with existing attribute
CX = Count of characters to write

AH = 0Ah Write Character Only at Current Cursor Position*
AL = Character to write
BH = Display window (Valid for alpha modes only; for)
graphics modes use AH = 09h)
CX = Count of characters to write

NOTE:

* In graphics modes, characters are formed in a character generator image located in system ROM. The character generator contains the first 128 characters. To read and write an additional 128 characters, initialize a pointer at 00:1Fh (address 0007Ch). The pointer should point to the 1k byte table containing code points for the second 128 characters (128-255).

Also, to write a character in graphics mode, the replication factor contained in register CX produces valid results on entry for characters contained on the same row. Continuation to a succeeding line does not operate.

AH = 0Bh Set Color Palette
BH = Palette color ID being set (0-127)
BL = Color value to be used with the color ID
Applies only to 320x200 color graphics mode
Color ID = 0 Selects background color (0-15)
Color ID = 1 Selects palette to be used
0 = Green(1)/Red(2)/Yellow(3)
1 = Cyan(1)/Magenta(2)/White(3)
In alpha modes, Color ID 0 sets border color to use (values 0-31); where 16-31 select the high-intensity background set

AH = 0Ch Write Pixel
AL = Color value. If bit 7 is set, color value is XORed with current pixel contents
CX = Column number
DX = Row number

AH = 0Dh Read Pixel (Graphics Modes only)
CX = Column number (1-639)
DX = Row number (1-639)
On Exit:
AL = Color Value
For 640 x 200/400 modes, AL = ON/OFF.
For 320 x 200 modes, AL is two bits as follows:
0 0 = black
0 1 = Color depends upon selected palette (AH=0Bh)
1 0 = Color depends upon selected palette (AH=0Bh)
1 1 = Color depends upon selected palette (AH=0Bh)

AH = 0Eh Write TTY to Active Window
AL = Character to write
BL = Foreground color in graphics in mode
Screen width is controlled by previous mode set.

AH = 0Fh Return Current Display State

On Exit:
AL = Mode currently set by register AH = 00h
AH = Number of character columns on the screen
BH = Currently active display window

AH = 10h-12h Reserved

AH = 13h Write Character String
AL = 0-3 For Type of String (See Note)
BH = Page Number
CX = Length of Character String
DX = Cursor Position for Character String
CS:BP = Pointer to Character String

NOTE: Specify the type of Character String as follows:

AL = 00h Write Character String
BL = Attribute
String is |char,char,...,char|
Cursor does not move from current position
AL = 01h Write Character String and Move Cursor
BL = Attribute
String is |char,char,...,char|
Cursor moves
AL = 02h Write Character and Attribute String
(Alpha Mode Only)
String is |char,attr,char,attr,...,char,attr|
Cursor does not move from current position
AL = 03h Write Character and Attribute String and Move Cursor (Alpha Mode Only)
String is |char,attr,char,attr,...,char,attr|
Cursor moves

Return, Line Feed, BkSp (Backspace), and Bell (tone) are treated as commands and not as printable characters.

Display Subsystem Functions

In addition to the Display Service Routines, BIOS subsystem functions were added to the ROM-BIOS to provide a logical interface for manipulating GRiD hardware specific functions. The subsystem functions currently defined for the Display Subsystem are as follows:

1. Select or Get Current Display (Function 20h)

2. Select or Get Color Map (Function 21h)

3. Select Current Display Font (Function 22h)

4. Turn Backlight ON or OFF (Function 23h)

5. Functions 24h-3Fh are reserved for future display functions.

To perform a subsystem function, an interrupt 15h is issued with the subsystem identifier, 0E4h, loaded in the AH register and the desired Display Subsystem Function number in the AL register. The combination of interrupt 15h and subsystem function identifier 0E4h

provide access to the BIOS subsystem functions. Once inside the subsystem, the specific function to be performed is selected by the function number in the AL register. The current Display Subsystem Functions are described in the following paragraphs:

Select or Get Current Display (Function 20h)

Function 20h selects either the internal display panel or the external RGB video port, or returns the currently active display. The value entered in DL register determines the display to be selected. When value 0FFh is entered in the DL register, the current display is returned. The value returned for the internal panel is the panel type.

Entry:

AH = 0E4h		Selects the ROM Subsystem
AL = 20h		Display Subsystem Function number
DL = 0		Selects the external RGB video port
= 1		Selects the internal panel
= 0FFh		Returns the display type

Exit:

Carry Flag = 0 (no error)

If 0FFh was entered in DL, then:

DL = 0	External video port is selected
DL = 1	Internal 640x200 plasma panel selected
DL = 5	Internal 640x400 plasma panel selected
DL = 7	Internal 640x400 LCD panel selected

Carry Flag = 1 (error returned)

AH = 86h	Subsystem not supported

NOTE: The internal display is also controlled by the MS-DOS MODE commands DISPLAY = {ON|OFF}

Select or Get Color Map (Function 21h)

Function 21h is used to select one of six predefined color maps and is also used to activate the currently selected color map. The color map index is stored at absolute address 40:A7h. The value entered in the DL register determines which color map is selected. When the value 0FFh is entered in the DL register, the currently selected color map is activated. The six predefined color maps are as follows:

NOTE: MAP 0 through MAP 5 in the following list correspond to MS-DOS MODE commands COLORMAP = 1-6, respectively.

Color	Map 0	Map 1	Map 2	Map 3	*Map 4	*Map 5
Black	Off	Off	Off	Off	Off	Off
Blue	Off	2/3	2/3	Off	Full	Full
Green	2/3	2/3	1/3	1/3	Full	Full/ul
Cyan	2/3	2/3	1/3	1/3	Full	Full
Red	1/3	1/3	2/3	2/3	Full	Full
Magenta	1/3	1/3	2/3	2/3	Full	Full
Brown	Full	Full	Full	Full	Full	Full
White	Full	Full	Full	Full	Full	Full

NOTES: The color maps are defined in terms of the gray scale as follows:
Off - nothing displayed
1/3 - one third bright
2/3 - two thirds bright
Full - full brightness
/ul - underlined
* - inverse video is displayed when the background color is white

Entry:

AH = 0E4h	Selects the ROM Subsystem
AL = 21h	Display Subsystem Function number
DL = nn	Enter 0 - 5 to select the color map or enter 0FFh to return the current color map.

Exit:
Carry Flag = 0 (no error)
If 0FFh was entered in DL, then:
DL = Currently active color map
Carry Flag = 1 (error returned)

AH = 2	Color map value out of range
= 86h	Subsystem not supported

Select Current Display Font (Function 22h)

Function 22h selects one of the four fonts that are available in the hardware font generator. The four available fonts and their selection values are as follows:

Font Name	Value
English	0
French Canadian	1
Norwegian	2
Hebrew	3

NOTE: Font values 0-3 correspond to MS-DOS MODE commands FONT = 1-4, respectively.

Entry:

AH = 0E4h		Selects the ROM Subsystem
AL = 22h		Display Subsystem Function number
DL = nn		Enter 0 - 3 to select the font number

Exit:

Carry Flag = 0 (no error)

No significant return

Carry Flag = 1 (error returned)

AH = 2	Font value out of range
= 86h	Subsystem not supported

Set Backlight Time Out (Function 23h)

Function 23h allows the backlight on an internal LCD display panel to be turned ON and OFF directly or turned on and timed to turn off after some period of keyboard inactivity. This function has no effect on a plasma display.

Entry:

AH = 0E4h	Selects the ROM Subsystem
AL = 23h	Display Subsystem Function number
DL = 0	Turns OFF the backlight
= 1-60	Turns ON the backlight and starts the timer to turn it OFF if the keyboard is inactive for a period of 1 to 60 (decimal) minutes.
DL = 0FFh	Turn ON backlight with no time out.

NOTE: The DL register values correspond to MS-DOS MODE commands BACKLITE = {1-60|ON|OFF}, respectively.

Exit:

Carry Flag = 0 (no error)

No significant return

Carry Flag = 1 (error returned)

AH = 86h	Subsystem not supported

PROGRAMMING THE DISPLAY CONTROLLER

The GRiDCase 1500 Series Computer uses a Yamaha V6366 Display Controller. All of the display controller operations are controlled through I/O Registers. The following paragraphs describe the registers in the controller that are available to interface directly with computer display logic. The direct interface bypasses the ROM-BIOS service routines. There are three sets of display controller registers, which are described in the following paragraphs. For additional information on the display controller, refer to the *Yamaha PCDC V6366 Application Manual*.

1. The primary set of registers are I/O Registers that provide a macro level interface to the controller logic. The primary register functions include address, data, mode, color, status, and control/ID.

2. A secondary set of registers provide a micro level control of display characteristics. The secondary register functions include horizontal and vertical timing, raster address, and cursor control.

3. The third set of registers provide expansion functions used for controlling flat-panel (LCD or Plasma) displays. Many of the expansion register functions are similar to the secondary register functions. In addition, the Expansion Function registers can be preset to allow standard 6845 values to be programmed into the secondary registers, but still have them work with the flat-panel displays.

Display Controller Primary Registers

The following I/O addresses, in hexadecimal (h), reference the display controller primary registers:

Address	Register
3D0h, 3D2h, 3D4h, 3D6h	Controller Address Registers (write)
3D1h, 3D3h, 3D5h, 3D7h	Controller Data Registers (read/write)
3D8h	Mode Control (read/write)
3D9h	Color Select (read/write)
3DAh	Status (read only)
3DBh	Light pen strobe clear (not supported)
3DCh	Light pen strobe set (not supported)
3DDh	Bank Address (write)
3DEh	Bank Data (write)
3DFh	Control/ID (read/write)

All of the Primary registers are cleared by a system reset and three of the registers (Mode Control, Color Select, and Control/ID) cannot be read unless the Read I/O Registers (RREG) bit is set. The RREG bit is set under normal operation. Operation of the RREG bit is described later in this chapter (refer to Table 7-6).

The main control registers are the Mode Control Register that determines the display operating mode, and the Color Select register. The Color Select register sets up the background and border colors and also selects which colors are displayed in graphics modes. The Status register provides information about the state of the controller.

The Controller Address registers and Controller Data registers provide the means of accessing the Secondary registers. The Secondary register are internal to the controller. The internal Secondary register provide the horizontal and vertical timing characteristics and also the cursor location and characteristics for the display.

The Bank Address and Bank Data Registers provide access to the Expansion Function registers that are internal to the controller. The Expansion Function registers allow alternate values for display operation to be stored and then quickly accessed as required. These alternate values are then substituted for standard 6845 values contained in the Primary and Secondary registers. The Expansion Function register values are used for operation of the LCD and plasma flat-panel displays.

The Control/ID register controls access to the Expansion Function registers and to certain bits of the Mode Control register. Also, the display controller identification (ID) code can be read by this register.

The following paragraphs provide more detailed descriptions of the display controller primary registers

Mode Control Register (03D8h)

The Mode Control Register at I/O address 3D8h determines the operating mode of the display controller. The register is cleared by a system reset and is read only if the RREG bit is set. The RREG bit is normally set. Operation of the RREG bit is described later in this chapter (refer to Table 7-6). The Mode Control register uses six of its eight bits to select the alphanumeric (alpha) or graphics modes, screen resolution, and monochrome or color display.

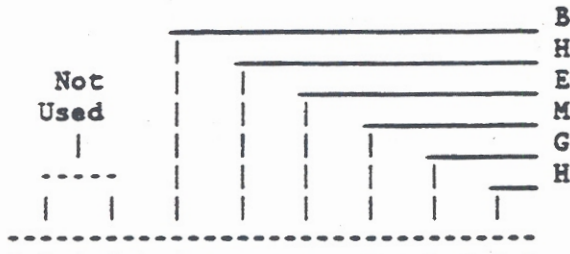
For alpha modes, the Mode Control Register allows selection of either 40 or 80 characters (or columns) in each row and the maximum of 25 rows. This provides a maximum of either 1000 or 2000 character boxes per screen page. Each character box consists of 8 by 16 pixels (dots) for the internal displays or 8 by 8 pixels for the external video display. Since only the first half of the 32k-byte Display Buffer is used to contain alphanumeric video, the buffer can contain up to eight screen pages in the 25 row by 40 character alpha mode. In the 25 row by 80 character alpha mode, the buffer can contain up to four screen pages.

For graphics modes, the Mode Control Register allows the selection of display resolution based on the number of pixels (light dots) that can be displayed on the screen. Resolution is expressed in the number of horizontal pixels (hor) multiplied by the number of vertical (ver) pixels (hor x vert). The Mode Control register allows selection of two resolutions in graphics mode. The two resolutions are 320x200 and 640x200. A third graphics mode

resolution (640x400) shares the same Mode Control register bit pattern as the 640x200 resolution. The actual resolution is then determined by the Graphics Mode Control (GRM2-GRM0) register (refer to Table 7-6).

The Mode Control Register also allows selection of monochrome or color display in both the alpha and graphics modes. With a monochrome adapter, the video display would usually begin at address B0000h. GRiDCase 1500 Series computers use a CGA compatible adapter with the starting address at B8000h for all display modes. The monochrome (black and white) modes operate in the same way as the color modes except that the color burst signal is not enabled.

The Mode Control Register bits and their use to program the display controller in its various modes are as follows:



7	6	5	4	3	2	1	0	Bit
								Display Mode
		1	x	1	1	0	0	40x25 Alphanumeric Monochrome
		1	x	1	0	0	0	40x25 Alphanumeric Color
Not	Used	1	x	1	1	0	1	80x25 Alphanumeric Monochrome
		1	x	1	0	0	1	80x25 Alphanumeric Color
		x	0	1	1	1	0	320x200 Monochrome Graphics
		x	0	1	0	1	0	320x200 Color Graphics
		x	1	1	1	1	0	640x200 Monochrome Graphics

NOTES: The 640x200 and 640x400 Graphics Modes share the same bit patterns in the Mode Control Register. The screen resolution is determined by the Graphics Mode Control (GRM2-GRM0) register (refer to Table 7-6).

The Blink Enable bit (5) enables blinking in alphanumeric modes.

The Monochrome Select bit (2) only affects 320x200 graphics mode and makes the blue and green outputs both have the same value as the least significant bit of the pixel attributes. It has no other affect.

The Graphics Mode Select bit (1) is enabled by the Control/ID register described later in this chapter.

Color Select Register (03D9h)

The Color Select Register at I/O address 3D9h specifies the display border color in alpha modes or the selectable color during graphics modes. The register can be read when the RREG bit is set to "1." Operation of the RREG bit is described later in this chapter (refer to Table 7-6). Six bits of the 8-bit Color Select register are used and all of the bits are read/write (as opposed to write-only in the IBM PC). The functions performed by the register bits will vary depending upon the operating mode as listed in Table 7-1.

Table 7-1. Color Select Register Functions

Bit #	Alphanumeric Modes	320x200 Graphics Mode	High Resolution Graphics Modes
Bit 7	Reserved	Reserved	Reserved
Bit 6	Not Used	Not Used	Not Used
Bit 5	Not Used	Color Palette	Not Used
Bit 4	Not Used	Foreground Intensity	Not Used
Bit 3	Border Intensity	Background Intensity	Intensity
Bit 2	Red Border	Red	Red
Bit 1	Green Border	Green	Green
Bit 0	Blue Border	Blue	Blue

Color Select register bits 0 through 3 select blue, green, and red color and intensity, respectively. In alpha modes, bits 0 through 3 specify the color of the border area around the outside edge of the screen. The output of the Color Select Register is "0" during these "border" periods to create a black border.

In the 320x200 graphics mode, Color Select register bits 0-3 specify the color, bit 3 specifies the background intensity, bit 4 specifies the foreground intensity, and bit 5 selects one of two color palettes. Each color palette provides four preset colors. The foreground intensity, bit 4, can be set to "1" to provide additional colors in each palette. The Mode Control register, bit 2, is also used to select the monochrome or color mode. When Mode Control Register bit 2 is set to "1," both color palettes display the same colors.

Each color in the selected palette is then selected for display by the attribute bits C1 and C0, which were previously described. The colors selected by the attribute bits are indicated in the following list:

C1	C0	Color	Bit 5 = 0 I	R	G	B	Bit 5 = 1 I	R	G	B	Mode Bit 2 = 1 I	R	G	B
0	0	Background/ Border	3	2	1	0	3	2	1	0	3	2	1	0
0	1	Color 1	x	0	1	0	x	0	1	1	x	0	1	1
1	0	Color 2	x	1	0	0	x	1	0	1	x	1	0	0
1	1	Color 3	x	1	1	0	x	1	1	1	x	1	1	1

NOTE: The x represents Color Select Register bit 4, which selects the foreground color intensity. Additional colors can be obtained by changing the intensity.

When C1, C0 = 00 and the Border Inhibit (BINH) bit = 0, the color selected by bits 3-0 is the background and border color . If the BINH bit is set to "1," the border is displayed in black (IRGB = 0000). For a description of the BINH bit refer to Table 7-2.

The four color bits (0-3) are set in combinations to produce the colors listed in Table 7-2.

Table 7-2. Summary of Available Colors

3 2 1 0 I R G B	Bit (39Dh) COLOR
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	Light gray
1 0 0 0	Dark 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	White

For high resolution modes (640x200, 640x400) the each displayed pixel has a 1:1 correspondence with each bit of the bytes read from display memory (SRAM). When data exists, one out the 16 possible colors is selected for display by Color Select register bits 3-0. If no data exits, both the display area and the border are displayed in black (IRGB = 0000).

Display Controller Status Register (03DAh)

The Controller Status Register at I/O address 3DAh contains status information indicating the current activity of the display controller.

The Status register bits provide the following information:

Bit 7	Not Used
Bit 6	Not Used
Bit 5	Not Used
Bit 4	Not Used
Bit 3	Vsync (Vertical sync)
Bit 2	Light pen switch (not supported - always 0)
Bit 1	Light pen strobe (not supported - always 0)
Bit 0	Dspen (Display Enable)

In the IBM PC, the Vsync and Dspen status bits (3 and 0, respectively) can be examined by application programs to determine when data in the screen buffer can be updated without interfering with the display of data on the screen. The screen buffer in the GRiDCase 1500 Series Computer can be altered at any time without disturbing screen data.

Controller Address and Data Registers (03D4h and 03D5h)

The Controller Address register (03D4h) is used to select one of the Internal Secondary registers that program the controller for the desired line widths, scan lines per character row, sync pulse positions and widths, displayed and blanking times per line and frame, etc. Once the Internal Secondary register is selected, the Controller Data register (03D5h) is used to write into or read from the selected register. The Internal Secondary registers are all read/write buffers, and are listed in Table 7-3. Subsequent paragraphs under Display Controller Secondary Registers describe the operation of each of the Secondary register.

Table 7-3. Display Controller Internal Secondary Registers

Reg No.	Secondary Registers Address (Hex)	Name	Default Settings (Hex) Alpha 40x25	Alpha 80x25	Graphics 320/640
R0	00	Horizontal Total	38	71	38
R1	01	Horizontal Displayed	28	50	28
R2	02	Horizontal Sync Position	2D	5A	2D
R3	03	Sync Pulse Width	06	0A	0A
R4	04	Vertical Total	1F	1F	7F
R5	05	Total Raster Adjust	06	06	06
R6	06	Vertical Displayed	19	19	64
R7	07	Vertical Sync Position	1C	1C	70
R8	08	Interlace Mode	02	02	02
R9	09	Maximum Raster Address	07	07	01
R10	0A	Cursor Start and Mode	06	06	06
R11	0B	Cursor End	07	07	07
R12	0C	Start Address (high)	00	00	00
R13	0D	Start Address (low)	00	00	00
R14	0E	Cursor Address (high)	xx	xx	xx
R15	0F	Cursor Address (low)	xx	xx	xx
R16	10	Light Pen, (high)	xx	xx	xx
R17	11	Light Pen (low)	xx	xx	xx

NOTE: All graphics modes use the same default settings.

Access to the Internal Secondary registers is provided by writing the address of the desired register to the Controller Address register at I/O address 3D4h. Then, data can be written to or read from the specified Internal Secondary register by using OUT or IN instructions directed to the Controller Data register at I/O address 3D5h. Additional Controller Address registers are located at I/O addresses 3D0h, 3D2h, and 3D6h. Additional Controller Data registers are located at I/O addresses 3D1h, 3D3h, and 3D7h. These additional Address and Data registers operate in an identical manner, but are completely separate from the previously described Address (3D4h) and Data (3D5h) registers.

Bank Address and Bank Data Registers (03DDh and 03DEh)

The Bank Address (03DDh) and Bank Data (03DEh) Registers are write-only registers that provide access to the Expansion Function registers that are internal to the controller. The Expansion Function registers provide the control for the internal LCD and plasma flat-panel displays.

The Bank Address register (03DDh) is used to select the first address of a register bank that contains a series of Expansion Function registers. Each time data is written to the selected address, the Bank Address register automatically increments to the next address in the Expansion Function register bank.

The Bank Data register (03DEh) is used to write data into the selected Expansion Function registers. Data written into the Expansion Function registers include a 16x16-bit color palette, 12x16-bit Color Graphics Adapter (CGA) registers, and 6845 Preset Emulation registers for graphics and alphanumerics.

The Expansion Function registers are all read/write buffers, and are listed in their hexadecimal (h) address sequence in Table 7-4. Descriptions of the Expansion Function registers are provided in subsequent paragraphs of this chapter.

Table 7-4. Display Controller Expansion Function Register Banks

Address (Hex)	Data D7	D6	D5	D4	D3	D2	D1	D0	Description
00	x	x	x	x	x	R2	R1	R0	Color Palette 0
01	x	G2	G1	G0	x	B2	B1	B0	Color Palette 0
02	x	x	x	x	x	R2	R1	R0	Color Palette 1
03	x	G2	G1	G0	x	B2	B1	B0	Color Palette 1
04	x	x	x	x	x	R2	R1	R0	Color Palette 2
05	x	G2	G1	G0	x	B2	B1	B0	Color Palette 2
06	x	x	x	x	x	R2	R1	R0	Color Palette 3
07	x	G2	G1	G0	x	B2	B1	B0	Color Palette 3
08	x	x	x	x	x	R2	R1	R0	Color Palette 4
09	x	G2	G1	G0	x	B2	B1	B0	Color Palette 4
0A	x	x	x	x	x	R2	R1	R0	Color Palette 5
0B	x	G2	G1	G0	x	B2	B1	B0	Color Palette 5
0C	x	x	x	x	x	R2	R1	R0	Color Palette 6
0D	x	G2	G1	G0	x	B2	B1	B0	Color Palette 6
0E	x	x	x	x	x	R2	R1	R0	Color Palette 7
0F	x	G2	G1	G0	x	B2	B1	B0	Color Palette 7
10	x	x	x	x	x	R2	R1	R0	Color Palette 8
11	x	G2	G1	G0	x	B2	B1	B0	Color Palette 8
12	x	x	x	x	x	R2	R1	R0	Color Palette 9
13	x	G2	G1	G0	x	B2	B1	B0	Color Palette 9
14	x	x	x	x	x	R2	R1	R0	Color Palette 10
15	x	G2	G1	G0	x	B2	B1	B0	Color Palette 10
16	x	x	x	x	x	R2	R1	R0	Color Palette 11
17	x	G2	G1	G0	x	B2	B1	B0	Color Palette 11

Table 7-4. Display Controller Expansion Function Register Banks (Continued)

Address (Hex)	D7	D6	D5	D4	D3	D2	D1	D0	Description
18	x	x	x	x	x	R2	R1	R0	Color Palette 12
19	x	G2	G1	G0	x	B2	B1	B0	Color Palette 12
1A	x	x	x	x	x	R2	R1	R0	Color Palette 13
1B	x	G2	G1	G0	x	B2	B1	B0	Color Palette 13
1C	x	x	x	x	x	R2	R1	R0	Color Palette 14
1D	x	G2	G1	G0	x	B2	B1	B0	Color Palette 14
1E	x	x	x	x	x	R2	R1	R0	Color Palette 15
1F	x	G2	G1	G0	x	B2	B1	B0	Color Palette 15
20	Duty (DUTY)								Address 20h
21	Horizontal Size (HSIZE)								Address 21h
22	Vertical Adjust (VADJ)								Address 22h
23	SCT	SCU	Horizontal Adjust (HADJ)						Address 23h
24	MON	ULE	GRM2	GRM1	GRM0	FON2	FON1	FON0	Address 24h
25	SYTH	HERC	SRAM	8BUS	16CP	BINH	CLK1	CLK0	Address 25h
26	RA4	A15	EXP	PB1	PB0	PAG2	PAG1	PAG0	Address 26h
27	LS1	LS0	IEN	EXTS	IINH	ACM	CPLE	PRE	Address 27h
28	SWC	VST	SSY	SSCK	H/E	R/M	SCR1	SCR0	Address 28h
29	STBY	RREG	ESIO	EH4	EH3	EH2	EH1	EH0	Address 29h
2A	TST2	TST1	TST0	SS4	SS3	SS2	SS1	SS0	Address 2Ah
2B	CON7	CON6	CON5	CON4	CON3	CON2	CON1	CON0	Address 2Bh
30	Horizontal Total (R0)								Alpha Preset (PRE)
31	Horiz Sync Position (R2)								Alpha Preset (OS)
32	Vertical Sync Position (R7)								Alpha Preset (OS)
33	Sync Pulse Width (R3)								Alpha Preset (PRE)
34	Vertical Total (R4)								Alpha Preset (PRE)
35	x	x	x	Total Raster Adjust (R5)					Alpha Preset (PRE)
36	x	x	x	Max Raster Address (R9)					Alpha Preset (PRE)
37	F/O	TDS1	TDS0	Cursor Addr Offset (CA0)					Alpha Preset (OS)
38	Horizontal Total (R0)								GraphicsPreset(PRE)
39	Horiz Sync Position (R2)								GraphicsPreset (OS)
3A	Vertical Sync Position (R7)								GraphicsPreset (OS)
3B	Sync Pulse Width (R3)								GraphicsPreset(PRE)
3C	Vertical Total (R4)								GraphicsPreset(PRE)
3D	x	x	x	Total Raster Adjust (R5)					GraphicsPreset(PRE)
3E	x	x	x	Max Raster Address (R9)					GraphicsPreset(PRE)
3F	45S	GDS1	GDS0	Underline Position (UP)					GraphicsPreset(PRE)

NOTES: Two registers are required for each of the 16 color palettes. Data must be written to the even address before the odd address for each color palette.

The "x" in any bit position indicates "don't care."

Access to the Expansion Function registers listed in Table 7-4 is provided by first writing an 80h to the Control/ID register (03DFh) to gain access to the Bank Address and Bank Data Registers (03DDh and 03DEh, respectively). Then, the address of the desired Expansion Function register is written to the Controller Bank Address register at I/O address 3DDh. Finally, Data can be written to or read from the specified Expansion Function register by using OUT or IN instructions directed to the Controller Bank Data register at I/O address 3DEh. Each time an Expansion Function register is written to, the address register (3DDh) increments to the next address. The display controller Expansion Function registers are described in subsequent paragraphs of this chapter.

Control/ID Register (03DFh)

The Control/ID register (03DFh) controls access to the Expansion Function registers and to certain bits of the Mode Control register (03D8h). Also, the display controller identification (ID) code can be read by this register. The Control/ID register is cleared by any controller reset and the data bits read from the register are dependent upon the state of Expansion Function register 29h, bit 6 (RREG). When RREG = 1, the control functions are read from the Control/ID register. If RREG = 0, the ID functions are read.

Because of their relationship to functions of the IBM PC, access to the Bank Address and Bank Data registers is inhibited by a protection feature. The protection feature requires that Pin 54 of the V6366 Display Controller (IOSEL-) be driven low and Control/ID register bit 7 (Protect-) be driven high before any Expansion Function register is selected or its contents modified. In the GRiDCase 1500 Series computers, IOSEL- is always low.

NOTE: The tilde symbol (-) following a signal name indicates active low (low-true) logic.

Bit 0 of the Control/ID register is used to control access to the Mode Control Register bit 1. Control/ID register bit 0 and Mode Control register bit 1 are both driven high (logic 1) to allow all graphics modes to be enabled.

If the Control/ID register is accessed when RREG = 0, the display controller ID code C1h is read.

Display Controller Secondary Registers

The display controller has internal Secondary registers that are accessed via the Primary Controller Address and Data registers (03D4h and 03D5h). The Primary Controller Address and Data registers were previously described. The following paragraphs describe the functions of the internal Secondary registers. The

internal Secondary registers are identified by their names and controller address in hexadecimal (h), which are listed in Table 7-3. All of the internal Secondary registers are compatible with the 6845 CRT controller used in the IBM PC.

NOTE: To reduce momentary flickering of the screen, avoid writing to the internal Secondary registers during display as much as possible. To eliminate the need to consider timing during register writes, set Enable Video (Mode Control Register, bit 3) to "0."

Horizontal Registers (00h, 01h, and 02h)

The first three Internal Secondary registers define horizontal timing for the display. Because it starts at zero, the count in the Horizontal Total register is one less than the total character positions in a complete horizontal line including displayed, blanking, and sync. The Horizontal Displayed Register gives the number of displayed character positions in the line and the Horizontal Sync Position register gives the starting character position of the horizontal sync pulse.

Figure 7-5 shows the relative timing of the Horizontal registers. The horizontal line contains *n* characters so that the Horizontal Total is programmed to *n*-1 characters. There are *j* displayed characters so that the Horizontal Displayed register is programmed to contain *j* character positions. The Horizontal Sync Position register contains *k* so that the horizontal sync pulse occurs *k* characters into the line. The Sync Width register (described in the following paragraph) controls the width of the sync pulse.

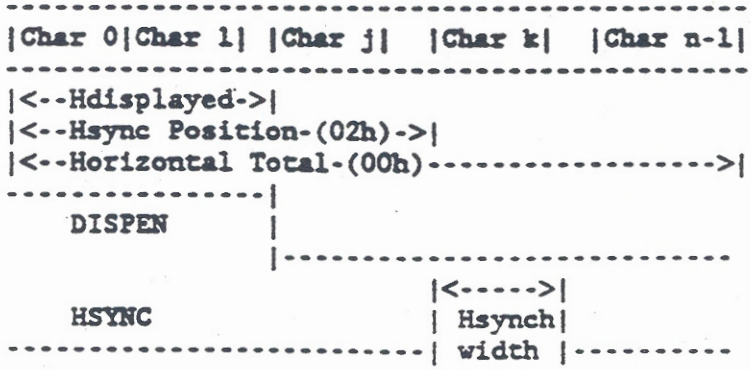


Figure 7-5. Horizontal Registers Relative Timing

Sync Pulse Width Register (03h)

This register is used to set the horizontal and vertical sync pulse widths. The most significant four bits (7-4) are used to set the vertical sync pulse width. The least significant four bits (3-0) are used to set the horizontal sync pulse width. The bits in the Sync Width register are defined as follows:

Bit 7	vertical 3
Bit 6	vertical 2
Bit 5	vertical 1
Bit 4	vertical 0
Bit 3	horizontal 3
Bit 2	horizontal 2
Bit 1	horizontal 1
Bit 0	horizontal 0

The horizontal sync width is set to the number of characters to be contained within the pulse. The number of characters is in the range of 1-15. A horizontal value of 0 is invalid and should not be used.

The vertical pulse width is set to the number of scan lines to be contained by the vertical sync pulse width. If the vertical value is 0 then 16 scan lines are used.

Vertical Registers (04h - 07h)

The Vertical Registers control the frame rate, vertical displayed area and position of the VSYNC pulse. The Vertical Total and Total Raster Adjust registers together determine the frame rate. The vertical total is the number of character rows -1 that the frame consists of. The total raster adjust is the number of extra scan lines in the frame. Thus the total number of scan lines in one frame is (vertical total) * (Max scan line + 1) + (total raster adjust). The Vertical Displayed register is the number of character rows that are displayed and the vertical sync position register contains the starting character row of the VSYNC pulse.

Figure 7-6 shows the relative timing of the Vertical registers. The frame contains *n* character rows so that the vertical total is programmed to *n*-1 rows. Then the Total Raster Adjust register (05h) is programmed for the desired number of extra scan lines to make up a complete frame. There are *j* displayed character rows so that the vertical displayed register is programmed to contain *j* character rows. The vertical sync position register contains *k* so that the vertical sync pulse occurs *k* character rows into the line. The Sync Width register (previously described) controls the width of the sync pulse.

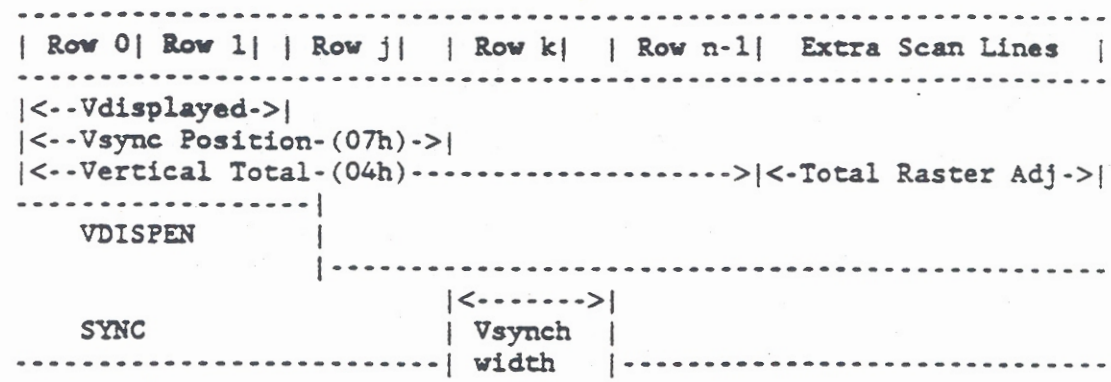


Figure 7-6. Vertical Registers Relative Timing

Interlace Mode (08h)

The Interlace Mode register is ignored because interlace is not used by the Display subsystem.

Maximum Raster Address Register (09h)

The lower 5 bits of the Maximum Raster Address register contain the maximum scan line address generated on the row outputs. The value is one less than the number of scan lines per character row. For example, for an 8 line character row, a 7 is used to obtain scan lines 0-7 for each row. The upper three bits are ignored.

Cursor Start/Mode and End Registers (0Ah and 0Bh)

The Cursor Start/Mode and the Cursor End registers determine the starting and ending line of the cursor. The Cursor Start/Mode register also contains the Mode bits (bits 6 and 5). The Mode bits select the cursor operating mode as nonblinking, blink rate, or nondisplayed. Bits 5 and 6 are defined in the following list. The mode decoding is different from the 6845 specification for the IBM PC. It allows the application program to set up the cursor blink logic as if there were a 6845 with external blink logic, which requires no blink out of the 6845. Since the blinking is accomplished internally, the 6845 "no-blink" condition corresponds to blinking at 1/16 of the frame rate, which is what the AT generates external to the 6845.

Bit 6	5	Mode
0	0	Blink at 16-field cycles (ON:OFF ratio = 1:1)
0	1	Cursor is not displayed
1	0	Not Used
1	1	Blink at 32-field cycles (ON:OFF ratio = 1:3)

The lower five bits of the Cursor Start/Mode register (0Ah) contain the starting scan line of the cursor. The lower five bits of the Cursor End register (0Bh) contain the ending scan line. To program an underlined cursor when the Maximum Raster Address (09h) is 7, a 6 is programmed for the starting cursor line and a 7 is programmed for the ending cursor line. To get a block cursor, a 0 is programmed for the starting cursor line and a 7 is programmed for the ending cursor line.

The general relationship between the Cursor Start Address, Cursor End Address, and the Maximum Raster Address is:

Cursor Start Address (CSA) is equal to or greater than 0.

Cursor End Address (CEA) is equal to or greater than CSA.

Maximum Raster Address (MRA) is equal to or greater than CEA.

If the relation ship is opposite to the standard, the cursor is displayed as split top and bottom halves. If only CEA is greater than MRA, the cursor is displayed at all character locations. The cursor is not displayed if only CSA is greater than MRA.

Start Address Registers (0Ch and 0Dh)

The two Start Address registers contain the display address of the character that is located in the upper left corner of the screen. The character display address is two times the buffer memory address. Start Address register 0Ch contains the high order portion of the character display address and the total address consists of 14 bits. Paging and scrolling are performed by dynamically rewriting the contents of these registers.

NOTE: The Start Address registers must be written during the Hdisplayed and Vdisplayed portion of the timing (see Figures 7-5 and 7-6).

Cursor Address Registers (0Eh and 0Fh)

The two Cursor Address registers contain the display address where the cursor is currently positioned. The cursor display address is two times the display buffer memory address. Cursor Address register 0Eh contains the high order portion of the cursor display address and the total address consists of 14 bits.

NOTE: The Cursor Address registers must be written during the retrace portion of the horizontal and vertical timing.

Light Pen Registers (10h and 11h)

The two Light Pen registers are not used in GRiDCase 1500 Series Computers.

Display Controller Expansion Function Registers

The display controller has internal Expansion Function registers that are accessed via the Primary Controller Bank Address and Data registers (03DDh and 03DEh, respectively). The Bank Address and Bank Data registers were previously described. The following paragraphs describe the specific operations of the Expansion Function registers. The Expansion Function registers are identified mainly by their internal addresses, which are given in hexadecimal (h), and are listed in Table 7-4.

The Expansion Function registers are arranged in four banks between internal addresses 00h and 3Fh. Each bank of registers is used for different display functions as follows:

1. Addresses 00h-1Fh provide a 32 register bank for 16 color palettes. Two registers make up each palette.

2. Addresses 20h-2Bh provide a 12 register bank that controls display operations. These registers provide discrete functions that are critical to proper operation of the Display Subsystem

3. Addresses 30h-37h provide an eight-register bank of preset values for display operation in alphanumeric modes. The values preset in these registers replace the values in equivalent Secondary registers (refer to Table 7-3) when the preset values are required.

4. Addresses 38h-3Fh are similar to addresses 30h-37h, except that they provide an eight-register bank of preset values for display operation in graphics modes. Addresses 37h and 3Fh are unique because they provide discrete functions required for preset operations

The four Expansion Function register banks are described in more detail in the following paragraphs.

Color Palettes (00h-1Fh)

The color palettes at addresses 00h through 1Fh receive the same IRGB data as the Color Select register (03D9h) that was previously described. Subsequent operation of the color palettes depends upon the setting of two discrete bits in Expansion Function registers at addresses 27h and 28h. When CPLE = 0 (27h, D1), the Color Palettes registers do not operate and the IRGB data is output directly from

the Color Select register. If CPLE = 1, the Color Palettes registers operate as described in the following paragraphs. Also, set SWC = 1 (28h, D7) in order to use a linear RGB monitor. Otherwise, SWC = 0.

Each color palette is arranged in nine bits over two address locations. The data must be written to these addresses with the even address followed by the odd address, and becomes valid when the odd address is written. The nine bits over two addresses provides a total of 16 color palettes. Three bits within each nine bits of data are then assigned to Red, Green, and Blue to allow any 16 out of 512 colors to be displayed simultaneously on a linear RGB monitor. On an EGA monitor, six bits of each palette are used to provide a simultaneous display of 16 out of 64 colors. IBM Color monitors can use four bits as a color lookup table, and monochrome monitors can use five bits to set gray scaling and hatching. The functions for each of the 16 color palettes are summarized in Table 7-5.

Table 7-5. Color Palette Bit Functions

Color Palette Bit Number	Linear RGB Monitor	EGA Monitor	IBM Color Monitor	Monochrome Monitor
R2	Red 22	-	-	Intensity
R1	Red 21	-	-	Intensity23
R0	Red 20	Secondary B	-	Intensity22
G2	Green 22	Secondary R	-	Intensity21
G1	Green 21	-	-	Intensity20
G0	Green 20	Secondary G	Intensity	-
B2	Blue 22	Primary R	Red	-
B1	Blue 21	Primary G	Green	-
B0	Blue 20	Primary B	Blue	-

When a monochrome display is used with Color Graphics Adapter (CGA) programs, the colors are displayed by gray scaling and hatching effects. Gray scaling is effective for LCD displays because of their slow response rate. However, converting the Gray scaling to a hatching pattern is recommended for plasma displays, electro-luminescent (EL) displays, and monochrome CRTs because of their faster response times. The gray scaling and hatching can be achieved by setting the R1, R2, G1, and G2 bits of the color palette (refer to Table 7-5). Figure 7-7 shows how the bits are set to achieve various levels and combinations of gray scaling and hatching.

Also for monochrome displays, it is recommended that MON = 1 (24h, D7) and a clear display be used (set Black as Display Off and White as 1/1). For monochrome displays that are capable of intensity input, set IEN = 1 (27h, D5) to achieve good gray scaling and hatching.

R^1	R^0	G^2	G^1	Description
0	0	0	0	Display Off
0	0	0	1	1/4 Gray Scale
0	0	1	0	2/4 Gray Scale
0	0	1	1	3/4 Gray Scale
0	1	0	0	1/1 Gray Scale
0	1	0	1	Combination GS&H(1/2)
0	1	1	0	Combination GS&H(1/2)
0	1	1	1	Combination GS&H(1/2)
1	0	0	0	1/3 Gray Scale
1	0	0	1	2/3 Gray Scale
1	0	1	0	1/3 Hatching
1	0	1	1	2/3 Hatching
1	1	0	0	3/1 Hatching
1	1	0	1	3/2 Hatching
1	1	1	0	2/3 Hatching
1	1	1	1	1/2 Hatching

NOTE: GS&H(1/2) = Gray Scale and Hatching (1/2).

Figure 7-7. Color Palette Conversion to Gray Scaling and Hatching

Control Function Registers (20h-2Bh, 37h, and 3Fh)

Expansion Registers 20h-2Bh are used to provide discrete control functions that are critical to proper operation of the Display Controller. These Control Function registers are written and read via the Bank Address and Bank Data Registers (03DDh and 03DEh, respectively). The Control Function registers, identified by their Controller Addresses, are eight bits wide. However, the register at each address may be further divided into smaller registers of one or more bits each. Each of the individual registers is identified by its mnemonic in Table 7-4. Table 7-6 lists the register mnemonics alphabetically and provides each register address, bit location(s), and name for quick reference. For example, ACM is located at register 27h, bit D7. A more detailed description of each control Function register is provided in the following paragraphs.

Expansion Registers 37h and 3Fh are also divided into smaller discrete registers and their mnemonics are listed in Table 7-6 for convenience. More detailed descriptions of registers 37h and 3Fh are contained in subsequent paragraphs describing the Preset Register Banks.

Table 7-6. Expansion Register Control Bits

Mnemonic	Register Addr	Bit	Name	Usage
ACM	27	D2	Consecutive Address Mode	Graphics mode
A15	26	D6	Display RAM Address 15	RAM or C/ID (3DFh)
BINH	25	D2	Inhibit Border Display	Alpha or Low Res
CAO 4-0	37	D4-D0	Cursor Address Offset	Cursor Control
CLK 1, 0	25	D1, D0	Clock Input Pins	Selects clock
CON 7-0	2B	D7-D0	Control Data Bits 7-0	External control
CPLE	27	D1	Color Palette Enable	Exp Regs (00h-1Fh)
DUTY	20	D7-D0	Duty	Dual Screen Display
EH 4-0	29	D4-D0	Enable Clock Cycle	Sets Display clock
ESIO	29	D5	External Sync I/O	Selects Ext Sync
EXP	26	D5	Address Expansion	Selects Exp Addr
EXTS	27	D4	External Sync Mode	Selects Sync Mode
FON 2-0	24	D2-D0	Font Select	Alpha Mode
F/O	37	D7	Fixed Cursor	Selects cursor
GDS 1, 0	3F	D6, D5	Graphics Scan Control	Graphics modes
GRM 2-0	24	D5-D3	Graphics Mode	Graphics modes
HADJ	23	D5-D0	Horizontal Adjust	Horizontal Position
HERC	24	D6	Hercules/CGA	Controls switching
H/E	28	D3	Horiz Sync Select	Selects Enable Clk
HSIZE	21	D7-D0	Horizontal Size	Pixels per Line
IEN	27	D5	Intensity Enable	Monochrome Display
IINH	27	D3	Inhibit Display OFF	Prevents Display
LS 1, 0	27	D7, D6	Monochrome Data Bits	Mono Data Transmit

Table 7-6. Expansion Register Control Bits (Continued)

Mnemonic	Register Addr	Bit	Name	Usage
MON	24	D7	Monochrome Mode	Color/Mono Convert
PAG 2-0	26	D2-D0	Page Address	Controls Addressing
PB 1, 0	26	D4, D3	Page Mode Bits	Address Byte Select
PRE	27	D0	Preset Enable	Enables Preset Regs
RA4	26	D7	Raster Address I/O	Raster I/O Status
RREG	29	D6	Read I/O Registers	Mode Reg (03D8h)
R/M	28	D2	Raster Address Type	Selects RA4 Function
SCR 1, 0	28	D1, D0	Screen Type	One or Two Screens
SCT	23	D7	Memory Timing	Display Memory
SCU	23	D6	Cursor Blink	Selects Blink Cycle
SRAM	25	D5	Select RAM	Display RAM Type
SSCK	28	D4	Shift Clock Output	Selects Clock Timing
SSY	28	D5	Sync Signal Polarity	Horiz/Vert Sync
SS 4-0	2A	D4-D0	Smooth Scrolling	Offset
STBY	29	D7	Standby Mode	Enables Standby
SWC	28	D7	Horiz Sync Width	Horizontal Display
SYTH	25	D7	Sync Pulse Width	Horiz/Vert Sync
TDS 1, 0	37	D6, D5	Alpha Scan Control	Raster Address
TST 2-0	2A	D7-D5	Controller Test	Must be 000 for use
ULE	24	D6	Underline	Monochrome Display
UP 4-0	3F	D4-D0	Underline Position	Dual Screen Graphics
VADJ	22	D7-D0	Vertical Adjust	Vertical Position
VST	28	D6	Vertical Sync	Vertical Sync Timing
16CP	25	D5	CPU Bus Width	Selects CPU Bus
45S	3F	D7	6845 Expansion	Vertical Regs Expand
8BUS	25	D4	Data Bus Width	Selects Data Bus

Address 20h

Address 20h provides an 8-bit DUTY register for dual screen displays. The DUTY register is set to the reciprocal of the duty value that equals the number of lines on the screen. A similar value is set for either of two types of dual screen displays as follows:

- Dual-drive type two-screen LCD that transmits data to the top and bottom screens simultaneously.
- Single-drive type two-screen LCD that alternately transmits one line each from the top and bottom screens.

This register is not used for one-screen panels such as Electro-Luminescent (EL), plasma, etc.

The duty can be any value in the range of 1/2 to 1/256. The value used should be greater than the total raster quantity, determined by the Vertical Total and Maximum Raster Address. When the selected duty

value is "1/D" the value written to the DUTY register is "D-1." For example; a duty of 1/100 requires the value 63h in the DUTY register, and a duty of 1/200 requires a value of C7h in the DUTY register.

Address 21h

Address 21h provides an 8-bit Horizontal Size (HSIZE) register. The register is used to set the number of pixels in a horizontal line of a flat panel display. The number of pixels per line should be less than the Horizontal Total value. For a dual-drive type two-screen LCD, the quantity of pixels per line is set in the range of 4 to 1024 in multiples of 4 (4, 8, 12, ..., 1016, 1020, 1024). With any other type of flat panel display, the quantity of pixels per line is set in the range of 8 to 2048 in multiples of 8 (8, 16, 24, ..., 2036, 2040, 2048). The value written to the HSIZE register, where the quantity of pixels per line is "n," is determined as follows:

- For dual-drive type two-screen LCD,
  HSIZE = n/4 (-1).
  For 640 pixels, the value is 9Fh.

- For all other flat panels:
  HSIZE = n/8 (-1).
  For 640 pixels, the value is 4Fh.

Address 22h

Address 22h provides an 8-bit Vertical Adjust (VADJ) register. The register is used to correct the display position in a vertical direction. The contents of the register specify (in line units) the number of lines between the rising edge of the Vertical Sync (VSY) signal (Controller pin 25) and the beginning line of the display. When this register is properly set, the display position is correct without losing any characters at the top and bottom edges of the display. The correct value is in the range of 1-256 lines, but normal operation is achieved only when the setting satisfies the following equation.

$$n + 1 + (2xD) <= Vt \times MRA$$

Where: n = Correct value for VADJ Register (22h)
D = DUTY Register (20h) value (-1)
Vt = Vertical Total Reg, R4, value (-1)
MRA = Max Raster Address Reg, R9, value (-1)
Register R4 and R9 are listed in Table 7-3.

Address 23h

Address 23h provides a 6-bit Horizontal Adjust (HADJ) register and two single bit discrete registers (SCT and SCU). The 6-bit HADJ register is used to correct the display position in a horizontal direction. The contents of the register specify the number of pixels between the rising edge of the Horizontal Sync (HSY) signal (Controller pin 24) and the beginning of the display. When this register is properly set,

the display position can be correctly adjusted between the right and left edges of the display. The correct value is in the range of 8 to 512 pixels, but normal operation is achieved only when the setting satisfies the following equation.

$$2M - {}^{n}/_{8} - i \geq H + HA$$

Where: HA = Correct value for HADJ Register (23h)
n = Pixels per line at Address 21h
M = Horiz Total Reg, R0, value (-1)
H = Horiz Sync Position Reg, R2, value (-1)
i = integer 12 for dual-drive two screen LCD
or integer 9 for all other flat panels
Registers R0 and R2 are listed in Table 7-3.

Memory Timing Bit, SCT (23h, D7)

SCT is a one-bit register that is cleared to "0" on a system reset, and then set to "1" so that high-speed timing is enabled. When the Display Controller is used with a high-speed CPU, the high-speed timing is selected to minimize the average Wait time. To enable the high-speed timing, SCT is set to "1."

Cursor Blink Bit, SCU (23h, D6)

SCU is a one-bit register that is cleared to "0" on a system reset. The cursor blink cycle is normally fixed to 16 fields. If the cursor is hard to see, such as on an LCD display with a slow response rate, the cursor can be made to blink in 32-field cycles by setting SCU to "1." When SCU is set to "1," it overrides the blink function set in the Cursor Start register (R10, Table 7-3).

Address 24h

Address 24h provides four discrete registers. Two registers are one-bit each (MON and ULE) and two registers are three-bits each (GRM2-GRM0 and FON2-FON0). The system reset state of all Address 24h registers depends upon the state of the RA4 register (Address 26h, D7). When the RA4 register is "0" (CGA Mode), the Address 24h registers are reset to 02h. If the RA4 register is "1" (Hercules Mode), the Address 24h registers are reset to E3h. Each Address 24h register is described in the following paragraphs.

Monochrome Mode Bit, MON (24h, D7)

MON is set to "1" to enable the monochrome display mode. When the monochrome display mode is enabled, the RGB values for CGA software are automatically converted for two color displays on monochrome monitors and flat panels. The intensity (I) bits for foreground and background function independently of the RGB conversion.

Underline Bit, ULE (24h, D6)

ULE is set to "1" to emphasize text on a monochrome display.

Graphics Mode Control, GRM2-GRM0 (24h, D5-D3)

GRM2 through GRM0 are used to select graphics functions in addition to the functions that are standard to the CGA and Hercules modes. GRM2 is used to select the standard CGA graphics mode ("0") or the Hercules mode ("1"). GRM1 and GRM0 are used to select the quantity of colors that can be simultaneously displayed. GRM1 and GRM0 are set to select the number of pixels used to specify colors and therefore the number of colors available for display as follows:

NOTE: In the following list, screen resolution indicates the quantity of horizontal pixels (640) displayed multiplied by the quantity of bits (1, 2, or 4) used per pixel.

GRM Bits 2	1	0	Screen Resolution	Number of Colors
0	0	0	640 x 1	2
0	0	1	640 x 2	4
0	1	0	640 x 4	16

In CGA mode (GRM2=0), the scan lines are divided into even and odd groups, with each group entered into a separate bank of memory addresses (two-bank mode). The even scan lines are located between addresses 0000 and 8191, and the odd scan lines are located between addresses 8192 and 16,384. . If Hercules mode is selected (GRM2=1), data is separated into four banks (four-bank mode)as follows (see Figure 6-4):

- Bank 1 contains lines 0, 4, 8, etc.
- Bank 2 contains lines 1, 5, 9, etc.
- Bank 3 contains lines 2, 6, 10, etc
- Bank 4 contains lines 3, 7, 11, etc.

Font Select, FON2-FON0 (24h, D2-D0).

FON2-FON0 are used in alphanumeric modes for selecting the number of horizontal pixels in the character font. In the 40x25 and 80x25 Alphanumeric modes, the number of horizontal pixels can be set to 6 through 10 or any multiple of 8 (8, 16, 24, etc.). Also, when the 40x25 Alphanumeric mode is selected, the number of horizontal pixels can be set to a multiple of 16 (16, 32, 48, etc.). The number of horizontal pixels in a character font are selected as follows:

FON Bits 2	1	0	Number of Horiz Pixels	Comment
0	0	0	6	GRM0 = 0
0	0	1	7	"
0	1	0	8	"
0	1	1	9	"
1	0	0	10	GRM0 = 0
1	0	1	8xn	n=any integer
1	1	0	16xn	n=any integer

Address 25h

Address 25h provides seven discrete control registers. Six registers are one-bit each (SYTH, HERC, SRAM, 8BUS, 16CP, and BINH) and one register has two bits (CLK1 and CLK0). The system reset state of all Address 25h registers depends upon the state of the RA4 register (Address 26h, D7). When the RA4 register is "0" (CGA Mode), the Address 25h registers are reset to 30h. If the RA4 register is "1" (Hercules Mode), the Address 25h registers are reset to F4h. Each Address 25h register is described in the following paragraphs.

