THE FOLLOWING PARAGRAPH DOES NOT APPLY TO THE UNITED KINGDOM OR ANY COUNTRY WHERE SUCH PROVISIONS ARE INCONSISTENT WITH LOCAL LAW:

International Business Machines Corporation provides this manual "as is," without warranty of any kind, either express or implied, including, but not limited to, the particular purpose. IBM may make improvements and changes in the products and the programs described in this manual at any time.

This product could include technical inaccuracies or typographical errors. Changes are periodically made to the information herein; these changes will be incorporated in new editions of the publication.

It is possible that this material may contain reference to, or information about, IBM products (machines and programs), programming, or services that are not announced in your country. Such references or information must not be construed to mean that IBM intends to announce such IBM products, programming, or services in your country.

Products are not stocked at the address below. Requests for copies of the product and for technical information about the system should be made to your authorized IBM Personal Computer Dealer.

THE FOLLOWING PARAGRAPH APPLIES ONLY TO THE UNITED STATES AND PUERTO RICO: A form for reader's comments is provided at the back of this publication. If the form has been removed, comments may be addressed to IBM Corporation, Department 95H, 11400 Burnet Road, Austin, Texas 78758, U.S.A. IBM may use or distribute whatever information you supply in any way it believes appropriate without incurring any obligation to you.

All specifications are subject to change without notice.

©Copyright International Business Machines 1986
Preface

The IBM PC Convertible Technical Reference consists of two volumes. Volume 1 describes the hardware design and provides interface information for the IBM PC Convertible. Volume 1 also has information about the basic input/output system (BIOS) and programming support. Volume 2 contains the BIOS listings.

The information in these volumes is both descriptive and reference-oriented and is intended for hardware and software designers, programmers, engineers, and other interested persons who need to understand the design and operation of the IBM PC Convertible. These users should be familiar with the use of the IBM PC Convertible and understand the concepts of computer architecture and programming.

Volume 1 has five sections:

- Section 1, “Introduction” is an overview of the system and the available options.

- Section 2, “System Unit” describes each functional part of the base system. This section also contains the description of the interfaces. Programming considerations are supported by command code and register descriptions.

- Section 3, “System Options” describes each available option.

- Section 4, “System BIOS and Usage” describes the basic input/output system (BIOS) and its use. This section also contains the software interrupt listing, a system memory map, descriptions of vectors with special meanings, and a set of low-storage maps. In addition, Section 4 describes keyboard encoding and usage.

- Section 5, “Compatibility with the IBM Personal Computer Family” describes programming concerns for maintaining compatibility among the IBM PC Convertible system and the other IBM Personal Computers.
Volume 1 has the following appendixes:

- Appendix A, "Characters and Keystrokes"
- Appendix B, "Unit Specifications"
- Appendix C, "Logic Timing Diagrams"
- Appendix D, "Power-On Self Test Error Codes".

Volume 2 contains the BIOS listing and is to be used in conjunction with Volume 1.

**Prerequisite Publications**

*Guide to Operations*: for the IBM PC Convertible.

**Suggested Reading**

- *BASIC* for the IBM Personal Computer
- *Disk Operating System*
- *Hardware Maintenance and Service* for the IBM PC Convertible
- *MACRO Assembler* for the IBM Personal Computer.
Contents

Section 1. Introduction ............................................. 1-1

Section 2. System Unit ............................................. 2-1
  Processor ...................................................... 2-10
  System Clock ................................................. 2-11
  System Timer .................................................. 2-12
  Interrupt Controller ........................................ 2-16
  Direct Memory Access (DMA) Controller ....................... 2-24
  Read-Only Memory (ROM) ..................................... 2-32
  Random Access Memory (RAM) ................................ 2-33
  I/O Channel .................................................... 2-36
  Liquid Crystal Display (LCD) .................................. 2-45
  Liquid Crystal Display (LCD) Controller ....................... 2-46
  Keyboard and Keyboard Controller ............................ 2-62
  Audio Controller and Speaker ................................ 2-72
  Diskette Drive .................................................. 2-74
  Diskette Controller .......................................... 2-75
  Portable Printer Interface .................................... 2-84
  Real-Time Clock ............................................... 2-88
  Power Supply ................................................... 2-93

Section 3. System Options ........................................... 3-1
  Serial/Parallel Adapter (Serial Interface) ...................... 3-3
  Serial/Parallel Adapter (Parallel Interface) ................... 3-19
  Internal Modem ............................................... 3-28
  CRT Display Adapter .......................................... 3-53
  Portable Printer .............................................. 3-68
  Monochrome Display .......................................... 3-89
  Color Display ................................................... 3-90
  Automobile Power Adapter .................................... 3-90
  Battery Charger ................................................ 3-90

Section 4. System BIOS and Usage ................................. 4-1
  ROM BIOS ......................................................... 4-3
  Use of BIOS ....................................................... 4-3
  Reserved Interrupt Allocations ................................ 4-21
  Other Reserved Areas ......................................... 4-23
  Adapters with System-Accessible ROM .......................... 4-24
BIOS Programming Guidelines ......................... 4-26
Keyboard Encoding and Usage .......................... 4-27
Special BIOS Functions ................................. 4-37

Section 5. Compatibility with the IBM Personal Computer
Family ............................................... 5-1
  Compatibility Overview ............................. 5-2
  Special Programming Considerations .............. 5-3
  Timing Dependencies ................................ 5-6
  Unequal Configurations ............................ 5-6
  Hardware Differences ............................... 5-6
  Summary .......................................... 5-14

Appendix A. Character Sets and Keystrokes ........... A-1
  Character Set Values ............................... A-2
  Color Attribute Mapping ........................... A-5
  LCD Attribute Mapping .............................. A-6

Appendix B. Unit Specifications ........................ B-1
  System Unit ........................................ B-2
  Diskette Drives ................................... B-6
  Random Access Memory ............................. B-8
  Serial/Parallel Adapter ............................ B-9
  Portable Printer .................................... B-11
  CRT Display Adapter ................................ B-15
  Monochrome Display ................................ B-18
  Color Display ...................................... B-20
  Internal Modem ..................................... B-22
  AC Adapter (100-240 Vac) ......................... B-24
  Battery Charger .................................... B-25
  Automobile Adapter ................................. B-25

Appendix C. Logic Timing Diagrams ................. C-1

Appendix D. Power-On Self Test Error Codes ....... D-1

Glossary ............................................ X-1

Index ................................................ X-23
Section 1. Introduction

The IBM PC Convertible is a battery-powered, portable member of the IBM Personal Computer family. The system unit is the center of the IBM PC Convertible and contains the processor, diskette drives, keyboard, display, internal memory, power supply, and battery pack. An ac adapter for powering the system unit and recharging the battery is included with the system unit.

Optional features can be added to the IBM PC Convertible system unit. These features are either installed internally or connected to the system unit by using the appropriate connectors and cables. The features provide additional function or capability to the system unit. The following options are available for the system unit:

- 128K-byte memory card (expansion up to 512K bytes)
- Internal modem
- Portable Printer
- Serial/Parallel Adapter
- Automobile Power Adapter
- Battery Charger
- CRT Display Adapter
- Monochrome Display and Color Display
- Cables.
Section 2. System Unit

Processor .................................................. 2-10
System Clock ............................................... 2-11
  Programming Considerations ....................... 2-11
System Timer ............................................... 2-12
  Programming Considerations ....................... 2-12
Interrupt Controller .................................... 2-16
  Programming Considerations ....................... 2-18
Direct Memory Access (DMA) Controller .......... 2-24
  Programming Considerations ....................... 2-24
Read-Only Memory (ROM) ............................... 2-32
Random Access Memory (RAM) ......................... 2-33
I/O Channel ............................................... 2-36
  I/O Channel Interface ............................... 2-37
Liquid Crystal Display (LCD) ......................... 2-45
Liquid Crystal Display (LCD) Controller ........ 2-46
  LCD Interface ......................................... 2-54
  Programming Considerations ....................... 2-57
Keyboard and Keyboard Controller ............... 2-62
  Programming Considerations ....................... 2-67
Audio Controller and Speaker ..................... 2-72
  Programming Considerations ....................... 2-73
Diskette Drive ........................................... 2-74
Diskette Controller ..................................... 2-75
  Diskette Driver Interface ......................... 2-76
  Programming Considerations ....................... 2-80
Portable Printer Interface ........................... 2-84
  Portable Printer Interface ......................... 2-84
  Programming Considerations ....................... 2-85
Real-Time Clock .......................................... 2-88
  Programming Considerations ....................... 2-89
Power Supply ............................................. 2-93
  Power Supply Interface ............................. 2-93
Notes:

2-2 System Unit
The major components of the IBM PC Convertible system unit are:

- Processor
- System clock
- System timer
- Interrupt controller
- Direct-memory-access (DMA) controller
- Read-only memory (ROM)
- Random-access (read-write) memory (RAM)
- Input/output (I/O) channel
- Liquid crystal display (LCD)
- LCD controller
- Keyboard
- Keyboard controller
- Audio controller and speaker
- Diskette drive
- Diskette controller
- Portable printer interface
- Real-time clock
- Power supply.

Figure 2-1 on page 2-4 shows an overview of the functional units.
Many of the components are installed on the system board. Figure 2-2 on page 2-5 shows the major components on the system board.

2-4 System Unit
Interrupt Controller, Keyboard Controller, Audio Controller, System Clock, and I/O Controller

Figure 2-2. System Board Components
Figure 2-3 shows the memory mapping used on the IBM PC Convertible. Figure 2-4 on page 2-7 shows the I/O and storage addresses for the IBM PC Convertible.

The input/output registers at addresses hex 060, 061, and 062 are used by some of the components to communicate with the processor. Figure 2-5 on page 2-8 shows the meaning of the bits in these registers.

The system control registers in the hex 07x range provide special internal functions on the IBM PC Convertible. Figure 2-6 on page 2-9 shows the usage of these registers. Refer to the BIOS listings in Volume 2 for further information concerning these registers.

<table>
<thead>
<tr>
<th>Address (hex)</th>
<th>Length (K-bytes)</th>
<th>Actual (K-bytes)</th>
<th>Usage</th>
</tr>
</thead>
<tbody>
<tr>
<td>00000</td>
<td>640</td>
<td>512</td>
<td>System RAM</td>
</tr>
<tr>
<td>A0000</td>
<td>128</td>
<td>16</td>
<td>Display RAM</td>
</tr>
<tr>
<td>C0000</td>
<td>32</td>
<td>0</td>
<td>Video Feature ROM</td>
</tr>
<tr>
<td>C8000</td>
<td>160</td>
<td>0</td>
<td>I/O Feature ROM</td>
</tr>
<tr>
<td>F0000</td>
<td>64</td>
<td>64</td>
<td>System ROM</td>
</tr>
</tbody>
</table>

Figure 2-3. System Memory Map
<table>
<thead>
<tr>
<th>Range (hex)</th>
<th>Usage</th>
</tr>
</thead>
<tbody>
<tr>
<td>000-00F</td>
<td>DMA control</td>
</tr>
<tr>
<td>020-021</td>
<td>Interrupt control</td>
</tr>
<tr>
<td>040-043</td>
<td>System timer control</td>
</tr>
<tr>
<td>060-062</td>
<td>Input/output registers</td>
</tr>
<tr>
<td>070-080</td>
<td>System control registers</td>
</tr>
<tr>
<td>081-083</td>
<td>DMA page registers</td>
</tr>
<tr>
<td>0A0</td>
<td>I/O channel check mask register</td>
</tr>
<tr>
<td>2F8-2FF</td>
<td>Asynchronous communications (RS-232), secondary</td>
</tr>
<tr>
<td>378-37F</td>
<td>Parallel printer</td>
</tr>
<tr>
<td>3B0-3BF</td>
<td>LCD controller (monochrome mode)</td>
</tr>
<tr>
<td>3D0-3DF</td>
<td>LCD controller (color/graphics mode) or CRT display adapter</td>
</tr>
<tr>
<td>3F0-3F7</td>
<td>Diskette controller</td>
</tr>
<tr>
<td>3F8-3FF</td>
<td>Modem or asynchronous communications adapter, primary</td>
</tr>
</tbody>
</table>

Figure 2-4. I/O Address Map
<table>
<thead>
<tr>
<th>Address (hex)</th>
<th>Bit</th>
<th>Usage</th>
</tr>
</thead>
<tbody>
<tr>
<td>060</td>
<td>7</td>
<td>I/O register 1:</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Keyboard scan code 7</td>
</tr>
<tr>
<td></td>
<td>6</td>
<td>Keyboard scan code 6</td>
</tr>
<tr>
<td></td>
<td>5</td>
<td>Keyboard scan code 5</td>
</tr>
<tr>
<td></td>
<td>4</td>
<td>Keyboard scan code 4</td>
</tr>
<tr>
<td></td>
<td>3</td>
<td>Keyboard scan code 3</td>
</tr>
<tr>
<td></td>
<td>2</td>
<td>Keyboard scan code 2</td>
</tr>
<tr>
<td></td>
<td>1</td>
<td>Keyboard scan code 1</td>
</tr>
<tr>
<td></td>
<td>0</td>
<td>Keyboard scan code 0</td>
</tr>
<tr>
<td>061</td>
<td>7</td>
<td>I/O register 2:</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Clear keyboard</td>
</tr>
<tr>
<td></td>
<td>6</td>
<td>Reserved</td>
</tr>
<tr>
<td></td>
<td>5</td>
<td>Disable I/O Channel Check NMI</td>
</tr>
<tr>
<td></td>
<td>4</td>
<td>Reserved</td>
</tr>
<tr>
<td></td>
<td>3</td>
<td>Disable real-time clock NMI</td>
</tr>
<tr>
<td></td>
<td>2</td>
<td>Enable speaker output</td>
</tr>
<tr>
<td></td>
<td>1</td>
<td>Speaker data</td>
</tr>
<tr>
<td></td>
<td>0</td>
<td>Timer 2 gate (speaker)</td>
</tr>
<tr>
<td>062</td>
<td>7</td>
<td>I/O register 3:</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Diskette power-on request NMI</td>
</tr>
<tr>
<td></td>
<td>6</td>
<td>I/O channel check NMI</td>
</tr>
<tr>
<td></td>
<td>5</td>
<td>Timer channel-2 out</td>
</tr>
<tr>
<td></td>
<td>4</td>
<td>Keyboard clear NMI</td>
</tr>
<tr>
<td></td>
<td>3</td>
<td>System Suspend NMI</td>
</tr>
<tr>
<td></td>
<td>2</td>
<td>Real-time clock alarm NMI</td>
</tr>
<tr>
<td></td>
<td>1</td>
<td>Reserved</td>
</tr>
<tr>
<td></td>
<td>0</td>
<td>Keyboard data NMI</td>
</tr>
</tbody>
</table>

Figure 2-5. I/O Registers

2-8 System Unit
<table>
<thead>
<tr>
<th>Address (hex)</th>
<th>R/W</th>
<th>Usage</th>
</tr>
</thead>
<tbody>
<tr>
<td>070-071</td>
<td>R/W</td>
<td>Real-time clock control</td>
</tr>
<tr>
<td>072</td>
<td>R/W</td>
<td>Sleep clock control/interrupt diagnostics</td>
</tr>
<tr>
<td></td>
<td></td>
<td>global NMI enable</td>
</tr>
<tr>
<td>074-075</td>
<td>R/W</td>
<td>LCD control/diagnostics</td>
</tr>
<tr>
<td>077</td>
<td>R/W</td>
<td>Diskette control/diagnostics</td>
</tr>
<tr>
<td>078-07A</td>
<td>R/W</td>
<td>Portable printer control</td>
</tr>
<tr>
<td>07C</td>
<td>R/W</td>
<td>Keyboard and feature control</td>
</tr>
<tr>
<td>07D</td>
<td>R/W</td>
<td>Keyboard internal scan code (read)/diagnostics (write)</td>
</tr>
<tr>
<td>07F</td>
<td>R/W</td>
<td>Power system control</td>
</tr>
</tbody>
</table>

Figure 2-6. System Control Registers
Processor

The processor used on the IBM PC Convertible is directly compatible with the Intel\textsuperscript{1} 8088 Microprocessor, and it is compatible with the instruction set used for the Intel 8086 Microprocessor family. The processor on the IBM PC Convertible uses a 16-bit internal architecture with an 8-bit data bus interface. This processor has a 20-bit address bus that supports a 1-megabyte address space.

The internal circuitry of the processor is static and the internal registers, counters, and latches do not require continuous clocking for refresh.

The processor operates at a 4.77 MHz clock rate.

\textsuperscript{1}Registered trade mark of Intel Corporation.

2-10 System Unit
System Clock

The system clock consists of a continuously running clock and a software-controlled clock. The continuously running clock provides continuous timing pulses at the maximum clock frequency. The software-controlled clock (the sleep clock) is used to operate the processor and other system components that do not require constant operation. The sleep clock can be stopped using a BIOS function call that causes the processor to enter standby mode to conserve battery power.

The 4.77 MHz processor clock rate is derived by dividing a 14,3181-MHz oscillator frequency by 3. The clock period is 210 nanoseconds. The clock has a 33% duty cycle.

Programming Considerations

The system clock supports a sleep mode that stops the system clock when the system is waiting on some nonprocessor event to occur. The wait on external event BIOS function call (interrupt hex 15) is used to access the sleep function. BIOS automatically invokes the sleep function for the keyboard and diskette functions. When the system is in sleep mode, interrupts are processed as normal. After each interrupt is serviced, control is returned to the sleep function to determine if the requested event has occurred. If the event has occurred, control is returned to the application; otherwise, sleep mode is re-entered.
System Timer

The system timer provides functions and modes similar to those provided by an Intel 8253 Programmable Interval Timer. The timer provides two programmable timer channels. Timer 0 is used as a general-purpose and software-interrupt timer. Timer 0 does not support modes 1 and 5. Timer 2 is used to support the tone generation for the audio speaker.

The timer channel clock rate is derived by dividing a 4.77-MHz clock frequency by 4. Each channel has a minimum timing resolution of 840 nanoseconds.

Programming Considerations

The timers are programmed by writing a control word and loading the initial count. The control word must be written before the initial count, and the count must follow the count format specified in the control word. The timers are countdown counters.

Count values are loaded into either address hex 040 (timer 0) or hex 042 (timer 2). If the count value is only 1 byte long, the remaining byte is set to zero. Figure 2-7 on page 2-13 shows the timer control registers.

The count value can be read by using the timer latch command and reading the data at the appropriate timer register. This command does not affect the operation of the timer.

The power-on routines initializes timer 0 to mode 3 which provide an interrupt rate of 18.2 times per second. The power-on routines do not initialize the audio timer (channel 2).
<table>
<thead>
<tr>
<th>Address (hex)</th>
<th>R/W</th>
<th>Usage</th>
</tr>
</thead>
<tbody>
<tr>
<td>040</td>
<td>R/W</td>
<td>Timer 0 count load/Read</td>
</tr>
<tr>
<td>042</td>
<td>R/W</td>
<td>Timer 2 count load/Read</td>
</tr>
<tr>
<td>043</td>
<td>R/W</td>
<td>Timer control word 0 and 2</td>
</tr>
<tr>
<td>061</td>
<td>R</td>
<td>I/O register 2</td>
</tr>
<tr>
<td>062</td>
<td>R</td>
<td>I/O register 3</td>
</tr>
</tbody>
</table>

Figure 2-7. Timer Control Registers
Timer Control Word (Hex 043)

**Bit Meaning**

**7-6** Timer control word:

- 00 = Timer 0 control word
- 10 = Timer 2 control word
- x1 = Used for system suspend function

**5-4** Latch command or counter read/write control:

- 00 = Timer latch command
- 01 = Read/write least significant byte (LSB)
- 10 = Read/write most significant byte (MSB)
- 11 = Read/write LSB first, then MSB

**3-1** Mode control:

- 000 = Interrupt on terminal count (mode 0)
- 001 = Programmable one shot (mode 1)
- x10 = Rate generator (mode 2)
- x11 = Square wave (mode 3)
- 100 = Software triggered strobe (mode 4)
- 101 = Hardware triggered strobe (mode 5)

**0** Count mode:

- 1 = Binary coded decimal
- 0 = Binary
I/O Register 2 (Hex 061)

Only a portion of I/O register 2 is used by the system timer. The following describes that portion of the register used by the system timer. For a complete description of the register, see Figure 2-5 on page 2-8.

<table>
<thead>
<tr>
<th>Bit</th>
<th>Meaning</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>Timer-2 gate. This bit is used to enable timer 2. Setting this bit to 0 stops the timer.</td>
</tr>
</tbody>
</table>

I/O Register 3 (Hex 062)

Only a portion of I/O register 3 is used by the system timer. The following describes that portion of the register used by the system timer. For a complete description of the register, see Figure 2-5 on page 2-8.

<table>
<thead>
<tr>
<th>Bit</th>
<th>Meaning</th>
</tr>
</thead>
<tbody>
<tr>
<td>5</td>
<td>Timer channel 2 out. Sensing this bit enables the application to monitor the output of timer 2.</td>
</tr>
</tbody>
</table>
Interrupt Controller

The interrupt controller is fully programmable and uses an interface and command set that are compatible with the Intel 8259 Programmable Interrupt Controller.

The interrupt controller does not support rotating priority. Additionally, when programmed in the special mask mode, the current level must be masked as soon as it is entered.

The power-on routines set the controller to edge-triggered mode with the interrupt vectors assigned to processor vectors 8 through 16. Each interrupt requires an end of interrupt command to reset the interrupt service.

The interrupt controller supports nine levels of interrupts; there are one nonmaskable interrupt (NMI) level and eight maskable interrupt levels. Figure 2-8 on page 2-17 shows the hardware interrupt levels in descending order of priority. Interrupt levels 2-7 are available at the system I/O channel. Interrupt levels 0, 1, 6, and 7 are used on the system board.
<table>
<thead>
<tr>
<th>Interrupt</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>NMI</td>
<td>I/O channel check, diskette power-on request, keyboard, real-time clock alarm, or system suspend</td>
</tr>
<tr>
<td>0</td>
<td>Timer output 0</td>
</tr>
<tr>
<td>1</td>
<td>Keyboard (output buffer full)</td>
</tr>
<tr>
<td>2</td>
<td>Reserved</td>
</tr>
<tr>
<td>3</td>
<td>Async (secondary)</td>
</tr>
<tr>
<td>4</td>
<td>Async (primary) or modem</td>
</tr>
<tr>
<td>5</td>
<td>Reserved</td>
</tr>
<tr>
<td>6</td>
<td>Diskette controller</td>
</tr>
<tr>
<td>7</td>
<td>Printers</td>
</tr>
</tbody>
</table>

Figure 2-8. Hardware Interrupt Levels
Programming Considerations

The interrupt controller accepts two types of command words from the processor. These are the initialization command words (ICWs) and the operation command words (OCWs). Before normal operation can begin, the controller must be initialized by writing a sequence of ICWs. Once this has been done, the controller can accept interrupt requests. The application can then select mask priorities and other operating modes by writing OCWs.

When initialization command word 1 is written, the next two write operations to address hex 21 sequentially go to initialization command word 2 and initialization command word 4. Initialization command word 3 is not used on the IBM PC Convertible, because the IBM PC Convertible does not support cascading of interrupts.

Figure 2-9 on page 2-19 shows the register interface used for the vectored interrupts. The NMI function uses portions of other control registers. Figure 2-10 on page 2-20 shows these registers.
<table>
<thead>
<tr>
<th>Address (hex)</th>
<th>R/W</th>
<th>Usage</th>
</tr>
</thead>
<tbody>
<tr>
<td>020</td>
<td>W</td>
<td>ICW1</td>
</tr>
<tr>
<td></td>
<td>W</td>
<td>OCW2</td>
</tr>
<tr>
<td></td>
<td>W</td>
<td>OCW3</td>
</tr>
<tr>
<td></td>
<td>R</td>
<td>Interrupt request register</td>
</tr>
<tr>
<td></td>
<td>R</td>
<td>Interrupt in-service register</td>
</tr>
<tr>
<td></td>
<td>R</td>
<td>Polling data byte</td>
</tr>
<tr>
<td>021</td>
<td>W</td>
<td>ICW2</td>
</tr>
<tr>
<td></td>
<td>W</td>
<td>ICW4</td>
</tr>
<tr>
<td></td>
<td>W</td>
<td>OCW1</td>
</tr>
<tr>
<td></td>
<td>R</td>
<td>Interrupt mask register</td>
</tr>
<tr>
<td>063</td>
<td>W</td>
<td>Interrupt simulation register (diagnostics)</td>
</tr>
<tr>
<td></td>
<td>R</td>
<td>Interrupt control diagnostic register 0</td>
</tr>
<tr>
<td></td>
<td>R</td>
<td>Interrupt control diagnostic register 1</td>
</tr>
<tr>
<td>072</td>
<td>R/W</td>
<td>Interrupt controller diagnostic control</td>
</tr>
<tr>
<td>0A0</td>
<td>R</td>
<td>Interrupt control diagnostic sense</td>
</tr>
</tbody>
</table>

Figure 2-9. Vectored Interrupt Control Registers
<table>
<thead>
<tr>
<th>Address (hex)</th>
<th>R/W</th>
<th>Usage</th>
</tr>
</thead>
<tbody>
<tr>
<td>061</td>
<td>R/W</td>
<td>I/O register 3</td>
</tr>
<tr>
<td>062</td>
<td>R</td>
<td>I/O register 3</td>
</tr>
<tr>
<td>072</td>
<td>R/W</td>
<td>Sleep clock control/Interrupt diagnostics register, global NMI enable</td>
</tr>
<tr>
<td>077</td>
<td>R/W</td>
<td>Diskette control register</td>
</tr>
<tr>
<td>07C</td>
<td>R/W</td>
<td>Keyboard control register</td>
</tr>
<tr>
<td>07F</td>
<td>R/W</td>
<td>Power system control register</td>
</tr>
<tr>
<td>0A0</td>
<td>R/W</td>
<td>I/O channel check mask register</td>
</tr>
</tbody>
</table>

Figure 2-10. NMI Control Registers

I/O Register 2 (Hex 061)

The interrupt function uses only a portion of the register at hex 61. Only the bit or bits used by this function are described here; the remaining bits are described with the function that they support. The following describes that portion. For a complete description of the register, see Figure 2-5 on page 2-8.

**Bit Meaning**

7 Clear keyboard. The keyboard handler toggles this bit after the keyboard scan code has been read from the I/O register at address hex 60. This generates an NMI.

5 Disable I/O channel check NMI. Setting this bit to 1 inhibits NMIs generated from the I/O channel check condition.

3 Disable real-time clock NMI. Setting this bit to 1 inhibits NMIs generated from the real-time clock.
I/O Register 3 (Hex 062)

Only a portion of I/O register 3 is used by the interrupt controller. The following describes that portion. For a complete description of the register, see Figure 2-5 on page 2-8. The following bits are sensed by the BIOS NMI handler to determine the cause of an NMI. The global NMI mask bit at address hex 072 does not affect the setting of these bits.

<table>
<thead>
<tr>
<th>Bit</th>
<th>Meaning</th>
</tr>
</thead>
<tbody>
<tr>
<td>7</td>
<td>Diskette power on request NMI</td>
</tr>
<tr>
<td>6</td>
<td>I/O channel check NMI</td>
</tr>
<tr>
<td>4</td>
<td>Keyboard clear command NMI</td>
</tr>
<tr>
<td>3</td>
<td>System suspend NMI</td>
</tr>
<tr>
<td>2</td>
<td>Real-time clock alarm NMI</td>
</tr>
<tr>
<td>0</td>
<td>Keyboard data latched NMI</td>
</tr>
</tbody>
</table>

Sleep Clock Control/Interrupt Diagnostics (Hex 072)

The interrupt function uses only a portion of the register at hex 072. Only the bit or bits used by this function are described here; the remaining bits are described with the function that they support.

<table>
<thead>
<tr>
<th>Bit</th>
<th>Meaning</th>
</tr>
</thead>
<tbody>
<tr>
<td>5</td>
<td>Global NMI enable. Setting this bit to 0 disables all NMIs. If NMIs remain disabled for over 5 milliseconds, data from the keyboard may be lost. Applications should avoid disabling NMIs.</td>
</tr>
</tbody>
</table>
Diskette Control/Diagnostics (Hex 077)

The interrupt function uses only a portion of the register at hex 077. Only the bit or bits used by this function are described here; the remaining bits are described with the function that they support.

<table>
<thead>
<tr>
<th>Bit</th>
<th>Meaning</th>
</tr>
</thead>
<tbody>
<tr>
<td>7</td>
<td>Enable diskette controller power-on NMI. Setting this bit to 0 inhibits NMIs generated when the diskette controller is accessed in the powered-off condition. Accessing the diskette controller with this bit set to 0 may cause undetermined diskette errors.</td>
</tr>
</tbody>
</table>

Keyboard Control Register (Hex 07C)

The interrupt function uses only a portion of the register at hex 07C. Only the bit or bits used by this function are described here; the remaining bits are described with the function that they support.

<table>
<thead>
<tr>
<th>Bit</th>
<th>Meaning</th>
</tr>
</thead>
<tbody>
<tr>
<td>7</td>
<td>Enable keyboard NMI. Setting this bit to 0 inhibits NMIs generated from the keyboard. If this NMI remains disabled for over 5 milliseconds, data from the keyboard may be lost.</td>
</tr>
</tbody>
</table>
Power System Control (Hex 07F)

The interrupt function uses only a portion of the register at hex 07F. Only the bit or bits used by this function are described here; the remaining bits are described with the function that they support.

**Bit  Meaning**

2  Enable system suspend NMI. Setting this bit to 0 inhibits NMIs generated by a system power off. If the system is powered off with this bit set to 0, the state of the system is not saved and the application cannot be resumed during the next power-on sequence.

1  Request power off. Setting this bit to 1 causes a system suspend NMI to be generated that causes a system power off. This bit should be accessed through a BIOS function call (interrupt hex 15).

I/O Channel Check Mask Register (Hex 0A0)

**Bit  Meaning**

7  Enable I/O channel check condition NMI. Setting this bit to 1 enables NMIs generated by an I/O channel check condition.
Direct Memory Access (DMA) Controller

The direct memory access controller provides functions similar to those provided by the Intel 8237 Programmable Direct Memory Access Controller. The IBM PC Convertible DMA controller supports three DMA channels on the I/O channel. Memory refresh using the DMA controller is not supported because the system RAM does not require refresh.

The DMA method allows certain I/O devices to transfer information to or from memory without using the processor. The DMA channels are accessed by the I/O devices through commands sent over the I/O channel.

The DMA controller supports single, block, and demand transfer modes. All DMA data transfers require five clock cycles or 1.05 microseconds per byte.

The DMA controller does not support automatic initialization, rotating priority, cascading, or memory-to-memory transfers.

Programming Considerations

Because the internal DMA counters are only 16 bits wide, 4-bit page registers are used to form the required 20-bit address. There is a page register for each of the three DMA channels.

```
1<---------------- DMA Address ---------------->1

19<--------16 15<------------------------------>0

Page Register

DMA Address
```
Figure 2-11 shows the DMA control registers. These registers define the modes of operation, transfer addresses, transfer word counts, and status. All the registers are readable either directly or through the checkout register.

<table>
<thead>
<tr>
<th>Address (hex)</th>
<th>R/W</th>
<th>Usage</th>
</tr>
</thead>
<tbody>
<tr>
<td>000</td>
<td>R/W</td>
<td>Diagnostic checkout register (register pointer)</td>
</tr>
<tr>
<td>001</td>
<td>R/W</td>
<td>Diagnostic register defined by checkout register</td>
</tr>
<tr>
<td>002</td>
<td>R/W</td>
<td>Channel-1 current address</td>
</tr>
<tr>
<td>003</td>
<td>R/W</td>
<td>Channel-1 word current count</td>
</tr>
<tr>
<td>004</td>
<td>R/W</td>
<td>Channel-2 current address</td>
</tr>
<tr>
<td>005</td>
<td>R/W</td>
<td>Channel-2 current word count</td>
</tr>
<tr>
<td>006</td>
<td>R/W</td>
<td>Channel-3 current address</td>
</tr>
<tr>
<td>007</td>
<td>R/W</td>
<td>Channel-3 current word count</td>
</tr>
<tr>
<td>008</td>
<td>W</td>
<td>DMA Command Register</td>
</tr>
<tr>
<td></td>
<td>R</td>
<td>DMA Status register</td>
</tr>
<tr>
<td>009</td>
<td>W</td>
<td>Request register</td>
</tr>
<tr>
<td>00A</td>
<td>W</td>
<td>Write single mask register bit</td>
</tr>
<tr>
<td>00B</td>
<td>W</td>
<td>Mode register</td>
</tr>
<tr>
<td>00C</td>
<td>W</td>
<td>Clear byte pointer</td>
</tr>
<tr>
<td>00D</td>
<td>W</td>
<td>Master clear</td>
</tr>
<tr>
<td>00E</td>
<td>W</td>
<td>Clear mask register</td>
</tr>
</tbody>
</table>

Figure 2-11 (Part 1 of 2). DMA Controller Registers
<table>
<thead>
<tr>
<th>Address (hex)</th>
<th>R/W</th>
<th>Usage</th>
</tr>
</thead>
<tbody>
<tr>
<td>00F</td>
<td>W</td>
<td>Write all mask register bits</td>
</tr>
<tr>
<td>080</td>
<td>W</td>
<td>Manufacturing mode</td>
</tr>
<tr>
<td>081</td>
<td>W</td>
<td>Channel-2 page register</td>
</tr>
<tr>
<td>082</td>
<td>W</td>
<td>Channel-3 page register</td>
</tr>
<tr>
<td>083</td>
<td>W</td>
<td>Channel-1 page register</td>
</tr>
</tbody>
</table>

Figure 2-11 (Part 2 of 2). DMA Controller Registers

DMA Command Register (Hex 008)

The command register can be indirectly read by setting the checkout register pointer to hex 07.

<table>
<thead>
<tr>
<th>Bit</th>
<th>Meaning</th>
</tr>
</thead>
<tbody>
<tr>
<td>7</td>
<td>DACK polarity (must be 0)</td>
</tr>
<tr>
<td>6</td>
<td>DREQ polarity (must be 1)</td>
</tr>
<tr>
<td>5-3</td>
<td>Not used</td>
</tr>
<tr>
<td>2</td>
<td>Controller enable</td>
</tr>
<tr>
<td>1-0</td>
<td>Not used</td>
</tr>
</tbody>
</table>

2-26 System Unit
DMA Status Register (Hex 008)

<table>
<thead>
<tr>
<th>Bit</th>
<th>Meaning</th>
</tr>
</thead>
<tbody>
<tr>
<td>7</td>
<td>Channel-3 request</td>
</tr>
<tr>
<td>6</td>
<td>Channel-2 request</td>
</tr>
<tr>
<td>5</td>
<td>Channel-1 request</td>
</tr>
<tr>
<td>4</td>
<td>Not used</td>
</tr>
<tr>
<td>3</td>
<td>Channel-3 terminal count</td>
</tr>
<tr>
<td>2</td>
<td>Channel-2 terminal count</td>
</tr>
<tr>
<td>1</td>
<td>Channel-1 terminal count</td>
</tr>
<tr>
<td>0</td>
<td>Not used</td>
</tr>
</tbody>
</table>

Request Register (Hex 009)

The request register can be indirectly read by setting the checkout register pointer to hex 0B.

<table>
<thead>
<tr>
<th>Bit</th>
<th>Meaning</th>
</tr>
</thead>
<tbody>
<tr>
<td>7-3</td>
<td>Not used</td>
</tr>
<tr>
<td>2</td>
<td>Request bit:</td>
</tr>
<tr>
<td></td>
<td>0 = Reset</td>
</tr>
<tr>
<td></td>
<td>1 = Set</td>
</tr>
<tr>
<td>1-0</td>
<td>Channel select:</td>
</tr>
<tr>
<td></td>
<td>01 = Channel 1</td>
</tr>
<tr>
<td></td>
<td>10 = Channel 2</td>
</tr>
<tr>
<td></td>
<td>11 = Channel 3</td>
</tr>
</tbody>
</table>
Write Single Mask Register Bit (Hex 00A)

The write single mask register bit can be read indirectly by setting the checkout register pointer to hex 03.

<table>
<thead>
<tr>
<th>Bit</th>
<th>Meaning</th>
</tr>
</thead>
<tbody>
<tr>
<td>7-3</td>
<td>Not used</td>
</tr>
<tr>
<td>2</td>
<td>Mask bit:</td>
</tr>
<tr>
<td></td>
<td>0 = Clear</td>
</tr>
<tr>
<td></td>
<td>1 = Set</td>
</tr>
<tr>
<td>1-0</td>
<td>Channel select:</td>
</tr>
<tr>
<td></td>
<td>01 = Channel 1</td>
</tr>
<tr>
<td></td>
<td>10 = Channel 2</td>
</tr>
<tr>
<td></td>
<td>11 = Channel 3</td>
</tr>
</tbody>
</table>
Mode Register (Hex 00B)

The mode register can be indirectly read by setting the checkout register pointer to hex 08 for channel 1, hex 09 for channel 2, and hex 0A for channel 3.

**Bit Meaning**

7-6 Transfer mode:

- 00 = Demand mode transfer
- 01 = Single mode transfer
- 10 = Block mode transfer

5 Address increment/decrement:

- 0 = Address decrement
- 1 = Address increment

4 Not used

3-2 Transfer request:

- 00 = Verify transfer (diagnostic)
- 01 = Write transfer (I/O to memory)
- 10 = Read transfer (memory to I/O)

1-0 Mode:

- 01 = Select channel-1 mode register
- 10 = Select channel-2 mode register
- 11 = Select channel-3 mode register
Clear Byte Pointer (Hex 00C)

The clear byte pointer register is not data-bit sensitive, it is used to determine which byte is addressed when reading from or writing to a 16-bit register. When the pointer is clear, the low-order byte is accessed; when the pointer is set, the high-order byte is accessed. The pointer automatically toggles when reading from or writing to any 16-bit register. The pointer is cleared by a master clear command, reset, or clear byte pointer command.

Master Clear (Hex 00D)

The master clear register is not data-bit sensitive; it is used to initialize the standard DMA registers. A write to this register causes the DMA controller to initialize all registers to the power-up default values and mask all DMA channels.

Clear Mask Register (Hex 00E)

The clear mask register is not data-bit sensitive, it is used to set the bits in the mask register. A write to this register causes all channel mask bits to be cleared which enables all DMA channels.
### Write All Mask Register Bits Register (Hex 00F)

The write all mask register bits register can be indirectly read by setting the checkout register pointer to hex 03.

<table>
<thead>
<tr>
<th>Bit</th>
<th>Meaning</th>
</tr>
</thead>
<tbody>
<tr>
<td>7-4</td>
<td>Not used</td>
</tr>
<tr>
<td>3</td>
<td>Channel-3 mask:</td>
</tr>
<tr>
<td></td>
<td>0 = Clear</td>
</tr>
<tr>
<td></td>
<td>1 = Set</td>
</tr>
<tr>
<td>2</td>
<td>Channel-2 mask:</td>
</tr>
<tr>
<td></td>
<td>0 = Clear</td>
</tr>
<tr>
<td></td>
<td>1 = Set</td>
</tr>
<tr>
<td>1</td>
<td>Channel-1 mask:</td>
</tr>
<tr>
<td></td>
<td>0 = Clear</td>
</tr>
<tr>
<td></td>
<td>1 = Set</td>
</tr>
<tr>
<td>0</td>
<td>Not used</td>
</tr>
</tbody>
</table>
Read-Only Memory (ROM)

The system ROM is made up of two 32K by 8-bit modules. The two modules are arranged in a 64K by 8-bit configuration. The ROM is assigned addresses hex F0000 through hex FFFFF. Figure 2-12 shows the mapping of ROM.

ROM does not use parity.

<table>
<thead>
<tr>
<th>Address (hex)</th>
<th>Length (K-bytes)</th>
<th>Usage</th>
</tr>
</thead>
<tbody>
<tr>
<td>F0000</td>
<td>24</td>
<td>POST, BIOS</td>
</tr>
<tr>
<td>F6000</td>
<td>32</td>
<td>Resident BASIC</td>
</tr>
<tr>
<td>FE000</td>
<td>8</td>
<td>POST, BIOS</td>
</tr>
</tbody>
</table>

Figure 2-12. ROM Map
Random Access Memory (RAM)

The RAM is used to store application programs and user data. All RAM is static and does not require refresh. RAM does not use parity. The RAM is contained on cards; each card has 128K bytes of storage. A maximum of four cards can be installed in the system.

The starting address of RAM is hex 00000, as shown in Figure 2-3 on page 2-6. Figure 2-13 on page 2-34 shows the expansion RAM connectors at the system board. The plus (+) or minus (−) preceding the signal name indicates the active state of the signals. The input/output column indicates whether the signal direction is to or from the system board.
<table>
<thead>
<tr>
<th>I/O Pin</th>
<th>Signal Name</th>
<th>Input/Output</th>
</tr>
</thead>
<tbody>
<tr>
<td>A01</td>
<td>+ Address/Data Bit 0</td>
<td>Input/Output</td>
</tr>
<tr>
<td>A02</td>
<td>+ Address/Data Bit 1</td>
<td>Input/Output</td>
</tr>
<tr>
<td>A03</td>
<td>+ Address/Data Bit 2</td>
<td>Input/Output</td>
</tr>
<tr>
<td>A04</td>
<td>+5 V dc</td>
<td>Power</td>
</tr>
<tr>
<td>A05</td>
<td>+ Address/Data Bit 3</td>
<td>Input/Output</td>
</tr>
<tr>
<td>A06</td>
<td>+ Address/Data Bit 4</td>
<td>Input/Output</td>
</tr>
<tr>
<td>A07</td>
<td>+ Address/Data Bit 5</td>
<td>Input/Output</td>
</tr>
<tr>
<td>A08</td>
<td>+5 V dc</td>
<td>Power</td>
</tr>
<tr>
<td>A09</td>
<td>+ Address/Data Bit 6</td>
<td>Input/Output</td>
</tr>
<tr>
<td>A10</td>
<td>+ Address/Data Bit 7</td>
<td>Input/Output</td>
</tr>
<tr>
<td>A11</td>
<td>+ Address Bit 8</td>
<td>Output</td>
</tr>
<tr>
<td>A12</td>
<td>+5 V dc</td>
<td>Power</td>
</tr>
<tr>
<td>A13</td>
<td>+ Address Bit 9</td>
<td>Output</td>
</tr>
<tr>
<td>A14</td>
<td>+ Address Latch Enable</td>
<td>Output</td>
</tr>
<tr>
<td>A15</td>
<td>Reserved</td>
<td></td>
</tr>
<tr>
<td>A16</td>
<td>+5 V dc</td>
<td>Power</td>
</tr>
<tr>
<td>A17</td>
<td>− Memory Card Select 0</td>
<td>Output</td>
</tr>
<tr>
<td>A18</td>
<td>− Memory Card Select 2</td>
<td>Output</td>
</tr>
<tr>
<td>A19</td>
<td>− Memory Card Select 3</td>
<td>Output</td>
</tr>
<tr>
<td>A20</td>
<td>− Memory Card Select 4</td>
<td>Output</td>
</tr>
<tr>
<td>B01</td>
<td>+ A10</td>
<td>Output</td>
</tr>
<tr>
<td>B02</td>
<td>Ground</td>
<td>Ground</td>
</tr>
<tr>
<td>B03</td>
<td>+ Address Bit 11</td>
<td>Output</td>
</tr>
<tr>
<td>B04</td>
<td>Ground</td>
<td>Ground</td>
</tr>
<tr>
<td>B05</td>
<td>+ Address Bit 12</td>
<td>Output</td>
</tr>
<tr>
<td>B06</td>
<td>Ground</td>
<td>Ground</td>
</tr>
<tr>
<td>B07</td>
<td>+ Address Bit 13</td>
<td>Output</td>
</tr>
<tr>
<td>B08</td>
<td>Ground</td>
<td>Ground</td>
</tr>
<tr>
<td>B09</td>
<td>+ Address Bit 14</td>
<td>Output</td>
</tr>
<tr>
<td>B10</td>
<td>Ground</td>
<td>Ground</td>
</tr>
</tbody>
</table>

Figure 2-13 (Part 1 of 2). System Board RAM Connector

2-34 System Unit
The following is a description of the signals on the RAM connectors.

**Address/Data Bit 0 through 7**

These multiplexed lines are used to form either the low-order bits of an address or a byte of data. At the falling edge of the ‘address latch enable’ signal, the RAM cards access these eight lines along with lines ‘address bit 8’ through ‘address bit 16’ to form an address. At the low level of either ‘memory read’ or ‘memory write’, the RAM cards access these eight lines to form a data byte. The least significant bit is ‘address/data bit 0.’

**Address Bit 8 through 16**

These lines are used to address memory and I/O devices within the system. The data on these lines is valid through the entire bus cycle.
Memory Card Select 0 through 4

These lines are used to select the card to be accessed. These lines are shifted at the output of the RAM cards.

Address Latch Enable

This line is used to indicate that the address/data bus contains a valid address.

Memory Write

This line is used to instruct a RAM card to store the data present on the data bus.

Data Enable

This line indicates when data should be gated onto the multiplexed address/data bus.

RAM Enable

This line enables the RAM card to be accessed. When this line is low, all other signals to the card are ignored.

I/O Channel

The I/O channel is an extension of the internal bus used by the processor and other functional units in the IBM PC Convertible. The channel contains multiplexed low-order address/data lines (bidirectional), high-order address lines, six interrupt control lines, memory and I/O read or write control lines, clock and timing lines, three DMA control lines, a channel-check line, and power and ground lines for the optional features. A serial printer interface for the IBM PC Convertible Portable Printer is also provided on the I/O channel connector.
Memory (RAM or ROM) can be addressed on the I/O channel only above address hex A0000 (640K). Memory reads or writes below that address are directed to the RAM connector.