Sync Pulse Width, SYTH (25h, D7)

SYTH selects how to use the output of the Sync Pulse Width register (R3, Table 7-3). With EGA and monochrome monitors, SYTH is set to "1" for normal use of the R3 output and negative polarity. To use an IBM color monitor, or change the polarity from negative to positive, SYTH is set to "0."

Hercules/CGA Switching, HERC (25h, D6)

HERC allows the Display Controller to switch between CGA software support (two-bank mode) and Hercules software support (four-bank mode). Since CGA is the default mode, this bit is reset to "0." For operation in 640x400 mode, the Hercules support is used by setting this bit to "1."

Select RAM, SRAM (25h, D5)

SRAM selects the type of devices used for Display Memory. For Static RAM devices, SRAM is set to "1," and for Dynamic RAM devices, SRAM is set to "0." At initialization and reset, the value of SRAM is determined by the LTD- signal (Controller pin 36). Since GRiDCase 1500 Series computers use Static RAM devices, this bit should be set to "1."

Data Bus Width, 8BUS (25h, D4)

The 8BUS bit selects the Display Memory data bus width. For an 8-bit data bus, 8BUS is set to "1." If a 16-bit data bus is used, 8BUS is set to "0" and the data bits being accessed during a read or write operation are determined by Address bit A0. With A0 = "0," data bits

D7-D0 are accessed and with A0 = 1, data bits D15-D8 are accessed. The GRiDCase 1500 Series computers use an 8-bit bus.

CPU Bus Width, 16CP (25h, D3)

The 16CP bit selects the Display Controller CPU bus width. When an 8-bit CPU bus is used, 16CP is set to "0." Setting 16CP to "1" allows a 16-bit CPU bus to be connected. When a 16-bit CPU bus is used, the high-order bits D15-D8 (odd address bytes) are connected through an external circuit. The GRiDCase 1500 Series computers use an 8-bit bus.

Inhibit Border Display, BINH (25h, D2)

BINH is used with monochrome and EGA monitors and flat panel displays to provide a clear (black) border.

Clock Input Pins, CLK1 and CLK0 (25h, D1 and D0)

CLK1 and CLK0 select the Display Subsystem clock signal input to the Display Controller. The clock input is selected from the Display Controller inputs X1 and X2 (pins 6 and 7, respectively) as follows:

CLK Bits 1	0	Clock Input
0	0	14.318 MHz CGA Clock (Default)
0	1	17.745 MHz 640x400 Mode Clock
1	x	Not Used

Address 26h

Address 26h provides five discrete control registers. Three registers are one-bit each (RA4, A15, and EXP), one register has two bits (PB1 and PB0), and one register has three bits (PAG2-PAG0). All of the Address 26h registers are cleared by a system reset. Each Address 26h register is described in the following paragraphs.

Raster Address I/O, RA4 (26h, D7)

RA4 is used to select either the input or output status of Raster Address 4 (RA4) pin (Controller pin 68). Register RA4 is normally set to "0" (default) and the RA4 pin is an input pin. For some flat panels, the RA4 register is set to "1," which changes the RA4 pin to an output pin. The output signal is called AC Conversion Signal M.

Display RAM Address 15, A15 (26h, D6)

A15 is not used by the GRiDCase 1500 Series computers.

Address Expansion, EXP (26h, D5)

EXP selects between the IBM compatible mode and the Address Expansion mode. For IBM Compatible mode, EXP is set to "0." If the Address Expansion mode is used, EXP is set to "1." In the IBM compatible mode, there are 11 or 13 bits used for addressing the Monochrome Display Adapter (MDA) or Color Graphics Adapter (CGA), respectively. However, the Display Controller provides a 14-bit address capability. These additional bits become available when EXP is set to "1."

Page Mode Bits, PB1 and PB0 (26h, D4 and D3)
PB1 and PB0 allow multiple pages to be used for large display memories. The GRiDCase 1500 Series computer does not use these bits.

Page Address, PAG2-PAG1 (26h, D2-D0)

PAG2 through PAG0 select specific pages of display memory for display on the screen. The GRiDCase 1500 Series computer does not use these bits.

Address 27h

Address 27h provides seven discrete control registers. One register has two bits (LS1 and LS0) and six registers are one-bit each (IEN, EXTS, IINH, ACM, CPLE, and PRE). All of the Address 27h registers are cleared by a system reset. Each Address 25h register is described in the following paragraphs.

Monochrome Data Bits, LS1 and LS0 (27h, D7 and D6)

LS1 and LS2 select the quantity of bits and their method of transmission to a monochrome monitor or flat panel as follows:

LS Bits 1	0	Transmission
0	0	1 Bit Serial (IBM Compatible)
0	1	2 Bit Even/Odd
1	0	4 Bit Parallel
1	1	8 Bit Parallel (No Double Sequence)

Intensity Enable, IEN (27h, D5)

IEN is used to select gray scaling and hatching patterns for monochrome and flat panel displays. The monochrome monitors and flat panels are normally capable of displaying four shades of gray (half-tones). With IEN set to "1," four additional shades are available to choose from.

External Sync Mode, EXTS (27h, D4)

EXTS enables external synchronization when it is set to "1." The GRiDCase 1500 Series computer does not use this bit. Refer to ESIO (29h, D5).

Inhibit Display OFF, IINH (27h, D3)

IINH controls the Enable Video bit (bit 3) of the Mode Control register (03D8h). If IINH is set to "1" during high-resolution alphanumeric mode (80x25) operation, the Enable Video bit will remain ON. The result is that the display will remain ON during screen refresh cycles and provide a more visible screen.

Consecutive Address Mode, ACM (27h, D2)

ACM enables the selection consecutive addresses during Graphics mode operation. The GRiDCase 1500 Series computer does not use this bit.

Color Palette Enable, CPLE (27h, D1)

CPLE is used to enable the color palette registers (Display Controller Expansion registers 00h through 1Fh). Refer to Table 7-4. When CPLE is set to "0" for a CGA monitor, the Color Palette registers are disabled and color data is provided directly from the Color (IRGB) Select register (03D9h). CPLE is set to "1" when the Color Palette registers are used as a Color Lookup table for to linear RGB, EGA, or CGA monitors. Also, CPLE is set to "1" when the Color Palette registers are used as a Conversion Table for Gray Scaling and Hatching in Monochrome modes.

Preset Enable, PRE (27h, D0)

PRE is used to enable the Preset Data registers (Display Controller Expansion registers 30h through 3Fh). Refer to Table 7-4. When PRE is set to "0," the Preset Data registers are not used and Internal Secondary registers R0, R2-R5, R7 and R9 are used directly (refer to Table 7-3). If PRE is set to "1," the Preset Data registers are used in place of their respective Internal Secondary register.

Address 28h

Address 28h provides seven discrete control registers. Six registers are one-bit each (SWC, VST, SSY, SSCK, H/E, and R/M) and one register has two bits (SCR1 and SCR0). All of the Address 28h registers are cleared by a system reset. Each Address 28h register is described in the following paragraphs.

Horizontal Sync Width, SWC (28h, D7)

SWC controls the width of the Horizontal Sync Pulse (HSY, Controller pin 24). When SWC is set to "1," the HSY pulse width is narrowed for use by panel displays (see Figure 7-8). Also, when an RGB monitor is

used, both the SWC and CPLE (Address 27h, D1) are set to "1" to allow all nine bits of the Color Palette to be output.

Vertical Sync, VST (28h, D6)

VST selects output timing for the Vertical Sync Pulse (VSY, Controller pin 25). When VST is set to "0," VSY is synchronized with the falling edge of HSY (see Figure 7-8). If VST is set to "1," VSY is synchronized approximately 1 microsecond after the rising edge of HSY. In general, VST = 0 is used for EL displays and VST = 1 is used for LCD and Plasma displays.

Sync Signal Polarity, SSY (28h, D5)

SSY selects the polarity of the Horizontal and Vertical Sync pulses (HSY and VSY at Controller pins 24 and 25, respectively) for flat panel displays. For EL and plasma displays, SSY is set to "0" to provide a positive polarity pulse. SSY is set to "1" to provide a negative polarity pulse for LCD displays (see Figure 7-8).

Shift Clock Output, SSCK (28h, D4)

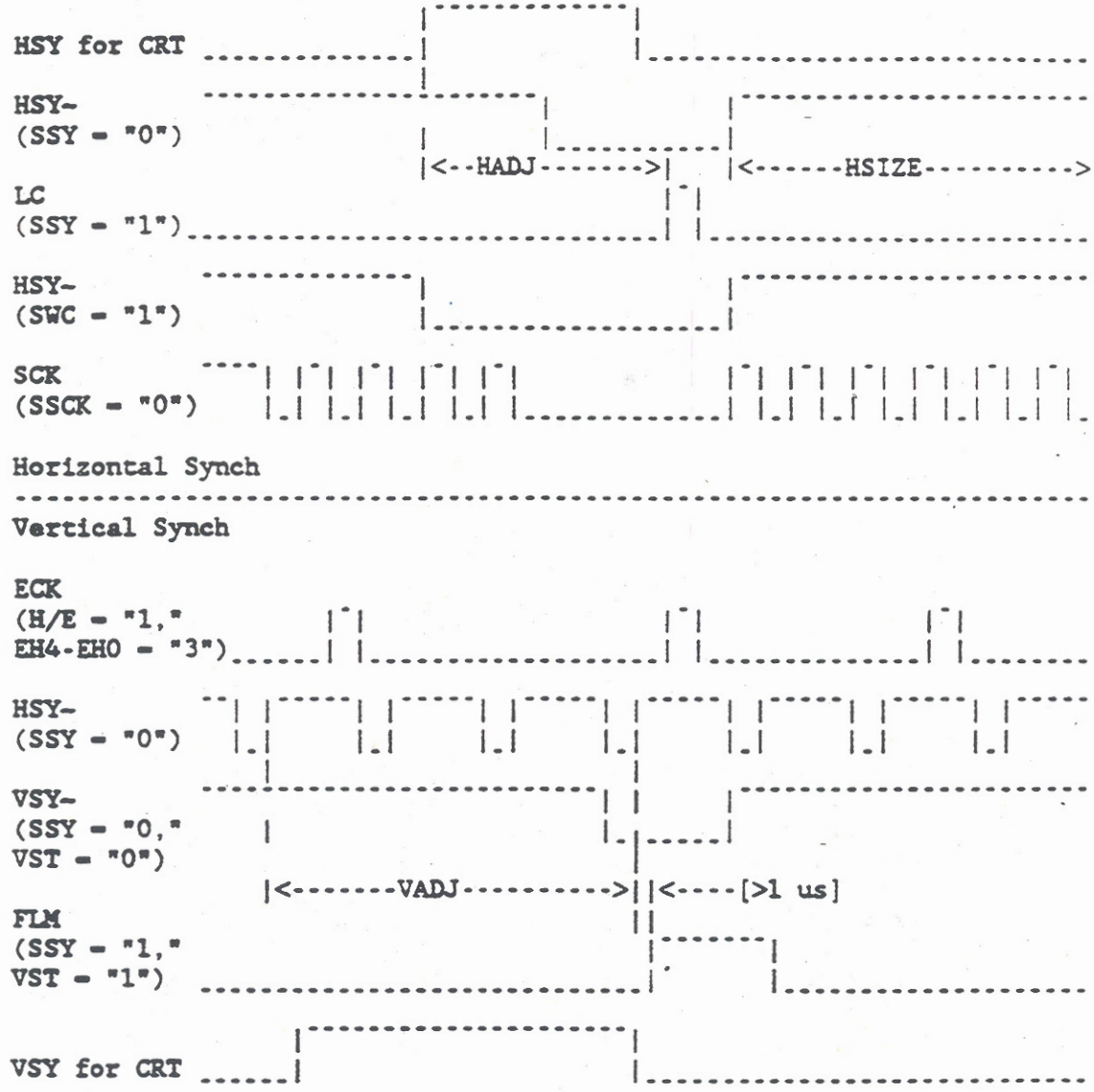
SSCK selects the Shift Clock Output. For EL displays, SSCK is set to "1" to provide a constant output clock. Generally, all other flat panel displays require an output clock only during horizontal display time. Then, SSCK is set to "0" (see Figure 7-8).

Horizontal Sync Select, H/E (28h, D3)

H/E is used to select the signal that is output on the Horizontal Sync pin (HSY, Controller pin 24). When H/E is set to "0," the horizontal sync signal for a CRT is output. With H/E set to "1," the Enable Clock (ECK) signal for an LCD display is output (see Figure 7-8).

Raster Address Type, R/M (28h, D2)

R/M is used to select the signal that is output on the Raster Address 4 pin (RA4, Controller pin 68). When R/M is set to "0," Raster Address RA4 is output. With R/M set to "1," a signal called AC Conversion Signal M is output for LCD displays. In either case, before pin RA4 will function as an output pin, register RA4 (Address 26, D7) must be set to "1."



NOTES:

1. The following mnemonics represent Expansion Register Control Bits: EH4-EH0, HADJ, H/E, HSIZE, SSCK, SSY, SWC, VADJ, VST. For register locations of these bits, refer to Table 7-6.

2. The Horizontal (HSY) and Vertical (VSY) Synchronization are provided at Controller I/O pins 24 and 25, respectively. The inverted version of the same signals are indicated by a tilde after the mnemonic (HSY-, VSY-).

3. The following signals, internal to the Controller, are shown for reference only: ECK, FLM, HSY-, LC, SCK, and VSY-.

Figure 7-8. Horizontal and Vertical Synchronization

Screen Type, SC1 and SC0 (28h, D1 and D0)

SC1 and SC0 provide a two-bit buffer for selecting the type of display screen as follows:

SC Bits 1	0	Screen Type
0	0	No Gray Scale or Hatching
0	1	One-Screen Flat Panel (EL, Plasma, etc.)
1	0	Dual-drive type Two-screen LCD
1	1	Single-drive type Two-screen LCD

Address 29h

Address 29h provides four discrete control registers. Three registers are one-bit each (STBY, RREG, and ESIO) and one register has four bits (EH3 - EH0). All of the Address 29h registers are cleared by a system reset. Each Address 29h register is described in the following paragraphs.

Standby Mode, STBY (29h, D7)

STBY is used to conserve power. The Standby mode is enabled by setting STBY to "1." In Standby mode, the internal clock stops and most Display Controller circuits cease operation. To cancel the Standby mode, the Enable Video bit (bit 3 of the Mode Control register, 03D8h), is first set to "0." Setting the Enable Video bit to "0" sets the Display Status to OFF. With the Display status set to OFF, the STBY bit is set to "0." Then, after an interval of two vertical scans to allow the internal circuits to stabilize, set the Enable Video bit to "0." Read and write operations with the I/O registers can be performed while the Display Controller is in Standby mode.

Read I/O Registers, RREG (29h, D6)

RREG controls the operation of three Primary registers. The Mode Control register (03D8h), the Color Select register (03D9h), and the Control/ID register (03DFh). The Mode Control and Color Select registers cannot be read until RREG is set to "1."

The type of data read from the Control/ID register (03DFh) is dependent upon the state of RREG. When RREG = 1, the control functions are read from the Control/ID register. If the Control/ID register is accessed when RREG = 0, the Display Controller ID code C1h is read.

External Sync I/O, ESIO (29h, D5)

ESIO works with EXTS (27h, D4) to set the input/output status for the External Synchronization (ESY-) pin (Controller pin 26). The GRiDCase 1500 Series computer does not use either of these bits.

Enable Clock Cycle, EH4-EH0 (29h, D4-D0)

EH4 - EH0 set the cycle for the Enable Clock signal (ECK) for LCD display drivers and also change the phase of the Color Burst Clock signal for use with a CRT. The following paragraphs describe the operations of EH4 - EH0.

Internal to the Display Controller, when W/E is set to "1," an ECK signal is output during the first half cycle of the LC signal (see Figure 7-8). Subsequently, another ECK signal is output each time the Shift Clock (SCK) count is one greater than the value contained in EH4-EH0. For example, Figure 7-8 shows EH4-EH0 set to "3," and a subsequent ECK pulse is output on every forth SCK.

For operation with a CRT, the internal clock signals can be halted in order to change the phase of the Color Burst Clock signal for composite use as required by each line. Halting the internal clock signals will lengthen the horizontal scan cycle time by an equivalent amount.

When the Master Clock frequency is 14.318 MHz, the Enable Clock Cycle lines EH1 and EH0 are set to control the phase of the Color Burst Clock signal as follows:

EH1	EH0	Phase Change
0	0	No Change
0	1	90 Degree Change
1	0	180 Degree Change
1	1	270 Degree Change

Address 2Ah

Address 2Ah provides two discrete control registers. One register has three bits (TST2 -TST0) and the other register has five bits (SS4 - SS0). Both of the Address 2Ah registers are cleared by a system reset. Each Address 2Ah register is described in the following paragraphs.

Controller Test, TST2-TST0 (2Ah, D7-D5)

TST2-TST0 are used for testing the Display Controller and cannot be used for normal operation. If any of the bits are set to "1," normal operations cannot be performed.

Smooth Scrolling, SS4-SS0 (2Ah, D4-D0)

SS4 - SS0 can be set to a maximum raster offset value in the same range as the Maximum Raster Address Register R9 (refer to Table 7-3). If the offset value exceeds the address range, abnormal operation will occur. The value set in SS4 - SS0 causes the display to shift upward by some quantity of lines. Any lines that overflow the top of the

screen are not displayed. To perform smooth scrolling, the offset value is increased sequentially from zero, to the quantity of rasters forming the font. When the offset value reaches the Maximum Raster Address, line scrolling is performed by the Start Address registers (refer to Table 7-3) and the offset value is reset to zero.

Address 2Bh

Address 2Bh provides an 8-bit Control Data register. The contents of the register are output on the display data lines (D0-D7) near the rising edge of the Vertical Synchronization (VSY) signal (Controller pin 25). When the register is latched on the rising edge of VSY, its contents can be used for external control switching such as monitor, Font ROM, etc. A scrambled display may occur if this function is used when the rising edge of VSY occurs during the vertical display time. Register 2Bh is cleared by a system reset.

Preset Data Registers (30h-3Fh)

Expansion Registers 30h-3Fh are used to provide preset values for display operations when nonstandard combinations of software and monitors are used with the Display Controller. Nonstandard combinations of software and monitors include CGA (color) software with a black and white monitor or CGA software with enhanced resolution (640x400 pixel) monitors. Both of these nonstandard combinations occur with the flat panel displays that are internal to the GRiDCase 1500 Series computers. With the nonstandard combinations it is necessary to convert horizontal and vertical values to new values that match the needs of the flat-panel display that is being used. The Preset Data registers are loaded and selected to provide the required conversion values.

Also, different horizontal and vertical values are needed when the display switches between alphanumeric and graphics modes. To provide for these differences, the Preset Data registers are divided into two banks. The bank that contains registers 30h through 37h are used for preset alphanumeric display values, and the bank that contains registers 38h through 3Fh are used for preset graphics display values (refer to Table 7-4).

When each bank is loaded with its preset values, and PRE (27h, D0) is set to "1," the preset values replace the values contained in the equivalent Secondary Registers. The equivalent Secondary Registers are indicated by the register number (Rn) following the name of the Preset Data register and the Secondary Registers are also listed separately in Table 7-3.

The Alphanumeric Preset registers are always loaded with values for the high resolution mode (80x25). The values loaded are automatically halved for use in low resolution (40x25) mode.

When standard combinations of software and monitors are being used, the preset registers are not required and PRE (27h, D0) is set to "0." Standard combinations of software and monitors include CGA software and the IBM color monitor or Hercules type software and an IBM monochrome monitor.

The Preset Data registers are identified by their Display Controller address and the operation of each register is described in the following paragraphs. Two discrete control registers (37h and 3Fh) are included with the Preset Data registers. The mnemonics for the discrete control registers are listed alphabetically for reference in Table 7-6 and the register operations are also described in the following paragraphs.

Addresses 30h and 38h

Addresses 30h and 38h provide Preset Data registers for the Horizontal Total value to replace the value in Secondary Register R0. The parameters for setting 30h and 38h are identical to those used for setting R0.

Addresses 31h and 39h

Addresses 31h and 39h provide Preset Data Registers for Horizontal Sync Position Offset. In Preset Mode, the value in 31h or 39h is added to the value in Secondary Register R2. The offset value must be expressed in two's complement and can be set in the range of -128 to +127. If no offset value is required, the value of 31h and 39h is set to 00h. Also, if the offset value causes an overflow, the Horizontal Sync Position is considered to be 00h.

Addresses 32h and 3Ah

Addresses 32h and 3Ah provide Preset Data Registers for Vertical Sync Position Offset. In Preset Mode, the value in 32h or 3Ah is added to the value in Secondary Register R7. The offset value must be expressed in two's complement and can be set in the range of -128 to +127. When a negative offset value is used, bit 45S (3Fh, D7) must be set to "1." Otherwise, a "1" is not written into the offset value MSB. If no offset value is required, the value of 32h and 3Ah is set to 00h. Also, if the offset value causes an overflow, the Vertical Sync Position is considered to be 00h.

Addresses 33h and 3Bh

Addresses 33h and 3Bh provide Preset Data registers for the Sync Pulse Width value to replace the value in Secondary Register R3. The parameters for setting 33h and 3Bh are identical to those used for setting R3. In Preset Mode, the operation of 33h and 3Bh is equivalent to bit 45S (3Fh, D7) being set to "1" regardless of how 45S is actually set.

Addresses 34h and 3Ch

Addresses 34h and 3Ch provide Preset Data registers for the Vertical Total value to replace the value in Secondary Register R4. The parameters for setting 34h and 3Ch are identical to those used for setting R4.

Addresses 35h and 3Dh

Addresses 35h and 3Dh provide Preset Data registers for the Total Raster Adjust value to replace the value in Secondary Register R5. The parameters for setting 35h and 3Dh are identical to those used for setting R5. The Total Raster Adjust value loaded into 35h or 3Dh requires only five bits, therefore, bits D7-D5 are not used.

Addresses 36h and 3Eh

Addresses 36h and 3Eh provide Preset Data registers for the Maximum Raster Address value to replace the value in Secondary Register R9. The parameters for setting 36h and 3Eh are identical to those used for setting R9. The Maximum Raster Address value loaded into 36h or 3Eh requires only five bits, therefore, bits D7-D5 are not used.

These registers enable the use of different fonts that may be more suitable to a particular monitor. For example, when the CGA software provides a font of 8x8 pixels, a 9x14 pixel font can be displayed on a monochrome monitor if the register (36h) contents are set to 0Dh.

Address 37h

Address 37h provides three discrete control registers. One register has one bit (F/O), one register has two bits (TDS1 and TDS0), and the third register has five bits (CA04 - CA00). All of the Address 37h registers are cleared by a system reset. Each Address 37h register is described in the following paragraphs.

Fixed Cursor, F/O (37h, D7)

F/O controls the position of the cursor with respect to the displayed characters. If F/O is set to "0," the cursor position depends upon the software setting for the Cursor Start and End registers (Secondary Registers R10 and R11). When F/O is set to "1," the cursor position is fixed in relation to the Maximum Raster Address (Address 36h). The Maximum Raster Address is offset for the cursor position by the value contained in Cursor Address Offset bits (37h, D4-D0). Fixing the cursor position. in some cases, prevents the cursor from being displayed over or in the center of the characters.

Alpha Scan Control, TDS1 and TDS0 (37A, D6 and D5)

TDS1 and TDS0 are used in Alphanumeric modes to control the double scan features. CGA software for a a 640x200 pixel resolution is based on a font with eight vertical pixels. A choice of two methods must be considered if the CGA software is operated with monochrome or EGA

monitor having 350 vertical display lines or with any display that has a 640x400 resolution. The first method to consider is preparing a separate font and then setting the Maximum Raster Address (Secondary Register R9) to a value suitable to the monitor.

The second method to consider is using the TDS1 and TDS0 bits to perform double scan control of the CGA rasters. The monochrome or EGA monitor uses a 1.75 x CGA scan for a double scan. A monitor with 640x400 resolution uses a 2 x CGA scan for a double scan. TDS1 and TDS0 select the desired scan as follows:

TDS 1	TDS 0	Scan Factor	
0	0	None	Provides consecutive CGA scan lines.
0	1	1.5x	Double scans all even scan lines. Ten CGA scan lines become 15 scan lines.
1	0	1.75x	Double scans all even scan lines and ever other odd scan line. Ten CGA scan lines become 17 scan lines.
1	1	2x	Double scans all scan lines. Ten CGA scan lines become 20 scan lines.

Cursor Address Offset, CA04-CA00 (37h, D4-D0)

CA04 - CA00 are used to set the cursor position in relation to the displayed characters. Whenever a nonstandard combination of software and monitor is used, the possibility exists that the cursor will not be properly positioned in relation to the characters. If this occurs, CA04 - CA00 are loaded with the two's complement of an offset value to correct the cursor position. The offset value must be determined in relation to the value contained in the Maximum Raster Address register (Address 36h). The range of the Cursor Address Offset is from -16 to +15. If an offset is not required, CA04-CA00 are set to 0h, which is the default or reset value. Also, if an overflow occurs because of the offset, the offset value is corrected to 0h. The Cursor Address Offset value is enabled when the F/O bit (37h, D7) is set to "1."

Address 3Fh

Address 3Fh provides three discrete control registers. One register has one bit (45S), one register has two bits (GDS1 and GDS0), and the third register has five bits (UP4 - UP0). On system reset, the Address 3Fh register is set to 0Ch. Each Address 37h register is described in the following paragraphs.

6845 Expansion, 45S (3Fh, D7)

45S controls writing to the Most Significant Bit (MSB) of all the vertical direction registers. When 45S is set to "1," writing is enabled to the MSB of the Vertical Total, Vertical Displayed, and Vertical Sync Position registers (Secondary Registers R4, R6, and R7,

respectively) and also to Preset Data registers at Addresses 32h, 34h, 3Ah, and 3Ch. Writing to the MSB allows negative values to be written to the registers. If 45S is set to "0," a zero is always written to the MSB of the same registers, which allows the Display Controller to execute IBM software. In Preset Mode, the Preset Data registers always operate as if 45S is set to "1" regardless of how the bit is actually set.

Graphics Scan Control, GDS1 and GDS0 (3Fh, D6 and D5)

GDS1 and GDS0 are used in Graphics modes to control the double scan features similar to the way TDS1 and TDS0 (37h, D6 and D5) are used to control double scan features in Alphnumeric modes. Since the maximum raster quantity for CGA graphics software is two, double scan is performed in two-line cycles, which regard the maximum raster quantity as equivalent to four. Therefore, the double scan increases the raster quantity by the factor determined by the value of GDS1 and GDS0.

When CGA software for a 640x200 pixel resolution is used in graphics mode with a monochrome monitor, a 1.5x double scan is required. When an EGA monitor is used, a 1.75x double scan is required, and a monitor with 640x400 pixel resolution requires a 2x double scan. GDS1 and GDS0 select the required double scan as follows:

GDS 1	GDS 0	Scan Factor	
0	0	None	No increase in CGA scan lines.
0	1	1.5x	200 lines increases to 300 lines
1	0	1.75x	200 lines increases to 350 lines
1	1	2x	200 lines increases to 400 lines

Underline Position, UP4-UP0 (3Fh, D4-D0)

UP4 - UP0 are used in alphanumeric modes with the underline bit ULE (24h, D6) to control the position of the underline with respect to displayed characters. For a monochrome monitor, the default underline position following a system reset is Ch. For an IBM color monitor the underline position is 7h, and for a monitor with 640x400 pixel resolution the underline position is Eh. If the value in UP4-UP0 is greater than the Maximum Raster Address (36h or 3Eh), no underline is displayed.

For characters that are specified to have a blue background (Lower 3 bits of the attribute byte are 001), an underline equivalent to one raster is displayed. In double scan mode, a two-raster underline may be displayed, depending upon the Maximum Raster Address.

UNSUPPORTED AND ADDED OPERATIONS

Most of the modes of the IBM AT color card are supported by the GRiDCase 1500 Series Computer Display subsystem. The following operating modes of the IBM color card are not supported:

1. The low resolution 160x200 Graphics mode is NOT SUPPORTED. The IBM AT documentation makes no mention of how to use the 160x200 Graphics mode and the IBM ROM-BIOS does not support it. The GRiDCase 1500 Series Computer hardware architecture is very similar to the IBM AT. Therefore, any applications that use the 160x200 Graphics mode, and are designed to run on the IBM AT, should not experience difficulty running on the GRiDCase 1500 Series Computer.

2. Interlace modes of the 6845. IBM does not use interlace mode on either of their display boards. Also, when the interlace sync mode is selected on the color card with the IBM color monitor the display appears to vibrate.

The Yamaha V6366 Display Controller emulates all of the features and functions found in the 6845 Display Controller used by the IBM AT. In addition the V6366 provides Preset Data registers for switching between different software and monitor combinations. Additional registers are also supplied for up to 16 color palettes and discrete control functions.

EXTERNAL MONITOR CONNECTOR PIN DEFINITIONS

A CGA-compatible video monitor can be connected to the GRiDCase 1500 Series Computer. The connection is made through the 9-pin, subminiature, "D" shell female receptacle on the back panel of the computer. The Video Output connector pin layout is shown in Figure 7-9 and the pin definitions are given in Table 7-7.

An EGA or CGA video monitor can also be supported through an optional interface module that connects to the computer through the expansion bus interface. The expansion bus interface is described in Chapter 12.

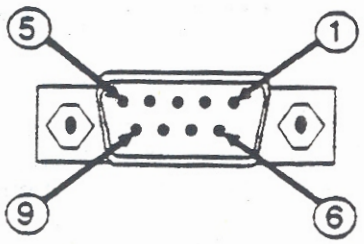


Figure 7-9. Video Output Connector Pin Layout

Table 7-7. Video Output Connector Pin Definitions

Pin Number	Signal Name
1	Signal Ground
2	Signal Ground
3	Red
4	Green
5	Blue
6	Intensity
7	Reserved
8	Horizontal Sync
9	Vertical Sync

CHAPTER 8: KEYBOARD SUBSYSTEM

The GRiDCase 1500 Series computer provides a full-size, 72-key keyboard that emulates all 84-keys of the IBM AT keyboard by using a combination of the "Fn" key and other specially labeled keys. All key combinations required to emulate the AT keyboard are labeled using easy-to-follow color codes. For example, on the Fn key, the letters "Fn" are in blue to indicate that this key is used with other keys that are also labeled in blue. The Fn key is pressed in conjunction with the other keys labeled in blue to emulate the IBM AT operation keys (NmLk, Pause, ...) The standard keyboard operation keys (Shift, CpLk, Return, ...) are labeled in yellow, and the keypad number keys activated by NmLk are labeled in red. All of the standard keyboard letter and number keys, and function keys F1 through F10, are labeled in white. The keystrokes and key-stroke combinations required to emulate the IBM AT keyboard are listed at the end of this chapter.

The keyboard controller is a version of the Intel 8042/8742 Universal Peripheral Interface (UPI-42). The UPI-42 contains an 8-bit microcontroller, 2048 x 8-bit ROM, 256 x 8-bit RAM, dual I/O ports, a timer/counter, and a clock with an external 10 MHz crystal control. Operation of the UPI-42 is controlled through two I/O registers (60h and 64h). Figure 8-1 provides a block diagram of the keyboard subsystem.

NOTE: Throughout this chapter, numbers given in hexadecimal notation have a letter "h" suffix.

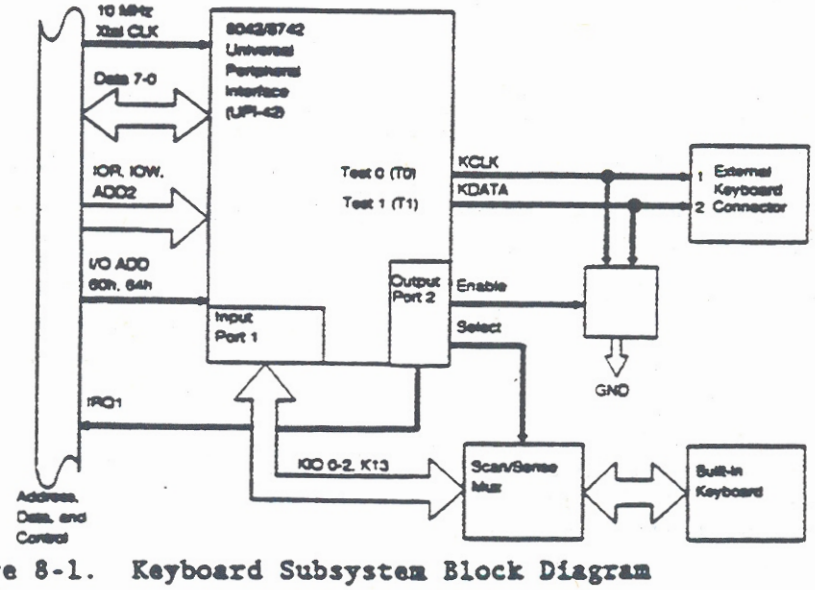


Figure 8-1. Keyboard Subsystem Block Diagram

In addition to the built-in keyboard, the Keyboard Subsystem supports an externally connected IBM AT compatible 84-key keyboard, an IBM AT compatible enhanced keyboard with 101/102 keys, or an external keypad for numbers and mathematic functions. The external keyboard or keypad is connected via a 5-pin DIN type receptacle that is located on the back panel of the GRiDCase 1500 Series computer. The descriptions in this chapter apply to the built-in keyboard unless otherwise specified.

Programs interface to the Keyboard Subsystem via the interrupt handler and service routine provided by the ROM-BIOS. Functionally, the keyboard implementation through ROM-BIOS is identical to the IBM AT keyboard. Also, custom keyboard routines can be implemented by accessing the keyboard controller directly through its dedicated I/O registers.

KEYBOARD CONTROLLER HARDWARE

Communication between the 80286/80386 Microprocessor and the built-in keyboard is handled by the 8042/8742 Universal Peripheral Interface (UPI-42) that is used as a microcontroller-based keyboard controller. The keyboard is a passive array of normally open switches. The keyboard controller scans the switch array and provides the means to encode and debounce the keys. The keyboard controller also provides the 2-key rollover and 3-key lockout features. Therefore, if two keys are depressed simultaneously, the codes for both keys are returned, but if 3 keys are depressed simultaneously, no code is returned. The 3-key lockout does not apply to the Right Shift, Left Shift, Alt/Code, Ctrl, and Fn keys. The keyboard controller communicates with the 80286/80386 Microprocessor through its three-state, 8-bit, bidirectional data bus buffer.

The keyboard controller scans the built-in keyboard for keystrokes or receives serial data from an external keyboard. When the keyboard controller senses a keystroke in the built-in keyboard, it debounces the keystroke and then acts as a keyboard emulator to generate the appropriate keyboard scan code. If the PC-compatible bit is set (Table 8-2), the controller translates the keyboard scan code into a system scan code. In either case, the scan code that results from depressing a key on the built-in keyboard is presented to the microprocessor as a byte of data in the output buffer of the controller.

If a keystroke is sent from an external keyboard, the keyboard controller receives a keyboard scan code in the form of serial data and checks the parity of the data. If the PC-compatible bit is set (Table 8-2), the controller translates the keyboard scan code into a system scan code. In either case, the scan code that results from depressing a key on the external keyboard is presented to the microprocessor as a byte of data in the same output buffer of the

controller that is used for the built-in keyboard. There is no way for the microprocessor to tell if the keystroke came from the built-in keyboard or the external keyboard.

The keyboard controller interrupts the microprocessor when data is placed in its output buffer. With data in its output buffer, the keyboard controller generates an Output Buffer Full (OBF) signal. The OBF signal is sent to the microprocessor as an interrupt request (IRQ1), which initiates ROM-BIOS interrupt INT 09h. The ROM-BIOS provides a 16-character keyboard buffer and a service routine that converts the system scan code into the appropriate ASCII code for use by the display and printer. A status register in the keyboard controller contains data bits that indicate if an error was detected while sending or receiving data.

Command bytes are sent to the keyboard by writing to its input buffer over the same 8-bit data bus that was used to output data. The keyboard is required to acknowledge all data transmissions. The command bytes are described later in this chapter.

RECEIVING DATA FROM THE EXTERNAL KEYBOARD

The external keyboard sends data in a serial format using an 11-bit frame. The first bit is a start bit, and is followed by eight data bits, an odd parity bit, and a stop bit. Data bits are synchronized by a clock pulse supplied by the external keyboard. The data is placed in a 16 byte First-In First-Out (FIFO) buffer. The controller takes the next appropriate byte from the FIFO, places the byte in its output buffer, and then continues with its normal operation. The controller does not put another byte in its output buffer until the previous byte was accepted by the microprocessor.

If the byte of data is received with a parity error, a Resend Command is automatically sent to the external keyboard. If the keyboard controller is unable to receive the data correctly, an FFh is placed in its output buffer, and the parity bit in the status register is set to "1," indicating a receive parity error. The keyboard controller times out if a keyboard transmission does not end within 2 milliseconds. If a time-out occurs, an FFh is placed in the output buffer of the keyboard controller, and the receive time-out bit in the status register is set to "1." No retries are attempted on a receive time-out error.

Scan Code Translation

Keyboard scan codes are generated by the external keyboard when any key is pressed or released. A 16-byte first-in first-out (FIFO) buffer stores the keyboard scan codes until the keyboard controller is ready to receive them. If more than 16 bytes of data accumulate

in the keyboard buffer, the 17th byte is an overrun code and all additional bytes are lost.

The external keyboard scan codes received from the keyboard buffer are placed in the output buffer of the keyboard controller unless the PC-compatible bit (Table 8-2) is set. If the PC-compatible bit is set, the keyboard scan codes are translated into system scan codes before being placed in the output buffer of the keyboard controller. When the output buffer is loaded, the Output Buffer Full (OBF) flag is set, which generates an interrupt request (IRQ1). IRQ1 initiates ROM-BIOS interrupt INT 09h to the microprocessor. Each key or combination of keys that is pressed generates a different keyboard scan code and results in a byte of data being sent to the microprocessor.

SENDING DATA TO THE EXTERNAL KEYBOARD

Data bits are sent from the keyboard controller to the external keyboard in the same serial format used to receive data from the keyboard. A parity bit is inserted by the keyboard controller. The keyboard must start clocking data from the keyboard controller within 15 milliseconds, and complete clocking within 2 milliseconds, or a time-out error occurs. A time-out error while sending data causes code FEh to be placed in the output buffer of the keyboard controller, and the transmit time-out error bit in the status register is set to "1." No retries are attempted by the keyboard controller for any transmission error.

Inhibit Function

The GRiDCase 1500 Series computer does not use the keyboard inhibit switch (input port 60h, bit 7). However, the keyboard can be disabled by a keyboard command (ADh) and reenabled by a second keyboard command (AEh). Keyboard commands are described later in this chapter.

Keyboard Controller Interface

The keyboard controller communicates with the microprocessor through two separate registers. One register serves as a Status register and Input Buffer, while the second register serves as an Input and Output Buffer. These registers are accessed via the microprocessor I/O addresses 60h and 64h, as described in subsequent paragraphs.

ROM-BIOS SERVICE ROUTINE

The keyboard interrupt handler and service routine functions provided by ROM-BIOS are summarized in the following list and described in the subsequent paragraphs. The ROM-BIOS also provides a 16-character buffer for characters from the keyboard.

Interrupt (Hex)	Function No. (AH reg)	Description
09	--	Keyboard hardware interrupt (IRQ1)
16	00	Read next character from the keyboard
16	01	Test for a character from the keyboard
16	02	Read shift status from the keyboard
16	05	Place ASCII Character and Scan Code in keyboard Buffer as if a key was pressed.
16	10	Extended Read for Enhanced Keyboard
16	11	Extended ASCII Status for Enhanced Keyboard
16	12	Read Extended Shift Status Register for Enhanced Keyboard

NOTE: Throughout this chapter the term "enhanced keyboard" refers to an IBM AT compatible, 101/102-key keyboard connected through the external keyboard connector. The term "external keyboard" refers to any externally connected AT compatible standard 84-key or enhanced keyboard.

ROM-BIOS Service Routine Descriptions

The keyboard interrupt service routine uses the 80286/80386 Microprocessor accumulator (AX). The 16-bit AX accumulator is separated into two 8-bit registers, which are designated AH (high 8-bits) and AL (low 8-bits). All of the service routine functions can change the contents of AX and the microprocessor Flag registers. However, the contents of all other registers are unchanged. The microprocessor Zero (Z) flag is used in some routines to indicate the results of operations.

When a keyboard key is depressed or released, the keyboard controller receives the keystroke information from the built-in keyboard, or it receives the equivalent scan code in the form of serial data from the external keyboard. By placing the byte of data in its output buffer, the keyboard controller initiates an interrupt request. The keyboard interrupt request (IRQ1) is issued by the keyboard controller when its Output Buffer Full (OBF) flag is set to "1." IRQ1 is sent to the microprocessor interrupt controller (Part of the F3010 Peripheral Controller), which causes the hardware interrupt (09h) to invoke the ROM-BIOS keyboard interrupt routine.

During each hardware interrupt (09h) from the keyboard the ROM-BIOS generates an interrupt 15h with the AH register set to 4Fh, which initiates the keyboard interrupt handler. The keystroke input is then passed as a system level scan code to the AL register with the carry flag set to "1." This operation allows an application program to intercept each system level scan code prior to its being handled by the INT 09h interrupt service routine. During the intercept operation, the application program may modify or disregard the scan code. On exit from interrupt 15h, if the carry flag is set to "0," the scan code is ignored by the keyboard interrupt handler.

The ROM-BIOS interrupt routine (INT 09h) then translates the system level scan code of the depressed or released key into an extended ASCII code. The ROM-BIOS uses software interrupt 16h to handle the operations specified by the extended ASCII code.

Extended ASCII code supports one-byte character codes, with values from 0 to 255. It also supports code for some extended keyboard functions and functions handled within the keyboard ROM-BIOS routine and interrupts.

The software interrupt (16h) routine supports the following functions:

AH = 00h Read Next Character from the Keyboard

On Exit:
- AH = Scan Code
- AL = ASCII character

AH = 01h Test for a Character from the Keyboard

On Exit:
- Zero Flag = 1 No code is available
- Zero Flag = 0 Keyboard code available
- AH = Next available scan code
- AL = Next available ASCII character

AH = 02h Read Current Shift Status from Keyboard

On Exit:
- AL = Keyboard shift status
- When set to "1," the AL register bits indicate:
  - 0 = Right Shift key depressed
  - 1 = Left Shift key depressed
  - 2 = Ctrl key depressed
  - 3 = Alt key depressed
  - 4 = Scroll Lock (ScrLk) active
  - 5 = Number Lock (NmLk) active
  - 6 = Caps Lock (CpLk) active
  - 7 = Insert (Ins) key active

AH = 05 Place ASCII Character and Keystroke input in Keyboard Buffer as if the key was pressed

On Entry:
    CH = Scan Code
    CL = ASCII Character

On Exit:
    AL = 00h Successful operation
    AL = 01h Unsuccessful operation (Buffer Full)
Carry Flag = 1 if error

AH = 10h Extended Read for Enhanced Keyboard

AH = 11h Extended ASCII Status for Enhanced Keyboard

On Exit:
    Zero Flag = 1 No code is available
    Zero Flag = 0 Keyboard code available
    AH = Next available scan code
    AL = Next available ASCII character

AH = 12h Read Extended Shift Status for Enhanced Keyboard

On Exit:
    AL = Keyboard shift status
    When set to "1," the AL register bits indicate:
        0 = Right Shift key depressed
        1 = Left Shift key depressed
        2 = Ctrl key depressed
        3 = Alt key depressed
        4 = Scroll Lock (ScrLk) active
        5 = Number Lock (NmLk) active
        6 = Caps Lock (CpLk) active
        7 = Insert (Ins) key active
    AH = Extended Shift Status
    When set to "1," the AH register bits indicate:
        0 = Left Ctrl key depressed
        1 = Left Alt key depressed
        2 = Right Ctrl key depressed
        3 = Right Alt key depressed
        4 = Scroll Lock (ScrLk) depressed
        5 = Number Lock (NmLk) depressed
        6 = Caps Lock (CpLk) depressed
        7 = System Request (Sys Req) depressed

Extended ASCII Characters

An extended ASCII code is required whenever the character being entered from the keyboard is not represented by a standard ASCII code character. When an extended ASCII code is used, a character code of 000 (null) is returned in the AL register. The AL register contents provide a flag for the operating system and application programs. When the 000 character code is read, the program must examine a second code, which is the actual code for the function to be performed. Generally, the second code is the scan code of the function to be performed.

Shift Operations

The keyboard subsystem supports seven types of shift operations that are activated by depressing one or more keys. When a shift operation is activated, it causes another set of functions to be performed by depressing associated keys. The six types of shift operations are:

1. Shift
2. Function (Fn)
3. Control (Ctrl)
4. Alternate (Alt)
5. Caps Lock (CpLk)
6. Scroll Lock (ScrLk)
7. Number Lock (NmLk)

Shift operations are generally handled within the ROM-BIOS and are not apparent to the operating system or application program. The current status of the active shift operations can be determined at anytime via the appropriate ROM-BIOS service routine function.

Several combinations of shift operations are used to provide special operations. These combinations are described in subsequent paragraphs under special operations. The following paragraphs describe the shift operations.

Shift

The right and left Shift keys (yellow marking) are depressed and held to change the scan code produced by depressing any letter, number, or symbol key on the standard keyboard (white marking). The effect is to change the number and symbol keys from lower case to upper case. That is, the symbol shown on the upper portion of the key cap is generated instead of the number or symbol shown on the lower portion of the key cap. Also, the letter keys change case depending upon the state of the Caps Lock (CpLk key) operation. If CpLk (yellow marking) is toggled ON, depressing Shift and any letter key produces a lower case letter. When CpLk is toggled OFF, depressing the Shift key and any letter key produces an upper case letter.

Function (Fn)

The Fn key is unique to the built-in keyboard of GRiD Systems computers. Its operation is similar to a Ctrl key in that the Fn key is depressed and held while depressing a second associated key to activate a desired operation. Both shift operations and special operations are activated through use of the Fn key. The Fn key is marked in blue and operations activated via the Fn are also marked in blue as an aid to identification. Activating operations via the Fn key allows the operations to be overlaid on other keys, which reduces the total number of keys required on the built-in keyboard without reducing the number of operations available.

Control (Ctrl)

The Ctrl key (yellow marking) is depressed and held while depressing a specified number and letter key (white marking). The effect is to generate a scan code that is used to control operations. The number or letter keys used with the Ctrl key, and the resulting operation may be defined by either the operating system or by an application program.

Alternate Control (Alt)

The Alt key (yellow marking) works in a similar manner to the Ctrl key, and thereby generates an alternate, or second, set of scan codes used to control operations within an application program.

Caps Lock (CpLk)

The CpLk key (yellow marking) provides an ON/OFF toggle operation that controls the scan codes produced by depressing any letter key (white marking). When the CpLk operation is ON, depressing the letter keys produces scan codes for capital letters. If the CpLk operation is OFF, depressing the letter keys produces lower case letters. Depressing the Shift key reverses the operation selected by the CpLk key. The number and symbol keys are not affected by the CpLk operation.

Scroll Lock (ScrLk)

The ScrLk key (blue marking) provides an operation that is toggled ON and OFF by holding down the Fn key and then depressing the letter S key. The ROM-BIOS keyboard service routine keeps a flag to indicate whether the ScrLk operation is toggled ON or OFF. An application program can use the resulting scan code to control scrolling operations on the display screen.

Number Lock (NmLk)

The NmLk key (blue marking) provides an operation that is toggled ON and OFF by holding down the Fn key and then depressing the letter N key. The NmLk key operation is handled by the ROM-BIOS and the keyboard controller. If the NmLk operation is ON, the keypad internal to the keyboard (red numbers) is activated via the keyboard controller. Depressing the number keys then results in generating scan codes for the numbers. If the NmLk operation is OFF, the normal operations of the keypad keys (numbers and letters) are performed. However, the keypad can be temporarily activated by just holding down the Fn key and then pressing a keyboard key. Also, the MS-DOS command MODE NUMPAD=(OFF|ON) can be used to active or deactivate the built-in keyboard number pad.

When the GRiD Systems Model 34160 External Keypad is connected, it also deactivates the keypad imbedded in the built-in keyboard (red numbers). If the external keypad is connected and NmLk is active, depressing the keypad keys will generate letter scan code instead of number scan codes.

Special Operations

The keyboard subsystem also supports nine types of special operations that are activated by depressing a key or a combination of several keys. Each special operation and the key or key-combination used to invoke it is unique. The special operations are generally handled within the ROM-BIOS keyboard routine and are not apparent to the operating system or application program. The nine types of special operations are:

1. System Reset
2. Break
3. Pause
4. Print Screen
5. Echo
6. System Request
7. Display
8. Color Map
9. Processor Speed

Combinations of the Alt, Ctrl, and Shift keys are used to initiate some special operations. If the combination being used is not recognized by the keyboard interrupt routine, only one of the keys may be recognized depending upon the following priority: Alt is the highest, Ctrl is next, and Shift is the lowest.

The built-in keyboard has the special operations overlaid on the letter keys. These special operations become active when the Fn key is depressed and held while depressing the associated letter key. This method of overlaying special operations on the letters

keys reduces the total number of keys required on the built-in keyboard. The special operations overlaid on the letter keys include Break, Pause, PrtSc, and Echo.

System Reset

A system reset is initiated by depressing and holding the Alt and Ctrl keys and then pressing the Del key. This combination of keys is recognized by the ROM-BIOS as a system reset. A system reset results in returning the entire computer to a known initialized state. A system reset operation may cause a loss of data if performed while a program is running or when data or pointers are contained in system registers.

Break

A Break operation is initiated by depressing and holding the Fn key while depressing the letter B key. This key combination is recognized by the ROM-BIOS as a user initiated interrupt equivalent to depressing the Ctrl and Break keys on the IBM keyboard. The operating system interprets this as a command to clear the keyboard buffer and the insert a Ctrl-C in the buffer.

Pause

The Pause operation is initiated by depressing and holding the Fn key while depressing the letter Q key. This key combination is recognized by the ROM-BIOS as a command to temporarily suspend an operation such as scrolling or printing. Once the operation is suspended, it will resume operation when any character key is depressed. The Pause is handled by the ROM-BIOS and is not apparent to either the operating system or application program, and the keystroke used to resume operation after a Pause is discarded.

Print Screen (PrtSc)

The Print Screen (PrtSc) operation is initiated by depressing and holding the Fn key while depressing the letter W key. This key combination is recognized by the ROM-BIOS as a command to print the contents of the display screen.

When the ROM-BIOS receives a Print Screen command, interrupt 05H invokes a service routine that first saves the current cursor position so that it can be restored at completion of the operation. Next, the printer status is checked for on-line, not busy, and paper available. Then, if no errors are detected, the contents of the screen are output to the printer. Finally, the cursor is restored to its previous position on the screen. If subsequent Print Screen commands are issued while one is being performed, the subsequent commands are ignored.

The status of the print screen operation is contained at location 50:00h as follows:

Contents	Remarks
0	Indicates a successful print screen operation or that no operation was called.
1	Print screen operation is in progress. Ignore any subsequent Print Screen commands.

Echo

The Echo operation is initiated by depressing and holding the Fn key while depressing the letter E key. This key combination is recognized by the ROM-BIOS as an extended ASCII code 72h, which acts as an ON/OFF toggle. When the Echo operation is ON, characters sent to the display screen are also sent to the printer. The Echo operation is turned OFF when the same keys are pressed a second time.

If the Echo operation is ON and no printer is recognized, depressing a key can cause the system to hang. When the key is pressed, the system detects that a printer is not present, and after it times out, an error message is written to the display. Each character of the error message is also sent to the printer. This, in turn, causes another error message to be sent to the display. The result is an endless loop that prevents other operations from being performed. If the system hangs, a system reset is required to reinitialize the system, and data in the registers will be lost.

System Request (Sys Req)

The System Request operation is initiated by depressing the Sys Req key. Depressing the Sys Req key is recognized by the ROM-BIOS as a special operation command. The special operation initiates an interrupt 15h with 8500h in the microprocessor AX register. When the Sys Req key is released, the ROM-BIOS generates interrupt 15h with 8501h in the microprocessor AX register.

If an application program uses Sys Req to initialize a particular process, it must provide the following operations:

1. Save the previous address.

2. Overlay interrupt vector 15h with a new interrupt routine.

3. Preserve the values in all microprocessor registers except AX using the new interrupt routine.

4. Check register AH for the value 85h.
   If yes, the process may begin.
   If no, return to address saved in Step 1.

5. Complete the process and return to the address saved in Step 1.

NOTE: The Display, Color Map, and Processor Speed are special operations unique to GRiD Systems computers. These operations are provided in the ROM-BIOS to provide operating flexibility. The same operations are available through the MS-DOS MODE command and through the ROM Subsystem Functions for interrupt 15h (refer to Chapter 3).