I/O Channel Interface

The lines of the I/O channel are provided at the rear of the system unit on a 72-pin connector. Figure 2-14 on page 2-38 shows I/O channel connector pins. The plus (+) or minus (−) preceding the signal name indicates the active state of the signals. The input/output column indicates whether the signal direction is to or from the system unit. The I/O signals have sufficient drive to power up to two CMOS loads plus one low-power Schotty (LS) TTL load.
<table>
<thead>
<tr>
<th>I/O Pin</th>
<th>Signal Name</th>
<th>Input/Output</th>
</tr>
</thead>
<tbody>
<tr>
<td>01</td>
<td>Shell ground</td>
<td>Ground</td>
</tr>
<tr>
<td>02</td>
<td>+ Address Latch Enable</td>
<td>Output</td>
</tr>
<tr>
<td>03</td>
<td>+5 V dc</td>
<td>Power</td>
</tr>
<tr>
<td>04</td>
<td>+ Address Enable</td>
<td>Output</td>
</tr>
<tr>
<td>05</td>
<td>+ I/O Channel Ready</td>
<td>Input</td>
</tr>
<tr>
<td>06</td>
<td>Ground</td>
<td>Ground</td>
</tr>
<tr>
<td>07</td>
<td>Spare</td>
<td></td>
</tr>
<tr>
<td>08</td>
<td>Ground</td>
<td>Ground</td>
</tr>
<tr>
<td>09</td>
<td>- I/O Channel Check</td>
<td>Input</td>
</tr>
<tr>
<td>10</td>
<td>+ Sleep Clock</td>
<td>Output</td>
</tr>
<tr>
<td>11</td>
<td>+ Terminal Count</td>
<td>Output</td>
</tr>
<tr>
<td>12</td>
<td>+12 V dc</td>
<td>Power</td>
</tr>
<tr>
<td>13</td>
<td>- I/O Write</td>
<td>Output</td>
</tr>
<tr>
<td>14</td>
<td>- Memory Read</td>
<td>Output</td>
</tr>
</tbody>
</table>

Figure 2-14 (Part 1 of 3). System I/O Connector
<table>
<thead>
<tr>
<th>I/O Pin</th>
<th>Signal Name</th>
<th>Input/Output</th>
</tr>
</thead>
<tbody>
<tr>
<td>15</td>
<td>Memory Write</td>
<td>Output</td>
</tr>
<tr>
<td>16</td>
<td>Adapter Power</td>
<td>Power</td>
</tr>
<tr>
<td>17</td>
<td>Address Bit 10</td>
<td>Output</td>
</tr>
<tr>
<td>18</td>
<td>Address Bit 11</td>
<td>Output</td>
</tr>
<tr>
<td>19</td>
<td>Address Bit 12</td>
<td>Output</td>
</tr>
<tr>
<td>20</td>
<td>Address Bit 13</td>
<td>Output</td>
</tr>
<tr>
<td>21</td>
<td>Address Bit 14</td>
<td>Output</td>
</tr>
<tr>
<td>22</td>
<td>Address Bit 15</td>
<td>Output</td>
</tr>
<tr>
<td>23</td>
<td>Address Bit 16</td>
<td>Output</td>
</tr>
<tr>
<td>24</td>
<td>Address Bit 17</td>
<td>Output</td>
</tr>
<tr>
<td>25</td>
<td>–13 Vdc</td>
<td>Power</td>
</tr>
<tr>
<td>26</td>
<td>Ground</td>
<td>Ground</td>
</tr>
<tr>
<td>27</td>
<td>Address Bit 18</td>
<td>Output</td>
</tr>
<tr>
<td>28</td>
<td>Address Bit 19</td>
<td>Output</td>
</tr>
<tr>
<td>29</td>
<td>Power Adapter Active</td>
<td>Output</td>
</tr>
<tr>
<td>30</td>
<td>Printer Power</td>
<td>Power</td>
</tr>
<tr>
<td>31</td>
<td>Ground</td>
<td>Ground</td>
</tr>
<tr>
<td>32</td>
<td>Transmit Data</td>
<td>Output</td>
</tr>
<tr>
<td>33</td>
<td>Printer Error</td>
<td>Input</td>
</tr>
<tr>
<td>34</td>
<td>Printer Enable</td>
<td>Output</td>
</tr>
<tr>
<td>35</td>
<td>Printer Busy</td>
<td>Input</td>
</tr>
<tr>
<td>36</td>
<td>Shell Ground</td>
<td>Ground</td>
</tr>
<tr>
<td>37</td>
<td>Address/Data Bit 0</td>
<td>Input/output</td>
</tr>
<tr>
<td>38</td>
<td>Address/Data Bit 1</td>
<td>Input/output</td>
</tr>
<tr>
<td>39</td>
<td>Address/Data Bit 2</td>
<td>Input/output</td>
</tr>
<tr>
<td>40</td>
<td>Address/Data Bit 3</td>
<td>Input/output</td>
</tr>
<tr>
<td>41</td>
<td>Address/Data Bit 4</td>
<td>Input/output</td>
</tr>
<tr>
<td>42</td>
<td>Address/Data Bit 5</td>
<td>Input/output</td>
</tr>
<tr>
<td>43</td>
<td>Address/Data Bit 6</td>
<td>Input/output</td>
</tr>
<tr>
<td>44</td>
<td>Address/Data Bit 7</td>
<td>Input/output</td>
</tr>
</tbody>
</table>

Figure 2-14 (Part 2 of 3). System I/O Connector
<table>
<thead>
<tr>
<th>I/O Pin</th>
<th>Signal Name</th>
<th>Input/Output</th>
</tr>
</thead>
<tbody>
<tr>
<td>45</td>
<td>+ Address Bit 8</td>
<td>Output</td>
</tr>
<tr>
<td>46</td>
<td>Ground</td>
<td>Ground</td>
</tr>
<tr>
<td>47</td>
<td>+ Address Bit 9</td>
<td>Output</td>
</tr>
<tr>
<td>48</td>
<td>+ Interrupt Request 4</td>
<td>Input</td>
</tr>
<tr>
<td>49</td>
<td>- I/O Read</td>
<td>Output</td>
</tr>
<tr>
<td>50</td>
<td>+ Reset</td>
<td>Output</td>
</tr>
<tr>
<td>51</td>
<td>- Data Enable</td>
<td>Output</td>
</tr>
<tr>
<td>52</td>
<td>Reserved</td>
<td></td>
</tr>
<tr>
<td>53</td>
<td>Ground</td>
<td>Ground</td>
</tr>
<tr>
<td>54</td>
<td>+ DMA Request 1</td>
<td>Input</td>
</tr>
<tr>
<td>55</td>
<td>- DMA Acknowledge 1</td>
<td>Output</td>
</tr>
<tr>
<td>56</td>
<td>Ground</td>
<td>Ground</td>
</tr>
<tr>
<td>57</td>
<td>+ DMA Request 2</td>
<td>Input</td>
</tr>
<tr>
<td>58</td>
<td>- DMA Acknowledge 2</td>
<td>Output</td>
</tr>
<tr>
<td>59</td>
<td>+ DMA Request 3</td>
<td>Input</td>
</tr>
<tr>
<td>60</td>
<td>Ground</td>
<td>Ground</td>
</tr>
<tr>
<td>61</td>
<td>- DMA Acknowledge 3</td>
<td>Output</td>
</tr>
<tr>
<td>62</td>
<td>+ Interrupt Request 2</td>
<td>Input</td>
</tr>
<tr>
<td>63</td>
<td>+ Interrupt Request 3</td>
<td>Input</td>
</tr>
<tr>
<td>64</td>
<td>Ground</td>
<td>Ground</td>
</tr>
<tr>
<td>65</td>
<td>+ Interrupt Request 6</td>
<td>Input</td>
</tr>
<tr>
<td>66</td>
<td>Spare</td>
<td></td>
</tr>
<tr>
<td>67</td>
<td>+ Interrupt Request 5</td>
<td>Input</td>
</tr>
<tr>
<td>68</td>
<td>Spare</td>
<td></td>
</tr>
<tr>
<td>69</td>
<td>Reserved</td>
<td></td>
</tr>
<tr>
<td>70</td>
<td>+ Power Enable</td>
<td>Output</td>
</tr>
<tr>
<td>71</td>
<td>Ground</td>
<td>Ground</td>
</tr>
<tr>
<td>72</td>
<td>+ Interrupt Request 7</td>
<td>Input</td>
</tr>
</tbody>
</table>

Figure 2-14 (Part 3 of 3). System I/O Connector
The following paragraphs describe the interface lines.

**Address/Data Bit 0 through 7**

These multiplexed lines are used to form either the low-order bits of an address or a byte of data. At the falling edge of the ‘address latch enable’ signal, the attachments access these eight lines along with lines ‘address bit 8’ through ‘address bit 19’ to form a 20-bit address. At the low level of signal ‘memory read’, ‘memory write’, ‘I/O read’, or ‘I/O write’, the attachments access these eight lines to form a data byte. The least significant bit is ‘address/data bit 0’.

**Address Bit 8 through 15**

These lines are used to address memory and I/O devices within the system.

**Address Bit 16 through 19**

These multiplexed lines contain either address or status bits. At the falling edge of the ‘address latch enable’ signal, the attachments access these lines to complete a 20-bit address. These lines are inactive during I/O operations.

**Sleep Clock**

This line provides the sleep clock pulses. It has a 210 nanoseconds (4.77 MHz) cycle.

During sleep mode (waiting for I/O activity), this clock may be stopped. It resumes at normal speed when an interrupt or DMA request is received. When the clock is stopped, this line is set to the low level.
**Reset**

This line is used to reset or initialize the system logic during the power-on sequence.

**Address Latch Enable**

This line is used to indicate that the address bus (address and data bit 0 through 19) contains a valid address. Because the address bus is multiplexed, the I/O attachments must use the falling edge of the ‘address latch enable’ signal to latch processor addresses 0 through 7 and 16 through 19.

**I/O Channel Check**

This line indicates an I/O device error.

**I/O Channel Ready**

This line is used by I/O devices or memory devices to lengthen the I/O or memory cycle. This is done by forcing the line to low level (not ready); the cycle is then extended by any number of complete ‘clock’ cycles (210 nanoseconds). Devices using this line should force it to low level immediately after detecting a valid address and either a read or write command.

**Interrupt Request 2 through 7**

These lines are used to signal the processor that an I/O device requires attention. The lines are in order of priority with ‘interrupt request 2’ having the highest priority and ‘interrupt request 7’ having the lowest priority. An interrupt request is generated by raising a line to the high level and maintaining that level until the interrupt service routine acknowledges the request.

**I/O Read**

This command line is used by either the processor or DMA controller to instruct an I/O device to place data on the data bus.
I/O Write
This command line is used by either the processor or DMA controller to instruct an I/O device to read the data on the data bus.

Memory Read
This command line is used by either the processor or DMA controller to instruct the addressed memory device to place data on the data bus.

Memory Write
This command line is used by either the processor or DMA controller to instruct the addressed memory device to store the data that is present on the data bus.

DMA Request 1, 2, and 3
These lines are used by the I/O devices to request DMA data transfers. The lines are in order of priority with 'DMA request 1' having the highest priority and 'DMA request 3' the lowest. A request is generated when a 'DMA request' line is activated. The line must be held active until the corresponding 'DMA acknowledge' line is activated.

DMA Acknowledge 1, 2, and 3
These lines are used by the DMA controller to acknowledge DMA data transfer requests.

Terminal Count
This line is used to indicate that the byte count has reached a count of zero and is active at the completion of a DMA operation.
Address Enable

This line indicates that the DMA controller has control of the I/O channel during DMA transfer operations.

Data Enable

This line indicates when data should be gated onto the multiplexed address/data bus.

Adapter Power

This line provides an unregulated dc output (+9.2 to +16.0 volts) when either the automobile or the ac adapter is powering the system, regardless of whether or not the power enable signals are active. The output is routed to the I/O connector to power external attachments.

Power Adapter Active

This line is used by the attached devices to detect when the IBM PC Convertible is operating on the battery, so that the devices can reduce their power consumption or turn themselves off. The line is provided by the power supply. An up level indicates that system power is being supplied by the ac adapter or automobile power adapter.

Power Enable

This line is used to indicate when power is applied to the system.

Transmit Data

This line contains the serial data for the portable printer. A low level indicates a “mark” condition and a high level indicates a “space” condition. The line operates at 1200 bits per second and the data consists of 8 data bits and 2 stop bits. Parity bits are not used.
**Printer Busy**

This line is used by the portable printer to indicate when it is no longer able to accept data (offline, printing, buffer full, page eject, or error condition). When this line is at the low level, the printer cannot accept data.

**Printer Error**

This line is used by the portable printer to indicate when it has an error condition that needs attention from the operator. A low level indicates a printer error (offline, paper out, or end of ribbon).

**Printer Power**

This line provides an unregulated dc output (+8.0 to +16.0 volts) when a charged battery pack, ac adapter, or automobile power adapter is powering the system. The output is used to power the portable printer.

**Printer Enable**

This line is used to reset the portable printer and control the standby/active status of the portable printer. An up level on this line causes the printer to become ready.

**Liquid Crystal Display (LCD)**

The display for the IBM PC Convertible is a liquid-crystal, dot matrix display capable of displaying 25 lines of 80 characters or 640 by 200 picture elements (pels).

The LCD can be disconnected from the system when a monitor is connected to the system.
Liquid Crystal Display (LCD) Controller

The LCD controller provides the interface to the liquid crystal display.

The LCD controller is compatible with programs that use the IBM Monochrome Display Adapter or the IBM Color/Graphics Display Adapter. To accomplish this, the LCD controller uses two address ranges for control registers, one for monochrome operations and one for color/graphics operations. The operation and register interface of the LCD controller are similar to operation and interface of the Motorola 6845 CRT Controller.

The LCD controller can address 16K bytes of display storage (used for video refresh) and supports up to 512 different character codes in two font storage areas. One of the font storage areas contains the main font, while the other contains the alternate font. The power-on routines initialize the font areas to the IBM Personal Computer character set (shown in Appendix A, “Character Sets and Keystrokes”) that is stored in read-only memory. Both areas are initialized to the same character set at system power-on time.

The LCD controller supports two basic modes of operation, alphanumeric and graphics (all points addressable). In alphanumeric mode, the LCD controller maps the character and attribute information in the refresh buffer to the display panel, using the data in the font storage area. In graphics mode, the LCD controller directly maps the refresh buffer to the display panel on a bit-per-pel (picture element) basis.

For monochrome operations, the IBM PC Convertible LCD controller operates similarly to the IBM Personal Computer Monochrome Display Adapter. Characters are displayed within an 8-by-8 dot matrix, using the character set stored in the font storage area. Monochrome operations use a 16K-byte refresh buffer that starts at address hex B0000.
For color/graphics operations, the IBM PC Convertible LCD controller operates similarly to the IBM Personal Computer Color/Graphics Display Adapter. Characters are displayed within an 8 by 8 dot matrix. The characters are stored in the font storage area in RAM. The LCD can display 640 by 200 pels of information in color/graphics mode. Color/graphics operations use a 16K-byte refresh buffer that starts at address hex B8000.

**Monochrome Operations**

Monochrome applications are supported using alphanumeric mode with an 80-column by 25-row display. The characters are displayed using an 8-wide by 8-high character box.

Every display-character position in alphanumeric mode is defined by 2 bytes in the refresh buffer. The following shows the format of these bytes.

<table>
<thead>
<tr>
<th>Character Code Byte (Even)</th>
<th>Attribute Byte (Odd)</th>
</tr>
</thead>
<tbody>
<tr>
<td>7 6 5 4 3 2 1 0</td>
<td>7 6 5 4 3 2 1 0</td>
</tr>
</tbody>
</table>

- **Blink**
- **Background**
- **Foreground**
- **Intensity**
The foreground and background bit settings provide the following functions:

<table>
<thead>
<tr>
<th>Background/Foreground Bits</th>
<th>Attributes</th>
</tr>
</thead>
<tbody>
<tr>
<td>6 5 4 2 1 0</td>
<td></td>
</tr>
<tr>
<td>0 0 0 0 0 0</td>
<td>Solid grey (nondisplay)</td>
</tr>
<tr>
<td>0 0 0 0 0 1</td>
<td>Normal video underscored</td>
</tr>
<tr>
<td>0 0 0 1 1 1</td>
<td>Normal video</td>
</tr>
<tr>
<td>1 1 1 0 0 0</td>
<td>Reverse video</td>
</tr>
<tr>
<td>1 1 1 1 1 1</td>
<td>Solid black (nondisplay)</td>
</tr>
</tbody>
</table>

The intensity attribute does not affect the intensity of the LCD, but may be used to select one of the display attributes shown in the following by using a BIOS function call (interrupt hex 10):

- Underscore
- Reverse video
- Select alternate font
- Ignore intensity bit.
A full screen of alphanumeric text requires 4000 bytes of refresh buffer. Up to four screens (pages) can be stored in the refresh buffer and each screen can be selectively displayed using a BIOS function call. Figure 2-15 shows the mapping of the refresh buffer for one full screen of alphanumeric text.

<table>
<thead>
<tr>
<th>Address (hex)</th>
<th>Usage</th>
</tr>
</thead>
<tbody>
<tr>
<td>B0000</td>
<td>Code for character at upper-left corner of screen</td>
</tr>
<tr>
<td>B0001</td>
<td>Attribute for first character</td>
</tr>
<tr>
<td>B0...</td>
<td></td>
</tr>
<tr>
<td>B07CF</td>
<td>Attribute for character at bottom-right corner of screen</td>
</tr>
</tbody>
</table>

Figure 2-15. Monochrome Operations Refresh buffer
# Color/Graphics Operations

Color/graphics applications are supported using the following modes:

- **Alphanumeric**
  - 40-character by 25-line
  - 80-character by 25-line

- **Graphics (all-points-addressable)**
  - 320 pels by 200 rows
  - 640 pels by 200 rows.

In alphanumeric mode, the format of the refresh buffer is similar to the monochrome mapping, except for the definition of the attribute bits:

<table>
<thead>
<tr>
<th>Character Code Byte (Even)</th>
<th>Attribute Byte (Odd)</th>
</tr>
</thead>
<tbody>
<tr>
<td>7 6 5 4 3 2 1 0</td>
<td>7 6 5 4 3 2 1 0</td>
</tr>
<tr>
<td>Blink</td>
<td>Foreground</td>
</tr>
<tr>
<td>Background</td>
<td>Intensity</td>
</tr>
</tbody>
</table>

![Attribute Bit Diagram](image-url)
The following shows the color mapping for alphanumeric mode:

- Any color foreground with black background: Normal display
- Black foreground with any color background: Reverse video
- Any color foreground with different color background: Reverse video
- Any color foreground with same color background: Nondisplay (all pels on)
- Black foreground with black background: Nondisplay (all pels on).

A full screen of alphanumeric requires 2000 bytes of refresh buffer for 40-by-25 mode and 4000 bytes for 80-by-25 mode. Up to eight (four if 80-by-25 mode) screens (pages) can be stored in the refresh buffer and each screen can be selectively displayed using a BIOS function call. Figure 2-16 shows the mapping of the refresh buffer for one full screen of alphanumeric text in 80-by-25 mode.

<table>
<thead>
<tr>
<th>Address (hex)</th>
<th>Usage</th>
</tr>
</thead>
<tbody>
<tr>
<td>B8000</td>
<td>Code for character at upper-left corner of screen</td>
</tr>
<tr>
<td>B8001</td>
<td>Attribute for first character</td>
</tr>
<tr>
<td>B8...</td>
<td>Subsequent codes and attributes</td>
</tr>
<tr>
<td>B87CF</td>
<td>Attribute for character at bottom-right corner of screen</td>
</tr>
</tbody>
</table>

*Figure 2-16. Color/Graphics Operations Refresh Buffer*
In graphics (all-points-addressable) mode, the LCD controller directly maps the display refresh buffer to the display on a bit-per-pel basis. Applications operating in all-points-addressable mode can have a display area that is either 320-pels by 200-rows (medium resolution) or 640-pels by 200-rows (high resolution).

Two 8000-byte areas in the refresh buffer are used to map the display. Figure 2-17 shows the layout of the refresh buffer. The first area contains pel information for the even-numbered display row, beginning with the pel information for the upper-left corner of the display. The second area contains pel information for the odd-numbered display rows.

<table>
<thead>
<tr>
<th>Address (hex)</th>
<th>Length (bytes)</th>
<th>Usage</th>
</tr>
</thead>
<tbody>
<tr>
<td>B8000</td>
<td>8000</td>
<td>Even Rows 0-198</td>
</tr>
<tr>
<td>B9F3F</td>
<td>192</td>
<td>Reserved</td>
</tr>
<tr>
<td>BA000</td>
<td>8000</td>
<td>Odd Rows 1-199</td>
</tr>
<tr>
<td>BBF3F</td>
<td>192</td>
<td>Reserved</td>
</tr>
</tbody>
</table>

Figure 2-17. APA Mode Refresh Buffer

For high-resolution operations, each bit in the display storage represents a pel on the display. Each physical pel is set to on or off according to the bit mapping in the display storage area. Figure 2-18 shows the bit-to-pel relationship for high resolution.

![Bit Position and Pel Position](image)

Figure 2-18. High Resolution Bit-to-Pel Relationship
For medium resolution operations, the physical pels are considered in pairs to be a logical pel. That is, the first and second physical pels on the display are considered as a logical pel. Figure 2-19 shows the bit-to-pel relationship for medium resolution. Figure 2-20 shows how the bit pairs are mapped for medium resolution. Alternating the two dark gray tones gives the appearance of two different shades. That is, a dark grey 1 that is adjacent to a figure mapped in dark grey 2 creates a border that separates the two grey figures.

<table>
<thead>
<tr>
<th>Bit Position</th>
<th>Pel Position</th>
</tr>
</thead>
<tbody>
<tr>
<td>7 6 5 4 3 2 1 0</td>
<td>1 2 3 4</td>
</tr>
</tbody>
</table>

Left Pel | Right Pel

Figure 2-19. Medium Resolution Bit-to-Pel Relationship

Note: The bits are considered in pairs, such as bits 7 and 6 are pairs that map pel position 1. See Figure 2-20 for how to map the image to the pairs for medium resolution.

<table>
<thead>
<tr>
<th>Bit Position</th>
<th>Image</th>
</tr>
</thead>
<tbody>
<tr>
<td>1st 2nd</td>
<td></td>
</tr>
<tr>
<td>0 0</td>
<td>Grey (background)</td>
</tr>
<tr>
<td>0 1</td>
<td>Dark grey 1</td>
</tr>
<tr>
<td>1 0</td>
<td>Dark grey 2</td>
</tr>
<tr>
<td>1 1</td>
<td>Black (foreground)</td>
</tr>
</tbody>
</table>

Figure 2-20. Medium Resolution Pel Mapping
LCD Interface

The lines to the LCD are provided on the system unit on a 20-pin connector. Figure 2-21 on page 2-55 shows LCD connector pins. The plus (+) or minus (−) preceding the signal names indicates the active state of the signals. The input/output column indicates whether the signal direction is to or from the system unit.
I/O Pin  | Signal Name     | Input/Output
-------|----------------|---------
01      | +Scan Data 0   | Output  
02      | +Scan Data 1   | Output  
03      | +Scan Data 2   | Output  
04      | +Scan Data 3   | Output  
05      | +Display Enable| Output  
06      | +Driver Oscillator | Output  
07      | +Y Shift Clock | Output  
08      | −LCD Sense     | Input   
09      | +X Enable Clock| Output  
10      | +X Shift Clock | Output  
11      | +5 V dc        | Power   
12      | Ground         |         
13      | Ground         |         
14      | +Start Frame   | Output  
15      | Ground         |         
16      | −13 V dc       | Power   
17      | Ground         |         
18      | +Latch Pulse   | Output  
19      | Reserved       |         
20      | Reserved       |         

Figure 2-21. LCD Connector

The following paragraphs describe the interface lines.

**Scan Data 0 - 3**

The lines contain the serial character data for the LCD. Each line corresponds to a pel.

**Display Enable**

This line is used to enable and disable the display.

**Drive Oscillator**

This line provides the ac waveform used to drive the LCD.
Y Shift Clock

A high to low transition on this line enables the next row shift register to receive input data. Data is gated into the row drivers during the next ‘latch pulse’ signal.

LCD Sense

This line is used to determine if an LCD is connected to the system. Connecting an LCD causes this line to be pulled to the low level.

X Enable Clock

A high to low transition on this line enables the next column register.

X Shift Clock

A high to low transition on this line gates the display data into the enabled column registers.

Start Frame

This line is used to set the row shift registers to the top of the scan line. This pulse occurs at the beginning of each screen period.

Latch Pulse

This line is used to transfer the data in the column and row shift registers into the column and row drivers.
Programming Considerations

Depending on the application, the display control registers and refresh buffer can occupy one of two possible ranges. Monochrome applications use display control registers in the hex 3B0 through hex 3BF range and refresh buffer in the hex B0000 through hex B3FFF range. Color/graphics applications use display control registers in the hex 3D0 through hex 3DF range and refresh buffer in the hex B8000 through hex BBFFF range.

Monochrome or color/graphics operation can be selected through the system profile. An application can change the operating mode by using the video I/O function call provided by BIOS. This is accomplished by modifying the video bit mask in the equipment word and using the mode set function of the interrupt hex 10 BIOS function call. If the IBM PC Convertible CRT Display Adapter is installed, the LCD is set up to emulate the IBM Personal Computer Monochrome Display Adapter and this mode cannot be changed.

Applications that directly access the refresh buffer do not need to synchronize the access with vertical and horizontal syncs. Additionally, the application does not need to disable video while accessing the refresh buffer. The LCD controller automatically resolves any memory contention without affecting the display. Disabling video during an access may result in faint scan lines at the top and middle of the display panel.

Figure 2-22 on page 2-58 shows the LCD controller register assignments. Substitute the x shown in the address with either a B for monochrome operations or D for color/graphics operations.
<table>
<thead>
<tr>
<th>Address (hex)</th>
<th>R/W</th>
<th>Usage</th>
</tr>
</thead>
<tbody>
<tr>
<td>074</td>
<td>R/W</td>
<td>Control index register</td>
</tr>
<tr>
<td>075</td>
<td>R/W</td>
<td>Control data register</td>
</tr>
<tr>
<td>3x0, 3x2, 3x4, 3x6</td>
<td>R/W</td>
<td>Index register</td>
</tr>
<tr>
<td>3x1, 3x3, 3x5, 3x7</td>
<td>R/W</td>
<td>Data register</td>
</tr>
<tr>
<td>3x8</td>
<td>R/W</td>
<td>Mode control register</td>
</tr>
<tr>
<td>3x9</td>
<td></td>
<td>Reserved</td>
</tr>
<tr>
<td>3xA</td>
<td>R/W</td>
<td>Status register</td>
</tr>
<tr>
<td>3xB-3xF</td>
<td></td>
<td>Reserved</td>
</tr>
</tbody>
</table>

Figure 2-22. LCD Controller Registers

**Control Index and Control Data Registers (Hex 074 and 075)**

The control index register (hex 074) and control data register (hex 075) are set by BIOS and are used to access the control and diagnostic registers.

**Index and Data Registers (Hex 3x4 and 3x5)**

The index and data registers are used to access the LCD controller internal registers. The index register is loaded with the index number of the register to be accessed through the data register. The data register is loaded with the data to be placed into the selected register. Figure 2-23 on page 2-59 shows the internal registers, the index numbers, and their initial settings in hexadecimal. The registers are all read/write registers.
**Note:** The power-on routines initialize these registers. BIOS function calls should be used to change the values in these registers in order to preserve application program compatibility.

<table>
<thead>
<tr>
<th>Index No.</th>
<th>Usage</th>
<th>Mono</th>
<th>A/N*</th>
<th>APA*</th>
</tr>
</thead>
<tbody>
<tr>
<td>00</td>
<td>Reserved</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>01</td>
<td>R1, number of characters displayed horizontally</td>
<td>50</td>
<td>--</td>
<td>--</td>
</tr>
<tr>
<td>02</td>
<td>Reserved</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>03</td>
<td>Reserved</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>04</td>
<td>Reserved</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>05</td>
<td>Reserved</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>06</td>
<td>R6, number of characters displayed vertically</td>
<td>19</td>
<td>19</td>
<td>64</td>
</tr>
<tr>
<td>07</td>
<td>Reserved</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>07</td>
<td>Reserved</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>08</td>
<td>Reserved</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>09</td>
<td>R9, maximum scan line address</td>
<td>--</td>
<td>07</td>
<td>01</td>
</tr>
<tr>
<td>0A</td>
<td>R10, cursor start scan line address</td>
<td>--</td>
<td>06</td>
<td>06</td>
</tr>
<tr>
<td>0B</td>
<td>R11, cursor end scan line address</td>
<td>--</td>
<td>07</td>
<td>07</td>
</tr>
</tbody>
</table>

*Figure 2-23 (Part 1 of 2). Index and Data Registers*

*Color/graphics mode only.*
<table>
<thead>
<tr>
<th>Index No.</th>
<th>Usage</th>
<th>Mono</th>
<th>A/N*</th>
<th>APA*</th>
</tr>
</thead>
<tbody>
<tr>
<td>0C</td>
<td>R12, start address, high</td>
<td>xx</td>
<td>xx</td>
<td>xx</td>
</tr>
<tr>
<td>0D</td>
<td>R13, start address, low</td>
<td>xx</td>
<td>xx</td>
<td>xx</td>
</tr>
<tr>
<td>0E</td>
<td>R14, cursor address, high</td>
<td>xx</td>
<td>xx</td>
<td>xx</td>
</tr>
<tr>
<td>10</td>
<td>Reserved</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>11</td>
<td>Reserved</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>12</td>
<td>R18, font select</td>
<td>00</td>
<td>00</td>
<td>xx</td>
</tr>
<tr>
<td>13</td>
<td>Reserved</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>14</td>
<td>R20, intensity and color enable</td>
<td>xx</td>
<td>xx</td>
<td>xx</td>
</tr>
<tr>
<td>15</td>
<td>R21, number of horizontal LCD characters displayed</td>
<td>50</td>
<td>50</td>
<td>50</td>
</tr>
<tr>
<td>16</td>
<td>R22, number of LCD scan lines</td>
<td>07</td>
<td>07</td>
<td>01</td>
</tr>
<tr>
<td>17</td>
<td>R23, LCD cursor start scan line address</td>
<td>06</td>
<td>06</td>
<td>06</td>
</tr>
<tr>
<td>18</td>
<td>R24, LCD cursor end scan line address</td>
<td>07</td>
<td>07</td>
<td>07</td>
</tr>
<tr>
<td>19</td>
<td>R25, LCD alphanumeric cursor row address, high</td>
<td>03</td>
<td>03</td>
<td>xx</td>
</tr>
<tr>
<td>1A</td>
<td>R26, LCD alphanumeric cursor row address, low</td>
<td>C0</td>
<td>C0</td>
<td>C0</td>
</tr>
<tr>
<td>1B</td>
<td>R27, scan line address of character box center</td>
<td>04</td>
<td>04</td>
<td>04</td>
</tr>
<tr>
<td>1C</td>
<td>R28, LCD color/graphics cursor row address, low</td>
<td>xx</td>
<td>xx</td>
<td>0F</td>
</tr>
<tr>
<td>1D</td>
<td>R29, LCD color/graphics cursor row address, low</td>
<td>xx</td>
<td>xx</td>
<td>A0</td>
</tr>
</tbody>
</table>

Figure 2-23 (Part 2 of 2). Index and Data Registers
## Mode Control Register (Hex 3B8/3D8)

<table>
<thead>
<tr>
<th>Bit</th>
<th>Meaning</th>
</tr>
</thead>
<tbody>
<tr>
<td>7-6</td>
<td>Reserved</td>
</tr>
<tr>
<td>5</td>
<td>Enable blink</td>
</tr>
<tr>
<td>4</td>
<td>Not used</td>
</tr>
<tr>
<td>3</td>
<td>Video enable</td>
</tr>
<tr>
<td>2</td>
<td>Not used</td>
</tr>
<tr>
<td>1</td>
<td>Color/graphics mode only:</td>
</tr>
<tr>
<td></td>
<td>0 = Alphanumeric mode</td>
</tr>
<tr>
<td></td>
<td>1 = Graphics mode</td>
</tr>
<tr>
<td>0</td>
<td>Alphanumeric mode:</td>
</tr>
<tr>
<td></td>
<td>0 = 40 X 25 alphanumeric mode</td>
</tr>
<tr>
<td></td>
<td>1 = 80 X 25 alphanumeric mode</td>
</tr>
</tbody>
</table>
### Status Register (Hex 3BA/3DA)

**Bit Meaning**

7-4 Reserved

3 Vertical sync (50 Hz rate)

2-1 Reserved

0 Horizontal sync (toggles on status register read)

### Keyboard and Keyboard Controller

The keyboard consists of either 78 or 79 keys, a printed circuit board, a cable, and a connector to attach the keyboard to the system board. The system board supplies the drive and sense lines for the key switches.

The keyboard controller provides key detection, debounce, and typematic functions for the keyboard. The keyboard controller does not perform scan code translation or keystroke queuing functions (these functions are performed by the system processor).

When the keyboard controller detects a keystroke, the controller places the scan code data in a register (address hex 07D), generates an NMI, and stops further scanning until the BIOS keyboard routine has read the register.

The debounce timer is started whenever the logic detects that a key has been pressed. If the change is still present after 5 milliseconds, the new keystroke is presented to the processor. The typematic timer is set when any key is pressed. The initial period is 500 milliseconds. If after this period the key is still pressed, an additional interrupt is generated and passed to the processor at the rate of one every 100 milliseconds. This process continues until the key is released or another key is pressed. Pressing another key resets the timer to the 500 milliseconds period.
Together, the keyboard and keyboard controller provide all
the keyable functions (scan codes) other IBM Personal
Computers provide, but the layout and usage is somewhat
different. Figure 2-24 through Figure 2-29 show the keyboard
layouts for the various countries. The number in the
lower-right corner of the key button indicates the key number.

The function (Fn) key is used with other keys to generate
keystrokes normally found in the keypad section on the
keyboards of other IBM Personal Computers. For example,
holding the Fn key and pressing the cursor movement keys
will generate Home, End, Pg Up, and Pg Dn. The keypad
numerics, the plus sign (+), the minus sign (−), the multiply
sign (∗), the divide sign (/), and the period (.) are generated by
pressing the corresponding key when the keypad is active (Fn
and Num Lock keys have been pressed and released). These
characters are also generated when the corresponding key is
pressed along with the Fn key if the keypad is not active. The
keypad numeric keys are imbedded in the typing area.

Figure 2-24. U.S. English Keyboard
Figure 2-25. U.K. English Keyboard

Figure 2-26. French Keyboard

2-64 System Unit
Figure 2-27. German Keyboard

Figure 2-28. Italian Keyboard
Figure 2-29. Spanish Keyboard
Programming Considerations

Although the IBM PC Convertible keyboard layout is somewhat different than other IBM Personal Computer products, the IBM PC Convertible keyboard appears the same as other keyboards to the application program. Each keystroke causes a nonmaskable interrupt that causes a BIOS routine to execute. This BIOS routine translates the internal code into the standard scan code and stores the code in the register at hex 060, causing a level-1 hardware interrupt. Figure 2-30 on page 2-68 shows both the internal and the standard scan codes.

Note: Figure 2-30 shows only the make code. The internal break code can be found by adding hex 80 to the make code except in the following cases. In the cases where two scan codes are shown, both scan codes are generated for each make or break of the key.

<table>
<thead>
<tr>
<th>Key</th>
<th>Break Scan Code</th>
</tr>
</thead>
<tbody>
<tr>
<td>55</td>
<td>E0 B5 (when functioning as keypad /)</td>
</tr>
<tr>
<td>57</td>
<td>E0 B7 (when functioning as keypad *)</td>
</tr>
<tr>
<td>62</td>
<td>E0 B8</td>
</tr>
</tbody>
</table>

Note: BIOS does not provide ASCII translation for non-U.S. keyboards.
<table>
<thead>
<tr>
<th>Key Number</th>
<th>Character</th>
<th>Internal Code (hex)</th>
<th>Standard Scan Code (hex)</th>
</tr>
</thead>
<tbody>
<tr>
<td>01</td>
<td>~</td>
<td>11</td>
<td>29</td>
</tr>
<tr>
<td>02</td>
<td>!</td>
<td>12</td>
<td>02</td>
</tr>
<tr>
<td>03</td>
<td>@</td>
<td>13</td>
<td>03</td>
</tr>
<tr>
<td>04</td>
<td>#</td>
<td>14</td>
<td>04</td>
</tr>
<tr>
<td>05</td>
<td>$</td>
<td>15</td>
<td>05</td>
</tr>
<tr>
<td>06</td>
<td>%</td>
<td>16</td>
<td>06</td>
</tr>
<tr>
<td>07</td>
<td>^</td>
<td>17</td>
<td>07</td>
</tr>
<tr>
<td>08</td>
<td>&amp;</td>
<td>18</td>
<td>08</td>
</tr>
<tr>
<td>09</td>
<td>*</td>
<td>19</td>
<td>09</td>
</tr>
<tr>
<td>10</td>
<td>(</td>
<td>1A</td>
<td>0A</td>
</tr>
<tr>
<td>11</td>
<td>)</td>
<td>1B</td>
<td>0B</td>
</tr>
<tr>
<td>12</td>
<td>_</td>
<td>1C</td>
<td>0C</td>
</tr>
<tr>
<td>13</td>
<td>+</td>
<td>1D</td>
<td>0D</td>
</tr>
<tr>
<td>14</td>
<td>\</td>
<td>1E</td>
<td>0E</td>
</tr>
<tr>
<td>15</td>
<td>Backspace</td>
<td>1F</td>
<td>0E</td>
</tr>
<tr>
<td>16</td>
<td>Tab Btab</td>
<td>21</td>
<td>0F</td>
</tr>
</tbody>
</table>

Figure 2-30 (Part 1 of 4). Keyboard Scan Code Mapping

1This key number 14 exists only on U.S. keyboards.

2-68 System Unit
<table>
<thead>
<tr>
<th>Key Number</th>
<th>Character</th>
<th>Internal Code (hex)</th>
<th>Standard Scan Code (hex)</th>
</tr>
</thead>
<tbody>
<tr>
<td>17</td>
<td>q Q</td>
<td>22</td>
<td>10</td>
</tr>
<tr>
<td>18</td>
<td>w W</td>
<td>23</td>
<td>11</td>
</tr>
<tr>
<td>19</td>
<td>e E</td>
<td>24</td>
<td>12</td>
</tr>
<tr>
<td>20</td>
<td>r R</td>
<td>25</td>
<td>13</td>
</tr>
<tr>
<td>21</td>
<td>t T</td>
<td>26</td>
<td>14</td>
</tr>
<tr>
<td>22</td>
<td>y Y</td>
<td>27</td>
<td>15</td>
</tr>
<tr>
<td>23</td>
<td>u U</td>
<td>28</td>
<td>16</td>
</tr>
<tr>
<td></td>
<td>Keypad 4</td>
<td>28</td>
<td>4B</td>
</tr>
<tr>
<td>24</td>
<td>i I</td>
<td>29</td>
<td>17</td>
</tr>
<tr>
<td></td>
<td>Keypad 5</td>
<td>29</td>
<td>4C</td>
</tr>
<tr>
<td>25</td>
<td>o O</td>
<td>2A</td>
<td>18</td>
</tr>
<tr>
<td></td>
<td>Keypad 6</td>
<td>2A</td>
<td>4D</td>
</tr>
<tr>
<td>26</td>
<td>p P</td>
<td>2B</td>
<td>19</td>
</tr>
<tr>
<td>27</td>
<td>[ {</td>
<td>2C</td>
<td>1A</td>
</tr>
<tr>
<td>28</td>
<td>] }</td>
<td>2D</td>
<td>1B</td>
</tr>
<tr>
<td>30</td>
<td>Caps Lock</td>
<td>31</td>
<td>3A</td>
</tr>
<tr>
<td>31</td>
<td>a A</td>
<td>32</td>
<td>1E</td>
</tr>
<tr>
<td>32</td>
<td>s S</td>
<td>33</td>
<td>1F</td>
</tr>
<tr>
<td>33</td>
<td>d D</td>
<td>34</td>
<td>20</td>
</tr>
<tr>
<td>34</td>
<td>f F</td>
<td>35</td>
<td>21</td>
</tr>
<tr>
<td>35</td>
<td>g G</td>
<td>36</td>
<td>22</td>
</tr>
<tr>
<td>36</td>
<td>h H</td>
<td>37</td>
<td>23</td>
</tr>
<tr>
<td>37</td>
<td>j J</td>
<td>38</td>
<td>24</td>
</tr>
<tr>
<td></td>
<td>Keypad 1</td>
<td>38</td>
<td>4F</td>
</tr>
<tr>
<td>38</td>
<td>k K</td>
<td>39</td>
<td>25</td>
</tr>
<tr>
<td></td>
<td>Keypad 2</td>
<td>39</td>
<td>50</td>
</tr>
<tr>
<td>39</td>
<td>l L</td>
<td>3A</td>
<td>26</td>
</tr>
<tr>
<td></td>
<td>Keypad 3</td>
<td>3A</td>
<td>51</td>
</tr>
</tbody>
</table>

Figure 2-30 (Part 2 of 4). Keyboard Scan Code Mapping
<table>
<thead>
<tr>
<th>Key Number</th>
<th>Character</th>
<th>Internal Code (hex)</th>
<th>Standard Scan Code (hex)</th>
</tr>
</thead>
<tbody>
<tr>
<td>40</td>
<td>; :</td>
<td>3B</td>
<td>27</td>
</tr>
<tr>
<td>41</td>
<td>; &quot;</td>
<td>3C</td>
<td>28</td>
</tr>
<tr>
<td>42²</td>
<td></td>
<td>3D</td>
<td>2B</td>
</tr>
<tr>
<td>43</td>
<td>Enter</td>
<td>3E</td>
<td>1C</td>
</tr>
<tr>
<td>44</td>
<td>Shift (L)</td>
<td>41</td>
<td>2A</td>
</tr>
<tr>
<td>45²</td>
<td></td>
<td>54</td>
<td>56</td>
</tr>
<tr>
<td>46</td>
<td>z Z</td>
<td>42</td>
<td>2C</td>
</tr>
<tr>
<td>47</td>
<td>x X</td>
<td>43</td>
<td>2D</td>
</tr>
<tr>
<td>48</td>
<td>c C</td>
<td>44</td>
<td>2E</td>
</tr>
<tr>
<td>49</td>
<td>v V</td>
<td>45</td>
<td>2F</td>
</tr>
<tr>
<td>50</td>
<td>b B</td>
<td>46</td>
<td>30</td>
</tr>
<tr>
<td>51</td>
<td>n N</td>
<td>47</td>
<td>31</td>
</tr>
<tr>
<td>52</td>
<td>m M</td>
<td>48</td>
<td>32</td>
</tr>
<tr>
<td></td>
<td>Keypad 0</td>
<td>48</td>
<td>52</td>
</tr>
<tr>
<td>53</td>
<td>, &lt;</td>
<td>49</td>
<td>33</td>
</tr>
<tr>
<td>54</td>
<td>. &gt;</td>
<td>4A</td>
<td>34</td>
</tr>
<tr>
<td></td>
<td>Keypad .</td>
<td>4A</td>
<td>53</td>
</tr>
<tr>
<td>55</td>
<td>/ ?</td>
<td>4B</td>
<td>35</td>
</tr>
<tr>
<td></td>
<td>Keypad /</td>
<td>4B</td>
<td>E0 35³</td>
</tr>
<tr>
<td>56</td>
<td>Shift (R)</td>
<td>4C</td>
<td>36</td>
</tr>
<tr>
<td>57</td>
<td>* PrtSc</td>
<td>4E</td>
<td>37</td>
</tr>
<tr>
<td></td>
<td>Keypad *</td>
<td>4E</td>
<td>E0 37³</td>
</tr>
<tr>
<td>58</td>
<td>Ctrl</td>
<td>51</td>
<td>1D</td>
</tr>
<tr>
<td>59</td>
<td>Fn</td>
<td>52</td>
<td></td>
</tr>
<tr>
<td>60</td>
<td>Alt (L)</td>
<td>53</td>
<td>38</td>
</tr>
<tr>
<td>61</td>
<td>Space</td>
<td>56</td>
<td>39</td>
</tr>
<tr>
<td>62</td>
<td>Alt (R)</td>
<td>5A</td>
<td>E0 38³</td>
</tr>
<tr>
<td>63</td>
<td>Cur (L)</td>
<td>5B</td>
<td>4B</td>
</tr>
<tr>
<td></td>
<td>Home</td>
<td>5B</td>
<td>47</td>
</tr>
</tbody>
</table>

Figure 2-30 (Part 3 of 4). Keyboard Scan Code Mapping

²This key number exists only on non-U.S. keyboards.
³This key generates two scan codes for each make or break of the key.

2-70 System Unit
<table>
<thead>
<tr>
<th>Key Number</th>
<th>Character</th>
<th>Internal Code (hex)</th>
<th>Standard Scan Code (hex)</th>
</tr>
</thead>
<tbody>
<tr>
<td>64</td>
<td>Cur (up)</td>
<td>5C</td>
<td>48</td>
</tr>
<tr>
<td></td>
<td>Page Up</td>
<td>5C</td>
<td>49</td>
</tr>
<tr>
<td>65</td>
<td>Cur (dn)</td>
<td>5E</td>
<td>50</td>
</tr>
<tr>
<td></td>
<td>Page Dn</td>
<td>5E</td>
<td>51</td>
</tr>
<tr>
<td>66</td>
<td>Cur (R)</td>
<td>5F</td>
<td>4D</td>
</tr>
<tr>
<td></td>
<td>End</td>
<td>5F</td>
<td>4F</td>
</tr>
<tr>
<td>67</td>
<td>Esc</td>
<td>01</td>
<td>01</td>
</tr>
<tr>
<td></td>
<td>Sys Req</td>
<td>01</td>
<td>54</td>
</tr>
<tr>
<td>68</td>
<td>F1</td>
<td>02</td>
<td>3B</td>
</tr>
<tr>
<td></td>
<td>F11</td>
<td>02</td>
<td>57</td>
</tr>
<tr>
<td>69</td>
<td>F2</td>
<td>03</td>
<td>3C</td>
</tr>
<tr>
<td></td>
<td>F12</td>
<td>03</td>
<td>58</td>
</tr>
<tr>
<td>70</td>
<td>F3</td>
<td>04</td>
<td>3D</td>
</tr>
<tr>
<td>71</td>
<td>F4</td>
<td>05</td>
<td>3E</td>
</tr>
<tr>
<td>72</td>
<td>F5</td>
<td>06</td>
<td>3F</td>
</tr>
<tr>
<td>73</td>
<td>F6</td>
<td>07</td>
<td>40</td>
</tr>
<tr>
<td>74</td>
<td>F7</td>
<td>08</td>
<td>41</td>
</tr>
<tr>
<td>75</td>
<td>F8</td>
<td>09</td>
<td>42</td>
</tr>
<tr>
<td>76</td>
<td>F9</td>
<td>0A</td>
<td>43</td>
</tr>
<tr>
<td>77</td>
<td>F10</td>
<td>0B</td>
<td>44</td>
</tr>
<tr>
<td>78</td>
<td>Num Lock</td>
<td>0C</td>
<td>45</td>
</tr>
<tr>
<td>79</td>
<td>Scr Lock</td>
<td>0D</td>
<td>46</td>
</tr>
<tr>
<td>80</td>
<td>Ins</td>
<td>0E</td>
<td>52</td>
</tr>
<tr>
<td>81</td>
<td>Del</td>
<td>0F</td>
<td>53</td>
</tr>
</tbody>
</table>

Figure 2-30 (Part 4 of 4). Keyboard Scan Code Mapping
I/O Register 1 (Hex 60)

The I/O register located at hex 060 is used to hold the translated scan code data for the application. BIOS writes the translated scan code to this register and automatically generates a level-1 hardware interrupt. The data in the register is then available to the application.

**Bit Meaning**

7  Keyboard scan code 7
6  Keyboard scan code 6
5  Keyboard scan code 5
4  Keyboard scan code 4
3  Keyboard scan code 3
2  Keyboard scan code 2
1  Keyboard scan code 1
0  Keyboard scan code 0

Audio Controller and Speaker

The audio controller is used to drive the speaker. The controller receives control signals from both the I/O register and the system timers. The channel from the system timers is programmable within the functions of the timer with a 1.19 MHz input frequency. The speaker connects to a connector on the system board.
Programming Considerations

The audio controller uses a portion of the I/O register located at hex 061 for audio control functions. See Figure 2-5 on page 2-8 for a complete definition of this register.

I/O Register 2 (Hex 061)

The meanings of the bits assigned to the audio controller are:

<table>
<thead>
<tr>
<th>Bit</th>
<th>Meaning</th>
</tr>
</thead>
<tbody>
<tr>
<td>2</td>
<td>Enable speaker</td>
</tr>
<tr>
<td>1</td>
<td>Speaker data</td>
</tr>
<tr>
<td>0</td>
<td>Timer 2 gate</td>
</tr>
</tbody>
</table>

Note: BIOS controls the enable speaker bit through the Fn/Scroll Lock key sequence. The key sequence allows the user to disable the speaker.
Diskette Drive

The IBM PC Convertible diskette drive supports 90-millimeter (3.5-inch) double-sided, double-density diskettes with a formatted capacity of 720K bytes. Appendix B, "Unit Specifications" provides the functional characteristics of the diskette drives. "Disk Drive Interface" on page 2-76 provides a description of the interface to the drive.