If an application program takes over the keyboard interrupt service routine, the key combinations used for special operations may not be interpreted correctly.

Display

The Display operation is initiated by depressing and holding the Ctrl and Alt keys while depressing the Tab key. This key combination is recognized by the ROM-BIOS as a toggle command to switch the video output between the built-in display and an external display connected through the video output connector. Each time the key combination is depressed, the video output is switched between the two display devices.

Color Map

The color map operation is initiated by depressing and holding the Ctrl and Alt keys while depressing the Fn and (right arrow) keys. This key combination is recognized by the ROM-BIOS as a command to increment through the six color mapping modes. A similar command is implemented by depressing and holding the Ctrl and Alt keys while depressing the Fn and (left arrow) keys. This second command decrements the six color mapping modes. The color map index is stored at absolute address 40:A7h.

Processor Speed

The processor speed operation is initiated by depressing and holding the Ctrl and Alt keys while depressing the Fn and (down arrow) keys. This key combination is recognized by the ROM-BIOS as a command to change the microprocessor speed from high to low. The low processor speed is 5 MHz for the Model 1520 and 6.5 MHz for the Model 1530. A similar command is implemented by depressing and holding the Ctrl and Alt keys while depressing the Fn and (up

arrow) keys. This second command changes the microprocessor speed from low to high. The high processor speed is 10 Mhz for the Model 1520 and 12.5 Mhz for the Model 1530.

DIRECT KEYBOARD ACCESS

The 80286/80386 microprocessor communicates with the keyboard controller through two bidirectional I/O registers and a series of keyboard controller commands. The keyboard controller then communicates with the built-in keyboard via two 8-bit parallel ports and a series of keyboard commands. For the built-in keyboard, scan and sense lines are connected to the two 8-bit parallel ports, and the controller uses these lines to detect keystrokes and send keyboard control signals. The controller emulates the keyboard operation by generating the appropriate keyboard scan codes and interpreting the keyboard commands. Except for the keystroke data, the controller emulates all other activity of the built-in keyboard.

To communicate with an external keyboard, the keyboard controller uses its two test ports, TEST0 and TEST1, to send and receive serial data. The TEST0 port supports the keyboard clock (KCLK) line and the TEST1 port supports the keyboard data (KDATA) line. Since the external keyboard provides its own data buffers and scan code generator, the data transferred between it and the keyboard controller is always in serial format. The transferred data includes the keyboard commands and responses as well as keystrokes.

The keyboard controller I/O registers that interface with the microprocessor are described in the following paragraphs. Descriptions of the keyboard controller ports that interface with the keyboards are provided following the I/O register descriptions.

I/O Registers

The keyboard controller communicates with the microprocessor through two bidirectional I/O registers. The I/O register addresses, names, and the register uses are given in the following list. Descriptions of the register are provided in the following paragraphs.

I/O Address	Register Name	Use
60h Read	Keyboard Output	Keyboard Response Codes, system scan codes, data, etc.
60h Write	Keyboard Input	Keyboard Commands and Data
64h Write	Controller Input	Keyboard Controller Commands
64h Read	Keyboard Status	Keyboard Subsystem Status

To provide the necessary communications, the keyboard controller supports two types of commands that use the I/O registers as follows:

1. Keyboard Controller commands are output from the microprocessor through the Controller Input register (64h, Write). These commands may require additional data, which is supplied by writing to the Keyboard Input Buffer (60h Write).

2. Keyboard commands are output from the microprocessor through the Keyboard Input Buffer (60h, Write). The Keyboard Status register (64h, Read) provides a Command/Data bit (bit 3) that is cleared when a keyboard command is written. Responses to Keyboard commands are returned through the Keyboard Output Buffer (60h, Read).

Keyboard Output Buffer (60h, Read)

The Keyboard Output buffer is an 8-bit output register that is located at I/O address 60h. The output buffer is used to send data, in the form of system scan codes from the keyboard controller to the microprocessor. Responses to keyboard commands are also sent through the same output buffer. Before reading the output buffer, the microprocessor should check that the Output Buffer Full (OBF) flag (Status register, 64h, Read, bit 0) is set to "1."

The keystroke data sent to the microprocessor is in the form of scan codes. Either of two scan codes sets can be used depending upon the state of the IBM PC Compatibility bit (Keyboard Controller Command byte, bit 6) as follows:

Bit 6	Mode	Selected Scan Code Set
0	Not Compatible	Keyboard Scan Code set
1	IBM PC Compatible	System Scan Code set

The scan code sets are described in subsequent paragraphs under Key Definitions.

Keyboard Input Buffer (60h, Write)

The Keyboard Input buffer is an 8-bit input register that is located at I/O address 60h. In order to write to the input buffer, the Input Buffer Full (IBF) flag (Status register, 64h, Read, bit 1) must be cleared to indicate that the input buffer is not full. Unless the Controller Input register (64h, Write) is written first, writing to I/O address 60h clears the command/data flag (Status register 64h, Read, bit 3). Clearing the command/data flag indicates that the byte in register 60h is a keyboard command.

When the Controller Input register (64h) is written first, a subsequent byte written to address 60h is interpreted as keyboard controller data that must accompany a keyboard controller command.

Since the controller also emulates keyboard operations for the built-in keyboard, it interprets both the keyboard commands and the keyboard controller commands and sends the appropriate signals on to the keyboard as required to enable the scan and sense lines. The keyboard controller also generates responses to the keyboard commands and sends the responses to the microprocessor.

The same keyboard commands are sent directly to external keyboard if one is connected to the subsystem. The external keyboard interprets the keyboard commands and generates the appropriate responses, which are returned to the keyboard controller.

Controller Input Register (64h, Write)

The Controller Input register is an 8-bit buffer that is located at I/O address 64h. Writing to I/O address 64h sets the command/data flag (Status register 64h, Read, bit 3) to indicate that a keyboard controller command is being written. Subsequent bytes written to the Keyboard Input Buffer (60h) are then interpreted as data to accompany the keyboard controller command. The keyboard controller commands written to I/O address 64h are listed in Table 8-1.

Table 8-1. Keyboard Controller Commands

Command (Hex)	Command Name	Definition
20	Read Command Byte	Places the keyboard controller command byte in the Keyboard Output Buffer (60h, Read). The keyboard controller command byte is defined in Table 8-2.
60	Write Command Byte	Writes the keyboard controller command byte as the next byte of data in the Keyboard Input Buffer (60h, Write). The keyboard controller command byte is defined in Table 8-2.
AA	Self-Test	Initiates a keyboard controller internal diagnostic and if successfully completed, it places 55h in the Output Buffer (60h). If not successful, FFh is placed in the Output Buffer.

Table 8-1. Keyboard Controller Commands (Continued)

Command (Hex)	Command Name	Definition
AB	Interface Test	Initiates a test of the keyboard clock and data lines. The test results, placed in the Output Buffer (60h), are as follows: Hex — Error 00 — No Error Detected 01 — Clock line stuck low 02 — Clock line stuck high 03 — Data line stuck low 04 — Data line stuck high
AC	Diagnostic Dump	Reserved
AD	Disable Keyboard	Sets bit 4 of the keyboard controller command byte, which inhibits input from the external keyboard and stops scanning of the built-in keyboard. Refer to Table 8-2.
AE	Enable Keyboard	Clears bit 4 of the keyboard controller command byte. See command ADh.
C0	Read Input Port	Transfers data from the keyboard input port to the Output Buffer (60h). Use only when the Output Buffer is empty.
D0	Read Output Port	Transfers data from the keyboard output port to the Output Buffer (60h). Use only when the Output Buffer is empty.
D1	Write Output Port	Transfers the next byte of data written in the Input Buffer (60h) to the keyboard output port. NOTE: Bit 0 of the output port is connected to system reset. Therefore, writing a 0 to this bit causes a microprocessor reset.
E0	Read Test Inputs	Transfers data from lines T0 and T1 into the Output Buffer (60h). Output buffer bit 0 represents T0 and bit 1 represents T1.
F0-FF	Pulse Output Port	Pulses the selected bits of the output port. Bits 0 through 3 of this command are used to select bits 0 through 3, respectively of the output port. If a

Table 8-1. Keyboard Controller Commands (Continued)

Command (Hex)	Command Name	Definition
		bit of this command is set to "0," the respective bit of the output port is pulsed low for approximately 6 microseconds. When a bit in this command is set to "1," the respective bit of the output port is not pulsed.
NOTE:		Bit 0 of the output port is connected to system reset. Therefore, writing a 0 to this bit causes a microprocessor reset.
		The following keyboard controller commands are unique to the GRiDCase 1500 Series computers.
0A	Input Translation Codes	Changes the keyboard scan codes. Following the translate command (0Ah), two bytes are written to the Output Buffer (60h). The first byte written is the original scan code. Then, a replacement scan code is written. When the original scan code key is subsequently depressed, the controller generates the replacement scan code. Refer to the following paragraphs on Key Translation.
09	Set Backlight Time Out	Sets the number of minutes that the keyboard can remain inactive before the display backlight is turned OFF. This function is also controlled by the MODE command and by a BIOS Subsystem Function. Refer to Chapter 7.
08	Keyboard Controller Version	Returns the date in two bytes as follows: Byte 1 = mmmddddd Byte 2 = yyyyyyym Where: m is the month (1-12), d is the day (1-31), and y is the year (0-128, with 0=1980 and 8=1988).
07	Disable Numpad	Disables the number pad (red numbers) on the built-in keyboard. If the number pad is disabled and NmLk is activated the built-in numpad keys return letters instead of numbers.
06	Enable Numpad	Enables the number pad (red numbers) on the built-in keyboard.

Table 8-2. Keyboard Controller Command Byte Bit Definitions

Bit	Name	Definition
7	Reserved	Should be written as 0.
6	Compatibility	IBM PC compatibility mode. When set to "1," the keyboard scan codes are translated into system scan codes, which are IBM PC compatible. Also, the 2-byte break sequence is converted to the 1-byte IBM PC format. If the bit is cleared (0), no conversion occurs and the keyboard scan codes are sent to the microprocessor. Refer to the following paragraphs on Key Definitions.
5	Mode	Not used.
4	Disable Keyboard	When set to "1," the keyboard interface is disabled by driving the Clock signal line low. The keyboard controller then stops scanning the built-in keyboard and communication with the external keyboard is inhibited. Bit 4 must be cleared to enable the keyboard.
3	Inhibit Override	This bit is always set to "1" because the inhibit function is not supported.
2	System Flag	The value written to this bit is placed in the Status register (64h, Read) bit 2.
1	Reserved	Should be written as 0.
0	Enable Output Buffer Full (OBF) Interrupt	When set to "1," OBF is enabled to generate an interrupt when the output buffer (60h) contains data.

Key Translation

Key translation is a feature of the GRiDCase 1500 Series computers. This feature allows the default keystroke data (scan code) to be changed to a different value. This has the effect of moving frequently used keys to a more convenient location on the keyboard. Although the location where a key will be recognized can swapped to other locations on the keyboard, it is seldom desirable to swap more than one or two key locations. For example; the Escape (Esc)

key function may be moved to the location of the Caps Lock (CpLk) key and vice versa. The physical location of the keys remains the same, only the scan codes are changed so that pressing the Esc key causes the caps lock action and pressing the CpLk key causes the Escape action. This capability is used by international keyboards.

NOTE: The scan codes used for the key translation feature are not the system scan codes recognized by the ROM-BIOS. The scan codes used for key translation are generated as a result of keystrokes and are referred to as keyboard scan codes. Refer to the following paragraphs on Key Definitions.

In addition to swapping the locations where keys are recognized, the key translation feature can be used to disable a key or to reenable a key that was previously disabled. The key translation feature uses the Controller Input register by writing an 0Ah command to address 64h. Next, the current or original scan code is output to the Input Buffer (60h). Then, the desired scan code is output to the Keyboard Input Buffer (60h). The command syntax is as follows:

0Ah	Translate Key Command
xxh	Current scan code
yyh	Desired scan code

The current scan code refers to the keyboard scan code originally assigned to the key as listed in Table 8-5, or the scan code currently assigned to the key if the key was previously translated.

The desired scan code refers to the keyboard scan code to be generated by depressing the same key after the translation is complete. The desired scan code must be selected from the keyboard scan codes listed in Table 8-5.

The keyboard status and input/output buffers must be checked when sending data. Therefore, the following protocol is necessary to successfully complete the translation operation.

1. Wait until the keyboard controller is ready for input.

2. Send command 0Ah to address 64h.

3. Wait until the keyboard controller is ready for input.

4. Send the current keyboard scan code to address 60h.

5. Wait until the keyboard controller is ready for input.

6. Send desired keyboard scan code to address 60h.

7. Wait until the keyboard controller is ready for input.

Restrictions:

The following keys cannot be translated:

1. Four special operation keys cannot be translated to other keys because they return two keyboard scan codes. The four special operations are:

   Break
   Echo
   Pause
   PrtSc

   These four special operations are overlaid on letter keys b, e, q, and w. The letters are translatable to other keys, but the operations specified by depressing the same keys after depressing the Fn key are not translatable.

2. The imbedded numeric keypad (red numbers) are not translatable. The function and number keys are:

Function	Number (red)
Fn + 7	7
Fn + 8	8
Fn + 9	9
Fn + u	4
Fn + i	5
Fn + o	6
Fn + j	1
Fn + k	2
Fn + l	3
Fn + m	0

   Scan codes for the usual number and letter operations of the keypad keys (white numbers and letters) are translatable. However, to translate the usual scan code without affecting the number pad capability, add 80h to the desired scan code.

3. The Left Shift, Right Shift, Alt, and Fn keys cannot be translated.

Examples:

For the following examples, there are two types of keyboard operations on the built-in keyboard. The two types are called "defined" and "undefined." Defined operations are assigned a unique scan code that is generated by depressing a key or combination of keys. Typical defined operations occur by depressing any letter key, a letter and shift key combination, or depressing the Ctrl key and a letter key combination. An undefined operation is a key or key combination that does not have a scan code currently assigned. Only GRiD computer function (Fn) key sequences can be undefined. To define an undefined function key sequence, the value 80h must be added to the desired scan code. The following examples show how key translation is used swap the scan codes in defined and undefined operations.

1. To swap two defined keyboard operations such as CpLK and Ctrl:

   a. Locate the keyboard scan codes for CpLk (58h) and Ctrl (14h) in Table 8-5.

   b. Enter the translate commands for either scan code (this example selects CpLk first).

      0Ah Translate Key Command
      14h Ctrl scan code
      58h CpLk scan code

      At this point, depressing the Ctrl key or the CpLk key will return a CpLk scan code.

   c. Enter the translate commands for the other scan code (in this case the Ctrl scan code).

      0Ah Translate Key Command
      58h CpLk scan code
      14h Ctrl scan code

      At this point, depressing the Ctrl key returns the CpLk scan code and depressing the CpLk key returns the Ctrl scan code.

2. To disable a previously defined function (Fn) such as the "[" (left bracket) key:

   a. Locate the keyboard scan code for the "[" character (54h) in Table 8-5.

b. Enter the translate commands to disable the scan code for "[."

```
0Ah  Translate Key Command
54h  "[" scan code
00h  Entering 00h disables the scan code
```

The scan code for "[" is not output when the Fn and "," (comma) keys are depressed.

3. To output the "=" scan code when the "[" key is depressed:

a. Locate the keyboard scan code for the "=" (55h) and "[" (54h) characters in Table 8-5.

b. Enter the translate commands to swap the "=" scan code for the "[" scan code.

```
0Ah  Translate Key Command
54h  "[" scan code
55h  "=" scan code
```

The "=" scan code is now output when the Fn and "," keys are depressed.

4. To overlay a previously undefined function over a defined letter key:

NOTE: This example causes the "/" (virgule) scan code (4Ah) to be output when the function (Fn) and letter "d" keys are depressed. The normal operation of the "d" key is not affected.

a. Locate the keyboard scan codes for the letter "d" key (23h) and the "/" character (4Ah) in Table 8-5.

b. Enter the translate commands to overlay an undefined function on the letter "d" key.

```
0Ah  Translate Key Command
23h  Letter D scan code
CAh  The value for an undefined operation (80h)
     is added to the scan code value for the "/"
     character (4Ah).
```

When the function (Fn) and letter "d" keys are depressed, scan code 4Ah (/) will be output.

5. To define a function (Fn) that previously was undefined:

NOTE: This example causes the "]" (right bracket) scan code (5Bh) to be output when the function (Fn) and letter "d" keys are depressed. The normal operation of the "d" key is not affected.

a. Locate the keyboard scan code for the letter "d" key (23h) and the "]" character (5Bh) in Table 8-5.

b. Enter the translate commands to overlay an undefined function on the letter "d" key.

```
0Ah  Translate Key Command
23h  Letter D scan code
BDh  The value for an undefined operation (80h)
     is added to the scan code value for the "]"
     character (5Bh).
```

When the function (Fn) and letter "d" keys are depressed, scan code 5Bh (]) will be output.

Status Register (64h, Read)

The Status register is an 8-bit buffer that is located at I/O address 64h. The Status register provides information about the state of the keyboard subsystem, and may be read at any time. The bit definitions for the Status register are as follows:

Bit	Bit Name	Definition
7	Parity	A "0" indicates odd parity on the last byte received. A "1" indicates even parity on the last byte received. The external keyboard always sends odd parity.
6	Receive Time-Out	A "1" indicates that the keyboard controller began receiving data, but the keyboard did not complete the operation within the time programmed for a receive time-out.
5	Transmit Time-Out	A "1" indicates that the keyboard controller began sending data, but the keyboard did not complete the operation within the time programmed for a transmit time-out.

Bit	Bit Name	Definition
NOTE:		If a transmit byte was clocked out but a response was not received within the time-out limit, both the transmit and receive time-out bits are set to "1." If the transmit byte was clocked out but the response was received with a parity error, the transmit time out and parity error bits are both set to "1."
4	Inhibit Switch	This bit indicates the state of the inhibit switch. For the GRiDCase 1500 Series computer, this bit is always set to "1," which indicates that the keyboard is not inhibited.
3	Command/Data	The keyboard controller input buffers are addressed by either I/O address 60h or 64h. Address 60h is a keyboard data and command port while address 64h is a keyboard controller command and status port. Writing to address 60h clears this bit to "0." Writing to address 64h sets this bit to "1." The keyboard controller uses this bit to determine if the byte in its Input Buffer is a data byte or a command byte.
2	System Flag	The keyboard controller clears this bit to "0" following a successful Power-On Self-Test (POST). The bit is monitored by the microprocessor during a reset operation. The bit value can be changed by writing to the system flag bit in the keyboard controller command byte. Refer to Table 8-2.
1	Input Buffer Full	A "1" indicates that the data in its Input Buffer (60h) has not been read by the keyboard controller. When the buffer is read this bit is set to "0."
0	Output Buffer Full	A "1" indicates that the Output Buffer (60h) is full but the data has not been read by the microprocessor. When the microprocessor reads the output buffer, this bit is set to "0." The Output Buffer Full (OBF) bit is used to generate the system interrupt (09h) that indicates to the BIOS that a key has been pressed.

Keyboard Ports

For communication between the keyboard controller and the built-in keyboard, the keyboard controller provides two 8-bit parallel ports. One port is used for input to the keyboard from the keyboard controller. The other port is for output from the keyboard to the keyboard controller. Both ports provide discrete signals for control and status information in addition to the scan and sense lines used to transfer keystroke information. Keystroke information received from the built-in keyboard is converted by the keyboard controller into keyboard scan codes. Then, if the IBM PC Compatibility mode is selected (refer to Table 8-2), the controller converts the keyboard scan codes into system scan codes, which are recognized by the ROM-BIOS.

The external keyboard or keypad, if connected, inputs its keyboard data (KDATA) and keyboard clock (KCLK) signals to the keyboard controller. The KDATA signal is connected to the TEST1 input and KCLK is connected to the TEST0 input. All data and control information passed through the TEST0 and TEST1 inputs is in a serial data format as previously described. For each keystroke on the external keyboard or keypad, the keyboard controller receives an appropriate keyboard scan code. Then, if the IBM PC Compatibility mode is selected (refer to Table 8-2), the controller converts the keyboard scan codes into system scan codes, which are recognized by the ROM-BIOS.

There is no way for the ROM-BIOS to tell whether the system scan code was generated as a result of depressing a key on the built-in keyboard or as a result of a keyboard scan code from the external keyboard or keypad. The following paragraphs describe the I/O Ports and Test Inputs.

Input Port

The built-in keyboard input port bits 0 through 7 are assigned to Port 1 (pins 27 through 34, respectively) on the keyboard controller. The input port bit definitions are given in the following list.

Input Port			Mask	
Pin	Bit	Desig	Bit	Definition
34	7	P17	1	Enable Scan Lines: 0 = OFF, 1 = ON
33	6	P16	0	Display type (Always 0)
32	5	P15	1	Backlight: 0 = OFF, 1 = ON
31	4	P14	1	Scanlatch: 0 = Scan Line 1, 1 = Enable Sense Lines
30	3	P13	-	Scan/Sense Line K13, Bit 3
29	2	P12	-	Scan/Sense Line K102, Bit 2
28	1	P11	-	Scan/Sense Line K101, Bit 1
27	0	P10	-	Scan/Sense Line K100, Bit 0

Output Port

The built-in keyboard output port bits 0 through 7 are assigned to Port 2 (pins 21 through P24 and 35 through 38, respectively) on the keyboard controller. The output port bit definitions are given in the following list.

Output Port 1

Pin	Bit	Desig	Definition
38	7	P27	External Data (KDATA) Enable: 0 = OFF, 1 = ON
37	6	P26	External Clock (KCLK) Enable: 0 = ON, 1 = OFF
36	5	P25	Input Buffer Full (Not Used)
35	4	P24	Keyboard Interrupt IRQ1 (via OBF)
24	3	P23	Control Sense 1
23	2	P22	Control Sense 0
22	1	P21	Keyboard Controller Gate Address (A20GT)
21	0	P20	System Reset

External Keyboard or Keypad

The external keyboard or keypad is connected to the keyboard controller through its Test Input pins as follows:

Pin 1	TEST0 (T0)	External Keyboard Clock (KCLK)
Pin 39	TEST1 (T1)	External Keyboard Data (KDATA)

Keyboard Commands/Responses

Keyboard commands are sent from the microprocessor to the keyboard controller and keyboard responses are sent from the keyboard controller to the microprocessor. If an external keyboard is connected, the keyboard commands are passed through the controller to the keyboard and responses are passed through the controller to the microprocessor. The keyboard commands and responses are sent as hexadecimal (h) values via Input/Output Buffers (60h). All keyboard commands can be issued at any time, but the keyboard controller must respond to the commands that it receives within 20 milliseconds. Table 8-3 lists the keyboard commands and responses. Commands received by the keyboard controller are indicated by a letter "K" and responses received by the microprocessor are indicated by a letter "M." A brief description of each keyboard command/response is provided following the table.

Table 8-3. Keyboard Commands (K) and Responses (M)

Command or Response Name	Receiving Device M/K	Value(h)
Keyboard Overrun	M	00/FF (Note 1)
Enhanced Keyboard ID	M	xxAB
Basic Assurance Test (BAT)	M	AA/FC
Set or Reset Status Indicators	K	ED
Echo	K/M	EE
No Operation/Invalid Command	K	EF (Note 2)
Break Code Prefix/Select Alternate Scan Codes	K	F0 (Note 2)
No Operation/Invalid Command	K	F1 (Note 2)
No Operation/Read Keyboard ID	K	F2 (Note 2)
Set Typematic Rate and Delay	K	F3
Enable Default Values	K	F4
Default Disable	K	F5
Set Default	K	F6
No Operation/Set All Keys	K	F7-FA (Note 2)
Typematic	K	F7
Make or Break	K	F8
Make	K	F9
Typematic or Make or Break	K	FA
Acknowledge (ACK)	M	FA
No Operation/Set Key Type	K	FB-FD (Note 2)
Typematic	K	FB
Make or Break	K	FC
Make	K	FD
Resend	K/M	FE
Reset	K	FF

NOTE:

1. The value 00h is used for the keyboard scan code set and the value FFh is used for system scan code set (refer to Table 8-5).
2. Commands xxAB, EF, F1, F2, and F7 through FD are not supported by the built-in keyboard or the external 84-key keyboard. The virgule (/) in the Command or Response name separates the name used for the 84-key keyboard from the name used for the enhanced (101/102-key) keyboard.

Keyboard Overrun (00h or FFh) Response

If the keyboard buffer capacity is exceeded, an overrun character replaces the last character in the buffer. When the overrun character reaches the top of the buffer queue, an error code is generated.

When the keyboard is using the system scan code set, the error code sent to the microprocessor is FFh. If the keyboard scan code set is being used, the error code is 00h. For all keyboards except the enhanced keyboard, the scan code set is selected by bit 6 of the keyboard controller command byte. The enhanced keyboard scan code set is selected by keyboard command F0h. Refer to the following paragraphs on Key Definitions.

Enhanced Keyboard ID (xxABh) Response

For the enhanced keyboard only, an identification (ID) response consisting of two bytes, xx and ABh, is used to identify the keyboard. The xx represents any of several codes depending upon which enhanced keyboard is connected. When an enhanced keyboard receives a Read Keyboard ID command (F2h), it responds by returning an Acknowledge (ACK) response (FAh) and the two keyboard ID bytes (xxABh) to the microprocessor. The low byte of the keyboard ID is returned first and then followed by the high byte within 500 microseconds.

Basic Assurance Test Complete or Failure (AAh/FCh) Response

The keyboard subsystem generates a Power-ON Reset (POR) when power is first applied. The POR occurs between 150 milliseconds (ms) and 2.0 seconds after the power is applied. A Basic Assurance Test (BAT) is then performed, and if satisfactorily completed, a completion command (AAh) is sent to the microprocessor. The microprocessor receives the completion command between 450 ms and 2.5 seconds after POR. The BAT completion command is also sent 300 to 500 ms after a keyboard controller Self-Test command (AAh) is acknowledged.

The BAT consists of checking operation of the keyboard controller processor, ROM, and RAM. While the BAT is performed, the activity of the built-in keyboard and on the serial clock and data lines is ignored. The test requires between 300 and 500 milliseconds to complete. When BAT is satisfactorily completed, an AAh response is sent to the microprocessor. If the BAT fails, FCh or any response other than AAh is returned to the microprocessor, the keyboard controller stops scanning, and then waits for another command or

reset from the microprocessor. Immediately following POR, the keyboard controller begins to monitor the signals from the built-in keyboard and from the serial clock and data lines, and sets the line protocol.

Set or Reset Status Indicators (EDh) Command

An external keyboard may have LED indicators associated with the NmLk, CpLk, and ScrLk keys. These LEDs are accessible through the Set/Reset Status Indicators command (EDh). When an EDh command is sent to the keyboard controller, it responds by returning an Acknowledge (ACK) response, stops scanning the keyboard, and waits for the following option byte from the microprocessor. The associated indicator is turned ON if the bit is set to "1," and is turned off if the bit is reset to "0."

Bit	Indicator
0	Scroll Lock (ScrLk)
1	Number Lock (NmLk)
2	Caps Lock (CpLk)
3-7	Reserved (must be 0)

The keyboard controller also responds to the option byte by returning an Acknowledge (ACK) response and, if the keyboard was previously enabled, resumes scanning the keyboard. If another command is received instead of the option byte, the status indicators do not change states, and the new command is processed.

NOTE: The built-in keyboard uses the EDh command to keep track of the NmLk status and to support the internal numeric keypad (red numbers). The keyboard controller will not enable the internal numeric keypad if it detects that a GRiD Systems Model 34160 External Numeric Keypad is connected.

Echo (EEh) Command

The Echo command (EEh) is used for diagnostic tests. When command EEh is received by the keyboard controller, it returns the same value (EEh) as a response. If the keyboard controller was previously enabled, the controller resumes scanning the keyboard.

Invalid Commands (EFh and F1h)

For the enhanced keyboard, commands EFh and F1h are invalid commands. If one of these commands is sent, the keyboard acknowledges and continues its previous scanning state. No other activities occur.

Break Code Prefix (F0h) Response

For the keyboard scan code set, the code F0h is sent as the first byte of a two byte sequence to indicate that a key was released.

Select Alternate Scan Codes (F0h) Command

The Select Alternate Scan Codes (F0h) command is used to change the scan code set for the enhanced keyboard only. To select the alternate scan code set, an F0h command is sent from the microprocessor to the keyboard controller. The keyboard controller responds by returning an Acknowledge (ACK) response, clears the keyboard buffer, cancels any typematic key operation, and then waits for the following option byte.

Byte Value	Selected Scan Code Set
00h	Return value indicating current scan code set.
01h	Select System Scan Code Set
02h	Select Keyboard Scan Code Set
03h	Select Scan Code Set 3

The scan code sets are defined under key definitions in subsequent paragraphs. After completing the command, the keyboard controller resumes scanning the keyboard.

Read Keyboard ID (F2h) Command

For the enhanced keyboard only, this command is used to request identification (ID) information. The keyboard controller responds by returning an Acknowledge (ACK) response, stops scanning, and sends a two-byte keyboard ID (xxABh) to the microprocessor. After the keyboard ID is sent, the keyboard controller then resumes scanning.

Set Typematic Rate and Delay (F3h) Command

Typematic defines the automatic repeat operation of the keys when they are held down. All of the built-in keyboard key operations are typematic. For external keyboard, all of the key operations are typematic except Ctrl, Shift, Alt, NmLk, ScrLk, CpLk, and Ins. The F3h command allows adjustment of the typematic repeat rate and delay time before the repeats begin.

The keyboard receives the F3h command and responds by returning an Acknowledge (ACK) response, stops scanning, and waits for a value byte. When the value byte is received, the keyboard responds by

returning another ACK response, sets the rate and delay to the indicated values, and continues scanning, if scanning was previously enabled. If another command is received before the value byte, the F3h command is ignored.

For the built-in keyboard, typematic operations are handled by the keyboard emulation operations of the keyboard controller. The same operations are handled by an external keyboard through the controller input and output buffers (60h). Therefore, for external keyboards, the controller waits for the second ACK response and then sends only one ACK to the microprocessor.

The F3h command format is as follows:

Bits	Use
4-0	Typematic Rate (0 is least significant bit)
6, 5	Typematic Delay
7	Always 0 (most significant bit)

The delay before repeat operations begin is determined by the binary value of bits 6 and 5, plus 1, and multiplied by 250 milliseconds +/-20%.

The typematic rate is the frequency that a depressed keystroke is repeated. The frequency is one per period with the period determined by the following equation.

$$\text{Period} = (8 + A) \times 2^{B} \times 0.00417 \text{ seconds}$$

Where: A = binary value of bits 2-0,
B = binary value of bits 4, 3.

The typematic rates selected by bits 4 through 0 are listed in Table 8-4. The default value for the built-in keyboard typematic rate is 15 (+/- 2) characters per second. For external keyboards the typical default value for typematic rate is 10 (+/- 2) characters per second. The default value for the delay before repeat operations begin is 500 +/-100 milliseconds.

Enable Default (F4h) Command

When the keyboard controller receives an Enable Default (F4h) command, it responds by returning an Acknowledge (ACK) response, clearing the keyboard buffer, canceling any typematic operation, and then begins scanning the keyboard for scan codes.

Table 8-4. Typematic Rate Table

Bit 43210	Rate	Bit 43210	Rate
00000	30.0	10000	7.50
00001	26.7	10001	6.70
00010	24.0	10010	6.00
00011	21.8	10011	5.50
00100	20.0	10100	5.00
00101	18.5	10101	4.60
00110	17.1	10110	4.30
00111	16.0	10111	4.00
01000	15.0	11000	3.70
01001	13.3	11001	3.30
01010	12.0	11010	3.00
01011	10.9	11011	2.70
01100	10.0	11100	2.50
01101	9.2	11101	2.30
01110	8.0	11110	2.10
01111	8.0	11111	2.00

NOTE: The rate is given in the number of repeats per second with a tolerance of 20 percent.

Default Disable (F5h) Command

The Default Disable (F5h) command resets all keyboard conditions to the Power-ON default state. When the keyboard controller receives an F5h command, it responds by returning an Acknowledge (ACK) response, clears the keyboard buffer and cancels any typematic operation, sets the typematic rate/delay to the default values, stops scanning the keyboard and waits for the next command.

Set Default (F6h) Command

When the Set Default (F6h) command is received, the controller returns an Acknowledge (ACK) response and then resets all keyboard subsystem conditions to the Power-ON default state. It then clears the keyboard buffer and cancels any typematic operation, sets the typematic rate/delay to the default values, and continues scanning the keyboard.

Set All Keys (F7h-FAh) Command

These commands only affect an enhanced keyboard when scan code set 3 is selected (refer to command F0h). Scan code set 3 allows the type of operation for all keys to be changed by these commands as follows:

Command	Type of Operation
F7h	Set All Keys to Typematic
F8h	Set All Keys to Make or Break
F9h	Set All Keys to Make
FAh	Set All Keys to Typematic or Make or Break

When one of these commands is sent, the keyboard returns an Acknowledge (ACK) response, clears the Output Buffer (60h), sets all keys to the type indicated by the command, and continues its previous scanning state. These commands are ignored by all keyboards except an enhanced keyboard.

Acknowledge (FAh) Response

The Acknowledge (FAh) response is the first response from the keyboard controller after it has received a command from the microprocessor. The FAh (ACK) response is sent after every valid command except Echo (EEh) and Resend (FEh). If another command is received while ACK is being sent, ACK is disregarded and the new command is serviced.

Set Key Type (FBh, FCh, and FDh)

These commands only affect an enhanced keyboard when scan code set 3 is selected (refer to command F0h). Scan code set 3 allows the type of operation for each key to be changed by these commands as follows:

Command	Type of Operation
FBh	Set Key Type to Typematic
FCh	Set Key Type to Make or Break
FDh	Set Key Type to Make

When one of these commands is sent, the keyboard returns an Acknowledge (ACK) response, clears the Output Buffer (60h), and prepares to receive the key identification byte. The key identification byte is the key scan code selected from scan code set 3 (refer to Table 8-6). After receiving the key identification byte, the key is set to the type indicated by the command, and the keyboard continues its previous scanning state. These commands are ignored by all keyboards except an enhanced keyboard.

Diagnostic Failure (FDh) Response

This command only affects the 84-key keyboard. The keyboard periodically tests the sense amplifier and sends a diagnostic failure code (FDh) if it detects an error. If the failure occurs during BAT, the keyboard stops scanning and waits for a system command or a power turn-OFF to restart. When a failure is reported after scanning is enabled, scanning continues.

Resend (FEh) Command

The Resend (FEh) command is generated by the microprocessor if an error is detected in any data received from the keyboard. The command is sent to the keyboard before the next data is allowed to be sent. When the keyboard first receives the Resend command, it resends the previous output. If the Resend command has been received before, the keyboard resends the byte that was sent before it received the first Resend command.

Reset (FFh) Command

The microprocessor sends a Reset (FFh) command to initiate a keyboard subsystem reset and self-test. The keyboard responds to the Reset command by returning an Acknowledge (ACK) response and then ensures that the microprocessor has accepted the ACK before executing the Reset command. The microprocessor responds to the ACK by raising the clock and data lines for at least 500 microseconds. The keyboard subsystem is effectively disabled from the time it receives the Reset command until the ACK response is accepted, or until another command is received that overrides the Reset command.

KEY DEFINITIONS

For the built-in keyboard, the keyboard controller scans and senses the keystrokes and generates scan codes whenever a key is depressed and also whenever the key is released. A "make" code is generated when a key is depressed and a "break" code is generated when a key is released. The scan codes are subsequently translated into the appropriate ASCII codes by the ROM-BIOS. The translation process allows software to define keyboard operations.

An externally connected keyboard or keypad also generates scan codes whenever a key is depressed or released. These external scan codes are sent to the keyboard controller in the form of 8-bit serial data. The keyboard controller converts the external scan codes from serial to parallel format. The scan codes are subsequently translated into the appropriate ASCII codes by the ROM-BIOS.

The scan codes generated by the make and break operation of each key are determined by bit 6 of the keyboard controller command byte as follows (refer to Table 8-2):

Bit 6	Keyboard Mode	Scan Code Set
0	Not IBM PC Compatible	Keyboard Scan Codes
1	IBM PC Compatible	System Scan Codes

When the system scan code set is selected, the keyboard controller translates the keystrokes from the built-in keyboard, or scan codes from the external keyboard, into the IBM PC compatible system scan code set. For the system scan code set, the break scan code is obtained by adding 80h to the make scan code. Most make and break scan codes require one 8-bit byte. To provide different scan codes for duplicate operations on enhanced keyboards, the scan code for the duplicate key is prefixed with extended code E0h.

If the keyboard scan code set is selected, no translation takes place at the keyboard controller level and the non-IBM AT compatible scan codes are sent to the ROM-BIOS for translation. For the keyboard scan code set, the break scan code is obtained by adding an F0h prefix to the make scan code, so that every break scan code is at least two bytes wide. Refer to Table 8-5.

An enhanced keyboard supports three sets of scan codes. The three scan codes sets are selectable via the keyboard command F0h (refer to Table 8-3). Two of the enhanced keyboard scan code sets are the system scan codes and keyboard scan codes previously described. The scan code set 3 is similar to the keyboard scan code set except that each key sends only one scan code and no keys are affected by the state of other keys. Also, scan code set 3 allows use of the keyboard commands to Set All Keys (F7h-FAh) and Set Key Type (FBh, FCh, and FDh). Refer to Table 8-6.

The ROM-BIOS initiates the keyboard to the IBM PC compatible mode, so that after power turn-ON, the system scan code set is the default set for all keyboard configurations including the enhanced keyboard. The system scan code set remains selected for the enhanced keyboard unless a different scan code set is specifically selected via keyboard command F0h.

All of the built-in keyboard keys and most of the external keyboard keys are typematic. That is, their operation repeats as long as the key is held down. While a key is repeating, multiple make codes are generated, but only one break code is generated when the key is released.

The GRiDCase 1500 Series computer 72-key keyboard emulates an IBM AT 84-key keyboard. In most cases, the emulation is provided by depressing similarly marked keys. Where the GRiDCase 1500 Series computer does not have a similarly marked key, combinations of keys

are depressed to generate the same scan code as the IBM AT keys. Figure 8-2 shows the GRiDCase 1500 Series computer keyboard layout and the labels on each key. Table 8-5 lists the keys on the IBM AT keyboard, the equivalent keys on the GRiDCase 1500 Series computer keyboard, and the scan codes generated by system scan code set and the keyboard scan code set when the key(s) is depressed (make) and released (break). All of the scan code values are given in hexadecimal.

In the GRiDCase column of Table 8-5, a plus sign (+) between two characters indicates that keys listed on both sides of the plus sign are depressed to generate the required scan code. The key to the left of the plus sign is depressed and held while the key to the right of the plus sign is being depressed and released.

In Table 8-5, additional information such as the key location, color of marking on the GRiDCase keyboard if other than white, and multiple keystrokes are also listed. The key location, such as left or right, is listed in braces. The color of key marking if other than white is shown in parentheses. Where multiple keystrokes are required to generate the scan code, such as Fn + S to generate ScrLk, the multiple keystrokes are given in brackets. The color markings on the GRiDCase keyboard are as follows:

1. Standard keyboard letters and numbers, function keys F1 through F10, punctuation symbols, and the Ins and Del keys are marked with white lettering.

2. Standard keyboard control keys such as Tab, Return, Shift, and BkSp are marked in yellow (Yel).

3. Keys used to emulate IBM AT compatible keyboard functions such as Pause, ScrLk, NmLk, and PgDn are marked in blue (Blu). These keys are enabled by first depressing and holding the Fn key, which is also marked in blue, and then depressing and releasing the key for the desired function.

4. The imbedded keypad keys are marked in red (Red). These keys are enabled by the Number Lock function. Number Lock (NmLk) is an ON/OFF toggle function. When ON, it allows the use of the internal keypad (red numbers) for data entry. This operation is handled in the keyboard controller and can be overridden by the MODE NUMPAD=ON/OFF command.

The IBM keypad is defined as the rightmost block of 17 or 18 keys on the IBM AT compatible keyboards. The IBM keypad keys used for cursor movement are contained within the keypad. On enhanced keyboards, the same cursor movement keys are repeated in an adjacent keyboard location, but with scan codes prefixed with E0h. The GRiDCase computer keyboard does not support the extra set of cursor movement keys and scan codes.

Also, the GRiDCase keyboard supports one each of the Ctrl, Alt, and Enter (Return) keys where the IBM AT compatible keyboard supports two each. In addition, the GRiDCase keyboard supports an "Fn" key that is used for mapping the GRiDCase keyboard to the IBM AT compatible keyboard. The "Fn" key, when depressed by itself, does not generate a scan code.

Since most shift operations are handled in the ROM-BIOS, only the unshifted operation of each key is listed in Table 8-5.

NOTE: In the keyboard scan code set given in Table 8-5, the * symbol is used to indicate the "make" scan codes that are not translatable via keyboard controller command 0Ah. Refer to the previous description of Key Translation.

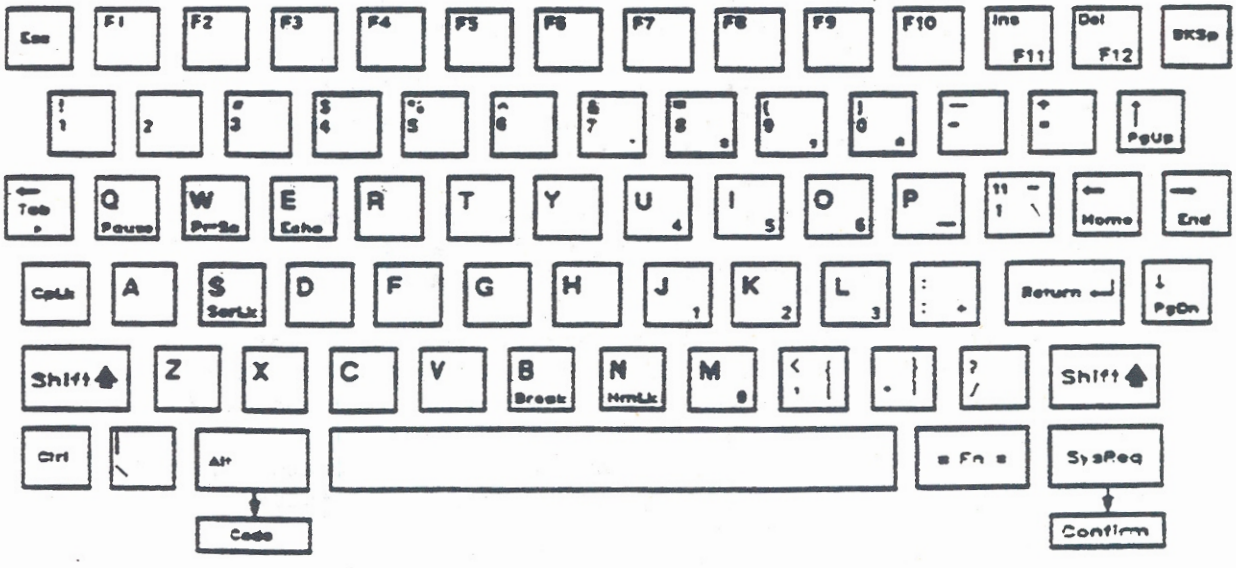


Figure 8-2. GRiDCase 1500 Series Computer Keyboard Layout

Table 8-5. Scan Codes

IBM Key(s)	GRid Key(s)	System Scan Codes Make	System Scan Codes Break	Keyboard Scan Codes Make	Keyboard Scan Codes Break
Esc	Esc (Yel)	01	81	76	F0 76
1	1	02	82	16	F0 16
2	2	03	83	1E	F0 1E
3	3	04	84	26	F0 26
4	4	05	85	25	F0 25
5	5	06	86	2E	F0 2E
6	6	07	87	36	F0 36
7	7	08	88	3D	F0 3D
8	8	09	89	3E	F0 3E
9	9	0A	8A	46	F0 46
0	0	0B	8B	45	F0 45
-	-	0C	8C	4E	F0 4E
=	=	0D	8D	55	F0 55
Back-space	BkSp (Yel)	0E	8E	66	F0 66
Tab	Tab (Yel)	0F	8F	0D	F0 0D
Q	Q	10	90	15	F0 15
W	W	11	91	1D	F0 1D
E	E	12	92	24	F0 24
R	R	13	93	2D	F0 2D
T	T	14	94	2C	F0 2C
Y	Y	15	95	35	F0 35
U	U	16	96	3C#	F0 3C
I	I	17	97	43#	F0 43
O	O	18	98	44#	F0 44
P	P	19	99	4D	F0 4D
[	[(Blu) {Fn + ,}	1A	9A	54	F0 54
]	] (Blu) {Fn + .}	1B	9B	5B	F0 5B
Enter	Return(Yel)	1C	9C	5A	F0 5A
Ctrl [left]	Ctrl (Yel)	1D	9D	14	F0 14
Ctrl-PrtSc	Echo (Blu) {Fn + e}	1D 37	9D B7	14 7C#	F0 14 F0 7C
Pause	Pause (Blu) {Fn + q}	1D 45	9D C5	14 77#	F0 14 F0 77
Ctrl-ScrLk	Break (Blu) {Fn + b}	1D 46	9D C6	14 7E#	F0 14 F0 7E
A	A	1E	9E	1C	F0 1C
S	S	1F	9F	1B	F0 1B
D	D	20	A0	23	F0 23

Table 8-5. Scan Codes (Continued)

IBM Key(s)	GRid Key(s)	System Scan Codes Make	System Scan Codes Break	Keyboard Scan Codes Make	Keyboard Scan Codes Break
F	F	21	A1	2B	F0 2B
G	G	22	A2	34	F0 34
H	H	23	A3	33	F0 33
J	J	24	A4	3B	F0 3B
K	K	25	A5	42	F0 42
L	L	26	A6	4B	F0 4B
;	;	27	A7	4C	F0 4C
'	'	28	A8	52	F0 52
`	` (Blu) {Fn + '}	29	A9	0E	F0 0E
Shift [left]	Shift (Yel) [left]	2A	AA	12	F0 12
\	\	2B	AB	5D	F0 5D
Z	Z	2C	AC	1A	F0 1A
X	X	2D	AD	22	F0 22
C	C	2E	AE	21	F0 21
V	V	2F	AF	2A	F0 2A
B	B	30	B0	32	F0 32
N	N	31	B1	31	F0 31
M	M	32	B2	3A	F0 3A
,	,	33	B3	41	F0 41
.	.	34	B4	49	F0 49
/	/	35	B5	4A	F0 4A
Shift [right]	Shift (Yel) [right]	36	B6	59#	F0 59
Sys Req (Note)	SysReq(Yel) /Confm(Red)	54	D4	7F	F0 7F
Print-Screen (Note)	PrtSc (Blu) {Fn + w}	36 37	B6 B7	59 7C#	F0 59 F0 7C
* [Keypad]	* (Blu) {Fn + 0}	37	B7	7C	F0 7C
Alt	Alt (Blu) /Code (Red)	38	B8	11#	F0 11
space	space	39	B9	29	F0 29
CapsLock	CpLk (Yel)	3A	BA	58	F0 58

NOTE: Sys Rq is generated by Alt-PrtSc and PrintScreen is generated by Shift-* [keypad] on some IBM compatible keyboards.

Table 8-5. Scan Codes (Continued)

IBM Key(s)	GRid Key(s)	System Scan Codes Make	System Scan Codes Break	Keyboard Scan Codes Make	Keyboard Scan Codes Break
F1	F1	3B	BB	05	F0 05
F2	F2	3C	BC	06	F0 06
F3	F3	3D	BD	04	F0 04
F4	F4	3E	BE	0C	F0 0C
F5	F5	3F	BF	03	F0 03
F6	F6	40	C0	0B	F0 0B
F7	F7	41	C1	83 or 2	F0 83 or F0 02
F8	F8	42	C2	0A	F0 0A
F9	F9	43	C3	01	F0 01
F10	F10	44	C4	09	F0 09
Num Lock [Keypad]	NmLk (Blu) (Fn + n)	45	C5	77	F0 77
Scroll-Lock	ScrLk (Blu) (Fn + s)	46	C6	7E	F0 7E
Home [Keypad]	Home (Blu) (Fn + left cursor key)	47	C7	6C	F0 6C
Upcursor [Keypad]	Upcursor (Yel)	48	C8	75	F0 75
Pg Up [Keypad]	PgUp (Blu) (Fn + up cursor key)	49	C9	7D	F0 7D
- [Keypad]	- (Fn + p)	4A	CA	7B	F0 7B
Left-cursor [Keypad]	Left-cursor(Yel)	4B	CB	6B	F0 6B
5 [Keypad]	(Fn + Retn)	4C	CC	73	F0 73
Right-cursor	Right-cursor(Yel)	4D	CD	74	F0 74
+ [Keypad]	(Fn + =)	4E	CE	79	F0 79
End [Keypad]	End (Blu) (Fn + right cursor key)	4F	CF	69	F0 69

Table 8-5. Scan Codes (Continued)

IBM Key(s)	GRid Key(s)	System Scan Codes Make	System Scan Codes Break	Keyboard Scan Codes Make	Keyboard Scan Codes Break
Down-cursor [Keypad]	Down-cursor(Yel)	50	D0	72	F0 72
PgDn [Keypad]	PgDn (Blu) (Fn + Down-cursor)	51	D1	7A	F0 7A
Ins [Keypad]	Ins	52	D2	70	F0 70
Delete [Keypad]	Del	53	D3	71	F0 71
F11	F11 (Blu) (Fn + Ins)	57	D7	78	F0 78
F12	F12 (Blu) (Fn + Del)	58	D8	07	F0 07
NOTE:	The following keys are the GRiDCase keyboard imbedded keypad (red keys) when the NmLk function is OFF. When NmLk function is ON, 36h and B6h are not prefixed on the scan code.				
	7 (Fn + 7)	36 47	B6 C7	59 6C#	F059 F06C
	8 (Fn + 8)	36 48	B6 C8	59 75#	F059 F075
	9 (Fn + 9)	36 49	B6 C9	59 7D#	F059 F07D
	4 (Fn + U)	36 4B	B6 CB	59 6B#	F059 F06B
	5 (Fn + I)	36 4C	B6 CC	59 73#	F059 F073
	6 (Fn + O)	36 4D	B6 CD	59 74#	F059 F074
	1 (Fn + J	36 4F	B6 CF	59 69#	F059 F069
	2 (Fn + K)	36 50	B6 D0	59 72#	F059 F072
	3 (Fn + L	36 51	B6 D1	59 7A#	F059 F07A
	0 (Fn + M	36 52	B6 D2	59 70#	F059 F070

Table 8-6. Scan Code Set 3 (Enhanced Keyboard Only)

IBM Key(s)	Scan Codes Make	Break	Default Key State
Esc	08	F0 08	Make only
1	16	F0 16	Typematic
2	1E	F0 1E	Typematic
3	26	F0 26	Typematic
4	25	F0 25	Typematic
5	2E	F0 2E	Typematic
6	36	F0 36	Typematic
7	3D	F0 3D	Typematic
8	3E	F0 3E	Typematic
9	46	F0 46	Typematic
0	45	F0 45	Typematic
-	4E	F0 4E	Typematic
=	55	F0 55	Typematic
Back-space	66	F0 66	Typematic
Tab	0D	F0 0D	Typematic
Q	15	F0 15	Typematic
W	1D	F0 1D	Typematic
E	24	F0 24	Typematic
R	2D	F0 2D	Typematic
T	2C	F0 2C	Typematic
Y	35	F0 35	Typematic
U	3C	F0 3C	Typematic
I	43	F0 43	Typematic
O	44	F0 44	Typematic
P	4D	F0 4D	Typematic
[	54	F0 54	Typematic
]	5B	F0 5B	Typematic
Enter	5A	F0 5A	Typematic
Ctrl [left]	11	F0 11	Make/Break
Break	62	F0 62	Make only
A	1C	F0 1C	Typematic
S	1B	F0 1B	Typematic
D	23	F0 23	Typematic
F	2B	F0 2B	Typematic
G	34	F0 34	Typematic
H	33	F0 33	Typematic
J	3B	F0 3B	Typematic
K	42	F0 42	Typematic
L	4B	F0 4B	Typematic

Table 8-6. Scan Code Set 3 (Enhanced Keyboard Only) Continued

IBM Key(s)	Keyboard Scan Codes Make	Break	Default Key State
;	4C	F0 4C	Typematic
'	52	F0 52	Typematic
`	0E	F0 0E	Typematic
Shift [left]	12	F0 12	Make/Break
\	5C	F0 5C	Typematic
Z	1A	F0 1A	Typematic
X	22	F0 22	Typematic
C	21	F0 21	Typematic
V	2A	F0 2A	Typematic
B	32	F0 32	Typematic
N	31	F0 31	Typematic
M	3A	F0 3A	Typematic
,	41	F0 41	Typematic
.	49	F0 49	Typematic
/	4A	F0 4A	Typematic
Shift [right]	59	F0 59	Make/Break
Sys Req	57	F0 57	Make only
* [Keypad]	7E	F0 7E	Make only
Alt [left]	19	F0 19	Make/Break
Alt [right]	39	F0 39	Make only
Space	29	F0 29	Typematic
CapsLock	14	F0 14	Make/Break
F1	07	F0 07	Make only
F2	0F	F0 0F	Make only
F3	17	F0 17	Make only
F4	1F	F0 1F	Make only
F5	27	F0 27	Make only
F6	2F	F0 2F	Make only
F7	37	F0 37	Make only
F8	3F	F0 3F	Make only
F9	47	F0 47	Make only
F10	4F	F0 4F	Make only
Num Lock [Keypad]	76	F0 76	Make only
Scroll-Lock	5F	F0 5F	Make only
Home [Keypad]	6C	F0 6C	Make only
Upcursor [Keypad]	75	F0 75	Make only

Table 8-6. Scan Code Set 3 (Enhanced Keyboard Only) Continued

IBM Key(s)	Keyboard Scan Codes: Make	Keyboard Scan Codes: Break	Default Key State
Pg Up [Keypad]	7D	F0 7D	Make only
- [Keypad]	84	F0 84	Make only
Left-cursor [Keypad]	6B	F0 6B	Make only
5 [Keypad]	73	F0 73	Make only
Right-cursor	74	F0 74	Make only
+ [Keypad]	7C	F0 7C	Typematic
End [Keypad]	69	F0 69	Make only
Down-cursor [Keypad]	72	F0 72	Make only
PgDn [Keypad]	7A	F0 7A	Make only
Ins [Keypad]	70	F0 70	Make only
Delete [Keypad]	71	F0 71	Make only
F11	56	F0 56	Make only
F12	5E	F0 5E	Make only
NOTE:	The following scan codes are for duplicate function keys located on the extra keypad of an enhanced keyboard.		
Insert	75	F0 75	Make only
Delete	76	F0 76	Typematic
Left-cursor	61	F0 61	Typematic
Home	6E	F0 6E	Make only
End	65	F0 65	Make only
Up-cursor	63	F0 63	Typematic
Down-cursor	60	F0 60	Typematic
Page Up	6F	F0 6F	Make only
Page-Down	6D	F0 6D	Make only
Right-cursor	6A	F0 6A	Typematic
/	77	F0 77	Make only
Enter	79	F) 79	Make only

EXTERNAL KEYBOARD CONNECTOR

An external keyboard connector is provided on the back panel of the GRiDCase 1500 Series computer. The keyboard connector is a 5-pin DIN type receptacle that is commonly used for keyboard interfaces. Figure 8-3 shows the connector pin layout and the connector pin definitions are provided in Table 8-7.

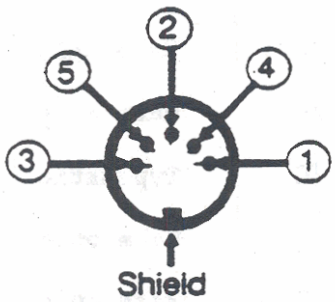


Figure 8-3. External Keyboard Connector Pin Layout

Table 8-7. External Keyboard Connector Pin Definitions

Pin No.	Description
1	KCLK (Keyboard Clock)
2	KDATA (Keyboard Data)
3	Not Used
4	Ground
5	+5V dc
(keyway)	Shield

CHAPTER 9: FLOPPY DISK DRIVE SUBSYSTEM

The GRiDCase 1500 Series computer Floppy Disk Drive Subsystem supports two internal 1.4M byte, 3.5-inch floppy disk drives and an external peripheral port for connecting an external drive (see Figure 9-1). One or both of the internal floppy disk drives can be replaced by an optional hard disk drive with storage capacities from 10M bytes to 100M bytes. The external peripheral port supports one external device, which can be either a 3.5-inch or 5.25-inch floppy disk drive or a 40M byte backup tape unit. The standard and optional disk drive configurations and formatted capacities are listed in Table 9-1.

This chapter (9) describes operation of Floppy Disk Drive Subsystem and PC XT compatible Hard Disk Drive configurations listed as items 1 through 3 in Table 9-1. Chapter 10 describes operation of the Hard Disk Drive Subsystem containing IBM PC AT compatible hard disk drives, which are listed in Table 9-1 as items 4 through 6. The external peripheral port connector is described at the end of this chapter. Operations of the external peripherals (items 7 through 10) are described in separate manuals, which are supplied with the listed devices.

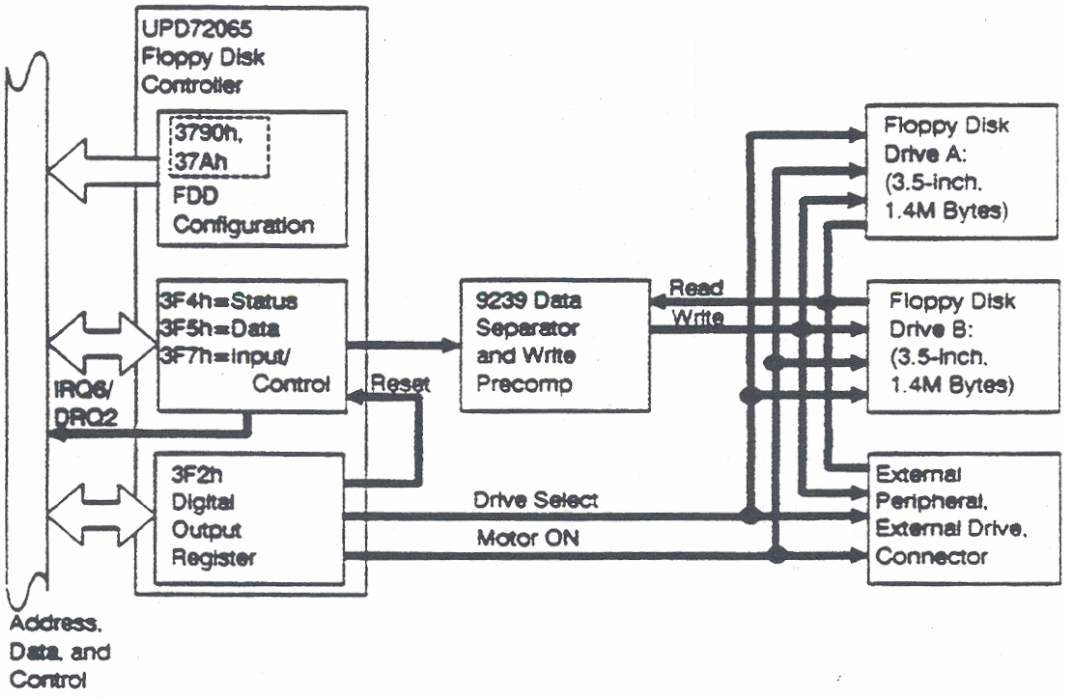


Figure 9-1. Floppy Disk Drive Subsystem Block Diagram

Table 9-1. Disk Drive Configurations

Item	Option	Qty	Configuration	Capacity
1	Standard	2	3.5-inch Floppy Disk Drives (IBM-PC AT Compatible)	1.4M bytes each 2.8M bytes total
2	320	1 1	3.5-inch Floppy Disk Drive 10M byte Hard Disk Drive (Not IBM-PC AT Compatible)	1.4M bytes 10M bytes 11.4M bytes total
3	321	1 1	3.5-inch Floppy Disk Drive 20M byte Hard Disk Drive (Not IBM-PC AT Compatible)	1.4M bytes 20M bytes 21.4 bytes total
4	321	1 1	3.5-inch Floppy Disk Drive 20M byte Hard Disk Drive (IBM-PC AT Compatible)	1.4M bytes 20M bytes 21.4 bytes total
5	324	1	40M byte Hard Disk Drive (IBM-PC AT Compatible)	40M bytes total
6	325	1	100M byte Hard Disk Drive (IBM-PC AT Compatible)	100M bytes total
7	3401	1	External 3.5-inch Floppy Disk Drive (1.4M Byte Pocket Floppy)	Adds 1.4M bytes to any drive configuration
8	3402	1	External 5.25-inch Floppy Disk Drive (360k Byte Pouch Floppy)	Adds 360k bytes to any drive configuration
9	3403	1	External 40M Byte Backup Tape System	Adds disk backup capability
10	3404	1	External 5.25-inch Floppy Disk Drive (1.2M Byte Pouch Floppy)	Adds 1.2M bytes to any drive configuration

This chapter is divided into three major sections as follows:

1. Floppy Disk Drive (Standard)

2. IBM XT Compatible Hard Disk Drives (Options 320 and 321)

3. Floppy Disk Drive Expansion Port

The IBM PC AT compatible Hard Disk Drive Subsystem is described in Chapter 10.

FLOPPY DISK DRIVE

The 3.5-inch floppy disk drive used in the GRiDCase 1500 Series computer is double-sided and double-density with a formatted capacity of 1.4 megabytes. The disk is data compatible with the 5.25-inch, 9 sector, double-sided, double-density (48 tracks-per-inch) disk drives.

Floppy Disk Software Interface

The program interface to the internal floppy disk drives is through a standard set of ROM-BIOS calls. These calls mask the differences between the internal 3.5-inch disk drives and an external 5.25-inch disk drive so that both types of drives are supported by the same controller. The disk drive software also provides a device driver interface to support a hard disk drive for GRiD Systems software compatibility.

The floppy disk controller software is functionally identical to that used on the IBM PC AT and the controller interface software is the same as the IBM PC AT at the ROM-BIOS level. To directly access the floppy disk controller, refer to the I/O register description later in this chapter.

Floppy Disk Drive Service Routine

The ROM-BIOS supports the following disk-related functions in its interrupt service routine:

Interrupt (Hex)	Function No. (AH reg)	Description
13	00	Reset Disk Drive Controller
13	01	Read Disk Drive Status from Last Operation
13	02	Read Data from Desired Sectors Into Memory
13	03	Write Data from Memory to Desired Sectors
13	04	Verify that Disk Sectors are Valid
13	05	Format Desired Track on Disk
13	06	Unused
13	07	Unused
13	08	Read Current Disk Drive Parameters
13	09-14	Hard Disk Only
13	15	Read DASD Type
13	16	Disk Change Line Status
13	17	Set DASD Type for Format
13	18	Set Media Type for Format

The following paragraphs summarize the operation of the floppy disk drive functions. The functions that support the hard disk drive operation are described later in this chapter.

Floppy Disk Drive Service Routine Descriptions

The 80286/80386 Microprocessor register inputs and outputs for the disk drive interrupt service routine functions are listed in the following paragraphs. For the read, write, and verify functions, the contents of registers DS, BX, CX, and DX, are preserved, and register AL returns the number of sectors actually read, written, or verified. Register AH determines which function within the service routine is invoked, while other microprocessor registers further define the action to be performed. The register contents are specified in hexadecimal (h). On exit, register AH may contain an Error Code. The Error Codes are listed following the interrupt service routine descriptions.

AH = 00h Reset Floppy Disk Drive Controller

On Exit:
- AH = Error code
- Carry = 0 (no error)
- Carry = 1 (error)

AH = 01h Read Disk Drive Status From Last Operation
- DH = Head number (0-1)
- DL = Drive number (0-2)
- CH = Track number
- CL = Sector number
- AL = Number of Sectors to Read
- ES:BX = Memory address of destination buffer

NOTE: For track and sector values, refer to Table 9-2.

On Exit:
- AH = Status from last disk operation
- Carry = 0 (no error)
- Carry = 1 (error)

AH = 02h Read Data from Desired Sectors into Memory
- DH = Head number (0-1)
- DL = Drive number (0-2)
- CH = Track number
- CL = Sector number
- AL = Number of sectors to Read
- ES:BX = Memory address of destination buffer

On Exit:
- AL = Number of sectors read
- AH = Error code
- Carry = 0 (no error)
- Carry = 1 (error)

AH = 03h Write Data From Memory to Desired Sectors

DH = Head number (0-1)
DL = Drive number (0-2)
CH = Track number
CL = Sector number
AL = Number of sectors
ES:BX = Memory address of source buffer

On Exit:

AL = Number of sectors written
AH = Error code
Carry = 0 (no error)
Carry = 1 (error)

AH = 04h Verify That Disk Sectors Are Valid

DH = Head number (0-1)
DL = Drive number (0-2)
CH = Track number
CL = Sector number
AL = Number of sectors

On Exit:

AL = Number of sectors verified
AH = Error code
Carry = 0 (no error)
Carry = 1 (error)

AH = 05h Format Desired Track on Disk

ES:BX = Points to multiple address fields. One field is required for each sector on the track. Each field has four bytes called C, H, R, and N. The bytes contain the following data:

- C = Cylinder Number
- H = Head Number (0-1)
- R = Sector Number
- N = Number of Bytes per Sector
  - 00h = 128 bytes
  - 01h = 256 bytes
  - 02h = 512 bytes
  - 03h = 1024 bytes

This information is required to find the requested sector during read/write access.

NOTE: When formatting a drive that supports more than one media capacity, the functions called by AH = 17h or AH = 18h must be used to set the desired capacity. If the capacity is not set, the format operation assumes the maximum capability of the drive.

DISK_BASE is a set of parameters required for floppy disk operation. The parameters are pointed to by the double word variable called Disk Pointer at absolute address 00:78h. Two values in the DISK_BASE (GLP and EOT) must be changed to format a specified media as follows:

Media	Drive	GPL	EOT	
360 kB	360 kB/1.2 MB	50h	9	
720 kB	720 kB/1.4 MB	50h	9	
1.2 MB	1.2 MB	54h	15	(default)
1.4 MB	1.4 MB	6Ch	18	

The GPL parameter sets the gap length for format, and the EOT (End of Track) parameter sets the last sector available on the track. When format operations are complete, restore the parameters to their default values as previously listed. To modify the parameters permanently, build another parameter block and point to it instead of to Disk Pointer.

AH = 08h Read Current Disk Drive Parameters

Input Register
DL = Drive Number (0-2)
Output Registers
AX = 0
BH = 0
BL = Valid drive type in CMOS (Bits 3-0, Bits 7-4 = 0)
CH = Low order 8 of 10 bits for maximum number of tracks
CL = High order two bits (7 and 6) for max. number of tracks
DH = Maximum Head Number (0-1)
DL = Number of Floppy Disk Drives Installed
ES:D1 = Points to Drive Parameter table

The following conditions cause changes in the Current Disk Drive Parameters.

1. If the Drive Number is not valid or the Drive Type is not known and CMOS is not present, defective, or invalid then the registers AX, BX, CX, DH, and ES:D1 all contain 0 and register DL contains the number of drives in the system.

2. If no drives are present, registers AX, BX, CX, DX, and ES:D1 all contain 0, the Diskette Status is 0, and the Carry bit is reset.

AH = 15h Read DASD Type
     Output Registers
On Exit if Carry Flag is not set; otherwise, an error
     AH = 00 Drive not present
          01 Floppy Disk, No change line available
          02 Floppy Disk, Change line available
          03 Reserved
     DL = Drive Number (0-2)

AH = 16h Disk Change Line Status
     Output Registers
     AH = 00 Disk Change Line not active
          06 Disk Change Line active and carry bit on.
     DL = Drive Number (0-2)
On Exit:
     AH = Error code
     Carry = 0 (no error)
     Carry = 1 (error)

NOTE: The Disk Change Line Status is checked when the media is specified as other than 360k bytes.

When the Disk Change Line is active, the following actions occur:

1. The program attempts to reset the Disk Change Line to the inactive state.

2. If the attempt is successful, the DASD Type for Format (AH = 17h) is set and the Disk Change Error Code is returned.

3. If the attempt fails, the DASD Type for Format is set to a state meaning media type is unknown and a Timeout Error Code is returned.

If the Disk Change Line is inactive, the Set DASD Type for Format (AH = 17h) operation is performed.

AH = 17h Set DASD Type for Format (AH = 5)
     Input Registers
     AL = 00 Not Used
          01 360 kB Media in 360 kB drive
          02 360 kB Media in 1.2 MB drive
          03 1.2 MB Media in 1.2 MB drive
          04 720 kB Media in 720 kB drive
     DL = Drive Number (0-2)
On Exit:
     AH = Error code
     Carry = 0 (no error)
     Carry = 1 (error)

AH = 18h Set Media Type for Format (AH = 5)
Input Registers
CH = Low order 8 of 10 bits for maximum number of tracks
CL = High order two bits (7 and 6) for max. number of tracks and maximum sectors per track (bits 5-0)
DL = Drive Number (0-2)
Output Registers
AH = 00h and Carry = 0 if track/sector combination is valid
AH = 01h and Carry = 1 if function is not available
AH = 0Ch and Carry = 1 if incorrect track/sector combination
ES:DI = Pointer to the parameter table for the drive that supports the desired media type. The value is unchanged if AH = 0.

Track and Sector Values for Floppy Disk Drive Service Routines

Track and sector values are entered in microprocessor registers AL, CH, and CL for the floppy disk drive service routines. The values to be entered are dependent on the capacity of the media (floppy diskette) and the capability of the floppy disk drive. The values required for the Read, Write, Verify, and Format Operations are listed in Table 9-2.

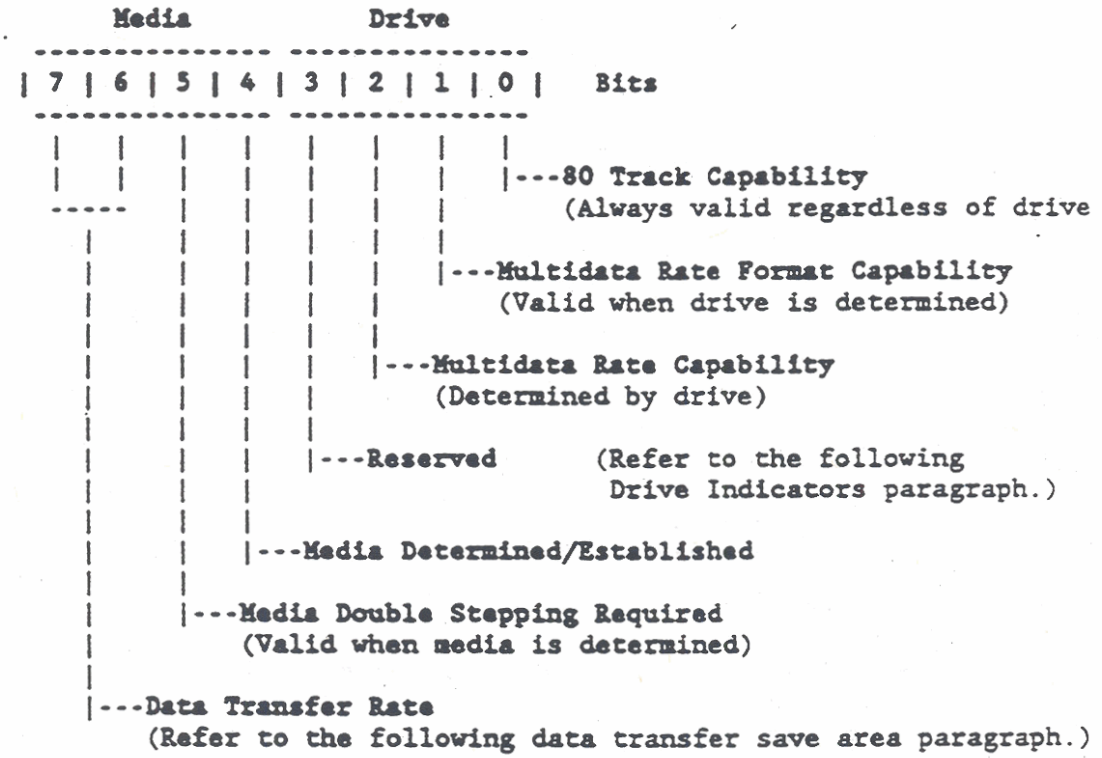
Table 9-2. Track and Sector Values

Media Capacity	Drive Capability	Max. No. of Sectors (AL)	Track No. (CH)	Sector No. (CL)
360 kB	360 kB	8/9	0-39	1-8/9
360 kB	1.2 MB	8/9	0-39	1-8/9
720 kB	1.4 MB	9	0-79	1-9
1.2 MB	1.2 MB	15	0-79	1-15
1.4 MB	1.4 MB	18	0-79	1-18

Floppy Disk Drive State Machine

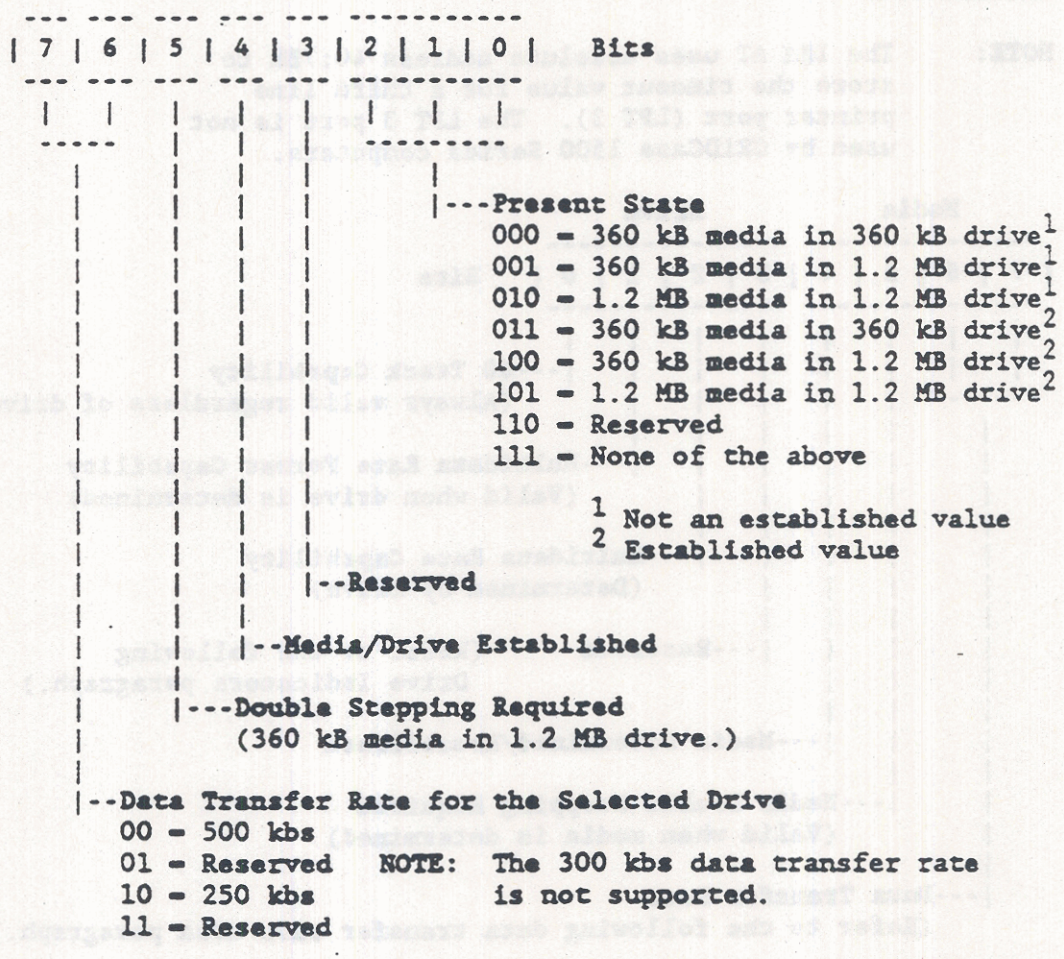
The floppy disk drive state machine is located at absolute address 40:90h for drive A:, 40:91h for drive B:, and 40:7Bh for drive E:. During execution of any BIOS operation for a floppy disk drive, the state machine for the applicable disk drive contains the following information:

NOTE: The IBM AT uses absolute address 40:7Bh to store the timeout value for a third line printer port (LPT 3). The LPT 3 port is not used by GRiDCase 1500 Series computers.



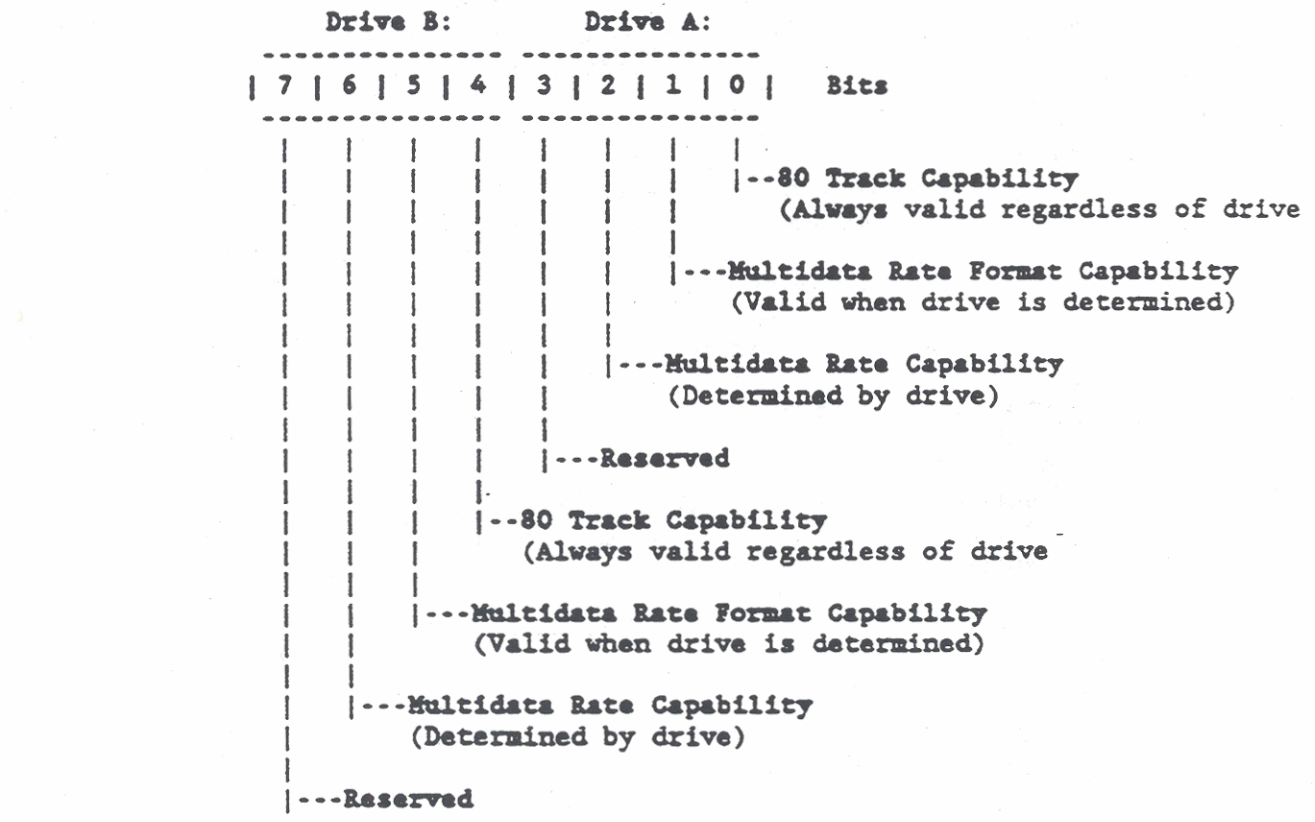
State Machine Compatibility

To ensure compatibility with older lower capacity disk drives, the State Machine contains the following information when returned to the caller of the Floppy Disk Drive BIOS operation:



Drive Indicators

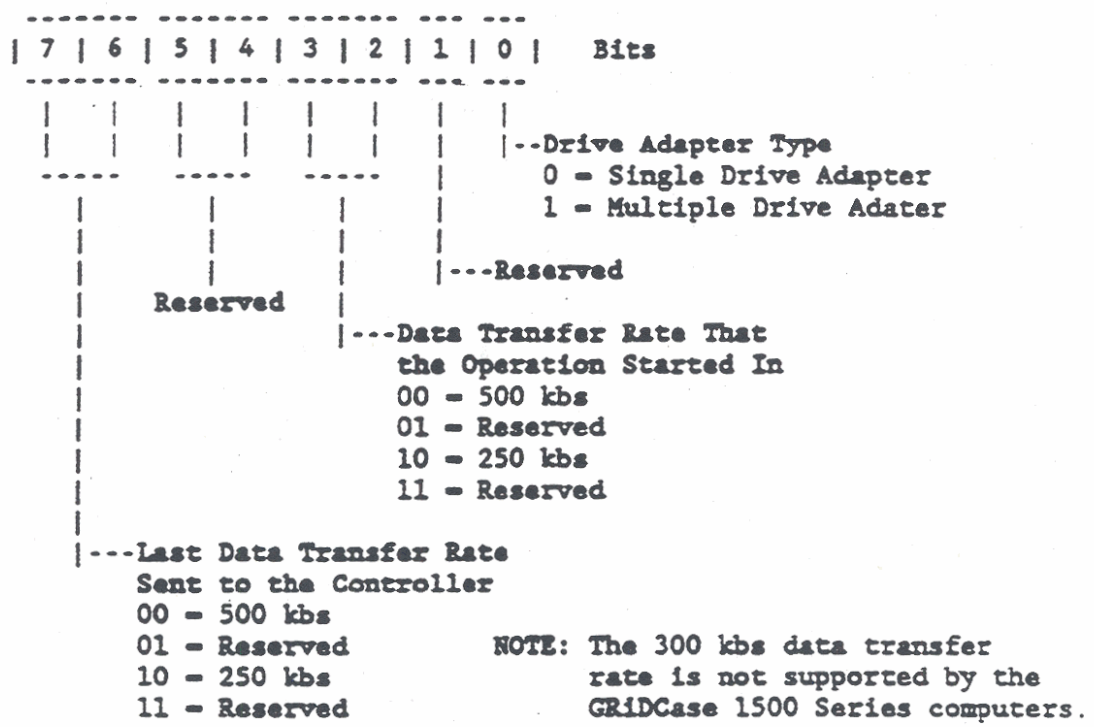
The Drive Indicators for drive A: and B: are located at absolute address 40:8Fh. Before returning, and before translating the State Machine information to compatibility mode, the drive indicator information is copied from the State Machine location to the Drive Indicators location. The Drive A: indicators are copied into the lowest four bits and Drive B: indicators are copied into the highest four bits of the Drive Indicators location as follows:



The Drive Indicators for drive E: are located at absolute address 40:A6h. The drive E: drive indicator contents and operation are identical to drive A:.

Data Transfer Rate Save Area

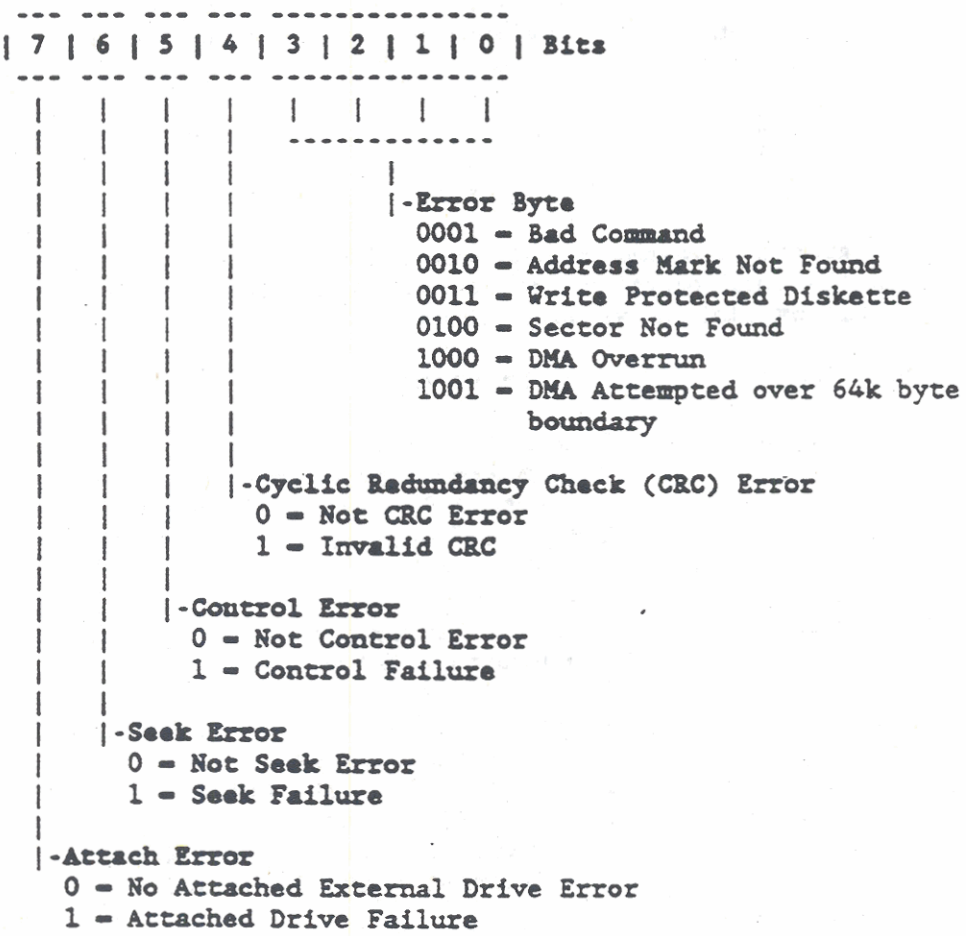
The data transfer rate save area is located at absolute address 40:8Bh and contains the Data Transfer Rate information as follows:



Floppy Disk Drive Service Routine Error Codes

The floppy disk drive service routines, on exit, may return an error code in microprocessor register AH. Bit definitions for the 8-bit register are given in the following description.

The appropriate response to an error code is to reset the Disk Drive Controller and then retry the operation. Since no motor start delays are taken for a read operation, three retries are required on a read operation to ensure that the motor start did not cause the error.



Floppy Disk Drive Configuration

In the standard configuration, the GRiDCase 1500 Series computer internal floppy disk drives are designated as drive A: in the upper position and drive B: in the lower position. The external floppy disk drive designation depends upon its address switch settings and on the number of internal floppy disk drives installed. For the address switch settings refer to the GRiCase 1500 Series Owner's manual. The external drive designations based on the number of internal floppy disk drives are as follows:

No. of Internal Drives	External Drive Physical Designation	External Drive Logical Designation
0	A:	B:
1	B: or A:	E:
2	E: or A:	N/A

The I/O registers located at addresses 379h and 37Ah are used to determine the configuration and operating characteristics of the external floppy disk drive as follows:

I/O Address	Function
379h (Bit 0)	External Disk Drive Address 0 = External Drive is A: 1 = External Drive is B: or E:
379h (Bit 2)	External Floppy Disk Drive Type 0 = 5.25-inch 1 = 3.5-inch
37Ah (Bit 5)	External Floppy Disk Drive Attached 0 = External Drive 1 = No External Drive
37Ah (Bit 6)	External Backup Tape Drive Attached 0 = External Tape Drive 1 = No External Tape Drive
37Ah (Bit 7)	External High Density Drive Attached 0 = External Drive is High Density 1 = External Drive is Normal Density

NOTE: The output of register 37Ah is inverted so that a "0" in any bit position send a "1" to the microprocessor.

The floppy disk drive configuration is also indicated by the ROM-BIOS Equipment Check Service Routine. The service routine invoked via interrupt INT 11h returns a 16-bit word to microprocessor register AX. Register AX bits 0, 6, and 7 provide floppy disk drive status information as follows:

Register AX Bits	Description
0	Floppy Disk Drive (FDD) Identifier 0 = No FDD Installed 1 = FDD(s) Installed
7, 6	Number of FDD(s) Installed 0 0 = 1 FDD 0 1 = 2 FDDs 1 0 = 3 FDDs 1 1 = Reserved

There are also two BIOS Subsystem Functions that affect the Floppy Disk Drive Subsystem configuration. These subsystem functions are accessible through interrupt 15h, and are used to determine the external floppy disk drive designation and to check for a Backup Tape Unit (Pouch Tape) in the disk drive subsystem. Refer to Chapter 3 for a description of the BIOS Subsystem Functions.

Floppy Disk Controller Interface

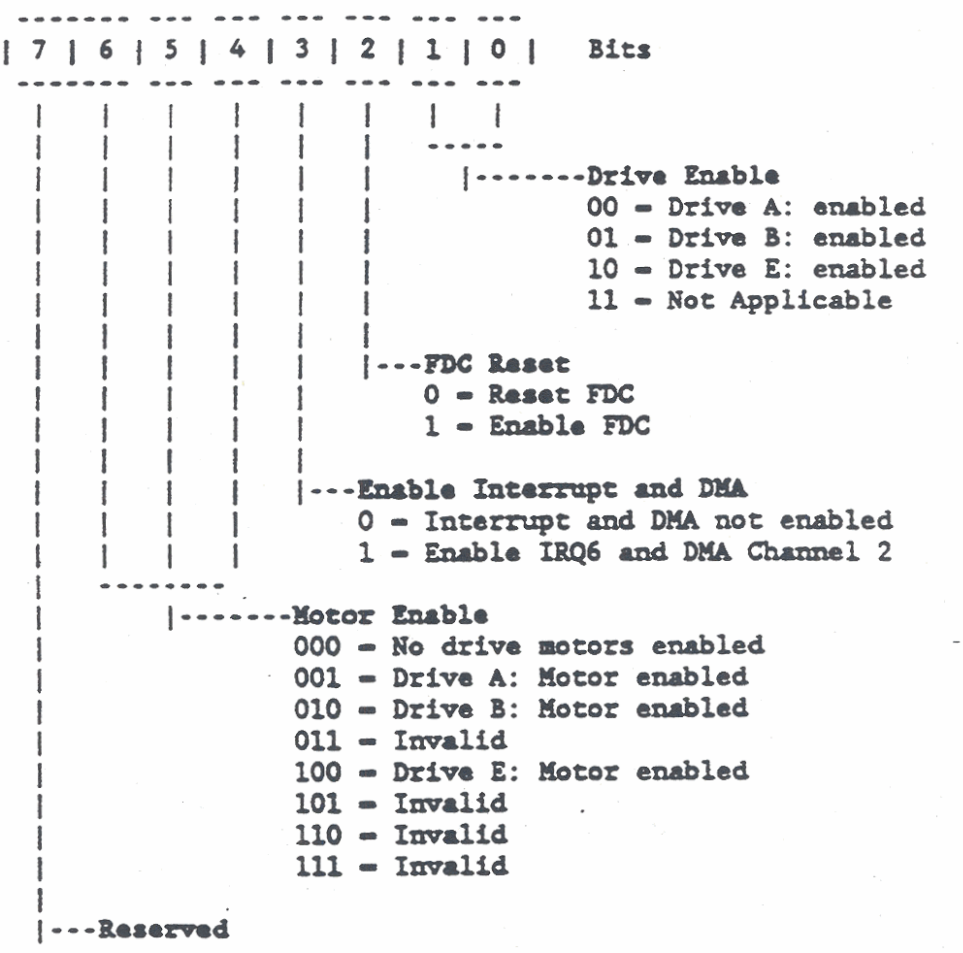
The uPD72065 Floppy Disk Controller (FDC) is used by the GRiDCase 1500 Series computer to support operations of both the 3.5-inch internal drives and to provide the interface to an external drive. The FDC provides four I/O registers that are used to monitor and control all of the floppy disk drive operations. The I/O registers used to support drive operations are:

I/O Address	Function
3F2h	Digital Output Register (Write Only)
3F4h	Main Status Register (Read Only)
3F5h	Floppy Disk Data Register (Read/Write)
3F7h	Digital Input Register (Read) or Floppy Disk Control Register (Write)

To maintain compatibility with the IBM PC AT, the four registers at I/O addresses 3F2h, 3F4h, 3F5h, and 3F7h are used to handle requests for all disk drives in the system. Since only one drive can be active at a time, the same set of registers is used for all floppy disk drives in the system. Data contained within the registers determine which drive is addressed and the operation to be performed. A description of each FDC I/O registers is provided in the following paragraphs.

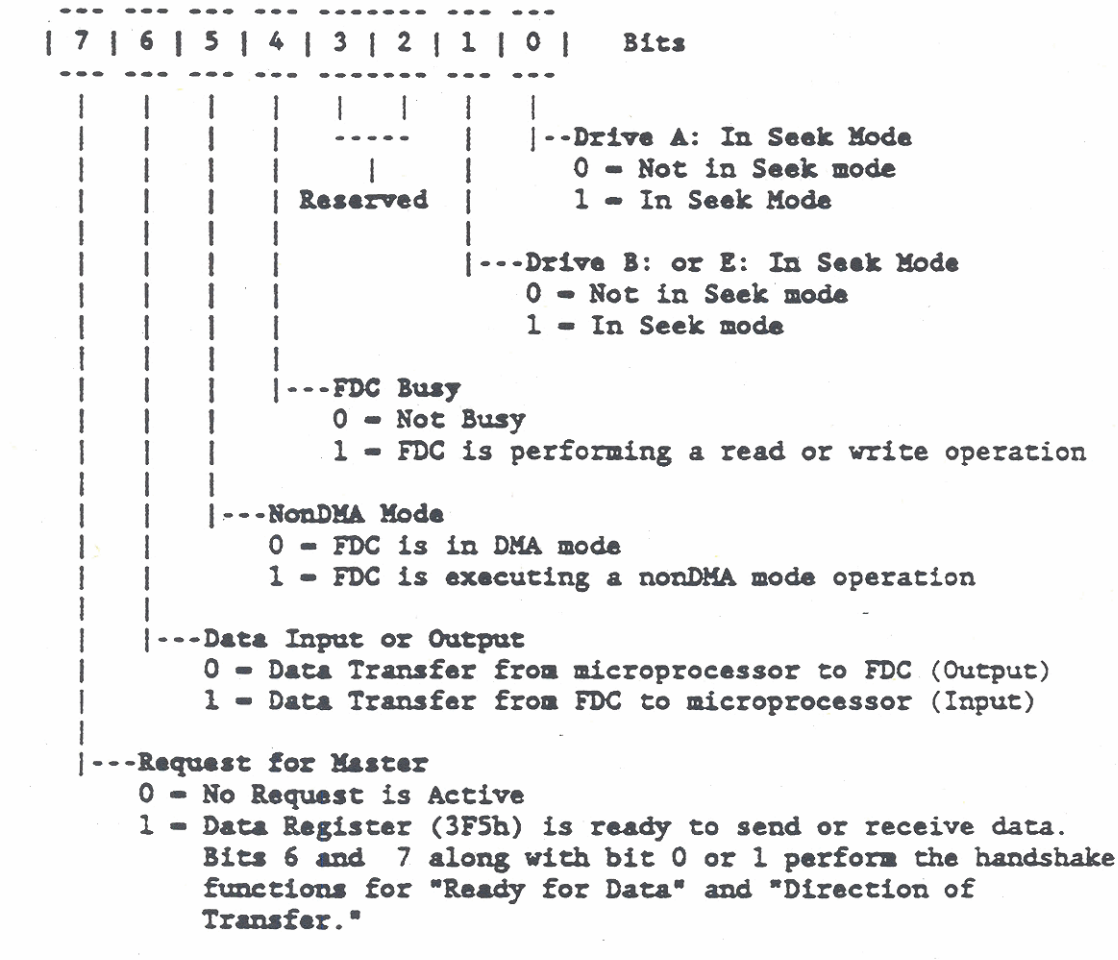
Digital Output Register (3F2h)

The Digital Output register (3F2h) is an 8-bit, write-only register that controls the drive, drive motor, and special feature selection. Bit definitions for the register are as follows:



Main Status Register (3F4h)

The Main Status Register (3F4h) is an 8-bit read-only register that contains the status information to support the transfer of data between the FDC and microprocessor. The Main Status register can be accessed at any time. Bit definitions for the register are as follows:



Floppy Disk Data Register (3F5h)

The Floppy Disk Data register (3F5h) is an 8-bit read and write buffer. Data buffering occurs as commands, data, and result information transfers in either direction between the FDC and the 80286/80386 Microprocessor. Prior to each data transfer, a series of command bytes are written through the buffer. And, following each data transfer, a series of status bytes are read through the same buffer. Bits 6, 7, and 0 or 1 in the Main Status register (3F4h) are used to sequence the transfer of data bytes through the Floppy Disk Data register. The data transferred through the Floppy Disk Data register is described in subsequent paragraphs under Programming the Floppy Disk Controller.

Digital Input/Floppy Disk Control Register (3F7h)

The register at I/O address 3F7h is an 8-bit dual-purpose register with its purpose depending upon whether a read or a write operation is being performed. During a read operation, the register is a Digital Input register used for diagnostic purposes. As a Digital Input register, bits 6 through 0 are reserved for hard disk applications and bit 7 only is used to indicate Floppy Disk Change as follows:

Bit 7

0 = Disk Change Line is not active
1 = Disk Change Line is active. The Disk Change Line is checked when the specified media is not 360k bytes.

During a write operation, the register at I/O address 3F7h is a Floppy Disk Control Register. As a Floppy Disk Control register, bit 7 through 2 are reserved and bits 1 and 0 are used to select the media density and the data transfer rate in Bits Per Second (bps). The media density data transfer rates are as follows:

Bits		Media	Data Transfer
1	0	Density	Rate
0	0	Normal	500k bps
0	1	Normal	Not Supported
1	0	High	250k bps
1	1	High	Invalid

NOTE: The 300k bps data transfer rate (bits 1, 0 = 01) is not supported on the GRiDCase 1500 Series computers.

Programming the uPD72065 Floppy Disk Controller (FDC)

If the ROM-BIOS is not used for floppy disk operations, the uPD72065 FDC can be directly accessed through the I/O registers. The FDC has two registers that are accessed by the 80286/80386 Microprocessor. The Main Status Register (3F4h) provides information about the activities of the floppy disk drives and of the FDC. The Diskette Data Register (3F5h) provides an 8-bit buffer for transferring data, commands, and status information through the FDC to the microprocessor or disk drive. Both of these I/O registers were previously described.

Many FDC commands require that the contents of the Main Status register be read prior to each byte of the multibyte command sequence to determine if the FDC is ready to receive the next byte. Therefore, bits 7 and 0 or 1 of the Main Status register perform a handshake function to implement a software interface between the microprocessor and the FDC.

In addition to the Main Status register, there are four Command Status registers (ST0, ST1, ST2, and ST3). These Command Status registers are accessed through the programmed command operations and return information about command execution. The Command Status registers are described following the command descriptions.

FDC Commands

The FDC provides 16 different programmed commands. The commands are initiated by a multibyte transfer from the microprocessor through the Floppy Disk Data Register (3F5h). Each command has three phases as follows:

1. Command phase

   In the Command phase, the microprocessor initiates a command by writing a specified sequence of bytes to the FDC.

2. Execution phase

   After all of the required command bytes have been sent to the FDC, it proceeds to the Execution phase where the operations specified by the commands are executed. No attention to or supervision of the FDC is required during the Execution phase.

3. Result phase

   When execution is complete, the FDC enters the Result phase and the microprocessor then reads a series of bytes back from the FDC to obtain status information about the operations just performed. Status information for the commands is read from Command Status registers ST0 through ST3 through the Floppy Disk Data Register (3F5h).

Table 9-3 lists each of the 16 commands supported by the FDC and Table 9-4 defines each FDC command in the same sequence that the commands are listed in Table 9-3. In Table 9-4, a 1 means a logical "1" for that bit, a 0 indicates logical "0", and an X means "don't care." Table 9-5 lists the abbreviations that are used in the command definitions.

Table 9-3. Floppy Disk Controller Commands

Command Name	Command Code	Description
Read Data	x6h	Transfer data from the FDC to the microprocessor
Read Deleted Data	xEh	Transfer deleted data from the FDC to the microprocessor
Write Data	x5h	Transfer data to the FDC from the microprocessor
Write Deleted Data	09h	Transfer deleted data to the FDC from the microprocessor
Read ID	xAh	Store the first correct ID on the cylinder in the data register
Read a Track	x2h	Read contents of cylinder from index hole to EOT
Scan Equal	11h	Compare data between FDC and microprocessor
Scan Low or Equal	19h	Compare data between FDC and microprocessor
Scan High or Equal	1Dh	Compare data between FDC and microprocessor
Format a Track	xCh	FDC formats an entire cylinder
Recalibrate	x7h	Retract head to track 0
Sense Interrupt Status	x8h	Determine the cause of an interrupt
Specify	03h	Set head load time and drive stepping rate
Sense Drive Status	x4h	Read drive status from Command Status register ST3
Seek	xFh	Move head to new cylinder
Invalid	---	Any command code not listed in this table.

Table 9-4. Floppy Disk Controller (FDC) Command Definitions

PHASE	R/W	DATA BUS D7	D6	D5	D4	D3	D2	D1	D0	REMARKS
Read Data Command (x6h)										
Command	W	MT	MF	SK	0	0	1	1	0	Command
	W	X	X	X	X	X	HD	US1	US0	Codes
	W		Cylinder Number				(C)			Sector ID
	W		Head		(0/1)		(H)			information
	W		Sector				(R)			before command
	W		Number of Bytes				(N)			execution.
	W		End of Track				(EOT)			
	W		Gap Length				(GPL)			
	W		Data Length				(DTL)			
Execution										Data transfer from FDC to microprocessor
Result	R		Status Register				ST0			Status information after command execution.
	R		Status Register				ST1			
	R		Status Register				ST2			
	R		C							Sector ID information after command execution.
	R		H							
	R		R							
	R		N							
Read Deleted Data Command (xEh)										
Command	W	MT	MF	SK	0	1	1	0	0	Command
	W	X	X	X	X	X	HD	US1	US0	Codes
	W		Cylinder Number				(C)			Sector ID
	W		Head		(0/1)		(H)			information
	W		Sector				(R)			before command
	W		Number of Bytes				(N)			execution.
	W		End of Track				(EOT)			
	W		Gap Length				(GPL)			
	W		Data Length				(DTL)			
Execution										Data transfer from FDC to microprocessor
Result	R		Status Register				ST0			Status information after command execution.
	R		Status Register				ST1			
	R		Status Register				ST2			
	R		C							Sector ID information after command execution.
	R		H							
	R		R							
	R		N							

Table 9-4. Floppy Disk Controller (FDC) Command Definitions (Continued)

PHASE	R/W	DATA BUS D7	D6	D5	D4	D3	D2	D1	D0	REMARKS
Write Data Command (x5h)										
Command	W	MT	MF	0	0	0	1	0	1	Command
	W	X	X	X	X	X	HD	US1	US0	Codes
	W		Cylinder Number				(C)			Sector ID
	W		Head (0/1)				(H)			information
	W		Sector				(R)			before command
	W		Number of Bytes				(N)			execution.
	W		End of Track				(EOT)			
	W		Gap Length				(GPL)			
	W		Data Length				(DTL)			
Execution										Data transfer to FDC from microprocessor
Result	R		Status Register ST0							Status information after command execution.
	R		Status Register ST1							
	R		Status Register ST2							
	R		C							Sector ID information after command execution.
	R		H							
	R		R							
	R		N							
Write Deleted Data Command (0x9h)										
Command	W	MT	MF	0	0	1	0	0	1	Command
	W	X	X	X	X	X	HD	US1	US0	Codes
	W		Cylinder Number				(C)			Sector ID
	W		Head (0/1)				(H)			information
	W		Sector				(R)			before command
	W		Number of Bytes				(N)			execution.
	W		End of Track				(EOT)			
	W		Gap Length				(GPL)			
	W		Data Length				(DTL)			
Execution										Data transfer to FDC from microprocessor
Result	R		Status Register ST0							Status information after command execution.
	R		Status Register ST1							
	R		Status Register ST2							
	R		C							Sector ID information after command execution.
	R		H							
	R		R							
	R		N							

Table 9-4. Floppy Disk Controller (FDC) Command Definitions (Continued)

PHASE	R/W	DATA BUS D7	D6	D5	D4	D3	D2	D1	D0	REMARKS
Read ID Command (xAh)										
Command	W	0	MF	0	0	1	0	1	0	Command
	W	X	X	X	X	X	HD	US1	US0	Codes
Execution										Store first correct ID on cylinder in data register.
Result	R		Status Register ST0							Status information after command execution.
	R		Status Register ST1							
	R		Status Register ST2							
	R		C							Sector ID information after command execution.
	R		H							
	R		R							
	R		N							
Read a Track Command (x2h)										
Command	W	0	MF	SK	0	0	0	1	0	Command
	W	X	X	X	X	X	HD	US1	US0	Codes
	W		Cylinder Number				(C)			Sector ID information before command execution.
	W		Head		(0/1)		(H)			
	W		Sector				(R)			
	W		Number of Bytes				(N)			
	W		End of Track				(EOT)			
	W		Gap Length				(GPL)			
	W		Data Length				(DTL)			
Execution										Data transfer from disk to FDC. The FDC reads contents of cylinder from index hole to EOT.
Result	R		Status Register ST0							Status information after command execution.
	R		Status Register ST1							
	R		Status Register ST2							
	R		C							Sector ID information after command execution.
	R		H							
	R		R							
	R		N							

Table 9-4. Floppy Disk Controller (FDC) Command Definitions (Continued)

PHASE	R/W	DATA BUS D7	D6	D5	D4	D3	D2	D1	D0	REMARKS
Scan Equal Command (11h)										
Command	W	MT	MF	SK	1	0	0	0	1	Command
	W	X	X	X	X	X	HD	US1	US0	Codes
	W	Cylinder Number (C)								Sector ID
	W	Head (0/1) (H)								information
	W	Sector (R)								before command
	W	Number of Bytes (N)								execution.
	W	End of Track (EOT)								
	W	Gap Length (GPL)								
	W	Scan Test (STP)								
Execution										Compare data between FDC and microprocessor
Result	R	Status Register ST0								Status information after command execution.
	R	Status Register ST1								
	R	Status Register ST2								
	R	C								Sector ID information after command execution.
	R	H								
	R	R								
	R	N								
Scan Low or Equal Command (19h)										
Command	W	MT	MF	SK	1	1	0	0	1	Command
	W	X	X	X	X	X	HD	US1	US0	Codes
	W	Cylinder Number (C)								Sector ID
	W	Head (0/1) (H)								information
	W	Sector (R)								before command
	W	Number of Bytes (N)								execution.
	W	End of Track (EOT)								
	W	Gap Length (GPL)								
	W	Scan Test (STP)								
Execution										Compare data between FDC and microprocessor
Result	R	Status Register ST0								Status information after command execution.
	R	Status Register ST1								
	R	Status Register ST2								
	R	C								Sector ID information after command execution.
	R	H								
	R	R								
	R	N								

Table 9-4. Floppy Disk Controller (FDC) Command Definitions (Continued)

PHASE	R/W	DATA BUS D7	D6	D5	D4	D3	D2	D1	D0	REMARKS
Scan High or Equal Command (1Dh)										
Command	W	MT	MF	SK	1	1	1	0	1	Command
	W	X	X	X	X	X	HD	US1	US0	Codes
	W		Cylinder Number				(C)			Sector ID
	W		Head		(0/1)		(H)			information
	W		Sector				(R)			before command
	W		Number of Bytes				(N)			execution.
	W		End of Track				(EOT)			
	W		Gap Length				(GPL)			
	W		Scan Test				(STP)			
Execution										Compare data between FDC and microprocessor
Result	R		Status Register ST0							Status information after command execution.
	R		Status Register ST1							
	R		Status Register ST2							
	R		C							Sector ID information after command execution.
	R		H							
	R		R							
	R		N							
Format a Track (xCh)										
Command	W	0	MF	0	0	1	1	0	0	Command
	W	X	X	X	X	X	HD	US1	US0	Codes
	W		Number of Bytes				(N)			
	W		Sectors per Cylinder				(SC)			
	W		Gap Length				(GPL)			
	W		Data to be Written				(D)			
Execution										FDC formats an entire cylinder
Result	R		Status Register ST0							Status information after command execution.
	R		Status Register ST1							
	R		Status Register ST2							
	R		C							Sector ID information after command execution.
	R		H							
	R		R							
	R		N							

Table 9-4. Floppy Disk Controller (FDC) Command Definitions (Continued)

PHASE	R/W	DATA BUS D7	D6	D5	D4	D3	D2	D1	D0	REMARKS
Recalibrate (x7h)										
Command	W	0	0	0	0	0	1	1	1	Command
	W	X	X	X	X	X	0	US1	US0	Codes
Execution										Retracts head to track 0.
No Result										
Sense Interrupt Status (x8h)										
Command	W	0	0	0	0	1	0	0	0	Command
	W	X	X	X	X	X	0	US1	US0	Codes
Execution										Determine cause of an interrupt
Result	R	Status Register ST0								Status info about FDC at end of seek.
	R	Present Cylinder Number								
Specify (03h)										
Command	W	0	0	0	0	0	0	1	1	Command
	W	|-----SRT-----|					|-----HUT-----|		
	W	|-----------HLT-----------|						ND|	
Execution										Set head load time and drive stepping rate
No Result										
Sense Drive Status (x4h)										
Command	W	0	0	0	0	0	0	1	0	Command
	W	X	X	X	X	X	HD	US1	US0	
Execution										Read drive status from register ST3
Result	R	Status Register ST3								Status info about drive.