The diskette drive uses modified frequency modulation (MFM) to read and write digital data, with a track-to-track access time of 6 milliseconds.

The diskette is loaded by inserting the diskette into the slot. Guides in the slot ensure that the diskette is in the correct position. When the drive is selected, the servo-controlled dc drive motor starts and drives the hub at a constant speed of 300 revolutions per minute (RPM). The head positioning system, which consists of a stepper motor and its associated electronics, moves the magnetic head to the desired track of the diskette. The stepper motor assembly uses one step-pulse to cause a one-track linear movement of the magnetic head. During a write operation, a 0.115-millimeter (0.0045-inch) data track is recorded with a 0.1875-millimeter (0.0073-inch) spacing (center-to-center) between the tracks. This allows 135 tracks per inch.

Data is read from the diskette by the data-recovery circuitry, which consists of a low-level read amplifier, differentiator, zero-crossing detector, and digitizing circuits. All data decoding is done by the diskette controller.
The diskette drive also has the following sensor systems:

- A track 00 sensor that detects when the head/carriage assembly is at track 00
- The write protect sensor that senses the position of the write protect tab
- The diskette-changed sensor that detects when a diskette has been removed from the drive
- The index sensor that detects the index marker.

**Diskette Controller**

The diskette controller consists of custom logic and a NEC µPD765 Floppy Disk Controller (or equivalent) that resides on the system board. The diskette controller attaches to the diskette drives through an internal interface. The timings and signal sequences are similar to the industry standard 133.4 millimeter (5.25-inch) diskette drive specification.

The diskette controller supports double-density, modified frequency modulation (MFM)-coded diskette drives and uses write precompensation with an analog phase-lock loop for clock and data recovery. The diskette drive parameters are programmable. In addition, the controller supports the diskette drive’s write-protect feature. The controller uses direct memory access (DMA) for record data transfers. Interrupt level 6 is used to indicate when an operation is complete and that a status condition requires processor attention.

In order to conserve power, power to the controller is removed whenever the diskettes drives are powered off. Any attempt to access the diskette controller when power is off causes an NMI to be signaled to the processor. BIOS then restores power to, and initializes, the controller before returning control to the requesting program.
Diskette Drive Interface

The drives are attached to the diskette controller and the power supply through a 40-pin connector. Figure 2-31 shows the diskette drive connector pins. The plus (+) or minus (−) preceding the signal name indicates the active state of the signals. The input/output column indicates whether the signal direction is to or from the diskette controller.

<table>
<thead>
<tr>
<th>I/O Pin</th>
<th>Signal Name</th>
<th>Input/Output</th>
</tr>
</thead>
<tbody>
<tr>
<td>01-04</td>
<td>No connection</td>
<td></td>
</tr>
<tr>
<td>05</td>
<td>Ground</td>
<td></td>
</tr>
<tr>
<td>06</td>
<td>No connection</td>
<td></td>
</tr>
<tr>
<td>07</td>
<td>Ground</td>
<td></td>
</tr>
<tr>
<td>08</td>
<td>− Index</td>
<td>Input</td>
</tr>
<tr>
<td>09</td>
<td>Ground</td>
<td></td>
</tr>
<tr>
<td>10</td>
<td>No connection</td>
<td></td>
</tr>
<tr>
<td>11</td>
<td>Ground</td>
<td></td>
</tr>
</tbody>
</table>

Figure 2-31 (Part 1 of 2). Diskette Drive Connector
<table>
<thead>
<tr>
<th>I/O Pin</th>
<th>Signal Name</th>
<th>Input/Output</th>
</tr>
</thead>
<tbody>
<tr>
<td>12</td>
<td>— Drive Select</td>
<td>Output</td>
</tr>
<tr>
<td>13</td>
<td>Ground</td>
<td></td>
</tr>
<tr>
<td>14</td>
<td>No connection</td>
<td></td>
</tr>
<tr>
<td>15</td>
<td>Ground</td>
<td></td>
</tr>
<tr>
<td>16</td>
<td>— Motor Enable</td>
<td>Output</td>
</tr>
<tr>
<td>17</td>
<td>Ground</td>
<td></td>
</tr>
<tr>
<td>18</td>
<td>— Direction</td>
<td>Output</td>
</tr>
<tr>
<td>19</td>
<td>Ground</td>
<td></td>
</tr>
<tr>
<td>20</td>
<td>— Step</td>
<td>Output</td>
</tr>
<tr>
<td>21</td>
<td>Ground</td>
<td></td>
</tr>
<tr>
<td>22</td>
<td>— Write Data</td>
<td>Output</td>
</tr>
<tr>
<td>23</td>
<td>Ground</td>
<td></td>
</tr>
<tr>
<td>24</td>
<td>— Write Enable</td>
<td>Output</td>
</tr>
<tr>
<td>25</td>
<td>Ground</td>
<td></td>
</tr>
<tr>
<td>26</td>
<td>— Track 0</td>
<td>Input</td>
</tr>
<tr>
<td>27</td>
<td>Ground</td>
<td></td>
</tr>
<tr>
<td>28</td>
<td>— Write Protect</td>
<td>Input</td>
</tr>
<tr>
<td>29</td>
<td>Ground</td>
<td></td>
</tr>
<tr>
<td>30</td>
<td>— Read Data</td>
<td>Input</td>
</tr>
<tr>
<td>31</td>
<td>Ground</td>
<td></td>
</tr>
<tr>
<td>32</td>
<td>— Head Select</td>
<td>Output</td>
</tr>
<tr>
<td>33</td>
<td>Ground</td>
<td></td>
</tr>
<tr>
<td>34</td>
<td>— Diskette Changed</td>
<td>Input</td>
</tr>
<tr>
<td>35</td>
<td>Ground</td>
<td></td>
</tr>
<tr>
<td>36</td>
<td>Ground</td>
<td></td>
</tr>
<tr>
<td>37</td>
<td>Ground</td>
<td></td>
</tr>
<tr>
<td>38</td>
<td>+5 V dc</td>
<td>Power</td>
</tr>
<tr>
<td>39</td>
<td>Return for pin 38</td>
<td></td>
</tr>
<tr>
<td>40</td>
<td>+12 V dc</td>
<td>Power</td>
</tr>
</tbody>
</table>

Figure 2-31 (Part 2 of 2). Diskette Drive Connector
The following paragraphs describe the interface lines.

**Drive Select**

The 'drive select' line is used to enable or disable all other drive interface lines except 'motor enable.' When the 'drive select' line is set to the low level, the drive is enabled and considered active. When the 'select' line is set to the high level, input lines are ignored and all output lines are disabled. If both the 'select' and the 'motor enable' lines are set to the high level, the drive is considered as being in a standby (low power) mode.

**Motor Enable**

The 'motor enable' line is used to start the spindle motor. When the 'motor enable' line is set to the low level, the motor is activated; when the line is set to the high level, the motor decelerates and stops.

**Step**

A 1-microsecond (minimum) low-level pulse on this line causes the read/write head to move one track. The direction of the motion is determined by the level of the 'direction' line at the trailing edge of the pulse.

**Direction**

When the 'direction' line is at the high level, a pulse on the 'step' line causes the read/write head to move one track away from the drive spindle. When this line is set to the low level, a pulse on the 'step' line causes the read/write head to move one track toward the drive spindle.

**Head Select**

When the 'head select' line is set to the low level, head 1 (upper) is selected. When this line is set to the high level, head 0 (lower) is selected.

2-78 System Unit
Write Enable

When the ‘write enable’ line is set to the low level, the write circuits are enabled and information can be written to the diskette under control of the ‘write data’ line.

Write Data

A 250 nanosecond (minimum) low-level pulse on this line causes a bit to be written onto the diskette, if the ‘write enable’ line is at the low level.

Index

The ‘index’ line provides a 1.0 millisecond (minimum) low-level pulse for each revolution of the motor.

Track 0

The ‘track 0’ line is set to the low level when the read/write head is positioned on track 0 of the diskette.

Write Protect

The ‘write protect’ line is set to the low level, if the diskette inserted in the selected drive is write-protected.

Read Data

The ‘read data’ line provides a 250 nanosecond (minimum) low-level pulse for each bit read from the diskette.

Diskette Changed

The ‘diskette changed’ line is set to the low level, if power is set to on or if a diskette is removed from the drive. This line is set to the high level when a diskette is in the drive and a ‘step’ pulse is sent to the drive while the drive is selected.
Programming Considerations

Figure 2-32 shows the diskette control register assignments.

**Warning:** Application programs that do not use the BIOS timer 0 interrupt routine to turn off the diskette motor must ensure that a drive select is not done within 2 milliseconds of a motor off command. Failing to observe this rule can result in data loss on the drive being selected.

<table>
<thead>
<tr>
<th>Address (hex)</th>
<th>R/W</th>
<th>Usage</th>
</tr>
</thead>
<tbody>
<tr>
<td>077</td>
<td>R/W</td>
<td>Diskette control register</td>
</tr>
<tr>
<td>3F2</td>
<td>W</td>
<td>Digital output register</td>
</tr>
<tr>
<td>3F4</td>
<td>R</td>
<td>Main status register</td>
</tr>
<tr>
<td>3F5</td>
<td>R/W</td>
<td>Data register</td>
</tr>
<tr>
<td>3F7</td>
<td>R</td>
<td>Digital input register</td>
</tr>
</tbody>
</table>

Figure 2-32. Diskette Control Registers

**Diskette Control Register (Hex 077)**

The diskette control register is used to provide the various control functions. BIOS power-on routines use this register when powering up the controller. This register is unique to the IBM PC Convertible and application programs should avoid using this register in order to preserve application compatibility.
Digital-Output Register (Hex 3F2)

The digital-output register (DOR) is an output-only register used to control the drive motor and selection. All bits are cleared by the 'reset' line. The bits have the following meaning:

<table>
<thead>
<tr>
<th>Bit</th>
<th>Meaning</th>
</tr>
</thead>
<tbody>
<tr>
<td>6-7</td>
<td>Reserved</td>
</tr>
<tr>
<td>5</td>
<td>Drive 1 motor enable</td>
</tr>
<tr>
<td>4</td>
<td>Drive 0 motor enable</td>
</tr>
<tr>
<td>3</td>
<td>Enable interrupt/DMA. This bit allows interrupt and DMA request from the diskette controller to be gated onto the I/O interface. If this bit is cleared, the interrupt and DMA request interfaces are disabled.</td>
</tr>
<tr>
<td>2</td>
<td>Reset controller. The diskette controller is held reset when this bit is clear. The bit is set by BIOS to enable the diskette controller.</td>
</tr>
<tr>
<td>1-0</td>
<td>The hardware uses these bits to select the drive:</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Bit</th>
<th>Selected Drive</th>
</tr>
</thead>
<tbody>
<tr>
<td>1 0</td>
<td></td>
</tr>
<tr>
<td>0 0</td>
<td>0</td>
</tr>
<tr>
<td>0 1</td>
<td>1</td>
</tr>
</tbody>
</table>
The microprocessor in the diskette controller contains two registers that can be accessed by the system processor: a main status register and a data register. The 8-bit main status register contains the status information of the diskette controller and can be accessed at any time. The 8-bit data register (actually consisting of several registers in a stack, with only one register present to the data bus at a time) stores data, commands, parameters, and diskette drive status information. Data bytes are read from or written to the data register in order to program or obtain results after a particular command. The main status register is read-only and is used during processor/controller data transfer operations.

### Bit Meaning

- **7** Request for master. When this bit is set to 1, the data register is available for use.
- **6** Indicates the direction of data transfer between the system processor and the diskette controller:
  
  - 1 = Transfer data from diskette controller
  - 0 = Transfer data from system processor
- **5** If this bit is set to 1, the diskette controller is not in DMA mode.
- **4** A read, write, or seek operation is in process (busy).
- **3-2** Reserved
- **1** Diskette drive 1 is in seek mode.
- **0** Diskette drive 0 is in seek mode.
**Digital Input Register (Hex 3F7)**

This register is a read-only register and is used for diagnostics.

<table>
<thead>
<tr>
<th>Bit</th>
<th>Meaning</th>
</tr>
</thead>
<tbody>
<tr>
<td>7</td>
<td>Media change line active</td>
</tr>
<tr>
<td>6</td>
<td>Drive select 0</td>
</tr>
<tr>
<td>5</td>
<td>Drive select 1</td>
</tr>
<tr>
<td>4</td>
<td>Drive 0 motor enable</td>
</tr>
<tr>
<td>3</td>
<td>Drive 1 motor enable</td>
</tr>
<tr>
<td>2</td>
<td>Write data buffer</td>
</tr>
<tr>
<td>1</td>
<td>Step buffer</td>
</tr>
<tr>
<td>0</td>
<td>Track 0 indicator</td>
</tr>
</tbody>
</table>

*System Unit 2-83*
Portable Printer Interface

The printer interface supports the attachment of the IBM PC Convertible Portable Printer.

The printer interface uses a programmable rate generator. The power-on routines check to determine if the system is configured with the IBM PC Convertible Portable Printer. If this is the case, the power-on routines set the generator to 1200 bits per second for the IBM PC Convertible Portable Printer.

Interface

The printer adapter interface is provided at the I/O channel connector. The interface consists of four signal lines:

- Transmit Data
- Printer Busy
- Printer Error
- Printer Enable

These lines are described in "I/O Channel" on page 2-36.

The printer data is provided on the 'transmit data' line in the following format:

Data bit 0 is the first bit to be transmitted. The interface automatically inserts the start bit, and two stop bits.

2-84 System Unit
Programming Considerations

Figure 2-33 shows the register assignments for the printer adapter. Bit 7 of the printer mode control register is used to control accesses to the other two registers. When this bit is set to 1, the rate divisor latches are accessed; when this bit is set to 0, the data and status registers are accessed.

<table>
<thead>
<tr>
<th>Address (hex)</th>
<th>R/W</th>
<th>Usage</th>
</tr>
</thead>
<tbody>
<tr>
<td>078</td>
<td>R/W</td>
<td>Data register/rate divisor latch (LSB)</td>
</tr>
<tr>
<td>079</td>
<td>R/W</td>
<td>Status register/rate divisor latch (MSB)</td>
</tr>
<tr>
<td>07A</td>
<td>R/W</td>
<td>Mode control register</td>
</tr>
</tbody>
</table>

Figure 2-33. Printer Adapter Control Registers

Data Register (Hex 078)

*Bit Meaning*

7-0 Data bits 7 through 0

Rate Divisor Latch Least Significant Bits (Hex 078)

*Bit Meaning*

7-0 Bits 7 through 0
Rate Divider Latch Most Significant Bits (Hex 079)

*Bit Meaning*

7-0 Bits 15 through 8

Status Register (Hex 079)

*Bit Meaning*

7 Printer not busy
6 Set to 1
5 Set to 0
4 Set to 1
3 Not printer error
2 Interrupt request
1 Diagnostic sense
0 Not transmitting

2-86 System Unit
Mode Control Register (Hex 07A)

The mode control register is used to access the registers at hex 078 and hex 079. When bit 7 of the register is set to 1, writing to these locations loads the rate divisor latches. (The rate divisor latch is loaded by the power-on routines with a value of hex 0F89 to establish the 1200 rate for the printer.) When bit 7 is set to 0, a write-to-address hex 078 loads the data into the transmit buffer and a read-to-address hex 079 accesses the status register.

**Bit Meaning**

7  Access rate divisor latch

6  Diagnostic mode

5  Enable error status interrupt

4  Enable data register empty interrupt

3  Busy (diagnostic):
   - 0 = Reset busy
   - 1 = Set busy

2  Printer enable

1-0  Reserved
Real-Time Clock

The real-time clock provides the time of day with alarm, 100-year calendar, and programmable interrupt functions. The clock operates in either 12- or 24-hour mode and compensates for daylight savings time, end of month, and leap years. The real-time clock uses a Motorola MC146818A Real-Time Clock (or equivalent).

The IBM PC Convertible uses three types of real-time clock interrupts:

Periodic Interrupt

This interrupt is used by the post/wait-on-time (interrupt hex 15) function calls. This interrupt can occur once every 976.56 microseconds.

Alarm Interrupt

This interrupt is used by the time-of-day (interrupt hex 1A) interrupt to activate the system at a specified time and to notify an application that a specified time of day has been reached. This interrupt causes an alarm (interrupt hex 4A) function call and can occur once every 24 hours, unless the interrupt is set to a new time or reset within a 24-hour period.

Update Ended Interrupt

This interrupt is activated by BIOS when the LCD blank, low-battery warning, or auto-power-off options are enabled in the system profile. The interrupt is used as a time base to determine if keyboard or diskette activity does not occur within a given period of time. The interrupt can occur once each second.

The real-time clock function and registers should be accessed through BIOS function interrupts (interrupt hex 1A).
Programming Considerations

The programming interface to the real-time clock consists of 64 memory locations. These locations are divided into clock control locations (14) and RAM locations (50). The power-on routines and BIOS use these locations for system profile and system configuration information.

These registers are accessed indirectly. In order to access these locations, the register address must first be written into I/O address hex 070. Data bits 5 through 0 should contain the register address; data bits 6 and 7 are not used. Data can then be written to or read from the register by writing to or reading from address hex 071. Figure 2-34 on page 2-90 shows the register assignments for the real-time clock.

**Note:** Interrupts must be disabled during the time the registers at hex 070 and 071 are being accessed.
<table>
<thead>
<tr>
<th>Address (hex)</th>
<th>R/W</th>
<th>Usage</th>
</tr>
</thead>
<tbody>
<tr>
<td>000</td>
<td>R/W</td>
<td>Seconds</td>
</tr>
<tr>
<td>001</td>
<td>R/W</td>
<td>Seconds alarm</td>
</tr>
<tr>
<td>002</td>
<td>R/W</td>
<td>Minutes</td>
</tr>
<tr>
<td>003</td>
<td>R/W</td>
<td>Minutes alarm</td>
</tr>
<tr>
<td>004</td>
<td>R/W</td>
<td>Hours</td>
</tr>
<tr>
<td>005</td>
<td>R/W</td>
<td>Hours alarm</td>
</tr>
<tr>
<td>006</td>
<td>R/W</td>
<td>Day of week</td>
</tr>
<tr>
<td>007</td>
<td>R/W</td>
<td>Day of month</td>
</tr>
<tr>
<td>008</td>
<td>R/W</td>
<td>Month</td>
</tr>
<tr>
<td>009</td>
<td>R/W</td>
<td>Year</td>
</tr>
<tr>
<td>00A</td>
<td>R/W</td>
<td>Register A</td>
</tr>
<tr>
<td>00B</td>
<td>R/W</td>
<td>Register B</td>
</tr>
<tr>
<td>00C</td>
<td>R</td>
<td>Register C</td>
</tr>
<tr>
<td>00D</td>
<td>R</td>
<td>Register D</td>
</tr>
<tr>
<td>00E-03F</td>
<td>R/W</td>
<td>50 bytes reserved RAM</td>
</tr>
</tbody>
</table>

Figure 2-34. Real-time Clock Control Registers
### Register A (Hex 0A)

**Bit Meaning**

**7** Update in progress flag

**6-4** Crystal frequency:

- **010** = 32.768K Hz

**3-0** Interrupt rate:

- **0000** = No periodic interrupt
- **0001** = 3.90625 milliseconds
- **0010** = 7.8125 milliseconds
- **0011** = 122.070 microseconds
- **0100** = 244.141 microseconds
- **0101** = 488.281 microseconds
- **0110** = 976.562 microseconds
- **0111** = 1.953125 milliseconds
- **1000** = 3.90625 milliseconds
- **1001** = 7.8123 milliseconds
- **1010** = 15.625 milliseconds
- **1011** = 31.25 milliseconds
- **1100** = 62.5 milliseconds
- **1101** = 125 milliseconds
- **1110** = 250 milliseconds
- **1111** = 500 milliseconds
Register B (Hex 0B)

Bit  Meaning

7  Set clock
6  Periodic interrupt enable
5  Alarm interrupt enable
4  Update ended interrupt enable
3  Square wave interrupt enable
2  Binary data mode (not binary coded decimal)
1  24-hour mode
0  Daylight savings enable

Register C (Hex 0C)

Bit  Meaning

7  Interrupt request flag
6  Periodic interrupt flag
5  Alarm interrupt flag
4  Update ended interrupt flag
3-0  Always set to 0
Register D (Hex 0D)

<table>
<thead>
<tr>
<th>Bit</th>
<th>Meaning</th>
</tr>
</thead>
<tbody>
<tr>
<td>7</td>
<td>Valid RAM and time</td>
</tr>
<tr>
<td>6-0</td>
<td>Always set to 0</td>
</tr>
</tbody>
</table>

Power Supply

The power supply is contained inside the system unit and provides the power for the system unit and attachments. The supply provides five voltage levels and is rated at 12 watts.

The supply accepts input from four dc sources: a battery pack, an ac adapter, an automobile adapter, or a battery charger (used only to charge the batteries).

Power Supply Interface

The external interface to the power supply is made through either the battery pack connector or the input power connector. Electrical specifications for the input connectors are given in Appendix B, “Unit Specificaitons.”
Section 3. System Options

Serial/Parallel Adapter (Serial Interface) ........... 3-3
I/O Channel Interface .......................... 3-4
Communications Interface ....................... 3-5
Programming Considerations ...................... 3-11
Serial/Parallel Adapter (Parallel Interface) ........ 3-19
I/O Channel Interface .......................... 3-19
Printer Interface ............................... 3-20
Programming Considerations ...................... 3-24
Internal Modem .................................. 3-28
System Board Interface ........................... 3-29
Programming Considerations ....................... 3-32
Modem Commands .................................. 3-41
CRT Display Adapter .............................. 3-53
I/O Channel Interface ........................... 3-60
Programming Considerations ....................... 3-63
Portable Printer ................................. 3-68
Portable Printer Character Set ..................... 3-70
Portable Printer Commands ........................ 3-73
Monochrome Display .............................. 3-89
Color Display ..................................... 3-90
Automobile Power Adapter ......................... 3-90
Battery Charger .................................. 3-90
Serial/Parallel Adapter (Serial Interface)

The IBM PC Convertible Serial/Parallel Adapter is a feature that provides both serial (RS-232C) communications and parallel printer interface adapters in a single external module. These adapters share a common system interface; however, the functions of these adapters are logically separate and are described in two parts. The description of the parallel interface portion of the IBM PC Convertible Serial/Parallel Adapter feature begins on page 3-19.

The IBM PC Convertible Serial/Parallel Adapter provides the same basic functions as the IBM Personal Computer Asynchronous Communications Adapter. However, additional commands are provided to allow programming to control local power to the adapter. These commands are processed by the PC Convertible BIOS.

The serial adapter provides functions equivalent to those provided by an INS8250A Asynchronous Communications Element in conjunction with system and EIA interfaces. The power-on self-test routines determine the presence of communications adapters by using the work (scratch) register within the INS8250A Asynchronous Communications Element. These routines will not be able to detect the presence of adapters that do not contain this register.

Applications that process multiple interrupt conditions from the INS8250A Asynchronous Communications Element must service and clear the interrupt conditions before exiting the interrupt service routine. Failure to clear the interrupt conditions can result in failure of the application program.

The serial interface is set to primary or secondary by BIOS. If the IBM PC Convertible Internal Modem is installed, BIOS sets the serial interface to secondary; otherwise, the serial interface is set to primary.

The adapter can be programmed to operate from 110 baud to 9600 baud through a BIOS function call (interrupt hex 14).
Local power to the serial/parallel adapter is controlled through system software. When the system unit is powered on and external power is being used, the serial/parallel adapter is automatically activated by the power-on routines. The system profile is used to determine if power is to be applied to the adapter when the system unit is operating on battery power.

**I/O Channel Interface**

The lines of the I/O channel are provided at the rear of the system unit on a 72-pin connector. All 72 lines pass through the IBM PC Convertible Serial/Parallel Adapter attachment; however, only those lines shown in Figure 1-3 are used within the serial/parallel adapter. Refer to “I/O Channel” on page 2-36 for a complete description of the I/O channel. The plus (+) or minus (−) preceding the signal name indicates the active state of the signals. The input/output column indicates whether the signal direction is to or from the serial/parallel adapter.

<table>
<thead>
<tr>
<th>Signal Name</th>
<th>Input/Output</th>
</tr>
</thead>
<tbody>
<tr>
<td>+ Address/Data Bits 0-7</td>
<td>Input/Output</td>
</tr>
<tr>
<td>+ Address Bits 8 and 9</td>
<td>Input</td>
</tr>
<tr>
<td>+ Address Enable</td>
<td>Input</td>
</tr>
<tr>
<td>− Data Enable</td>
<td>Input</td>
</tr>
<tr>
<td>+ Address Latch Enable</td>
<td>Input</td>
</tr>
<tr>
<td>+ Reset</td>
<td>Input</td>
</tr>
<tr>
<td>− I/O Write</td>
<td>Input</td>
</tr>
<tr>
<td>− I/O Read</td>
<td>Input</td>
</tr>
<tr>
<td>+ Interrupt Request 3, 4, and 7</td>
<td>Output</td>
</tr>
</tbody>
</table>

*Figure 3-1. System Interface Signals*
The following paragraphs describe how some of the lines are used within the serial/parallel adapter. For a description of the remainder of the lines, refer to “I/O Channel” on page 2-36.

**Reset**

The ‘reset’ line is used to remove local power from the serial/parallel adapter, isolate the serial/parallel adapter from the system bus, and set the serial/parallel adapter as secondary.

After local power to the serial/parallel adapter is turned on or if the adapter is reset, the serial interface is enabled by an I/O read at address hex 3FF if the adapter is set for primary, or 2FF if the adapter has been set for secondary. A 100-millisecond delay is required after the power on or reset is required before the read is issued.

The power-on routines automatically provide a ‘reset’ before the application program is given control.

**Interrupt Request 3 and 4**

Interrupt request 4 is used if the asynchronous adapter is set to primary. Interrupt request 3 is used if the adapter is set to secondary.

**Communications Interface**

The electrical interface to the serial adapter complies with the Electronic Industries Association RS-232C specification for interface of Data Terminal Equipment and Data Communication Equipment (August 1969). Not all of the signals allowed in the RS-232C specification are supported by the serial adapter. Only those signals shown in Figure 3-2 on page 3-7 are supported. The input/output column indicates whether the direction is to or from the adapter.
3-6 System Options
<table>
<thead>
<tr>
<th>I/O Pin</th>
<th>Signal Name</th>
<th>Input/Output</th>
</tr>
</thead>
<tbody>
<tr>
<td>01</td>
<td>No connection</td>
<td></td>
</tr>
<tr>
<td>02</td>
<td>Transmitted Data</td>
<td>Output</td>
</tr>
<tr>
<td>03</td>
<td>Received Data</td>
<td>Input</td>
</tr>
<tr>
<td>04</td>
<td>Request to Send</td>
<td>Output</td>
</tr>
<tr>
<td>05</td>
<td>Clear to Send</td>
<td>Input</td>
</tr>
<tr>
<td>06</td>
<td>Data Set Ready</td>
<td>Input</td>
</tr>
<tr>
<td>07</td>
<td>Ground</td>
<td>Ground</td>
</tr>
<tr>
<td>08</td>
<td>Carrier Detect</td>
<td>Input</td>
</tr>
<tr>
<td>09-10</td>
<td>No Connection</td>
<td></td>
</tr>
<tr>
<td>11</td>
<td>Diagnostic(^1)</td>
<td>Output</td>
</tr>
<tr>
<td>12-19</td>
<td>No Connection</td>
<td></td>
</tr>
<tr>
<td>20</td>
<td>Data Terminal Ready</td>
<td>Output</td>
</tr>
<tr>
<td>21</td>
<td>No Connection</td>
<td></td>
</tr>
<tr>
<td>22</td>
<td>Ring Indicator</td>
<td>Input</td>
</tr>
<tr>
<td>23-25</td>
<td>No Connection</td>
<td></td>
</tr>
</tbody>
</table>

\(^1\)Pin 11 is connected to pin 20 on this connector for diagnostic wrap purposes.

Figure 3-2. Parallel Interface Connector
The EIA drivers and receivers used on the serial/parallel adapter are of the inverting type. The following paragraphs describe these signals.

**Carrier Detect**

When low, this signal indicates that the data carrier has been detected by the modem or data set. This signal is a modem control function input whose condition can be tested by the system by reading bit 7 of the modem status register. Bit 3 of the modem status register indicates whether the ‘carrier detect’ line has changed since the previous reading.

Whenever bit 7 of the modem status register changes state, an interrupt is generated, if the modem status interrupt is enabled.

‘Carrier detect’ is also called ‘data carrier detect’ or ‘received line signal detect.’

**Data Set Ready**

When low, this signal indicates that the modem or data set is ready to establish the communications link and transfer data with the serial adapter. The ‘data set ready’ signal is a modem control function whose condition can be tested by the processor by reading bit 5 of the modem status register. Bit 1 of the modem status register indicates whether the ‘data set ready’ line has changed since the previous reading of the modem status register.

Whenever bit 5 of the modem status register changes state, an interrupt is generated if the modem status interrupt is enabled.
Clear to Send

This signal is a modem control function input whose condition can be tested by the processor by reading bit 4 of the modem status register. Bit 0 of the modem status register indicates whether the ‘clear to send’ line has changed state since the previous reading of the modem status register.

Whenever bit 4 of the modem status register changes state, an interrupt is generated if the modem status interrupt is enabled.

Request to Send

When this signal is low, this signal informs the modem or data set that the serial adapter is ready to transmit data. The ‘request to send’ signal can be set to an active low by setting bit 1 of the modem control register. The ‘request to send’ signal is set high upon a master reset operation, and it is forced to high during loop mode operation.

Receive Data

This is serial data input from the communication link (peripheral device, modem, or data set).

Transmit Data

This is the serial data sent on the communications link. The format of the transmitted data is identical to the received data.

Data bit 0 is the first bit to be transmitted or received. The adapter automatically inserts the start bit, the correct parity bit (if programmed to do so), and the stop bit (1, 1-1/2, or 2, depending on the command in the line control register).
Ring Indicator

When low, this signal indicates that a telephone ringing signal has been received by the modem or data set. The signal is a modem control function input whose condition can be tested by the processor by reading bit 6 of the modem status register. Bit 2 of the modem status register indicates whether the ‘ring indicator’ line has changed from a low to high state since the previous reading of the modem status register.

Whenever bit 6 of the modem status register changes from a high to a low state, an interrupt is generated if the modem status interrupt is enabled.

Data Terminal Ready

When low, this signal informs the modem or data set that the adapter is ready to communicate. The ‘data terminal ready’ signal can be set to an active low by programming bit 0 of the modem control register to a high level. The ‘data terminal ready’ signal is set high upon a master reset operation, and is forced to high during loop mode operation.
Programming Considerations

The serial adapter is programmed to perform the various functions by using the registers at addresses hex 3F8 through hex 3FF (primary) or hex 2F8 through 2FF (secondary).

Figure 3-3 shows the register assignment for the asynchronous adapter. Bit 7, the divisor latch access bit (DLAB), in the line control register, is used to select certain registers. The setting of this bit is indicated when it is needed to select a register.

<table>
<thead>
<tr>
<th>Address (hex)</th>
<th>R/W</th>
<th>Usage</th>
</tr>
</thead>
<tbody>
<tr>
<td>07C</td>
<td>R/W</td>
<td>System register</td>
</tr>
<tr>
<td>xF8</td>
<td>R/W</td>
<td>Transmit buffer (DLAB = 0 Write)</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Receive buffer (DLAB = 0 Read)</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Divisor latch LSB (DLAB = 1)</td>
</tr>
<tr>
<td>xF9</td>
<td>R/W</td>
<td>Divisor latch MSD (DLAB = 1)</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Interrupt enable register</td>
</tr>
<tr>
<td>xFA</td>
<td>R</td>
<td>Interrupt identification registers</td>
</tr>
<tr>
<td>xFB</td>
<td>R/W</td>
<td>Line control register</td>
</tr>
<tr>
<td>xFC</td>
<td>R/W</td>
<td>Modem control register</td>
</tr>
<tr>
<td>xFD</td>
<td>R/W</td>
<td>Line status register</td>
</tr>
<tr>
<td>xFE</td>
<td>R/W</td>
<td>Modem status register</td>
</tr>
<tr>
<td>xFF</td>
<td>R/W</td>
<td>Work (scratch) register</td>
</tr>
</tbody>
</table>

Figure 3-3. Asynchronous Adapter Control Registers
System Register (07C)

The power-on routines and BIOS control the setting of this register. Applications should avoid using this register in order to preserve compatibility among application programs.

The system register is an 8-bit read/write register. Bits 0 and 2 are associated with the serial/parallel adapter. The remaining bits are used by other parts of the system and must be preserved.

Bit  Meaning

7-3  Must be preserved

2  Adapter local power control:
   0 = Power off  
   1 = Power on  
   This bit affects both the serial and parallel interfaces.

1  Must be preserved

0  Primary/secondary:
   0 = Secondary  
   1 = Primary  

Transmit Buffer (Hex 3F8/2F8)

The transmit buffer contains the character to be serially transmitted. Bit 0 is the least significant bit and is the first bit serially transmitted.

Receive Buffer (Hex 3F8/2F8)

The receive buffer contains the received data. Bit 0 is the least significant bit and is the first bit serially received.
Divisor Register (Hex 3F8/2F8 and 3F9/2F9)

The serial/parallel adapter contains a programmable rate generator that can divide the clock input (1.8432 MHz) by any divisor from 1 to \((2^{16}-1)\). The output frequency of the baud generator is 16 times the rate \[\text{divisor \#} = \frac{\text{frequency input}}{(\text{rate} \times 16)}\]. Two 8-bit registers store the divisor in a 16-bit binary format. These divisor latches must be loaded during initialization in order to ensure desired operation of the rate generator. Upon loading either of the divisor latches, a 16-bit rate counter is immediately loaded. This prevents long counts on initial load.

Interrupt Enable Register (Hex 3F9/2F9)

The interrupt enable register is used to selectively enable and disable the interrupts. Disabling an interrupt also inhibits setting the appropriate identification bit in the interrupt identification register. All other system functions operate in their normal manner, including the setting of the line status and modem status registers. Bit 3 of the modem control register is used as a master interrupt enable/disable control.

**Bit Meaning**

4-7 These bits are always 0.

3 Enable modem status interrupt

2 Enable receiver line status interrupt

1 Enable transmitter holding register empty interrupt

0 Enable received data available interrupt
Interrupt Identification Register (Hex 3FA/2FA)

The interrupt identification is used to signal that an interrupt is pending and to identify the source of the interrupt.

<table>
<thead>
<tr>
<th>Bit</th>
<th>Meaning</th>
</tr>
</thead>
<tbody>
<tr>
<td>3-7</td>
<td>These bits are always set to 0.</td>
</tr>
<tr>
<td>1-2</td>
<td>Interrupt identification:</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Bit</th>
<th>Interrupt Source</th>
</tr>
</thead>
<tbody>
<tr>
<td>2</td>
<td>Interrupt Source</td>
</tr>
<tr>
<td>1</td>
<td>Receiver line status</td>
</tr>
<tr>
<td>1</td>
<td>Received data available</td>
</tr>
<tr>
<td>0</td>
<td>Transmitter holding register empty</td>
</tr>
<tr>
<td>0</td>
<td>Modem status</td>
</tr>
</tbody>
</table>

0 Interrupt pending indicator. This bit is set to 1 when an interrupt is pending.

Line Control Register (Hex 3FB/2FB)

The application specifies the format of the asynchronous data communications exchange through the line control register. In addition to controlling the format, the application may retrieve the contents of the line control register for inspection. This feature simplifies system programming and eliminates the need for separate storage in system memory of the line characteristics.

<table>
<thead>
<tr>
<th>Bit</th>
<th>Meaning</th>
</tr>
</thead>
<tbody>
<tr>
<td>7</td>
<td>Divisor latch access bit (DLAB):</td>
</tr>
</tbody>
</table>

0 = Access receive buffer, transmit holding register, or interrupt enable register
1 = Access divisor latches
6 Set break control. If this bit is set to 1, the serial data output is forced to a 0 (space).

5 Stick parity. If this bit is set to 0, the parity bit is transmitted as a 0 (space).

4 Select even/odd parity. If this bit is set to 0, odd parity is used, otherwise even parity is used.

3 Enable parity. If this bit is set to 1, parity is enabled.

2 Number of stop bits transmitted or received in each serial character. If bit 2 is a 0, 1 stop bit is generated or checked in the transmit or receive data respectively. If bit 2 is a 1 when a 5-bit word length is selected, 1-1/2 stop bits are generated or checked. If bit 2 is a 1 when either a 6-, 7-, or 8-bit word length is selected, 2 stop bits are generated or checked.

0-1 Length of word transmitted or received:

<table>
<thead>
<tr>
<th>Bit</th>
<th>Length of Word</th>
</tr>
</thead>
<tbody>
<tr>
<td>0 0</td>
<td>5 bits</td>
</tr>
<tr>
<td>0 1</td>
<td>6 bits</td>
</tr>
<tr>
<td>1 0</td>
<td>7 bits</td>
</tr>
<tr>
<td>1 1</td>
<td>8 bits</td>
</tr>
</tbody>
</table>
Modem Control Register (Hex 3FC/2FC)

The modem control register controls the interface with the modem or data set (or other peripheral device).

**Bit Meaning**

5-7 These bits are always 0.

4 This bit provides a loopback feature for diagnostic testing.

3 Output 2. If this bit is set to 1, interrupt 4 (primary) or interrupt 3 (secondary) is enabled. If this bit is set to 0, all interrupts are disabled regardless of the setting of the interrupt enable register.

2 Output 1

1 Request to send

0 Data terminal ready

Line Status Register (Hex 3FD/2FD)

This 8-bit register provides status information to the processor concerning the data transfer.

**Bit Meaning**

7 This bit is permanently set to 0.

6 Transmitter empty indicator. This bit is set to 0 when both the transmitter holding and the transmitter shift registers are empty.

5 Transmitter holding register empty indicator. This bit is set to 0 when the holding register is empty.

3-16 System Options
4 Break interrupt indicator. This bit is set to 1 when the received data was held in spacing state longer than full-word transmission time.

3 Frame error indicator. This bit is set to 1 when the received character did not have a valid stop character.

2 Parity error indicator. This bit is set to 1 when a parity error occurs.

1 Overrun error indicator. This bit is set to 1 when the data in the receive buffer has been overwritten.

0 Data ready indicator. This bit is set to 1 when the receive buffer contains valid data.

**Modem Status Register (Hex 3FE/2FE)**

The modem status register provides the current state of the control lines from the modem to the processor. In addition, 4 bits in this register provides change information. These bits are set to a logical 1 whenever a control input from the modem changes state. They are reset when they are read.

**Bit Meaning**

7 Received line signal detect. If bit 4 (loop) of the modem control register is set to a 1, this bit is equivalent to ‘output 2’ of the modem control register.

6 Ring indicator. If bit 4 (loop) of the modem control register is set to a 1, this bit is equivalent to ‘output 1’ in the modem control register.

5 Data set ready. If bit 4 (loop) of the modem control register is set to a 1, this bit is equivalent to ‘data terminal ready’ in the modem control register.
4 Clear to send. If bit 4 (loop) of the modem control register is set to a 1, this is equivalent to ‘request to send’ in the modem control register.

3 Delta received line signal detector indicator. This bit indicates that the ‘received line signal detector’ has changed state since last time it was read.

   **Note:** Whenever bit 0, 1, 2, or 3 is set to a 1, a modem status interrupt is generated, if the appropriate interrupt enable bit is set in the interrupt enable register.

2 Ring indicator detector. This bit indicates that the ‘ring indicator’ signal has changed from logical 1 level to logical 0 level.

1 Delta data set ready indicator. This bit indicates that the ‘data set ready’ signal has changed state since the last time it was read.

0 Delta clear to send indicator. This bit indicates that the ‘clear to send’ signal has changed state since the last time it was read.
Serial/Parallel Adapter (Parallel Interface)

The IBM PC Convertible Serial/Parallel Adapter is a feature that provides both serial (RS-232C) communications and parallel printer interface adapters in a single external attachment. These adapters share a common system interface; however, the functions of these adapters are logically separate and are described in two parts. The description of the serial interface portion of the IBM PC Convertible Serial/Parallel Adapter feature begins on page 3-3.

The parallel interface (parallel printer interface) is specifically designed to attach printers that have a parallel interface. The interface can also be used as a general purpose input/output port for any device or application that matches its input/output capabilities.

I/O Channel Interface

The parallel adapter shares the system interface with the serial adapter. See “I/O Channel Interface” on page 3-4 for a description of the lines used in the IBM PC Convertible Serial/Parallel Adapter feature.

The parallel adapter uses the ‘interrupt request 7’ for interrupts.
Printer Interface

Devices can attach to the parallel adapter through the connector. Figure 3-4 on page 3-21 shows the connector pin assignments. The plus (+) or minus (−) preceding the signal name indicates the active state of the signals. The input/output column indicates whether the signal direction is to or from the adapter.
<table>
<thead>
<tr>
<th>I/O Pin</th>
<th>Signal Name</th>
<th>Input/Output</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>- Strobe</td>
<td>Output</td>
</tr>
<tr>
<td>2</td>
<td>+ Printer Data Bit 0</td>
<td>Output</td>
</tr>
<tr>
<td>3</td>
<td>+ Printer Data Bit 1</td>
<td>Output</td>
</tr>
<tr>
<td>4</td>
<td>+ Printer Data Bit 2</td>
<td>Output</td>
</tr>
<tr>
<td>5</td>
<td>- Printer Data Bit 3</td>
<td>Output</td>
</tr>
<tr>
<td>6</td>
<td>+ Printer Data Bit 4</td>
<td>Output</td>
</tr>
<tr>
<td>7</td>
<td>+ Printer Data Bit 5</td>
<td>Output</td>
</tr>
<tr>
<td>8</td>
<td>+ Printer Data Bit 6</td>
<td>Output</td>
</tr>
<tr>
<td>9</td>
<td>+ Printer Data Bit 7</td>
<td>Output</td>
</tr>
<tr>
<td>10</td>
<td>- Acknowledge</td>
<td>Input</td>
</tr>
<tr>
<td>11</td>
<td>+ Busy</td>
<td>Input</td>
</tr>
<tr>
<td>12</td>
<td>+ Paper End</td>
<td>Input</td>
</tr>
<tr>
<td>13</td>
<td>+ Selected</td>
<td>Input</td>
</tr>
<tr>
<td>14</td>
<td>- Automatic Feed</td>
<td>Output</td>
</tr>
<tr>
<td>15</td>
<td>- Error</td>
<td>Input</td>
</tr>
<tr>
<td>16</td>
<td>- Initialize</td>
<td>Output</td>
</tr>
<tr>
<td>17</td>
<td>- Select In</td>
<td>Output</td>
</tr>
<tr>
<td>18 - 25</td>
<td>Ground</td>
<td>Ground</td>
</tr>
</tbody>
</table>

Figure 3-4. Parallel Printer Connector

The connector signals are described in the following section.

**Strobe**

The ‘strobe’ pulse is used to write data to the printer. Data is strobed into the printer at the low level.

This line is set to the high level by a system reset.
**Printer Data Bits 0 through 7**

These lines contain the 8 bits of printer data.

**Acknowledge**

This line is used to signal that data has been received by the printer and the printer is ready to receive other data. This line is active when set to the low level. The signal pulse is approximately 5 microseconds in duration. This signal is used to activate the printer interrupt (interrupt request 7).

**Busy**

This line is used to indicate that the printer cannot receive data. The line is active when it is set to the high level and is made active during the following conditions:

- During data entry
- During printing operations
- In the off-line state
- In a printer-error status.

**Paper End**

This line is used to indicate that the printer is out of paper. The line is active when set to the high level.

**Selected**

This line indicates that the printer is in the selected state.

**Auto Feed**

If this line is low, paper is automatically fed one line after printing.

3-22 System Options
**Initialize**

This line is used to reset the printer controller to the initial state and to reset the print buffer. The line is normally set to the high level. The signal pulse must be greater than 50 microseconds at the receiving terminal.

**Error**

This line is used to signal the following conditions:

- End-of-paper
- Off-line state
- Printer error status.

This line is active when set to the low level.

**Select In**

This line enables the printer to receive data when set to the low level.
Programming Considerations

Figure 3-5 shows the register assignments for the parallel adapter.

<table>
<thead>
<tr>
<th>Address (hex)</th>
<th>R/W</th>
<th>Usage</th>
</tr>
</thead>
<tbody>
<tr>
<td>07C</td>
<td>R/W</td>
<td>System register</td>
</tr>
<tr>
<td>378</td>
<td>R/W</td>
<td>Printer data register</td>
</tr>
<tr>
<td>379</td>
<td>R</td>
<td>Printer status register</td>
</tr>
<tr>
<td>37A</td>
<td>R/W</td>
<td>Printer control register</td>
</tr>
</tbody>
</table>

Figure 3-5. Parallel Adapter Control Registers

3-24 System Options
System Register (07C)

The power-on routines and BIOS control the setting of this register. Applications should avoid using this register in order to preserve compatibility among application programs.

The system register is an 8-bit read/write register. Bits 0 and 2 are associated with the Serial/Parallel Adapter. The remaining bits are used by other parts of the system and must be preserved.

<table>
<thead>
<tr>
<th>Bit</th>
<th>Meaning</th>
</tr>
</thead>
<tbody>
<tr>
<td>7-3</td>
<td>Must be preserved</td>
</tr>
<tr>
<td>2</td>
<td>Adapter local power control:</td>
</tr>
<tr>
<td></td>
<td>0 = Power off</td>
</tr>
<tr>
<td></td>
<td>1 = Power on</td>
</tr>
<tr>
<td></td>
<td>This bit affects both the serial and parallel interfaces.</td>
</tr>
<tr>
<td>1</td>
<td>Must be preserved</td>
</tr>
<tr>
<td>0</td>
<td>Primary/secondary:</td>
</tr>
<tr>
<td></td>
<td>0 = Secondary</td>
</tr>
<tr>
<td></td>
<td>1 = Primary</td>
</tr>
</tbody>
</table>

Printer Data Register (Hex 378)

The printer data register is an 8-bit read/write register used to write data out to the printer. The latched outputs of this register are present on the output connector.

<table>
<thead>
<tr>
<th>Bit</th>
<th>Meaning</th>
</tr>
</thead>
<tbody>
<tr>
<td>7-0</td>
<td>Printer data bits 7 through 0</td>
</tr>
</tbody>
</table>
Printer Status Register (Hex 379)

The printer status register is a 6-bit read-only register that provides printer status and interrupt status to the system. This register differs from other IBM Personal Computers in that bit 2 is used to provide the user with interrupt status.

Bit Meaning

7 Busy. Note, if this bit is set to 1, then the 'busy' input signal is at low level.

6 Acknowledge

5 Paper end

4 Select

3 Error

2 Interrupt status. If the interrupt is enabled, the 'acknowledge' signal from the printer sets this bit to 1. The bit is cleared when the status register is read.

1-0 Reserved
Printer Control Register (Hex 37A)

The printer control register is a 5-bit read/write register used for the parallel printer control lines and the printer interrupt enable. For three of the control lines (‘strobe,’ ‘auto feed,’ and ‘select in’), the data present on the output connector pin is the complement of what was written to the register. When the register is read, however, it is the register contents that are read, not the pin status.

**Bit Meaning**

7-5  Reserved

4  Interrupt enable. If this bit is set to 1, interrupt request 7 is signalled.

3  Select in

2  Initialize

1  Auto feed

0  Strobe
Internal Modem

The IBM PC Convertible Internal Modem is a feature that provides a phone line interface. The feature can be programmed to operate at line speeds of 1200, 300, or 110 bits per second. The feature is connected to the system board.

The feature consists of two major elements: a communication element (an INS8250A Asynchronous Communications Element or equivalent) and a modem (modulator-demodulator) element. The communications element is controlled through the system registers. These registers are described in “Programming Considerations” on page 3-32. The modem element is controlled by the modem commands that are passed to the modem element in the data stream. The modem commands are described in “Modem Commands” on page 3-41. The modem commands are stripped from the data stream and executed, they are not transmitted to the receiving station.

The power-on routines initialize the communications element. The modem element is initialized automatically at power on from parameters in the system profile:

- Baud rate: 110, 300, or 1200 (1200 is the default)
- Parity: Even, odd, mark, space, or none (even is the default)
- Answer: Automatic or manual (manual is the default).

The default parameters can be changed through a system profile utility or through a BIOS function call (interrupt hex 15).

The power-on self-test routines determine the presence of communications adapters by using the work (scratch) register within the INS8250A Asynchronous Communications Element. These routines will not be able to detect the presence of adapters that do not contain this register.
Applications that process multiple interrupt conditions from the INS8250A Asynchronous Communications Element must service and clear the interrupt conditions before exiting the interrupt service routine. Failure to clear the interrupt conditions can result in failure of the application program.

Local power to the internal modem is controlled through system software. When the system is powered on and external power is being used, the internal modem is automatically activated by the power-on routines. The system profile is used to determine if power is to be applied to the internal modem when the system is operating on battery power. Applications can control power to the internal modem through a BIOS function call (interrupt hex 15).

All pacing of the interface and control signal status must be handled by the application program.

**System Board Interface**

The internal modem feature is attached to the system unit through a connector on the system board. Figure 3-6 on page 3-30 shows the connector pins. The plus (+) or minus (−) preceding the signal name indicates the active state of the signals. The input/output column indicates whether the signal direction is to or from the system unit.
<table>
<thead>
<tr>
<th>I/O Pin</th>
<th>Signal Name</th>
<th>Input/Output</th>
</tr>
</thead>
<tbody>
<tr>
<td>01</td>
<td>+ Address/Data Bit 0</td>
<td>Input/Output</td>
</tr>
<tr>
<td>02</td>
<td>+ Address/Data Bit 1</td>
<td>Input/Output</td>
</tr>
<tr>
<td>03</td>
<td>+ Address/Data Bit 2</td>
<td>Input/Output</td>
</tr>
<tr>
<td>04</td>
<td>+ Address/Data Bit 3</td>
<td>Input/Output</td>
</tr>
<tr>
<td>05</td>
<td>+ Address/Data Bit 4</td>
<td>Input/Output</td>
</tr>
<tr>
<td>06</td>
<td>+ Address/Data Bit 5</td>
<td>Input/Output</td>
</tr>
<tr>
<td>07</td>
<td>+ Address/Data Bit 6</td>
<td>Input/Output</td>
</tr>
<tr>
<td>08</td>
<td>+ Address/Data Bit 7</td>
<td>Input/Output</td>
</tr>
<tr>
<td>09</td>
<td>+ Address Bit 8</td>
<td>Input</td>
</tr>
<tr>
<td>10</td>
<td>+ Address Bit 9</td>
<td>Input</td>
</tr>
</tbody>
</table>

3-30 System Options
<table>
<thead>
<tr>
<th>I/O Pin</th>
<th>Signal Name</th>
<th>Input/Output</th>
</tr>
</thead>
<tbody>
<tr>
<td>11</td>
<td>+ Interrupt Request 4</td>
<td>Output</td>
</tr>
<tr>
<td>12</td>
<td>– I/O Read</td>
<td>Input</td>
</tr>
<tr>
<td>13</td>
<td>+ Reset</td>
<td>Input</td>
</tr>
<tr>
<td>14</td>
<td>– Data Enable</td>
<td>Input</td>
</tr>
<tr>
<td>15</td>
<td>Ground</td>
<td>Ground</td>
</tr>
<tr>
<td>16</td>
<td>+ Address Latch Enable</td>
<td>Input</td>
</tr>
<tr>
<td>17</td>
<td>Ground</td>
<td>Ground</td>
</tr>
<tr>
<td>18</td>
<td>+ Address Enable</td>
<td>Input</td>
</tr>
<tr>
<td>19</td>
<td>Ground</td>
<td>Ground</td>
</tr>
<tr>
<td>20</td>
<td>+12 V dc</td>
<td>Power</td>
</tr>
<tr>
<td>21</td>
<td>Ground</td>
<td>Ground</td>
</tr>
<tr>
<td>22</td>
<td>Ground</td>
<td>Ground</td>
</tr>
<tr>
<td>23</td>
<td>Ground</td>
<td>Ground</td>
</tr>
<tr>
<td>24</td>
<td>Ground</td>
<td>Ground</td>
</tr>
<tr>
<td>25</td>
<td>–13 V dc</td>
<td>Power</td>
</tr>
<tr>
<td>26</td>
<td>Ground</td>
<td>Ground</td>
</tr>
<tr>
<td>27</td>
<td>– I/O Write</td>
<td>Input</td>
</tr>
<tr>
<td>28</td>
<td>+5 V dc</td>
<td>Power</td>
</tr>
<tr>
<td>29</td>
<td>Ground</td>
<td>Ground</td>
</tr>
<tr>
<td>30</td>
<td>+High Z</td>
<td>Input</td>
</tr>
</tbody>
</table>

Figure 3-6 (Part 2 of 2). System Board Modem Connector

All of the preceding interface lines (with the exception of the ‘high Z’) are common to the I/O channel and are described in “I/O Channel” on page 2-36. The description of the ‘high Z’ line follows.

High Z

The ‘high Z’ line is generated by the system unit. This line indicates that all high-order address lines (10-15) are at the low level. The internal modem uses this line to form an I/O select.
Common Carrier Interface

The internal modem has one USOC RJ11 jack used for the telephone cable connector.

The common carrier interface is a 600-ohm, balanced, two-wire telephone interface design that meets the FCC Part 68 rules. One 2.13-meter (7-foot) modular telephone cord is included with the IBM PC Convertible internal modem.

When the internal modem is operating at 1200 bits per second, the modulation methods and frequency tolerances conform to Bell 212A specifications. When the internal modem is operating at either 110 or 300 bits per second, the modulation methods and frequency tolerances conform to Bell 103A specifications.

Programming Considerations

Figure 3-7 on page 3-33 shows the register assignments for the communications element in the internal modem. Bit 7, the divisor latch access bit (DLAB), in the line control register is used to select certain registers. The setting of this bit is indicated if it is needed to select a register.
<table>
<thead>
<tr>
<th>Address (hex)</th>
<th>R/W</th>
<th>Usage</th>
</tr>
</thead>
<tbody>
<tr>
<td>07C</td>
<td>R/W</td>
<td>System register</td>
</tr>
</tbody>
</table>
| 3F8          | R/W | Transmit buffer (DLAB = 0 Write)  
Receive buffer (DLAB = 0 Read)  
Divisor latch LSB (DLAB = 1) |
| 3F9          | R/W | Divisor latch MSB (DLAB = 1)  
Interrupt enable register |
| 3FA          | W   | Interrupt identification registers |
| 3FB          | W   | Line control register |
| 3FC          | W   | Modem control register |
| 3FD          | W   | Line status register |
| 3FE          | W   | Modem status register |
| 3FF          | R/W | Work (scratch) register |

Figure 3-7. Control Registers
System Register (07C)

The power-on routines and BIOS control the setting of this register. Applications should avoid using this register in order to preserve compatibility among application programs.

The system register is an 8-bit read/write register. Only bit 1 is associated with the modem. The remaining bits are used by other parts of the system and must be preserved.

<table>
<thead>
<tr>
<th>Bit</th>
<th>Meaning</th>
</tr>
</thead>
<tbody>
<tr>
<td>7-2</td>
<td>Must be preserved</td>
</tr>
<tr>
<td>1</td>
<td>Modem power control:</td>
</tr>
<tr>
<td></td>
<td>0 = Power off</td>
</tr>
<tr>
<td></td>
<td>1 = Power on</td>
</tr>
<tr>
<td>0</td>
<td>Must be preserved</td>
</tr>
</tbody>
</table>

Transmit Buffer (Hex 3F8)

The transmit buffer contains the character to be serially transmitted. Bit 0 is the least significant bit and is the first bit serially transmitted.

Receive Buffer (Hex 3F8)

The receive buffer contains the received data. Bit 0 is the least significant bit and is the first bit serially received.
Divisor Register (Hex 3F8 and 3F9)

The internal modem contains a programmable rate generator that can divide the clock input (1.8432 MHz) by any divisor from 1 to \((2^{16}-1)\). The output frequency of the baud generator is 16 times the rate \([\text{divisor} = \frac{\text{frequency input}}{\text{rate} \times 16}]\). Two 8-bit registers store the divisor in a 16-bit binary format. These divisor latches must be loaded during initialization in order to ensure desired operation of the rate generator. Upon loading either of the divisor latches, a 16-bit rate counter is immediately loaded. This prevents long counts on initial load.

Interrupt Enable Register (Hex 3F9)

The interrupt enable register is used to selectively enable and disable the interrupts. Disabling an interrupt also inhibits setting the appropriate identification bit in the interrupt identification register. All other system functions operate in their normal manner, including the setting of the line status and modem status registers. Bit 3 of the modem control register is used as a master interrupt enable/disable control.

**Bit Meaning**

4-7 These bits are always 0.

3 Enable modem status interrupt

2 Enable receiver line status interrupt

1 Enable transmitter holding register empty interrupt

0 Enable received data available interrupt
Interrupt Identification Register (Hex 3FA)

The interrupt identification is used to signal that an interrupt is pending and to identify the source of the interrupt.

**Bit Meaning**

3-7 These bits are always set to 0.

1-2 Interrupt identification:

<table>
<thead>
<tr>
<th>Bit</th>
<th>Meaning</th>
</tr>
</thead>
<tbody>
<tr>
<td>2</td>
<td><strong>Interrupt Source</strong></td>
</tr>
<tr>
<td>1</td>
<td>Receiver line status</td>
</tr>
<tr>
<td>1</td>
<td>Received data available</td>
</tr>
<tr>
<td>0</td>
<td>Transmitter holding register empty</td>
</tr>
<tr>
<td>0</td>
<td>Modem status</td>
</tr>
<tr>
<td>0</td>
<td>Interrupt pending indicator. This bit is set to 1 when an interrupt is pending.</td>
</tr>
</tbody>
</table>

Line Control Register (Hex 3FB)

The application specifies the format of the asynchronous data communications exchange through the line control register. In addition to controlling the format, the application may retrieve the contents of the line control register for inspection. This feature simplifies system programming and eliminates the need for separate storage in system memory of the line characteristics.
**Bit Meaning**

7 Divisor latch access bit (DLAB):

0 = Access receive buffer, transmit holding register, or interrupt enable register
1 = Access divisor latches

6 Set break control. If this bit is set to 1, the serial data output is forced to a 0 (space).

5 Stick parity. If this bit is set to 0, the parity bit is transmitted as a 0 (space).

4 Select even/odd parity. If this bit is set to 0, odd parity is used, otherwise even parity is used.

3 Enable parity. If this bit is set to 1, parity is enabled.

2 Number of stop bits transmitted or received in each serial character. If bit 2 is a 0, 1 stop bit is generated or checked in the transmit or receive data respectively. If bit 2 is a 1 when a 5-bit word length is selected, 1-1/2 stop bits are generated or checked. If bit 2 is a 1 when either a 6-, 7-, or 8-bit word length is selected, 2 stop bits are generated or checked.

0-1 Length of word transmitted or received:

<table>
<thead>
<tr>
<th>Bit</th>
<th>Length of Word</th>
</tr>
</thead>
<tbody>
<tr>
<td>00</td>
<td>5 bits</td>
</tr>
<tr>
<td>01</td>
<td>6 bits</td>
</tr>
<tr>
<td>10</td>
<td>7 bits</td>
</tr>
<tr>
<td>11</td>
<td>8 bits</td>
</tr>
</tbody>
</table>
Modem Control Register (Hex 3FC)

The modem control register controls the interface with the modem or data set (or other peripheral device).

Bit Meaning

5-7 These bits are always 0.

4 This bit provides a loopback feature for diagnostic testing.

3 Output 2. If this bit is set to 1, interrupt 4 (primary) or interrupt 3 (secondary) is enabled. If this bit is set to 0, all interrupts are disabled regardless of the setting of the interrupt enable register.

2 Output 1

1 Request to send

0 Data terminal ready
**Line Status Register (Hex 3FD)**

This 8-bit register provides status information to the processor concerning the data transfer.

<table>
<thead>
<tr>
<th>Bit</th>
<th>Meaning</th>
</tr>
</thead>
<tbody>
<tr>
<td>7</td>
<td>This bit is permanently set to 0.</td>
</tr>
<tr>
<td>6</td>
<td>Transmitter empty indicator. This bit is set to 0 when both the transmitter holding and the transmitter shift registers are empty.</td>
</tr>
<tr>
<td>5</td>
<td>Transmitter holding register empty indicator. This bit is set to 0 when the holding register is empty.</td>
</tr>
<tr>
<td>4</td>
<td>Break interrupt indicator. This bit is set to 1 when the received data was held in spacing state longer than full word transmission time.</td>
</tr>
<tr>
<td>3</td>
<td>Frame error indicator. This bit is set to 1 when the received character did not have a valid stop character.</td>
</tr>
<tr>
<td>2</td>
<td>Parity error indicator. This bit is set to 1 when a parity error occurs.</td>
</tr>
<tr>
<td>1</td>
<td>Overrun error indicator. This bit is set to 1 when the data in the receive buffer has been overwritten.</td>
</tr>
<tr>
<td>0</td>
<td>Data ready indicator. This bit is set to 1 when the receive buffer contains valid data.</td>
</tr>
</tbody>
</table>
Modem Status Register (Hex 3FE)

The modem status register provides the current state of the control lines from the modem to the processor. In addition, 4 bits in this register provide change information. These bits are set to a logical 1 whenever a control input from the modem changes state. They are reset when they are read.

Bit Meaning

7  Received line signal detect. If bit 4 (loop) of the modem control register is set to a 1, this bit is equivalent to ‘output 2’ of the modem control register.

6  Ring indicator. If bit 4 (loop) of the modem control register is set to a 1, this bit is equivalent to ‘output 1’ in the modem control register.

5  Data set ready. If bit 4 (loop) of the modem control register is set to a 1, this bit is equivalent to ‘data terminal ready’ in the modem control register.

4  Clear to send. If bit 4 (loop) of the modem control register is set to a 1, this is equivalent to ‘request to send’ in the modem control register.

3  Delta received line signal detector indicator. This bit indicates that the ‘received line signal detector’ has changed state since last time it was read.

Note: Whenever bit 0, 1, 2, or 3 is set to a 1, a modem status interrupt is generated, if the appropriate interrupt enable bit is set in the interrupt enable register.

2  Ring indicator detector. This bit indicates that the ‘ring indicator’ signal has changed from logical 1 level to logical 0 level.

3-40 System Options
Delta data set ready indicator. This bit indicates that the 'data set ready' signal has changed state since the last time it was read.

Delta clear to send indicator. This bit indicates that the 'clear to send' signal has changed state since the last time it was read.

**Modem Commands**

The modem commands in the data stream to the modem element have the following structure:

```
[cc][command word][delimiter][arguments][,more][CR]
```

where:

- `[cc]` is the command code.
- `[command word]` is the command word or the first letter of the command word.
- `[delimiter]` is always a space when separating an argument and command word. Any spaces thereafter are ignored until the modem receives a comma, an argument, or a carriage return.
- `[arguments]` are variables that are replaced by any character allowed by the command definition.
- `[,,more]` is any additional commands, preceded by a comma.
- `[CR]` is a carriage return that completes the command sequence and causes the modem to execute the commands. The modem responds with a question mark when a command has not been entered correctly.
**Answer (A)**

This command takes the modem off-hook and begins the answer handshake with the calling modem. If the modem is in the voice mode when the incoming call is received, the Answer command puts it in the modem mode.

**Break (Bn)**

This command sends a break character for a duration of a multiple of 100 milliseconds (n X 100 milliseconds). The parameter n is any hexadecimal digit 1 - F.

**Count (C n)**

This command sets the ring count for automatically answering an incoming call and for dialing. The modem will auto answer the phone after the number of incoming rings equals the count (n) specified. When dialing, the modem will wait n + 3 rings. A typical ring cycle is 6 seconds. Because a remote modem may timeout a call after 45 seconds, a value of n greater than 8 may not be useful. A parameter of 0 specifies that the modem should not answer an incoming call, but should still perform instructions from the system. The parameter n is any hexadecimal digit from 0 - F.

At initialization time, the count is set to 0.
Dial (D m..m)

This command takes the modem off-hook and searches for a dial tone. When one is found, the modem dials the phone number in the ASCII string, then searches for line busy, ringing, or incoming carriers. If the parameter is omitted, the number from the most recently entered Dial, Redial, or Xmit command is redialed. If a parameter has never been entered, the modem responds with a question mark (?).

The characters in the dial string (m...m) may be any ASCII character except the backspace or abort characters. Only the following characters will be executed:

- 0 through 9, #, and * are the digits to be dialed. There can be up to 33 characters in the character string.

- I - Dial the digits that follow the I using pulse dialing. Dialing ends when a P or a W is encountered. The I must be at the beginning of the dial string or be preceded by P or W.

- P - Wait for a dial tone, then immediately dials the next dial string. If a dial tone is not received within five seconds (or five times the number of Ps, if more than one P has been entered), dialing stops and a 'no dial tone' response is returned to the application.

  If the string does not start with a P, then one P is assumed.

- W - Wait for dial tone (five seconds), then immediately dials the next string, if no P follows. Multiples of W extend the time.
**Format (F n)**

This command sets the format of the data being sent by the modem to the system. It affects the number of stop bits transmitted, but not the parity of the data on the telephone line. Telephone line data duplicates the parity sent by the system.

The parameter n is interpreted by the modem as follows.

<table>
<thead>
<tr>
<th>n</th>
<th>Parity</th>
<th>Data Length</th>
<th>Stop Bit</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>Mark</td>
<td>7</td>
<td>1</td>
</tr>
<tr>
<td>1</td>
<td>Space</td>
<td>7</td>
<td>1</td>
</tr>
<tr>
<td>2</td>
<td>Odd</td>
<td>7</td>
<td>1</td>
</tr>
<tr>
<td>3</td>
<td>Even</td>
<td>7</td>
<td>1</td>
</tr>
<tr>
<td>4</td>
<td>None</td>
<td>8</td>
<td>1</td>
</tr>
</tbody>
</table>

Although other combinations are possible, the IBM PC Convertible internal modem supports only these formats.

The application must set the same format as defined in the Format command into the line control register in order to enable data or command communication.

The response to the Format command will be in the old format.

Do not combine this command with any other commands, except the Speed command.

At initialization time, the format is set to format 3.

**Hangup (H)**

This command causes the modem to immediately end communication, send a 2-second long space, then go on-hook. During this process, the modem disregards system commands and data.
Initialize (I)

This command causes the modem to run the self-test and initialize the modem to the default values. When this command is used, the application should wait 2 seconds before issuing the next command. When the Initialize command is issued, or when power is turned on, the communications element must be set to the initialization state:

- A verification of hardware integrity is performed and the result is posted to the status characters.
- The status is cleared.
- The modem is placed in the data state to await a dialing request or incoming ring.
- The transparent mode is cleared.
- All loopback modes are cleared.
- The wait mode is cleared.
- The command character is set to Control-N.
- The data format is set to 7 data bits, even parity, and 1 stop bit.
- Ringback count is set to 0 (auto-answer disabled).
- The modem is set to on-hook.
- The message mode is set to long format.
- The speed is set to 1200 bits per second.
- The dial buffer is cleared.
Long Response o (L o)

This command causes the modem to respond to the system with either a long or short format of a response message. When the long format is used, the modem sends a carriage return with the response so that the cursor returns to the left margin. When the short format is used, no carriage return is sent and the cursor remains on that line until another command is sent. Figure 3-8 shows the format of the responses.

The parameter 0 may be either 0 for the long message format or 1 for short message format. The dial string is not returned when the short format is used.

<table>
<thead>
<tr>
<th>Long Format</th>
<th>Short Format (hex)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Busy</td>
<td>30</td>
</tr>
<tr>
<td>Connected</td>
<td>31</td>
</tr>
<tr>
<td>No answer</td>
<td>32</td>
</tr>
<tr>
<td>No dial tone</td>
<td>33</td>
</tr>
<tr>
<td>OK</td>
<td>34</td>
</tr>
<tr>
<td>Ring</td>
<td>35</td>
</tr>
<tr>
<td>Unsuccessful</td>
<td>36</td>
</tr>
<tr>
<td>?</td>
<td>37</td>
</tr>
</tbody>
</table>

Figure 3-8. Response Formats

At initialization state time, the response mode is set to long format.
Modem (M)

This command causes the modem to return from voice mode to the data communications mode. If the modem is on-hook, no further action occurs. If the modem is off-hook and the answer mode was the last data mode used in the current call, then the Modem command begins the answer handshake. In any other situation where the Modem command is issued off-hook, the modem begins the originate handshake.

New (N p)

This command changes the command character (cc) to a new value. The former command character is then treated as a data character and can be transmitted as normal data. The parameter p can be any ASCII character.

The character that is used as the control character may not be used as part of a command stream or text stream, except as a control character in a command string. The hex combinations 08, 0D, 18, 20, 2C, 8E, 8D, 98, A0, and AC should not be used as command characters.

The default is Ctrl N (hex 0E).

Originate (O)

This command causes the modem to go off-hook, forces the modem mode, and begins the originate handshake. If the modem is in the voice mode when this command is used, it returns to the modem mode.

Pickup (P)

This command takes the modem off-hook and puts the modem in the voice mode. After a Pickup command has been issued, a Hangup command must be issued when the call is finished to return to the on-hook state.
**Query (Q)**

This command is used to retrieve modem status information:

<table>
<thead>
<tr>
<th><strong>Status</strong></th>
<th><strong>Meaning</strong></th>
</tr>
</thead>
<tbody>
<tr>
<td>H0</td>
<td>On hook</td>
</tr>
<tr>
<td>H1</td>
<td>Off hook</td>
</tr>
<tr>
<td>S0 to SF</td>
<td>Current ring count setting in hex</td>
</tr>
<tr>
<td>B</td>
<td>Line busy</td>
</tr>
<tr>
<td>D</td>
<td>No dial-tone</td>
</tr>
<tr>
<td>L</td>
<td>Successful dial</td>
</tr>
<tr>
<td>N</td>
<td>Dial tone present after dialing or unsuccessful handshake</td>
</tr>
<tr>
<td>X</td>
<td>No answer, ring count plus n exceeded</td>
</tr>
<tr>
<td>T0</td>
<td>Integrity test passed</td>
</tr>
<tr>
<td>T1</td>
<td>Integrity test failed</td>
</tr>
</tbody>
</table>

H0, H1, or S0 to SF status is always returned for a Query command. B, D, L, N, or X status is returned only after a dialing sequence has been started or a change has occurred in the dialing status. If there is no current response for these responses, a space (Hex 20) is returned in the position for these responses. Either T0 or T1 is returned when an Initialize command has issued. All responses, except H0, H1, and S0 to SF, are reset after they are read and do not appear in response to the next Query, unless the condition has recurred. The Query response overrides any incoming data from the telephone line.
Redial (R m...m)

When the Redial command is issued, the modem executes up to 10 redials at a rate of one every 40 seconds. The redials are triggered by the detection of a busy signal after dialing.

The dial string (m...m) is the same as that described for the Dial command.

Speed n (S n)

This command sets the bit rate for the modem where n represents the line speed:

\[
\begin{array}{c|c}
  n & \text{Line speed (bits per second)} \\
  \hline
  0 & 110 \\
  1 & 300 \\
  2 & 1200 \\
\end{array}
\]

After the Speed command has changed the line speed, the application must update the divisor registers to reflect the new line speed.

The modem responds to the Speed command at the old speed rate.

If programming in BASIC, the Speed command must be used in addition to specifying the same line speed rate in the BASIC OPEN statement.

The default is 2 (1200 bps).
**Transparent n...n (T n ...)**

This command places the modem in the transparent mode for the next n bytes. The n can be up to four digits long where n is defined as any hexadecimal digit 0 - F. If n is specified as 0, then the modem remains in transparent mode until bit 2 of the modem control register is set to 1.

When the modem receives this command, it transmits the number of characters specified as data and does not interpret the data for command or control characters.

If an argument is not included with the Transparent command, the command is invalid and the modem responds with a question mark (?).

Transparent mode ends when:

- The specified number of bytes have been transmitted.
- The carrier has been lost.
- Bit 2 of the modem control register is set to 1.

The modem must exit the transparent mode before processing the next complete character from the system.

Transparent mode is restarted when bit 2 of the modem control register is reset and a new Transparent command is issued.
Voice (V)

This command forces the modem to the voice state and the modem does not monitor the communications line. This state is used for voice communications.

This command disables the auto-answer function.

The status responses when dialing in voice mode are:

If a busy signal is detected - Busy OK.

Any other condition - OK...(roughly one dot each second for 13 seconds)....Connected.

Wait (W)

This command causes the modem to take no action, including auto-answer, until the next command is received from the system. All commands following the WAIT command in a single command line are ignored.

Xmit m...m (X m...m)

This command instructs the modem to transmit the DTMF tone pairs found in the argument string m...m. This is valid only after the modem is off-hook in the voice mode. If no argument string is entered but a valid dial string was entered previously, the Xmit command uses the argument from the previous Dial command. If no parameter was ever entered, the modem responds with a question mark (?) followed by an OK.
Ztest n (Z n)

This command places the modem in the test mode specified by the argument n:

\[ n \quad Test \]

0 Hardware integrity test (on-hook only)
1 Analog loop back test (on-hook only)
2 Remote digital loopback (1200 bps only)
3 Local digital loopback (1200 bps only)

For the hardware integrity test, the test is performed, status is posted, and then the modem returns to service immediately. The integrity test takes about 2 seconds to complete, and its completion is signaled by an OK message.

For modes other than the hardware integrity test:

- The modem stays in the test mode until any other command is received.
- The modem may take up to 1 second to enter the test mode.
- The receive buffer may be loaded with an extraneous character after issuing the Ztest command.

All commands following the Ztest command in a single command line are ignored.
CRT Display Adapter

The IBM PC Convertible CRT Display Adapter is a feature that provides the interface required to attach compatible direct drive and composite monitors to the IBM PC Convertible. This feature also allows a television set to be connected to the IBM PC Convertible when a radio frequency (rf) modulator is used.

The CRT display adapter is compatible with programs that use the IBM Color/Graphics Display Adapter. The operation and register interface of the CRT display adapter are similar to operation and interface of the Motorola 6845 CRT Controller.

The adapter contains 16K bytes of display storage (refresh buffer) and supports up to 256 different character codes. The refresh buffer is located at address hex B8000.

The adapter has two basic modes of operation, alphanumeric and graphics (all points addressable). In alphanumeric mode, the adapter uses a character generator to map character information to the display. Characters are displayed in an 8-by-8 dot matrix. Alphanumeric mode supports two resolutions, low resolution (25 rows by 40 characters) and high resolution (25 rows by 80 characters).

In graphics mode, the adapter maps the information to the display on a bit-per-pel (picture element) basis. Two resolutions are available in this mode, 320 pels by 200 rows (medium resolution) and 640 pels by 200 rows (high resolution). High resolution supports only black and white images.

The adapter is active only when the IBM PC Convertible is connected to external power. When the adapter is active and the LCD is attached, the LCD is configured to emulate the IBM Personal Computer Monochrome Adapter.
If the display attached to the CRT display adapter is the active display and the system is powered off, the application that is operating in the system is not saved and cannot be resumed. The application must be reloaded after the next power on if the application is to be run.

Alphanumeric Mode

Every display-character position in alphanumeric mode is defined by 2 bytes in the refresh buffer. The following shows the format of these bytes.

<table>
<thead>
<tr>
<th>Character Code Byte (Even)</th>
<th>Attribute Byte (Odd)</th>
</tr>
</thead>
<tbody>
<tr>
<td>7 6 5 4 3 2 1 0</td>
<td>7 6 5 4 3 2 1 0</td>
</tr>
</tbody>
</table>

- **Blink**
- **Background**
- **Intensity**

The foreground and background bit setting provide the following functions:

**Background/Foreground Bits**

<table>
<thead>
<tr>
<th>Attributes</th>
</tr>
</thead>
<tbody>
<tr>
<td>Solid black (nondisplay)</td>
</tr>
<tr>
<td>Normal video</td>
</tr>
<tr>
<td>Reverse video</td>
</tr>
<tr>
<td>Solid white (nondisplay)</td>
</tr>
</tbody>
</table>

3-54 System Options
The following shows the color mapping for alphanumeric mode. Note that the I (intensity) bit provides extra luminance to each available shade for monitors that recognize the intensity attribute. This results in the light colors listed in the figure.

**Background/Foreground**

<table>
<thead>
<tr>
<th>Bits</th>
<th>Attributes</th>
</tr>
</thead>
<tbody>
<tr>
<td>1000</td>
<td>Black; grey if I = 1</td>
</tr>
<tr>
<td>1001</td>
<td>Blue; light blue if I = 1</td>
</tr>
<tr>
<td>1010</td>
<td>Green; light green if I = 1</td>
</tr>
<tr>
<td>1011</td>
<td>Cyan; light cyan if I = 1</td>
</tr>
<tr>
<td>1100</td>
<td>Red; light red if I = 1</td>
</tr>
<tr>
<td>1101</td>
<td>Magenta; light magenta if I = 1</td>
</tr>
<tr>
<td>1110</td>
<td>Brown; light brown if I = 1</td>
</tr>
<tr>
<td>1111</td>
<td>White; high intensity white if I = 1</td>
</tr>
</tbody>
</table>

A full screen of alphanumeric text requires 2000 bytes of refresh buffer for 40-by-25 mode and 4000 bytes for 80-by-25 mode. Up to eight (four if 80-by-25 mode) screens (pages) can be stored in the refresh buffer and each screen can be selectively displayed using a BIOS function call. Figure 3-9 on page 3-56 shows the mapping of the refresh buffer for one full screen of alphanumeric text in 80-by-25 mode.
<table>
<thead>
<tr>
<th>Address (hex)</th>
<th>Usage</th>
</tr>
</thead>
<tbody>
<tr>
<td>B8000</td>
<td>Code for character at upper-left corner of screen</td>
</tr>
<tr>
<td>B8001</td>
<td>Attribute for first character</td>
</tr>
<tr>
<td>B8...</td>
<td>Subsequent codes and attributes</td>
</tr>
<tr>
<td>B87CF</td>
<td>Attribute for character at bottom-right corner of screen</td>
</tr>
</tbody>
</table>

Figure 3-9. Color/Graphics Operations Refresh Buffer

3-56 System Options
Graphics Mode

In graphics (all-points-addressable) mode, the adapter directly maps the display refresh buffer to the display on a bit-per-pel basis. Applications operating in all-points-addressable mode can have a display area that is either 320 pels by 200 rows (medium resolution) or 640 pels by 200 rows (high resolution).

Two 8000-byte areas in the refresh buffer are used to map the display. Figure 3-10 shows the layout of these storage areas. The first area contains pel information for the even-numbered display row beginning with the pel information for the upper-left corner of the display. The second area contains pel information for the odd-numbered display rows.

<table>
<thead>
<tr>
<th>Address (hex)</th>
<th>Length (bytes)</th>
<th>Usage</th>
</tr>
</thead>
<tbody>
<tr>
<td>B8000</td>
<td>8000</td>
<td>Even Rows 0-198</td>
</tr>
<tr>
<td>B9F3F</td>
<td>192</td>
<td>Reserved</td>
</tr>
<tr>
<td>BA000</td>
<td>8000</td>
<td>Odd Rows 1-199</td>
</tr>
<tr>
<td>BBF3F</td>
<td>192</td>
<td>Reserved</td>
</tr>
</tbody>
</table>

Figure 3-10. Graphics Mode Refresh Buffer
For high-resolution operations, each bit in the display storage represents a pel on the display. Each physical pel is set to on or off according to the bit mapping in the display storage area. Figure 3-11 shows the bit-to-pel relationship for high resolution.

**Note:** High-resolution graphics mode supports only black and white images.

![Bit Position and Pel Position for High Resolution](image)

**Figure 3-11. High Resolution Bit-to-Pel Relationship**

For medium resolution operations, the physical pels are considered in pairs to be a logical pel. That is, the first and second physical pels on the display are considered as a logical pel. Figure 3-12 shows the bit-to-pel relationship for medium resolution. Figure 3-13 on page 3-59 shows how the bit pairs are mapped for medium resolution.

![Bit Position and Pel Position for Medium Resolution](image)

**Figure 3-12. Medium Resolution Bit-to-Pel Relationship**

**Note:** The bits are considered in pairs, such as bits 7 and 6 are pairs that map pel position 1. See Figure 3-13 on page 3-59 for how to map the image to the pairs for medium resolution.
The color sets are defined as follows:

- **Color set 1:**
  - Green (color 1)
  - Red (color 2)
  - Brown (color 3)

- **Color set 2:**
  - Cyan (color 1)
  - Magenta (color 2)
  - White (color 3)
I/O Channel Interface

The I/O channel lines to the CRT display adapter are provided on the system unit on a 72-pin connector. See "I/O Channel" on page 2-36 for a complete description of these lines.

Direct Drive Interface

Figure 3-14 on page 3-61 shows the direct drive output from the CRT display adapter. The plus (+) and minus (−) preceding the line names indicates the active state of the signals. The input/output column indicates whether the signal direction is to or from the CRT display adapter.
### Direct Drive Video Connector

<table>
<thead>
<tr>
<th>I/O Pin</th>
<th>Signal Name</th>
<th>Input/Output</th>
</tr>
</thead>
<tbody>
<tr>
<td>A1</td>
<td>No connection</td>
<td></td>
</tr>
<tr>
<td>A2</td>
<td>Ground</td>
<td></td>
</tr>
<tr>
<td>A3</td>
<td>No connection</td>
<td></td>
</tr>
<tr>
<td>A4</td>
<td>Blue</td>
<td>Output</td>
</tr>
<tr>
<td>A5</td>
<td>Red</td>
<td>Output</td>
</tr>
<tr>
<td>A6</td>
<td>Intensity</td>
<td>Output</td>
</tr>
<tr>
<td>A7</td>
<td>Green</td>
<td>Output</td>
</tr>
<tr>
<td>A8</td>
<td>No connection</td>
<td></td>
</tr>
<tr>
<td>A9</td>
<td>Reserved</td>
<td></td>
</tr>
<tr>
<td>B1</td>
<td>+ Vertical Sync</td>
<td>Output</td>
</tr>
<tr>
<td>B2</td>
<td>No connection</td>
<td></td>
</tr>
<tr>
<td>B3</td>
<td>+ Horizontal Sync</td>
<td>Output</td>
</tr>
<tr>
<td>B4</td>
<td>No connection</td>
<td></td>
</tr>
<tr>
<td>B5</td>
<td>Ground</td>
<td></td>
</tr>
<tr>
<td>B6</td>
<td>No connection</td>
<td></td>
</tr>
<tr>
<td>B7</td>
<td>No connection</td>
<td></td>
</tr>
<tr>
<td>B8</td>
<td>No connection</td>
<td></td>
</tr>
<tr>
<td>B9</td>
<td>No connection</td>
<td></td>
</tr>
</tbody>
</table>

Figure 3-14. Direct Drive Video Connector

### Composite Video Interface

Figure 3-15 shows the direct drive output from the CRT display adapter. The connector is shown in page 3-60.

<table>
<thead>
<tr>
<th>I/O Pin</th>
<th>Signal Name</th>
<th>Input/Output</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>Composite Video</td>
<td>Output</td>
</tr>
<tr>
<td>2</td>
<td>Ground</td>
<td>Ground</td>
</tr>
</tbody>
</table>

Figure 3-15. Composite Video Connector
RF Modulator Interface

Figure 3-16 shows the rf modulator output from the CRT display adapter. The connector is shown in page 3-60.

<table>
<thead>
<tr>
<th>I/O Pin</th>
<th>Signal Name</th>
<th>Input/Output</th>
</tr>
</thead>
<tbody>
<tr>
<td>A1</td>
<td>+12 V dc</td>
<td>Power</td>
</tr>
<tr>
<td>A3</td>
<td>Composite Video</td>
<td>Output</td>
</tr>
<tr>
<td>B1</td>
<td>Ground</td>
<td>Ground</td>
</tr>
<tr>
<td>B2</td>
<td>Reserved</td>
<td></td>
</tr>
<tr>
<td>B3</td>
<td>No connection</td>
<td></td>
</tr>
</tbody>
</table>

Figure 3-16. RF Modulator Connector
Programming Considerations

Applications that directly access the refresh buffer do not need to disable video while doing so; the adapter resolves any memory contention without affecting the display.

Figure 3-17 shows the adapter register assignments.

<table>
<thead>
<tr>
<th>Address (hex)</th>
<th>R/W</th>
<th>Usage</th>
</tr>
</thead>
<tbody>
<tr>
<td>3D0, 3D4</td>
<td>W</td>
<td>Index registers</td>
</tr>
<tr>
<td>3D1, 3D5</td>
<td>R/W</td>
<td>Data register</td>
</tr>
<tr>
<td>3D8</td>
<td>W</td>
<td>Mode control register</td>
</tr>
<tr>
<td>3D9</td>
<td>W</td>
<td>Color Select</td>
</tr>
<tr>
<td>3DA</td>
<td>R</td>
<td>Status register</td>
</tr>
</tbody>
</table>

Figure 3-17. CRT Controller Registers
Index and Data Registers (Hex 3D0, 3D1, 3D4 and 3D5)

The index and data registers are used to access the CRT display adapter internal registers. The index register is loaded with the index number of the register to be accessed through the data register. The data register is loaded with the data to be placed into the selected register. Figure 3-18 shows the internal registers, the index numbers, and their initial settings in hexadecimal. The registers are write-only except for R14 and R15, which are read/write.

Note: The power-on routines initialize these registers. BIOS function calls should be used to change the values in these registers in order to preserve application program compatibility.

<table>
<thead>
<tr>
<th>Index No.</th>
<th>Usage</th>
<th>40 by 25</th>
<th>80 by 25</th>
<th>APA*</th>
</tr>
</thead>
<tbody>
<tr>
<td>00</td>
<td>Total number of characters displayed horizontally</td>
<td>38</td>
<td>71</td>
<td>38</td>
</tr>
<tr>
<td>01</td>
<td>R1, number of characters displayed horizontally</td>
<td>28</td>
<td>50</td>
<td>28</td>
</tr>
<tr>
<td>02</td>
<td>R2, horizontal sync character position</td>
<td>2D</td>
<td>5A</td>
<td>2D</td>
</tr>
<tr>
<td>03</td>
<td>R3, horizontal synch character width</td>
<td>0A</td>
<td>0A</td>
<td>0A</td>
</tr>
</tbody>
</table>

Figure 3-18 (Part 1 of 2). Index and Data Registers

*APA = graphics mode.

3-64 System Options
<table>
<thead>
<tr>
<th>Index No.</th>
<th>Usage</th>
<th>40 by 25</th>
<th>80 by 25</th>
<th>APA*</th>
</tr>
</thead>
<tbody>
<tr>
<td>04</td>
<td>R4, total number of vertical character rows</td>
<td>1F</td>
<td>1F</td>
<td>7F</td>
</tr>
<tr>
<td>05</td>
<td>R5, total number of scan lines adjustment</td>
<td>06</td>
<td>06</td>
<td>06</td>
</tr>
<tr>
<td>06</td>
<td>R6, number of characters displayed vertically</td>
<td>19</td>
<td>19</td>
<td>64</td>
</tr>
<tr>
<td>07</td>
<td>R7, vertical sync character row position</td>
<td>1C</td>
<td>1C</td>
<td>70</td>
</tr>
<tr>
<td>08</td>
<td>R8, interlace mode</td>
<td>02</td>
<td>02</td>
<td>02</td>
</tr>
<tr>
<td>09</td>
<td>R9, maximum scan line address</td>
<td>07</td>
<td>07</td>
<td>01</td>
</tr>
<tr>
<td>0A</td>
<td>R10, cursor start scan line address</td>
<td>06</td>
<td>06</td>
<td>06</td>
</tr>
<tr>
<td>0B</td>
<td>R11, cursor end scan line address</td>
<td>07</td>
<td>07</td>
<td>07</td>
</tr>
<tr>
<td>0C</td>
<td>R12, start address, high</td>
<td>00</td>
<td>00</td>
<td>00</td>
</tr>
<tr>
<td>0D</td>
<td>R13, start address, low</td>
<td>00</td>
<td>00</td>
<td>00</td>
</tr>
<tr>
<td>0E</td>
<td>R14, cursor address, high</td>
<td>xx</td>
<td>xx</td>
<td>xx</td>
</tr>
<tr>
<td>0F</td>
<td>R15, cursor address, low</td>
<td>xx</td>
<td>xx</td>
<td>xx</td>
</tr>
<tr>
<td>10</td>
<td>Reserved</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>11</td>
<td>Reserved</td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

Figure 3-18 (Part 2 of 2). Index and Data Registers
Mode Control Register (Hex 3D8)

Bit Meaning

7-6  Reserved

5  Change background intensity to blink

4  High resolution (640 by 200) black and white mode

3  Video enable

2  Black and white select

  0 = Color
  1 = Black and white

1  Graphics select:

  0 = Alphanumeric mode
  1 = Graphics (320 by 200)

0  Alphanumeric mode:

  0 = 40 by 25
  1 = 80 by 25
Color Select Register (Hex 3D9)

**Bit Meaning**

7-6 Reserved

5 Selects active color set for 320 by 200 graphics mode
   - 0 = Color set 1
   - 1 = Color set 2
   The color sets are described on page 3-59.

4 Selects alternate, intensified set of colors in graphics mode

3 Selects intensified for:
   - Border in 40 by 25 alphanumeric mode
   - Background in 320 by 200 graphics mode
   - Foreground in 640 by 200 graphics mode

2 Selects red for:
   - Border in 40 by 25 alphanumeric mode
   - Background in 320 by 200 graphics mode
   - Foreground in 640 by 200 graphics mode

1 Selects green for:
   - Border in 40 by 25 alphanumeric mode
   - Background in 320 by 200 graphics mode
   - Foreground in 640 by 200 graphics mode

0 Selects blue for:
   - Border in 40 by 25 alphanumeric mode
   - Background in 320 by 200 graphics mode
   - Foreground in 640 by 200 graphics mode
Status Register (Hex 3DA)

<table>
<thead>
<tr>
<th>Bit</th>
<th>Meaning</th>
</tr>
</thead>
<tbody>
<tr>
<td>7-4</td>
<td>Reserved</td>
</tr>
<tr>
<td>3</td>
<td>Vertical sync</td>
</tr>
<tr>
<td>2-1</td>
<td>Reserved</td>
</tr>
<tr>
<td>0</td>
<td>Display enable</td>
</tr>
</tbody>
</table>

Portable Printer

The IBM PC Convertible Portable Printer is a low-power serial dot matrix printer that attaches to the back of the system unit. An optional cable is available that allows the printer to be used near the system unit. Printing speed (in 10 pitch) is approximately 40 characters per second (cps) burst. In addition to the standard ASCII character set, the printer can print bit-image graphics.

The portable printer has a buffer that can store up to 2000 bytes of character and bit-image graphics data.

Programming access to the printer is through the interface described in “Portable Printer Interface” on page 2-84. The command set is described later in this section.

Refer to Appendix B, “Unit Specifications” for specifications concerning the IBM PC Convertible Portable Printer.
Following a power on or a reset, the printer is set to the following defaults:

- Horizontal tab stops: Set every eight columns starting in the ninth column
- Vertical tab stops: None set
- Line spacing: Six lines per inch
- Character pitch: 10 characters per inch
- Condensed print: Off
- Double-width print: Off
- Emphasized print: Off
- Subscript/superscript: Off
- Underline: Off
- Honor paper-end switch: On
- Auto line feed: Off
- Page length: 11 inches
- Skip paper perforation: Off
- Top of form: The current location of the paper.
The portable printer is capable of several print modes that provide a variety of printing styles:

- Standard
- Condensed
- Double-wide
- Emphasized
- Underline
- Superscript
- Subscript.

These print modes may be combined to provide other printing styles. The following combinations, however, produce these results:

<table>
<thead>
<tr>
<th>Combination</th>
<th>Result</th>
</tr>
</thead>
<tbody>
<tr>
<td>Condensed with Emphasized</td>
<td>Emphasized</td>
</tr>
<tr>
<td>Superscript with Subscript</td>
<td>Last mode selected</td>
</tr>
</tbody>
</table>

All other combinations are valid.