Table 9-4. Floppy Disk Controller (FDC) Command Definitions (Continued)

PHASE	R/W	DATA BUS D7	D6	D5	D4	D3	D2	D1	D0	REMARKS
Seek (xFh)										
Command	W	0	0	0	0	1	1	1	1	Command
	W	X	X	X	X	X	HD	US1	US0	
	W	New Cylinder Number								
Execution										Move head to new cylinder.
No Result										
Invalid										
Command	W	Any Invalid Codes								Any command code not listed in Table 9-3.
Execution										FDC goes to standby mode.
Result	R	Status Register ST0								Contains 80h

Table 9-5. FDC Command Definition Abbreviations

Abbr	Name	Description
C	Cylinder Number	Contains the current or selected cylinder number in binary notation.
D	Data	Contains the data pattern to be written to a sector.
DTL	Data Length	When N = 00, DTL indicates the length of data to be read out of or written into the sector.
EOT	End of Track	The last sector number on a cylinder.
GPL	Gap Length	The length of the space between sectors, excluding VCO sync field.
H or HD	Head Number	The selected head number, 0 or 1.
HLT	Head Load Time	Load time for drive as follows: For 360k byte drive, 4 to 512 ms in 4 ms increments. For 1.2M byte drive, 2 to 256 ms in 2 ms increments.

Table 9-5. FDC Command Definition Abbreviations (Continued)

Abbr	Name	Description
HUT	Head Unload Time	Unload time after read or write: For 360k byte drive, 0 to 480 ms in 32 ms increments. For 1.2k byte drive, 0 to 240 ms in 16 ms increments.
MF	FM or MFM	Mode Selected: 0 = FM, 1 = MFM
MT	Multitrack	A "1" selects multitrack operation to read or write a cylinder under both heads (0 and 1) at once.
N	Number of bytes per sector	The number of bytes written in a sector: 00h = 128 bytes 01h = 256 bytes 02h = 512 bytes 03h = 1024 bytes
ND	Nondata	Indicates a nondata mode operation.
R/W	Read/Write	Read (R) or Write (W) operation.
SK	Skip	A "1" indicates skip deleted-data address mark.
SRT	Step Rate	Provides four-bits to indicate the drive stepping rate as follows: 4-Bit Value: 1111 — Drive Stepping Rate: 360k/720k byte 2 ms, 1.2M/1.44M byte 1 ms 4-Bit Value: 1110 — Drive Stepping Rate: 360k/720k byte 4 ms, 1.2M/1.44M byte 2 ms 4-Bit Value: 1101 — Drive Stepping Rate: 360k/720k byte 6 ms, 1.2M/1.44M byte 3 ms
STP	Scan Test	If 1, STP compares data in contiguous sectors with data sent by the microprocessor during the scan operation. If 2, STP reads and compares data in alternate sections.
US0, US1	Unit Select bits 0, 1	Selected drive number encoded the same as bits 0 and 1 of the Digital Out reg (3F2h)
VCO	Voltage Controlled Oscillator	The VCO provides a nominal 2.0 MHz output. The VCO frequency increases when a high-density disk is inserted in the drive, and decreases when a normal density disk is inserted in the drive. The VCO output is used in the data recovery circuit to synchronize the data and speed up rise times.

FDC Command Status Registers

In addition to the Main Status Register that was previously described, there are four Command Status Registers that return status information after a command has been executed. The Command Status Registers (ST0, ST1, ST2, ST3) are examined only during the command result phase after a command has been successfully completed. The Command Status registers provide additional information about the operations that occurred during command execution. The Result Phase of each command in Table 9-4 indicates which registers are available for examination at I/O address 3F5h after each command. Tables 9-6 through 9-9 define the contents of the four Command Status Registers.

Table 9-6. FDC Command Status Register (ST0)

7	6	5	4	3	2	1	0	Bits

- Bits 1, 0 — Unit Select Bits (US1, US0)
  Indicates the drive selected at interrupt as follows:
  - 0 0 = Drive A:
  - 0 1 = Drive B:
  - 1 0 = Drive E:
  - 1 1 = Unused
- Bit 2 — Head Address (HD)
  - 0 = Head 0 selected at interrupt
  - 1 = Head 1 selected at interrupt
- Bit 3 — Not Ready Flag
  - 0 = Drive is ready to accept data
  - 1 = A Read or Write command was issued when the drive was not ready.
- Bit 4 — Equipment Check Flag
  - 0 = Normal operation
  - 1 = The drive sent a fault signal or the Track 0 signal did not occur in Recalibrate command after 77 step pulses.
- Bit 5 — Seek End Flag
  - 0 = Seek command in process
  - 1 = Seek command complete
- Bits 7, 6 — Interrupt Code
  - 0 0 = Normal termination of command
  - 0 1 = Abrupt termination of command, incomplete execution
  - 1 0 = Invalid command was issued, execution was not begun
  - 1 1 = Abnormal termination because disk drive READY signal changed states during command execution

Table 9-7. FDC Command Status Register (ST1)

7	6	5	4	3	2	1	0	Bits

- Bit 0 — Missing Address Mark
  - 0 = Normal operation
  - 1 = No ID address mark was detected (Bit 0 of ST2 is also set)
- Bit 1 — Not Writable
  - 0 = Normal operation
  - 1 = A Write Protect signal was received during a write, write deleted data, or format a track command.
- Bit 2 — No Data
  - 0 = Normal Operation
  - 1 = A sector specified in the ID register was not found during a subsequent read data, write deleted data, or a scan command. Or, during a Read ID, the ID field cannot be read without an error, or the starting sector cannot be found while executing a Read Cylinder command.
- Bit 3 — Not Used, Always Set to 0
- Bit 4 — Overrun
  - 0 = Normal operation
  - 1 = The FDC was not serviced by the microprocessor within a certain time limit during a data transfer.
- Bit 5 — Data Error
  - 0 = Normal operation
  - 1 = The FDC detected a CRC error in either the ID or data field.
- Bit 6 — Not Used, Always Set to 0.
- Bit 7 — End of Cylinder
  - 0 = Normal Operation
  - 1 = The FDC tried to access a sector beyond the last sector of a cylinder.

Table 9-8. FDC Command Status Register (ST2)

7	6	5	4	3	2	1	0	Bits

- Bit 0 — Missing Data Field Address Mark
  - 0 = Normal operation
  - 1 = The FDC cannot find a data address mark or deleted data address mark when reading data from the disk.
- Bit 1 — Bad Cylinder Flag
  - 0 = Normal operation
  - 1 = The cylinder number read from the disk differs from that stored in the ID register and the cylinder number is FFh. See ST1, Bit 2
- Bit 2 — Scan Not Satisfied
  - 0 = Normal operation
  - 1 = The FDC cannot find a sector on the cylinder that meets the conditions specified in the scan command.
- Bit 3 — Scan Equal Hit
  - 0 = No data match found.
  - 1 = The contiguous sector data matches the data from the microprocessor during execution of the scan command.
- Bit 4 — Wrong Cylinder Flag
  - 0 = Normal operation
  - 1 = The contents of the sector ID on the disk do not match the cylinder number stored in the ID register. See ST1, Bit 2.
- Bit 5 — Data Error in Data Field
  - 0 = Normal operation
  - 1 = The FDC detected a CRC error in a data field.
- Bit 6 — Control Mark Flag
  - 0 = Normal operation
  - 1 = The FDC encountered a sector containing a deleted data address mark during execution of a Read Data or Scan command.
- Bit 7 — Not Used, Always Set to 0

Table 9-9. FDC Command Status Register (ST3)

7	6	5	4	3	2	1	0	Bits

- Bit 0 — Unit Select 0 Flag
  - 0 = Disk drive "unit 0 select" signal is low.
  - 1 = Disk drive "unit 0 select" signal is high.
- Bit 1 — Unit Select 1 Flag
  - 0 = Disk drive "unit 1 select" signal is low.
  - 1 = Disk drive "unit 1 select" signal is high.
- Bit 2 — Head Address Flag
  - 0 = Disk drive "side select" signal is low.
  - 1 = Disk drive "side select" signal is high.
- Bit 3 — Two Side Signal Flag
  - 0 = Disk drive "two side" signal is low.
  - 1 = Disk drive "two side" signal is high.
- Bit 4 — Track 0 Flag
  - 0 = Disk drive "track 0" signal is low.
  - 1 = Disk drive "track 0" signal is high.
- Bit 5 — Ready Flag
  - 0 = Disk drive "ready" signal is low.
  - 1 = Disk drive "ready" signal is high.
- Bit 6 — Write Protect Flag
  - 0 = Disk drive "write protect" signal is low.
  - 1 = Disk drive "write protect" signal is high.
- Bit 7 — Fault Flag
  - 0 = Disk drive "fault" signal is low.
  - 1 = Disk drive "fault" signal is high.

HARD DISK DRIVE

An IBM XT compatible internal hard disk drive (Option 320 or 321) can be factory installed in place of the lower internal floppy disk drive. The hard disk drive is a 3.5-inch Winchester technology drive with a formatted capacity of 10M bytes (Option 320) or 20M bytes (Option 321). Other hard disk drive characteristics are listed in Table 9-10. Information provided in this chapter is applicable to XT-compatible hard disk drives only.

NOTE: The Option 320 and some option 321 hard disk drives are not IBM AT compatible since they support 8-bit wide DMA data transfers. IBM AT compatible hard disk drive options are described in Chapter 10.

Table 9-10. Hard Disk Drive Specifications

Characteristic	10M Byte (Option 320)	20M Byte (Option 321)
Capacity (Formatted)		
Sectors Per Track:	17	34
Bytes Per Sector:	512	512
Bytes per Track:	8704	17408
Bytes per Surface:	5.3M	10.6M
Bytes per Disk:	10.6M	21.1M
Data Transfer Rate (Mbytes/Second)		
Maximum:	3.2	7.5
Access Time (milliseconds)		
Minimum:	8	18
Average:	100	78
Maximum:	230	130
Functional		
Rotation Speed (RPM):	2322	2597
Recording Density (bpi):	13083	27410
Track Density (tpi):	849	849
Cylinders:	612	615
Tracks:	1224	1230
Encoding Method:	MFM	RLL
Head Positioning:	Head Access Motor and Embedded Servo	Head Access Motor and Embedded Servo

NOTE: Throughout this chapter, hexadecimal values are shown with a letter "h" suffix.

Hard Disk Drive Software Interface

The hard disk drive is accessed by the same ROM-BIOS that is used by the floppy disk drive. Most of the service routine functions used by the floppy disk drive are also used by the hard disk drive. But, the hard disk drive has additional functions that are not required by the floppy disk drive. The routine sharing is accomplished by using 80h and 81h as hard disk drive numbers. The DL register in the 80286/80386 microprocessor is then used to pass the disk drive number.

Hard Disk Drive Service Routines

The ROM-BIOS supports the following hard disk-related interrupt service routine functions:

Interrupt (Hex)	Function No. (AH reg)	Description
13	00h	Reset Hard Disk Drive System
13	01h	Read Disk Drive Status from Last Operation
13	02h	Read Data From Desired Sectors Into Memory
13	03h	Write Data From Memory to Desired Sectors
13	04h	Verify that Disk Sectors are Valid
13	05h	Format Desired Track on Disk
13	06h	Unused
13	07h	Unused
13	08h	Read Current Disk Drive Parameters
13	09h	Initialize Drive Pair Characteristics
13	0Ah	Read Long
13	0Bh	Write Long
13	0Ch	Seek
13	0Dh	Reset Hard Disk Drive System
13	0Eh	Unused
13	0Fh	Unused
13	10h	Test Drive Ready
13	11h	Recalibrate
13	12h	Unused
13	13h	Unused
13	14h	Controller Internal Diagnostic
13	15h	Read DASD Type

The following paragraphs summarize the operation of the interrupt service routine functions.

Hard Disk Drive Service Routine Description

The 80286/80386 Microprocessor register inputs and outputs for the disk drive interrupt service routine functions are listed in the following paragraphs. For the read, write, verify, and format functions, the contents of registers DS, BX, CX, and DX, are

preserved, and register AL returns the number of sectors actually read, written, or verified. Register AH determines the funcion within the service routine that is invoked, while the other microprocessor registers further define the action to be performed. On exit, if an error occurred, the Carry flag is set to "1" and register AH contains an Error Code. The Error Codes and their descriptions are listed following the interrupt service routine.

The hard disk drive parameters are requested by using 80h or 81h as the hard disk drive numbers. The microprocessor DL register is then used to pass the disk drive number as follows:

On Input
DL < 80h indicates Floppy Disk Drive Operation
DL => 80h indicates Hard Disk Drive Operation

On Output (Exit)
DH = Maximum useable value for head number
DL = Number of consecutive acknowledging drives attached (1-2)
CH = Low-order 8 bits of the maximum useable value for the 10-bit cylinder number
CL = Two high-order bits for the cylinder number maximum useable value and six bits (bits 5-0) for the sector number maximum useable value.

AH = 00h Reset Hard Disk System
Refer to the following paragraph on Hard Disk System Reset.

AH = 01h Read Disk Drive Status From Last Operation
DL = Drive number (80h-81h)
On Exit:
AH = Error code ("0" if no error)
AL = Disk Status
Carry = 0 (no error)
Carry = 1 (error)

AH = 02h Read Data From Desired Sectors Into Memory
DL = Drive number (80h-81h)
DH = Head number (0-1)
CH = Cylinder number (0-612/615)[1]
CL = Sector number (1-17/34)[2]
AL = Number of sectors to read
ES:BX = Buffer Address for Reads and Writes
On Exit:
AL = Number of sectors read
AH = Error code
Carry = 0 (no error)
Carry = 1 (error)

AH = 03h Write Data from Memory to Desired Sectors
DL = Drive number (80h-81h)
DH = Head number (0-1)
CH = Cylinder number (0-612/615)[1]
CL = Sector number (1-17/34)[2]
AL = Number of sectors to write
ES:BX = Buffer Address for Reads and Writes
On Exit:
AL = Number of sectors written
AH = Error code
Carry = 0 (no error)
Carry = 1 (error)

NOTE:

1. For cylinder and sector numbers, the virgule (/) separates the maximum values for each option (320/321).

2. The two high order bits of the 10-bit cylinder number are provided by bits 7 and 6 of the sector number register CL.

AH = 04h Verify That Disk Sectors are Valid
DL = Drive number (80h-81h)
DH = Head number (0-1)
CH = Cylinder number (0-612/615)[1]
CL = Sector number (1-17/34)[2]
AL = Number of sectors
On Exit:
AL = Number of sectors verified
AH = Error code
Carry = 0 (no error)
Carry = 1 (error)

AH = 05h Format Desired Track on Disk
DL = Drive number (80h-81h)
DH = Head number (0-1)
CH = Cylinder number (0-612/615)[1]
ES:BX = Points to a 512 byte buffer (NOTE)
On Exit:
AL = Number of sectors verified
AH = Error code
Carry = 0 (no error)
Carry = 1 (error)

NOTE: In the 512 byte format buffer, the first 2x(sector/track) bytes contain values F and N for each sector. The value F represents 00h for a good sector or 80h for a bad sector. The value N is the sector number. For an interleave of 2 and 17 sectors/track the buffer contains:

00h,01h,00h,0AH,00H,02H,00H,0Bh,00h,03h,00h,0Ch
00h,004,00h,0Dh,00h,05h,00h,0Eh,00h,06h,00h,0Fh
00h,07h,00h,10h,00h,08h,00h,11h,00h,09h

AH = 08h Read Current Disk Drive Parameters
DL = Drive number (80h-81h)
ES:BX = Buffer Address for Reads and Writes
On Exit:
AH = Error code
Carry = 0 (no error)
Carry = 1 (error)

AH = 09h Initialize Drive Pair Characteristics
DL = Drive number (80h-81h)
ES:BX = Buffer Address for Drive Characteristics
On Exit:
AH = Error code
Carry = 0 (no error)
Carry = 1 (error)

NOTE: Refer to the following paragraph on initializing the drive characteristics.

AH = 0Ah Read Long
DL = Drive number (80h-81h)
DH = Head number (0-1)
CH = Cylinder number (0-612/615)[1]
CL = Sector number (1-17/34)[2]
AL = Number of sectors to read
ES:BX = Buffer Address for Reads and Writes
Reads 512 data bytes and 4 ECC bytes
On Exit:
AL = Number of sectors read
AH = Error code
Carry = 0 (no error)
Carry = 1 (error)

AH = 0Bh Write Long
DL = Drive number (80h-81h)
DH = Head number (0-1)
CH = Cylinder number (0-612/615)[1]
CL = Sector number (1-17/34)[2]
AL = Number of sectors to write
ES:BX = Buffer Address for Reads and Writes
Writes 512 data bytes and 4 ECC bytes
On Exit:
AL = Number of sectors written
AH = Error code
Carry = 0 (no error)
Carry = 1 (error)

AH = 0Ch Seek
DL = Drive number (80h-81h)
DH = Head number (0-1)
CH = Cylinder number (0-612/615)[1]
CL = Sector number (1-17/34)[2]
On Exit:
AH = Error code
Carry = 0 (no error)
Carry = 1 (error)

AH = 0Dh Reset Hard Disk System
Refer to the following paragraph on Hard Disk System Reset.

AH = 10h Test Drive Ready
DL = Drive number (80h-81h)
On Exit:
AH = Error code
Carry = 0 (no error)
Carry = 1 (error)

AH = 11h Recalibrate
DL = Drive number (80h-81h)
On Exit:
AH = Error code
Carry = 0 (no error)
Carry = 1 (error)

AH = 14h Controller Internal Diagnostic
DL = Drive number (80h-81h)
On Exit:
AH = Error code
Carry = 0 (no error)
Carry = 1 (error)

AH = 15h Read DASD Type

On Exit if Carry Flag is not set: otherwise, an error
AH = 00 Drive not present
AH = 01 Floppy Disk Drive, No change line available
AH = 02 Floppy Disk Drive, Change line available
AH = 03 Hard Disk Drive

NOTE: When AH = 03, registers CX,DX provide the number of 512 byte blocks.

Hard Disk System Reset

Two interrupt service routine functions (AH = 00 and AH = 0D) provide a disk system reset. Either reset causes the ROM-BIOS to write a reset value to a Hard Disk Controller (HDC) register. After a reset, all HDC registers are set to their default values.

Hard Disk Drive Service Routine Error Codes

The hard disk drive service routine functions, on Exit, may return an Error code in microprocessor register AH. An error is indicated if the Carry flag is set to "1" and the error code can then be read from register AH. The error codes are listed and described in Table 9-11. If an error is returned, the appropriate action is to first reset the disk and then retry the failed operation.

Table 9-11. Hard Disk Drive Service Routine Error Codes

Error Code (Hex)	Error Name	Description
00	No Error	No error detected
01	Bad Command	Bad command passed to disk I/O
02	Bad Address Mark	Address mark not found
04	Record Not Found	Requested sector not found
05	Bad Reset	Reset failed
07	Initialize Failed	Drive parameter activity failed
09	DMA Boundary	Data extends too far
0A	Bad Sector	Bad sector flag detected
0B	Bad Track	Not used
10	Bad ECC	Bad ECC on disk read
11	Data Corrected	ECC corrected data error
20	Bad Controller	Controller failed
40	Bad Seek	Seek operation failed
80	Time Out	Attachment failed to respond
AA	Not Ready	Drive not ready
BB	Undefined Error	An undefined error occurred
CC	Write Fault	Write fault on selected drive
E0	No Error	Status error/Error Register=0
FF	Sense Failed	Not used

Error code 11h indicates that a data read had a recoverable error that was corrected by the ECC algorithm. The data may be good. However, the BIOS service routine indicates an error to allow the controlling program to determine if it is good data. With good data, the error is not repeatable.

Initializing Drive Characteristic

To initialize the drive characteristics per interrupt service routine function 09 (AH=09), the ROM-BIOS issues an interrupt. Interrupt 41h points to a data block for drive 0 (DL = 80h) and interrupt 46h points to a data block for drive 1 (DL = 81h). In either case, the interrupt is vectored to a 16 byte memory location for the Fixed Disk Parameter Table. The Fixed Disk Parameter Table contains the following information:

Number of Bytes	Description
2	Maximum number of cylinders
1	Maximum number of heads
2	Not used
2	Starting Write Precompensation cylinder
1	Maximum ECC Error Data Burst length
1	Control Byte Bit 7 Disable disk-access retries Bit 6 Disable ECC retries Bit 3 More than 8 heads
3	Not used
2	Landing Zone
1	Number of Sectors/Track
1	Reserved

To define a new set of parameters, first build a table for up to 15 types. Then, place the corresponding vector into interrupt 41h (drive 0) or interrupt 46h (drive 1).

Hard Disk Controller Interface

The GRiDCase 1500 Series computer, with options 320 and 321, uses a JVC Model 5523, 3.5-inch Winchester Hard Disk Controller (HDC). The Model 5523 HDC has a different dash number suffix depending upon the type of disk drive that it supports. The dash numbers and related drive types used in the GRiDCase 1500 Series computers are as follows:

Dash No.	Disk Drive Type
-1	10M Byte JVC
-2	20M Byte JVC
-3	20M Byte ALPS

The 20M byte JVC and the 20M byte ALPS disk drives are interchangeable when their respective HDCs are also interchanged. Unless otherwise noted, the following paragraphs are applicable to all HDC dash numbers.

The primary interface between the computer and HDC is provided by four I/O registers. The four registers are bidirectional to allow the computer to send commands and data to the HDC and also to receive data and status information from the HDC. The four registers and their bidirectional functions are given in the following list and they are described in the subsequent paragraphs.

I/O Address	Register Name	Read Function	Write Function
320h	Data Register	Receive data or status from the HDC.	Send commands or data to the HDC.
321h	Hardware Status/ Reset Register	Read hardware status	Reset HDC
322h	Configuration/ Control Select Register	Read configuration information	Select and initialize the HDC for access
323h	Interrupt and DMA Enable Register	Not used	Write DMA and interrupt mask

Data Register (320h)

The Data register at I/O address 320h provides an access port to the 8-bit wide bus that connects the microprocessor to the HDC. All data, command parameters, and status information to be written to, or read from the disk, are transferred between the HDC and microprocessor via this port. The HDC also contains a separate Sector Buffer that supports data transfers between the disk drive and the HDC. Using separate buffers allows data transfers between the HDC and disk drive to be executed independently of the data transfers between the HDC and microprocessor.

Hardware Status/Reset Register (321h)

When the register at I/O address 321h is read, it returns the following status information:

Bit	Signal	Description
5	IRQ14	0 = No hard disk interrupt pending 1 = Hard disk interrupt pending
4	DRQ3	0 = No DMA request 1 = Controller ready for DMA transfer
3	Busy	0 = HDC is idle 1 = HDC is executing a command
2	Control/ Data	0 = Data transfer in progress 1 = Command or Status transfer in progress
1	Input/ Output	0 = Transfer from computer to HDC 1 = Transfer from HDC to computer
0	Request	Handshake for byte transfers 0 = Not ready for transfer 1 = Ready for transfer

Configuration/Control Select Register (322h)

Reading the register at I/O address 322h returns the factory-set hard disk drive configuration information. The configuration information indicates the disk capacity. When the same register is written, the HDC is prepared to accept a command and the BUSY bit in register 321h is set.

Interrupt and DMA Enable Register (323h)

The register at I/O address 323h is used to enable interrupt requests and DMA data transfers. The two-bit register is used as follows:

Bits 1	Bits 0	Description
0	0	No interrupt or DMA
0	1	Enable DMA request (DRQ3)
1	0	Enable Interrupt Request (IRQ14)
1	1	DMA and Interrupt Request Enabled

Hard Disk Controller (HDC) Operation

The four registers that were previously described provide the primary interface between the computer and the HDC. The bidirectional registers are used to transfer commands, status information, and data between the computer and HDC. Operation of the I/O registers and their affect on HDC operation is described in the following paragraphs.

The HDC supports 15 operational commands (Class Code 0) and 5 diagnostic commands (Class Code E). A summary of the operations required to use the commands is given in the following steps. Table 9-12 provides a summary of the commands. For additional command information, refer to the manufacturer's (JVC) User's manual.

1. A write to I/O address 322h selects the HDC and sets BUSY (BSY) and REQUEST (REQ) in the Status register (321h). The computer completes the selection process by reading the status register.

2. When the computer reads the status register (321h) with REQ set, it proceeds to send the first byte of a six-byte command block. After each byte is received by the HDC, REQ is cleared and then set again for the next byte.

3. The command is sent to the HDC via the Data register at I/O address 320h. This is the same register that is used to read or write data following a Read Data or Write Data command.

4. After each six-byte command, the HDC clears BSY and sends a Command Completion Byte to the Data register (320h). The computer reads the Hardware Status register (321h) to determine when Input/Output = 1 and then reads the Command Completion Byte from the Data register.

5. Only bits 5 and 1 of the Command Completion Byte are significant. Bit 5 is set to 0 to indicate that the drive received the command. Bit 1 is set to 1 if an error occurred while the command was being transferred.

6. If the Command Completion byte indicates that an error occurred (bit 1 = 1), the computer should issue a Read Status command for the specified drive before any other commands are executed to prevent the loss of error status. The error status is read in four consecutive bytes from I/O register 320h. Error codes that indicate the cause of an error are read from bits 0 through 5 of byte 0. A list of possible error codes is given in Table 9-13.

7. Read Data or Write Data commands are used to transfer data via the I/O register at address 320h.

Table 9-12. HDC Command Summary

Class Code	Op Code	Command Name	Description
0	00	Test Drive Ready	Selects and verifies that a drive is ready.
0	01	Recalibrate	Positions the read/write head on track 0.
0	03	Read Status	Returns one status byte and a three-byte logical address for the specified drive.
0	04	Format Drive	Formats all sectors with ID and data fields and then writes 6Ch into the data fields from the starting address to the end of the disk.
0	05	Verify Sectors	Checks specified cylinder for correct ID and interleave. Does not read data.
0	06	Format Track	Formats a specified track and writes 6Ch into the data fields. Also clears bad-sector flags if the track was previously formatted with the Format Bad Track command.
0	07	Format Bad Track	Formats a specified track and sets the bad-sector flag in each sector header. The data fields are not written to.
0	08	Read Data	Reads the specified number of sectors starting with the initial Sector Address contained in the control byte.
0	0A	Write Data	Writes the specified number of sectors starting with the initial Sector Address contained in the control byte.
0	0B	Seek	Seeks to the track specified in the control byte. The drive must be formatted.

Table 9-12. HDC Command Summary (Continued)

Class Code	Op Code	Command Name	Description
0	0C	Set Parameters	Initializes the HDC to configure up to two drives with different capacities and characteristics. Both drives must have the same manufacturer and mode number. Parameters are sent with an 8-byte Parameter Block.
0	0D	Last Corrected Burst Length	Returns one byte of data containing the length of the last corrected error burst. If no burst was corrected since the last Power ON or Reset, the byte is all zeros.
0	0E	Read Sector Buffer	Returns the contents of the HDC Disk Controller Sector Buffer. The computer must accept as many bytes as there are in a sector in drive 0.
0	0F	Write Sector Buffer	Initializes the format data used optionally by the Format commands. It writes enough data bytes to the HDC Disk Controller Sector Buffer to fill a sector on drive 0. The bytes are not sent to a drive.
E	00	Execute Sector	Performs a data pattern test Buffer Diagnostic on the sector buffer.
E	03	Execute Drive Diagnostic	Tests the drive and drive to HDC interface.
E	04	Execute Controller Diagnostic	Tests the HDC and HDC to computer interface.
E	05	Read Long	Similar to Read Data (08) except ECC bytes are read directly from the disk and passed to the computer.

Table 9-12. HDC Command Summary (Continued)

Class Code	Op Code	Command Name	Description
E	06	Write Long	Similar to Write Data (0A) except that ECC bytes are passed unaltered to the controller which writes them on the disk following the data.
7	04	Controller Diagnostic	Initializes routines that test all HDC functions.
7	05	Read Long	Reads a specified number of consecutive sectors and the four ECC bytes with each sector. Errors are not corrected
7	06	Write Long	Writes a specified number of consecutive sectors and the four ECC bytes for each sector.

Table 9-13. Controller (HDC) Error Codes

Error Code (Hex)	Error Name	Description
00	No Error	No error was detected.
03	Write Fault	HDC received a Write Fault from the drive.
04	Drive Not Ready	HDC did not receive a ready signal from the drive or drive has been selected.
06	Track 0 Not Found	Recalibrate command failed
08	Drive Still Seeking	Drive is still performing a buffered seek.
11	Uncorrectable Data Error	The ECC detected an error burst greater than its correction capability.

Table 9-13. Controller (HDC) Error Codes (Continued)

Error Code (Hex)	Error Name	Description
12	Data Address Mark Not Found	Sector ID was read but Data Address Mark was not readable.
15	Seek Error	Sector ID was not found on the selected track or a CRC error occurred on the ID field.
18	Correctable Data Error	The ECC detected an error that was subsequently corrected.
19	Track is Flagged as a Bad Track	A sector was read with a Bad Block Mark in its ID field.
20	Invalid Command	A command was received with an invalid class or op code.
21	Illegal Sector Address	A sector was addressed beyond the capacity of the drive.
30	Sector Buffer Error	Sector buffer diagnostic (E0, E4) error.
31	Controller ROM Checksum Error	Checksum error during Controller Diagnostic (E4).
32	ECC Error	ECC generator failed Controller diagnostic (E4).

FLOPPY DISK DRIVE EXPANSION PORT

The Floppy Disk Drive Expansion port is supported with either the standard 3.5-inch floppy disk drives or an optional hard disk drive as the internal drive. The GRiDCase 1500 Series computer transfers data directly between its internal disk controller and an external disk drive. The direct transfers are accomplished through a dedicated interface connector located on the rear panel of the computer. This external drive bus structure allows the GRiDCase 1500 Series computer to support an external 3.5-inch disk drive or 5.25-inch disk drive that is compatible with the IBM PC/AT. Also, a 40M byte Backup Tape System is available for connection through the external bus. Refer to Table 9-1 for configurations.

The program interface to the internal disk drive is through a standard set of ROM-BIOS calls, which mask the differences between the 3.5-inch disk drive and the 5.25-inch disk drive. To provide compatibility with the internal disk, there are switches on the external drive that allow the user to designate it as the A:, B:, or E: drive. The switch setting is readable by the ROM-BIOS. The hardware then sets the chip select to the correct value for the drive select and motor on bits. The value of the switch setting is readable at I/O location 379h bit 0. Refer to the floppy disk drive configuration paragraph.

External Peripheral Connector (5) Pin Definitions

The External Peripheral connector (No. 5) is dedicated to the floppy disk drive interface. The connector is a 25-pin, subminiature, D-type shell, female connector mounted on the rear panel of the GRiDCase 1500 Series computer. The External Peripheral connector is identical to the Parallel Printer connector (No. 4), except that the External Peripheral connector has a plastic insert in pin 13. The plastic insert keys the connector for use by GRiD Systems cables and peripherals such as floppy disk drives and the tape backup unit. Keying the connector in this manner discourages the cross-connecting of cables from other types of peripheral devices. Connecting other types of devices such as printers to the External Peripheral connector can seriously damage the main logic board in the computer.

CAUTION

The External Peripheral connector contains a plastic insert in pin 13. DO NOT REMOVE THE INSERT. The insert keys the connector for use by GRiD Systems cables and peripherals such as floppy disk drives. DO NOT USE FORCE when making connections. Connection of peripheral devices such as printers can seriously damage the computer and peripheral device.

The External Peripheral connector layout is shown in Figure 9-2, and the interface connector pinouts and signal names are given in Table 9-14. All signals are TTL Compatible.

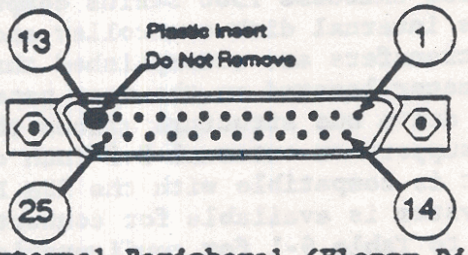


Figure 9-2. External Peripheral (Floppy Disk Drive Interface) Connector Pin Layout

Table 9-14. External Peripheral Connector Pin Definitions

Pin #	Signal	Direction	Description
1	CORDDCPWR	---	Do Not Use (Refer to NOTE 1)
2	B or A-	Input	Indicates the setting of the drive designation (address) switches on the external drive as follows: 00 = A: drive 01 = B: or E: drive Pin 2 is readable at I/O address 379h, Bit 0.
3	EFLOPPY-	Input	Goes low to indicate that an external drive is connected. Pin 3 is readable at address 37Ah, Bit 5.
4	Ground	---	Signal Ground
5	WRITE PROTECT-	Output	This line goes low if a write-protected disk is inserted in the external drive.
6	HDDRIVE-	Output	Goes low if a high density drive is attached to the external peripheral connector. Pin 6 is readable at address 37Ah, Bit 7.
7	Signal Ground	---	Signal Ground
8	TAPEDETECT	Input	Goes high if a Model 3403 Tape Backup System is attached. Pin 8 is readable at address 37Ah, Bit 6.
9	DIRECTION-	Input	Determines the direction that the stepper motor goes to position the read/write heads when a STEP- pulse is received. If this signal is low, the heads are moved toward the spindle by the STEP- pulse. If this signal is high, the heads are moved away from the spindle by the STEP- pulse.
10	DISKCHANGE-	Output	Goes low when a floppy disk is removed from the active disk drive. The signal goes low when power is turned ON and is reset by the STEP- signal.

Table 9-14. External Peripheral Connector Pin Definitions (Continued)

Pin #	Signal	Direction	Description
11	LOWDENSITY	Output	Sets the drive for normal or high density disk opertion. 0 = High-density operation 1 = Normal-density operation
12	LOWDENSITY-	Output	Inverse to pin 11.
13	Keyway	---	Plastic insert to guide installation of the external peripheral drive cable.
14	+ 5V dc +/-10%	---	Provides power to the 3.5-inch disk drives.
15	3OR5-	Input	A factory set jumper in the external drive sets this line low for a 5.25-inch drive and pulls the line high for a 3.5-inch drive. A status bit is readable from I/O location 379h, bit 2.
16	SIDE SELECT	Input	Selects the upper read/write head when low and the lower read/write head when high.
17	FLOPPYRDATA-	Output	Carries data read from the disk. The data is a combination of data and clock pulses.
18	TRK00-	Output	Goes low when the head is positioned over track 00 Track 00 is the outermost track on the disk.
19	WRITE GATE-	Input	Goes low if write data is valid. Goes high if the floppy diskette is write-protected.
20	FLOPPY_WDATA-	Input	Carries data to be written onto the disk. The read/write head current is reversed when this signal changes from high to low.

Table 9-14. External Peripheral Connector Pin Definitions (Continued)

Pin #	Signal	Direction	Description
21	STEP-	Input	This line is pulsed low to move the heads one step in the direction determined by the DIRECTION- signal.
22	MOTOR2-	Input	Goes low to start the spindle motor. Normal speed is reached within 500 ms.
23	DRIVE2-	Input	When low, the drive responds to all lines from the output interface except motor enable.
24	FDDINDEX-	Output	One low pulse is generated per revolution of the spindle. The falling edge of the pulse indicates the beginning of the track.
25	CORDDCPWR	---	Do Not Use (Refer to NOTE 1)

NOTES:

1. Pins 1 and 25 were originally connected to a source of +16V dc power for use in powering an external device. In later computer models and models that were subsequently reworked, pins 1 and 25 are open (no connection). This change prevents damage to the computer main logic board if a device is improperly connected.

2. The tilde symbol (~) indicates active low (low-true) logic.

CHAPTER 10: COMPATIBLE HARD DISK DRIVE SUBSYSTEM

The GRiDCase 1500 Series computer Compatible Hard Disk Drive Subsystem supports IBM AT compatible, Winchester technology hard disk drives. These compatible hard disk drives have a formatted capacity of 20M bytes, 40M bytes, or 100M bytes depending upon the system option that has been factory installed. The compatible hard disk drives support 512-byte sectors; high-speed, Programmed Input/Output (PIO) data transfers; an error correction code (ECC) of up to five bits on data fields; multiple sector operations across track and cylinder boundaries; and on-board diagnostic tests. Table 10-1 lists the compatible hard disk drive configurations by their option number.

Table 10-1. Compatible Hard Disk Drive Configurations

Item	Option	Qty	Configuration	Capacity
1	321	1	3.5-inch Floppy Disk Drive	1.4M bytes
		1	20M byte Hard Disk Drive (IBM AT Compatible)	20M bytes 21.4M bytes total
2	324	1	40M byte Hard Disk Drive (IBM AT Compatible)	40M bytes total
3	325	1	100M byte Hard Disk Drive (IBM AT Compatible)	100M bytes total

NOTES: A complete list of disk drive configurations is provided in Table 9-1.

Option 321 provides a 20M byte hard disk drive. Computers with tracking numbers greater than K15000 are supplied with AT compatible drives.

In the configurations listed in Table 10-1, an internal floppy disk drive is included with the 20M byte hard disk drive (Option 321). Operation of the floppy disk drive, the options that do not contain a compatible hard disk drive, and the external disk drive port are described in Chapter 9. This chapter (10) describes operation of the IBM AT compatible hard disk drives only.

Additional options that include other AT compatible hard disk drives and internal floppy disk drives may become available. This chapter contains information that is applicable to all AT compatible hard disk drives unless otherwise specified. Where extensive information is required that is unique to a particular

disk drive, the information is supplied in an appendix attached to the back portion of this manual. Refer to the Table of Contents for a list of applicable appendices.

AT compatible hard disk drives are different from drives that are XT or PC compatible mainly in the method of transferring data. AT compatible hard disk drives transfer data in 16-bit words using Programmed Input/Output (PIO) instructions. The disk drives previously used in GRiDCase computers (XT and PC compatible) transfer data in 8-bit bytes using Direct Memory Access (DMA) instructions. Operating characteristics of the AT compatible hard disk drives are listed in Table 10-2.

Table 10-2. Compatible Hard Disk Drive Specifications

Characteristic	Option 321 (20M Byte)	Option 324 (40M Byte)	Option 325 (100M Byte)
Capacity (Formatted)			
Total (Megabytes):	21.4	42.6	104
Tracks Per Surface:	616	976	776
Track Density (tpi):	1150	1000	1150
Track Capacity (bytes):	16,896	13,312	16,896
Sectors per Track:	17	17	33
Bytes per Block:	512	512	512
Blocks per Drive:	41,976	83,270	204,864
Read/Write Characteristics:			
Interface:	Task File	Task File	Task File
Recording Method:	2 of 7 RLL Code	2 of 7 RLL Code	2 of 7 RLL Code
Recording Density (bpi):	21,594	21,379	23,441
Flux Density (ID) Flux Reversals/inch:	14,396	14,253	15,627
Buffer Size (bytes):	512 bytes	8k bytes	32k bytes
Data Transfer Rate (kBytes/Second)			
Maximum:	280	575	280
Functional			
Interleave:	3:1	1:1	1:1
Seek Times (in ms at nominal dc)			
Track to Track:	8	10	8
Average Access:	27	27	25
Maximum:	50	50	45
Latency (average):	8.40	8.33	8.40
Rotational Speed (+/-0.1%):	3575 rpm	3600 rpm	3575 rpm

Table 10-2. Compatible Hard Disk Drive Specifications (Continued)

Characteristic	Option 321 (20M Byte)	Option 324 (40M Byte)	Option 325 (100M Byte)
Functional (Continued)			
Controller Overhead (ms):	1	1	1
Start Time (Power ON)			
0 rpm to READY			
Typical (sec):	7	7	15
Maximum (sec):	15	20	20
Stop Time (Power OFF)			
Typical (sec):	5	7	15
Maximum (sec):	10	20	20
Start/Stop Cycles:	10k	10k	--
Physical			
Actuator Type:	Rotary Voice Coil	Rotary Voice Coil	Rotary Voice Coil
Servo:	Embedded	Embedded	Embedded
Number of			
Disks:	1	2	4
Data Surfaces:	2	4	8
Data Heads:	4	4	8
R/W Head park at Power OFF:	Automatic	Automatic	Automatic
Power Consumption (Watts)			
Read/Write Mode:	4.2	4.2	--
Idle Mode:	2.0	2.5	--
Standby Mode:	0.5	0.5	--
Environmental Characteristics			
Temperature (^{o}Centigrade)			
Operating:	5 to 55	5 to 55	5 to 55
Non Operating:	-40 to 60	-40 to 60	-40 to 60
Thermal Gradient			
Maximum:	20/hour	20/hour	20/hour
Humidity (% Noncondensing)			
Operating:	8 to 80	8 to 80	8 to 80
Non Operating:	8 to 80	8 to 80	8 to 80
Max Wet Bulb			
Temp (^{o}C):	26	26	26
Altitude (Relative to Sea Level)			
Operating Range (kft):	-0.2/10	-0.2/10	-0.2/10
Nonoperating (kft):	40	40	40

NOTE: Throughout this chapter, hexadecimal values are shown with a letter "h" suffix.

COMPATIBLE HARD DISK DRIVE ROM-BIOS INTERFACE

The compatible hard disk drive is accessed by the same ROM-BIOS interface that is used for the hard disk drives described in Chapter 9. The interrupt service routine used by the noncompatible hard disk drives are also used by the compatible hard disk drives. The compatible hard disk drive ROM-BIOS service routine is different mainly in the number of heads, cylinders, and sectors that can be specified.

The hard disk drive is addressed by using 80h as the hard disk drive identifier. The DL register in the 80286/80386 microprocessor is used to pass the disk drive identifier. With any value in the DL register, the low seven bits (6-0) provide the drive number and the eighth bit (7) is set to "0" to indicate a floppy disk drive operation, or to "1" to indicate a hard disk drive operation.

Hard Disk Drive Service Routine

The ROM-BIOS supports the following hard disk-related interrupt service routine functions:

Interrupt (Hex)	Function No. (AH reg)	Description
13	00h	Reset Hard Disk Drive System
13	01h	Read Disk Drive Status from Last Operation
13	02h	Read Data From Desired Sectors Into Memory
13	03h	Write Data From Memory to Desired Sectors
13	04h	Verify that Disk Sectors are Valid
13	05h	Format Desired Track on Disk
13	06h	Unused
13	07h	Unused
13	08h	Read Current Disk Drive Parameters
13	09h	Initialize Drive Pair Characteristics
13	0Ah	Read Long
13	0Bh	Write Long
13	0Ch	Seek
13	0Dh	Reset Hard Disk Drive System
13	0Eh	Unused
13	0Fh	Unused
13	10h	Test Drive Ready
13	11h	Recalibrate
13	12h	Unused
13	13h	Unused
13	14h	Controller Internal Diagnostic
13	15h	Read DASD Type

The following paragraphs summarize the operation of the interrupt service routine functions.

Hard Disk Drive Service Routine Descriptions

The 80286/80386 Microprocessor register inputs and outputs for the disk drive interrupt service routine functions are listed in the following paragraphs. For the read, write, verify, and format functions, the contents of registers DS, BX, CX, and DX, are preserved, and register AL returns the number of sectors actually read, written, or verified. Register AH determines the service routine function that is invoked, while the other microprocessor registers further define the action to be performed. On exit, if an error occurred, the Carry flag is set to "1" and register AH contains an Error Code. The Error Codes and their descriptions are listed following the interrupt service routine.

The hard disk drive parameters are requested by using the DL register in the 80286/80386 microprocessor to pass the disk drive number as follows:

On Input
- DL = Disk Drive Number
  Bits 6-0 contain the drive number (0 for one hard disk drive) and bit 7 indicates floppy disk (0) or hard disk (1). For one hard disk drive, the drive number is 80h. If a second hard disk drive is added, its drive number would be 81h.

On Output (Exit)
- DH = Maximum useable value for head number
- DL = Number of attached hard disk drives (1)
- CH = Low-order 8 bits of the maximum useable value for the 10-bit cylinder number
- CL = Two high-order bits for the cylinder number maximum useable value and six bits (bits 5-0) for the sector number maximum useable value.

AH = 00h Reset Hard Disk System
Refer to the following paragraph on Hard Disk System Reset.

AH = 01h Read Disk Drive Status From Last Operation
DL = Drive number (80h-81h)
On Exit:
- AH = Error code ("0" if no error)
- AL = Disk Status
- Carry = 0 (no error)
- Carry = 1 (error)

NOTES:

1. For head, cylinder, and sector numbers, the virgule (/) separates the maximum values for each option (321/324/325).

2. The two high order bits of the 10-bit cylinder number are provided by bits 7 and 6 of the sector number register CL.

AH = 02h Read Data From Desired Sectors Into Memory
DL = Drive number (80h-81h)
DH = Head number (0-3/3/7)
CH = Cylinder number (0-635/804/775)[1]
CL = Sector number (1-33/26/33)[2]
AL = Number of sectors to read
ES:BX = Buffer Address for Reads and Writes
On Exit:
AL = Number of sectors read
AH = Error code
Carry = 0 (no error)
Carry = 1 (error)

AH = 03h Write Data from Memory to Desired Sectors
DL = Drive number (80h-81h)
DH = Head number (0-3/3/7)
CH = Cylinder number (0-635/804/775)[1]
CL = Sector number (1-33/26/33)[2]
AL = Number of sectors to write
ES:BX = Buffer Address for Reads and Writes
On Exit:
AL = Number of sectors written
AH = Error code
Carry = 0 (no error)
Carry = 1 (error)

AH = 04h Verify That Disk Sectors are Valid
DL = Drive number (80h-81h)
DH = Head number (0-3/3/7)
CH = Cylinder number (0-635/804/775)[1]
CL = Sector number (1-33/26/33)[2]
AL = Number of sectors
On Exit:
AL = Number of sectors verified
AH = Error code
Carry = 0 (no error)
Carry = 1 (error)

AH = 05h Format Desired Track on Disk
DL = Drive number (80h-81h)
DH = Head number (0-3/3/7)
CH = Cylinder number (0-635/804/775)[1]
ES:BX = Points to a 512 byte buffer (NOTE)
On Exit:
AL = Number of sectors verified
AH = Error code
Carry = 0 (no error)
Carry = 1 (error)

NOTE: In the 512 byte format buffer, the first 2x (sector/track) bytes contain values F and N for each sector. The value F represents 00h for a good sector or 80h for a bad sector. The value N is the sector number. For an interleave of 2 and 17 sectors/track the buffer contains:

00h,01h,00h,0AH,00H,02H,00H,0Bh,00h,03h,00h,0Ch
00h,004,00h,0Dh,00h,05h,00h,0Eh,00h,06h,00h,0Fh
00h,07h,00h,10h,00h,08h,00h,11h,00h,09h

AH = 08h Read Current Disk Drive Parameters
DL = Drive number (80h-81h)
ES:BX = Buffer Address for Reads and Writes
On Exit:
AH = Error code
Carry = 0 (no error)
Carry = 1 (error)

AH = 09h Initialize Drive Pair Characteristics
DL = Drive number (80h-81h)
ES:BX = Buffer Address for Drive Characteristics
On Exit:
AH = Error code
Carry = 0 (no error)
Carry = 1 (error)

NOTE: Refer to the following paragraph on initializing the drive characteristics.

AH = 0Ah Read Long
DL = Drive number (80h-81h)
DH = Head number (0-3/3/7)
CH = Cylinder number (0-635/804/775)[1]
CL = Sector number (1-33/26/33)[2]
AL = Number of sectors to read
ES:BX = Buffer Address for Reads and Writes
Reads 512 data bytes and 4 ECC bytes
On Exit:
AL = Number of sectors read
AH = Error code
Carry = 0 (no error)
Carry = 1 (error)

AH = 0Bh Write Long
DL = Drive number (80h-81h)
DH = Head number (0-3/3/7)
CH = Cylinder number (0-635/804/775)[1]
CL = Sector number (1-33/26/33)[2]
AL = Number of sectors to write
ES:BX = Buffer Address for Reads and Writes
Writes 512 data bytes and 4 ECC bytes
On Exit:
AL = Number of sectors written
AH = Error code
Carry = 0 (no error)
Carry = 1 (error)

AH = 0Ch Seek
DL = Drive number (80h-81h)
DH = Head number (0-3/3/7)
CH = Cylinder number (0-635/804/775)[1]
CL = Sector number (1-33/26/33)[2]
On Exit:
AH = Error code
Carry = 0 (no error)
Carry = 1 (error)

AH = 0Dh Reset Hard Disk System
Refer to the following paragraph on Hard Disk System Reset.

AH = 10h Test Drive Ready
DL = Drive number (80h-81h)
On Exit:
AH = Error code
Carry = 0 (no error)
Carry = 1 (error)

AH = 11h Recalibrate
DL = Drive number (80h-81h)
On Exit:
AH = Error code
Carry = 0 (no error)
Carry = 1 (error)

AH = 14Fh Controller Internal Diagnostic
DL = Drive number (80h-81h)
On Exit:
AH = Error code
Carry = 0 (no error)
Carry = 1 (error)

AH = 15h Read DASD Type
On Exit if Carry Flag is not set: otherwise, an error
AH = 00 Drive not present
AH = 01 Floppy Disk Drive, No change line available
AH = 02 Floppy Disk Drive, Change line available
AH = 03 Hard Disk Drive

NOTE: When AH = 03, registers CX, DX provide the number of 512 byte blocks.