**Portable Printer Character Set**

Figure 3-19 shows the character set used on the IBM PC Convertible Portable Printer. The figure also shows the hexadecimal and ASCII values assigned to the character set.
<table>
<thead>
<tr>
<th>Hex</th>
<th>0x</th>
<th>1x</th>
<th>2x</th>
<th>3x</th>
<th>4x</th>
<th>5x</th>
<th>6x</th>
<th>7x</th>
</tr>
</thead>
<tbody>
<tr>
<td>x0</td>
<td>NUL</td>
<td></td>
<td></td>
<td></td>
<td>@</td>
<td>P</td>
<td>’</td>
<td>p</td>
</tr>
<tr>
<td>x1</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>!</td>
<td>A</td>
<td>Q</td>
<td>a</td>
</tr>
<tr>
<td>x2</td>
<td>DC2</td>
<td>&quot;</td>
<td>2</td>
<td>B</td>
<td>R</td>
<td>b</td>
<td>r</td>
<td></td>
</tr>
<tr>
<td>x3</td>
<td>3</td>
<td></td>
<td>3</td>
<td>C</td>
<td>S</td>
<td>c</td>
<td>s</td>
<td></td>
</tr>
<tr>
<td>x4</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>4</td>
<td>D</td>
<td>T</td>
<td>d</td>
</tr>
<tr>
<td>x5</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>5</td>
<td>E</td>
<td>U</td>
<td>e</td>
</tr>
<tr>
<td>x6</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>6</td>
<td>F</td>
<td>V</td>
<td>f</td>
</tr>
<tr>
<td>x7</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>7</td>
<td>G</td>
<td>W</td>
<td>g</td>
</tr>
<tr>
<td>x8</td>
<td>CAN</td>
<td>(8</td>
<td>H</td>
<td>X</td>
<td>h</td>
<td>x</td>
<td></td>
<td></td>
</tr>
<tr>
<td>x9</td>
<td>HT</td>
<td>9</td>
<td>l</td>
<td>Y</td>
<td>i</td>
<td>y</td>
<td></td>
<td></td>
</tr>
<tr>
<td>xA</td>
<td>LF</td>
<td>*</td>
<td>:</td>
<td>J</td>
<td>Z</td>
<td>j</td>
<td>z</td>
<td></td>
</tr>
<tr>
<td>xB</td>
<td>VT</td>
<td>+</td>
<td>;</td>
<td>K</td>
<td>[</td>
<td>k</td>
<td>{</td>
<td></td>
</tr>
<tr>
<td>xC</td>
<td>FF</td>
<td>&lt;</td>
<td>L</td>
<td>l</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>xD</td>
<td>CR</td>
<td>-</td>
<td>=</td>
<td>M</td>
<td>]</td>
<td>m</td>
<td>}</td>
<td></td>
</tr>
<tr>
<td>xE</td>
<td>SO</td>
<td>&gt;</td>
<td>N</td>
<td>^</td>
<td>n</td>
<td>~</td>
<td></td>
<td></td>
</tr>
<tr>
<td>xF</td>
<td>SI</td>
<td>/</td>
<td>?</td>
<td>O</td>
<td>–</td>
<td>0</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

Figure 3-19 (Part 1 of 2). Portable Printer Character Set
<table>
<thead>
<tr>
<th>Hex</th>
<th>8x</th>
<th>9x</th>
<th>Ax</th>
<th>Bx</th>
<th>Cx</th>
<th>Dx</th>
<th>Ex</th>
<th>Fx</th>
</tr>
</thead>
<tbody>
<tr>
<td>x0</td>
<td>ç</td>
<td>é</td>
<td>á</td>
<td>l</td>
<td>l</td>
<td>α</td>
<td>γ</td>
<td>224</td>
</tr>
<tr>
<td>x1</td>
<td>ü</td>
<td>æ</td>
<td>i</td>
<td>β</td>
<td>±</td>
<td>225</td>
<td>241</td>
<td></td>
</tr>
<tr>
<td>x2</td>
<td>é</td>
<td>Æ</td>
<td>é</td>
<td>Γ</td>
<td>≥</td>
<td>226</td>
<td>242</td>
<td></td>
</tr>
<tr>
<td>x3</td>
<td>à</td>
<td>ò</td>
<td>ū</td>
<td>π</td>
<td>≤</td>
<td>227</td>
<td>243</td>
<td></td>
</tr>
<tr>
<td>x4</td>
<td>à</td>
<td>ö</td>
<td>ŋ</td>
<td>Σ</td>
<td>r</td>
<td>228</td>
<td>244</td>
<td></td>
</tr>
<tr>
<td>x5</td>
<td>à</td>
<td>ò</td>
<td>ŋ</td>
<td>σ</td>
<td>j</td>
<td>229</td>
<td>245</td>
<td></td>
</tr>
<tr>
<td>x6</td>
<td>à</td>
<td>ò</td>
<td>ŋ</td>
<td>μ</td>
<td>÷</td>
<td>230</td>
<td>246</td>
<td></td>
</tr>
<tr>
<td>x7</td>
<td>ç</td>
<td>ù</td>
<td>o</td>
<td>τ</td>
<td>z</td>
<td>231</td>
<td>247</td>
<td></td>
</tr>
<tr>
<td>x8</td>
<td>ë</td>
<td>ŋ</td>
<td>ŋ</td>
<td>Φ</td>
<td>ο</td>
<td>232</td>
<td>248</td>
<td></td>
</tr>
<tr>
<td>x9</td>
<td>ë</td>
<td>ï</td>
<td>ι</td>
<td>Θ</td>
<td>θ</td>
<td>233</td>
<td>249</td>
<td></td>
</tr>
<tr>
<td>xA</td>
<td>ë</td>
<td>Ü</td>
<td>ũ</td>
<td>Ω</td>
<td>Ω</td>
<td>234</td>
<td>250</td>
<td></td>
</tr>
<tr>
<td>xB</td>
<td>ï</td>
<td>ř</td>
<td>ř</td>
<td>δ</td>
<td>√</td>
<td>235</td>
<td>251</td>
<td></td>
</tr>
<tr>
<td>xC</td>
<td>1</td>
<td>ř</td>
<td>ř</td>
<td>∞</td>
<td>n</td>
<td>236</td>
<td>252</td>
<td></td>
</tr>
<tr>
<td>xD</td>
<td>1</td>
<td>ř</td>
<td>ř</td>
<td>φ</td>
<td>2</td>
<td>237</td>
<td>253</td>
<td></td>
</tr>
<tr>
<td>xE</td>
<td>1</td>
<td>ř</td>
<td>ř</td>
<td>∈</td>
<td>∈</td>
<td>238</td>
<td>254</td>
<td></td>
</tr>
<tr>
<td>xF</td>
<td>1</td>
<td>ř</td>
<td>ř</td>
<td>Ë</td>
<td>Ë</td>
<td>239</td>
<td>255</td>
<td></td>
</tr>
</tbody>
</table>

Figure 3-19 (Part 2 of 2). Portable Printer Character Set

3-72 System Options
Portable Printer Commands

The printer commands consist of control characters that are sent to the printer as single characters or character sequences. The commands can be entered from the keyboard or through programming as described in Guide to Operations for the IBM PC Convertible. Following is a description of the printer control characters. The printer code and the ASCII format of the commands are also shown.

Null

<table>
<thead>
<tr>
<th>Code</th>
<th>Command Stream Using ASCII Values</th>
</tr>
</thead>
<tbody>
<tr>
<td>NUL</td>
<td>0</td>
</tr>
</tbody>
</table>

This control character is used with other commands as a list terminator. NUL is also used with other printer control codes to select options.

Horizontal Tab

<table>
<thead>
<tr>
<th>Code</th>
<th>Command Stream Using ASCII Values</th>
</tr>
</thead>
<tbody>
<tr>
<td>HT</td>
<td>9</td>
</tr>
</tbody>
</table>

This command causes the print head to move to the next horizontal tab stop. If there are no tab stops between the position of the print head and the right margin, the command is ignored.
**Line Feed**

<table>
<thead>
<tr>
<th>Code</th>
<th>Command Stream Using ASCII Values</th>
</tr>
</thead>
<tbody>
<tr>
<td>LF</td>
<td>10</td>
</tr>
</tbody>
</table>

This command moves the print head to the first printing position on the next line. Line spacing is 1/6-inch (4.23 millimeter) unless reset by Set 1/8 Inch Line Spacing (ESC 0), Set 7/60 Inch Line Spacing (ESC 1), Start Text Line Spacing (ESC 2), Set Graphics Line Spacing (ESC 3), or Store Text Line Spacing (ESC A).

**Vertical Tab**

<table>
<thead>
<tr>
<th>Code</th>
<th>Command Stream Using ASCII Values</th>
</tr>
</thead>
<tbody>
<tr>
<td>VT</td>
<td>11</td>
</tr>
</tbody>
</table>

This command moves the paper to the next vertical tab stop. If no vertical tabs have been set, this command is treated as a line feed.

**Form Feed**

<table>
<thead>
<tr>
<th>Code</th>
<th>Command Stream Using ASCII Values</th>
</tr>
</thead>
<tbody>
<tr>
<td>FF</td>
<td>12</td>
</tr>
</tbody>
</table>

This command causes a carriage return and advances the paper to the top of the next page. Multiple FF commands must be separated by a space. The top of the page can be reset by Set Top of Page (ESC 4) or Set Page Length (ESC C).

3-74 System Options
Carriage Return

<table>
<thead>
<tr>
<th>Code</th>
<th>Command Stream Using ASCII Values</th>
</tr>
</thead>
<tbody>
<tr>
<td>CR</td>
<td>13</td>
</tr>
</tbody>
</table>

This command positions the print head to the first print position on a line and causes a line feed if auto line feed mode (ESC 5) has been set.

Double-Width Print, Line Mode

<table>
<thead>
<tr>
<th>Code</th>
<th>Command Stream Using ASCII Values</th>
</tr>
</thead>
<tbody>
<tr>
<td>SO</td>
<td>14</td>
</tr>
</tbody>
</table>

This command sets double-width print mode causing the subsequent graphic characters remaining on the line to be printed double-wide. Double-width print mode is canceled by Carriage Return (CR), Vertical Tab (VT), Line Feed (LF), Forms Feed (FF), and End Double-width Print (DC 4).

Condensed Print

<table>
<thead>
<tr>
<th>Code</th>
<th>Command Stream Using ASCII Values</th>
</tr>
</thead>
<tbody>
<tr>
<td>SI</td>
<td>15</td>
</tr>
</tbody>
</table>

This command sets condensed print mode. In condensed print mode, characters are printed in 16.4 pitch. If condensed and double-width print modes are combined, characters are printed in 8.2 pitch. Condensed print mode does not work with emphasized print; if Emphasized print and Condensed print commands are both active, emphasized print mode results.
End Condensed Print (DC 2)

<table>
<thead>
<tr>
<th>Code</th>
<th>Command Stream Using ASCII Values</th>
</tr>
</thead>
<tbody>
<tr>
<td>DC 2</td>
<td>18</td>
</tr>
</tbody>
</table>

This command resets condensed print mode.

End Double-Width Print, Line Mode

<table>
<thead>
<tr>
<th>Code</th>
<th>Command Stream Using ASCII Values</th>
</tr>
</thead>
<tbody>
<tr>
<td>DC 4</td>
<td>20</td>
</tr>
</tbody>
</table>

This command resets double-width print, line mode. This command does not reset double-width print, continuous mode.

Cancel

<table>
<thead>
<tr>
<th>Code</th>
<th>Command Stream Using ASCII Values</th>
</tr>
</thead>
<tbody>
<tr>
<td>CAN</td>
<td>24</td>
</tr>
</tbody>
</table>

This command clears the last line in the print buffer without printing the data. The print head is not moved.
Escape

<table>
<thead>
<tr>
<th>Code</th>
<th>Command Stream Using ASCII Values</th>
</tr>
</thead>
<tbody>
<tr>
<td>ESC</td>
<td>27</td>
</tr>
</tbody>
</table>

This command is used as the prefix to many printer commands.

End Auto Underline

<table>
<thead>
<tr>
<th>Code</th>
<th>Command Stream Using ASCII Values</th>
</tr>
</thead>
<tbody>
<tr>
<td>ESC - 0</td>
<td>27 45 48</td>
</tr>
</tbody>
</table>

This command reset auto underline mode.

Auto Underline

<table>
<thead>
<tr>
<th>Code</th>
<th>Command Stream Using ASCII Values</th>
</tr>
</thead>
<tbody>
<tr>
<td>ESC - 1</td>
<td>27 45 49</td>
</tr>
</tbody>
</table>

This command sets auto underline mode, causing graphic characters and spaces (hex 20 and FF) to be underlined. Spaces caused by Horizontal Tab are not underlined. Characters used for making boxes, such as the characters with ASCII value 179 through 223, are not underlined.
Set 1/8 Inch Line Spacing

<table>
<thead>
<tr>
<th>Code</th>
<th>Command Stream Using ASCII Values</th>
</tr>
</thead>
<tbody>
<tr>
<td>ESC 0</td>
<td>27 48</td>
</tr>
</tbody>
</table>

This command sets line spacing to 8 lines per inch. Some characters on adjacent lines may overlap.

Set 7/60 Inch Line Spacing

<table>
<thead>
<tr>
<th>Code</th>
<th>Command Stream Using ASCII Values</th>
</tr>
</thead>
<tbody>
<tr>
<td>ESC 1</td>
<td>27 49</td>
</tr>
</tbody>
</table>

This command sets line spacing to 8.57 lines per inch. This setting is used with some graphics applications.

Start Text Line Spacing

<table>
<thead>
<tr>
<th>Code</th>
<th>Command Stream Using ASCII Values</th>
</tr>
</thead>
<tbody>
<tr>
<td>ESC 2</td>
<td>27 50</td>
</tr>
</tbody>
</table>

This command sets the line spacing to the value specified in the last Store Text Line Spacing (ESC A). If a Store Text Line Spacing command has not been entered, line spacing is set to 6 lines per inch.

3-78 System Options
Set Graphics Line Spacing

<table>
<thead>
<tr>
<th>Code</th>
<th>Command Stream Using ASCII Values</th>
</tr>
</thead>
<tbody>
<tr>
<td>ESC 3 n</td>
<td>27 51 n</td>
</tr>
</tbody>
</table>

This command sets line spacing to \( \frac{n}{180} \) inch, where \( n \) is any value from 1 through 255. For bit-image graphics printing using 8 bits, \( n \) is set to 24.

Set Top of Page

<table>
<thead>
<tr>
<th>Code</th>
<th>Command Stream Using ASCII Values</th>
</tr>
</thead>
<tbody>
<tr>
<td>ESC 4</td>
<td>27 52</td>
</tr>
</tbody>
</table>

This command sets the current print line as the top of the paper. Set Page Length in Inches also sets the top of page.

Set Auto Line Feed On

<table>
<thead>
<tr>
<th>Code</th>
<th>Command Stream Using ASCII Values</th>
</tr>
</thead>
<tbody>
<tr>
<td>ESC 5 1</td>
<td>27 53 49</td>
</tr>
</tbody>
</table>

This command causes a line feed to automatically occur after each Carriage Return (CR).
End Auto Line Feed

<table>
<thead>
<tr>
<th>Code</th>
<th>Command Stream Using ASCII Values</th>
</tr>
</thead>
<tbody>
<tr>
<td>ESC 5 0</td>
<td>27 53 48</td>
</tr>
</tbody>
</table>

This command resets the automatic line feed to off.

Ignore Paper-End Switch

<table>
<thead>
<tr>
<th>Code</th>
<th>Command Stream Using ASCII Values</th>
</tr>
</thead>
<tbody>
<tr>
<td>ESC 8</td>
<td>27 56</td>
</tr>
</tbody>
</table>

This command allows the printer to print to the bottom of the paper. Print alignment may be lost within 1-1/2 inches from the bottom of the paper.

Honor Paper-End Switch

<table>
<thead>
<tr>
<th>Code</th>
<th>Command Stream Using ASCII Values</th>
</tr>
</thead>
<tbody>
<tr>
<td>ESC 9</td>
<td>27 57</td>
</tr>
</tbody>
</table>

This command causes the printer to stop printing 1-1/2 inches from the bottom of the page.
Store Text Line Spacing

<table>
<thead>
<tr>
<th>Code</th>
<th>Command Stream Using ASCII Values</th>
</tr>
</thead>
<tbody>
<tr>
<td>ESC A n</td>
<td>27 65 n</td>
</tr>
</tbody>
</table>

This command sets line spacing to \( \frac{n}{60} \) inch, where \( n \) is any value from 1 through 85. For 6-lines-per-inch spacing, \( n \) is set to 10. A Start Text Line Spacing (ESC 2) command must be used to start the line spacing.

Set Vertical Tabs

<table>
<thead>
<tr>
<th>Code</th>
<th>Command Stream Using ASCII Values</th>
</tr>
</thead>
<tbody>
<tr>
<td>ESC B n 0</td>
<td>27 66 n...n 0</td>
</tr>
</tbody>
</table>

This command clears the current vertical tab stop settings and sets up to 64 new tab stop settings, where \( n...n \) specifies the new tab stop settings in ascending numeric order. Tab stop values are specified in character widths of the character pitch that is currently in effect. Tab stop values can be set to values 1 through 127, but must not exceed page length.

If more than 64 settings or if values greater than the page length are specified, the remaining values are ignored.
Clear Vertical Tabs

<table>
<thead>
<tr>
<th>Code</th>
<th>Command Stream Using ASCII Values</th>
</tr>
</thead>
<tbody>
<tr>
<td>ESC B 0</td>
<td>27 66 0</td>
</tr>
</tbody>
</table>

This command clears the vertical tab settings.

Set Page Length in Lines

<table>
<thead>
<tr>
<th>Code</th>
<th>Command Stream Using ASCII Values</th>
</tr>
</thead>
<tbody>
<tr>
<td>ESC C n</td>
<td>27 67 n</td>
</tr>
</tbody>
</table>

This command sets the length to number of lines, where n can be set from 1 through 127 lines. The command must be issued at a page boundary. This command also sets Top of Page and cancels Auto Perforation Skip. The page length is converted to inches and is not affected by subsequent line spacing changes.

Set Page Length in Inches

<table>
<thead>
<tr>
<th>Code</th>
<th>Command Stream Using ASCII Values</th>
</tr>
</thead>
<tbody>
<tr>
<td>ESC C 0 n</td>
<td>27 67 0 n</td>
</tr>
</tbody>
</table>

This command sets the length of the page in inches, where n can be set from 1 through 22 inches. If n is set to 0, the command is ignored. The command must be issued at a page boundary. This command also sets Top of Page and cancels Auto Perforation Skip.

3-82 System Options
Set Horizontal Tabs

<table>
<thead>
<tr>
<th>Code</th>
<th>Command Stream Using ASCII Values</th>
</tr>
</thead>
<tbody>
<tr>
<td>ESC D n 0</td>
<td>27 68 n...n 0</td>
</tr>
</tbody>
</table>

This command clears the current horizontal tab stop settings and sets up to 28 new tab stop settings, where n...n specifies the new tab stop settings in ascending numeric order. Tab stop values are specified in character widths of the character pitch that is currently in effect. Double-width mode is ignored.

If more than 28 settings or if values greater than the line length are specified, the remaining values are ignored and may print as data on the output.

Clear Horizontal Tabs

<table>
<thead>
<tr>
<th>Code</th>
<th>Command Stream Using ASCII Values</th>
</tr>
</thead>
<tbody>
<tr>
<td>ESC D 0</td>
<td>27 68 0</td>
</tr>
</tbody>
</table>

This command clears the horizontal tab settings.
Emphasized Print

<table>
<thead>
<tr>
<th>Code</th>
<th>Command Stream Using ASCII Values</th>
</tr>
</thead>
<tbody>
<tr>
<td>ESC E</td>
<td>27 69</td>
</tr>
</tbody>
</table>

This command sets emphasized print mode. If condensed print mode is active when this command is issued, printing from this point is done with standard-width characters in emphasized print mode.

End Emphasized Print

<table>
<thead>
<tr>
<th>Code</th>
<th>Command Stream Using ASCII Values</th>
</tr>
</thead>
<tbody>
<tr>
<td>ESC F</td>
<td>27 70</td>
</tr>
</tbody>
</table>

This command resets emphasized print mode.
480 Bit-Image Graphics Mode

<table>
<thead>
<tr>
<th>Code</th>
<th>Command Stream Using ASCII Values</th>
</tr>
</thead>
<tbody>
<tr>
<td>ESC K ...</td>
<td>27 75 n₁ n₂ v₁ v₂ ... vₖ</td>
</tr>
</tbody>
</table>

This command changes from text mode to bit-image graphics mode. The symbols n₁ and n₂ are 1-byte values that together specify the total number of bit-image data bytes to be transferred:

- n₂ is a weighting factor used to indicate the number of whole 256-byte blocks of bit-image data. It is determined by dividing the total number of data bytes by 256 (n₁ = k ÷ 256).

- n₂ is set to the remainder after calculating the value of n₂.

For example, if 20 bit-image data bytes are to be transferred, n₁ contains hex 14 (20) and n₂ contains hex 00. If 300 bit-image data bytes are to be transferred, n₁ contains hex 2C (44) and n₂ contains hex 01. If both n₁ and n₂ are set to 0, the command is ignored.

The symbols v₁ through vₖ are the bytes of the bit-image data. Each byte represents a print area that is 24 dots vertically and 3 dots horizontally. Each bit represents a three-by-three dot matrix in the print area; setting a bit to 1 causes all nine dots in the matrix to print. Bit 7 represents the top matrix and bit 0 represents the bottom matrix. Setting all bits to 1 causes a vertical bar (24-by-3 dots) to be printed.

The total number of data bytes (k) cannot exceed 480 and must be equal to n₁ + (256 × n₂).
960 Bit-Image Graphics Mode

<table>
<thead>
<tr>
<th>Code</th>
<th>Command Stream Using ASCII Values</th>
</tr>
</thead>
<tbody>
<tr>
<td>ESC L ...</td>
<td>27 76 n₁ n₂ v₁ v₂ ... vₖ</td>
</tr>
</tbody>
</table>

This command sets 960 bit-image graphics mode. This command operates the same as the Set 480 Bit-Image Graphics Mode command, except that the number of bit-image data bytes (k) cannot exceed 960.

Set Automatic Perforation Skip

<table>
<thead>
<tr>
<th>Code</th>
<th>Command Stream Using ASCII Values</th>
</tr>
</thead>
<tbody>
<tr>
<td>ESC N n</td>
<td>27 78 n</td>
</tr>
</tbody>
</table>

This command sets the number of lines to skip at the bottom of a page, where n can be set from 2 through 126 lines or the length of the page. This command is canceled by Set Page Length (ESC C) and Cancel Automatic Perforation Skip (ESC O) commands.

Cancel Automatic Perforation Skip

<table>
<thead>
<tr>
<th>Code</th>
<th>Command Stream Using ASCII Values</th>
</tr>
</thead>
<tbody>
<tr>
<td>ESC O</td>
<td>27 79</td>
</tr>
</tbody>
</table>

This command cancels the perforation skip at the bottom of a page.
Set Default Tab Rack

<table>
<thead>
<tr>
<th>Code</th>
<th>Command Stream Using ASCII Values</th>
</tr>
</thead>
<tbody>
<tr>
<td>ESC R</td>
<td>27 82</td>
</tr>
</tbody>
</table>

This command clears the current tab stops (vertical and horizontal) and sets the horizontal tab rack for stops at every eight positions, starting with the ninth column.

Superscript Mode

<table>
<thead>
<tr>
<th>Code</th>
<th>Command Stream Using ASCII Values</th>
</tr>
</thead>
<tbody>
<tr>
<td>ESC S 0</td>
<td>27 83 48</td>
</tr>
</tbody>
</table>

This command sets superscript mode. This mode remains active until reset by an End Subscript/Superscript Mode (ESC T) or Subscript Mode (ESC S) command.

Subscript Mode

<table>
<thead>
<tr>
<th>Code</th>
<th>Command Stream Using ASCII Values</th>
</tr>
</thead>
<tbody>
<tr>
<td>ESC S 1</td>
<td>27 83 49</td>
</tr>
</tbody>
</table>

This command sets subscript mode. This mode remains active until reset by an End Subscript/Superscript Mode (ESC T) or Superscript Mode (ESC S) command.
End Superscript/Superscript Mode

<table>
<thead>
<tr>
<th>Code</th>
<th>Command Stream Using ASCII Values</th>
</tr>
</thead>
<tbody>
<tr>
<td>ESC T</td>
<td>27 84</td>
</tr>
</tbody>
</table>

This command cancels subscript and superscript mode.

End Double-Width Print, Continuous

<table>
<thead>
<tr>
<th>Code</th>
<th>Command Stream Using ASCII Values</th>
</tr>
</thead>
<tbody>
<tr>
<td>ESC W 0</td>
<td>27 87 48</td>
</tr>
</tbody>
</table>

This command cancels continuous double-width print mode.

Double-Width Print, Continuous

<table>
<thead>
<tr>
<th>Code</th>
<th>Command Stream Using ASCII Values</th>
</tr>
</thead>
<tbody>
<tr>
<td>ESC W 1</td>
<td>27 87 49</td>
</tr>
</tbody>
</table>

This command sets double-width print mode causing the subsequent graphic characters line to be printed double-wide. This mode remains active until reset by an End Double-Width Print Continuous (ESC W) command.
960 Bit-Image Graphics Mode

<table>
<thead>
<tr>
<th>Code</th>
<th>Command Stream Using ASCII Values</th>
</tr>
</thead>
<tbody>
<tr>
<td>ESC Y</td>
<td>27 89 n₁ n₂ v₁ v₂ .... vₖ</td>
</tr>
</tbody>
</table>

This command sets 960 bit-image graphics mode. This command operates the same as the Set 480 Bit-Image Graphics Mode command, except that the number of bit-image data bytes cannot exceed 960.

1920 Bit-Image Graphics Mode

<table>
<thead>
<tr>
<th>Code</th>
<th>Command Stream Using ASCII Values</th>
</tr>
</thead>
<tbody>
<tr>
<td>ESC Z</td>
<td>27 90 n₁ n₂ v₁ v₂ .... vₖ</td>
</tr>
</tbody>
</table>

This command sets 1920 bit-image graphics mode. This command operates the same as the Set 480 Bit-Image Graphics Mode command, except that the number of bit-image data bytes cannot exceed 1920.

Monochrome Display

The IBM PC Convertible monochrome display is a 9-inch (measured diagonally) composite video monitor, which is attached to the IBM PC Convertible CRT display adapter. The monitor operates on ac only.

Refer to Appendix B, “Unit Specifications” for specifications concerning the IBM PC Convertible Monochrome Display.
Color Display

The IBM PC Convertible Color Display is a 13-inch (measured diagonally) color monitor, which is attached to the IBM PC Convertible CRT Display Adapter. The monitor operates on ac only.

Refer to Appendix B, "Unit Specifications" for specifications concerning the IBM PC Convertible Color Display.

Automobile Power Adapter

The IBM PC Convertible Automobile Power Adapter is used to supply power to the system unit and to recharge the system battery by using a dc outlet in an automobile.

If the automobile motor is not running, the system may not receive a full charge.

Battery Charger

The IBM PC Convertible Battery Charger is used to recharge the battery in the system unit by using an ac wall outlet.
Section 4. System BIOS and Usage

ROM BIOS ........................................ 4-3
Use of BIOS ..................................... 4-3
  Stack Requirements .......................... 4-4
  BIOS and Hardware Interrupt Vectors ......... 4-4
Reserved Interrupt Allocations .................. 4-21
Other Reserved Areas ........................... 4-23
Adapters with System-Accessible ROM .......... 4-24
BIOS Programming Guidelines ................... 4-26
Keyboard Encoding and Usage .................... 4-27
  Encoding .................................... 4-27
  Character Codes ............................. 4-27
  Extended-ASCII Codes ...................... 4-34
  Special Handling ............................ 4-35
Special BIOS Functions ........................ 4-37
Notes:
The basic input/output system (BIOS) resides in read-only memory (ROM) on the system board and provides the device level control for the input and output (I/O) devices in the system. The BIOS routines enable the programmer to perform block- or character-level I/O operations without concern for device address and operating characteristics. System services, such as time of day and system configuration determination, are provided by BIOS.

BIOS provides an operational interface to the system and relieves the programmer of the concern about the characteristics of hardware devices. The BIOS interface insulates the user from the hardware, thus allowing new devices to be added to the system, yet retaining the BIOS level interface to the device. In this manner, user programs become transparent to hardware modifications and enhancements.

IBM Personal Computer Macro Assembler and IBM Personal Computer Disk Operating System (DOS) provide useful programming information related to this section.

Use of BIOS

Access to BIOS is through the software interrupts. Each BIOS entry point is available through its own interrupt as shown in Figure 4-1 on page 4-5.

The software interrupts access different BIOS routines. For example, to determine the amount of memory available in the system, interrupt hex 12 invokes the BIOS routine for determining memory size and returns the value to the caller.

All parameters passed to and from the BIOS routines go through the system registers. The prologue of each BIOS function indicates the registers used on the call and the return. For the memory size example, no input parameters are passed. The memory size, in 1K-byte increments, is returned in the AX register.
If a BIOS function call has several possible operations, the AH register is used at input to indicate the desired operation. For example, to set the time-of-day timer, the following code is required:

```
MOV AH,1 ;function is to set time of day.
MOV CX,HIGH-COUNT ;establish the current.
MOV DX,LOW-COUNT
INT 1AH ;set the time.
```

To read the time-of-day timer:

```
MOV AH,0 ;function to read the time of day.
INT 1AH ;read the timer.
```

Generally, the BIOS routines save all registers, except for AX and the flags. Other registers are modified on return only if they are returning a value to the caller. The exact register can be seen in the prologue of each BIOS function.

### Stack Requirements

The applications must reserve a minimum of 256 bytes on program stack for BIOS. This area is used to process interrupts. Any applications that use the real-time clock BIOS interrupt hex 15 (functions hex 83 and 86) should reserve an additional 30 bytes on the program stack for BIOS. This stack area is in addition to the area required by the application.

### BIOS and Hardware Interrupt Vectors

Figure 4-1 shows the assignment of the interrupt vectors. Following the figure is a description of each of the BIOS and hardware interrupt vectors. The function codes are also provided when appropriate.
<table>
<thead>
<tr>
<th>Addr. (hex)</th>
<th>Int. (hex)</th>
<th>Function</th>
<th>BIOS Entry</th>
</tr>
</thead>
<tbody>
<tr>
<td>0-3</td>
<td>0</td>
<td>Divide by Zero</td>
<td>D11</td>
</tr>
<tr>
<td>4-7</td>
<td>1</td>
<td>Single Step</td>
<td>D11</td>
</tr>
<tr>
<td>8-B</td>
<td>2</td>
<td>Nonmaskable Interrupt</td>
<td>NMI–FLIH</td>
</tr>
<tr>
<td>C-F</td>
<td>3</td>
<td>Breakpoint</td>
<td>D11</td>
</tr>
<tr>
<td>10-13</td>
<td>4</td>
<td>Overflow</td>
<td>D11</td>
</tr>
<tr>
<td>14-17</td>
<td>5</td>
<td>Print Screen</td>
<td>PRINT–SCREEN</td>
</tr>
<tr>
<td>18-1B</td>
<td>6</td>
<td>Reserved</td>
<td>D11</td>
</tr>
<tr>
<td>1C-1F</td>
<td>7</td>
<td>Reserved</td>
<td>D11</td>
</tr>
<tr>
<td>20-23</td>
<td>8</td>
<td>Level 0</td>
<td>TIMER–INT</td>
</tr>
<tr>
<td>24-27</td>
<td>9</td>
<td>Level 1</td>
<td>KB–INT</td>
</tr>
<tr>
<td>28-2B</td>
<td>A</td>
<td>Level 2</td>
<td>D11</td>
</tr>
<tr>
<td>2C-2F</td>
<td>B</td>
<td>Level 3</td>
<td>D11</td>
</tr>
<tr>
<td>30-33</td>
<td>C</td>
<td>Level 4</td>
<td>D11</td>
</tr>
<tr>
<td>34-37</td>
<td>D</td>
<td>Level 5</td>
<td>D11</td>
</tr>
<tr>
<td>38-3B</td>
<td>E</td>
<td>Level 6</td>
<td>DSKT–INT</td>
</tr>
<tr>
<td>3C-3F</td>
<td>F</td>
<td>Level 7</td>
<td>D11</td>
</tr>
<tr>
<td>40-43</td>
<td>10</td>
<td>Video</td>
<td>VIDEO–IO</td>
</tr>
<tr>
<td>44-47</td>
<td>11</td>
<td>Equipment</td>
<td>EQUIPMENT</td>
</tr>
<tr>
<td>48-4B</td>
<td>12</td>
<td>Memory</td>
<td>MEMORY–SIZE-DET</td>
</tr>
</tbody>
</table>

Figure 4-1 (Part 1 of 2). Interrupt Vector Assignments
<table>
<thead>
<tr>
<th>Addr. (hex)</th>
<th>Int. (hex)</th>
<th>Function</th>
<th>BIOS Entry</th>
</tr>
</thead>
<tbody>
<tr>
<td>4C–4F</td>
<td>13</td>
<td>Diskette DISKETTE-IO</td>
<td></td>
</tr>
<tr>
<td>50–53</td>
<td>14</td>
<td>Communications RS232-IO</td>
<td></td>
</tr>
<tr>
<td>54–57</td>
<td>15</td>
<td>System Services SYS-SERVICES</td>
<td></td>
</tr>
<tr>
<td>58–5B</td>
<td>16</td>
<td>Keyboard KEYBOARD-IO</td>
<td></td>
</tr>
<tr>
<td>5C–5F</td>
<td>17</td>
<td>Printer PRINTER-IO</td>
<td></td>
</tr>
<tr>
<td>60–63</td>
<td>18</td>
<td>Resident BASIC F600:0000</td>
<td></td>
</tr>
<tr>
<td>64–67</td>
<td>19</td>
<td>Bootstrap BOOT-STRAP</td>
<td></td>
</tr>
<tr>
<td>68–6B</td>
<td>1A</td>
<td>Time of Day TIME-OF-DAY</td>
<td></td>
</tr>
<tr>
<td>6C–6F</td>
<td>1B</td>
<td>Keyboard Break DUMMY-RETURN</td>
<td></td>
</tr>
<tr>
<td>70–73</td>
<td>1C</td>
<td>Timer Tick 0 DUMMY-RETURN</td>
<td></td>
</tr>
<tr>
<td>74–77</td>
<td>1D</td>
<td>Video VIDEO-PARMS</td>
<td></td>
</tr>
<tr>
<td>78–7B</td>
<td>1E</td>
<td>Diskette Parameters DISK-BASE</td>
<td></td>
</tr>
<tr>
<td>7C–7F</td>
<td>1F</td>
<td>Upper Character CRT-CHAR-GEN2</td>
<td></td>
</tr>
<tr>
<td>110–113</td>
<td>44</td>
<td>Lower Character CRT-CHAR-GEN1</td>
<td></td>
</tr>
<tr>
<td>128–12B</td>
<td>4A</td>
<td>Real-Time Clock Alarm DUMMY-RETURN</td>
<td></td>
</tr>
<tr>
<td>1B0–1B3</td>
<td>6C</td>
<td>System Resume Vector DUMMY-RETURN</td>
<td></td>
</tr>
</tbody>
</table>

Figure 4-1 (Part 2 of 2). Interrupt Vector Assignments

4-6 BIOS
Divide by Zero (Hex 0)

The divide by zero interrupt is generated after a division instruction, if the quotient exceeds the maximum allowed value.

Single Step (Hex 1)

The single step interrupt is used to execute a program one instruction at a time. An interrupt is generated after each program instruction is executed. This interrupt allows various diagnostic functions to be performed.

An IRET instruction is used to return from the interrupt service routine.

Nonmaskable (Hex 2)

Because the IBM PC Convertible uses the nonmaskable interrupt for BIOS support, applications should avoid changing this vector. The NMI is used for the following:

- I/O channel check. This NMI is generated when an I/O device activates the ‘I/O check’ line to signal a check condition. This causes BIOS to clear the screen and display the I/O check icon. This check is cleared by either a system power off/on sequence or a Ctrl, Fn, and Del key sequence.

- Diskette controller power on request. This NMI is generated when either BIOS or an application program attempts to access any of the following diskette controller registers when diskette controller power is off:
  - Digital output register (hex 3F2)
  - Main status register (hex 3F4)
  - Data register (3F5).

BIOS uses this NMI to restore power to, and initialize the diskette controller. Once this is accomplished, the instruction that attempted to access the diskette controller is re-executed.
This NMI is not generated when switching from one drive to another.

- System suspend. This NMI is generated when the system is powered off and it causes BIOS to save the current state of the system.

- Real-time clock. This NMI is generated when a periodic, update ended, or alarm interrupt occurs.

- Keyboard. This NMI is generated when the keyboard controller detects a keystroke. A second NMI is generated when either BIOS or an application has read the translated scan code from the I/O register at address hex 060 and has toggled the clear keyboard bit in the I/O register at address hex 061. The second NMI is held pending until a level-1 hardware interrupt has completed.

**Breakpoint (Hex 3)**

The breakpoint interrupt is used to set breakpoints in a software debug program. It is generated by executing a special interrupt request instruction.

**Overflow (Hex 4)**

The overflow interrupt is used when an overflow status is indicated and an INTO instruction is executed. The INTO instruction allows the processor to pass status information to an overflow error service routine.

**Print Screen (Hex 5)**

The print screen interrupt is invoked internally by the keyboard interrupt handler whenever a combination of the Shift and PrtSc keys is detected. The interrupt causes the image on the screen to be printed on the printer. The image does not print correctly, however, if the screen is in graphics mode.
Interrupts Hex 6 and Hex 7
These interrupts are reserved for future updates.

Level 0 (Hex 8)
The level-0 interrupt is a hardware interrupt. It is used by BIOS to service hardware interrupts from the system timer 0. The timer is set to interrupt the processor at 55 millisecond intervals. The timebase is used by BIOS to calculate timeouts and update the time-of-day counters.

Level 1 (Hex 9)
The level-1 (keyboard port hex 60 loaded) interrupt is a hardware interrupt. It is used by BIOS to service the IBM Personal Computer scan codes contained in the hex 60 scan code register. This interrupt routine clears the interrupt by toggling the clear keyboard bit in the register at hex 61 and issuing an end-of-interrupt instruction to the interrupt controller.

Level 2 (Hex A)
The level-2 interrupt is a hardware interrupt. It is used by application programs to service level-2 interrupts from I/O devices.

Level 3 (Hex B)
The level-3 interrupt is a hardware interrupt. It is used by application programs to service level-3 interrupt requests from communications devices, such as the secondary serial/parallel adapter or the internal modem.

Level 4 (Hex C)
The level-4 interrupt is a hardware interrupt. It is used by application programs to service level-4 interrupt requests from communications devices, such as the primary serial/parallel adapter.
Level 5 (Hex D)

The level-5 interrupt is a hardware interrupt. It is used by application programs to service level-5 interrupts from I/O devices.

Level 6 (Hex E)

The level-6 interrupt is a hardware interrupt. It is used by BIOS to service interrupts from the diskette controller. The controller uses this interrupt to signal operational and error status.

Level 7 (Hex F)

The level-7 interrupt is a hardware interrupt. It is used by application programs to service level-7 interrupt requests from I/O devices such as the IBM PC Convertible Printer and the parallel printer adapter. This interrupt level is also used to process a spurious interrupt. (A spurious interrupt is an interrupt request that was not held active for a sufficient amount of time.) Valid interrupts can be differentiated from spurious interrupts by checking the in-service flag bit in the interrupt service register; the flag bit will be set for valid interrupts and not set for spurious interrupts.

Video Input/Output (Hex 10)

The video I/O interrupt provides a common interface to the display and associated buffer. The function codes for this vector are:

<table>
<thead>
<tr>
<th>Code</th>
<th>Function</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>Set display mode</td>
</tr>
<tr>
<td>1</td>
<td>Set cursor type</td>
</tr>
<tr>
<td>2</td>
<td>Set cursor position</td>
</tr>
</tbody>
</table>
3 Read cursor position
4 Read light-pen position (not used for the LCD)
5 Set active display page (A/N only)
6 Scroll up active page
7 Scroll down active page
8 Read attribute and character at cursor position
9 Write attribute and character at cursor position
10 Write character only at the cursor position
11 Set color palette
12 Write dot
13 Read dot
14 Write teletype character to active page
15 Read current video state
16-18 Reserved
19 Write string
20 Load LCD character font or
   Set LCD high-intensity substitute
21 Return active display type and parameters

In graphics mode, the characters are accessed through the
graphics font areas, not the alphanumeric font areas. Storage
locations hex 110 through 113 (interrupt hex 44) point to the
lower 128-character font to be used in graphics mode and
locations hex 07C through 07F (interrupt hex 1F) point to the
upper 128-character font. During a cold start, the power-on
routines initialize these pointers to point to the default font in
ROM. Applications can specify other fonts by altering the
pointers, but the applications must restore the pointers before
the applications end. The fonts for alphanumeric modes on
the LCD can also be changed by using function code 20.
Equipment (Hex 11)

This interrupt returns 2 bytes that indicate the number of printers, communications devices, and diskette drives that are installed on the system. The interrupt also indicates the initial video mode.

Memory (Hex 12)

The memory size determination interrupt returns 2 bytes that contain the number of 1K bytes of contiguous blocks of memory that are available to the application. If the power-on routines find an error in memory, the memory beyond the failing position is not included in the amount.
## Diskette (Hex 13)

This interrupt provides the functions used to access the diskette hardware. The function codes for this routine are:

<table>
<thead>
<tr>
<th>Code</th>
<th>Function</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>Reset diskette system</td>
</tr>
<tr>
<td>1</td>
<td>Read diskette status from last operation</td>
</tr>
<tr>
<td>2</td>
<td>Read indicated sectors into memory</td>
</tr>
<tr>
<td>3</td>
<td>Write indicated sectors from memory</td>
</tr>
<tr>
<td>4</td>
<td>Verify indicated sectors with memory</td>
</tr>
<tr>
<td>5</td>
<td>Format indicated tracks</td>
</tr>
<tr>
<td>6–7</td>
<td>Reserved</td>
</tr>
<tr>
<td>8</td>
<td>Read drive parameters</td>
</tr>
<tr>
<td>9–14</td>
<td>Reserved</td>
</tr>
<tr>
<td>15</td>
<td>Read DASD type (determine media change line support)</td>
</tr>
<tr>
<td>16</td>
<td>Read media change line status</td>
</tr>
</tbody>
</table>
Communications (Hex 14)

This interrupt provides a common method of accessing the communications attachments. The function codes for this interrupt are:

<table>
<thead>
<tr>
<th>Code</th>
<th>Function</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>Initialize the selected communications port</td>
</tr>
<tr>
<td>1</td>
<td>Send a character (byte) over the line</td>
</tr>
<tr>
<td>2</td>
<td>Receive a character (byte) from the line</td>
</tr>
<tr>
<td>3</td>
<td>Return the port status (2 bytes)</td>
</tr>
</tbody>
</table>
## Event Post/Wait and System Services (Hex 15)

<table>
<thead>
<tr>
<th>Code</th>
<th>Function</th>
</tr>
</thead>
<tbody>
<tr>
<td>40</td>
<td>Read or modify system or modem profile</td>
</tr>
<tr>
<td>41</td>
<td>Wait on external event</td>
</tr>
<tr>
<td>42</td>
<td>Request system power off (system suspend)</td>
</tr>
<tr>
<td>43</td>
<td>Read current system status information</td>
</tr>
<tr>
<td>44</td>
<td>Activate or deactivate internal modem power</td>
</tr>
<tr>
<td>4F</td>
<td>BIOS keyboard interrupt hex 9 intercept</td>
</tr>
<tr>
<td>80–82</td>
<td>Reserved</td>
</tr>
<tr>
<td>83</td>
<td>Post event after elapsed time interval</td>
</tr>
<tr>
<td>84</td>
<td>Reserved</td>
</tr>
<tr>
<td>85</td>
<td>System request key changed state</td>
</tr>
<tr>
<td>86</td>
<td>Wait on elapsed time-interval</td>
</tr>
<tr>
<td>90</td>
<td>Device busy wait</td>
</tr>
<tr>
<td>91</td>
<td>Device interrupt complete</td>
</tr>
<tr>
<td>C0</td>
<td>Return system parameter pointer</td>
</tr>
</tbody>
</table>

The return system parameter pointer function returns a pointer to a table that defines the level of system support for BIOS and the system hardware. See the BIOS listing in Volume 2 for specific information.
Keyboard (Hex 16)

The keyboard interrupt provides access to the keyboard. The function codes for this interrupt are:

<table>
<thead>
<tr>
<th>Code</th>
<th>Function</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>Read the next ASCII character from the keyboard buffer area</td>
</tr>
<tr>
<td>1</td>
<td>Determine if an ASCII character is available to be read</td>
</tr>
<tr>
<td>2</td>
<td>Return the current state of the keyboard state keys (Alt, Shift, Ctrl)</td>
</tr>
<tr>
<td>3</td>
<td>Reserved</td>
</tr>
<tr>
<td>4</td>
<td>Enable or disable the keyboard clicker</td>
</tr>
</tbody>
</table>

The keyboard requires that an NMI be used to translate the IBM PC Convertible scan code to the IBM Personal Computer scan codes. The keyboard NMI routine writes the converted scan code into port hex 60 causing a level-1 hardware interrupt request. BIOS interrupt hex 9 routine is activated by the hardware interrupt to read the scan code from port hex 60. The interrupt-9 routine then converts the scan code to the appropriate ASCII or Extended ASCII code and places the code into a keyboard buffer. The code is then read by accessing the interrupt hex 16 keyboard routine.

Note: The read next character function call causes an internal wait on external event when a keystroke is not present. This is done to conserve battery power.
**Printer (Hex 17)**

The printer interrupt provides common access to all of the system printers. The function codes for this interrupt are:

<table>
<thead>
<tr>
<th>Code</th>
<th>Function</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>Print the indicated character</td>
</tr>
<tr>
<td>1</td>
<td>Initialize the printer port</td>
</tr>
<tr>
<td>3</td>
<td>Return the current printer status (1 byte)</td>
</tr>
</tbody>
</table>

**Resident BASIC (Hex 18)**

The resident BASIC interrupt transfers control to the resident BASIC program stored in ROM. The vector is initialized during power-on routines. The vector is accessed internally by the bootstrap routine when a diskette is not in drive A at power on and the F1 key has been pressed.

**Bootstrap (Hex 19)**

The bootstrap interrupt activates the bootstrap routine stored in system ROM. The routine clears the display, resets the diskette controller, and attempts to read in the boot record from drive 0. The boot record is located on track 0, sector 1, head 0 on the diskette. An insert diskette icon is displayed along with the F1 prompt, if drive 0 does not contain a diskette. When the F1 key is pressed, the system again attempts to read the boot record from drive 0. If drive 0 still does not contain a diskette, control is passed to the resident BASIC. If drive 0 contains a diskette and the boot record was successfully read, control is passed to the address in RAM (hex 07C00) where the boot record was stored.

If an error occurs while reading the boot record, or if the boot record was invalid, an bad diskette icon is displayed. The insert diskette icon, followed by the F1 prompt, is displayed. When the F1 key is pressed, the system again attempts to read the boot record and the sequence is restarted.
Time of Day (Hex 1A)

The time-of-day interrupt provides a common interface to the time, date, and alarm functions provided on the IBM PC Convertible. The function codes for this interrupt are:

<table>
<thead>
<tr>
<th>Code</th>
<th>Function</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>Read current timer 0 clock value</td>
</tr>
<tr>
<td>1</td>
<td>Set current timer 0 clock value</td>
</tr>
<tr>
<td>2</td>
<td>Read real-time clock time</td>
</tr>
<tr>
<td>3</td>
<td>Set real-time clock time</td>
</tr>
<tr>
<td>4</td>
<td>Read real-time clock date</td>
</tr>
<tr>
<td>5</td>
<td>Set real-time clock date</td>
</tr>
<tr>
<td>6</td>
<td>Set real-time clock alarm</td>
</tr>
<tr>
<td>7</td>
<td>Reset real-time clock alarm</td>
</tr>
<tr>
<td>8</td>
<td>Set real-time clock alarm activated power-on mode</td>
</tr>
<tr>
<td>9</td>
<td>Read alarm time and status</td>
</tr>
</tbody>
</table>

4-18 BIOS
Keyboard Break (Hex 1B)

The keyboard break interrupt points to a routine that is to be executed when the BREAK key on the keyboard is pressed and when responding to a keyboard interrupt. Control should be returned through an IRET instruction. The POST routines initialize this vector to an IRET instruction, so that nothing occurs when the BREAK key is pressed, unless the application modifies the vector.

Control may be retained by the break service routine, if the break occurs during interrupt processing. In this case, one or more 'end of interrupt' signals must be sent to the interrupt controller. Also, all I/O devices should be reset, in case an I/O operation was being processed when the break occurred.

Timer Tick (Hex 1C)

The timer tick interrupt points to the routine that is executed at every system-timer tick. This vector is used when responding to the 'timer 0 interrupt' signal, and control should be returned through an IRET instruction. The power-on routines initialize this vector to point to an IRET instruction, so that nothing occurs unless the application modifies the vector. The application must save and restore all registers that are modified by the application. The timer tick is set by the power-on sequence to occur 18.2 times per second.

Video Initialization (Hex 1D)

The video initialization interrupt points to a data region containing the parameters required for the initialization of the display controller. There are four separate tables that must be reproduced if all modes of display are supported. The power-on routines initialize this vector to point to the parameters in the system ROM. The application must restore the vector before the application ends.
Diskette Parameters (Hex 1E)

The diskette parameters interrupt points to a data region containing the parameters required for the diskette drive currently in use. The power-on routines initialize the vector to point to the parameters for drive A (system drive) in the system ROM. These default parameters represent the specified values for the IBM drives installed in the system unit. If other drives are installed, it may be necessary to modify these values.

Upper Character Graphics (Hex 1F)

When operating in graphics mode, the read/write character interface forms the characters from the ASCII code point by using a table of dot patterns. These patterns consist of 8 bytes of graphics information per character. The table of dot patterns for code points 128 through 255 is pointed to by the vector at hex 1F. BIOS uses this table as the default character set that is loaded into the LCD RAM font.

The application can change this vector to point to a different table of dot patterns, but the application must restore the pointers before the application ends.