Hard Disk System Reset

Two interrupt service routine functions (AH = 00 and AH = 0D) provide a disk system reset. Either reset causes the ROM-BIOS to write a reset value to an HDC register. After a reset, all HDC registers are set to their default values. Interrupt service routine function AH = 00 also resets the floppy disk drive if one is installed.

Hard Disk Drive Service Routine Error Codes

The hard disk drive service routine functions, on Exit, may return an Error code in microprocessor register AH. An error is indicated if the Carry flag is set to "1" and the error code can then be read from register AH. The error codes are listed and described in Table 10-3. If an error is returned, the appropriate action is to first reset the disk and then retry the failed operation.

Table 10-3. Hard Disk Drive Service Routine Error Codes

Error Code (Hex)	Error Name	Description
00	No Error	No error detected
01	Bad Command	Bad command passed to disk I/O
02	Bad Address Mark	Address mark not found
04	Record Not Found	Requested sector not found
05	Recalibrate Failure	Reset failed
07	Initialize Failed	Drive parameter activity failed
09	Data Size Boundary	Data extends too far
0A	Bad Sector	Bad sector flag detected
0B	Not used	Not used
10	Bad ECC	Uncorrectable data error
11	Data Corrected	ECC corrected data error
20	Bad Controller	Controller failed
40	Bad Seek	Seek operation failed
80	Time Out	Attachment failed to respond
AA	Not Ready	Drive not ready
BB	Undefined Error	An undefined error occurred
CC	Write Fault	Write fault on selected drive
E0	No Error	Status error/Error Register=0

Error code 11h indicates that a data read had a recoverable error that was corrected by the ECC algorithm. The data may be good. However, the BIOS service routine indicates an error to allow the controlling program to determine if it is good data. With good data, the error is not repeatable.

Initializing Drive Characteristic

To initialize the drive characteristics per interrupt service routine function 09h (AH=09h), the ROM-BIOS issues an interrupt. Interrupt 41h points to a data block for drive 0 (DL = 80h) and interrupt 46h points to a data block for drive 1 (DL = 81h). In either case, the interrupt is vectored to a 16 byte memory location for the Fixed Disk Parameter Table. The Fixed Disk Parameter Table contains the following information:

Number of Bytes	Description
2	Maximum number of cylinders
1	Maximum number of heads
2	Not used
2	Starting Write Precompensation cylinder
1	Not used
1	Control Byte Bit 7 Disable disk-access retries Bit 6 Disable ECC retries Bit 3 More than 8 heads
3	Not used
2	Landing Zone
1	Number of Sectors/Track
1	Reserved

To define a new set of parameters, first build a table for up to 15 types. Then, place the corresponding vector into interrupt 41h (drive 0) or interrupt 46h (drive 1). The hard disk drive type is selectable through Real-Time Clock I/O register 19h (Drive C Extended Byte) and register 1Ah (Drive D Extended Byte. Refer to Table 6-1.

COMPATIBLE HARD DISK CONTROLLER INTERFACE

The GRiDCase 1500 Series computers support the compatible hard disk drives listed in Table 10-4. Unless otherwise specified, information in the following paragraphs is applicable to all compatible hard disk drives.

Table 10-4. Compatible Hard Disk Drives

Drive Model No.	Drive Type	Capacity (Formatted)	Controller	Remarks
CP3022	5	20M Bytes	IBM AT Compatible Adapter	Manufactured by Conner Peripherals, Inc., San Jose, CA 95131
CP342 (Low-Power)	6	42M Bytes	IBM AT Compatible Adapter	Manufactured by Conner Peripherals, Inc., San Jose, CA 95131
CP344 (High-Power)	4	42M Bytes	IBM AT Compatible Adapter	Manufactured by Conner Peripherals, Inc., San Jose, CA 95131
CP3104	7	104.9M Bytes	IBM AT Compatible Adapter	Manufactured by Conner Peripherals, Inc., San Jose, CA 95131

The compatible hard disk drive uses a Task File interface between the microprocessor and the drive. The Task File interface is a set of registers that allow the ROM-BIOS to execute the drive commands and transfer data between the drive and computer memory. The drive adapter decodes microprocessor I/O addresses 1F0 through 1F7 and 3F6. These addresses are used to access the Task File registers.

The microprocessor addresses the drive using Programmed Input and Output (PIO). With PIO the general sequence of operation is as follows:

1. The microprocessor loads the Task File registers including a Command register.

2. The disk drive then locates the correct cylinder on the disk.

3. An interrupt (IRQ14) is generated by the drive to indicate that service is required.

4. A data transfer takes place.

Table 10-5 lists the Task File registers by their I/O address and function, and then the operation of each register is described in the subsequent paragraphs.

Table 10-5. Task File Registers

I/O Address	Read Function	Write Function
1F0h	Data	Data
1F1h	Error	Write Precomp
1F2h	Sector Count	Sector Count
1F3h	Sector Number	Sector Number
1F4h	Cylinder Low	Cylinder Low
1F5h	Cylinder High	Cylinder High
1F6h	Drive/Head (SDH)	Drive/Head (SDH)
1F7h	Status	Command
3F6h	Alternate Status	Digital Output
3F7h	Not Used	Not Used

NOTE: Write Precomp register (1F1h) is not used on the 20M Byte drive. Also, the Read Drive Address register (3F7h) is not used

Data Register (1F0h)

The Data register (1F0h) is a 16-bit, high-speed, read and write register. All data transferred to or from the disk drive must pass through the Data register. The register is also used to transfer the sector table during format commands and the data associated with the identify command. All data transfers are 16-bits wide except for ECC bytes, which are transferred during read-long and write-long commands. The ECC bytes are slower 8-bit operations that occur after the transfer of data.

Each word of data is stored on the disk with the Least Significant Byte (LSB) first and the Most Significant Byte (MSB) last. This first to last relationship is important to remember during ECC tests.

Error Register (1F1h)

The Error register (1F1h) is an 8-bit, read-only, register that contains the status from the last command executed by the drive or a status code. The status from the last command is valid only if Status register (1F7h) bit 0 is set to "1." The register contains an error code if the drive has just been turned on or has just completed the internal diagnostic. The internal diagnostic and resultant error codes are described later in this chapter under the Drive Diagnostic command. The status bits from the last command are defined as follows:

7	6	5	4	3	2	1	0	Bits
								---Not Used
								---Track 0 (TK0)
								0 = No TK0 Error
								1 = Track 0 not found during a recalibrate command
								---Aborted Command (ABRT)
								0 = Normal command execution
								1 = Command aborted due to drive status error (not ready, write fault, etc.) or the command code is invalid.
								---Not Used
								---Requested ID Not Found (IDNF)
								0 = Sector ID located
								1 = Requested Sector ID not found
								---Not Used
								---Uncorrectable Data Error (UNC)
								0 = No uncorrectable errors
								1 = Indicates that a data error was not correctable by the ECC
								---Bad Block Detected (BBK)
								0 = No bad data blocks
								1 = Indicates that a bad block mark was encountered in the sector ID field. A bad block is created with the format command.

Write Precompensation Register (1F1h)

The Write Precompensation register (1F1h) is an 8-bit, write-only, register that is used with the Set Buffer Mode command to control the Read Look-Ahead operation. The register is loaded with AAh to enable the operation or with 55h to disable the operation. The Set Buffer Mode command is described later in this chapter.

The write-only function of this register is not used with the 20M byte drives (Option 321).

Sector Count Register (1F2h)

The Sector Count register (1F2h) is an 8-bit read/write register that is used to define the number of sectors of data to be read or written. A count of 256 sectors is specified when the register contents are zero. The count is decremented as each sector is read so that the register contains the number of sectors remaining to be accessed if an error occurs in a multisector operation. This register also indicates the number of sectors per track when executing an initialize drive command.

If Power Commands are implemented (refer to Table 10-6), this register is used to provide power down and time-out parameters and status.

Sector Number Register (1F3h)

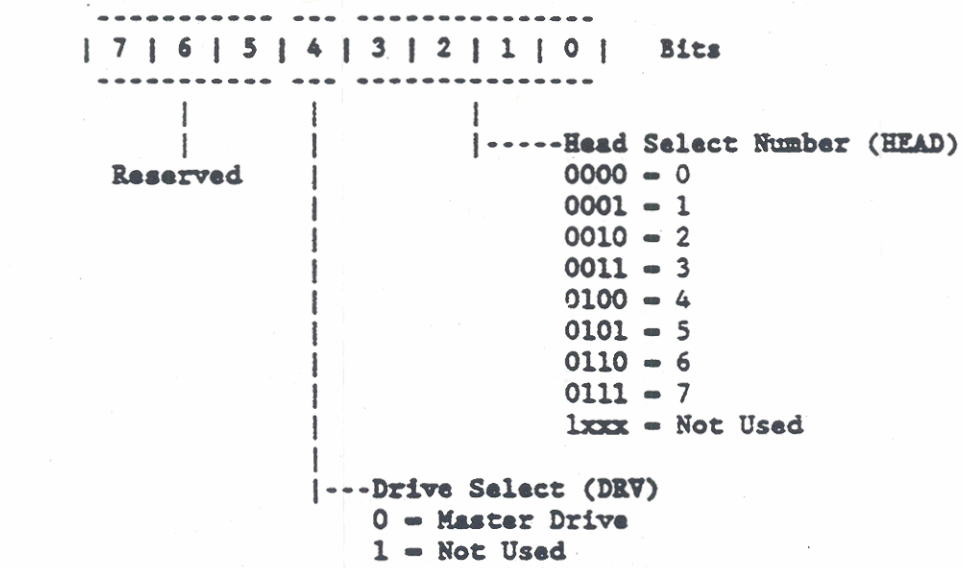
The Sector Number register (1F3h) is an 8-bit read/write register that contains the starting sector number for any disk access. During multiple sector data transfers, if the previous sector read or sector write was successful, the Sector Number register is updated to point to the next sector that will be read or written. At the completion of each sector access, and at the completion of a sector access command, the register contents are updated to indicate the last sector correctly read or the sector where an error occurred.

Cylinder Low/Cylinder High Registers (1F4h/1F5h)

The Cylinder Low and Cylinder High registers (1F4h and 1F5h, respectively) are read/write registers that contain the 10-bit starting cylinder number for any disk access. The Cylinder Low register contains the eight low order bits while the Cylinder High register contains the two high order bits. At the completion of each sector access, and at the completion of a sector access command, the Cylinder Low and Cylinder High register contents are updated to indicate the last cylinder correctly read or the cylinder where an error occurred.

SDH Register (1F6h)

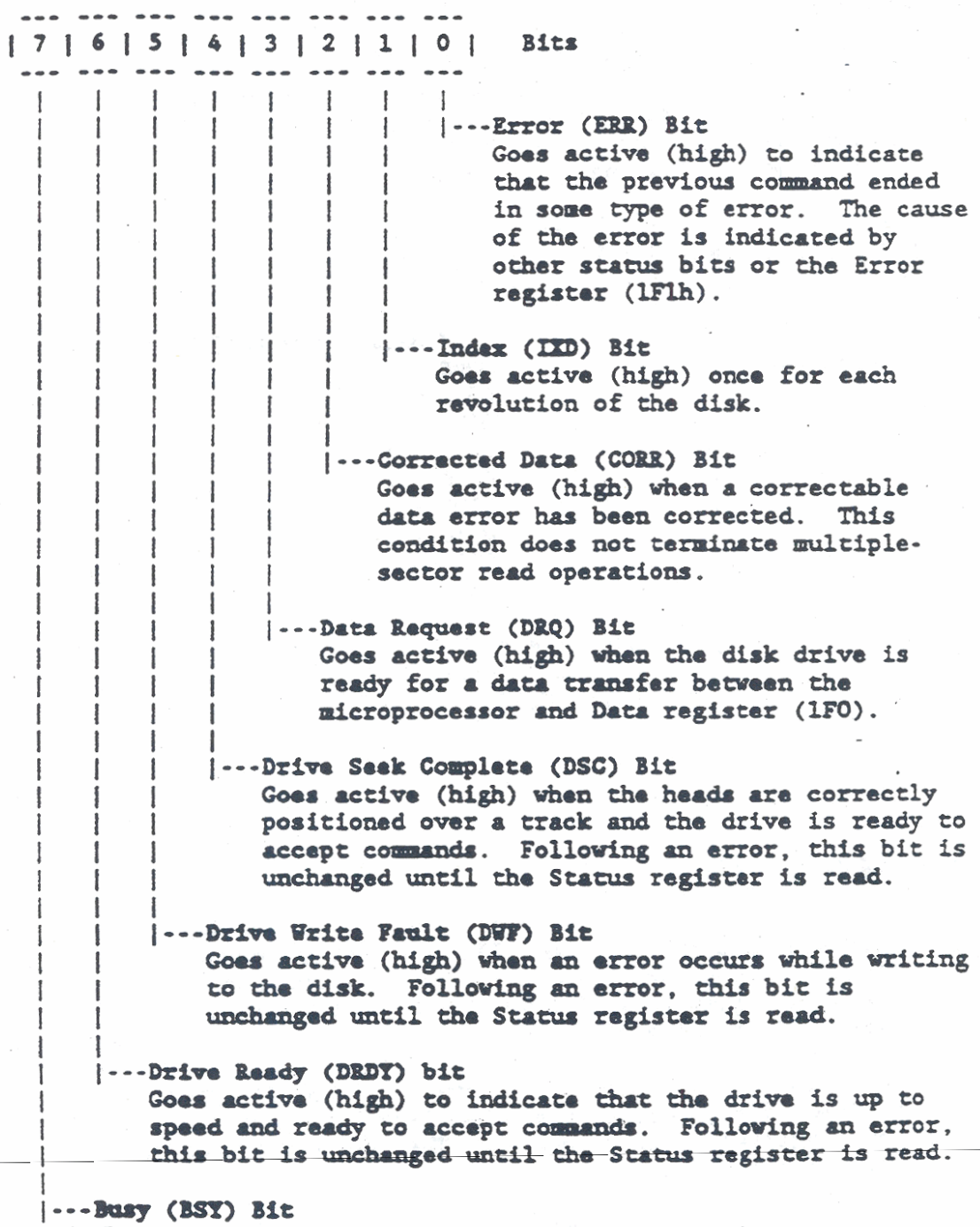
The SDH register (1F6h) is an 8-bit read/write register that contains the drive and head numbers. At the completion of each sector access, and at the completion of a sector access command, the SDH register contents are updated to indicate the currently selected head. The register bit definitions are as follows:



Status Register (1F7h)

The Status register (1F7h) is an 8-bit read-only register that contains disk drive and controller operation information. The register contents are updated at the completion of each command. The status register is cleared whenever it is read. If an interrupt is pending at the time the register is read, the pending interrupt is also cleared. If the Busy bit (bit 7) is active, all other bits of the register are invalid. The microprocessor is then inhibited from accessing the other Task File registers, and the Status register is read instead. The Busy conditions are described in subsequent paragraphs.

The Status register bit definitions are as follows:



Command Register (1F7h)

The Command register (1F7h) is an 8-bit write-only register that is used to pass commands from the microprocessor to the disk drive.

Command execution begins immediately after the Command register is written. The executable commands, their command codes in binary and hexadecimal (h), and the parameters used with each command are listed in Table 10-6. Each command is further described in the subsequent paragraphs under Command Descriptions.

Table 10-6. Compatible Hard Disk Drive Commands

Command Name	Command Code 7 6 5 4 3 2 1 0	Hex	Parameters Used SC	SN	CY	SDH
Recalibrate	0 0 0 1 x x x x	1x	n	n	n	d
Read Sector(s)	0 0 1 0 0 0 L r	20-23*	y	y	y	y
Write Sector(s)	0 0 1 1 0 0 L r	30-33*	y	y	y	y
Read Verify Sector(s)	0 1 0 0 0 0 0 r	40/41*	y	y	y	y
Format Track	0 1 0 1 0 0 0 0	50	n	n	y	y
Seek	0 1 1 1 x x x x	7x	n	n	y	y
Execute Drive Diag	1 0 0 1 0 0 0 0	90	n	n	n	d
Initialize Drive Params	1 0 0 1 0 0 0 1	91	y	n	n	y
Read Multiple	1 1 0 0 0 1 0 0	C4#	y	y	y	y
Write Multiple	1 1 0 0 0 1 0 1	C5#	y	y	y	y
Set Multiple Mode	1 1 0 0 0 1 1 0	C6#	y	n	n	d
Read Sector Buffer	1 1 1 0 0 1 0 0	E4	n	n	n	d
Write Sector Buffer	1 1 1 0 1 0 0 0	E8	n	n	n	d
Identify Drive	1 1 1 0 1 1 0 0	EC	n	n	n	d
Set Buffer Mode	1 1 1 0 1 1 1 1	EF#	n	n	n	d
Power Command(s)20/40 MB	1 1 1 0 p p p p	Ex	y	n	n	d

Where:

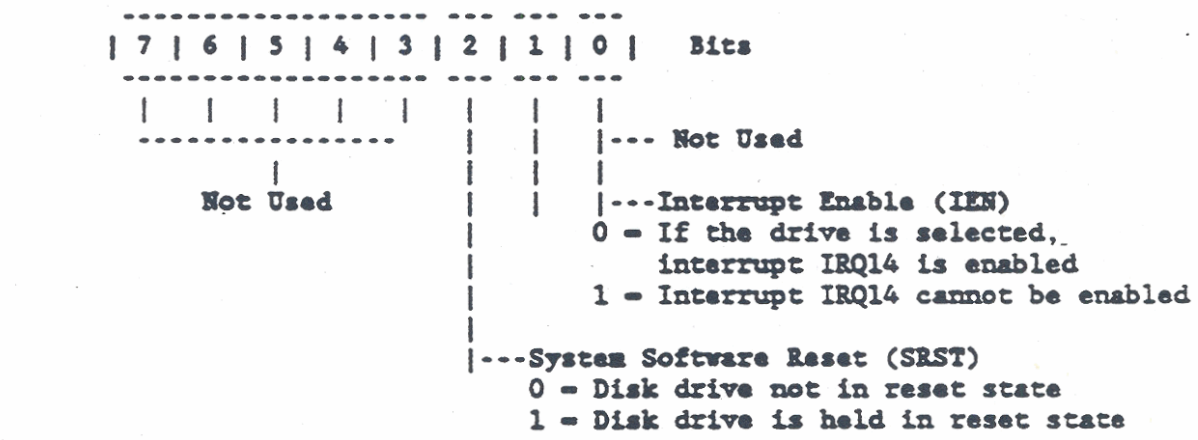
- * = with retries disabled
- # = These commands are not applicable to 20M byte drive.
- d = Register contains a valid drive parameter, but the head parameter is not valid. Refer to y.
- n = Register does not contain valid drive or head parameters for this command.
- y = Register contains valid drive and head parameters for this command.
- L = long bit. If L = 1 R/W long commands are executed. When L = 0 normal R/W commands are executed.
- r = retry bit. r = 0 enables the retry bit for ECC and data errors. r = 1 disables retries. If retries are disabled at the start of command, they are automatically enabled at the end of the command.
- p = Valid bit only for Power commands 94h through 99h.
- SC = Sector Count register (1F2h)
- SN = Sector Number register (1F3h)
- CY = Cylinder register (1F4h/1F5h)
- SDH = Drive/Head register (1F6h)
- x = Don't care, bit value is not significant.

Alternate Status Register (3F6h)

The Alternate Status Register (3F6h) is an 8-bit read-only register that provides the same status information as the Status register (1F7h). However, reading the Alternate Status register does not clear a pending interrupt. Refer to the Status register (1F7h) for definitions of the status bits.

Digital Output Register (3F6h)

The Digital Output register (3F6h) is an 8-bit write-only register in which only two bits are used to provide to provide interrupt and reset control. The two control bits and their operation are given in the following bit definitions:



Drive Address Register (3F7h)

The Drive Address Register (3F7h) is not used by the GRiDCase 1500 Series computers.

COMMAND DESCRIPTIONS

The compatible hard disk drive commands are issued and executed via the Command register (1F7, write-only). For each command to be completed the following operations are performed:

1. The microprocessor sends the selected parameters to the Task File interface registers (1F0 -1F7). The required parameters for each command are listed in Table 10-6.

2. The Interrupt Enable bit (-IEN) in the Digital Output register (3F6h, bit 1) is reset to "0" to enable interrupt request IRQ14.

3. A command is then loaded into the Command register and execution begins as soon as the command is loaded.

4. When command execution is complete, status information is returned to the microprocessor.

At the start of command execution, the Command register is checked for conditions that may cause an aborted command. When no error conditions are found, execution of the command is attempted. If any subsequent errors are found, the command is terminated at that point. The types of errors that can be returned during command execution are listed in Table 10-7. The same table lists the register address and bit position where the particular error is returned. The two registers that return command errors are the Error register (1F1h) and the Status register (1F7h). Table 10-8 lists the commands with their hexadecimal code and the mnemonics for the valid errors that can be returned for each command.

The Drive (DRV), Head (HEAD), Cylinder (CY), Sector Number (CN), and Sector Count (SC) are parameters that may need to be set before a command can be executed. Separate registers are used to set the Cylinder (1F4h/1F5h), Sector Number (1F3h), and Sector Count (1F2h). The Drive/Head register (1F6h) is used to select both the head and drive parameters. The drive is selected by resetting the Drive Select bit (bit 4) to "0" and by setting a jumper on the drive that designates its use as "master" or "slave." Since the GRiDCase 1500 Series computers use only one hard disk drive, the drive jumper is factory set to designate it as the "master" drive. Drive selection then refers to the Drive Select bit (1F6h, bit 4), which must be cleared to select the "master" drive.

Table 10-7. Types of Errors Reported During Command Execution

Returning Reg,Bit	Mnemonic	Description
1F1h,2	ABRT	Aborted Command
---	AMNF	Address Mark Not Found
1F1h,7	BBK	Bad Block Detected
1F7h,2	CORR	Corrected Data Error
1F7h,6	DRDY	Disk Drive Not Ready
1F7h,4	DSC	Drive Seek Complete Not Detected
1F7h,5	DWF	Disk Drive Write Fault
1F7h,0	ERR	Error Bit in Status Register
1F1h,4	IDNF	Requested ID Not Found
1F1h,1	TKO	Track Zero Not Found Error
1F1h,6	UNC	Uncorrectable Data Error

Table 10-8. Commands and Their Associated Valid Errors

Command Name	Code	Associated Valid Errors
Execute Drive Diagnostic	90h	Not Specified
Format Track	40h/41h	ABRT, DRDY, DSC, DWF, ERR, IDNF
Identify Drive	ECh	ABRT, ERR
Initialize Drive Params	91h	ABRT, ERR
Invalid Command	---	ABRT, ERR
Power Command(s)	Exh	Not Specified
Read Multiple	C4h	ABRT, BBK, CORR, DRDY, DSC, DWF, ERR, IDNF, UNC
Read Sector Buffer	E4h	ABRT, ERR
Read Sector(s)	22h/23h	ABRT, BBK, CORR, DRDY, DSC, DWF, ERR, IDNF, UNC
Read Verify Sector(s)	40h/41h	ABRT, BBK, CORR, DRDY, DSC, DWF, ERR, IDNF, UNC
Recalibrate	10h	ABRT, DRDY, DSC, DWF, ERR, TKO
Seek	70h	ABRT, DRDY, DSC, DWF, ERR, IDNF
Set Buffer Mode	EFh	ABRT, ERR
Set Multiple Mode	C6h	ABRT, ERR
Write Multiple	C5h	ABRT, BBK, DRDY, DSC, DWF, ERR, IDNF
Write Sector Buffer	E8h	ABRT, ERR
Write Sector(s)	32h/33h	ABRT, BBK, DRDY, DSC, DWF, ERR, IDNF

Recalibrate Command (10h)

The Recalibrate command (code 10h) moves the read/write heads to cylinder 0 (the outermost track). When the Recalibrate command is received, the disk drive sets the Busy (BSY) bit (Status register 17Fh, bit 7) and then executes a seek to cylinder 0. Following a successful seek, the disk drive updates the Status register, clears the BSY bit, and generates an interrupt request. With the command successfully completed the other Task File register contents are as follows:

Name	Addr	Contents
Error Register	(1F1h)	00h
Sector Count Register	(1F2h)	Unchanged
Sector Number Register	(1F3h)	Unchanged
Cylinder Low Register	(1F4h)	00h
Cylinder High Register	(1F5h)	00h
Drive/Head Register	(1F6h)	Unchanged

If the disk drive did not find cylinder 0, the Status register ERR bit is set (1F7h, bit 0) and the Error register TK0 bit is set (1F1h, bit 1). Also, the Error register ABRT bit is set (bit 2) if the command was aborted because the drive was not up to speed or the head was not on the selected track.

Read Sector(s) Command (22h/23h)

The Read Sector(s) command (code 22h or 23h) reads from 1 to 256 sectors as specified in the Sector Count resister (1F2h) and beginning at the sector specified in the Sector Number register (1F3h). A maximum count of 256 sectors is specified by a Sector Count register contents of 0. The Status register BSY bit is set (1F7h, bit 7) and command execution begins as soon as the command is loaded. To execute the command, if the drive is not already on the correct track, an implied seek operation is performed to seek to the desired track, and then the drive searches for the appropriate ID field. When the ID is read correctly, the data field is read into the Sector buffer, the Status register DRQ bit (3) is set, and an interrupt request is generated. Following successful completion of the command, the Task File register contents are as follows:

Name	Addr	Contents
Error Register	(1F1h)	00h
Sector Count Register	(1F2h)	00h
Sector Number Register	(1F3h)	Last Sector Read
Cylinder Low Register	(1F4h)	Last Sector Read
Cylinder High Register	(1F5h)	Last Sector Read
Drive/Head Register	(1F6h)	Last Sector Read

A Read Long command is executed by setting the long bit (bit 1) in the command code. With the long bit set, the command reads the 16-bit data words and also reads the four ECC bytes in the data field. During execution of a a Read Long command, the disk drive does not check the 8-bit ECC bytes to determine if there has been a data transfer error.

The 40M byte and 100M byte drives employ a 1:1 interleave and a larger Sector buffer to provide a read look-ahead capability. When read look-ahead is active, any read causes the 40M byte disk drive to read a minimum of 16 contiguous sectors of data and the 100M byte disk drive to read a minimum of 64 contiguous sectors of data. All subsequent reads operations then test to see if the requested data is already contained in the buffer before performing another read operation from the disk. If any command other than a read is issued subsequent to the first read command, the buffer is purged before the command is executed. The read look-ahead capability is enabled by the Set Buffer Mode command (code EFh), which is described in subsequent paragraphs.

When more than one sector (multiple sectors) is being read with the Read Sector(s) command, the Status register DRQ bit (bit 3) is set and the Sector buffer is filled at the completion of each sector. The microprocessor then reads the Sector buffer, which immediately sets the Status register BSY bit (bit 7) and resets the DRQ bit. The Text File register (cylinder, head, and sector) are then updated to point to the next sequential sector and the operation is repeated until all of the selected sectors are read.

If a Read Sector(s) command is attempted and bits 2 and 3 of the command code are not 0, the command is aborted and the Error register ABRT bit (bit 2) is set. A different error bit is set if an error occurs during a data transfer. The Status register Data Request bit (DRQ, bit 3) is always set at the end of a sector read regardless of the error condition. If an error occurs during a multiple sector read operation, the read terminates at the sector where the error occurred. The Task File registers can then be read to determine what error occurred and in what sector. The data read from the sectors is loaded into the Sector buffer regardless of whether the error is a correctable data error or a non-correctable data error. However, if the error was a correctable data error, the read operation is not terminated.

Write Sector(s) Command (32h/33h)

The Write Sector(s) command (code 32h or 33h) writes from 1 to 256 sectors as specified in the Sector Count resister (1F2h) and beginning at the sector specified in the Sector Number register (1F3h). A maximum count of 256 sectors is specified by a Sector Count register contents of 0. When the command is issued, the disk drive waits for the microprocessor to fill the Sector buffer with the data that will be written. No interrupt operation is required

to start the buffer fill operation. When the buffer is full, the disk drive set the Status register BSY bit (1F7h, bit 7) and begins command execution.

To execute the command, if the drive is not already on the correct track, an implied seek operation is performed to seek to the desired track, and then the drive searches for the appropriate ID field. When the ID is read correctly, the data that was previously loaded into the buffer is now written to the data field of the sector. Following successful completion of the command, the Task File register contents are as follows:

Name	Addr	Contents
Error Register	(1F1h)	00h
Sector Count Register	(1F2h)	00h
Sector Number Register	(1F3h)	Last Sector Read
Cylinder Low Register	(1F4h)	Last Sector Read
Cylinder High Register	(1F5h)	Last Sector Read
Drive/Head Register	(1F6h)	Last Sector Read

A Write Long command is executed by setting the long bit (bit 1) in the command code. With the long bit set, the command writes the 16-bit data words and also writes the four ECC bytes directly from the Sector buffer to the data field on the disk. During execution of a Write Long command, the disk drive does not generate the four 8-bit ECC bytes.

When more than one sector (multiple sectors) is being written with the Write Sector(s) command, the Status register DRQ bit (1F7h, bit 3) is set and an interrupt request is generated each time the buffer is ready to be filled. Once the buffer is full, the DRQ bit is reset and the Status register BSY bit (1F7h, bit 7) is set. The Text File registers (cylinder, head, and sector) are then updated to point to the next sequential sector and the operation is repeated until all of the selected sectors are written.

The 40M byte and 100M byte drives employ a 1:1 interleave capability, which results in slower operations for sequential single sector data transfers because of missed interleaves. However, multiple sector operations will be much faster since all the sectors except the last one are transferred in a sequential block . Then, while the last sector is written, the next block of data is received. After the last block of data is received, the disk drive sets the Status register BSY bit (bit 7) until it is done writing the data.

If a Write Sector(s) command is attempted and bits 2 and 3 of the command code are not 0, the command is aborted and the Error register ABRT bit (bit 2) is set. A different error bit is set if an error occurs during a data transfer. If an error occurs during a multiple sector write operation, the write terminates at the sector where the error occurred. The Task File registers can then be read to determine what error occurred and in what sector.

Read Verify Sector(s) Command (40h/41h)

The Read Verify Sectors command (code 40h or 41h) operates in the same way as he Read Sectors command except that no data transfer occurs. This command reads and verifies from 1 to 256 sectors as specified in the Sector Count resister (1F2h) and beginning at the sector specified in the Sector Number register (1F3h). A maximum count of 256 sectors is specified by a Sector Count register contents of 0. After each sector is verified, the Task File registers are update, but no data request or interrupt request is generated. After all the selected sectors are verified, an interrupt request is generated to indicate that verification is complete. The Sector Count register indicates the number of registers verified (a count of 00 indicates 256 sectors were verified). For the 40M byte and 100M byte disk drives, the read look ahead capability is enabled for the Read Verify Sector(s) command.

Format Track Command (50h)

The Format Track command (code 50h) formats the track specified by the Cylinder registers (1F4h/1F5h) and the Drive/Head register (1F6h). After the command is issued, the disk drive waits for the microprocessor to fill the Sector buffer with the format data. When the buffer is full, the disk drive resets the Status register Data Request (DRQ) bit (1F7h, bit 3), sets the Busy (BSY) bit (1F7h, bit 7), and begins the command execution.

To execute the command, if the drive is not already on the correct track, an implied seek operation is performed to seek to the desired track. When the specified track is found, formatting begins and the data that was previously loaded into the Sector buffer is now written to the disk. During the formatting operation, each sector is designated as "good" or "bad," and for a sector designated as "bad," an alternate sector can be "assigned/unassigned." Following successful completion of the command, the disk drive resets the BSY bit and requests an interrupt.

The format information must contain two bytes of data for each sector on the specified track. The least significant byte (LSB) is the Descriptor Byte, which is described in the following paragraph. The most significant byte (MSB) contains the sector number. When the data bytes are written to the Sector buffer, the LSB is written first and followed immediately by the MSB. After the information is written to all of the sectors on the logical track, the remaining bytes of each sector should be filled with zeros. The sectors can be arranged in any order, but each sector must contain the two bytes of data as previously described, and the format information must be sent to the Sector buffer as the first 34 or 52 bytes.

The Descriptor Byte is the LSB of the two bytes written to each sector during the format operation, and is used to designate the sector status as follows:

1. A value of 00h specifies a "good" sector.

2. A value of 80h specifies a "bad" sector.

3. A value of 40h causes the drive to logically "assign" the sector to an alternate sector in its set of spares.

4. A value of 20h causes the drive to logically "unassign" the alternate sector and recover the original sector.

If a sector was previously formatted as a "bad" sector, and is then reformatted as "good," the disk drive will attempt to format the sector as "good." However, if a previously "assigned" sector is "unassigned," the spare sector that was assigned is lost. This provides the capability to replace a bad sector, but following the "unassign" operation, the spare sector is not recovered.

CAUTION

Do not perform a diagnostic that "assigns" and then "unassigns" sectors across the disk. This type of diagnostic will use up the spare sectors so that none are available as spares for bad sectors.

If a Format Track command is attempted and bits 0 through 3 of the command code are not 0, the command is aborted and the Error register ABRT bit (bit 2) is set. When a zero sector number, or any number greater than the maximum for the mode, is sent, whether or not it is the logical number, the Error register Requested ID Not Found (IDNF) bit (1F1h, bit 4) is set. In the event that there are multiple bad sector numbers, the smallest illegal sector number is stored in the Sector Number register (1F3h). However, before the error is indicated, all legal sectors are formatted as specified by the descriptor byte.

Seek Command (70h)

The Seek command (code 70h) executes a search to the track specified by the Cylinder registers (1F4h/1F5h) and the Drive/Head register (1F6h). After the command is issued, the Seek command operations are executed whether or not the drive has been formatted. To execute the Seek command, the disk drive sets the Status register Busy (BSY) bit (1F7h, bit 7), begins the Seek operation, resets BSY, and requests an interrupt. The disk drive

does not wait for the seek to be completed before requesting an interrupt. When the seek operation is complete the Status register Drive Seek Complete (DSC) bit (1F7h, bit 4) is set.

If a new command is issued to the disk drive while a Seek command is being executed, the disk drive waits, with BSY active, for the DSC bit to become active before executing the new command. The validity of the sector number or head value is not checked as part of the Seek command operation. If the cylinder value is not valid, the Seek command is not executed, but the DSC bit is set. The ERR bit (1F7h, bit 0) is not set.

Execute Drive Diagnostic Command (90h)

The Execute Drive Diagnostic command (code 90h) initiates and internal diagnostic test that is implemented and controlled by the disk drive. To execute the command, the disk drive sets the Status register Busy (BSY) bit (1F7h, bit 7), performs the diagnostic operations, saves the test results, resets BSY, and then requests an interrupt. The diagnostic test results are stored in the Error register (1F1h) so that the register contents are viewed as a result byte rather than as discrete error flags. When the diagnostic test is complete, all of the Task File registers are set to their default values, except the Error register. The Error register is set to indicate the diagnostic test results as follows:

Register Contents	Description
01h	No diagnostic error detected
03h	Sector Buffer error detected
8xh	Reserved

Initialize Drive Parameters Command (91h)

The Initialize Drive Parameters command (code 91h) allows the microprocessor to set the head switch and cylinder increment points for multiple sector operations. To execute the command, the disk drive sets the Status register Busy (BSY) bit (1F7h, bit 7), saves the current parameters, activates the translate mode, resets BSY, and requests an interrupt. The disk drive can activate the translate mode only if the Sector Count register (1F2h) contains 17h when the command is issued. In the translate mode, the logical value in the Sector Number register (1F3h), Cylinder registers (1F4h/1F5h), and Drive/Head register (1F6h) are translated to their default values.

The default values for the Task File registers (1F2h through 1F6h) depend upon the capacity of the drive being used as follows:

Drive Capacity	Sectors Per Track	Number of Heads	Number of Cylinders
20M Bytes	17	4	615
40M Bytes	17	4 (NOTE)	976
100M Bytes	33	8	776

NOTE: The 40M byte drive has 5 heads when used in translate mode.

To specify the maximum number of heads for this command, always write one less than the maximum. For example, for a 40M byte drive, specify 3 as the maximum number of heads. The maximum number of sectors is specified as the actual number. For example, 26 is specified for a maximum of 26 sectors. Cylinder and head increments on subsequent commands occur after an access of the maximum cylinder and head values specified by this command.

The sector, head, and cylinder values in the Task File registers are not checked for validity by this command. Therefore, if they are not valid, no error is reported until an illegal access is made by another command.

For 100M byte disk drives, this command has been changed for use in systems with no exact device type. For these drives only, any head and sector values are accepted. The disk drive then computes the maximum cylinder from the values provided for the head and sector as follows:

Max Cylinder = 204,864/(head value x sector value)

Where: 204,864 is the total sectors on the disk drive.

CAUTION

The maximum number of cylinders expected by the device type cannot exceed the number possible for the drive. Specifying too many cylinders can cause errors during system boot operations.

Read Multiple Command (C4h)

The Read Multiple command (code C4h) is used for the 40M byte and 100M byte disk drives only. Operation of the Read Multiple command is similar to the Read Sector(s) command previously described, but it allows several sectors to be transferred as a block to the microprocessor without interveaning interrupts. For the Read Multiple command, the Status register Data Request (DRQ) bit (1F7h, bit 3) is set once at the start of the block count on each sector.

Long transfers, with ECC bytes, are not allowed. The number of sectors to be transferred as a block is specified by the Set Multiple Mode command (code C6h), which must be executed first and is described in subsequent paragraphs.

To execute the Read Multiple command, the Sector Count register (1F2h) must contain the number of sectors requested, not the number of blocks or the block count. If the number of sectors requested is not evenly divisible by the block count, this command transfers as many full blocks as allowed, and then transfers a partial block with the remaining sectors. The partial block transfer will be for "n" sectors, where:

n = (sector count) modulo (block count)

Interrupts are requested when the DRQ bit is set at the beginning of each block or partial block. The read-look ahead operations are not active with the Read Multiple command.

An Aborted command (ABRT) error (1F1h, bit 2) is generated if execution of the Read Multiple command is attempted before the Set Multiple Mode command (code C6h) is executed, or when the Read Multiple command has been disabled.

Disk errors that occur during execution of a Read Multiple command are reported at the beginning of a block or partial block transfer. If a disk error is reported, the DRQ bit is still set and the data transfer takes place as it normally would, including transfers of corrupt data. However, subsequent blocks or partial blocks of data are transferred only if the error was a correctable data error. All other errors cause the command to stop execution after transfer of the block that contained the error.

Write Multiple Command (C5h)

The Write Multiple command (code C5h) is used for the 40M byte and 100M byte disk drives only. Operation of the Write Multiple command is similar to the Write Sector(s) command previously described, except that the Status register Busy (BSY) bit (1F7h, bit 7) is set immediately when the command is issued. Also, the data transfers occur in multiple sector blocks and the long bit in the command code is not valid. Several sectors are transferred as a block without interleaving interrupts and the Status register Data Request (DRQ) bit (1F7h, bit 3) is set at the start of each block transfer, not sector. An interrupt request is not generated prior to the first block transfer, but is generated when the DRQ bit is set for all subsequent block or partial block transfers. The number of sectors to be transferred as a block is specified by the Set Multiple Mode command (code C6h), which must be executed first and is described in subsequent paragraphs.

To execute the Write Multiple command, the Sector Count register (1F2h) must contain the number of sectors requested, not the number of blocks or the block count. If the number of sectors requested is not evenly divisible by the block count, this command transfers as many full blocks as allowed, and then transfers a partial block with the remaining sectors. The partial block transfer will be for "n" sectors, where:

n = (sector count) modulo (block count)

An Aborted command (ABRT) error (1F1h, bit 2) is generated if execution of the Write Multiple command is attempted before the Set Multiple Mode command (code C6h) is executed, or when the Write Multiple command has been disabled.

Disk errors that occur during execution of a Write Multiple command are reported after the attempted disk write. The write operation terminates at the sector where the error occurred even if it is in the middle of a block. Subsequent blocks are not transferred.

Set Multiple Mode Command (C6h)

The Set Multiple Mode command (code C6h) is used for the 40M byte and 100M byte disk drives only. This command establishes the block count for the Read Multiple and Write Multiple commands, and must be executed before either of the multiple commands are executed. To enable the Set Multiple Mode command, the Sector Count register (1F2h) is loaded with the number of sectors to be contained in the block. The number of sectors, or block sizes, supported by this command are 1, 2, 4, 8, 16, 32, and 64.

When the Set Multiple Mode command is executed, the disk drive sets the Status register Busy (BSY) bit (1F7h, bit 7), and then reads the Sector Count register (1F2h) contents. If the Sector Count register contents is one of the supported values, the value is stored for comparison during subsequent execution of the Read Multiple and Write Multiple commands. Also, the Read Multiple and Write Multiple commands are enabled. The disk drive then resets the BSY bit and requests an interrupt.

If the Sector Count register (1F2h) contains an unsupported value when the Set Multiple Mode command is executed, the Error register Aborted (ABRT) command bit (1F1h, bit 3) is set and the Read Multiple and Write Multiple commands are disabled. The multiple commands are also disabled if the Sector Count register contains a 0 when the Set Multiple Mode command is executed.

Following power turn-on and hardware or software resets, the Read Multiple and Write Multiple commands are disabled. This Set Multiple Mode command must be successfully executed in order to enable the Read Multiple and Write Multiple commands.

Read Buffer Command (E4h)

The Read Buffer command (code E4h) allows the microprocessor to read up to 512 bytes of data from the Sector buffer in the disk drive. All data transferred between the microprocessor and disk drive is passed through the Sector buffer, so the buffer always contains the last 512 bytes that were written to the buffer. To execute the Read Buffer command, the disk drive sets the Status register Busy (BSY) bit (1F7h, bit 7), sets the Sector buffer for a read operation, sets the Status Register Data Request (DRQ) bit (1F7, bit 3), resets the BSY bit, and then requests an interrupt. The microprocessor then controls the data read operation.

Write Buffer Command (E8h)

The Write buffer command (code E8h) allows the microprocessor to write up to 512 bytes of data in any pattern into the Sector buffer of the disk drive. All data transferred between the microprocessor and disk drive is passed through the Sector buffer, so the buffer always contains the last 512 bytes that were written to the buffer. Operation of the Sector buffer can be checked by writing data patterns the buffer to and then reading the same data patterns from the buffer. To execute the Write Buffer command, the disk drive sets up the Sector buffer for a write operation and then sets the Status register Data Request (DRQ) bit (1F7h, bit 3). The microprocessor then controls the writing of data to the buffer.

Identify Drive Command (ECh)

The Identify Drive command (code ECh) permits the microprocessor to read specific disk drive parameter information through the Sector buffer. To execute the command, the disk drive sets the Status register Busy (BSY) bit (1F7h, bit 7), stores its parameter information in the Sector buffer, sets the Status Register Data Request (DRQ) bit (1F7h, bit 3), and generates an interrupt. The microprocessor then reads the parameter information from the Sector buffer. In the Sector buffer, the parameter information is arranged sequentially in 16-bit words with the reserved bits and words set to all zeros. The parameter words and its contents are listed in Table 10-9.

Table 10-9. Identify Drive Parameter Words

Word	Contents
00h	A constant value of 0A5Ah
01h	Number of Fixed Cylinders
02h	Number of Removable Cylinders
03h	Number of Heads
04h	Number of Unformatted Bytes per Physical Track
05h	Number of Unformatted Bytes per Sector
06h	Number of Physical Sectors per Track
07h	Number of Bytes in Intersector Gaps
08h	Number of Bytes in Sync Fields
09h	0000h
10h-19h	Serial Number
20h	Controller Type: 20M byte drive = 0001 single-ported sector buffer 2:1 40M byte drive = 0003 dual-ported multiple Sector buffer with read look-ahead capability 100M byte drive = Same as 40M byte drive
21h	Sector buffer Size in 512 byte increments
22h	Number of ECC bytes passed for Read Long and Write Long Commands
23h-26h	Controller Firmware Revision Level
27h-46h	Model Number
47h	Number of Sectors/Interrupt Flag 0 = Not Supported 1 or > = Operation Supported
48h	Double Word Transfer Flag 0 = Disabled 1 = Enabled
49h	Assign Alternate Flag 0 = Disabled 1 = Enabled Used for 40M byte and 100M byte drives only. This word is reserved for 20M byte drives.
50h-255h	Reserved

Set Buffer Mode Command (EFh)

The Set Buffer Mode command (code EFh) is used for the 40M byte and 100M byte disk drives only. This command enables or disables the read look-ahead capability for the Read Sector(s) command (22h/23h). Before issuing the Set Buffer Mode Command, the Write Precompensation register (1F1h) is loaded with AAh or 55h. The value AAh enables the read look-ahead capability and the value 55h disables the capability. When the Set Buffer Mode command is subsequently issued, the disk drive sets the Status register Busy

(BSY) bit (1F7h, bit 7), reads the contents of the Write Precompensation register, and sets the appropriate mode for the read look-ahead capability. The disk drive then resets the BSY bit and requests an interrupt.

At system power turn-on and following a hardware or software reset, the read look-ahead capability is enabled. If the Set Mode command is issued with any value other than AAh or 55h in the Write Precompensation register, the Error register Aborted Command (ABRT) bit (1F1h, bit 2) is set.

Power Commands (Exh)

The 20M byte drive and the 40M byte drive both use power commands to control the current disk drive operating mode. The five disk drive operating modes are described in Table 10-10.

Table 10-10. Disk Drive Operating Modes

No.	Operating Mode	Description
1	Read/Write	This mode occurs when data is being written to or read from the disk
2	Seek	This mode occurs while the heads are searching for a track and the actuator arm is in motion.
3	Idle	In idle mode, the drive motor is up to speed and the Status register Drive Ready (DRDY) bit (1F7h, bit 6) is set. However, the drive is not reading, writing, or seeking and the heads are positioned over the last track that was accessed. Idle mode is the default condition following initial power turn-ON.
4	Standby	In standby mode, the disk drive motor is stopped, the heads are parked at track 0, and the logic circuits, except for interface control, are turned OFF. Also, the Status register Drive Seek Complete (DSC) and Drive Ready (DRDY) bits (1F7h, bits 4 and 6, respectively) are set. The disk drive enters standby mode if a programmed time-out occurs after the last disk access. The disk drive leaves the standby mode when it receives a command that requires disk access and if it receives a disk spin-up command.

Table 10-10. Disk Drive Operating Modes (Continued)

No.	Operating Mode	Description
5	Sleep	The sleep mode is similar to standby mode except that the interface control logic is also turned OFF. The disk drive enters the sleep mode when it receives a sleep command. To leave the sleep mode, the disk drive must receive a hardware reset. When reset, the drive leaves sleep mode and enters standby mode. It does not reenter the mode it was in when the sleep command was issued.

To control the disk drive operating modes the 20M and 40M byte drives uses command codes E0h-E3h, E5h, and E6h. The following paragraphs provide the applicable command code, command name, and a description of operation power command.

All of the Power commands except E6h execute when issued and return an interrupt request after the spin up or spin down sequence is initiated. Returning the interrupt request does not indicated that the command execution is complete, except when entering the sleep mode. When entering sleep mode, an interrupt request is returned after the drive has spun down and stopped.

While in idle mode, if a spin-up command is issued when the drive is already spinning, the spin-up sequence is not initiated. In standby mode, if a spin-down command is issued when the drive is not spinning, the spin-down sequence is not initiated.

When the Auto Power-OFF feature is enabled, the value in the Sector Count register (1F2h) specifies the number of 5-second increments used as the timeout value. If the drive does not receive a command within the time limit specified by the timeout value, the drive automatically enters the standby mode. The default and minimum value for the Sector Count register is 0Ch (12 x 5 seconds) to provide a timeout value of 60 seconds. The maximum allowable value is DCh (220 x 5 seconds) to provide a timeout value of 1100 seconds or 18.3 minutes. If the Sector Count register contains the value 00h, the Auto Power-OFF feature cannot be enabled. Table 10-11 lists the disk drive Power commands.

NOTE: For low-power 20M byte and 40M byte compatible hard disk drives, a spin-down mode is supported by the ROM-BIOS at AX = E449h. Refer to the Hard Disk Drive Spin-Down Mode in Chapter 3

Table 10-11. Disk Drive Power Commands

Command Code	Command Name	Operation
E0h	Standby Mode	The drive immediately enters the standby mode.
E1h	Idle Mode	The drive immediately enters the idle mode.
E2h	Standby Mode with Auto Power-OFF	The drive immediately enters the standby mode and enables Auto Power-OFF if the Sector Count register contains a non-zero value.
E3h	Idle Mode with Auto Power-OFF	The drive immediately enters the idle mode and enables Auto Power-OFF if the Sector Count register contains a non-zero value.
E5h	Mode Test	Inserts the value FFh in the Sector Count register (1F2h) if the drive is in the idle mode. The value 00h is inserted in the Sector Count register if the drive is in standby mode.
E6h	Sleep Mode	The drive immediately enters the sleep mode.

RESET, BUSY, AND RETRY OPERATIONS

Reset, Busy, and Retry are disk drive operations that occur as a result of other operations. The Reset, Busy, and Retry operations and their affect on disk drive operation are described in the following paragraphs.

Reset Operation

The Reset operation sets the Status register Busy (BSY) bit (1F7h, bit 7), and initializes the disk drive to a known state in preparation for receiving commands and transferring data. The Reset operation is generated in two ways as follows:

1. Hardware Reset. The hardware reset is generated by either the microprocessor or the disk drive. The microprocessor generates a reset by sending an active low reset signal to the

disk drive during the power turn-ON operation. The disk drive generates a reset via its power sense circuits. In either case, the reset condition is removed to allow normal operation when microprocessor and the drive acknowledge that power to the drive is within the normal range.

2. Software Reset — The software reset is generated by setting the System Software Reset bit in the Digital Output register (3F6h, bit 2). This reset condition is removed only by resetting System Software Reset bit.
NOTE: Software reset is not implemented in all versions of the ROM-BIOS.

When the reset condition is subsequently removed, the BSY bit remains set while the disk drive is enabled and the drive performs the hardware initialization. The hardware initialization includes clearing any previously programmed parameters, resetting default conditions, loading the Task File registers with their initial values, and then resetting the BSY bit. No interrupt is generated.

The initial values for the Task File registers are as follows:

Name	Addr	Contents
Error Register	(1F1h)	01h
Sector Count Register	(1F2h)	01h
Sector Number Register	(1F3h)	01h
Cylinder Low Register	(1F4h)	00h
Cylinder High Register	(1F5h)	00h
Drive/Head Register	(1F6h)	00h

Busy Operation

A Busy operation is indicated by the set state of the Status register Busy (BSY) bit (1F7h, bit 7). The Status register Busy bit is activated by the following conditions:

1. A system reset at power ON or the software reset bit is set in the Digital Output register (3F6h, bit 2).

2. The microprocessor initiates one of the following commands (refer to Command register 1F7h):

   a. Read Sector(s)
   b. Read Long
   c. Read Sector Buffer
   d. Seek
   e. Recalibrate

d. Initialize Drive Parameters
e. Read Verify Sector
f. Identify Drive
g. Execute Drive Diagnostic

3. Whenever 512 bytes of data are transferred by execution of the following commands (refer to Command register 1F7h):

   a. Write Sector(s)
   b. Format Track
   c. Write Sector Buffer

4. Whenever 512 bytes of data and seven ECC bytes are transferred by execution of a Write Long Command.

Retry Operations

For commands that read data, execution requires a plan for handling events where data is missed on the first attempt. When these events are detected, a preprogrammed sequence of operations is performed in an effort to capture the missed data, correct the data using the ECC, and to report the event. The preprogrammed sequence of operations, or algorithm, varies depending upon the disk drive characteristics. In the GRiDCase 1500 Series computers with compatible hard disk drives, the algorithm used for the 20M byte disk drive differs from the algorithm used for the 40M byte and 100M byte disk drives. The following paragraphs describe both algorithms.

Retries with 20M Byte Disk Drive

With the 20M byte compatible hard disk drive, retries are handled on a global basis. Operations where errors are detected in the ID field are retried 10 times and may not be disabled from the interface. Operations where errors are detected in the data field may be disabled from the interface and are retried in the following sequence up to 128 times.

1. A second read operation is attempted on the same track.

2. A third read is attempted with a servo offset.

3. A fourth read is attempted with a servo offset in the opposite direction.

4. The ECC is applied in an attempt to correct the data. The ECC is applied with offsets in the opposite directions on successive tries. The offsets rotate between +-60 microinches and +-120 microinches.

If any of the preceding steps are successful, the retry operations are terminated, the data is sent to its destination, and the Status register Corrected Data (CORR) bit (1F7h, bit 2) is set. The CORR bit provides a flag so that corrected data can be further check for errors.

When the data cannot be corrected, retry operations are terminated, the command is aborted, and the Error register Uncorrectable Data Error (UNC) bit (1F1h, bit 6) is set. Additional errors may also be reported depending upon the cause (refer to Table 10-8).

Retries with 40M Byte and 100M Byte Disk Drives

The 40M byte and 100M byte compatible hard disk drives provide separate retry operation for data fields and header fields. Both retry operations are described in the following paragraphs.

Data Field Retries

With the 40M byte and 100M byte hard disk drives, errors detected in the data field during read operations initiate the following retry operations:

1. The data read operation is retried three times.
2. On the third retry, ECC is applied in an attempt to correct the data.
3. The data read operation is retried with a +65 microinch offset.
4. The ECC is applied in an attempt to correct data from the offset read.
5. The data read operation is retried.
6. The ECC is applied in an attempt to correct data from the previous read.
7. The data read operation is retried with a -65 microinch offset.
8. The ECC is applied in an attempt to correct data from the offset read.
9. The data read operation is retried six times.

10. The preceding steps are repeated up to eight times for a total of 128 retries. If 17 milliseconds are allowed for each read retry, and 75 milliseconds are allowed for each ECC operation, the total retry time is 4 seconds. Except for disabling retries, retry count = 0, the number of retries (128) cannot be changed.

If any of the preceding steps are successful, the retry operations are terminated, the data is sent to its destination, and the Status register Corrected Data (CORR) bit (1F7h, bit 2) is set. The CORR bit provides a flag so that corrected data can be further checked for errors.

When the data cannot be corrected, retry operations are terminated, the command is aborted, and the Error register Uncorrectable Data Error (UNC) bit (1F1h, bit 6) is set. Additional errors may also be reported depending upon the cause (refer to Table 10-8).

When the Read/Write heads are switched or a seek is completed, and the heads are less than 200 microinches from the center of the track, the drive attempts an offtrack read. If the attempt is successful, 17 milliseconds of latency are saved and the drive exceeds its seek performance specification. When the attempt is not successful, the drive reads the selected sector on the next pass as in a normal read operation and the seek performance specification is met.

Header Field Retries

With the 40M byte and 100M byte hard disk drives, errors detected in the header field during read operations initiate the following retry operations:

1. The header read operation is retried 20 times.

2. If any retry is successful, the header retry counter is reset and the data field is read.

3. If the retries are not successful, the read command is aborted and the applicable error is returned (refer to Table 10-8).

   The total time required for 20 retries is 0.34 seconds. The header reties cannot be disabled from the interface and the header retry count cannot be changed.

CHAPTER 11: PARALLEL PORT SUBSYSTEM

The Parallel Port Subsystem is functionally equivalent to the IBM AT printer adapter and also includes a parallel I/O capability. The parallel port interface is designed to attach to parallel Centronics-type printers. Additionally, a second group of registers allows input via the parallel port connector so that the interface can be used as a general purpose, parallel input/output port.

The Parallel Port controller is a custom gate array device with GRiD System Part Number 300613. The controller consists of nine registers and their control logic. Descriptions of the registers are provided in subsequent paragraphs of this chapter. A block diagram of the Parallel Port Subsystem is provided in Figure 11-1.

PARALLEL PORT ROM-BIOS SERVICE ROUTINE

The GRiDCase 1500 Series Computer ROM-BIOS provides the following interrupt service routine functions to interface to parallel printer.

Interrupt (Hex)	Function No. (AH reg)	Description
17	00	Print a character
17	01	Initialize the printer
17	02	Read printer status

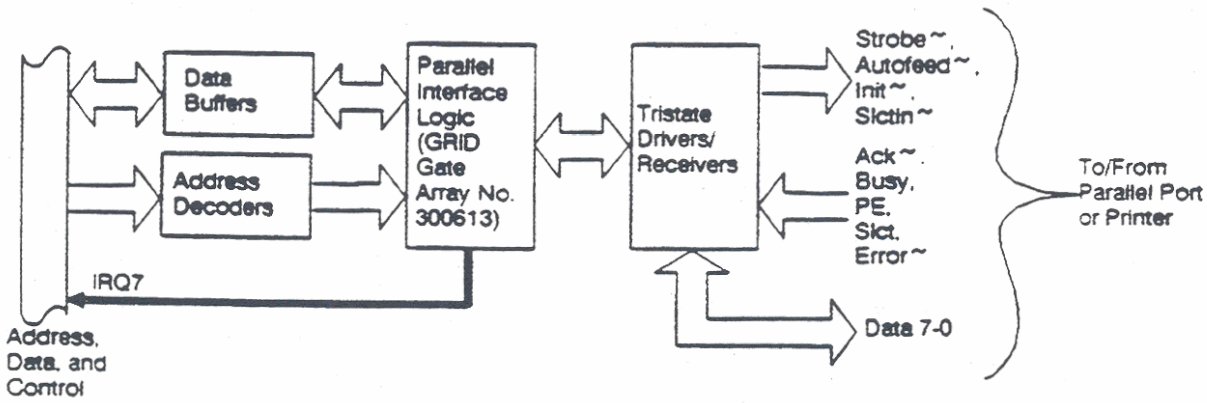


Figure 11-1. Parallel Port Subsystem Block Diagram

Parallel Printer Port Service Routine Description

The 80286/80386 Microprocessor register inputs and outputs for the parallel printer port interrupt service routine functions are listed in the following paragraphs. In each of the functions, only register AH is modified; all other registers are unchanged. Register AH determines the function that is invoked. The register contents are specified in hexadecimal (h).

Register DX specifies the printer to be used (0, 1, or 2). In the GRiDCase 1500 Series computers, I/O registers are provided to support printer 0 only. The ROM-BIOS can handle additional printers if the appropriate I/O registers are provided externally.

Register AH is also used to return the printer status information. Refer to the Printer Control and Status register for a description of the status information that is returned.

AH = 00h Print Character
AL = Character to print
DX = Printer to use (0)
On Exit:
AH = Printer status

Bit	Indication When Set to "1."
0	Time out
1,2	Not used
3	I/O error
4	Selected
5	Out of paper
6	Acknowledge
7	Not busy

AH = 01h Initialize Printer Port
DX = Printer Port to initialize (Always 0)
On Exit:
AH = Printer status
Same as function 00h.

AH = 02h Read Printer Status
DX = Printer whose status is returned (0)
On Exit:
AH = Printer status
Same as function 00h.

PARALLEL PORT SUBSYSTEM REGISTERS

The parallel port controller consists of nine registers and their control logic. One group of five registers is used for printer operation and system status checks. Another group of three

registers supports a parallel input/output port interface. One separate register selects between the printer registers and the parallel port registers. All of the registers are accessed through I/O addresses.

The printer operation and system status registers are accessed through I/O addresses 378h through 37Fh. The parallel I/O port registers are accessed through I/O addresses FF8h through FFDh. One of the parallel I/O registers (port B) has separate addresses for read and write operations.

The I/O register at address 423h is used to select either the printer registers or the parallel I/O registers. Table 11-1 lists the Parallel Port Subsystem registers and their I/O addresses. Descriptions of the registers are provided in the following paragraphs.

NOTE: Throughout this chapter, I/O register addresses and contents are given in hexadecimal (h).

Table 11-1. Parallel Port Subsystem Registers

Address 423h	I/O Address	I/O	Register Name
0	378h	I	Read Printer Data
0	379h	I	Read Status Register B
0	37Ah	I	Read Status Register C
0	37Bh	-	Undefined
0	37Ch	O	Write Printer Data
0	37Dh	-	No Operation
0	37Eh	O	Printer Control Register
0	37Fh	-	Undefined
1	FF8h	I	Read Port A
1	FF9h	I	Read Port B
1	FFAh	I	Read Port C
1	FFBh	-	Undefined
1	FFCh	-	Not Used
1	FFDh	O	Write Port B

Printer Registers (378h-37Fh)

The printer registers are used to control operation of an external printer that is connected to the computer through the parallel port connector on the rear connector panel. Three registers are used to read data and status information and two registers are used to write data and control information. The five printer registers are all 8-bit registers, which are accessed only when the register at address 423h is set to "0."

To send data or control information to the printer, the respective data or control register (37Ch or 37Eh) is loaded with an 8-bit byte and a strobe signal is activated. The printer status registers are then monitored to determine the response from the printer. When the printer acknowledges that the data or control byte was received, a hardware interrupt request (IR7) is generated. The hardware interrupt initializes the ROM-BIOS interrupt service routine at interrupt 0Fh. The interrupt service routine completes the operations required to send data to the printer.

The following paragraphs provide the bit definitions for each of the printer registers.

NOTE: A tilde (~) symbol following a bit name indicates active low (low-true) logic.

Read and Write Printer Data Registers (378h and 37Ch)

The Parallel Port controller uses separate registers to read and write data. The registers are located at I/O addresses 378h and 37Ch, respectively. These 8-bit buffers provide an interface between the microprocessor data bus and the parallel printer port. Bits 7-0 of the data registers correspond to data bits DATA7-DATA0 of the microprocessor data bus.

Printer Control Register (37Eh)

The Printer Control register is located at I/O address 37Eh. This 8-bit buffer provides a write only interface between the microprocessor and the parallel printer port. Bit definitions for the Printer Control register are as follows:

```
----------- -------------------
| 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |   Bits
----------- -------------------
  |   |   |   |   |   |   |   |
  ---------   |   |   |   |   |--- STROBE
      |       |   |   |   |        Enables valid data onto the
   Not Used   |   |   |   |        microprocessor data bus.
              |   |   |   |-- AFDXT
              |   |   |       Autofeeds one line of paper after
              |   |   |       printing has stopped.
              |   |   |-- INIT~
              |   |       Initializes the printer.
              |   |-- SLCTIN~
              |       Allows printer to accept data.
              |-- IRQEN
                  Enables Interrupt Request IRQ7.
```

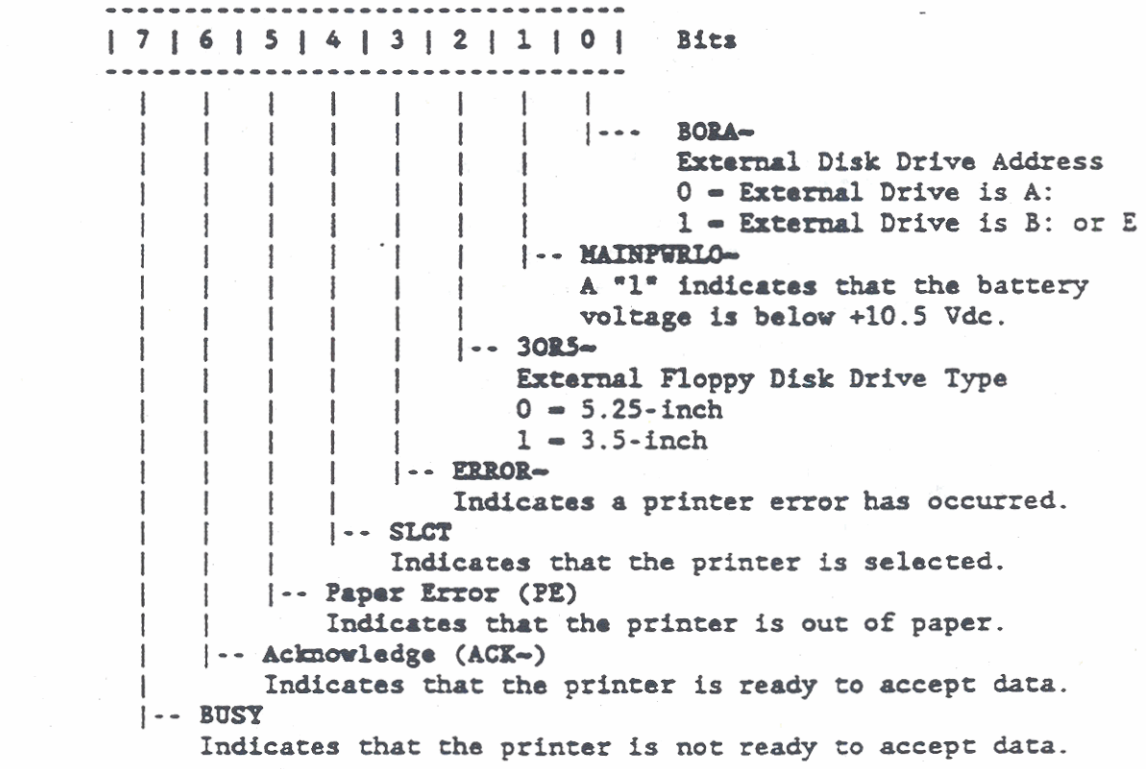
Printer and System Status Registers (0379h and 037Ah)

The Parallel Port Subsystem uses two status registers in the parallel port controller. Both registers are used to determine the printer status, and are also used for system status information. Status register B is located at I/O address 379h and status register C is located at I/O address 37Ah.

Both status registers are 8-bit buffers with three bits of each register used for system status information and the remaining bits are reserved for printer status information. For status register B, the three least significant bits (2-0) are system status bits. For status register C, the three most significant bits (7-5) are system status bits. Bit definitions for the two status registers are given in the following paragraphs.

Printer Status Register B (379h, Read Only)

Printer Status register B (379h) is an 8-bit read-only buffer that provides three bits of system status information and five bits of printer status information. The three least significant bits (2-0) are the system status bits and the remaining bits (7-3) are used for printer status. Bit definitions for Printer Status register B are as follows:



Printer Status Register C (37Ah, Read Only)

Printer Status register C (37Ah) is an 8-bit read-only buffer that provides three bits of system status information and five bits of printer status information. The most least significant bits (7-5) are the system status bits and the remaining bits (6-0) are used for printer status. Bit definitions for Printer Status register C are as follows:

NOTE: Printer Status register C has inverted outputs so that a "0" in any bit position sends a "1" to the microprocessor. For example; if EFLOPPY- = 0 (External Drive Attached), bit 5 would be a "1" at the microprocessor.

7	6	5	4	3	2	1	0	Bits

- **STROBE** — Indicates valid data on the printer data bus
- **AFDXT** — Indicates that Autofeed is ON, which feeds one line of paper after printing is complete
- **INIT-** — Indicates printer was initialized
- **SLCTIN-** — Indicates that the printer has accepted data
- **IRQEN** — Indicates the Interrupt Request IRQ7 in enabled
- **EFLOPPY-** — External Floppy Disk Drive Attached
  - 0 = External Drive Attached
  - 1 = External Drive Not Attached
- **TAPEDETECT** — External Backup Tape Drive Attached
  - 0 = External Tape Drive Attached
  - 1 = No External Tape Drive Attached
- **HDDRIVE** — External High Density Drive Attached
  - 0 = External Drive is High Density
  - 1 = External Drive in Normal Density

Parallel I/O Port Registers (FF8h-FFDh)

The parallel I/O port registers are used when the parallel I/O connector on the computer rear panel must function as an 8-bit parallel input/output port. Any 8-bit parallel device attached to the port can send data and status information to the microprocessor and can receive data and control information from the computer. The three parallel I/O registers can be used to read data or status information. One of the registers (Port B) can be used to write data and control information. The registers are all 8-bit registers, which are accessed only when the register at address 423h is set to "1."

The three registers have the same conventional 8-bit map as follows. Their only differences are in their addresses and function.

Port A Register (FF8h, Read Only)
Port B Register (FF9h for Read, FFDh for Write)
Port C Register (FFAh, Read Only)

7	6	5	4	3	2	1	0	Register Bits
D7	D6	D5	D4	D3	D2	D1	D0	Data Bits

PARALLEL PRINTER/PORT CONNECTOR PIN DEFINITIONS

The parallel printer/port connector is a 25-pin, subminiature, D-type female receptacle, which is mounted on the rear panel of the GRiDCase 1500 Series computer. In the following list, "Out" and "In" are referenced to the GRiDCase 1500 Series computer interface, and not to a printer or other device connected to the computer. For example, pin 1 is an output from the computer to an external device. Figure 11-2 shows the pin layout for the parallel printer/port connector and the connector pin definitions are given in Table 11-2.

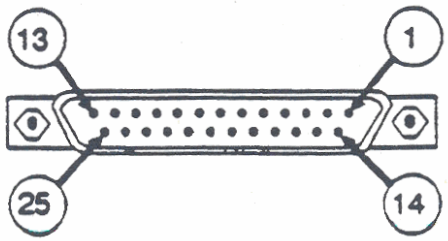


Figure 11-2. Parallel Port Connector Pin Layout

Table 11-2. Parallel Port Connector Pin Definitions

Pin #	Signal	Direction	Description
1	STROBE-	Out	Pulsed low to indicate output is available for printer. Set high during data input operations to hold the ACK- line (pin 10) low.
2	DATA 0	I/O	Data bit 0
3	DATA 1	I/O	Data bit 1
4	DATA 2	I/O	Data bit 2
5	DATA 3	I/O	Data bit 3
6	DATA 4	I/O	Data bit 4
7	DATA 5	I/O	Data bit 5
8	DATA 6	I/O	Data bit 6
9	DATA 7	I/O	Data bit 7
10	ACK-	In	Pulsed low by the printer to indicate that it has received data and is ready for the next byte.
11	BUSY	In	Goes high to indicate that printer cannot receive data.
12	PE	In	High when printer is out of paper.
13	SLCT	In	High when printer is selected.
14	AUTOFDXT-	Out	The printer feeds one line of paper each time auto feed signal goes low.
15	ERROR-	In	Low when the printer is off-line, out-of-paper, or in an error state.
16	INIT-	Out	A low pulse resets the printer and clears its print buffer.
17	SLCTIN-	Out	Set low to enable data transfers to the printer. For data input, this line is set high (+5V) to raise the sender's BUSY line to a high level.
18-25	GND		Twisted-pair signal return (GND).

NOTES:

1. Direction is relative to the computer.

2. The tilde symbol (~) following a signal name indicates active low (low-true) logic.

3. The DATA 0-7 lines are the eight parallel data bits. A high level indicates a logic "1" out, a low level indicates a logic "0" out.

CHAPTER 12: SERIAL I/O PORT AND MODEM SUBSYSTEM

The GRiDCase 1500 Series computers provide an RS-232-C asynchronous serial I/O port and an optional internal modem. In the Model 1520 computers (80C286 Microprocessor), an Intel 82C50 Universal Asynchronous Receiver-Transmitter (UART) is used to control the serial I/O port and a second UART is used to control the modem. The UARTs are designed to interface with asynchronous serial devices such as modems and and other RS-232-C compatible devices. Each UART provides full duplex, double-buffered asynchronous serial communications capability. In addition to the two UARTs, the subsystem includes the logic circuits required to interface between the microprocessor and the external serial I/O devices.

For Model 1530 (80386 Microprocessor) computers, the serial I/O port and modem are both controlled by a VL16C452 Dual Asynchronous Communications Element (DACE) manufactured by VLSI Technology, Inc. The DACE is designed to simultaneously interface with two input/output serial devices such as modems and other RS-232-C compatible devices. Each channel of the DACE provides a full duplex, double-buffered asynchronous serial communications capability. In addition to the DACE, the subsystem includes the logic circuits required to interface between the microprocessor and the external serial I/O devices. Figure 12-1 is a block diagram of the serial I/O port and modem subsystem.

NOTE: Throughout this chapter, register addresses and contents are specified in hexadecimal (h).

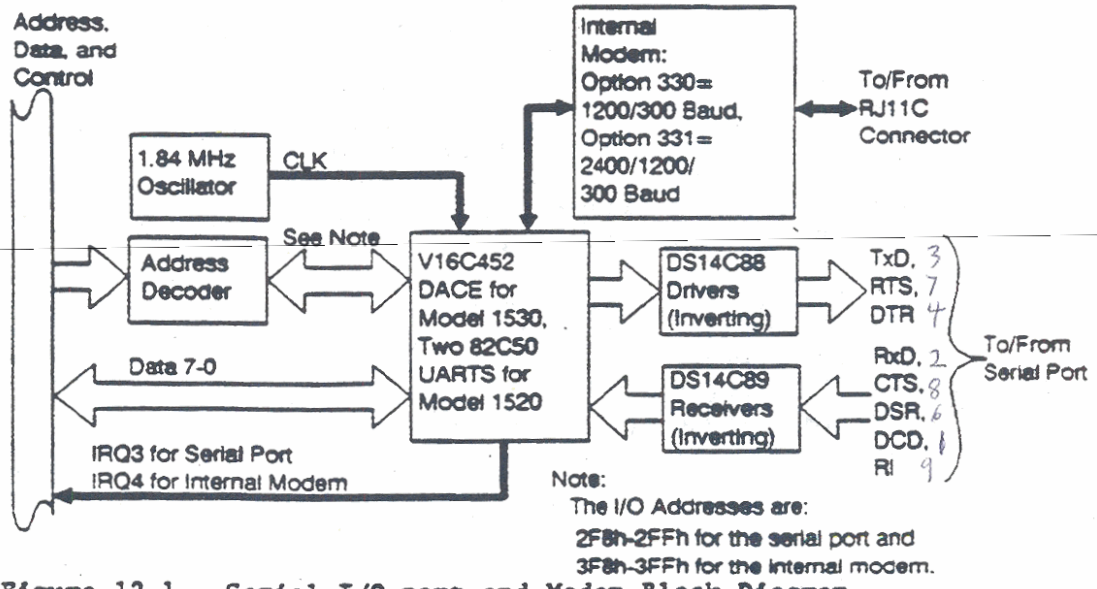


Figure 12-1. Serial I/O port and Modem Block Diagram

Operation of the DACE is similar in operation to two UARTs since both subsystems provide two independent serial communications channels. Therefore, the following descriptions apply to either the DACE or UART based subsystems, unless otherwise specified.

The internal modem is available as either of two options: Option 330 operates at 1200/300 Baud and Option 331 operates at 2400/1200/300 Baud. Both options are Bell 212A and 103 compatible and Option 331 is also V.22 bis compatible. Both modems are also compatible with the Hayes 1200 Smartmodem command set.

The internal modem and the serial I/O port use separate I/O channels so that both devices can operate simultaneously. In the standard configuration, the internal modem is configured as COM1 and the serial I/O port is configured as COM2. However, this configuration is program swappable in some versions of the MS-DOS operating system.

SERIAL I/O PORT ROM-BIOS SERVICE ROUTINE

The ROM-BIOS provides a set of service routine functions to handle the serial I/O port communication requirements. The functions are summarized in the following list and described in the subsequent paragraphs.

Interrupt (Hex)	Function No. (AH reg)	Description
14	00	Initializes the serial port device
14	01	Sends a character to the serial port
14	02	Receives a character from the serial port
14	03	Reads status of the serial port device

The microprocessor registers determine how the service routine functions are used. Register AH determines which function is invoked, while the other registers further define how the action is performed. The contents of Register DX specify the port number. The internal port is always port 0. If an external communications device is used, a port other than port 0 must be specified.

On exit, service functions 00h through 03h return the contents of the Line Status register (2FDh) to microprocessor register AH. Function 02h returns the character received in microprocessor register AL. Functions 00h and 03h return the contents of the Modem Status register (2FEh) to microprocessor register AL. Descriptions of the status registers are provided in subsequent paragraphs.

AH = 00h Initialize the Serial I/O Port
DX = Port Number
AL = Initialization Parameters (Refer to following paragraph)
On Exit
AH = Line Status Register (2FDh) contents
AL = Modem Status Register (2FEh) contents

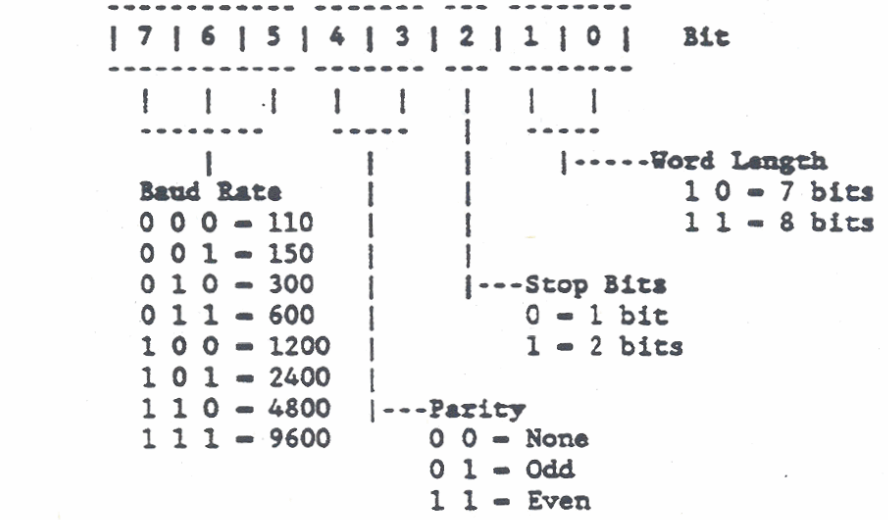
AH = 01h Send Character
DX = Port Number
AL = Character To Send
On Exit
AH = Line Status Register (2FDh) Contents

AH = 02h Receive Character
DX = Port Number
On Exit
AL = Character Received
AH = Line Status Register (2FDh) Contents

AH = 03h Read Status
DX = Port Number
On Exit
AH = Line Status Register (2FDh) Contents
AL = Modem Status Register (2FEh) Contents

Serial I/O Port Initialization Parameters

To initialize the Serial I/O Port (AH = 00h), the following parameters are loaded into 80286/80386 Microprocessor register AL:



CONTROLLER OPERATION

In the Model 1530 computers, the Dual Asynchronous Communications Element (DACE) provides two independent full-featured asynchronous communications channels in one integrated circuit. The operation of each channel is similar to the operation of each 82C50 Universal Asynchronous Receiver Transmitter (UART) used by the Model 1520 computers. Each channel in either subsystem performs serial-to-parallel conversion on data characters received from peripheral devices or the internal modem, and also performs parallel-to-serial conversion on characters received from the 80286/80386 Microprocessor.

Both subsystems (DACE or UART based) provide two separate channels, and the status of either channel can be read by the microprocessor at any time during operation. Each channel also includes a programmable Baud rate generator that divides the 1.84 MHz timing reference clock input by any divisor from 1 through 65,534.

The internal modem and the serial I/O port use separate channels so that both devices can operate simultaneously. In the standard configuration, the internal modem is configured as COM1 and the serial I/O port is configured as COM2. This configuration is program swappable with some versions of the MS-DOS operating system.

I/O Interrupt Capability

Both serial channels also have independent interrupt capability. Channels 0 and 1 support interrupt requests IRQ4 and IRQ3, respectively. Within each serial channel, the conditions that cause an interrupt are enabled and prioritized through programmable registers. When a programmed interrupt condition occurs, the respective interrupt request (IRQ4 or IRQ3) is generated and sent to the microprocessor interrupt controller. The ROM-BIOS then provides the service routine that processes the interrupt.

The optional internal modem uses serial channel 0 and the serial I/O port uses serial channel 1. In the standard configuration, the two channels are connected as follows:

Modem	Channel 0	COM1	IRQ4	INT0Ch
I/O Port	Channel 1	COM2	IRQ3	INT0Bh

NOTE: The serial channel configuration is swappable with some versions of the MS-DOS operating system.

I/O Register Assignments

Since the subsystem provides two identical channels, there are two similar sets of addressable I/O registers. The microprocessor distinguishes between the registers by their I/O addresses. The Serial I/O port registers are located at I/O addresses 2F8h through 2FFh and the internal modem I/O registers are located at I/O addresses 3F8h thought 3FFh. Both sets of registers include Modem Control and Modem Status registers. The modem registers on the serial I/O port allow an external modem to be connected to the subsystem.

The primary addresses of the I/O registers used by both serial channels are provided in the following list. The register bit definitions are provided in the following register descriptions. The serial I/O port connector pin definitions are provided at the end of this chapter.

NOTE: In the following list, the column labeled "DLAB" refers the Divisor Latch Access Bit in the Line Control register. The DLAB bit determines whether I/O addresses 2F8h and 3F8h reference their respective transmitter and receiver registers or their Baud divisor latch registers.

I/O Address Port	I/O Address Modem	Read/ Write	Function	DLAB
2F8h	3F8h	W	Transmitter Holding Register	0
2F8h	3F8h	R	Receiver Buffer Register	0
2F8h	3F8h	R/W	Baud Divisor Latch LSB	1
2F9h	3F9h	R/W	Baud Divisor Latch MSB	1
2F9h	3F9h	R/W	Interrupt Enable Register	0
2FAh	3FAh	R	Interrupt Identification	x
2FBh	3FBh	R/W	Line Control Register	x
2FCh	3FCh	R/W	Modem Control Register	x
2FDh	3FDh	R	Line Status Register	x
2FEh	3FEh	R	Modem Status Register	x
2FFh	3FFh	R/W	Scratchpad Register	x

Transmitter Operation

The transmitter operation of each channel uses the Transmitter Holding register (2F8h/3F8h), a Transmitter Shift register, and associated control logic. The status of the transmitter registers is indicated by two bits in the Line Status register (2FDh/3FDh). The two bits indicate Transmitter Holding Register Empty (THRE) and Transmitter Shift Register Empty (TEMT).

To transmit, a byte is written via Data Lines D0 through D7 to the Transmitter Holding register. The byte is automatically transferred from the Transmitter Holding register to the

Transmitter Shift register. During the transfer, when the start bit is sent, the THRE status bit is set (high). Once THRE is set, the microprocessor performs a write operation to shift the byte out of the shift register one bit at a time through the I/O port to the serial device.

When the transmitter operation is idle, both status bits (THRE and TEMT) are high. Writing the first byte resets THRE to low, but when the transmission is complete, THRE returns to high. The TEMT bit also goes low at least until transmission is complete. If a second byte (character) is written to the Transmitter Holding register, the THRE bit is again reset low.

The byte cannot transfer from the Transmitter Holding register to the Transmitter Shift register until the shift register is empty. Therefore, the THRE bit remains low until the write operation is complete. When the last data bit has been written, the TEMT status bit is set high, and the THRE bit is set high one transfer time later.

Receiver Operation

The receiver operation of each channel uses the Line Control register (2FBh/3FBh), Line Status Register (2FDh/3FDh), Receiver Buffer register (2F8h/3F8h), Receiver Shift register and associated control logic. A start bit detect circuit is continually searching for a high to low transition from the receiver idle state. When the transition is detected, a counter is reset, and counts a 16x clock to 7.5. The count of 7.5 represents the center of the start bit. The start bit is valid if the input line is still low at the center of the start bit. Verifying the start bit in this manner prevents a noise spike on the input from being detected as a start bit.

The Line Control register is programmed to determine the number of data bits in a character (byte), the number of stop bits, if parity is used, and the parity polarity. The Line Status register provides the Data Ready (DR), Overrun Error (OE), Parity Error (PE), and Framing Error (FE) bits. These status bits indicate the receiver operation as follows:

1. The DR status bit is set high when a data byte (character) is strobed a bit at a time into the Receiver Shift register.

2. The Data byte (character) is automatically transferred into the Receiver Buffer register.

3. The DR status bit is reset to low when the microprocessor reads the Receive Buffer register through Data Lines D0 through D7.

4. If the Data Lines (D0-D7) are not read before a new character is transferred from the Receiver Shift register to the Receiver Buffer register, the OE status bit is set.

5. The parity bit, which precedes the first stop bit, is checked against the programmed value. If the values do not match, the PE status bit is set.

6. Finally, the last stop bit is tested. If the last stop bit value is not high, the FE status bit is set.

Bit definitions for all of the registers are provided in the following paragraphs.

Reset Operation

After power is turned ON, the RESET input is held low for a minimum of 500 ns to ensure that the serial channel enters and remains in the idle mode until it is subsequently programmed. When the serial channel is reset, the following operations occur:

1. The transmitter and receiver internal clocks are initialized.

2. The Line Status Register is cleared, except for the Transmitter Holding Register Empty (THRE) and Transmitter Shift Register Empty (TEMT) bits. The THRE and TEMT status bits are set to "1" by a reset operation.

3. When interrupts are subsequently enabled, the THRE status bit causes an interrupt to occur.

4. The Modem Control Register (2FCh/3FCh) is cleared.

5. All discrete lines, memory elements, and associated register bits are either cleared or turned OFF.

6. The Line Control Register, Divisor Latch registers, Receiver Buffer register, and Transmitter Holding register are not affected.

To reset either serial channel to a known state without resorting to a complete system reset, first write to the Line Control register, Divisor Latch registers, and Modem Control register. Next, read the Line Status register and Receiver Buffer register to clear out any residual status or data bits. Finally, enable interrupts through the Interrupt Identification and Interrupt Enable registers. The channel will resume normal operation when the Interrupt Enable register is set.

I/O REGISTER DESCRIPTIONS

The subsystem provides two serial communication channels that interface between the 80286/80386 Microprocessor and two independent RS-232-C compatible devices. One channel is dedicated to an optional internal modem and the other channel can be connected through a serial I/O port to an external device. The external device can be a modem or any other compatible serial device. In the following register descriptions, address locations beginning with "2" are the serial I/O port register addresses, and address locations beginning with "3" are the optional internal modem addresses.

The service routine provided by the ROM-BIOS are adequate for most serial communications applications. However, the ROM-BIOS can be bypassed via the I/O addresses in order to interface directly with the registers. The following paragraphs describe the registers that are accessible via the I/O addresses.

For additional details, refer to the applicable manufacturer's information as follows:

1. The 82C50 UART is manufactured by Intel Corporation, Santa Clara, CA 95051.

2. The VL16C452 DACE is manufactured by VLSI Technology, Inc., Phoenix, AZ 85044.

The registers within each channel can be programmed in any order, except that the Interrupt Enable Register should be programmed last. After a channel has been programmed and is operational, its registers can be updated at anytime the channel is not transmitting or receiving data.

NOTE: The following register descriptions are given in ascending order of their I/O addresses and not in the order that they would be programmed and used.

Transmitter Holding Register (2F8h/3F8h)

The Transmitter Holding register (2F8h/3F8h) is a write-only buffer that holds a 5-bit to 8-bit character byte. The buffered character byte is automatically transferred to a Transmitter Shift register and then serially transmitted a bit at a time to a compatible serial device. The compatible device may be the internal modem or the character can be transferred through the serial I/O port to an external I/O device.

If the character byte is less than eight bits wide, the bits are right justified so that any unused bits are on the left. Bit 0 of each character byte is the first serial data bit to be transmitted and received. The format of the character byte is controlled by the Line Control register (2FBh/3FBh), and the DLAB bit in the Line Control Register must be set to "0" for access to the Transmitter Holding register.

Receiver Buffer Register (2F8h/3F8h)

The Receiver Buffer register (2F8h/3F8h) is a read-only register that holds a 5-bit to 8-bit character byte. The character in the buffer was received through a Receiver Shift register and Serial I/O Port from a compatible serial device. The compatible device can be the internal modem or the character can be received from through the serial I/O port from an external I/O device.

If the character byte is less than eight bits wide, the bits are right justified so that any unused bits are on the left. Bit 0 of each character byte is the first serial data bit to be transmitted and received. The format of the character byte is controlled by the Line Control register (2FBh/3FBh), and the DLAB bit in the Line Control Register must be set to "0" for access to the Receiver Buffer register.

Baud Divisor Latch Registers (2F8h/3F8h and 2F9h/3F9h)

The Baud Divisor Latch is two read/write registers (2F8h and 2F9h or 3F8h and 3F9h). The contents of the registers specify the output frequency of the Baud Rate generator contained within each serial channel. The DLAB bit in the Line Control Register (2FFh/3FBh) is set to "1" for access to the Baud Divisor Latch registers. Otherwise, the registers at addresses 2F8h/3F8h and 2F9h/3F9h are used to access the Transmitter Holding register, Receiver Buffer register, and the Interrupt Enable register, respectively.

The contents of the two 8-bit Baud Divisor Latch registers are treated as one 16-bit word. The register at address 2F8h/3F8h contains the least significant 8 bits, and the register at address 2F9h/3F9h contains the most significant 8 bits. The contents of the registers are used to divide the clock input (1.8432 MHz) to the serial channel. The divisor values used to obtain different baud rates are listed in Table 12-1.

Table 12-1. Baud Rates Determined by the Baud Divisor Latch Register Values

Desired Baud Rate	Divisor Latch Register Contents	% Error
50	900h	-
75	600h	-
110	417h	0.026
134.5	359h	0.058
150	300h	-
300	180h	-
600	0C0h	-
1200	060h	-
1800	040h	-
2000	03Ah	0.69
2400	030h	-
3600	020h	-
4800	018h	-
7200	010h	-
9600	00Ch	-
19200	006h	-
38400	003h	-
56000	002h	2.86

For Baud rates that are not listed, the Divisor latch value is calculated as follows:

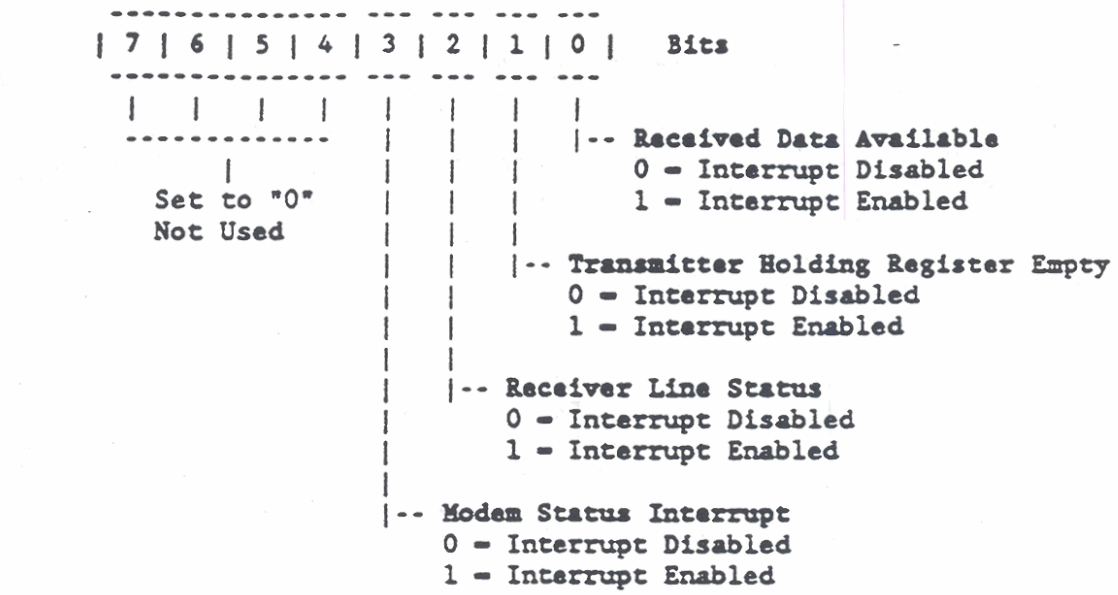
Divisor Latch Value = $(1.8432 \times 10^6 / \text{Baud Rate} \times 16)$ (Rounded Off)

Interrupt Enable Register (2F9h/3F9h)

The Interrupt Enable Register (2F9h/3F9h) is an 8-bit read/write buffer. It is used to independently enable up to four sources that can cause a serial channel to activate its interrupt request line. An active interrupt request for the serial I/O port generates IRQ3. An active interrupt request for the internal modem generates IRQ4.

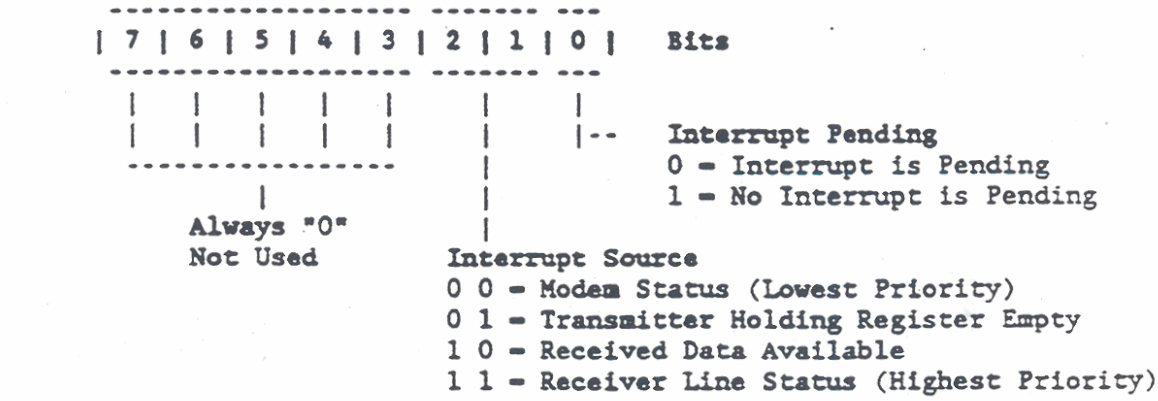
Before the Interrupt Enable register is effective, the DLAB bit in the Line Control Register (2FBh/3FBh) must be set to "0" for access to the Interrupt Enable register. Then, bit 3 of the Modem Control register (2FCh/3FCh) must be set to "1" to enable all interrupts.

When an Interrupt Enable Register bit is set to "1," the corresponding interrupt is enabled. If the bit is reset to "0," the corresponding interrupt is disabled. If bits 0-3 are all set to "0," all interrupts from the selected channel are disabled and the interrupt request signal is inhibited. The bits in the Interrupt Enable Register are assigned as follows:



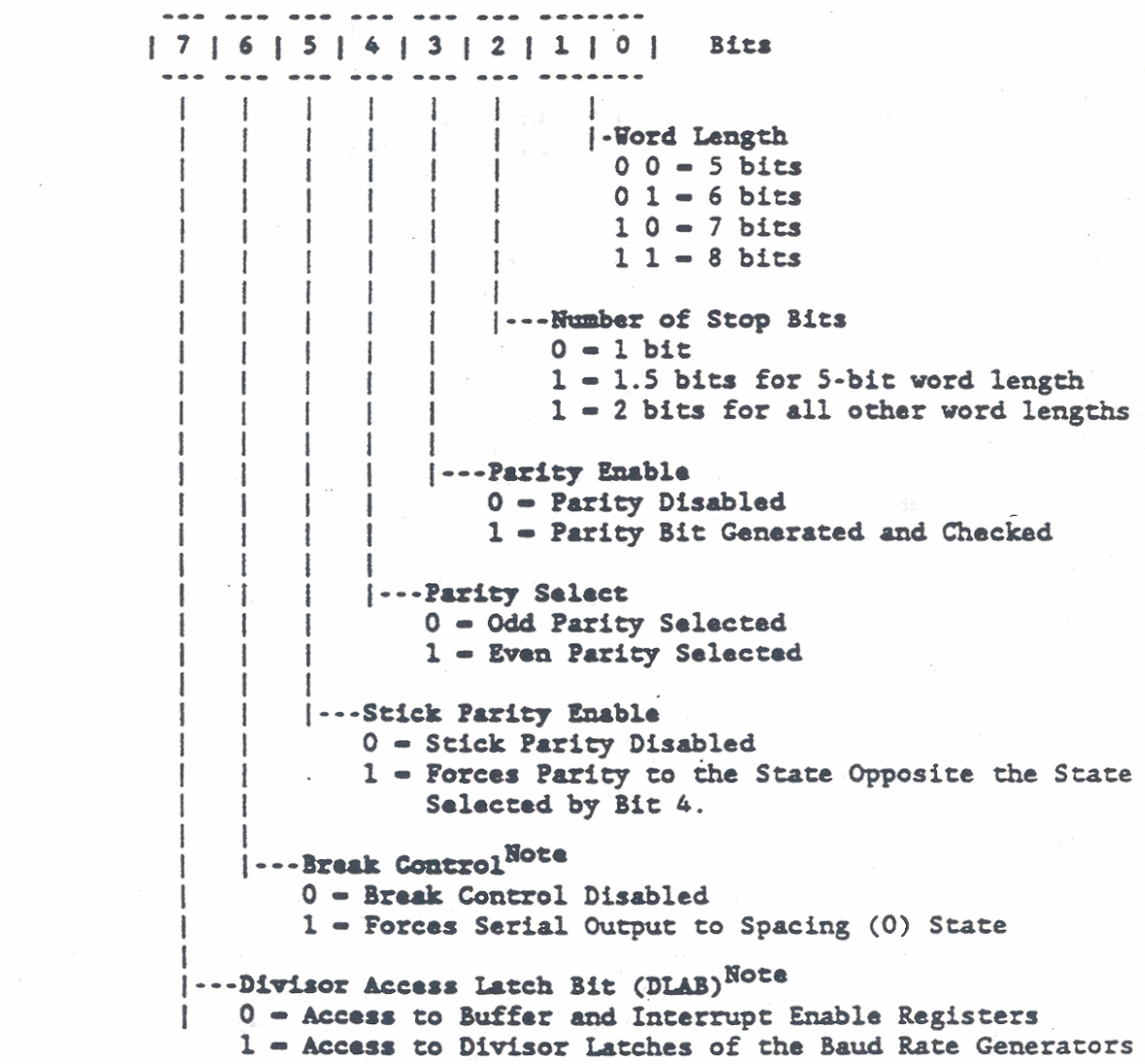
Interrupt Identification Register (2FAh/3FAh)

Each serial channel has an Interrupt Identification Register (2FAh/3FAh), which is an 8-bit read-only buffer. For its associated channel, bit 0 of the register indicates that an interrupt is pending. Bits 1 and 2 indicate which of four possible sources generated the interrupt. Bits 7 through 3 are always reset to "0." The register is polled by the system whenever more than one of the interrupt sources is enabled. The bits in the Interrupt Identification register are read as follows:



Line Control Register (2FBh/3FBh)

The Line Control Register (2FBh/3FBh) is an 8-bit read/write buffer that controls the format of the data character in each channel. It can be used to examine as well as to specify the format of the asynchronous data character transferred between the channel and its associates serial device. The Line Control register bit description is as follows:



NOTE: The following paragraphs further describe the Break Control and Divisor Access Latch bits.

Break Control Procedure

The Break Control (2FBh/3FBh, bit 6) works with the serial out data only and has no effect on the transmitter operation. The microprocessor uses the Break Control to signal a device in a communications system. The following procedure prevents erroneous or extraneous characters from being transmitted as the result of a break.

1. Load an all zeros (0) pad character in response to the active Transmitter Holding Register Empty (THRE) bit in the Line Status Register (2FDh/3FDh).

2. Set BREAK in response to the next active THRE bit.

3. Wait for the transmitter operation to be idle as indicated by an active Transmitter Shift Register Empty (TEMT) bit and when normal transmission operation has been restored, clear BREAK.

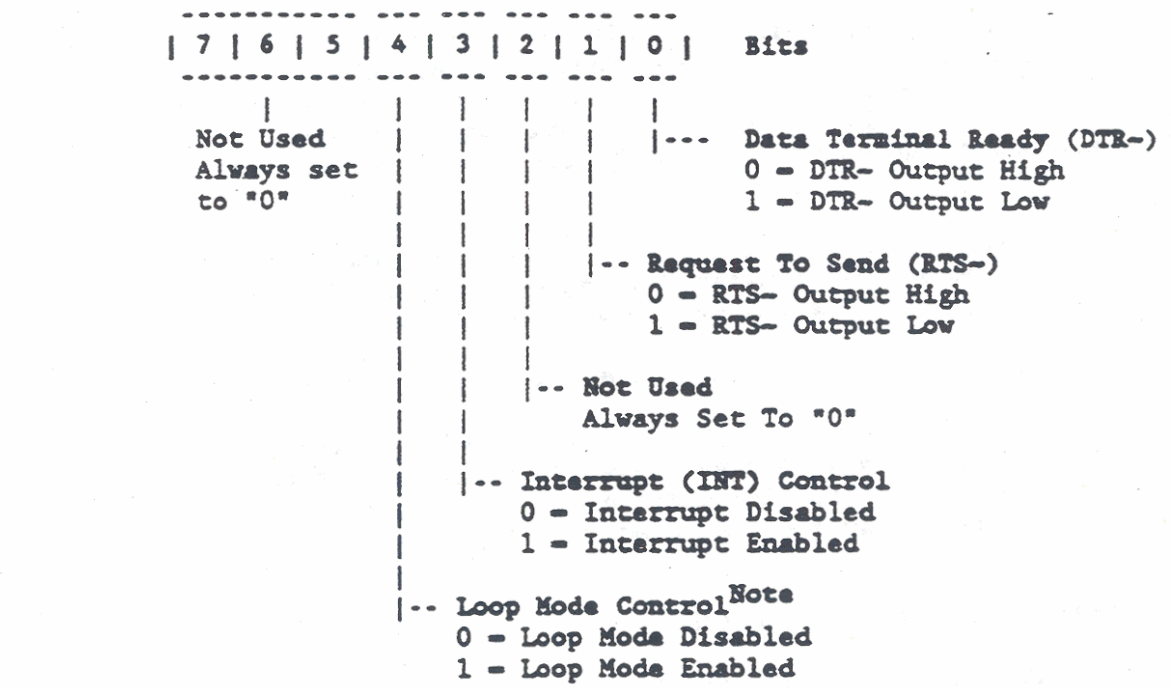
Divisor Latch Access Bit (DLAB)

The DLAB (2FBh/3FBh, bit 7) selects the registers to be accessed by addresses 2F8h/3F8h and 2F9h/3F9h. When DLAB is reset to "0," the Transmitter Holding register and Interrupt Enable register are accessed by an I/O Write command. Also, the Receiver Buffer register and the same Interrupt Enable register are accessed by an I/O Read command. If DLAB is set to "1," the Baud Divisor Latch LSB and Baud Divisor Latch MSB are accessed. These two latches are used to set the data communication (Baud) rate as previously described.

Modem Control Register (2FCh/3FCh)

The Modem Control Register (2FCh/3FCh) is an 8-bit read/write buffer. It is used to examine and to specify the interface to a modem or a data set. Both serial channels have Modem Control registers so that both internal and external modems can be used for data communication. The bits in the Modem Control register are used to manipulate and examine the typical signals required as inputs by a modem. The Modem Control register bit description is as follows:

NOTE: In the Modem Control (2FCh/3FCh) and Modem Status (2FEh/3FEh) registers the sense of some bits is inverted so that a bit set to "1" forces the corresponding output to "0" and vice versa. Signals with an inverted sense are indicated by a tilde (~) suffix.



NOTE: Loop mode control is further described in the following paragraph.

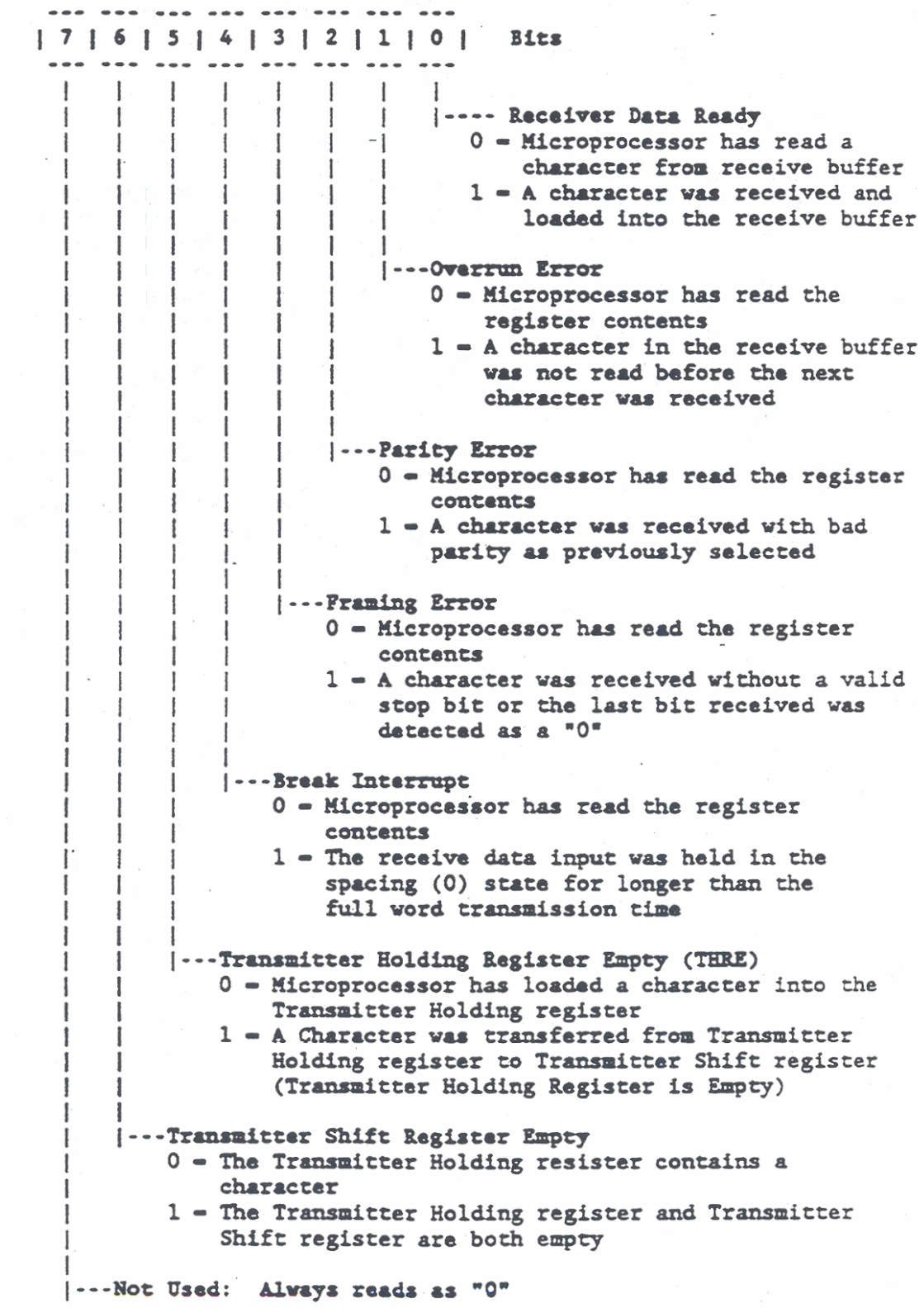
Loop Mode Control

Loop mode control (2FCh/3FCh, bit 4) supports diagnostic testing of the associated channel. If Loop mode is enabled, the Serial Data In is disconnected and Serial Data Out is set to a marking (1) state. The Transmitter Shift register output is then looped back into the Receiver Shift register.