Lower Character Graphics (Hex 44)

When operating in graphics mode, the read/write character interface forms the characters from the ASCII code point by using a table of dot patterns. These patterns consist of 8 bytes of graphics information. The table of dot patterns for the character code points 0 through 127 is contained in ROM and is pointed to by the vector at hex 44. BIOS uses this table as the default character set that is loaded into the LCD RAM font.

The application can change this vector to point to a different table of dot patterns, but the application must restore the pointers before the application ends.
Real-Time Clock Alarm (Hex 4A)

The alarm interrupt points to a user routine that is to be activated when the time in the real-time clock reaches the specified alarm time. The user routine is activated by the timer 0 interrupt routine after BIOS processes the RTC alarm interrupt (NMI) or when the system power was activated by the alarm and system operation was resumed successfully. The application should verify system status to determine if power was activated by the alarm function. The application must also restore the vector to the original pointer before the application ends.

The power-on routines initialize this vector to an IRET instruction.

System Resume Vector (Hex 6C)

The system resume interrupt points to a routine that is called by the power-on routines when the system is powered-on in resume mode, but prior to resuming the suspended application. The interrupt allows the operating system to correct real-time status, such as time and date, before the application is resumed. When the interrupt routine returns control to BIOS through an IRET instruction, the application’s registers are restored and control is returned to the application program.

This vector is initialized to an IRET when a cold start is performed.

Reserved Interrupt Allocations

Figure 4-2 on page 4-22 shows the interrupt vectors reserved for DOS, BIOS, BASIC, and application programs.
<table>
<thead>
<tr>
<th>Address (hex)</th>
<th>Irpt (hex)</th>
<th>Function</th>
</tr>
</thead>
<tbody>
<tr>
<td>80–83</td>
<td>20</td>
<td>DOS Program Terminate</td>
</tr>
<tr>
<td>84–87</td>
<td>21</td>
<td>DOS Function Call</td>
</tr>
<tr>
<td>88–8B</td>
<td>22</td>
<td>DOS Terminate Address</td>
</tr>
<tr>
<td>8C–8F</td>
<td>23</td>
<td>DOS Control Break Exit Address</td>
</tr>
<tr>
<td>90–93</td>
<td>24</td>
<td>DOS Fatal Error Vector</td>
</tr>
<tr>
<td>94–97</td>
<td>25</td>
<td>DOS Absolute Disk Read</td>
</tr>
<tr>
<td>98–9B</td>
<td>26</td>
<td>DOS Absolute Disk Write</td>
</tr>
<tr>
<td>9C–9F</td>
<td>27</td>
<td>DOS Terminate, Stay Resident</td>
</tr>
<tr>
<td>A0–FF</td>
<td>28–3F</td>
<td>Reserved for DOS</td>
</tr>
<tr>
<td>100–17F</td>
<td>40–5F</td>
<td>Reserved for BIOS</td>
</tr>
<tr>
<td>180–19F</td>
<td>60–67</td>
<td>Reserved for Application Programs</td>
</tr>
<tr>
<td>1A0–1BF</td>
<td>68–6F</td>
<td>Reserved</td>
</tr>
<tr>
<td>1C0–1DF</td>
<td>70–77</td>
<td>Reserved for hardware and BIOS</td>
</tr>
<tr>
<td>1E0–1FF</td>
<td>78–7F</td>
<td>Reserved</td>
</tr>
<tr>
<td>200–217</td>
<td>80–85</td>
<td>Reserved for BASIC and diagnostics</td>
</tr>
<tr>
<td>218–3C3</td>
<td>86–F0</td>
<td>Reserved for BASIC</td>
</tr>
<tr>
<td>3C4–3FF</td>
<td>F1–FF</td>
<td>Reserved</td>
</tr>
</tbody>
</table>

Figure 4-2. Reserved Interrupt Allocations
Other Reserved Areas

Figure 4-3 shows the low storage data area allocations that are reserved for the power-on routines, BIOS, DOS, and BASIC programs. Note that 256 bytes at locations hex 300 through hex 3FF are used as a stack area by the power-on and bootstrap routines. If the user desires the stack in a different area, the application must set the areas.

<table>
<thead>
<tr>
<th>Addr (hex)</th>
<th>Program</th>
<th>Function</th>
</tr>
</thead>
<tbody>
<tr>
<td>300–3FF</td>
<td>Stack</td>
<td>Reserved for POST and bootstrap</td>
</tr>
<tr>
<td>400–4EF</td>
<td>BIOS</td>
<td>See BIOS listing</td>
</tr>
<tr>
<td>4F0–4FF</td>
<td></td>
<td>Reserved as application communications area</td>
</tr>
<tr>
<td>500</td>
<td>DOS/BIOS</td>
<td>Print Screen Status Flag</td>
</tr>
<tr>
<td>504</td>
<td>DOS</td>
<td>Single Drive Mode Status Byte</td>
</tr>
<tr>
<td>510–511</td>
<td>BASIC</td>
<td>BASIC Segment Address</td>
</tr>
<tr>
<td>512–515</td>
<td>BASIC</td>
<td>Clock Interrupt Vector</td>
</tr>
<tr>
<td>516–519</td>
<td>BASIC</td>
<td>Break Key Interrupt Vector</td>
</tr>
<tr>
<td>51A–51D</td>
<td>BASIC</td>
<td>Disk Error Interrupt Vector</td>
</tr>
</tbody>
</table>

Figure 4-3. Reserved Low Storage Locations
Adapters with System-Accessible ROM

The IBM PC Convertible provides support for adapters that contain system-accessible ROM. This support allows control to be passed to the ROM in the adapter. To use this support, the adapters must have system-accessible ROM in the hex C0000 through F0000 address range. Address hex C0000 through C7FFF is reserved specifically for video adapters; the remaining addresses are available to all adapters.

At two points during the power-on sequence, the power-on routines check for the existence of valid 2K-byte blocks of adapter ROM. The power-on routines first check for adapter ROM in the video adapter address range (hex C0000 through C7800). This is done early in the power-on sequence, in order to determine which display is attached. Once the video adapter ROM has been validated, the adapter can intercept the following BIOS function calls:

- Video (interrupt hex 10)
- Video initialization (interrupt hex 1D)
- Upper-character graphics (interrupt hex 1F)
- Lower-character graphics (interrupt hex 1F).

Later in the power-on sequence, the power-on routines check for additional adapter ROM in the hex C8000 through F0000 range.
If the PC Convertible is to recognize the ROM in an adapter as valid, the first 4 bytes of ROM must contain the following:

- Byte 0 must contain hex 55.
- Byte 1 must contain hex AA.
- Byte 2 must contain a length indicator representing the number (in hex) of blocks in the ROM (length divided by 512).
- Byte 3 must contain an executable instruction.

The power-on routines also perform an integrity test on the ROM module. A checksum using modulo hex 100 is performed. The sum must be 0 for the module to be valid. If the ROM is valid, the power-on routines execute a far call to byte 3 of the ROM. The adapter can then perform test and initialization tasks.

If the adapter ROM detects a self-test error, the adapter should do the following before returning to the power-on routines:

- Set the feature error flag (bit 4 of absolute memory location hex 00412) to a 1.
- Set the device number for that adapter into register AH.
- Set a 2 digit error code into register AL.
If the adapter uses the video address range (hex C0000 through C7800), it must also set the following information into register BH before returning control to the power-on routines:

- 00 if the ROM does not support additional video adapters
- 02 if the adapter supports video in the color/graphics address space (buffer at hex B8000; registers in the hex 3Dx range)
- 04 if the adapter supports video in the monochrome address range (buffer at hex B0000, registers in the hex 3Bx range).

The power-on routines use this information to determine system configuration and to ensure that display contention does not occur if the LCD is installed. Failing to follow this procedure may cause unpredictable system operation and errors.

**BIOS Programming Guidelines**

The BIOS code is invoked through software interrupts. The programmer should not code BIOS addresses into applications.

**Warning:** The internal workings and absolute addresses within BIOS are subject to change without notice.

If an error is reported by the diskette code, the application should reset the drive adapter and retry the operation. A media-changed error does not need to be retried.

When altering I/O port bit values, the application should change only those bits that are necessary to complete the current task. Upon completion, the application should restore the original values. Failure to adhere to this practice may cause incompatibility with present and future systems.
Applications must allocate additional area in the stack segment for BIOS. See “Stack Requirements” on page 4-4 for these requirements.

Keyboard Encoding and Usage

The following explains how the keyboard interacts with BIOS and how the 83 key functions are accomplished on the IBM PC Convertible.

Encoding

The BIOS keyboard routine (interrupt hex 9) converts the keyboard scan codes into ASCII or Extended-ASCII codes. (Extended-ASCII codes are additional codes that cannot be represented in the standard ASCII codes.) The routine returns these codes in the keyboard buffer.

Character Codes

Figure 4-4 on page 4-28 shows the character codes returned by the BIOS keyboard routine to the system or application program. A blank in a column indicates that an ASCII code is not returned at the hex 16 interrupt level for a key or combination of keys. Figure 4-5 on page 4-33 and Figure 4-6 on page 4-33 show the key numbers. See “Keyboard and Keyboard Controller” on page 2-62 and Appendix A, “Character Sets and Keystrokes” for additional information.

Note: BIOS does not provide ASCII translation for non-U.S. keyboard.
<table>
<thead>
<tr>
<th>Key No.</th>
<th>Base</th>
<th>Case Upper</th>
<th>Ctrl</th>
<th>Alt</th>
<th>Fn</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>096</td>
<td>126</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>2</td>
<td>049</td>
<td>033</td>
<td></td>
<td>120</td>
<td></td>
</tr>
<tr>
<td>3</td>
<td>050</td>
<td>064</td>
<td>003¹</td>
<td>121</td>
<td></td>
</tr>
<tr>
<td>4</td>
<td>051</td>
<td>035</td>
<td></td>
<td>122</td>
<td></td>
</tr>
<tr>
<td>5</td>
<td>052</td>
<td>036</td>
<td></td>
<td>123</td>
<td></td>
</tr>
<tr>
<td>6</td>
<td>053</td>
<td>037</td>
<td></td>
<td>124</td>
<td></td>
</tr>
<tr>
<td>7</td>
<td>054</td>
<td>094</td>
<td>030</td>
<td>125</td>
<td></td>
</tr>
<tr>
<td>8</td>
<td>055</td>
<td>038</td>
<td></td>
<td>126</td>
<td>055</td>
</tr>
<tr>
<td>9</td>
<td>056</td>
<td>042</td>
<td></td>
<td>127</td>
<td>056</td>
</tr>
<tr>
<td>10</td>
<td>057</td>
<td>040</td>
<td></td>
<td>128</td>
<td>057</td>
</tr>
<tr>
<td>11</td>
<td>048</td>
<td>041</td>
<td></td>
<td>129</td>
<td></td>
</tr>
<tr>
<td>12</td>
<td>045</td>
<td>095</td>
<td>031</td>
<td>130</td>
<td>045</td>
</tr>
<tr>
<td>13</td>
<td>061</td>
<td>043</td>
<td></td>
<td>131</td>
<td>043</td>
</tr>
</tbody>
</table>

Figure 4-4 (Part 1 of 5). Key to ASCII Code Translation

¹Extended code, see “Extended-ASCII Codes” on page 4-34.
<table>
<thead>
<tr>
<th>Key No.</th>
<th>Base</th>
<th>Case</th>
<th>Ctrl</th>
<th>Alt</th>
<th>Fn</th>
</tr>
</thead>
<tbody>
<tr>
<td>14</td>
<td>092</td>
<td>124</td>
<td>028</td>
<td></td>
<td></td>
</tr>
<tr>
<td>15</td>
<td>008</td>
<td>008</td>
<td>127</td>
<td></td>
<td></td>
</tr>
<tr>
<td>16</td>
<td>009</td>
<td>015↑</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>17</td>
<td>113</td>
<td>081</td>
<td>017</td>
<td>016↑</td>
<td></td>
</tr>
<tr>
<td>18</td>
<td>119</td>
<td>087</td>
<td>023</td>
<td>017↑</td>
<td></td>
</tr>
<tr>
<td>19</td>
<td>101</td>
<td>069</td>
<td>005</td>
<td>019↑</td>
<td></td>
</tr>
<tr>
<td>20</td>
<td>114</td>
<td>082</td>
<td>018</td>
<td>018↑</td>
<td></td>
</tr>
<tr>
<td>21</td>
<td>116</td>
<td>084</td>
<td>020</td>
<td>020↑</td>
<td></td>
</tr>
<tr>
<td>22</td>
<td>121</td>
<td>089</td>
<td>025</td>
<td>021↑</td>
<td></td>
</tr>
<tr>
<td>23</td>
<td>117</td>
<td>085</td>
<td>021</td>
<td>022↑</td>
<td>052</td>
</tr>
<tr>
<td>24</td>
<td>105</td>
<td>073</td>
<td>009</td>
<td>023↑</td>
<td>053</td>
</tr>
<tr>
<td>25</td>
<td>111</td>
<td>079</td>
<td>015</td>
<td>024↑</td>
<td>054</td>
</tr>
<tr>
<td>26</td>
<td>112</td>
<td>080</td>
<td>016</td>
<td>025↑</td>
<td></td>
</tr>
<tr>
<td>27</td>
<td>091</td>
<td>123</td>
<td>027</td>
<td></td>
<td></td>
</tr>
<tr>
<td>28</td>
<td>093</td>
<td>125</td>
<td>029</td>
<td></td>
<td></td>
</tr>
<tr>
<td>30</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>31</td>
<td>097</td>
<td>065</td>
<td>001</td>
<td>030↑</td>
<td></td>
</tr>
<tr>
<td>32</td>
<td>115</td>
<td>083</td>
<td>019</td>
<td>031↑</td>
<td></td>
</tr>
<tr>
<td>33</td>
<td>100</td>
<td>068</td>
<td>004</td>
<td>032↑</td>
<td></td>
</tr>
<tr>
<td>34</td>
<td>102</td>
<td>070</td>
<td>006</td>
<td>033↑</td>
<td></td>
</tr>
</tbody>
</table>

Figure 4-4 (Part 2 of 5). Key to ASCII Code Translation
<table>
<thead>
<tr>
<th>Key No.</th>
<th>Case Base</th>
<th>Case Upper</th>
<th>Ctrl</th>
<th>Alt</th>
<th>Fn</th>
</tr>
</thead>
<tbody>
<tr>
<td>35</td>
<td>103</td>
<td>071</td>
<td>007</td>
<td>034</td>
<td></td>
</tr>
<tr>
<td>36</td>
<td>104</td>
<td>072</td>
<td>008</td>
<td>035</td>
<td></td>
</tr>
<tr>
<td>37</td>
<td>106</td>
<td>074</td>
<td>010</td>
<td>036</td>
<td>049</td>
</tr>
<tr>
<td>38</td>
<td>107</td>
<td>075</td>
<td>011</td>
<td>037</td>
<td>050</td>
</tr>
<tr>
<td>39</td>
<td>108</td>
<td>076</td>
<td>012</td>
<td>038</td>
<td>051</td>
</tr>
<tr>
<td>40</td>
<td>059</td>
<td>058</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>41</td>
<td>039</td>
<td>034</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>42</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>43</td>
<td></td>
<td>010</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>44</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>45</td>
<td>039</td>
<td>034</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>46</td>
<td>122</td>
<td>090</td>
<td>026</td>
<td>044</td>
<td></td>
</tr>
<tr>
<td>47</td>
<td>120</td>
<td>088</td>
<td>024</td>
<td>045</td>
<td></td>
</tr>
<tr>
<td>48</td>
<td>099</td>
<td>067</td>
<td>003</td>
<td>046</td>
<td></td>
</tr>
<tr>
<td>49</td>
<td>118</td>
<td>086</td>
<td>022</td>
<td>047</td>
<td></td>
</tr>
<tr>
<td>50</td>
<td>098</td>
<td>066</td>
<td>002</td>
<td>048</td>
<td></td>
</tr>
<tr>
<td>51</td>
<td>110</td>
<td>078</td>
<td>014</td>
<td>049</td>
<td></td>
</tr>
<tr>
<td>52</td>
<td>109</td>
<td>077</td>
<td>013</td>
<td>050</td>
<td>048</td>
</tr>
<tr>
<td>53</td>
<td>044</td>
<td>060</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>54</td>
<td>046</td>
<td>062</td>
<td></td>
<td></td>
<td>046</td>
</tr>
</tbody>
</table>

Figure 4-4 (Part 3 of 5). Key to ASCII Code Translation

\[2\text{For non-U.S. keyboards}\]
<table>
<thead>
<tr>
<th>Key No.</th>
<th>Base</th>
<th>Case Upper</th>
<th>Ctrl</th>
<th>Alt</th>
<th>Fn</th>
</tr>
</thead>
<tbody>
<tr>
<td>55</td>
<td>047</td>
<td>063</td>
<td>047</td>
<td></td>
<td></td>
</tr>
<tr>
<td>56</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>57</td>
<td>042</td>
<td>4</td>
<td>114</td>
<td>042</td>
<td></td>
</tr>
<tr>
<td>58</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>59</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>60</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>61</td>
<td>032</td>
<td>032</td>
<td>032</td>
<td>032</td>
<td>032</td>
</tr>
<tr>
<td>62</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>63</td>
<td>075</td>
<td>075</td>
<td>115</td>
<td>071</td>
<td>119</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>119</td>
<td>073</td>
<td>132</td>
</tr>
<tr>
<td>64</td>
<td>072</td>
<td>072</td>
<td>132</td>
<td>073</td>
<td>132</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td>081</td>
<td>118</td>
</tr>
<tr>
<td>65</td>
<td>080</td>
<td>080</td>
<td>118</td>
<td>081</td>
<td>118</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td>079</td>
<td>117</td>
</tr>
<tr>
<td>66</td>
<td>077</td>
<td>077</td>
<td>116</td>
<td>079</td>
<td>117</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>67</td>
<td>027</td>
<td>027</td>
<td>027</td>
<td></td>
<td>4</td>
</tr>
<tr>
<td>68</td>
<td>059</td>
<td>084</td>
<td>094</td>
<td>104</td>
<td>4</td>
</tr>
<tr>
<td>69</td>
<td>060</td>
<td>085</td>
<td>095</td>
<td>105</td>
<td>4</td>
</tr>
<tr>
<td>70</td>
<td>061</td>
<td>086</td>
<td>096</td>
<td>106</td>
<td></td>
</tr>
<tr>
<td>71</td>
<td>062</td>
<td>087</td>
<td>097</td>
<td>107</td>
<td></td>
</tr>
</tbody>
</table>

Figure 4-4 (Part 4 of 5). Key to ASCII Code Translation

1Indicates code returned when both Ctrl and Fn keys are pressed at the same time.
2Certain key combinations directly invoke the keyboard processor. See “Special Handling” on page 4-35 for additional information.
<table>
<thead>
<tr>
<th>Key No.</th>
<th>Case Base</th>
<th>Case Upper</th>
<th>Ctrl</th>
<th>Alt</th>
<th>Fn</th>
</tr>
</thead>
<tbody>
<tr>
<td>72</td>
<td>063\textsuperscript{1}</td>
<td>088\textsuperscript{1}</td>
<td>098\textsuperscript{1}</td>
<td>108\textsuperscript{1}</td>
<td></td>
</tr>
<tr>
<td>73</td>
<td>064\textsuperscript{1}</td>
<td>089\textsuperscript{1}</td>
<td>099\textsuperscript{1}</td>
<td>109\textsuperscript{1}</td>
<td></td>
</tr>
<tr>
<td>74</td>
<td>065\textsuperscript{1}</td>
<td>090\textsuperscript{1}</td>
<td>100\textsuperscript{1}</td>
<td>110\textsuperscript{1}</td>
<td></td>
</tr>
<tr>
<td>75</td>
<td>066\textsuperscript{1}</td>
<td>091\textsuperscript{1}</td>
<td>101\textsuperscript{1}</td>
<td>111\textsuperscript{1}</td>
<td></td>
</tr>
<tr>
<td>76</td>
<td>067\textsuperscript{1}</td>
<td>092\textsuperscript{1}</td>
<td>102\textsuperscript{1}</td>
<td>112\textsuperscript{1}</td>
<td></td>
</tr>
<tr>
<td>77</td>
<td>068\textsuperscript{1}</td>
<td>093\textsuperscript{1}</td>
<td>103\textsuperscript{1}</td>
<td>113\textsuperscript{1}</td>
<td></td>
</tr>
<tr>
<td>78</td>
<td></td>
<td></td>
<td></td>
<td>4</td>
<td>4</td>
</tr>
<tr>
<td>79</td>
<td></td>
<td></td>
<td></td>
<td>4</td>
<td>4</td>
</tr>
<tr>
<td>80</td>
<td>082\textsuperscript{1}</td>
<td>082\textsuperscript{1}</td>
<td></td>
<td>114\textsuperscript{1}</td>
<td></td>
</tr>
<tr>
<td>81</td>
<td>083\textsuperscript{1}</td>
<td>083\textsuperscript{1}</td>
<td></td>
<td>4</td>
<td>4</td>
</tr>
</tbody>
</table>

Figure 4-4 (Part 5 of 5). Key to ASCII Code Translation
Figure 4-5. IBM PC Convertible U.S. Keyboard Layout

Figure 4-6. IBM PC Convertible Non-U.S. Keyboard Layout
Extended-ASCII Codes

Extended-ASCII codes represent certain functions that cannot be represented in the standard ASCII code. When returning from a keyboard hex 16 interrupt, a character code of 000 (null) is returned in register AL. This indicates that either the system or the application program should examine a second code that will indicate the actual function. The second code is returned in register AH. The codes are shown in Figure 4-4 on page 4-28.
**Special Handling**

The following explains how certain keys cause special processing.

**System Reset**

The combination of the Ctrl, Alt, and Del keys results in the keyboard routine starting a software reset operation and a reload of the operating system.

This reset results in a hardware reset to the system, and the power-on self tests are run before the operating system is reloaded; however, memory is cleared and not tested.

**System Cold Start and Test**

The combination of the Ctrl, Fn, and Del keys results in the hardware resetting the system and initiating a complete power-on self-test procedure before reloading the system from the diskette. This sequence allows the user to force a full test of the system and memory before reloading the operating system.

**Pause**

The combination of the Ctrl and Num Lock keys causes the keyboard routine to stop noninterrupt-driven processing. The keyboard routine then waits for a key other than the Num Lock key to be pressed before resuming operation. This provides a system method of suspending functions such as list and print, then allows the function to resume with the next keystroke.

A wait on external event function call is internally invoked while in a pause in order to conserve battery power.

**Print Screen**

The combination of the Shift and PrtSc keys causes the contents of the display to be listed on the printer. Nonprintable characters are left blank on the listing. The contents of the display do not print correctly if the screen is in graphics mode.
Clicker Control

The combination of the Fn and the Caps Lock keys activates the keyboard clicker, if it is not active, and deactivates the keyboard clicker, if it is active. The keyboard routine handles this function internally and scan codes are not presented to the register at hex 60. The keyboard clicker is activated after a cold start.

Audio Control

The combination of the Fn and the Scroll Lock keys enables the speaker, if it is disabled, and disables the speaker, if it is enabled. The keyboard routine handles this function internally, and scan codes are not presented to the register at hex 60. During the power-on tests, the speaker is activated in order to signal error- or successful-completion conditions. The speaker is also activated (the default) after a cold start.

System Request

The combination of the Fn and Esc keys causes a system request scan code to be set by the keyboard routine. The keyboard routine checks for this scan code and issues a hex 15 interrupt with register AX set to either hex 8500 to indicate that the keys are held pressed (make condition) or hex 8501 to indicate that the keys were released (break condition).

Break

The combination of the Ctrl and Break keys results in the keyboard routine signaling interrupt hex 1A. Extended characters hex 00 are also returned in AL and AH.
Function Keys 11 and 12

The combination of the Fn key and F1 or F2 causes function 11 or function 12 to be performed.

<table>
<thead>
<tr>
<th>Key No.</th>
<th>Case Base</th>
<th>Case Upper</th>
<th>Ctrl</th>
<th>Alt</th>
</tr>
</thead>
<tbody>
<tr>
<td>F11</td>
<td>133</td>
<td>135</td>
<td>137</td>
<td>139</td>
</tr>
<tr>
<td>F12</td>
<td>134</td>
<td>136</td>
<td>138</td>
<td>140</td>
</tr>
</tbody>
</table>

Other Characteristics

The keyboard routine does its own buffering at both the NMI and hex 9 interrupt levels. If a key is pressed with the buffer full, the alarm is sounded if the speaker is active.

The keyboard routine suppresses the typematic action of the following keys: Ctrl, Shift, Alt, Num Lock, Scroll Lock, Caps Lock, System Request, and Ins.

Special BIOS Functions

Because the IBM PC Convertible is portable and can operate from an internal battery, BIOS provides special functions to save battery power.

Resume mode is a system profile option that enables the user to power off the system and save the application that was loaded into the system at the time the power off was requested. The application is automatically restored when the system is powered on again. This is accomplished through the system suspend and system restore BIOS functions.
The sleep (wait on external event), LCD blank, and system power-off functions are used to reduce battery usage when the system is waiting on some external event.

**System Suspend (Power Off)**

When the user presses the power switch to power-off the system or when an application requests a system power-off through the system services BIOS function call (interrupt hex 15), the BIOS system suspend function saves the current state of the system in a reserved area within the LCD RAM font area. A checksum is done on the first 128K bytes of the storage and a successful flag is set. After approximately 2 seconds, the power supply shuts down the system. The following information is saved:

- System flags and registers
- LCD controller state
- System timer 0 mode and initial counter value
- System timer 2 mode and initial counter value
- Interrupt controller mode and mask registers
- System I/O register 1
- Serial adapter and internal modem states
- Interrupt vectors hex 0 through 1F, and 44
- Power-on self test area (hex 0300 through 053A).
A system suspend is not allowed in the following conditions. Causing a system suspend when one of these conditions exists causes an warning to be displayed the next time the system is powered on.

- The LCD is not the active display at power off time.
- The diskette drive motor is active at power off time.
- The display configuration is changed during resume.

**System Resume (Power On)**

When the power-on sequence is started in resume mode, the self-test procedures are activated. If a system reset (cold start) is required, the hardware resets the processor and all devices and initiates a complete power-on self test procedure before reloading the system from the diskette. If system resume (warm start) is required, the hardware resets the processor and all devices and initiates a modified power-on self test procedure. At the successful completion of this procedure, the information saved during the system suspend sequence is restored and a return from NMI interrupt is performed. This causes the program that was executing when the suspend NMI was received to resume execution. The following devices, however, are not restored and are set to an initial state:

- Printers
- Keyboard.

In addition, BIOS indicates media changed during the first diskette access after a resume sequence. This allows the user to verify that the correct diskette is loaded in the system.
Sleep (Wait on External Event)

The sleep function is used to stop the system clocks when an application is waiting on some external event to occur and when BIOS has received a device-busy condition from either the keyboard or diskette. This function is used to reduce power consumption in order to extend battery life. The application can invoke this function through the event post/wait and systems services function call (interrupt hex 15). BIOS automatically invokes the sleep function for the device-busy conditions.

When sleep mode is active, interrupts are processed normally. After each interrupt is serviced, control is returned to the sleep function to determine if the event has occurred or the busy status has been cleared. If the event has not occurred, the sleep function again stops the system clocks. If the event has occurred or the condition has occurred, control is returned to the application.

An optional event time out (up to 14 seconds) can be specified. In this case, the sleep function returns control to the caller if the event does not occur within the specified time limit.
LCD Blank and Automatic Power-Off Functions

The LCD blank and automatic power-off functions are available only when operating the system on its internal battery. These functions are used to blank the display or automatically power off the IBM PC Convertible when there has been no keyboard or diskette activity for a period of time.

The system profile has two time values used to specify how long the IBM PC Convertible should wait for an entry from the keyboard or diskette access. One time-out value is used to specify the delay before the display is blanked; the other is used to specify the delay before the system is automatically powered off. Either or both values can be set by the user by using a system utility. During a cold start, these time-out functions are disabled (time values set to 0).

When the display has been blanked, it can be restored by pressing any key. However, the Fn key works best for this purpose as this key alone does not cause any other system action.

Thirty seconds before an automatic power-off occurs, an audible tone is issued to alert the user of the pending shutdown. If no keyboard or diskette activity is performed, the IBM PC Convertible begins a suspend (power-off) sequence. The IBM PC Convertible begins a resume sequence the next time the power on button is pressed and the suspended operation is resumed.

When the LCD blank, automatic power-off function, or low-battery warning is enabled, an interrupt from the real-time clock is enabled to cause an NMI once each second. This interrupt is used as a time base to calculate keyboard inactivity time and sample the low-battery signal. When all power-saving functions are disabled, the 1-second interrupt from the real-time clock is disabled.
Section 5. Compatibility with the IBM Personal Computer Family

<table>
<thead>
<tr>
<th>Topic</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>Compatibility Overview</td>
<td>5-2</td>
</tr>
<tr>
<td>Compatibility with the IBM Personal Computer AT</td>
<td>5-3</td>
</tr>
<tr>
<td>Special Programming Considerations</td>
<td>5-3</td>
</tr>
<tr>
<td>Altering the Nonmaskable Interrupt (NMI) Vector</td>
<td>5-4</td>
</tr>
<tr>
<td>Stack Manipulation</td>
<td>5-4</td>
</tr>
<tr>
<td>Stack Space</td>
<td>5-5</td>
</tr>
<tr>
<td>Idle Loops and Power Conservation</td>
<td>5-5</td>
</tr>
<tr>
<td>Timing Dependencies</td>
<td>5-6</td>
</tr>
<tr>
<td>Unequal Configurations</td>
<td>5-6</td>
</tr>
<tr>
<td>Hardware Differences</td>
<td>5-6</td>
</tr>
<tr>
<td>Clocks and Timers</td>
<td>5-7</td>
</tr>
<tr>
<td>Configuration Switches</td>
<td>5-7</td>
</tr>
<tr>
<td>Cassette</td>
<td>5-7</td>
</tr>
<tr>
<td>Liquid Crystal Displays</td>
<td>5-7</td>
</tr>
<tr>
<td>Direct Memory Access</td>
<td>5-11</td>
</tr>
<tr>
<td>Memory</td>
<td>5-11</td>
</tr>
<tr>
<td>Communications Adapters</td>
<td>5-11</td>
</tr>
<tr>
<td>Printers</td>
<td>5-12</td>
</tr>
<tr>
<td>Diskette Drives</td>
<td>5-13</td>
</tr>
<tr>
<td>Keyboard</td>
<td>5-13</td>
</tr>
<tr>
<td>Identification Byte</td>
<td>5-13</td>
</tr>
<tr>
<td>Summary</td>
<td>5-14</td>
</tr>
</tbody>
</table>
Compatibility Overview

The IBM PC Convertible differs from the other computers in the IBM Personal Computer family. Even though it is different, the IBM PC Convertible can run many applications designed for other IBM Personal Computers without requiring modifications to those applications. It is also possible to create applications for the IBM PC Convertible that will run without modifications on other IBM Personal Computers. In order to create such programs or to assess if current programs are compatible, you must understand the differences among the IBM Personal Computers and know the proper way to communicate with them.

Normally, it is not possible for a program written for one computer to run on a different computer, because the processors are different and the language of the application can not be executed by different processors. In this case, the application would have to be rewritten entirely in the language of the other processor. Because the IBM PC Convertible and the IBM Personal Computers use similar architecture and processors, most assembler language programs need not be modified.

Compatibility among processors alone is not enough, because the applications normally take advantage of device services (BIOS) and operating system (IBM DOS). In order for the applications to be compatible, the IBM PC Convertible has maintained all BIOS system interrupts and uses IBM DOS. This means that applications that use BIOS and IBM DOS interrupts on other IBM Personal Computers will operate in the same manner on the IBM PC Convertible.

Note: The BIOS microcode of the IBM PC Convertible is not identical to that of the other IBM Personal Computers. If an application bypasses the BIOS interrupt-calls and directly accesses routines and storage locations in one system, the application may not run in the other system. Some routines may be similar and some BIOS storage locations may be the same; however, it is strongly recommended that applications use only BIOS and IBM DOS interrupt interfaces in order to achieve compatibility in the IBM Personal Computer family.
Using the same language and the BIOS and IBM DOS interfaces goes a long way in achieving application compatibility. However, there are still several factors which need to be taken into consideration; this section describes those factors.

**Compatibility with the IBM Personal Computer AT**

The IBM Personal Computer AT has many capabilities that are not supported on the IBM PC Convertible. Refer to IBM Personal Computer AT Technical Reference for additional compatibility considerations.

**Special Programming Considerations**

In general, applications intended to run on any IBM Personal Computer family product should use programming practices that maximize code compatibility. One such practice is to use IBM DOS interfaces instead of going directly to the hardware, since hardware interfaces are likely to be different across the IBM Personal Computer family. As a result, compatibility must be considered in the design of an application. Applications that use these programming practices should have no problem running on the IBM PC Convertible. However, those applications that do not use these preferred programming practices must closely observe the compatibility exceptions.
Altering the Nonmaskable Interrupt (NMI) Vector

The IBM PC Convertible uses the NMI and its vector to access routines based in ROM. Routines, such as the keyboard, diskette controller power-on, real-time clock alarm processing, or system suspend are crucial to the operation of the system. Modifying the NMI vector prevents normal system operation, and the IBM PC Convertible will not operate with programs that alter the NMI vector.

Stack Manipulation

Certain compatibility aspects for the IBM PC Convertible require that the NMI be used to support normal operating functions and events. For example, a compatible keyboard interface is presented at I/O address hex 60 through a scan code preprocessor built into the NMI level. As a result, consideration must be given to sections of an application that manipulate the stack through the stack-segment and stack-pointer registers, to ensure that these two registers contain valid data when interrupts are possible. Masking the interrupts in this case is not sufficient on the IBM PC Convertible, because the interrupt mask has no effect on nonmaskable interrupts. Applications that manipulate the stack registers must ensure that the instruction that modifies the stack pointer (SP) immediately follows the instruction that modifies the stack segment (SS) as shown in the following example.

```
MOV SS, STACKSEG_VAL
MOV SP, STACKPTR_VAL
```

By following this rule, an application can prevent nonmaskable interrupts from interfering with the modification of the stack location.

In addition, the application must not use the stack pointer for any purpose other than when pointing to the stack, such as using the stack pointer as an intermediate register.

5-4 Compatibility
Failing to observe these rules may result in program malfunctions or loss of data.

**Stack Space**

Applications must allocate additional space in the stack segment for BIOS. See “Stack Requirements” on page 4-4 for these requirements.

**Idle Loops and Power Conservation**

The BIOS used on the IBM PC Convertible provides a sleep mode that is used to conserve power during periods of time when no specific processing is being done. BIOS automatically provides this function for applications that use BIOS to interact with the hardware. For example, BIOS automatically provides the sleep mode function for applications that use interrupt hex 16 to wait for keyboard activity and interrupt hex 13 to access the diskette drive.

However, applications that do their own idle processing while waiting on external events must observe special programming practices; otherwise, the application consumes as much power during the idle period as it does during normal processing. In this case, the application should use interrupt hex 15 (function code hex 41). See the BIOS listings for specific information concerning this function code.
Timing Dependencies

The internal storage in the IBM PC Convertible does not require refresh. Because the processor is not interrupted periodically to refresh memory, more processor cycles are available in a given period of time, and the processor appears to operate faster. This may affect a program that goes into a timing loop for delay.

Unequal Configurations

In designing an application to run on both the IBM PC Convertible and other IBM Personal Computers, ensure that the required hardware configuration is available on all machines. This means that all systems must meet the application's minimum requirements before the application can run properly.

Hardware Differences

To be able to run on any computer without change, an application using a specific I/O device must have access to identical devices or devices with identical operating characteristics and interfaces.

The following paragraphs describe the IBM PC Convertible-supported hardware functions and I/O devices that may differ from other IBM Personal Computers.
Clocks and Timers

**System Clock:** The IBM PC Convertible uses a system clock that supports sleep mode. Sleep mode is used to conserve battery power.

**Time of Day:** The IBM PC Convertible contains the circuitry to provide the time of day.

**Timers:** The IBM PC Convertible provides only timer channels 0 (modes 0, 2, 3, and 4) and 2 (all modes). Timer channel 1, dynamic memory refresh timing, is not required on the IBM PC Convertible.

Configuration Switches

The IBM PC Convertible does not contain configuration switches. Configuration is determined by power-on routines.

Cassette

The IBM PC Convertible does not support cassettes.

Liquid Crystal Displays

**Memory Mapping and Switching:** In order to be compatible with applications written for color/graphics and monochrome displays, the IBM PC Convertible uses two address ranges for control registers, one for color/graphics operations and one for monochrome operations. The address ranges can be selected by modifying the initial video mode bits in the BIOS equipment word and issuing a set mode function call (interrupt hex 10), if no other displays are attached. The default area for the LCD is the color/graphics area, if the IBM PC Convertible CRT display adapter is not installed. The default area is the monochrome area, if the IBM PC Convertible CRT Display Adapter is installed.
**RAM Fonts:** The LCD controller uses two fonts in RAM for character generation. Both the main and alternate fonts are loaded during power on with the standard IBM Personal Computer character set stored in system ROM. Font selection is accomplished through a BIOS function call (interrupt hex 10).

**Color Mapping:** Color is mapped on the LCD as follows:

- **Alphanumeric mode:**
  - White foreground with black background: Normal video
  - Black foreground with white background: Reverse video
  - Any color foreground with a different color background: Reverse video
  - Any color foreground with the same color background: Solid reverse video
  - Black foreground with black background: Nondisplay
  - Intensified characters: See “Intensity” on page 5-9.

- **Graphics mode (medium resolution):**
  - Background: Gray
  - Cyan or green: Dark gray 1
  - Magenta or red: Dark gray 2
  - White or brown: Black.

**LCD Aspect Ratio:** The LCD displays 640 pels horizontally with a pel density of 2.44 pels per millimeter (62 pels per inch) and 200 pels vertically with a pel density of 2.27 pels per millimeter (57.7 pels per inch). Cathode-ray displays typically have a lower vertical resolution pel density. This means that an image that appears as a square on an LCD appears as a vertical rectangle on a cathode-ray display. A BIOS function call (interrupt hex 15) allows applications with a scaling algorithm to adjust for physical display parameters.

---

5-8 Compatibility
**Intensity:** The intensity attribute of the cathode-ray tube display cannot be mapped properly onto the LCD, because there is no direct method of making a character darker on an LCD.

A programmable mapping of this attribute is provided through a BIOS function call that allows translation of the intensify attribute into reverse image, underline, select alternate font, or no attribute.

**Monochrome Emulation:** Monochrome emulation is supported through the LCD and LCD control logic and uses an 8-by-8 character box instead of a 9-by-14 character box. This support includes video buffer mapping and control ports. The IBM PC Convertible does not support the intensify attribute; see “Intensity” for additional information concerning this attribute.

**Color/Graphics Emulation:** The color/graphics emulation is supported through the LCD and the LCD control logic. This support includes video buffer mapping and control ports. The IBM PC Convertible LCD supports only two colors. The IBM PC Convertible LCD does not support the intensify attribute; see “Intensity” for additional information concerning this attribute.

**Accessing the Refresh Buffer:** Applications that directly access the refresh buffer do not need to synchronize the access with vertical and horizontal syncs. Additionally, the application does not need to disable video while accessing the refresh buffer. The LCD controller automatically resolves any memory contention without affecting the display.

Faded scan lines across the top and center of the display may occur if the application disables video during frequent updates to the refresh buffer.
Direct Memory Access

**Channels:** The IBM PC Convertible has three DMA channels instead of four. The DMA channels supported are 1, 2, and 3.

**Control Modes:** The IBM PC Convertible does not support the entire set of DMA control modes, but this does not affect compatibility if the applications use built-in BIOS and DOS routines to access the DMA channels. DMA channel 1, dynamic memory refresh timing, is not required on the IBM PC Convertible.

Memory

The IBM PC Convertible can support up to 512K bytes of user read/write memory. The IBM PC Convertible does not use this memory for the screen buffers. Therefore, the IBM PC Convertible video architecture does not affect the amount of user memory in the same way as that required by certain applications on other systems.

Communications Adapters

**Synchronous Communications:** The IBM PC Convertible does not support synchronous communications.

**Asynchronous Communication:** The IBM PC Convertible supports two coresident asynchronous-type adapters. One is the IBM PC Convertible Internal Modem and the other is the IBM PC Convertible Serial/Parallel Adapter. The modem adapter is always COM1 and the RS-232 will be either COM2 or COM1, depending on whether the IBM PC Convertible Internal Modem is installed. These features are described in Section 3, “System Options.”
The power-on self-test routines determine the presence of communications adapters by using the work (scratch) register within the INS8250A Asynchronous Communications Element. These routines will not be able to detect the presence of adapters that do not contain this register.

Applications that process multiple interrupt conditions from the INS8250A Asynchronous Communications Element must service and clear the interrupt conditions before exiting the interrupt service routine. Failure to clear the interrupt conditions can result in failure of the application program.

Printers

The IBM PC Convertible supports the IBM PC Convertible Portable Printer through the I/O connector. Other printers, such as the IBM Graphics Printer, can be attached by using the IBM PC Convertible Serial/Parallel Adapter. These features are described in Section 3, “System Options.”

The IBM PC Convertible Portable Printer is always LPT1 and the parallel interface interface is either LPT2 or LPT1, depending on whether the portable printer is installed.

Serial interface printers can be attached through the serial port. These printers are designated as either COM1 or COM2 by DOS and BASIC.

**IBM PC Convertible Portable Printer Aspect Ratio:** The IBM PC Convertible Portable Printer is designed to support the IBM Personal Computer Graphics Printer command stream. However, due to differences in the physical dimension of the print head, bit-image graphic prints documents about 20 percent longer in the vertical direction.
Diskette Drives

The IBM PC Convertible uses 88.9-millimeter (3.5-inch) drives instead of the 133.4-millimeter (5.25-inch) drives used on other IBM Personal Computers. The format and timing for the 88.9-millimeter (3.5-inch) drives is compatible with the 133.4 millimeter (5.25 inch) drives such that most applications function properly.

The IBM PC Convertible can support up to two 88.9-millimeter (3.5-inch) diskette drives that are capable of storing 720K bytes each. The interface to the diskette drives is based around the NEC µPD765 architecture and is compatible at this interface. The IBM PC Convertible also uses direct memory access (DMA) for data transfers to and from the diskettes. The IBM PC Convertible is capable of overlapped diskette I/O and other device I/O.

Keyboard

The IBM PC Convertible uses a 78-key keyboard that is capable of generating all 83 IBM Personal Computer scan codes. A nonmaskable interrupt is used to present compatible scan codes to the application.

Identification Byte

The IBM PC Convertible provides a byte in read-only memory that distinguishes an IBM PC Convertible system from other IBM Personal Computers. This byte is located at hex FFFFE and contains the value of hex F9.

5-12 Compatibility
Summary

The IBM PC Convertible is designed to be a member of the IBM Personal Computer family. The highest degree of compatibility can be achieved by using a common high-level language or accessing the system only through BIOS and DOS interrupts when designing an application for this family. Going below the BIOS level is not recommended, even though there may be compatible hardware on the different systems. When it is necessary to design for particular system differences, multiple paths can be built into the application, and the application can determine at execution time the particular system it is running on. This can be done by inspecting the byte at ROM location hex FFFFE for a specific value; this value will be hex F9 on the IBM PC Convertible.

Once the application has determined the IBM Personal Computer it is running on, the application can take the appropriate path.
5-14 Compatibility
The following table shows the hexadecimal and ASCII decimal values for the IBM PC Convertible character set. The value or character can be entered from the keyboard by pressing and holding the Alt key, then entering the digits for the decimal value on the numeric keypad. Note, however, that codes 0 through 31 are usually printer and communications control functions and may not display the character listed. Additionally, a null character (000) cannot be entered from the keypad.
<table>
<thead>
<tr>
<th>Hex</th>
<th>0x</th>
<th>1x</th>
<th>2x</th>
<th>3x</th>
<th>4x</th>
<th>5x</th>
<th>6x</th>
<th>7x</th>
</tr>
</thead>
<tbody>
<tr>
<td>x0</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>0</td>
<td>16</td>
<td>32</td>
<td>48</td>
<td>64</td>
<td>80</td>
<td>96</td>
<td>112</td>
</tr>
<tr>
<td>x1</td>
<td></td>
<td>!</td>
<td></td>
<td>A</td>
<td>Q</td>
<td>a</td>
<td>q</td>
<td></td>
</tr>
<tr>
<td>1</td>
<td>1</td>
<td>17</td>
<td>33</td>
<td>49</td>
<td>65</td>
<td>81</td>
<td>97</td>
<td>113</td>
</tr>
<tr>
<td>x2</td>
<td></td>
<td>&quot;</td>
<td></td>
<td>B</td>
<td>R</td>
<td>b</td>
<td>r</td>
<td></td>
</tr>
<tr>
<td>2</td>
<td>18</td>
<td>34</td>
<td>50</td>
<td>66</td>
<td>82</td>
<td>98</td>
<td>114</td>
<td></td>
</tr>
<tr>
<td>x3</td>
<td></td>
<td>#</td>
<td></td>
<td>C</td>
<td>S</td>
<td>c</td>
<td>s</td>
<td></td>
</tr>
<tr>
<td>3</td>
<td>19</td>
<td>35</td>
<td>51</td>
<td>67</td>
<td>83</td>
<td>99</td>
<td>115</td>
<td></td>
</tr>
<tr>
<td>x4</td>
<td></td>
<td>$</td>
<td></td>
<td>D</td>
<td>T</td>
<td>d</td>
<td>t</td>
<td></td>
</tr>
<tr>
<td>4</td>
<td>20</td>
<td>36</td>
<td>52</td>
<td>68</td>
<td>84</td>
<td>100</td>
<td>116</td>
<td></td>
</tr>
<tr>
<td>x5</td>
<td></td>
<td>%</td>
<td></td>
<td>E</td>
<td>U</td>
<td>e</td>
<td>u</td>
<td></td>
</tr>
<tr>
<td>5</td>
<td>21</td>
<td>37</td>
<td>53</td>
<td>69</td>
<td>85</td>
<td>101</td>
<td>117</td>
<td></td>
</tr>
<tr>
<td>x6</td>
<td></td>
<td>&amp;</td>
<td></td>
<td>F</td>
<td>V</td>
<td>f</td>
<td>v</td>
<td></td>
</tr>
<tr>
<td>6</td>
<td>22</td>
<td>38</td>
<td>54</td>
<td>70</td>
<td>86</td>
<td>102</td>
<td>118</td>
<td></td>
</tr>
<tr>
<td>x7</td>
<td></td>
<td>'</td>
<td></td>
<td>G</td>
<td>W</td>
<td>g</td>
<td>w</td>
<td></td>
</tr>
<tr>
<td>7</td>
<td>-23</td>
<td>39</td>
<td>55</td>
<td>71</td>
<td>87</td>
<td>103</td>
<td>119</td>
<td></td>
</tr>
<tr>
<td>x8</td>
<td></td>
<td>(</td>
<td></td>
<td>H</td>
<td>X</td>
<td>h</td>
<td>x</td>
<td></td>
</tr>
<tr>
<td>8</td>
<td>24</td>
<td>40</td>
<td>56</td>
<td>72</td>
<td>88</td>
<td>104</td>
<td>120</td>
<td></td>
</tr>
<tr>
<td>x9</td>
<td></td>
<td>)</td>
<td></td>
<td>I</td>
<td>Y</td>
<td>i</td>
<td>y</td>
<td></td>
</tr>
<tr>
<td>9</td>
<td>25</td>
<td>41</td>
<td>57</td>
<td>73</td>
<td>89</td>
<td>105</td>
<td>121</td>
<td></td>
</tr>
<tr>
<td>xA</td>
<td></td>
<td>:</td>
<td></td>
<td>J</td>
<td>Z</td>
<td>j</td>
<td>z</td>
<td></td>
</tr>
<tr>
<td>10</td>
<td>26</td>
<td>42</td>
<td>58</td>
<td>74</td>
<td>90</td>
<td>106</td>
<td>122</td>
<td></td>
</tr>
<tr>
<td>xB</td>
<td></td>
<td>;</td>
<td></td>
<td>K</td>
<td>[</td>
<td>k</td>
<td>{</td>
<td></td>
</tr>
<tr>
<td>11</td>
<td>27</td>
<td>43</td>
<td>59</td>
<td>75</td>
<td>91</td>
<td>107</td>
<td>123</td>
<td></td>
</tr>
<tr>
<td>xC</td>
<td></td>
<td>&lt;</td>
<td></td>
<td>L</td>
<td>\</td>
<td>l</td>
<td></td>
<td></td>
</tr>
<tr>
<td>12</td>
<td>28</td>
<td>44</td>
<td>60</td>
<td>76</td>
<td>92</td>
<td>108</td>
<td>124</td>
<td></td>
</tr>
<tr>
<td>xD</td>
<td></td>
<td>=</td>
<td></td>
<td>M</td>
<td>]</td>
<td>m</td>
<td></td>
<td></td>
</tr>
<tr>
<td>13</td>
<td>29</td>
<td>45</td>
<td>61</td>
<td>77</td>
<td>93</td>
<td>109</td>
<td>125</td>
<td></td>
</tr>
<tr>
<td>xE</td>
<td></td>
<td>&gt;</td>
<td></td>
<td>N</td>
<td>^</td>
<td>n</td>
<td>\</td>
<td></td>
</tr>
<tr>
<td>14</td>
<td>30</td>
<td>46</td>
<td>62</td>
<td>78</td>
<td>94</td>
<td>110</td>
<td>126</td>
<td></td>
</tr>
<tr>
<td>xF</td>
<td></td>
<td>/</td>
<td></td>
<td>O</td>
<td>-</td>
<td>o</td>
<td>\</td>
<td></td>
</tr>
<tr>
<td>15</td>
<td>31</td>
<td>47</td>
<td>63</td>
<td>79</td>
<td>95</td>
<td>111</td>
<td>127</td>
<td></td>
</tr>
</tbody>
</table>

**Character Sets A-3**
<table>
<thead>
<tr>
<th>Hex</th>
<th>8x</th>
<th>9x</th>
<th>Ax</th>
<th>Bx</th>
<th>Cx</th>
<th>Dx</th>
<th>Ex</th>
<th>Fx</th>
</tr>
</thead>
<tbody>
<tr>
<td>x0</td>
<td>Č</td>
<td>É</td>
<td>ā</td>
<td>ı</td>
<td>ı</td>
<td>ı</td>
<td>ı</td>
<td>ı</td>
</tr>
<tr>
<td></td>
<td>128</td>
<td>144</td>
<td>160</td>
<td>176</td>
<td>192</td>
<td>208</td>
<td>224</td>
<td>240</td>
</tr>
<tr>
<td>x1</td>
<td>Ž</td>
<td>Ž</td>
<td>Ž</td>
<td>Ž</td>
<td>Ž</td>
<td>Ž</td>
<td>Ž</td>
<td>Ž</td>
</tr>
<tr>
<td></td>
<td>129</td>
<td>145</td>
<td>161</td>
<td>177</td>
<td>193</td>
<td>209</td>
<td>225</td>
<td>241</td>
</tr>
<tr>
<td>x2</td>
<td>ė</td>
<td>ė</td>
<td>ė</td>
<td>ė</td>
<td>ė</td>
<td>ė</td>
<td>ė</td>
<td>ė</td>
</tr>
<tr>
<td></td>
<td>130</td>
<td>146</td>
<td>162</td>
<td>178</td>
<td>194</td>
<td>210</td>
<td>226</td>
<td>242</td>
</tr>
<tr>
<td>x3</td>
<td>ě</td>
<td>ě</td>
<td>ě</td>
<td>ě</td>
<td>ě</td>
<td>ě</td>
<td>ě</td>
<td>ě</td>
</tr>
<tr>
<td></td>
<td>131</td>
<td>147</td>
<td>163</td>
<td>179</td>
<td>195</td>
<td>211</td>
<td>227</td>
<td>243</td>
</tr>
<tr>
<td>x4</td>
<td>â</td>
<td>â</td>
<td>â</td>
<td>â</td>
<td>â</td>
<td>â</td>
<td>â</td>
<td>â</td>
</tr>
<tr>
<td></td>
<td>132</td>
<td>148</td>
<td>164</td>
<td>180</td>
<td>196</td>
<td>212</td>
<td>228</td>
<td>244</td>
</tr>
<tr>
<td>x5</td>
<td>ã</td>
<td>ã</td>
<td>ã</td>
<td>ã</td>
<td>ã</td>
<td>ã</td>
<td>ã</td>
<td>ã</td>
</tr>
<tr>
<td></td>
<td>133</td>
<td>149</td>
<td>165</td>
<td>181</td>
<td>197</td>
<td>213</td>
<td>229</td>
<td>245</td>
</tr>
<tr>
<td>x6</td>
<td>ã</td>
<td>ã</td>
<td>ã</td>
<td>ã</td>
<td>ã</td>
<td>ã</td>
<td>ã</td>
<td>ã</td>
</tr>
<tr>
<td></td>
<td>134</td>
<td>150</td>
<td>166</td>
<td>182</td>
<td>198</td>
<td>214</td>
<td>230</td>
<td>246</td>
</tr>
<tr>
<td>x7</td>
<td>ã</td>
<td>ã</td>
<td>ã</td>
<td>ã</td>
<td>ã</td>
<td>ã</td>
<td>ã</td>
<td>ã</td>
</tr>
<tr>
<td></td>
<td>135</td>
<td>151</td>
<td>167</td>
<td>183</td>
<td>199</td>
<td>215</td>
<td>231</td>
<td>247</td>
</tr>
<tr>
<td>x8</td>
<td>ê</td>
<td>ê</td>
<td>ê</td>
<td>ê</td>
<td>ê</td>
<td>ê</td>
<td>ê</td>
<td>ê</td>
</tr>
<tr>
<td></td>
<td>136</td>
<td>152</td>
<td>168</td>
<td>184</td>
<td>200</td>
<td>216</td>
<td>232</td>
<td>248</td>
</tr>
<tr>
<td>x9</td>
<td>ê</td>
<td>ê</td>
<td>ê</td>
<td>ê</td>
<td>ê</td>
<td>ê</td>
<td>ê</td>
<td>ê</td>
</tr>
<tr>
<td></td>
<td>137</td>
<td>153</td>
<td>169</td>
<td>185</td>
<td>201</td>
<td>217</td>
<td>233</td>
<td>249</td>
</tr>
<tr>
<td>xA</td>
<td>ê</td>
<td>ê</td>
<td>ê</td>
<td>ê</td>
<td>ê</td>
<td>ê</td>
<td>ê</td>
<td>ê</td>
</tr>
<tr>
<td></td>
<td>138</td>
<td>154</td>
<td>170</td>
<td>186</td>
<td>202</td>
<td>218</td>
<td>234</td>
<td>250</td>
</tr>
<tr>
<td>xB</td>
<td>ê</td>
<td>ê</td>
<td>ê</td>
<td>ê</td>
<td>ê</td>
<td>ê</td>
<td>ê</td>
<td>ê</td>
</tr>
<tr>
<td></td>
<td>139</td>
<td>155</td>
<td>171</td>
<td>187</td>
<td>203</td>
<td>219</td>
<td>235</td>
<td>251</td>
</tr>
<tr>
<td>xC</td>
<td>ê</td>
<td>ê</td>
<td>ê</td>
<td>ê</td>
<td>ê</td>
<td>ê</td>
<td>ê</td>
<td>ê</td>
</tr>
<tr>
<td></td>
<td>140</td>
<td>156</td>
<td>172</td>
<td>188</td>
<td>204</td>
<td>220</td>
<td>236</td>
<td>252</td>
</tr>
<tr>
<td>xD</td>
<td>ê</td>
<td>ê</td>
<td>ê</td>
<td>ê</td>
<td>ê</td>
<td>ê</td>
<td>ê</td>
<td>ê</td>
</tr>
<tr>
<td></td>
<td>141</td>
<td>157</td>
<td>173</td>
<td>189</td>
<td>205</td>
<td>221</td>
<td>237</td>
<td>253</td>
</tr>
<tr>
<td>xE</td>
<td>ê</td>
<td>ê</td>
<td>ê</td>
<td>ê</td>
<td>ê</td>
<td>ê</td>
<td>ê</td>
<td>ê</td>
</tr>
<tr>
<td></td>
<td>142</td>
<td>158</td>
<td>174</td>
<td>190</td>
<td>206</td>
<td>222</td>
<td>238</td>
<td>254</td>
</tr>
<tr>
<td>xF</td>
<td>ê</td>
<td>ê</td>
<td>ê</td>
<td>ê</td>
<td>ê</td>
<td>ê</td>
<td>ê</td>
<td>ê</td>
</tr>
<tr>
<td></td>
<td>143</td>
<td>159</td>
<td>175</td>
<td>191</td>
<td>207</td>
<td>223</td>
<td>239</td>
<td>255</td>
</tr>
</tbody>
</table>
Color Attribute Mapping

The following table shows the mapping of the color attribute byte. Note that the I (intensity) bit provides extra luminance to each available shade for monitors that recognize the intensity attribute. This results in the light colors listed in the figure.

<table>
<thead>
<tr>
<th>Character Code Byte (Even)</th>
<th>Attribute Byte (Odd)</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>B R G B I R G B</td>
</tr>
</tbody>
</table>

- Blink
- Background
- Foreground
- Intensity

### Background/Foreground Bits

<table>
<thead>
<tr>
<th>Bits</th>
<th>Attribute</th>
</tr>
</thead>
<tbody>
<tr>
<td>0000</td>
<td>Black; grey if I = 1</td>
</tr>
<tr>
<td>0001</td>
<td>Blue; light blue if I = 1</td>
</tr>
<tr>
<td>0010</td>
<td>Green; light green if I = 1</td>
</tr>
<tr>
<td>0011</td>
<td>Cyan; light cyan if I = 1</td>
</tr>
<tr>
<td>0100</td>
<td>Red; light red if I = 1</td>
</tr>
<tr>
<td>0101</td>
<td>Magenta; light magenta if I = 1</td>
</tr>
<tr>
<td>0110</td>
<td>Brown; light brown if I = 1</td>
</tr>
<tr>
<td>0111</td>
<td></td>
</tr>
<tr>
<td>1000</td>
<td></td>
</tr>
<tr>
<td>1001</td>
<td></td>
</tr>
<tr>
<td>1010</td>
<td></td>
</tr>
<tr>
<td>1011</td>
<td></td>
</tr>
<tr>
<td>1100</td>
<td></td>
</tr>
<tr>
<td>1101</td>
<td></td>
</tr>
<tr>
<td>1110</td>
<td></td>
</tr>
<tr>
<td>1111</td>
<td></td>
</tr>
</tbody>
</table>

Character Sets A-5
## LCD Attribute Mapping

The following table shows the mapping of the attribute byte for the LCD.

<table>
<thead>
<tr>
<th>Character Code Byte (Even)</th>
<th>Attribute Byte (Odd)</th>
</tr>
</thead>
<tbody>
<tr>
<td>7 6 5 4 3 2 1 0</td>
<td>7 6 5 4 3 2 1 0</td>
</tr>
</tbody>
</table>

- **Blink**
- **Background**
- **Foreground**
- **Intensity**

### Background/Foreground Bits

<table>
<thead>
<tr>
<th>Bits</th>
<th>Attribute</th>
</tr>
</thead>
<tbody>
<tr>
<td>6 5 4 2 1 0</td>
<td></td>
</tr>
<tr>
<td>0 0 0 0 0 0</td>
<td>Solid grey (nondisplay)</td>
</tr>
<tr>
<td>0 0 0 0 0 1</td>
<td>Normal video underscored</td>
</tr>
<tr>
<td>0 0 0 1 1 1</td>
<td>Normal video</td>
</tr>
<tr>
<td>1 1 1 0 0 0</td>
<td>Reverse video</td>
</tr>
<tr>
<td>1 1 1 1 1 1</td>
<td>Solid black (nondisplay)</td>
</tr>
</tbody>
</table>

The intensity attribute does not affect the intensity of the LCD, but may be used to select one of the display attributes shown in the following by using a BIOS function call (interrupt hex 10):

- Underscore
- Reverse video
- Select alternate font
- Ignore intensity bit.
Appendix B. Unit Specifications

System Unit ................................ B-2
Diskette Drives ............................. B-6
Random Access Memory ...................... B-8
Serial/Parallel Adapter ...................... B-9
Portable Printer ............................ B-11
CRT Display Adapter ........................ B-15
Monochrome Display ......................... B-18
Color Display .............................. B-20
Internal Modem ............................ B-22
AC Adapter (100–240 V ac) ................ B-24
Battery Charger ............................ B-25
Automobile Adapter ........................ B-25
## System Unit

### Physical Description

<p>| | |</p>
<table>
<thead>
<tr>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>Width</td>
<td>312 mm (12.28 inches)&lt;sup&gt;1&lt;/sup&gt;</td>
</tr>
<tr>
<td>Length</td>
<td>374 mm (14.72 inches)&lt;sup&gt;1&lt;/sup&gt;</td>
</tr>
<tr>
<td>Height</td>
<td>68 mm (2.68 inches)</td>
</tr>
<tr>
<td>Weight</td>
<td>5.5 kilograms (12.2 pounds) with 256K storage and the battery pack</td>
</tr>
</tbody>
</table>

<sup>1</sup>Measurement includes handle.

---

### B-2 Unit Specifications
## Environmental

| Temperature | System On: 10.0 to 40.6°C (50 to 105°F)  
|            | System Off (memory saved): 0.6 to 51.7°C (33 to 125°F)  
|            | Storage or Shipping: −40.0 to +60.0°C (−40 to 140°F) |
| Relative Humidity | Operating: 5 to 95%  
|                 | Storage or Shipping: 5 to 100% |
| Maximum Wet Bulb      | 29.4°C (85°F) |
| Acoustic Level       | 51 dB (operator position) |
| Heat Output           | 60 BTU per hour, maximum (256K storage) |
| Altitude               | 0 – 2135 meters (0 – 7000 feet) |

## Power Adapter Input

| Connector | Hosiden HEC, number 0470-01-250, or equivalent |
| Electrical | +10 to 16 V dc, 40 watts (maximum) |
## Battery Input

<table>
<thead>
<tr>
<th>Connector</th>
<th>Molex, number 70411-0001, or equivalent</th>
</tr>
</thead>
<tbody>
<tr>
<td>Electrical</td>
<td>+ 9.6 V dc, nominal (8 NiCad cells)</td>
</tr>
</tbody>
</table>

## I/O Channel

<table>
<thead>
<tr>
<th>Connector</th>
<th>Burndy, CEPW2X36V-1Z14, or equivalent</th>
</tr>
</thead>
</table>
| Output Signal Levels | Active: 3.8 V dc (minimum)  
Inactive: 0.4 V dc (maximum) |
| Input Signal Levels | Active: 2.0 to 5.25 V dc  
Inactive: −0.5 to +0.8 V dc |
| I/O Signal Drive | The I/O signals have sufficient drive (total power) to power up to 2 CMOS loads plus 1 low-power Schotty (LS) TTL load. |
| Output Voltages | + 5 V dc ± 5%, 290 mA²  
+ 12 V dc ± 10%, 75 mA²  
− 13 V dc ± 10%, 35 mA²  
+ Adapter Power (+ 9.2 to + 16 V dc) 5 watts² |

---

²The current and power values are typically available at the I/O channel connector of a full configured system (512K, 2 drives, and internal modem). External adapters should use ‘adapter power’ and provide any regulation if required.

---

### B-4 Unit Specifications
**LCD**

<table>
<thead>
<tr>
<th>Connector</th>
<th>26 pin, card edge</th>
</tr>
</thead>
</table>
| Output Signal Levels | Active: 3.8 V dc (minimum)  
                           Inactive: 0.4 V dc (maximum) |
| Input Signal Levels | Active: 3.15 to 5.25 V dc  
                           Inactive: −0.5 to +0.9 V dc |
| Voltages           | +5 V dc and −13 V dc             |
| Signal Drive       | All signals have sufficient drive to power 1 CMOS load. |

**System Operating Characteristics**

<table>
<thead>
<tr>
<th>Processor Instruction Set</th>
<th>80C88 compatible</th>
</tr>
</thead>
<tbody>
<tr>
<td>Clock Cycle Time</td>
<td>210 nanoseconds</td>
</tr>
<tr>
<td>Main Memory Cycle</td>
<td>840 nanoseconds</td>
</tr>
<tr>
<td>I/O Cycle</td>
<td>1.05 microseconds</td>
</tr>
<tr>
<td>DMA Cycle</td>
<td>1.05 microseconds</td>
</tr>
</tbody>
</table>
Diskette Drives

Physical Description

<table>
<thead>
<tr>
<th>Width</th>
<th>102 mm (4.08 inches)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Length</td>
<td>150 mm (6.0 inches)</td>
</tr>
<tr>
<td>Height</td>
<td>41 mm (1.64 inches)</td>
</tr>
<tr>
<td>Weight</td>
<td>680 grams (1.49 pounds)</td>
</tr>
</tbody>
</table>

Signal and Power Requirements

<table>
<thead>
<tr>
<th>Connector</th>
<th>40 pin, card edge</th>
</tr>
</thead>
<tbody>
<tr>
<td>Output Signal Levels</td>
<td>Active: 3.8 V dc (minimum)</td>
</tr>
<tr>
<td></td>
<td>Inactive: 0.4 V dc (maximum)</td>
</tr>
<tr>
<td>Input Signal Levels</td>
<td>Active: 2.0 V dc (minimum)</td>
</tr>
<tr>
<td></td>
<td>Inactive: +0.8 V dc (maximum)</td>
</tr>
<tr>
<td>Voltage and current</td>
<td>+5 V dc ± 10%, 200 mA</td>
</tr>
<tr>
<td></td>
<td>+12 V dc ± 10%, 258 mA</td>
</tr>
</tbody>
</table>
# Operating Characteristics

<table>
<thead>
<tr>
<th>Characteristic</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>Capacity unformatted</td>
<td>1.0M bytes</td>
</tr>
<tr>
<td>Recording density</td>
<td>8717 bits per inch</td>
</tr>
<tr>
<td>Track density</td>
<td>135 tracks per inch</td>
</tr>
<tr>
<td>Encoding method</td>
<td>Modified frequency modulation (MFM)</td>
</tr>
<tr>
<td>Rotational speed</td>
<td>300 RPM ± 3.0%</td>
</tr>
<tr>
<td>Transfer rate</td>
<td>250K bytes per second (MFM)</td>
</tr>
<tr>
<td>Access time</td>
<td>Track to track: 6 msec (maximum)</td>
</tr>
<tr>
<td></td>
<td>Head settle time: 15 msec (maximum)</td>
</tr>
<tr>
<td></td>
<td>Head load time: 0 msec (maximum)</td>
</tr>
</tbody>
</table>

# Diskette Requirements

<table>
<thead>
<tr>
<th>Requirement</th>
<th>Specification</th>
</tr>
</thead>
<tbody>
<tr>
<td>Certification</td>
<td>Double sided</td>
</tr>
<tr>
<td></td>
<td>135 tracks per inch</td>
</tr>
<tr>
<td></td>
<td>80 tracks per side</td>
</tr>
<tr>
<td></td>
<td>Soft sector</td>
</tr>
<tr>
<td>Recording density</td>
<td>8717 bits per inch</td>
</tr>
<tr>
<td>Medium</td>
<td>90 mm (3.5 inch) diskette cartridge</td>
</tr>
</tbody>
</table>
Random Access Memory

Physical Description

<p>| | |</p>
<table>
<thead>
<tr>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>Width</td>
<td>54.8 mm (2.19 inches)</td>
</tr>
<tr>
<td>Length</td>
<td>80.5 mm (3.22 inches)</td>
</tr>
<tr>
<td>Height</td>
<td>10 mm (0.4 inches)</td>
</tr>
<tr>
<td>Weight</td>
<td>38 grams (1.3 ounce)</td>
</tr>
</tbody>
</table>

Signal and Power Requirements

<table>
<thead>
<tr>
<th>Connectors</th>
<th>40 pin, right angle header and receptacle</th>
</tr>
</thead>
<tbody>
<tr>
<td>Output Signal Levels</td>
<td>Active: 3.8 V dc (minimum)</td>
</tr>
<tr>
<td></td>
<td>Inactive: 0.4 V dc (maximum)</td>
</tr>
<tr>
<td>Input Signal Levels</td>
<td>Active: 3.15 to 5.25 V dc</td>
</tr>
<tr>
<td></td>
<td>Inactive: −0.3 to +0.8 V dc</td>
</tr>
<tr>
<td>Voltage</td>
<td>+5 V dc ± 10%</td>
</tr>
</tbody>
</table>

Operating Characteristics

<table>
<thead>
<tr>
<th>Capacity</th>
<th>128K bytes per card, 4 cards maximum</th>
</tr>
</thead>
<tbody>
<tr>
<td>Interface</td>
<td>CMOS</td>
</tr>
<tr>
<td>Read and Write Cycles</td>
<td>840 nanoseconds</td>
</tr>
</tbody>
</table>
Serial/Parallel Adapter

Physical Description

<p>| | |</p>
<table>
<thead>
<tr>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>Width</td>
<td>312 mm (12.28 inches)</td>
</tr>
<tr>
<td>Length</td>
<td>33.7 mm (1.33 inches)³</td>
</tr>
<tr>
<td>Height</td>
<td>68 mm (2.68 inches)</td>
</tr>
<tr>
<td>Weight</td>
<td>470 grams (1.04 pounds)</td>
</tr>
</tbody>
</table>

Environmental

| Temperature          | System On: 10.0 to 40.6°C (50 to 105°F)  |
|                     | Storage or Shipping: -40.0 to +60.0°C (-40 to +140°F) |
| Relative Humidity   | Operating: 5 to 95%  |
|                    | Storage or Shipping: 5 to 100%  |
| Maximum Wet Bulb   | 29.4°C (85°F) |

³Apparent length when attached to the system unit.
## Signal and Power Requirements

### I/O Channel

<table>
<thead>
<tr>
<th>Connector</th>
<th>Burndy, FT72A-2, or equivalent</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Output Signal Levels</strong></td>
<td>Active: 3.8 V dc (minimum)</td>
</tr>
<tr>
<td></td>
<td>Inactive: 0.4 V dc (maximum)</td>
</tr>
<tr>
<td><strong>Input Signal Levels</strong></td>
<td>Active: 3.4 to 5.25 V dc</td>
</tr>
<tr>
<td></td>
<td>Inactive: −0.5 to +0.8 V dc</td>
</tr>
<tr>
<td><strong>Voltage and current</strong></td>
<td>+5 V dc ± 5%, 170 mA (maximum)</td>
</tr>
<tr>
<td></td>
<td>+12 V dc ± 15%, 22 mA (maximum)</td>
</tr>
<tr>
<td></td>
<td>−13 V dc ± 15%, 20 mA (maximum)</td>
</tr>
</tbody>
</table>

### Serial Port

<table>
<thead>
<tr>
<th>Connector</th>
<th>25 pin, D-shell</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Signal Levels</strong></td>
<td>Space: +3.0 to +15.0 V dc</td>
</tr>
<tr>
<td></td>
<td>Mark: −3.0 to −15.0 V dc</td>
</tr>
</tbody>
</table>

### Parallel Port

<table>
<thead>
<tr>
<th>Connector</th>
<th>25 pin, D-shell</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Output Signal Levels</strong></td>
<td>High level: +2.4 V dc (minimum) at 0.3 mA</td>
</tr>
<tr>
<td></td>
<td>Low level: +0.5 V dc (maximum) at 24 mA</td>
</tr>
<tr>
<td><strong>Input Signal Levels</strong></td>
<td>Low level: 0.8 V dc (maximum)</td>
</tr>
<tr>
<td></td>
<td>High level: 2.0 V dc (minimum)</td>
</tr>
</tbody>
</table>
Operating Characteristics

<table>
<thead>
<tr>
<th>Communications Interface</th>
<th>Asynchronous</th>
</tr>
</thead>
<tbody>
<tr>
<td>Data Rate (Asynchronous)</td>
<td>Supports up to 19200 baud</td>
</tr>
<tr>
<td>Processor Instruction Set</td>
<td>INS 8250A compatible</td>
</tr>
</tbody>
</table>

Portable Printer

Physical Description

<table>
<thead>
<tr>
<th>Width</th>
<th>309.9 mm (12.2 inches)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Length</td>
<td>110 mm (4.33 inches)(^4)</td>
</tr>
<tr>
<td>Height</td>
<td>68 mm (2.72 inches)</td>
</tr>
<tr>
<td>Weight</td>
<td>1.6 kilograms (3.5 pounds)</td>
</tr>
</tbody>
</table>

\(^4\)Apparent length when attached to system unit.
Environmental

<table>
<thead>
<tr>
<th>Temperature</th>
<th>System On:</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>10.0 to 35.0°C (50 to 95°F)</td>
</tr>
<tr>
<td>Storage or Shipping:</td>
<td>−28.9 to 45.0°C (−20 to 113°F)(^5)</td>
</tr>
<tr>
<td></td>
<td>5.0 to 35.0°C (41 to 95°F)(^6)</td>
</tr>
<tr>
<td></td>
<td>−5.0 to 45.0°C (−23 to 113°F)(^7)</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Relative Humidity</th>
<th>System On:</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>5 to 95%(^5)</td>
</tr>
<tr>
<td>Storage or Shipping:</td>
<td>45 to 85%(^6)</td>
</tr>
<tr>
<td></td>
<td>20 to 85%(^7)</td>
</tr>
</tbody>
</table>

| Maximum Wet Bulb     | 26.7°C (80°F)               |

| Acoustic Level       | 51 dB (operator position)   |

\(^5\)Without ribbon.  
\(^6\)With ribbon, up to two years.  
\(^7\)With ribbon, up to one month.
### Signal and Power Requirements

<table>
<thead>
<tr>
<th>Connector</th>
<th>Burndy, FT72-C2, or equivalent</th>
</tr>
</thead>
</table>
| **Output Signal Levels** | Active: 3.5 V dc (minimum)  
Inactive: 0.4 V dc (maximum) |
| **Input Signal Levels** | Active: 2.0 to 5.25 V dc  
Inactive: -0.5 to +0.8 V dc |
| **Voltage and current** | +8.0 to 16.0 V dc, 4.5 watts while printing average text |

### Operating Characteristics

<table>
<thead>
<tr>
<th>Print Method</th>
<th>Thermal or thermal transfer, nonimpact dot matrix</th>
</tr>
</thead>
<tbody>
<tr>
<td>Print Speed</td>
<td>40 characters per second (burst), 10 pitch</td>
</tr>
<tr>
<td>Print Direction</td>
<td>Left to right only</td>
</tr>
<tr>
<td>Carriage Return Speed</td>
<td>5 inches per second (reference)</td>
</tr>
<tr>
<td>Character Size</td>
<td>24 dots vertical by 18 dots horizontal (10 pitch)</td>
</tr>
<tr>
<td>Character Set</td>
<td>Full 96-character ASCII with descenders plus international characters and symbols (IBM Personal Computer Graphics Printer character set number 2)</td>
</tr>
</tbody>
</table>

**Unit Specifications**  B-13
<table>
<thead>
<tr>
<th>Line Spacing</th>
<th>6 and 8 lines per inch (character mode), and n/180 lines per inch (graphics mode)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Graphics Print Mode</td>
<td>Normal, dual, and high density all points addressable.</td>
</tr>
<tr>
<td>Print Modes</td>
<td>Standard at 10 characters per inch, 80 maximum characters per line</td>
</tr>
<tr>
<td></td>
<td>Double width at 5 characters per inch, 40 maximum characters per line</td>
</tr>
<tr>
<td></td>
<td>Compressed at 16.4 characters per inch, 132 maximum characters per line</td>
</tr>
<tr>
<td></td>
<td>Double-width compressed at 8.2 characters per inch, 66 maximum characters per line</td>
</tr>
<tr>
<td>Registration</td>
<td>± 6.0 millimeter (vertical) for each 61 printed lines (at 6 lines per inch)</td>
</tr>
<tr>
<td>Paper Type</td>
<td>Continuous roll or cut sheet</td>
</tr>
<tr>
<td>Paper Feed</td>
<td>Friction</td>
</tr>
<tr>
<td>Paper Width</td>
<td>216 millimeter (8.5 inch) maximum</td>
</tr>
<tr>
<td>Paper Type</td>
<td>Plain paper (20 pound or less) with thermal transfer ribbon or thermal paper (no ribbon required)</td>
</tr>
</tbody>
</table>
# CRT Display Adapter

## Physical Description

<table>
<thead>
<tr>
<th>Specification</th>
<th>Measurement</th>
</tr>
</thead>
<tbody>
<tr>
<td>Width</td>
<td>312 mm (12.28 inches)</td>
</tr>
<tr>
<td>Length</td>
<td>40.7 mm (1.64 inches)</td>
</tr>
<tr>
<td>Height</td>
<td>68 mm (2.68 inches)</td>
</tr>
<tr>
<td>Weight</td>
<td>635 grams (1.4 pounds)</td>
</tr>
</tbody>
</table>

## Environmental

<table>
<thead>
<tr>
<th>Temperature</th>
<th>Specification</th>
</tr>
</thead>
<tbody>
<tr>
<td>System On</td>
<td>10.0 to 40.6°C (50 to 105°F)</td>
</tr>
<tr>
<td>Storage or Shipping</td>
<td>−40.0 to 60.0°C (−40 to +140°F)</td>
</tr>
<tr>
<td>Relative Humidity</td>
<td>Operating: 5 to 95%</td>
</tr>
<tr>
<td></td>
<td>Storage: 5 to 100%</td>
</tr>
<tr>
<td>Maximum Wet Bulb</td>
<td>29.0°C (85°F)</td>
</tr>
</tbody>
</table>
Signal and Power Requirements

**I/O Channel**

<table>
<thead>
<tr>
<th>Connector</th>
<th>Burndy, FT72A-3, or equivalent</th>
</tr>
</thead>
</table>
| Output Signal Levels | Active: 3.8 V dc (minimum)  
|                 | Inactive: 0.4 V dc (maximum)         |
| Input Signal Levels  | Active: 3.4 to 5.25 V dc  
|                  | Inactive: -0.5 to +0.8 V dc       |
| Voltage and current | +5 V dc ± 5%, 2 mA  
|                   | +Adapter Power (+9.2 to +16 V dc),  
|                 | 2.5 watts                         |

**Direct Drive**

<table>
<thead>
<tr>
<th>Connector</th>
<th>Molex, PAX-70052, or equivalent</th>
</tr>
</thead>
<tbody>
<tr>
<td>Output Signal Levels</td>
<td>TTL compatible</td>
</tr>
</tbody>
</table>

**Composite Video**

<table>
<thead>
<tr>
<th>Connector</th>
<th>Standard phonograph jack</th>
</tr>
</thead>
</table>
| Output Signal Levels | 1 volt peak-to-peak biased at 0.7 volt  
|                  | with a 75-ohm load                   |
RF Modulator

<table>
<thead>
<tr>
<th>Connector</th>
<th>Molex, PAX-70052, or equivalent</th>
</tr>
</thead>
<tbody>
<tr>
<td>Output Signal</td>
<td>1 volt peak-to-peak biased at 0.7 volt</td>
</tr>
<tr>
<td>Levels</td>
<td>with a 75 ohm load</td>
</tr>
</tbody>
</table>

Operating Characteristics

<table>
<thead>
<tr>
<th>Video Interface</th>
<th>Composite, direct drive, and RF modulator</th>
</tr>
</thead>
<tbody>
<tr>
<td>Display Modes</td>
<td>Alphanumeric</td>
</tr>
<tr>
<td></td>
<td>Low resolution</td>
</tr>
<tr>
<td></td>
<td>High resolution</td>
</tr>
<tr>
<td></td>
<td>Graphics (all points addressable)</td>
</tr>
<tr>
<td>Characters</td>
<td>Generated from a ROM character generator</td>
</tr>
<tr>
<td>Bandwidth</td>
<td>7 or 14 MHz, depending on mode of operation</td>
</tr>
<tr>
<td>Vertical Drive</td>
<td>60 Hz, progressive raster scan</td>
</tr>
<tr>
<td>Horizontal Drive</td>
<td>15.75 KHz, positive level</td>
</tr>
</tbody>
</table>
## Monochrome Display

### Physical Description

<p>| | |</p>
<table>
<thead>
<tr>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>Width</td>
<td>244 mm (9.6 inches)</td>
</tr>
<tr>
<td>Length</td>
<td>260 mm (10.2 inches)</td>
</tr>
<tr>
<td>Height</td>
<td>187 mm (7.4 inches)</td>
</tr>
<tr>
<td>Weight</td>
<td>7.2 kilograms (15.8 pounds)</td>
</tr>
</tbody>
</table>

### Environmental

<table>
<thead>
<tr>
<th>Temperature</th>
<th>System On:</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>10.0 to 40.6°C (50 to 105°F)</td>
</tr>
<tr>
<td></td>
<td>Storage or Shipping:</td>
</tr>
<tr>
<td></td>
<td>−40.0 to +60.0°C (−40 to +140°F)</td>
</tr>
<tr>
<td>Relative Humidity</td>
<td>Operating: 8 to 80%</td>
</tr>
<tr>
<td></td>
<td>Storage: 5 to 100%</td>
</tr>
<tr>
<td>Maximum Wet Bulb</td>
<td>27.6°C (80°F)</td>
</tr>
</tbody>
</table>
Signal and Power Requirements

**AC Input**

<table>
<thead>
<tr>
<th>Plug</th>
<th>Standard, 3-prong</th>
</tr>
</thead>
<tbody>
<tr>
<td>Voltage</td>
<td>+100 to +120 V ac, 50/60 Hz</td>
</tr>
</tbody>
</table>

**Video Input**

<table>
<thead>
<tr>
<th>Connector</th>
<th>Standard phonograph jack</th>
</tr>
</thead>
<tbody>
<tr>
<td>Input Signal Level</td>
<td>1 volt peak-to-peak</td>
</tr>
</tbody>
</table>

**Operating Characteristics**

<table>
<thead>
<tr>
<th>Video Interface</th>
<th>Composite</th>
</tr>
</thead>
<tbody>
<tr>
<td>Input Impedance</td>
<td>75 ohm</td>
</tr>
<tr>
<td>Display</td>
<td>160 mm by 120 mm image area, P31 green phosphor, direct etched</td>
</tr>
</tbody>
</table>
**Color Display**

**Physical Description**

<table>
<thead>
<tr>
<th>Feature</th>
<th>Measurement</th>
</tr>
</thead>
<tbody>
<tr>
<td>Width</td>
<td>344 mm (13.5 inches)</td>
</tr>
<tr>
<td>Length</td>
<td>376 mm (14.8 inches)</td>
</tr>
<tr>
<td>Height</td>
<td>300 mm (11.8 inches)</td>
</tr>
<tr>
<td>Weight</td>
<td>16.8 kilograms (37 pounds)</td>
</tr>
</tbody>
</table>

**Environmental**

<table>
<thead>
<tr>
<th>Feature</th>
<th>Specification</th>
</tr>
</thead>
<tbody>
<tr>
<td>System On:</td>
<td>10.0 to 40.6°C (50 to 105°F)</td>
</tr>
<tr>
<td>Storage or Shipping:</td>
<td>−40.0 to +60.0°C (−40 to +140°F)</td>
</tr>
<tr>
<td>Relative Humidity</td>
<td>Operating: 8 to 80%</td>
</tr>
<tr>
<td></td>
<td>Storage: 5 to 100%</td>
</tr>
<tr>
<td>Maximum Wet Bulb</td>
<td>26.7°C (80°F)</td>
</tr>
</tbody>
</table>

**Signal and Power Requirements**

**AC Input**

<table>
<thead>
<tr>
<th>Feature</th>
<th>Specification</th>
</tr>
</thead>
<tbody>
<tr>
<td>Plug</td>
<td>Standard, 3-prong</td>
</tr>
<tr>
<td>Voltage</td>
<td>+100 to +120 V ac, 50/60 Hz</td>
</tr>
</tbody>
</table>
### Video Input

<table>
<thead>
<tr>
<th>Connector</th>
<th>18 pin, D-shell</th>
</tr>
</thead>
<tbody>
<tr>
<td>Input Signal Level</td>
<td>TTL compatible</td>
</tr>
</tbody>
</table>

### Operating Characteristics

<table>
<thead>
<tr>
<th>Video Interface</th>
<th>Direct drive</th>
</tr>
</thead>
<tbody>
<tr>
<td>Display</td>
<td>256 mm by 176 mm image area, direct etched</td>
</tr>
<tr>
<td>Audio Capability</td>
<td>Built-in speaker</td>
</tr>
</tbody>
</table>
Internal Modem

Physical Description

*Card 1*

<table>
<thead>
<tr>
<th>Width</th>
<th>133 mm (5.24 inches)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Length</td>
<td>76 mm (3.0 inches)</td>
</tr>
<tr>
<td>Height</td>
<td>6 mm (0.24 inches)</td>
</tr>
</tbody>
</table>

*Card 2*

<table>
<thead>
<tr>
<th>Width</th>
<th>92.7 mm (3.65 inches)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Length</td>
<td>35 mm (1.38 inches)</td>
</tr>
<tr>
<td>Height</td>
<td>16.5 mm (0.65 inches)</td>
</tr>
</tbody>
</table>
Signal and Power Requirements

<table>
<thead>
<tr>
<th>Connector to system</th>
<th>30 pin, vertical header</th>
</tr>
</thead>
<tbody>
<tr>
<td>Connector to line</td>
<td>One 2-pin RJ-11C modular phone jack</td>
</tr>
</tbody>
</table>
| Output Signal Levels (to system) | Active: 3.5 V dc (minimum)  
Inactive: 0.4 V dc (maximum) |
| Input Signal Levels (from system) | Active: 2.0 to 5.25 V dc  
Inactive: −0.5 to +0.8 V dc |
| Voltage and current | +5 V dc ± 5%  
−13 V dc ± 20% |

Operating Characteristics

<table>
<thead>
<tr>
<th>Protocol</th>
<th>Bell 212A</th>
</tr>
</thead>
<tbody>
<tr>
<td>Data rate</td>
<td>1200 bits per second QPSK and 110/300 bits per second FSK</td>
</tr>
<tr>
<td>Command set</td>
<td>Enhanced IBM PCjr Internal Modem command set</td>
</tr>
<tr>
<td>Dialing</td>
<td>Pulse or tone</td>
</tr>
</tbody>
</table>
AC Adapter (100–240 V ac)

Physical Description

<table>
<thead>
<tr>
<th>Width</th>
<th>165 mm (6.6 inches)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Length</td>
<td>80 mm (3.2 inches)</td>
</tr>
<tr>
<td>Height</td>
<td>55 mm (2.2 inches)</td>
</tr>
</tbody>
</table>

Environmental

<table>
<thead>
<tr>
<th>Temperature</th>
<th>System On: 0.6 to 40.6°C (33 to 105°F)</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Storage or Shipping: −40 to +60°C (−40 to 140°F)</td>
</tr>
<tr>
<td>Relative Humidity</td>
<td>Operating: 5 to 95%</td>
</tr>
<tr>
<td></td>
<td>Storage or Shipping: 5 to 100%</td>
</tr>
</tbody>
</table>

Power Requirements

<table>
<thead>
<tr>
<th>Input</th>
<th>90 to 265 V ac, 47 to 63 Hz</th>
</tr>
</thead>
<tbody>
<tr>
<td>Output</td>
<td>14 to 16 V dc, 40 watts</td>
</tr>
</tbody>
</table>
## Battery Charger

<table>
<thead>
<tr>
<th>Input</th>
<th>104 to 127 V ac, 47 to 63 Hz</th>
</tr>
</thead>
<tbody>
<tr>
<td>Output</td>
<td>14 to 16 V dc, 3 watts</td>
</tr>
</tbody>
</table>

## Automobile Adapter

<table>
<thead>
<tr>
<th>Input</th>
<th>10 to 16 V dc</th>
</tr>
</thead>
<tbody>
<tr>
<td>Note: Typically, a 13.0 V dc input to the automobile power adapter is required to charge the internal battery pack to 100% of the rated capacity. A nominal 12.0 V dc input only charges the battery pack to 40% of the rated capacity.</td>
<td></td>
</tr>
<tr>
<td>Output</td>
<td>9.2 to 16 V dc, 38 watts</td>
</tr>
</tbody>
</table>
Appendix C. Logic Timing Diagrams

I/O Interface, Processor Memory Read ............... C-4
Interface, Processor Memory Write ................ C-6
I/O Interface, DMA from Memory to I/O ............. C-8
I/O Interface, DMA from I/O to Memory ............. C-10
I/O Interface, Processor I/O Read .................. C-12
I/O Interface, Processor I/O Write ................. C-14
Memory Interface, Processor Read .................. C-16
Memory Interface, Processor Write ................. C-18
Memory Interface, DMA Read ......................... C-20
Memory Interface, DMA Write ...................... C-22
LCD Interface .......................................... C-24
Notes:

C-2 Logic Timing Diagrams
Most of the logic components used on the IBM PC Convertible are packaged to conserve power, space, and weight. Because of this, traditional logic diagrams, such as those used on the other IBM Personal Computer products, have little value for the IBM PC Convertible. Instead of logic diagrams, detailed timing diagrams are provided here. These timing diagrams, when used with the hardware and programming interface information provided elsewhere in this manual, provide the information required to design products capable of operating with the IBM PC Convertible.
## I/O Interface, Processor Memory Read

<table>
<thead>
<tr>
<th>Interface Signal</th>
<th>Reference</th>
<th>Timing (nanoseconds)</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td>Minimum</td>
</tr>
<tr>
<td>CPU Clock *</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Cycle time</td>
<td>1</td>
<td>210</td>
</tr>
<tr>
<td>Low time</td>
<td></td>
<td>128</td>
</tr>
<tr>
<td>High time</td>
<td></td>
<td>69</td>
</tr>
<tr>
<td>Sleep Clock</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Delay from 'clock'</td>
<td>2</td>
<td>0</td>
</tr>
<tr>
<td>Address Latch Enable</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Pulse duration</td>
<td>3</td>
<td>70</td>
</tr>
<tr>
<td>Inactive to 'clock'</td>
<td>4</td>
<td>0</td>
</tr>
<tr>
<td>Active from 'clock'</td>
<td>5</td>
<td>--</td>
</tr>
<tr>
<td>Memory Read</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Delay from 'clock'</td>
<td>6</td>
<td>-3</td>
</tr>
<tr>
<td>Active from 'address latch enable'</td>
<td>7</td>
<td>2</td>
</tr>
<tr>
<td>Data Enable</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Active from 'clock'</td>
<td>8</td>
<td>-2</td>
</tr>
<tr>
<td>Inactive from 'clock'</td>
<td>9</td>
<td>0</td>
</tr>
<tr>
<td>Address/Data Bits 0-7</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Valid from 'clock'</td>
<td>10</td>
<td>10</td>
</tr>
<tr>
<td>Hold time</td>
<td>11</td>
<td>15</td>
</tr>
<tr>
<td>Float delay</td>
<td>12</td>
<td>--</td>
</tr>
<tr>
<td>Read data setup time</td>
<td>13</td>
<td>90</td>
</tr>
<tr>
<td>Hold time</td>
<td>14</td>
<td>5</td>
</tr>
<tr>
<td>Read data float delay</td>
<td>15</td>
<td>--</td>
</tr>
<tr>
<td>Address Bits 8-15</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Valid from 'clock'</td>
<td>16</td>
<td>10</td>
</tr>
<tr>
<td>Bits 8 and 9</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Bits 10-13</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Bits 14 and 15</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Address Bits 16-19</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Valid from 'clock'</td>
<td>17</td>
<td>10</td>
</tr>
<tr>
<td>Hold time</td>
<td>18</td>
<td>10</td>
</tr>
</tbody>
</table>

* This signal is not present at this interface. These timings are included for reference only.

Figure C-1 (Part 1 of 2). I/O Interface, Processor Memory Read

---

C-4 Logic Timing Diagrams
Figure C-1 (Part 2 of 2). I/O Interface, Processor Memory Read
# Interface, Processor Memory Write

<table>
<thead>
<tr>
<th>Interface Signal</th>
<th>Reference</th>
<th>Minimum (nanoseconds)</th>
<th>Maximum (nanoseconds)</th>
</tr>
</thead>
<tbody>
<tr>
<td>CPU Clock *</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Cycle time</td>
<td>1</td>
<td>210</td>
<td>210</td>
</tr>
<tr>
<td>Low time</td>
<td></td>
<td>128</td>
<td>141</td>
</tr>
<tr>
<td>High time</td>
<td></td>
<td>69</td>
<td>82</td>
</tr>
<tr>
<td>Sleep Clock</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Delay from 'clock'</td>
<td>2</td>
<td>0</td>
<td>14</td>
</tr>
<tr>
<td>Address Latch Enable</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Pulse duration</td>
<td>3</td>
<td>70</td>
<td>259</td>
</tr>
<tr>
<td>Inactive to 'clock'</td>
<td>4</td>
<td>0</td>
<td>33</td>
</tr>
<tr>
<td>Active from 'clock'</td>
<td>5</td>
<td>--</td>
<td>114</td>
</tr>
<tr>
<td>Memory Write</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Delay from 'clock'</td>
<td>6</td>
<td>-9</td>
<td>40</td>
</tr>
<tr>
<td>Active from 'address latch enable'</td>
<td>7</td>
<td>0</td>
<td>--</td>
</tr>
<tr>
<td>Data Enable</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Active from 'clock'</td>
<td>8</td>
<td>3</td>
<td>110</td>
</tr>
<tr>
<td>Inactive from 'clock'</td>
<td>9</td>
<td>0</td>
<td>81</td>
</tr>
<tr>
<td>Address/Data Bits 0-7</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Valid from 'clock'</td>
<td>10</td>
<td>10</td>
<td>145</td>
</tr>
<tr>
<td>Hold time</td>
<td>11</td>
<td>15</td>
<td>--</td>
</tr>
<tr>
<td>Data delay</td>
<td>12</td>
<td>15</td>
<td>175</td>
</tr>
<tr>
<td>Write data hold time</td>
<td>13</td>
<td>15</td>
<td>--</td>
</tr>
<tr>
<td>Address/Data Bits 8-15</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Valid from 'clock'</td>
<td>14</td>
<td>10</td>
<td>--</td>
</tr>
<tr>
<td>Bits 8 and 9</td>
<td></td>
<td>110</td>
<td></td>
</tr>
<tr>
<td>Bits 10-13</td>
<td></td>
<td>105</td>
<td></td>
</tr>
<tr>
<td>Bits 14 and 15</td>
<td></td>
<td>100</td>
<td></td>
</tr>
<tr>
<td>Address Bits 16-19</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Valid from 'clock'</td>
<td>15</td>
<td>10</td>
<td>85</td>
</tr>
<tr>
<td>Hold time</td>
<td>16</td>
<td>10</td>
<td>--</td>
</tr>
</tbody>
</table>

* This signal is not present at this interface. These timings are included for reference only.