Also, modem control inputs CTS-, DSR-, and RI- are disconnected. Modem control outputs DTR- and RTS- are then internally connected to the modem control inputs, and the modem control output pins are forced to their inactive (high) state.

Line Status Register (2FDh/3FDh)

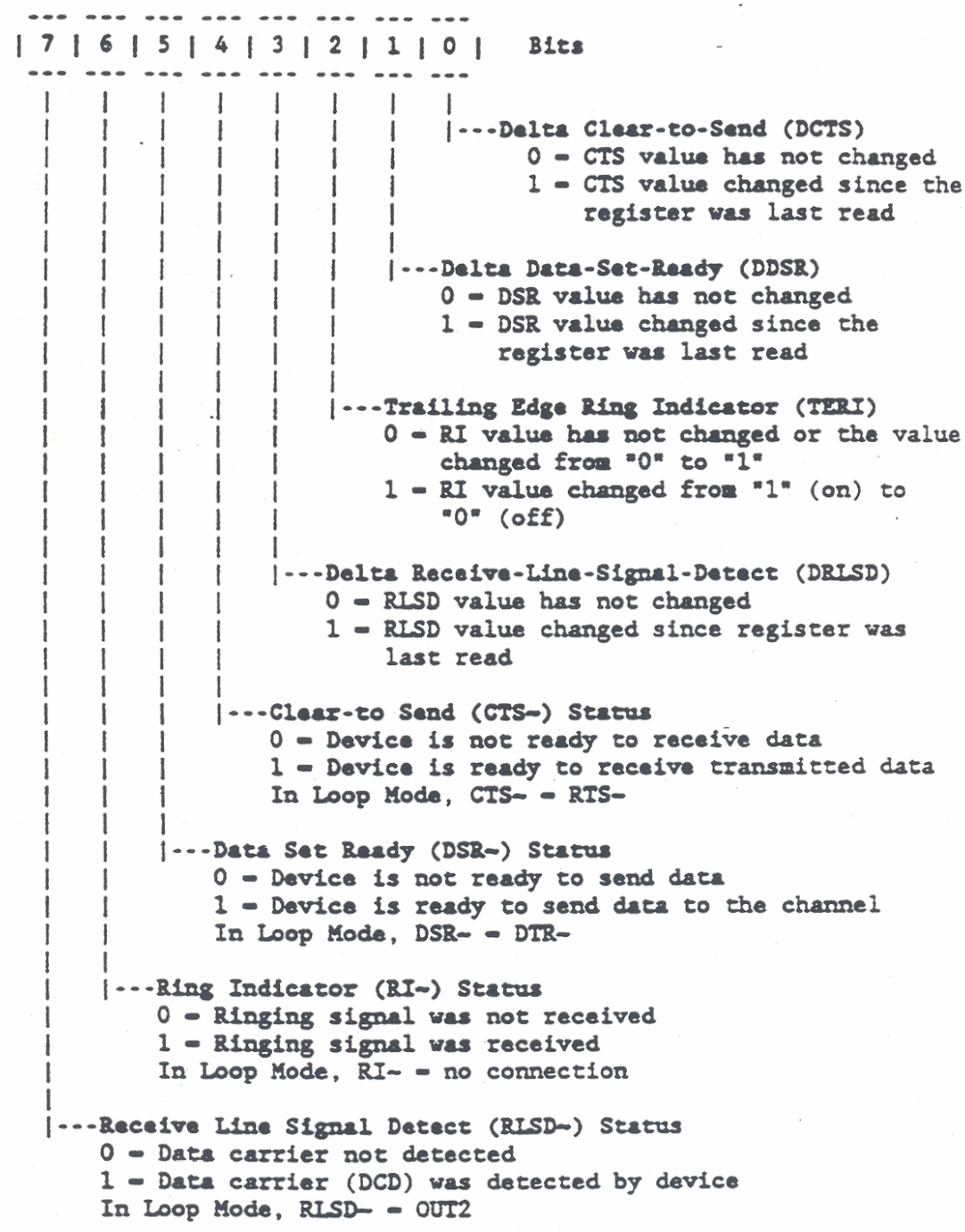
The Line Status register (2FDh/3FDh) is an 8-bit read/write buffer, which is used to indicate the status of transmitter/receiver operations in each serial channel. The Line Status register is usually the first register read by the microprocessor to determine the cause of an interrupt from the associated channel. The register bits are defined as follows:


| 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 | Bits
Receiver Data Ready
0 = Microprocessor has read a character from receive buffer
1 = A character was received and loaded into the receive buffer
Overrun Error
0 = Microprocessor has read the register contents
1 = A character in the receive buffer was not read before the next character was received
Parity Error
0 = Microprocessor has read the register contents
1 = A character was received with bad parity as previously selected
Framing Error
0 = Microprocessor has read the register contents
1 = A character was received without a valid stop bit or the last bit received was detected as a "0"
Break Interrupt
0 = Microprocessor has read the register contents
1 = The receive data input was held in the spacing (0) state for longer than the full word transmission time
Transmitter Holding Register Empty (THRE)
0 = Microprocessor has loaded a character into the Transmitter Holding register
1 = A Character was transferred from Transmitter Holding register to Transmitter Shift register (Transmitter Holding Register is Empty)
Transmitter Shift Register Empty
0 = The Transmitter Holding resister contains a character
1 = The Transmitter Holding register and Transmitter Shift register are both empty
Not Used: Always reads as "0"

NOTE: Bits 4-1 indicate error conditions that cause Receive Line Status (Priority One) interrupts. Bits 6 and 5 are not reset by reading the register contents.

Modem Status Register (2FEh/3FEh)

The Modem Status Register (2FEh/3FEh) is an 8-bit read/write buffer. The register contents provide the status of control lines from the internal modem or from a peripheral device attached to the serial I/O port. Register bits 3 through 0 indicate a change in the line status since the last time the register was read and are called delta bits. Register bits 7 through 4 always reflect the current state of the input lines and are called status bits. Reading the register clears the delta bits but does not affect the status bits. The delta bits do not change states if they are set to "1" and a change in line status occurs during a read operation. If the delta bits are set to "0" and a change in line status occurs during a read operation, the change of state is indicated after the read operation is complete. When any of the delta bits are set to "1," a modem status interrupt (priority 4) is generated. The register bit descriptions are as follows:


| 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 | Bits
Delta Clear-to-Send (DCTS)
0 = CTS value has not changed
1 = CTS value changed since the register was last read
Delta Data-Set-Ready (DDSR)
0 = DSR value has not changed
1 = DSR value changed since the register was last read
Trailing Edge Ring Indicator (TERI)
0 = RI value has not changed or the value changed from "0" to "1"
1 = RI value changed from "1" (on) to "0" (off)
Delta Receive-Line-Signal-Detect (DRLSD)
0 = RLSD value has not changed
1 = RLSD value changed since register was last read
Clear-to Send (CTS~) Status
0 = Device is not ready to receive data
1 = Device is ready to receive transmitted data
In Loop Mode, CTS~ = RTS~
Data Set Ready (DSR~) Status
0 = Device is not ready to send data
1 = Device is ready to send data to the channel
In Loop Mode, DSR~ = DTR~
Ring Indicator (RI~) Status
0 = Ringing signal was not received
1 = Ringing signal was received
In Loop Mode, RI~ = no connection
Receive Line Signal Detect (RLSD~) Status
0 = Data carrier not detected
1 = Data carrier (DCD) was detected by device
In Loop Mode, RLSD~ = OUT2

INTERNAL MODEM (OPTIONAL) DESCRIPTION

The internal modem is available as either of two options: Option 330 operates at 1200 and 300 Baud, and Option 331 operates at 2400, 1200, and 300 Baud. Both modems operate in full-duplex mode with autodial and autoanswer capabilities. The modems are Bell 212A and 103 compatible and the Option 331 is also V.22bis compatible. The modems are also compatible with the Hayes 1200 Smartmodem command set. Additional information provided in this section is applicable to both modems (330 and 331) unless specified otherwise.

The external modem interface is provided through two modular RJ11C phone jacks on the back panel of the GRiDCase 1500 Series computer. The two phone jacks are connected together in parallel to provide the external connection to either a data or voice quality telephone line. If one phone jack is connected to the telephone line, the second phone jack can be used to connect an external telephone to the same line. However, the telephone cannot be used while the modem is active.

Modem Control

NOTE: The following modem control information is applicable to Option 330 only. Option 331 does not use the I/O registers. Control of the Option 331 modem is provided through the use of the modem commands described later in this chapter.

Control of the Option 330 internal modem is provided through three one-bit I/O Registers as follows:

I/O Address	Function
420h	0 = Normal Operation 1 = Analog Loopback (Diagnostic Mode)
421h	0 = Smart Terminal Mode 1 = RS-232-C Handshake Modem

For the Option 330 modem, the I/O Register at address 421h determines how the Data Set Ready (DSR) and Data Carrier Detect (DCD) lines operate. When the register is set to "0," the state of DSR and DCD follows the state of the Data Terminal Ready (DTR) line. This mode of operation is required to operate some "smart" terminals. If the register is set to "1," the state of DSR and DCD follows the state of the data carrier, and the Clear to Send (CTS) line is always active. This is the normal RS-232-C handshake mode and also the Hayes modem compatible mode.

Modem Extensions

The Option 330 internal modem provides several extensions to the Hayes 1200 Smartmodem. The extensions provided by the GRidCase 1500 Series Option 330 internal modem are as follows:

1. Electronic Call Progress Tone Detection. Four status messages (6-9) were added to enable the modem to determine the condition of the telephone line. The messages are enabled by using Modem Command X2 (refer to Table 12-3).

2. Autoselection of Tone or Rotary Pulse Dialing. The modem provides autoselection if Modem Command X2 has been issued since the last system reset.

3. Additional Test Modes for Diagnostics. Five test modes were added to Modem Register S16 for telephone line and modem diagnostics. The Modem Registers are described later in this chapter.

Modem Commands

The internal modem accepts commands after it has been selected via the ROM-BIOS interrupt service routine or via the I/O registers. Commands are sent to the modem through the Transmit Holding Register of the serial channel where the modem is connected. The commands are sent as command characters by using the ROM-BIOS interrupt service routine or a write operation to I/O address 3F8h. Address 3F8h assumes the default modem interface (COM1). To address a modem connected to the serial I/O port, use I/O address 2F8h.

Each modem command character must be prefixed with the two-character sequence AT (abbreviation for attention) and followed by the Return character (0Dh). The prefix AT must be in upper case letters as shown, but the command characters can be either upper or lower case. For example, a reset command is sent to the modem by writing the following command to I/O register 3F8h:

ATz(Return)

In the previous example, AT is the required prefix, z is the reset command character, and (Return) represents the Return character (0Dh). Multiple commands can be sent as long as the command string is prefixed by the AT characters, does not exceed 40 characters, and a Return character is used to terminate the string. The internal modem supports the following command characters (command characters can be either upper or lower case):

Command Character	Command Description
A	Answer Mode. The modem is place in answer mode even though it has not received an incoming call. Modem Register S7 specifies how long to wait for a carrier.
A/	Repeat the last command string. This command is typically used to redial a number that was busy. The command does not require a prefix or a (Return).
Cn	Carrier Control. If n = 0 the carrier is turned OFF. If n = 1 the carrier is turned ON. The modem is initialized with the carrier on/off state controlled automatically. The Cn command over-rides the automatic control.
D	Dial the telephone number that follows. Option commands used with the Dial (D) command include: P, T, R, W, @, , ;, "..., and !. The option command descriptions are included in the following list.
En	Echo Control. If n = 0 echo is off. If n = 1 all keyboard commands are echoed to the display.
Fn	Duplex Control. If n = 0 local echo is on (half duplex) and the modem sends a copy of transmitted data to the display. If n = 1 local echo is off (full duplex) and a remote terminal can send data to the display.
Hn	Control ON/OFF hook. If n = 0 receiver is on hook (hung up). If n = 1 receiver is off hook.
In	Inquiry Code. Returns the status information specified by n as follows: n = 0 Returns product code. n = 1 Returns Checksum. n = 2 Run modem memory (RAM) test. n = 3 Returns call duration or real time. n = 4 Returns current modem settings.
Kn	Modem clock. Not used.
Mn	Modem speaker control. Controls the speaker as follows: n = 0 Turns speaker OFF. n = 1 Turns speaker ON until carrier is established (default). n = 2 Speaker is always ON. n = 3 Turns speaker ON after the last digit is dialed and until the carrier is established.
O	Return to on-line state after command execution.

Command Character	Command Description
P	Pulse dialing. Used alone or with the D command.
Qn	Quiet mode. Modem result messages are displayed if n = 0 (default). If n = 1 messages are suppressed.
R	Reverse frequency for answer-mode handshake after dialing.
Sn	Modem Register commands. n = 0 thru 16 (refer to Table 12-2).
T	Touch-tone dialing. Used alone or with the D command.
Vn	Verbal/numeric display of result codes. n = 0 selects numeric mode to display Result Codes. n = 1 selects verbal mode to display messages (refer to Table 12-3).
W	Wait for second dial tone (Option 331 only).
Xn	Selects result code options (refer to Table 12-3).
Z	Resets modem to default configuration.
@	Wait for answer. Used with the D command.
;	Return to command mode while on line. Used with D command.
,	(Comma) Specify a 2 second pause in command string. Used with D command.
+	Specify pause for dial tone (Option 330 only).
+++	Escape Command. Takes modem off line and returns it to the command state. This command is not preceded by the attention command (AT) or followed by a return command.
*...	Dial the letters that follow. Used with D command for Option 331 only.
!	Transfer call. Used with D command for Option 331 only.
>	Repeat Current Command. Used with D command to cause the command to be repeated until the connection is made, the command is repeated 10 times without an answer, or any key is depressed to cancel the command. When used with any other command, the command is repeated until any key is depressed.

Modem Registers

The internal modem (optional) provides 17 registers as described in Table 12-2. All of the registers are initially set to a default value. The Option 330 default values are the same values used by the Hayes 1200 Smartmodem. To change the register values, the register to change is first selected by sending an S command character followed by the register number. Then, a write operation is performed. For example, the number of seconds that the modem waits for a dial tone before dialing is changed from the default value of 2 to the new value of 4 with the following command:

ATs6=4(Return)

In the previous example, AT is the required prefix, s6 is the modem register number, =4 writes the value 4 into the register, and (Return) represents the Return character 0Dh.

The S-register functions and commands provide greater flexiblity in controlling modem operations. Table 12-2 lists and describes the S registers, and provids the default setting for each register in ASCII equivalent decimal and hexadecimal values. Operation of the S registers is similar in both the Option 330 and 331 modems. Any specific differences in operation are noted in Table 12-2.

Table 12-2. Internal Modem (Optional) Register Descriptions

Register Designation	Register Description	Default Value Dec	Hex
S0	Sets the number of rings allowed before the modem autoanswers incoming calls. If set to default (0), the modem does not autoanswer.	1	1
S1	Counts and stores the number of rings for incoming calls.	na	
S2	Sets ASCII value of escape code sequence.	43	2B
S3	Sets ASCII value of Carriage Return.	13	0D
S4	Sets ASCII value of Line Feed.	10	0A
S5	Sets ASCII value of Backspace	8	08
S6	Sets time delay in seconds that the modem waits for a dial tone before dialing.	2	02
S7	Sets time delay in seconds that the modem waits for the carrier from a remote modem.	30	1E
S8	Sets time delay in seconds to pause for a comma inserted in a dial command.	2	02
S9	Sets time delay in tenths of seconds to wait for a carrier from a remote modem.	6	06
S10	Sets time delay in tenths of seconds to wait before disconnecting after a remote modem hangs up.	7	07
S11	Sets the speed in milliseconds of the touch-tone dialer.	70	46

Table 12-2. Internal Modem (Optional) Register Descriptions (Continued)

Register Designation	Register Description	Default Value Dec	Hex
S12	Set the Escape Guard Time:		
	In seconds for Option 330	1	01
	In 50ths of a second for Option 331	50	32
S13	Not supported. Used by the Smartmodem as a bit-mapped status register.	0	00
S14	For Option 330, register S14 is not supported. S14 is used by the Smartmodem as the bit-mapped Option register.	0	00
	For Option 331, S14 sets the product code returned by the ATi0 command	24n	18n
S15	Not Supported. Used by the Smartmodem as the bit-mapped Flag register.	0	00
S16	Enables the modem self-test mode. Extends the Smartmodem 1200 capability by adding four test modes for telephone line and modem diagnostics. The test modes are: 0 = Data Mode (No Testing) 1 = Analog Loopback 2 = Dial Test 4 = Test Pattern 5 = Analog Loopback and Test Pattern	0	00

Result Code Options

The Result Code options are selected by modem command Xn, where n is any value from 0 thru 6. Each value of n selects a group of display options, and values 2 thru 6 also select special modem functions. For example, a group of display options and special functions are selected with the following command:

ATx4(Return)

In the previous example, AT is the required prefix, x4 is the result codes option group number, and (Return) represents the return character 0Dh.

Table 12-3 lists the messages that are displayed, and the special modem functions that are selected, by each of the Xn commands.

Table 12-3. Modem Result Code Options

Result Code	Message Displayed	Modem Command X0	X1	X2	X3	X4	X5	X6
0	OK	Y	Y	Y	Y	Y	Y	Y
1	CONNECT	Y	Y	Y	Y	Y	Y	Y
2	RING	Y	Y	Y	Y	Y	Y	Y
3	NO CARRIER	Y	Y	Y	Y	Y	Y	Y
4	ERROR	Y	Y	Y	Y	Y	Y	Y
5	CONNECT 1200	-	Y	Y	Y	Y	Y	Y
6	NO DIAL TONE	-	-	Y	-	Y	-	Y
7	BUSY	-	-	-	Y	Y	Y	Y
8	NO ANSWER (Replaces No Carrier, requires @)	-	-	-	Y	Y	Y	Y
9	Reserved							
10	CONNECT	-	Y	Y	Y	Y	Y	Y
11	RINGING	-	-	-	-	Y	Y	
12	VOICE	-	-	-	-	-	Y	Y
	Special Functions							
	Adaptive Dialing	-	-	Y	Y	Y	Y	Y
	Wait for Second Dial Tone (W)	-	-	-	Y	Y	Y	Y
	Wait for Answer (@)	-	-	-	Y	Y	Y	Y
	Fast Dial	-	-	Y	-	Y	-	Y

NOTE: Only one Modem Command (Xn) can be active. A "Y" indicates that the message and the special function are enabled if the command at the top of the column is active. The basic result codes (0 thru 4) are displayed with all Xn commands. Modem command X1 sets the default group of messages (0 thru 5 and 10). An "-" indicates that the message and the function are suppressed when the command at the top of the column is active.

SERIAL I/O PORT AND MODEM EXTERNAL CONNECTIONS

The serial I/O port and internal modem have connectors on the GRiDCase 1500 Series computer back panel for connecting external devices. The following paragraphs provide connector pin definitions for the two types of connectors.

Serial I/O port Connector Pin Definitions

For the serial I/O port, there is an interface connector located on the rear panel of the GRiDCase 1500 Series computer (connector 3, Serial RS232C). The 9-pin, subminiture, D-type, male connector provides connections to interface RS-232-C compatible signals with external serial devices. The connector pin layout is shown in Figure 12-2 and the interface connector pinouts and signal names are given in Table 12-4. The direction of flow listed in Table 12-2 is relative to the computer.

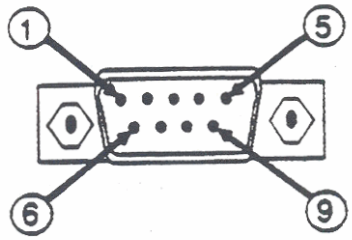


Figure 12-2. Serial I/O port Connector Pin Layout

Table 12-4. Serial I/O port Connector Pin Definitions

Pin #	Function	Direction
1	Carrier Detect (DCD)	In
2	Received Data (RxD)	In
3	Transmitted Data (TxD)	Out
4	Data Terminal Ready (DTR)	Out
5	Signal Ground	--
6	Data Set Ready (DSR)	In
7	Request to Send (RTS)	Out
8	Clear To Send (CTS)	In
9	Ring Indicator (RI)	In

Serial I/O port Interface Signals

The serial I/O port interface hardware consists of one DS14C88 inverting line driver and two DS14C89 inverting line receivers. The driver and receivers translate standard TTL or CMOS logic levels to or from levels that conform to RS-232-C or CCITT V.24 standards. The transmit and receive signal relative timing is shown in Figure 12-3, and signal levels required for the input and output lines are given in the following list:

Driver	Signal Level
Supply Voltage, dc:	+/-4.5 to +/-12
Output Voltage, dc	
Logic 0, space:	-3 minimum, -7 typical
Logic 1, mark:	+9 maximum, +7 typical
Output Current, mA	
Shorted to ground:	+/-25
Output Resistance, Ohms	300

Receivers	Signal Level
Supply Voltage, dc:	+4.5 to +5.5
Input Voltage, dc	
Logic 0, space:	+0.5 to +1.7
Logic 1, mark:	+1.3 to +2.5
Input Current, mA	-0.43 to +8.3

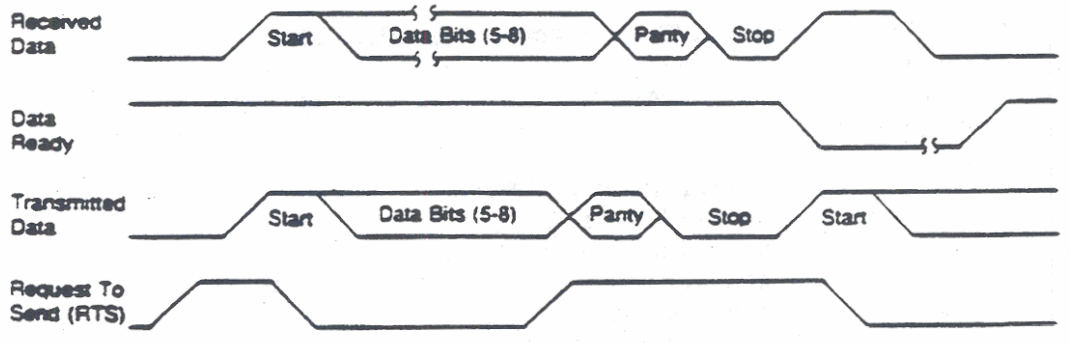


Figure 12-3. Serial I/O port Transmit and Receive Signal Relative Timing

Modem Connector Pin Definitions

Two RJ11C modular telephone jacks are provided on the rear panel of the GRiDCase 1500 Series computer. One connector is located directly below the other and they are electrically connected together in parallel so that the external connections are identical. Either jack can be used to connect the internal modem to a telephone line, and the other jack can be used to connect an ordinary telephone. A telephone connected in this manner can be used as a normal telephone, except when the modem is sending and receiving data. Since connections to both jacks are the same, the pinouts for either connector are as follows:

Pin No.	Description
1	EQL 3
2	EQL 2*
3	RING
4	TIP
5	EQL 1*
6	IA

*Indicates a connection via an internal selectable jumper.

CHAPTER 13: I/O EXPANSION CAPABILITY

The GRiDCase 1500 Series Computer provides connectors on its rear panel for interfacing with external peripheral devices. Table 13-1 identifies each connector by name and the connector number as it is marked on the rear panel. Pinout information for connectors 1 through 7 is provided near the end of the referenced chapters. This chapter provides detailed information about the Expansion Bus Interface only.

Table 13-1. GRiDCase 1500 Series Computer External Connectors

Connector Number	Connector Name	Reference Chapter	Purpose of Connector
1	KEYBOARD	8	Connects to an external Keypad or Enhanced IBM-PC/AT Keyboard
2	PHONE LINE (two available)	12	Connects to an external telephone line
3	SERIAL RS232	12	Connects to external Serial I/O Devices
4	PARALLEL PRINTER	11	Connects to external Printer or Parallel I/O Devices
5	EXTERNAL PERIPHERAL	9	Provides connections to portable peripheral devices (Pocket Floppy, Pouch Floppy, Pouch Tape)
6	VIDEO OUTPUT	7	Connects to external CGA compatible monitors
7	INPUT POWER 16VDC		Connects to an external input power adapter
-	Expansion Bus Interface (Battery Backplane)		Provides the I/O bus expansion capability for connecting Expansion Cartridges or an external Expansion Chassis

EXPANSION BUS INTERFACE

The expansion bus interface is used to extend the internal system I/O bus to external devices. The interface is located inside the slot where a battery is normally installed. When the battery is removed, any one of several expansion cartridges, which have the same form factor as the internal battery, can be installed on the expansion bus interface. The expansion cartridge provides the interface, adapter, and controller logic as required for a specific peripheral device or application.

Expansion cartridges for some popular devices and applications are currently available as options. Also, a kit is available that provides the form factored parts necessary to design and build custom application cartridges. The available cartridges and their applications are listed in Table 13-2. Information about operation of the cartridges is contained in separate manuals, which are supplied with the cartridges.

Table 13-2. Expansion Cartridges

Model No.	Name	Application
32157	Expansion Cartridge Developer's Kit	Contains form factored enclosure parts for designing and building a custom expansion cartidge.
34010	High Speed Serial Expansion Cartridge	Provides RS-232 and RS-422 compatible, high-speed, serial communications. The interface uses asynchronous, byte synchronous, and bit synchronous communications protocols.
34012	3270 Expansion Cartridge	Includes PCOX/ONE software for a single host session and interface to the IBM 3270 Information Display System.
34013	VGA Video Controller Expansion Cartridge	Provides an interface for the external Video Graphics Adapter (VGA) monitor.
34014	Ethernet Network Expansion Cartridge	Provides an interface for the Ethernet Network.
34015	3270 Expansion Cartridge	Includes PCOX/MULTI software for a multiple host session and interface to the IBM 3270 Information Display System.

NOTE: A Model 34102 XT-Compatible Expansion Chassis is available. The Model 34102 is a separate chassis that provides mounting capacity for two full-length PC cards. No additional chassis interface hardware is required.

The expansion bus interface consists of one 60-pin and one 40-pin connector (see Figure 13-1). The 60-pin connector provides the connections required to emulate the IBM XT I/O bus. The 60-pin and 40-pin connectors together provide the connections required to emulate the IBM AT I/O bus. Signal connections to the two connectors are listed in Tables 13-3 and 13-4, and descriptions of the expansion bus signals are given in the following paragraphs.

Table 13-3. Expansion Bus 60-Pin Connector

Pin No.	Signal Name	Direction of Flow	Pin No.	Signal Name	Direction of Flow
1	DACK2-	Output	31	OSC	Output
2	SWPOWER	Power	32	DATA6	I/O
3	DRQ1	Input	33	IOR-	Output
4	DRQ2	Input	34	DATA5	I/O
5	DACK3-	Output	35	P12V	Power
6	DRQ3	Input	36	DATA4	I/O
7	DACK1-	Output	37	ADD0	I/O
8	IOCHRDY	Input	38	DATA3	I/O
9	IRQ5	Input	39	ADD1	I/O
10	IOCK-	Input	40	DATA2	I/O
11	+VDC	Power	41	ADD2	I/O
12	BALE	Output	42	DATA1	I/O
13	Signal Grd	Ground	43	ADD3	I/O
14	DATA7	I/O	44	DATA0	I/O
15	Signal Grd	Ground	45	ADD4	I/O
16	IRQ6	Input	46	ADD19	I/O
17	IOW-	Output	47	ADD5	I/O
18	REFRESH-	Output	48	ADD18	I/O
19	IRQ7	Input	49	ADD6	I/O
20	RESET	Output	50	ADD17	I/O
21	SMEMWR-	Output	51	ADD7	I/O
22	BUFAEN	Output	52	ADD16	I/O
23	SMEMRD-	Output	53	ADD8	I/O
24	TC	Output	54	ADD15	I/O
25	IRQ9	Input	55	ADD9	I/O
26	IRQ4	Input	56	ADD14	I/O
27	GRIDBITS3WR-	Output	57	ADD10	I/O
28	SYSCLK	Output	58	ADD13	I/O
29	P12V	Power	59	ADD11	I/O
30	IRQ3	Input	60	ADD12	I/O

NOTES:
1. Direction is relative to the microprocessor.
2. The tilde symbol (-) indicates active low (low-true) logic.

Table 13-4. Expansion Bus 40-Pin Connector

Pin No.	Signal Name	Direction
1	Signal Ground	Ground
2	IRQ12	Input
3	+VDC	Power
4	IRQ11	Input
5	Signal Ground	Ground
6	IRQ10	Input
7	IRQ15	Input
8	SPARE1	Spare
9	IRQ14	Input
10	LA17	I/O
11	MASTER-	Input
12	DRQ7	Input
13	IOCS16-	Input
14	DACK7-	Output
15	MEMCS16-	Input
16	DRQ6	Input
17	DACK0-	Output
18	DACK6-	Output
19	DRQ0	Input
20	DRQ5	Input
21	DACK5-	Output
22	LA18	I/O
23	ZEROWS-	Input
24	LA19	I/O
25	EBHE-	I/O
26	LA20	I/O
27	MEMWR-	Output
28	LA23	I/O
29	MEMRD-	Output
30	LA22	I/O
31	SWPOWER	Power
32	LA21	I/O
33	DATA8	I/O
34	DATA15	I/O
35	DATA9	I/O
36	DATA14	I/O
37	DATA10	I/O
38	DATA13	I/O
39	DATA11	I/O
40	DATA12	I/O

NOTES:

1. Direction is relative to the microprocessor.
2. The tilde symbol (~) indicates active low (low-true) logic.

Expansion Bus Signal Descriptions

The following paragraphs provide brief descriptions of the signals on the I/O and expansion bus. All of the signals lines are TTL compatible. I/O adapters designed for use on the I/O and expansion bus should have a maximum of two low-power Shottky (LS) loads per line.

NOTE: Where pin locations are given in the following descriptions, the designations such as (60/10) indicate pin number 10 of the 60-pin connector.

ADD0-ADD19

The ADD0 through ADD19 (60-pin connector) lines provide a 20-bit address for system memory and I/O devices. The 20-bit address is gated onto the I/O and expansion bus when BALE (60/12) is high and is latched onto the bus on the falling edge of BALE. Twenty-bit addresses can be generated by any microprocessor or DMA controller that resides on the I/O or expansion bus. Refer to LA17-LA23.

BALE

The BALE line (60/12) provides the Buffered Address Latch Enable signal that controls the address line access to the I/O and expansion bus. BALE is also used to indicate when there is a valid microprocessor or DMA address on the I/O or expansion bus. Address lines ADD0 through ADD19 are latched on the falling edge of BALE, and BALE is driven high during DMA cycles.

BUFAEN

The BUFAEN line (60/22) provides the Buffered Address Enable signal. BUFAEN is driven high by the DMA controller to enable the upper 8 address bits onto the address bus. BUFAEN can also be used to prevent other system bus drivers from accessing the address bus during DMA transfers.

DACK0--DACK3- and DACK5--DACK7-

The DACK0- through DACK3- and DACK5- through DACK7- lines provide seven DMA Acknowledge signals, which are used to acknowledge the seven DMA Requests. The DACK0-, and DACK5- through DACK7- lines are on the 40-pin connector and the DACK1- through DACK3- lines are on the 60-pin connector. The DMA controller drives the DACKnline low (active) that corresponds to the highest priority DRQn signal that is active. DACK4- is not used because DRQ4 is not available to the I/O or expansion bus. Refer to DRQ0-DRQ3 and DRQ5-DRQ7.

DATA0-DATA15

The DATA0 through DATA15 lines provide a 16-bit data bus. The data bus is used to transfer information between the microprocessor, memory, and internal and external I/O devices. DATA0 is the least significant bit and DATA15 is the most significant bit. Eight-bit devices on the I/O and expansion bus should use DATA0 through DATA7 (60-pin connector) for communication with the microprocessor. To support 8-bit devices, data on the DATA8 through DATA15 lines (40-pin connector) are gated to the DATA0 through DATA7 lines during 8-bit transfers. Also, 16-bit microprocessor transfers to 8-bit devices are converted to two 8-bit transfers.

DRQ0-DRQ3 and DRQ5-DRQ7

The DRQ0 through DRQ3 and DRQ5 through DRQ7 lines provide seven Direct Memory Access (DMA) Request lines. The DRQ0 and DRQ5 through DRQ7 lines are on the 40-pin connector and DRQ1 through DRQ3 are on the 60-pin connector. The DMA request lines are used by microprocessors and other devices on the I/O and expansion bus to request service from the DMA controller. The DMA requests are assigned priorities with DRQ0 having the highest and DRQ7 having the lowest. The device that is requesting service drives its respective DRQn line low (active) and holds it low until the corresponding DMA Acknowledge (DACK) line goes low.

DMA requests DRQ0 through DRQ3 are used for 8-bit to 8-bit DMA transfers between I/O adapters and 16-bit or 32-bit memory locations. DRQ5 through DRQ7 are used to perform 16-bit DMA transfers between I/O adapters and 16-bit or 32-bit memory locations. DRQ4 is reserved for cascading together the two DMA controllers and is not available to the I/O or expansion bus. Table 13-5 lists the DMA request data for quick reference and also lists the associated page registers, including the refresh page register, for quick reference. Additional DMA data is contained in Chapter 4.

Table 13-5. DMA Request Lines

DMA Request	Page Register	Internal Function	Interface Priority	Bus Width	Expansion Bus Connector/Pin
DRQ0	87h	Spare	1 (Highest)	8-bit	40/19
DRQ1	83h	Reserved	2	8-bit	60/3
DRQ2	81h	Floppy Disk	3	8-bit	60/4
DRQ3	82h	Hard Disk	4	8-bit	60/6
DRQ4	--	Cascade Channel	-	---	---
DRQ5	8Bh	Spare	5	16-bit	40/20
DRQ6	89h	Spare	6	16-bit	40/16
DRQ7	8Ah	Spare	7 (Lowest)	16-bit	40/12
Refresh	8Fh	Memory Refresh	-	---	---

EBHE~

The EBHE~ line (40/25) provides the Bus High Enable signal that indicates a data transfer on the upper 8 bits of the 16-bit data bus. Sixteen-bit I/O devices use EBHE~ to condition the data bus buffers for bits DATA8 through DATA15.

GRIDBITS3WR~

The GRIDBITS3WR~ (60/27) line is unique to the GRiD Systems computers. GRIDBITS3WR~ is derived from an address decode and an active I/O Write (IOW~). The resultant signal is used to select the unique GRiD ROM-BIOS Subsystem functions.

IOCHRDY

The IOCHRDY line (60/8) is used to lengthen the I/O or memory cycles. Any slow device on the I/O or expansion bus should drive IOCHRDY low as soon as it detects its own address and a Read or Write command. When IOCHRDY is driven low, machine cycles are extended by an integral number of clock cycles. The IOCHRDY line should not be driven low for longer than 15 clock cycles.

IOCK~

The IOCK~ line (60/10) provides I/O and expansion bus parity error information. If the IOCK~ line goes low (active) it indicates that there is an uncorrectable system error.

IOCS16~

The IOCS16~ (40/13) line provides a flag if the transfer to be performed is a 1 wait-state, 16-bit data transfer. IOCS16~ is derived from an address decode to provide the 16-bit chip select signal for I/O cycles. The IOCS16~ line should be driven low (active) with an open collector or tri-state driver that is capable of sinking 20 mA. For memory transfers, refer to MEMCS16~.

IOR~ and IOW~

The IOR~ (pin 60/33) and IOW~ (pin 60/17) lines provide data bus I/O Read and I/O Write flags, respectively. These lines can be driven low (active) by any microprocessor or DMA controller on the I/O or expansion bus. The IOR~ line is driven low to allow an I/O device to drive its data onto the bus. The IOW~ line is driven low to tell an I/O device when to accept data from the bus.

IRQ3-IRQ7, IRQ9-IRQ12, IRQ14, and IRQ15

The IRQ3 through IRQ7, IRQ9 through IRQ12, IRQ14, and IRQ15 lines provide 11 interrupt request lines on the I/O expansion bus. To request service by the microprocessor, I/O devices drive their associated IRQ line high. The requesting device must hold its respective IRQ line high until it is acknowledged by the microprocessor. Table 13-6 lists the interrupt request lines provided by the system, the internal device connected to each interrupt request, the interrupt priority, and the connector pin location. Additional interrupt request information is contained in Chapter 5.

Interrupt request lines IRQ3 through IRQ7 and IRQ9 are used for 8-bit to 8-bit data transfers between I/O adapters and 16-bit or 32-bit memory locations. Interrupt request lines IRQ10 through IRQ12, IRQ14, and IRQ15 are used for 16-bit data transfers between I/O Adapters and 16-bit or 32-bit memory locations. Interrupt requests IRQ0 through IRQ2, IRQ8 and IRQ13 are reserved for system functions and are not available on the I/O or expansion bus.

Table 13-6. Interrupt Request Lines

Interrupt Request	Internal Device	Interface Priority	Bus Width	Expansion Bus Connector/Pin
IRQ0	Timer (8254) Output	*	---	None
IRQ1	Keyboard	*	---	None
IRQ2	Cascade to Slave	*	---	None
IRQ3	Serial Port (COM 2)	7	8-bit	60/30
IRQ4	Modem (COM 1)	8	8-bit	60/26
IRQ5	Reserved	9	8-bit	60/9
IRQ6	Floppy Disk Drive	10	8-bit	60/16
IRQ7	Parallel Printer Port	11(Low)	8-bit	60/19
IRQ8	Real Time Clock	*	---	None
IRQ9	Redirected to IRQ2	1(High)	8-bit	60/25
IRQ10	Reserved	2	16-bit	40/6
IRQ11	Reserved	3	16-bit	40/4
IRQ12	Reserved	4	16-bit	40/2
IRQ13	Math Coprocessor (80287)	*	---	None
IRQ14	Hard Disk Controller	5	16-bit	40/9
IRQ15	Reserved	6	16-bit	40/7

NOTE: The COM 1 and COM 2 serial I/O ports are swappable in some versions of the MS-DOS operating system.

LA17-LA23

The LA17 through LA23 (40-pin connector) lines provide seven additional address bits to increase the maximum addressable memory space from 1M byte to 16M bytes. The seven address lines are gated onto the I/O and expansion bus when BALE (60/12) is high, but since these lines are not latched during microprocessor cycles, they do not remain valid for a whole cycle. LA17 through LA23 are used to generate memory decodes for 1 wait-state memory cycles. The memory decodes should be latched by external I/O adapters on the falling edge of BALE. The seven address bits can be generated by any microprocessor or DMA controller that resides on the I/O or expansion bus. Refer to ADD0-ADD19.

MASTER-

The MASTER- line (40/11) is used with any active DRQn line to acquire control of the I/O and expansion bus. After driving DRQn low and receiving a DACK- to acknowledge, any I/O processor on the I/O or expansion bus can drive MASTER- low (active). With MASTER- driven low, the I/O processor controls the I/O and expansion bus address, data, and control lines. After MASTER- is driven low, the I/O processor must wait one clock cycle before driving the address and data lines, and must wait two clock cycles before issuing Read or Write commands. MASTER- cannot be held low for longer than 15 microseconds because it inhibits REFRESH- and may cause loss of system memory.

MEMCS16-

The MEMCS16- (40/15) line provides a flag if the transfer to be performed is a 1 wait-state, 16-bit data transfer. MEMCS16- is derived from an address decode of LA17 through LA23 to provide the 16-bit chip select signal for memory cycles. The MEMCS16- line should be driven low (active) with an open collector or tri-state driver that is capable of sinking 20 mA. For I/O transfers, refer to IOCS16-.

MEMRD- and SMEMRD-

The MEMRD- (pin 40/29) and SMEMRD- (pin 60/23) lines provide Memory Read flags. The MEMRD- line is driven low (active) on all memory read cycles. The SMEMRD- line is derived from MEMRD- and a decode from the lowest 1M byte of memory. The SMEMRD- line is driven low (active) during a memory read cycle when data from the lowest 1M byte of memory are driven onto the data bus. Before MEMRD- and subsequently SMEMRD- are driven low, a valid address must be on the address bus for one system clock cycle. The MEMRD- and SMEMRD- lines can be driven low (active) by any microprocessor or DMA controller on the I/O or expansion bus.

MEMWR- and SMEMWR-

The MEMWR- (pin 40/27) and SMEMWR- (pin 60/21) lines provide Memory Write flags. The MEMWR- line is driven low (active) on all memory write cycles. The SMEMWR- line is derived from MEMWR- and a decode from the lowest 1M byte of memory. The SMEMWR- line is driven low (active) during a memory write cycle when data from the data bus are stored in the lowest 1M byte of memory. Before MEMWR- and subsequently SMEMWR- are driven low, a valid address must be on the address bus for one system clock cycle. The MEMWR- and SMEMWR- lines can be driven low (active) by any microprocessor or DMA controller on the I/O or expansion bus.

OSC

The OSC (60/31) line provides 14.3 MHz high-speed clock signal. The high-speed clock is not synchronous with the system clock.

REFRESH-

The REFRESH- line (60/18) is driven low (active) by any microprocessor on the I/O or expansion bus to indicate a memory refresh cycle.

RESET

The RESET line (60/20) provides a reset signal that restores system logic to a known state during a system reset or initialization during power-on time. RESET is also generated if the input voltage falls below a preset level.

SMEMRD- and SMEMWR-

Refer to MEMRD- and MEMWR-, respectively.

SWPOWER

The SWPOWER (60/2 and 40/31) line is unique to the GRiD Systems computers. The line comes from the computer power ON/OFF switch and provides switchable power for external peripheral devices. The switchable power level is +8 to +16 Vdc at 2A peak.

SYSCLK-

The SYSCLK- line (60/28) provides a synchronous microprocessor cycle clock signal that should be used for synchronization only. The SYSCLK- line is not suitable for timing fixed frequency applications.

TC

The TC line (60/24) is driven high to indicate the terminal count has been reached for any DMA channel.

ZEROWS-

The ZEROWS- (40/23) line provides the Zero Wait State flag. ZEROWS- is driven low (active) by an I/O device when the microprocessor can complete the current bus cycle without inserting additional wait states. To run a memory cycle without wait states to a 16-bit device, the ZEROWS- signal is derived from an address decode with either a Read or Write command.

To run a memory cycle with a minimum of two wait states to an 8-bit device, ZEROWS- is driven low (active) one clock cycle after an address decode and a Read or Write command goes active. Memory Read and Write commands to an 8-bit device are active on the falling edge of the system clock. The ZEROWS- line should be driven low (active) with an open collector or tri-state driver that is capable of sinking 20 mA.

M12V

The M12V line (60/35) supplies -12 Vdc at 50 mA maximum.

P12V

The P12V line (60/29) supplies +12 Vdc at 50 mA maximum.

+VDC

The +VDC line (60/11 and 40/3) provides +5 Vdc at 200 mA maximum.

NOTE: The average power drawn from all power lines must not exceed 5- Watts total. The power lines include SWPOWER, M12V, P12V, and +VDC.

Signal Ground

Two Signal Ground lines are provided for each connector (40/1, 40/5, 60/13, and 60/15).

Spare

A spare (unused) line is provided at 40/8.

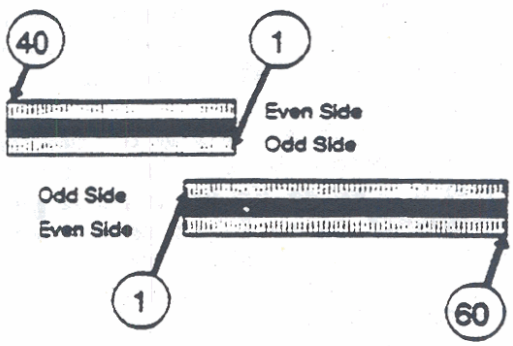


Figure 13-1. Expansion Bus Connector Pin Layout

APPENDIX A: IBM AT COMPATIBILITY

The hardware and software interfaces provided in the IBM™ Personal Computers (IBM AT) are recognized as industry standards. Therefore, a large volume of application programs have been written to operate with these interfaces. The GRiDCase 1500 Series Computer is hardware and software compatible with the IBM AT. Therefore, any hardware or software designed to run on the IBM AT should run equally well on the GRiDCase 1500 Series computer.

NOTE: Throughout this chapter, hexadecimal values are given with a letter "h" suffix.

The GRiDCase 1500 Series computer is compatible with the IBM AT while providing additional capabilities that are not found in the IBM AT. These additional capabilities are unique to GRiD Systems computers. Many of the addtional capabilities that the GRiDCase 1500 Series Computer provides can be enabled/disabled by using a special case of ROM-BIOS service routines for interrupt 15h. These special BIOS Subsystem Functions provide additional control of the system ROM, Display Subsystem, and System Configuration (refer to Chapter 3).

The following paragraphs summarize the GRiDCase 1500 Series Computer additional capabilities to the IBM AT standard.

GENERAL SYSTEM DIFFERENCES

- A cassette interface is not supported.
- A game controller is not supported.
- A light pen interface is not supported.

DIRECT MEMORY ACCESS (DMA)

For GRiDCase 1500 Series computer operations, only the single-transfer mode is used for DMA data transfers. In single-transfer mode, up to 512k bytes can be transferred, but an interrupt and

™IBM is a Trademark of the IBM Corporation.

acknowledge sequence is required for each byte. The DMA controller keeps track of how many bytes were programmed to transfer and how many were actually transferred (refer to Chapter 4).

DISPLAY SUBSYSTEM

Most of the modes of the IBM AT color card are supported by the GRiDCase 1500 Series Computer Display subsystem. The following operating modes of the IBM color card are not supported:

1. The low resolution 160x200 Graphics mode. The IBM AT documentation makes no mention of how to use the 160x200 Graphics mode and the IBM ROM-BIOS does not support it. The GRiDCase 1500 Series Computer hardware architecture is very similar to the IBM AT. Therefore, any applications that use the 160x200 Graphics mode, and are designed to run on the IBM AT, should not experience difficulty running on the GRiDCase 1500 Series Computer.

2. Interlace modes of the 6845. IBM does not use interlace mode on either of their display boards. Also, when the interlace sync mode is selected on the color card with the IBM color monitor the display appears to vibrate.

3. Display Parameters and Graphic Character Extensions invoked via software interrupts 1Dh and 1Fh, respectively. These functions are implemented by the display controller.

The Yamaha V6366 Display Controller emulates all of the features and functions found in the 6845 Display Controller used by the IBM AT. In addition the V6366 provides Preset Data registers for switching between different software and monitor combinations. Additional registers are also supplied for up to 16 color palettes and discrete control functions. Additional display subsystem information is contained in Chapter 7.

KEYBOARD SUBSYSTEM

The GRiDCase 1500 Series computers support a built-in keyboard with 72-keys that emulates all 84-keys of the IBM PC standard keyboard. In addition, an external IBM AT compatible 84-key keyboard, enhanced 101/102-key keyboard, or numerical keypad is supported via a 6-pin DIN connector. Additional keyboard subsystem information is contained in Chapter 8.

DISK DRIVE SUBSYSTEM

The GRiDCase 1500 Series computer supports two internal 1.4M byte, 3.5-inch floppy disk drives and an external peripheral port for connecting an external drive. Operation of all floppy disk drives is compatible with the IBM AT except that the Floppy Disk Parameters (interrupt 1Eh) and the revectoring interrupt 40h are not supported.

Depending upon the installed options, the one or more floppy disk drives can be replaced by a hard disk drive. The hard disk drive options provide either IBM XT compatible or IBM AT compatible disk drives. The XT compatible drives support 8-bit wide DMA data transfers. The AT compatible drives support 16-bit wide Programmed I/O (PIO) data transfers.

Additional information for the floppy disk drives and IBM XT compatible hard disk drives is contained in Chapter 9. Additional information for the IBM AT compatible hard disk drives is contained in Chapter 10.

PARALLEL PRINTER PORT SUBSYSTEM

The parallel port control logic contains nine registers. One group of five registers is used for printer operation and system status checks. A second group of three registers supports a parallel input/output port interface.

Registers at I/O addresses 378h through 37Ah, 37Ch, and 37Eh are used to control and monitor printer operation. In the GRiDCase 1500 Series Computer, the three least-significant bits of the I/O register at address 379h (bits 2-0), and the three most-significant bits of the I/O register at address 37Ah (bits 7-5), are used to provide system status information that is not related to operation of the printer.

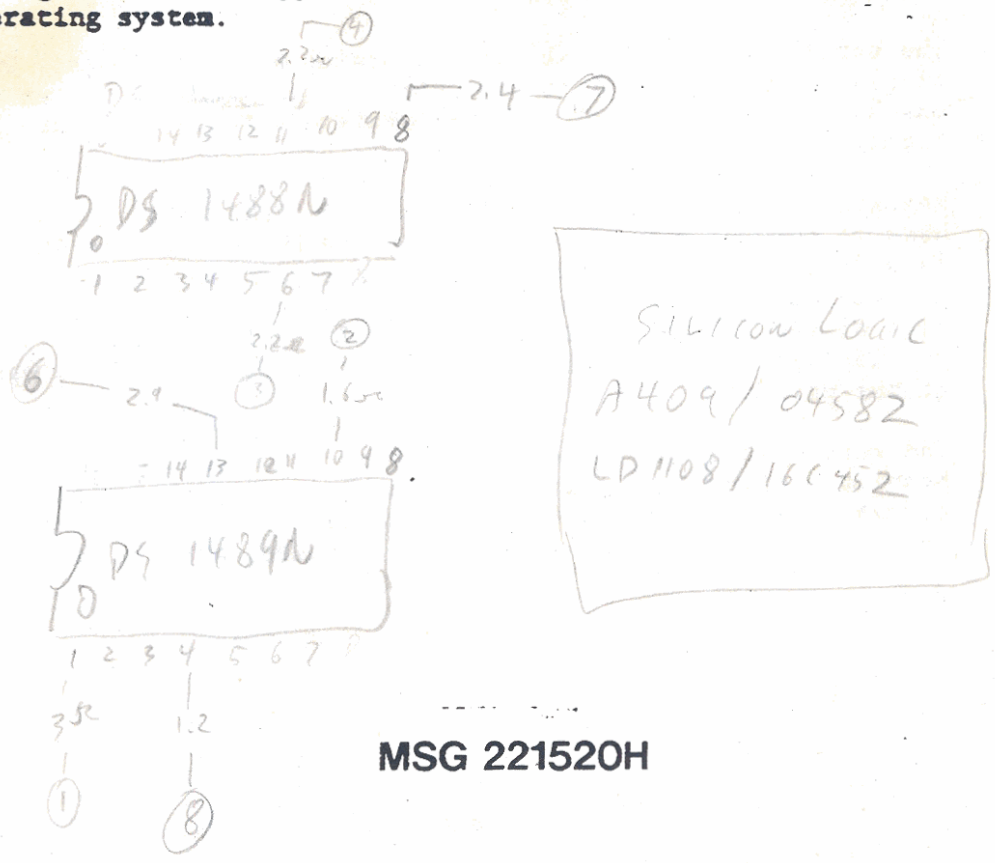
The second group of three registers at I/O addresses FF8h-FFFh provide bidirectional parallel data transfers. Parallel I/O data transfers are unique to the GRiDCase computers. This group of registers is accessed only when I/O register 423h is set to "1."

SERIAL I/O PORT SUBSYSTEM

The GRiDCase 1500 Series computer provides an RS-232-C asynchronous serial I/O port and an optional internal modem. In the Model 1520 (80C286 Microprocessor) computers, the serial I/O port and optional modem are controlled by two separate 82C50 Universal Asynchronous Receiver-Transmitters (UARTs). The serial I/O port and modem for the Model 1530 (80386 Microprocessor) are controlled by a VL16C452 Dual Asynchronous Communications Element (DACE). The DACE is designed to simultaneously interface with two serial I/O devices and is equivalent to the two 82C50 UARTs used by the Model 1520 computers.

The internal modem is available as either of two options: Option 330 operates at 1200/300 Baud and Option 331 operates at 2400/1200/300 Baud. Both options are Bell 212A and 103 compatible and Option 331 is also V.22 bis- compatible. Both modems are also command compatible with the Hayes 1200 Smartmodem.

The internal modem and the serial I/O port use separate I/O channels so that both devices can operate simultaneously. In the default configuration, the internal modem is configured as COM1 and the serial I/O port is configured as COM2. This serial port configuration is swappable with some versions of the MS-DOS operating system.



MSG 221520H