Figure C-2 (Part 1 of 2). Interface, Processor Memory Write

C-6 Logic Timing Diagrams
Figure C-2 (Part 2 of 2). Interface, Processor Memory Write
### I/O Interface, DMA from Memory to I/O

<table>
<thead>
<tr>
<th>Interface Signal</th>
<th>Reference</th>
<th>Timing (nanoseconds)</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td>Minimum</td>
</tr>
<tr>
<td>Sleep Clock *</td>
<td></td>
<td>210</td>
</tr>
<tr>
<td>Cycle time</td>
<td>1</td>
<td></td>
</tr>
<tr>
<td>Low time</td>
<td>128</td>
<td></td>
</tr>
<tr>
<td>High time</td>
<td>69</td>
<td></td>
</tr>
<tr>
<td>DMA Acknowledge</td>
<td></td>
<td>174</td>
</tr>
<tr>
<td>Active from 'sleep clock'</td>
<td>2</td>
<td></td>
</tr>
<tr>
<td>Inactive from 'sleep clock'</td>
<td>3</td>
<td></td>
</tr>
<tr>
<td>Address Latch Enable</td>
<td></td>
<td>61</td>
</tr>
<tr>
<td>Pulse duration</td>
<td>4</td>
<td></td>
</tr>
<tr>
<td>Inactive to 'clock'</td>
<td>5</td>
<td></td>
</tr>
<tr>
<td>I/O Write</td>
<td></td>
<td>121</td>
</tr>
<tr>
<td>Active to 'sleep clock'</td>
<td>6</td>
<td></td>
</tr>
<tr>
<td>Inactive from 'sleep clock'</td>
<td>7</td>
<td></td>
</tr>
<tr>
<td>Memory Read</td>
<td></td>
<td>141</td>
</tr>
<tr>
<td>Active to 'I/O write'</td>
<td>8</td>
<td></td>
</tr>
<tr>
<td>Active from 'sleep clock'</td>
<td>9</td>
<td></td>
</tr>
<tr>
<td>Active from 'address latch enable'</td>
<td>10</td>
<td></td>
</tr>
<tr>
<td>Inactive</td>
<td>11</td>
<td></td>
</tr>
<tr>
<td>inactive from 'I/O write'</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Data Enable</td>
<td></td>
<td>153</td>
</tr>
<tr>
<td>Active from 'I/O write'</td>
<td>12</td>
<td></td>
</tr>
<tr>
<td>inactive from 'I/O write'</td>
<td>13</td>
<td></td>
</tr>
<tr>
<td>Address/Data Bits 0-7</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Address setup time</td>
<td>14</td>
<td>108</td>
</tr>
<tr>
<td>Address hold time</td>
<td>15</td>
<td>16</td>
</tr>
<tr>
<td>Float before 'data enable'</td>
<td>16</td>
<td>100</td>
</tr>
<tr>
<td>Data setup time **</td>
<td>17</td>
<td>100</td>
</tr>
<tr>
<td>Data hold time **</td>
<td>18</td>
<td>20</td>
</tr>
<tr>
<td>Float delay from first 'memory read'</td>
<td>19</td>
<td></td>
</tr>
<tr>
<td>or 'data enable' ***</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Address Bits 8-19</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Valid from 'sleep clock'</td>
<td>20</td>
<td>0</td>
</tr>
<tr>
<td>Invalid from 'memory read'</td>
<td>21</td>
<td></td>
</tr>
<tr>
<td>inactive</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

* This signal is not present at this interface. These timings are included for reference only.
** Main memory access.
*** Block or demand mode transfer, 160 nanoseconds for single-mode transfer.

Figure C-3 (Part 1 of 2). I/O Interface, DMA from Memory to I/O

C-8 Logic Timing Diagrams
Figure C-3 (Part 2 of 2). I/O Interface, DMA from Memory to I/O
## I/O Interface, DMA from I/O to Memory

<table>
<thead>
<tr>
<th>Interface Signal</th>
<th>Reference</th>
<th>Timing (nanoseconds)</th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td>Minimum</td>
<td></td>
<td>Maximum</td>
</tr>
<tr>
<td>Sleep Clock *</td>
<td>1</td>
<td>210</td>
<td>210</td>
<td></td>
</tr>
<tr>
<td>Cycle time</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Low time</td>
<td>2</td>
<td>128</td>
<td>141</td>
<td></td>
</tr>
<tr>
<td>High time</td>
<td>3</td>
<td>69</td>
<td>82</td>
<td></td>
</tr>
<tr>
<td>DMA Acknowledge</td>
<td>4</td>
<td>61</td>
<td>--</td>
<td></td>
</tr>
<tr>
<td>Active from 'sleep clock'</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>inactive from 'sleep clock'</td>
<td>5</td>
<td>0</td>
<td>174</td>
<td></td>
</tr>
<tr>
<td>Address Latch Enable</td>
<td>6</td>
<td>0</td>
<td>115</td>
<td></td>
</tr>
<tr>
<td>Pulse duration</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>inactive to 'clock'</td>
<td>7</td>
<td>0</td>
<td>113</td>
<td></td>
</tr>
<tr>
<td>Memory Write</td>
<td>8</td>
<td>17</td>
<td>141</td>
<td></td>
</tr>
<tr>
<td>Active from 'sleep clock'</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>inactive from 'sleep clock'</td>
<td>9</td>
<td>0</td>
<td>193</td>
<td></td>
</tr>
<tr>
<td>Active from 'I/O read'</td>
<td>10</td>
<td>56</td>
<td>--</td>
<td></td>
</tr>
<tr>
<td>I/O Read</td>
<td>11</td>
<td>97</td>
<td>--</td>
<td></td>
</tr>
<tr>
<td>Data Enable</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Active from 'memory write'</td>
<td>12</td>
<td>11</td>
<td>143</td>
<td></td>
</tr>
<tr>
<td>invalid from 'memory write'</td>
<td>13</td>
<td>11</td>
<td>143</td>
<td></td>
</tr>
<tr>
<td>Address/Data Bits 0-7</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Address setup time</td>
<td>14</td>
<td>108</td>
<td>--</td>
<td></td>
</tr>
<tr>
<td>Address hold time</td>
<td>15</td>
<td>16</td>
<td>--</td>
<td></td>
</tr>
<tr>
<td>Float before 'data enable'</td>
<td>16</td>
<td>100</td>
<td>--</td>
<td></td>
</tr>
<tr>
<td>Data setup time **</td>
<td>17</td>
<td>180</td>
<td>--</td>
<td></td>
</tr>
<tr>
<td>Data hold time **</td>
<td>18</td>
<td>53</td>
<td>--</td>
<td></td>
</tr>
<tr>
<td>Float delay from first 'I/O read' or 'data enable' **</td>
<td>19</td>
<td>--</td>
<td>64</td>
<td></td>
</tr>
<tr>
<td>Address Bits 8-19</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Valid from 'sleep clock'</td>
<td>20</td>
<td>0</td>
<td>233</td>
<td></td>
</tr>
<tr>
<td>Invalid from 'I/O read' inactive</td>
<td>21</td>
<td>69</td>
<td>--</td>
<td></td>
</tr>
</tbody>
</table>

* This signal is not present at this interface. These timings are included for reference only.

** Main memory requirement.

*** Block or demand mode transfer. 160 nanoseconds for single-mode transfer.

Figure C-4 (Part 1 of 2). I/O Interface, DMA from I/O to Memory

C-10 Logic Timing Diagrams
Figure C-4 (Part 2 of 2). I/O Interface, DMA from I/O to Memory
## I/O Interface, Processor I/O Read

<table>
<thead>
<tr>
<th>Interface Signal</th>
<th>Reference</th>
<th>Timing (nanoseconds)</th>
<th>Minimum</th>
<th>Maximum</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>CPU Clock</strong></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Cycle time</td>
<td>1</td>
<td>210</td>
<td>210</td>
<td></td>
</tr>
<tr>
<td>Low time</td>
<td></td>
<td>128</td>
<td>141</td>
<td></td>
</tr>
<tr>
<td>High time</td>
<td></td>
<td>69</td>
<td>82</td>
<td></td>
</tr>
<tr>
<td><strong>Sleep Clock</strong></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Delay from 'clock'</td>
<td>2</td>
<td>0</td>
<td>14</td>
<td></td>
</tr>
<tr>
<td><strong>Address Latch Enable</strong></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Pulse duration</td>
<td>3</td>
<td>70</td>
<td>259</td>
<td></td>
</tr>
<tr>
<td>Inactive to 'clock'</td>
<td>4</td>
<td>0</td>
<td>33</td>
<td></td>
</tr>
<tr>
<td>Active from 'clock'</td>
<td>5</td>
<td>--</td>
<td>114</td>
<td></td>
</tr>
<tr>
<td><strong>I/O Read</strong></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Delay from 'clock'</td>
<td>6</td>
<td>-4</td>
<td>54</td>
<td></td>
</tr>
<tr>
<td>Active from 'address latch enable'</td>
<td>7</td>
<td>2</td>
<td>--</td>
<td></td>
</tr>
<tr>
<td>Inactive</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td><strong>Data Enable</strong></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Active from 'clock'</td>
<td>8</td>
<td>-2</td>
<td>66</td>
<td></td>
</tr>
<tr>
<td>Inactive from 'clock'</td>
<td>9</td>
<td>0</td>
<td>80</td>
<td></td>
</tr>
<tr>
<td><strong>Address/Data Bits 0-7</strong></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Valid from 'clock'</td>
<td>10</td>
<td>10</td>
<td>145</td>
<td></td>
</tr>
<tr>
<td>Hold time</td>
<td>11</td>
<td>15</td>
<td>--</td>
<td></td>
</tr>
<tr>
<td>Float delay</td>
<td>12</td>
<td>--</td>
<td>113</td>
<td></td>
</tr>
<tr>
<td>Read data setup time</td>
<td>13</td>
<td>90</td>
<td>--</td>
<td></td>
</tr>
<tr>
<td>Read data hold time</td>
<td>14</td>
<td>5</td>
<td>--</td>
<td></td>
</tr>
<tr>
<td>Read data float delay</td>
<td>15</td>
<td>--</td>
<td>0</td>
<td></td>
</tr>
<tr>
<td><strong>Address Bits 8-15</strong></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Valid from 'clock'</td>
<td>16</td>
<td>--</td>
<td>110</td>
<td></td>
</tr>
<tr>
<td>Bits 8 and 9</td>
<td></td>
<td></td>
<td>105</td>
<td></td>
</tr>
<tr>
<td>Bits 10-13</td>
<td></td>
<td></td>
<td>100</td>
<td></td>
</tr>
<tr>
<td>Bits 14 and 15</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

* This signal is not present at this interface. These timings are included for reference only.

Figure C-5 (Part 1 of 2). I/O Interface, Processor I/O Read

C-12 Logic Timing Diagrams
Figure C-5 (Part 2 of 2). I/O Interface, Processor I/O Read
### I/O Interface, Processor I/O Write

<table>
<thead>
<tr>
<th>Interface Signal</th>
<th>Reference</th>
<th>Timing (nanoseconds)</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td>Minimum</td>
</tr>
<tr>
<td>CPU Clock *</td>
<td></td>
<td>210</td>
</tr>
<tr>
<td>Cycle time</td>
<td>1</td>
<td>128</td>
</tr>
<tr>
<td>Low time</td>
<td></td>
<td>69</td>
</tr>
<tr>
<td>High time</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Sleep Clock</td>
<td></td>
<td>2</td>
</tr>
<tr>
<td>Delay from 'clock'</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Address Latch Enable</td>
<td></td>
<td>3</td>
</tr>
<tr>
<td>Pulse duration</td>
<td>4</td>
<td>0</td>
</tr>
<tr>
<td>Inactive to 'clock'</td>
<td>5</td>
<td>--</td>
</tr>
<tr>
<td>Active from 'clock'</td>
<td></td>
<td></td>
</tr>
<tr>
<td>I/O Write</td>
<td></td>
<td>6</td>
</tr>
<tr>
<td>Delay from 'clock'</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Inactive from 'address latch enable'</td>
<td>7</td>
<td>2</td>
</tr>
<tr>
<td>Data Enable</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Active from 'clock'</td>
<td>8</td>
<td>3</td>
</tr>
<tr>
<td>Inactive from 'clock'</td>
<td>9</td>
<td>0</td>
</tr>
<tr>
<td>Address/Data Bits 0-7</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Valid from 'clock'</td>
<td>10</td>
<td>10</td>
</tr>
<tr>
<td>Hold time</td>
<td>11</td>
<td>15</td>
</tr>
<tr>
<td>Data delay</td>
<td>12</td>
<td>15</td>
</tr>
<tr>
<td>Write data hold time</td>
<td>13</td>
<td>15</td>
</tr>
<tr>
<td>Address Bits 8-15</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Valid from 'clock'</td>
<td>14</td>
<td>--</td>
</tr>
<tr>
<td>Bits 8 and 9</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Bits 10-13</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Bits 14 and 15</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

* This signal is not present at this interface. These timings are included for reference only.

Figure C-6 (Part 1 of 2). I/O Interface, Processor I/O Write
Figure C-6 (Part 2 of 2). I/O Interface, Processor I/O Write
## Memory Interface, Processor Read

<table>
<thead>
<tr>
<th>Interface Signal</th>
<th>Reference</th>
<th>Timing (nanoseconds)</th>
<th>Minimum</th>
<th>Maximum</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>CPU Clock *</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Cycle time</td>
<td>1</td>
<td>210</td>
<td>210</td>
<td></td>
</tr>
<tr>
<td>Low time</td>
<td></td>
<td>128</td>
<td>141</td>
<td></td>
</tr>
<tr>
<td>High time</td>
<td></td>
<td>69</td>
<td>82</td>
<td></td>
</tr>
<tr>
<td>Address Latch Enable</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Pulse duration</td>
<td>2</td>
<td>70</td>
<td>259</td>
<td></td>
</tr>
<tr>
<td>Inactive to 'clock'</td>
<td>3</td>
<td>0</td>
<td>33</td>
<td></td>
</tr>
<tr>
<td>Active from 'clock'</td>
<td>4</td>
<td>--</td>
<td>114</td>
<td></td>
</tr>
<tr>
<td>Data Enable</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Active from 'clock'</td>
<td>5</td>
<td>-2</td>
<td>56</td>
<td></td>
</tr>
<tr>
<td>Inactive from 'clock'</td>
<td>6</td>
<td>0</td>
<td>73</td>
<td></td>
</tr>
<tr>
<td>Address/Data Bits 0-7</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Valid from 'clock'</td>
<td>7</td>
<td>10</td>
<td>90</td>
<td></td>
</tr>
<tr>
<td>Hold time</td>
<td>8</td>
<td>10</td>
<td>--</td>
<td></td>
</tr>
<tr>
<td>Float delay</td>
<td>9</td>
<td>--</td>
<td>80</td>
<td></td>
</tr>
<tr>
<td>Read data setup time</td>
<td>10</td>
<td>30</td>
<td>--</td>
<td></td>
</tr>
<tr>
<td>Read data hold time</td>
<td>11</td>
<td>10</td>
<td>--</td>
<td></td>
</tr>
<tr>
<td>Read data float delay</td>
<td>12</td>
<td>--</td>
<td>175</td>
<td></td>
</tr>
<tr>
<td>Address Bits 8-16</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Valid from 'clock'</td>
<td>13</td>
<td>10</td>
<td>--</td>
<td></td>
</tr>
<tr>
<td>Bits 8 and 9</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Bits 10-13</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Bits 14 and 15</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Bit 16</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Card Select</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Active from 'clock'</td>
<td>14</td>
<td>3</td>
<td>106</td>
<td></td>
</tr>
</tbody>
</table>

* This signal is not present at this interface. These timings are included for reference only.

Figure C-7 (Part 1 of 2). Memory Interface, Processor Read
Figure C-7 (Part 2 of 2). Memory Interface, Processor Read
## Memory Interface, Processor Write

<table>
<thead>
<tr>
<th>Interface Signal</th>
<th>Reference</th>
<th>Timing (nanoseconds)</th>
<th>Minimum</th>
<th>Maximum</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>CPU Clock</strong></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Cycle time</td>
<td>1</td>
<td>210</td>
<td>210</td>
<td></td>
</tr>
<tr>
<td>Low time</td>
<td></td>
<td>128</td>
<td>141</td>
<td></td>
</tr>
<tr>
<td>High time</td>
<td></td>
<td>69</td>
<td>82</td>
<td></td>
</tr>
<tr>
<td><strong>Address Latch Enable</strong></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Pulse duration</td>
<td>2</td>
<td>70</td>
<td>259</td>
<td></td>
</tr>
<tr>
<td>Inactive to 'clock'</td>
<td>3</td>
<td>0</td>
<td>33</td>
<td></td>
</tr>
<tr>
<td>Active from 'clock'</td>
<td>4</td>
<td>--</td>
<td>114</td>
<td></td>
</tr>
<tr>
<td><strong>Memory Write</strong></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Delay from 'clock'</td>
<td>5</td>
<td>1</td>
<td>64</td>
<td></td>
</tr>
<tr>
<td>Active from 'address latch enable'</td>
<td>6</td>
<td>3</td>
<td>--</td>
<td></td>
</tr>
<tr>
<td>Inactive</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td><strong>Data Enable</strong></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Active from 'clock'</td>
<td>7</td>
<td>3</td>
<td>102</td>
<td></td>
</tr>
<tr>
<td>Inactive from 'clock'</td>
<td>8</td>
<td>0</td>
<td>73</td>
<td></td>
</tr>
<tr>
<td><strong>Address/Data Bits 0-7</strong></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Valid from 'clock'</td>
<td>9</td>
<td>10</td>
<td>90</td>
<td></td>
</tr>
<tr>
<td>Hold time</td>
<td>10</td>
<td>10</td>
<td>--</td>
<td></td>
</tr>
<tr>
<td>Data delay</td>
<td>11</td>
<td>10</td>
<td>110</td>
<td></td>
</tr>
<tr>
<td>Write data hold time</td>
<td>12</td>
<td>10</td>
<td>--</td>
<td></td>
</tr>
<tr>
<td><strong>Address Bits 8-16</strong></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Valid from 'clock'</td>
<td>13</td>
<td>10</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Bits 8 and 9</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Bits 10-13</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Bits 14 and 15</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Bit 16</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td><strong>Card Select</strong></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Active from 'clock'</td>
<td>14</td>
<td>-4</td>
<td>94</td>
<td></td>
</tr>
</tbody>
</table>

*This signal is not present at this interface. These timings are included for reference only.*

**Figure C-8 (Part 1 of 2). Memory Interface, Processor Write**

C-18 Logic Timing Diagrams
Figure C-8 (Part 2 of 2). Memory Interface, Processor Write
Memory Interface, DMA Read

<table>
<thead>
<tr>
<th>Interface Signal</th>
<th>Reference</th>
<th>Timing (nanoseconds)</th>
<th>Reference</th>
<th>Timing (nanoseconds)</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td>Minimum</td>
<td>Maximum</td>
<td></td>
</tr>
<tr>
<td>Sleep Clock *</td>
<td></td>
<td>210</td>
<td>210</td>
<td></td>
</tr>
<tr>
<td>Cycle time</td>
<td>1</td>
<td>210</td>
<td>210</td>
<td></td>
</tr>
<tr>
<td>Low time</td>
<td></td>
<td>128</td>
<td>141</td>
<td></td>
</tr>
<tr>
<td>High time</td>
<td></td>
<td>69</td>
<td>82</td>
<td></td>
</tr>
<tr>
<td>Address Latch Enable</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Pulse duration</td>
<td>2</td>
<td>61</td>
<td>3</td>
<td>0</td>
</tr>
<tr>
<td>Inactive to 'sleep clock'</td>
<td>2</td>
<td></td>
<td>3</td>
<td>159</td>
</tr>
<tr>
<td>Card Select</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Delay from 'sleep clock'</td>
<td>4</td>
<td>0</td>
<td>5</td>
<td>241</td>
</tr>
<tr>
<td>Active from 'address latch enable'</td>
<td>5</td>
<td>70</td>
<td>6</td>
<td>164</td>
</tr>
<tr>
<td>Inactive from 'sleep clock'</td>
<td>6</td>
<td>0</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Data Enable</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Active from 'card select'</td>
<td>7</td>
<td>-14</td>
<td>8</td>
<td>250</td>
</tr>
<tr>
<td>Inactive from 'card select'</td>
<td>8</td>
<td>-25</td>
<td></td>
<td>200</td>
</tr>
<tr>
<td>Address/Data Bits 0-7</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Address setup time</td>
<td>9</td>
<td>48</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Address hold time</td>
<td>10</td>
<td>16</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Address float from 'data enable'</td>
<td>11</td>
<td>61</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Data setup time time</td>
<td>12</td>
<td>120</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Data hold time from first 'card select' or 'data enable'</td>
<td>13</td>
<td>0</td>
<td>14</td>
<td>--</td>
</tr>
<tr>
<td>Float delay from first 'card select' or 'data enable'</td>
<td>14</td>
<td>30</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Address Bits 8-16</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Valid from 'sleep clock'</td>
<td>15</td>
<td>0</td>
<td></td>
<td>223</td>
</tr>
</tbody>
</table>

* This signal is not present at this interface. These timings are included for reference only.

Figure C-9 (Part 1 of 2). Memory Interface, DMA Read
Figure C-9 (Part 2 of 2). Memory Interface, DMA Read
# Memory Interface, DMA Write

<table>
<thead>
<tr>
<th>Interface Signal</th>
<th>Reference</th>
<th>Timing (nanoseconds)</th>
<th>Minimum</th>
<th>Maximum</th>
</tr>
</thead>
<tbody>
<tr>
<td>Sleep Clock *</td>
<td>1</td>
<td>210</td>
<td>210</td>
<td></td>
</tr>
<tr>
<td>Cycle time</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Low time</td>
<td></td>
<td>128</td>
<td>141</td>
<td></td>
</tr>
<tr>
<td>High time</td>
<td></td>
<td>69</td>
<td>82</td>
<td></td>
</tr>
<tr>
<td>Address Latch Enable</td>
<td>2</td>
<td>61</td>
<td>--</td>
<td></td>
</tr>
<tr>
<td>Pulse duration</td>
<td>3</td>
<td></td>
<td>0</td>
<td>159</td>
</tr>
<tr>
<td>Inactive to 'sleep clock'</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Card Select</td>
<td>4</td>
<td>0</td>
<td>168</td>
<td></td>
</tr>
<tr>
<td>Active from 'sleep clock'</td>
<td>5</td>
<td>168</td>
<td>--</td>
<td></td>
</tr>
<tr>
<td>Inactive</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Active from 'address latch enable'</td>
<td>6</td>
<td>0</td>
<td>167</td>
<td></td>
</tr>
<tr>
<td>Inactive from 'sleep clock'</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Memory Write</td>
<td>7</td>
<td>-42</td>
<td>101</td>
<td></td>
</tr>
<tr>
<td>Delay from 'card select'</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Data Enable</td>
<td>8</td>
<td>-40</td>
<td>135</td>
<td></td>
</tr>
<tr>
<td>Active from 'card select'</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Address/Data Bits 0-7</td>
<td>9</td>
<td>48</td>
<td>--</td>
<td></td>
</tr>
<tr>
<td>Address setup time</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Address hold time</td>
<td>10</td>
<td>16</td>
<td>--</td>
<td></td>
</tr>
<tr>
<td>Data setup time to first 'card select' or 'memory write'</td>
<td>11</td>
<td>120</td>
<td>--</td>
<td></td>
</tr>
<tr>
<td>Data hold time</td>
<td>12</td>
<td>0</td>
<td>--</td>
<td></td>
</tr>
<tr>
<td>Address Bits 8-16</td>
<td>13</td>
<td>0</td>
<td>223</td>
<td></td>
</tr>
<tr>
<td>Valid from 'sleep clock'</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

* This signal is not present at this interface. These timings are included for reference only.

Figure C-10 (Part 1 of 2). Memory Interface, DMA Write

C-22 Logic Timing Diagrams
Figure C-10 (Part 2 of 2). Memory Interface, DMA Write
## LCD Interface

<table>
<thead>
<tr>
<th>Interface Signal</th>
<th>Reference</th>
<th>Timing (nanoseconds)</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td>Minimum</td>
</tr>
<tr>
<td><strong>Start Frame</strong></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Setup to 'Y shift clock' Inactive</td>
<td>1</td>
<td>100</td>
</tr>
<tr>
<td><strong>Y Shift Clock</strong></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Pulse duration</td>
<td>2</td>
<td>400</td>
</tr>
<tr>
<td><strong>Latch</strong></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Pulse duration</td>
<td>3</td>
<td>300</td>
</tr>
<tr>
<td>Active to 'X enable clock' Inactive</td>
<td>4</td>
<td>150</td>
</tr>
<tr>
<td><strong>X Enable Clock</strong></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Pulse duration</td>
<td>5</td>
<td>250</td>
</tr>
<tr>
<td>Inactive to 'Latch' Inactive</td>
<td>6</td>
<td>100</td>
</tr>
<tr>
<td><strong>X Shift Clock</strong></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Period</td>
<td>7</td>
<td>480</td>
</tr>
<tr>
<td>Pulse duration</td>
<td>8</td>
<td>240</td>
</tr>
<tr>
<td>Inactive</td>
<td>9</td>
<td>240</td>
</tr>
<tr>
<td>Inactive to 'X enable clock' Inactive</td>
<td>10</td>
<td>70</td>
</tr>
<tr>
<td>Active from 'X enable clock' Inactive</td>
<td>11</td>
<td>100</td>
</tr>
<tr>
<td><strong>Data Bits 0-3</strong></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Setup to 'X shift clock' Inactive</td>
<td>12</td>
<td>100</td>
</tr>
<tr>
<td>Hold time to 'X shift clock' Inactive</td>
<td>13</td>
<td>150</td>
</tr>
<tr>
<td><strong>Frame</strong></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Time from 'latch' Inactive</td>
<td>14</td>
<td>-50</td>
</tr>
</tbody>
</table>

Figure C-11 (Part 1 of 2). LCD Interface

C-24 Logic Timing Diagrams
Figure C-11 (Part 2 of 2). LCD Interface
Notes:
Appendix D. Power-On Self Test
Error Codes

System Unit Failures .............................. D-2
Feature Failures .................................. D-3
Time and Date Errors ............................. D-3
Low Battery Warning .............................. D-3
Audible Signals ................................. D-4
All power-on self test errors are depicted by an icon, an error code, or both. This appendix lists the error codes with a brief meaning.

## System Unit Failures

The following codes are issued for failures within the system unit.

<table>
<thead>
<tr>
<th>Code</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>0101</td>
<td>Interrupt controller failure</td>
</tr>
<tr>
<td>0102</td>
<td>System timer 2 failure</td>
</tr>
<tr>
<td>0103</td>
<td>System timer 0 failure</td>
</tr>
<tr>
<td>0105</td>
<td>DMA controller failure</td>
</tr>
<tr>
<td>0163</td>
<td>Real-time clock not updating</td>
</tr>
<tr>
<td>0164</td>
<td>Memory size miscompare</td>
</tr>
<tr>
<td>0170</td>
<td>LCD not in use when suspended (possible operator error)</td>
</tr>
<tr>
<td>0171</td>
<td>Base 128K checksum failure</td>
</tr>
<tr>
<td>0172</td>
<td>Diskette active when suspended (possible operator error)</td>
</tr>
<tr>
<td>0173</td>
<td>Real-time clock RAM verification error</td>
</tr>
<tr>
<td>0174</td>
<td>LCD configuration changed (possible operator error)</td>
</tr>
<tr>
<td>0201</td>
<td>Memory pattern test failure</td>
</tr>
<tr>
<td>0202</td>
<td>Memory address test failure</td>
</tr>
<tr>
<td>0301 xx</td>
<td>Stuck key detected (xx indicates internal scan code)</td>
</tr>
<tr>
<td>0303</td>
<td>Keyboard controller failure</td>
</tr>
<tr>
<td>0304</td>
<td>Keyboard cable not attached</td>
</tr>
<tr>
<td>0601</td>
<td>Diskette controller or drive failure</td>
</tr>
<tr>
<td>1101</td>
<td>8250 function of modem failure</td>
</tr>
<tr>
<td>1102</td>
<td>Internal modem failure</td>
</tr>
<tr>
<td>5001</td>
<td>LCD buffer failure</td>
</tr>
<tr>
<td>5002</td>
<td>LCD font buffer failure</td>
</tr>
<tr>
<td>5003</td>
<td>LCD controller failure</td>
</tr>
<tr>
<td>5101</td>
<td>Portable printer interface failure</td>
</tr>
</tbody>
</table>

D-2  Error Codes
Feature Failures

The following codes are issued for failures on adapters not within the system unit.

xxxxx ROM  Checksum failure detected on feature ROM (xxxxx indicates feature address in the C0000 to F0000 range)

05xx  Color/graphics adapter failure

1101  RS-232 communications adapter as COM1 failure (if no internal modem feature is installed)

1201  RS-232 communications adapter as COM2 failure (internal modem installed)

Time and Date Errors

Other errors, such as time or date not set, are indicated by an icon.

The time/date not set icon means that the real-time clock value is not valid, because it has never been set or that the system has lost standby power.

Low Battery Warning

The power-on and BIOS routines provide functions to monitor the condition of the battery. If a low-battery condition is detected and external power is not being supplied during power on, the power-on routines issue three short beeps through the speaker and then power-down the system. If a low-battery condition is detected and external power is being supplied during power on, either a low-battery icon is displayed prior to an initial program load (IPL) or two short beeps are issued before an application resumes.
If during normal operation, the low-battery warning is enabled, a low-battery condition is detected, and the system is operating without external power, BIOS issues three short beeps through the speaker, causes the LCD screen to blink off and on at 1-second intervals, and stops all main level processing. This stops all noninterrupt-driven processing; however, interrupt-driven processing continues. When a key on the keyboard is pressed, the screen stops blinking and main-level processing is resumed. If the low-battery condition remains and external power is not supplied, the low-battery warning is repeated every 2 minutes. If keyboard activity does not occur or if external power is not supplied to the system unit within the 2 minutes, BIOS automatically suspends the application and powers off the system in order to conserve power.

If external power is supplied during the warning, the blinking stops and normal processing is restored.

Audible Signals

The power-on routines use the following audible signals:

- One short beep: No errors found
- Two short beeps: Nonfatal errors found
- One long and one short beep: Fatal errors found
- One long and two short beeps: Fatal errors found; LCD controller failed, or, if LCD is detached, CRT display adapter failed
- Three short beeps: Low battery without external power (fatal error)
- No beeps: The power-on self-test routines could not run; speaker failed.
Glossary

µ Prefix micro; 0.000 001.

µs Microsecond; 0.000 001 second.

A. Ampere.

ac. Alternating current.

active high. Designates a signal that has to go high to produce an effect. Synonymous with positive true.

active low. Designates a signal that has to go low to produce an effect. Synonymous with negative true.

adapter. An auxiliary device or unit used to extend the operation of another system.

address bus. One or more conductors used to carry the binary-coded address from the processor throughout the rest of the system.

all points addressable (APA). A mode in which all points of a displayable image can be controlled by the user.

alphabetic. Synonym for alphanumerical.

alphanumeric (A/N). Pertaining to a character set that contains letters, digits, and usually other characters, such as punctuation marks. Synonymous with alphabetic.

alternating current (ac). A current that periodically reverses its direction of flow.

American National Standard Code for Information Exchange (ASCII). The standard code, using a coded character set consisting of 7-bit coded characters (8 bits including parity check), used for information exchange between data processing systems, data communication systems, and associated equipment. The ASCII set consists of control characters and graphic characters.

ampere (A). The basic unit of electric current.

A/N. Alphanumeric.
**analog.** (1) Pertaining to data in the form of continuously variable physical quantities. (2) Contrast with digital.

**AND.** A logic operator having the property that if $P$ is a statement, $Q$ is a statement, $R$ is a statement, ..., then the AND of $P$, $Q$, $R$, ... is true if all statements are true, false if any statement is false.

**AND operation.** The boolean operation whose result has the boolean value 1, if and only if, each operand has the boolean value 1. Synonymous with conjunction.

**APA.** All points addressable.


**asynchronous transmission.** (1) Transmission in which the time of occurrence of the start of each character, or block of characters, is arbitrary; once started, the time of occurrence of each signal representing a bit within a character, or block, has the same relationship to significant instants of a fixed time frame. (2) Transmission in which each information character is individually transmitted (usually timed by the use of start elements and stop elements).

**audio frequencies.** Frequencies that can be heard by the human ear (approximately 15 hertz to 20 000 hertz).

**BASIC.** Beginner's all-purpose symbolic instruction code.

**basic input/output system (BIOS).** The feature of the IBM Personal Computer that provides the level control of the major I/O devices, and relieves the programmer from concern about hardware device characteristics.

**baud.** (1) A unit of signaling speed equal to the number of discrete conditions or signal events per second. For example, one baud equals one bit per second in a train of binary signals, one-half dot cycle per second in Morse code, and one 3-bit value per second in a train of signals each of which can assume one of eight different states. (2) In asynchronous transmission, the unit of modulation rate corresponding to one unit of interval per second; that is, if the duration of the unit...
interval is 20 milliseconds, the modulation rate is 50 baud.

**beginner’s all-purpose symbolic instruction code (BASIC).** A programming language with a small repertoire of commands and a simple syntax, primarily designed for numeric applications.

**binary.** (1) Pertaining to a selection, choice, or condition that has two possible values or states. (2) Pertaining to a fixed radix numeration system having a radix of 2.

**binary digit.** (1) In binary notation, either of the characters 0 or 1. (2) Synonymous with bit.

**binary notation.** Any notation that uses two different characters, usually the binary digits 0 and 1.

**binary synchronous communications (BSC).** A uniform procedure, using a standardized set of control characters and control character sequences for synchronous transmission of binary-coded data between stations.

**BIOS.** Basic input/output system.

**bit.** Synonym for binary digit.

**bits per second (bps).** A unit of measurement representing the number of discrete binary digits transmitted by a device in one second.

**bootstrap.** A technique or device designed to bring itself into a desired state by means of its own action; for example, a machine routine whose first few instructions are sufficient to bring the rest of itself into the computer from an input device.

**bps.** Bits per second.

**BSC.** Binary synchronous communications.

**buffer.** (1) An area of storage that is temporarily reserved for use in performing an input/output operation, into which data is read or from which data is written. Synonymous with I/O area. (2) A portion of storage for temporarily holding input or output data.

**bus.** One or more conductors used for transmitting signals or power.

**byte.** (1) A sequence of eight adjacent binary digits that are operated upon as a unit. (2) A binary character.
operated upon as a unit. (3) The representation of a character.

C. Celsius.

cathode ray tube (CRT). A vacuum tube in which a stream of electrons is projected onto a fluorescent screen producing a luminous spot. The location of the spot can be controlled.

cathode ray tube display (CRT display). (1) A CRT used for displaying data. For example, the electron beam can be controlled to form alphanumeric data by use of a dot matrix. (2) The data display produced by the device as in (1).

CCITT. International Telegraph and Telephone Consultative Committee.

Celsius (C). A temperature scale. Contrast with Fahrenheit (F).

channel. A path along which signals can be sent; for example, data channel, output channel.

character generator. (1) In computer graphics, a functional unit that converts the coded representation of a graphic character into the shape of the character for display. (2) In word processing, the means within equipment for generating visual characters or symbols from coded data.

character set. (1) A finite set of different characters upon which agreement has been reached and that is considered complete for some purpose. (2) A set of unique representations called characters. (3) A defined collection of characters.

characters per second (cps). A standard unit of measurement for the speed at which a printer prints.

CMOS. Complementary metal oxide semiconductor.

code. (1) A set of unambiguous rules specifying the manner in which data may be represented in a discrete form. Synonymous with coding scheme. (2) A set of items, such as abbreviations, representing the members of another set. (3) To represent data or a computer program in a symbolic form that can be accepted by a data processor. (4) Loosely, one or more computer programs, or part of a computer program.

X-4 Glossary
**compile.** (1) To translate a computer program expressed in a problem-oriented language into a computer-oriented language. (2) To prepare a machine-language program from a computer program written in another programming language by making use of the overall logic structure of the program, or generating more than one computer instruction for each symbolic statement, or both, as well as performing the function of an assembler.

**computer program.** A sequence of instructions suitable for processing by a computer.

**configuration.** (1) The arrangement of a computer system or network as defined by the nature, number, and the chief characteristics of its functional units. More specifically, the term configuration may refer to a hardware configuration or a software configuration. (2) The devices and programs that make up a system, subsystem, or network.

**complementary metal oxide semiconductor (CMOS).** A logic circuit family that uses very little power. It works with a wide range of power supply voltages.

**computer.** A functional unit that can perform substantial computation, including numerous arithmetic operations or logic operations, without intervention by a human operator during a run.

**computer instruction code.** A code used to represent the instructions in an instruction set. Synonymous with machine code.

**control character.** A character whose occurrence in a particular context initiates, modifies, or stops a control operation.

**control operation.** An action that affects the recording, processing, transmission, or interpretation of data; for example, starting or stopping a process, carriage return, font change, rewind, and end of transmission.

**cps.** Characters per second.

**CRT.** Cathode ray tube.

**CRT display.** Cathode ray tube display.
CTS. Clear to send. Associated with modem control.

cursor. (1) In computer graphics, a movable marker that is used to indicate a position on a display. (2) A displayed symbol that acts as a marker to help the user locate a point in text, in a system command, or in storage. (3) A movable spot of light on the screen of a display device, usually indicating where the next character is to be entered, replaced, or deleted.

cylinder. (1) The set of all tracks with the same nominal distance from the axis about which the disk rotates. (2) The tracks of a disk storage device that can be accessed without repositioning the access mechanism.

data. (1) A representation of facts, concepts, or instructions in a formalized manner suitable for communication, interpretation, or processing by human or automatic means. (2) Any representations, such as characters or analog quantities, to which meaning is, or might be assigned.

data processing system. A system that performs input, processing, storage, output, and control functions to accomplish a sequence of operations on data.

data transmission. Synonym for transmission.

dB. Decibel.

dBa. Adjusted decibels.

dc. Direct current.

debounce. An electronic means of overcoming the make/break bounce of switches to obtain one smooth change of signal level.

decibel. (1) A unit that expresses the ratio of two power levels on a logarithmic scale. (2) A unit for measuring relative power.

digit. (1) A graphic character that represents an integer; for example, one of the characters 0 to 9. (2) A symbol that represents one of the non-negative integers smaller than the radix. For example, in decimal notation, a digit is one of the characters 0 to 9.
digital. (1) Pertaining to data in the form of digits. (2) Contrast with analog.

direct current (dc). A current that always flows in one direction.

direct memory access (DMA). A method of transferring data between main storage and I/O devices that does not require processor intervention.

disable. To stop the operation of a circuit or device.

disabled. Pertaining to a state of a processing unit that prevents the occurrence of certain types of interruptions. Synonymous with masked.

disk. Loosely, a magnetic disk unit.

disk drive. A mechanism for moving a disk pack and controlling its movements.

diskette. A thin, flexible magnetic disk and a semirigid protective jacket, in which the disk is permanently enclosed. Synonymous with flexible disk.

diskette drive. A mechanism for moving a diskette and controlling its movements.

display. (1) A visual presentation of data. (2) A device for visual presentation of information on any temporary character imaging device. (3) To present data visually. (4) See cathode ray tube display.

display attribute. In computer graphics, a particular property that is assigned to all or part of a display; for example, low intensity, green color, blinking status.

DMA. Direct memory access.

dot matrix. (1) In computer graphics, a two-dimensional pattern of dots used for constructing a display image. This type of matrix can be used to represent characters by dots. (2) In word processing, a pattern of dots used to form characters. This term normally refers to a small section of a set of addressable points; for example, a representation of characters by dots.

dot printer. Synonym for matrix printer.

dot-matrix character generator. In computer graphics, a character
generator that generates character images composed of dots.

disable. To initiate the operation of a circuit or device.

DSR. Data set ready. Associated with modem control.

EPROM. Erasable programmable read-only memory.

eraseable programmable read-only memory (EPROM). A PROM in which the user can erase old information and enter new information.

DTR. In the IBM Personal Computer, data terminal ready. Associated with modem control.

erodable programmable read-only memory (EPROM). A PROM in which the user can erase old information and enter new information.

duplex. (1) In data communication, pertaining to a simultaneous two-way independent transmission in both directions. (2) Contrast with half-duplex.

duty cycle. In the operation of a device, the ratio of on time to idle time. Duty cycle is expressed as a decimal or percentage.

EBCDIC. Extended binary-coded decimal interchange code.

ESD. The escape character.

escape character (ESC). A code extension character used, in some cases, with one or more succeeding characters to indicate by some convention or agreement that the coded representations following the character or the group of characters are to be interpreted according to a different code or according to a different coded character set.

edge connector. A terminal block with a number of contacts attached to the edge of a printed-circuit board to facilitate plugging into a foundation circuit.

eXtended decimal interchange code (EDIC). A set of 256 characters, each represented by eight bits.

EIA. Electronic Industries Association.

Fahrenheit (F). A temperature scale. Contrast with Celsius (C).
falling edge. Synonym for negative-going edge.

FCC. Federal Communications Commission.

FF. The form feed character.

fixed disk. In the IBM Personal Computer, synonym for disk drive.

flag. (1) Any of various types of indicators used for identification. (2) A character that signals the occurrence of some condition, such as the end of a word. (3) Deprecated term for mark.

flexible disk. Synonym for diskette.

font. A family or assortment of characters of a given size and style; for example, 10 point Press Roman medium.

foreground. (1) In multiprogramming, the environment in which high-priority programs are executed. (2) On a color display screen, the characters as opposed to the background.

form feed. (1) Paper movement used to bring an assigned part of a form to the printing position. (2) In word processing, a function that advances the typing position to the same character position on a predetermined line of the next form or page.

form feed character. A control character that causes the print or display position to move to the next predetermined first line on the next form, the next page, or the equivalent.

format. The arrangement or layout of data on a data medium.

g. Gram.

G. (1) Prefix giga: 1 000 000 000. (2) When referring to computer storage capacity, 1 073 741 824. (1 073 741 824 = 2 to the 30th power.)

gate. (1) A combinational logic circuit having one output channel and one or more input channels, such that the output channel state is completely determined by the input channel states. (2) A signal that enables the passage of other signals through a circuit.

Gb. 1 073 741 824 bytes.

giga (G). Prefix 1 000 000 000.
gram (g). A unit of weight (equivalent to 0.035 ounces).

graphic. A symbol produced by a process such as handwriting, drawing, or printing.

graphic character. A character, other than a control character, that is normally represented by a graphic.

half-duplex. (1) In data communication, pertaining to an alternate, one way at a time, independent transmission. (2) Contrast with duplex.

hardware. (1) Physical equipment used in data processing, as opposed to programs, procedures, rules, and associated documentation. (2) Contrast with software.

head. A device that reads, writes, or erases data on a storage medium; for example, a small electromagnet used to read, write, or erase data on a magnetic disk.

hertz (Hz). A unit of frequency equal to one cycle per second.

hex. Common abbreviation for hexadecimal.

hexadecimal. (1) Pertaining to a selection, choice, or condition that has 16 possible different values or states. These values or states are usually symbolized by the ten digits 0 through 9 and the six letters A through F. (2) Pertaining to a fixed radix numeration system having a radix of 16.

high-order position. The leftmost position in a string of characters. See also most-significant digit.

Hz. hertz

image. A fully processed unit of operational data that is ready to be transmitted to a remote unit; when loaded into control storage in the remote unit, the image determines the operations of the unit.

index register. A register whose contents may be used to modify an operand address during the execution of computer instructions.

indicator. (1) A device that may be set into a prescribed state, usually according to the result of a previous process or on the occurrence of a specified condition in the equipment, and that
usually gives a visual or other indication of the existence of the prescribed state, and that may in some cases be used to determine the selection among alternative processes; for example, an overflow indicator. (2) An item of data that may be interrogated to determine whether a particular condition has been satisfied in the execution of a computer program; for example, a switch indicator, an overflow indicator.

inhibited. (1) Pertaining to a state of a processing unit in which certain types of interruptions are not allowed to occur. (2) Pertaining to the state in which a transmission control unit or an audio response unit cannot accept incoming calls on a line.

initialize. To set counters, switches, addresses, or contents of storage to 0 or other starting values at the beginning of, or at prescribed points in, the operation of a computer routine.

input/output (I/O). (1) Pertaining to a device or to a channel that may be involved in an input process, and, at a different time, in an output process. In the English language, “input/output” may be used in place of such terms “input/output data,” “input/output signal,” and “input/output terminals,” when such usage is clear in a given context. (2) Pertaining to a device whose parts can be performing an input process and an output process at the same time. (3) Pertaining to either input or output, or both.

instruction. In a programming language, a meaningful expression that specifies one operation and identifies its operands, if any.

instruction set. The set of instructions of a computer, of a programming language, or of the programming languages in a programming system.

interface. A device that alters or converts actual electrical signals between distinct devices, programs, or systems.

interrupt. (1) A suspension of a process, such as the execution of a computer program, caused by an event external to that process, and performed in such a way that the process can be resumed. (2) In a data transmission, to take an action at a receiving station...
that causes the transmitting station to terminate a transmission. (3) Synonymous with interruption.

I/O. Input/output.

I/O area. Synonym for buffer.

irrecoverable error. An error that makes recovery impossible without the use of recovery techniques external to the computer program or run.

joystick. In computer graphics, a lever that can pivot in all directions and that is used as a locator device.

k. Prefix kilo; 1000.

K. When referring to storage capacity, 1024. (1024 = 2 to the 10th power.)

Kb. 1024 bytes.

kg. Kilogram; 1000 grams.

kHz. Kilohertz; 1000 hertz.

kilo (k). Prefix 1000

kilogram (kg). 1000 grams.

kilohertz (kHz). 1000 hertz

latch. (1) A simple logic-circuit storage element. (2) A feedback loop in sequential digital circuits used to maintain a state.

least-significant digit. The rightmost digit. See also low-order position.

load. In programming, to enter data into storage or working registers.

low-order position. The rightmost position in a string of characters. See also least-significant digit.

low power Schottky TTL. A version (LS series) of TTL giving a good compromise between low power and high speed. See also transistor-transistor logic and Schottky TTL.

m. (1) Prefix milli; 0.001. (2) Meter.

M. (1) Prefix mega; 1,000,000. (2) When referring to computer storage capacity, 1,048,576. (1,048,576 = 2 to the 20th power.)

mA. Milliampere; 0.001 ampere.

magnetic disk. (1) A flat circular plate with a
magnetizable surface layer on which data can be stored by magnetic recording. (2) See also diskette.

mark. A symbol or symbols that indicate the beginning or the end of a field, of a word, of an item of data, or of a set of data such as a file, a record, or a block.

mask. (1) A pattern of characters that is used to control the retention or elimination of portions of another pattern of characters. (2) To use a pattern of characters to control the retention or elimination of portions of another pattern of characters.

masked. Synonym for disabled.

matrix. (1) A rectangular array of elements, arranged in rows and columns, that may be manipulated according to the rules of matrix algebra. (2) In computers, a logic network in the form of an array of input leads and output leads with logic elements connected at some of their intersections.

matrix printer. A printer in which each character is represented by a pattern of dots; for example, a stylus printer, a wire printer. Synonymous with dot printer.

Mb. 1 048 576 bytes.

mega (M). Prefix 1 000 000.

megahertz (MHz). 1 000 000 hertz.

memory. Term for main storage.

meter (m). A unit of length (equivalent to 39.37 inches).

MFM. Modified frequency modulation.

MHz. Megahertz; 1 000 000 hertz.

micro (μ). Prefix 0.000 001.

microprocessor. An integrated circuit that accepts coded instructions for execution; the instructions may be entered, integrated, or stored internally.

microsecond (μ). 0.000 001 second.

milli (m). Prefix 0.001.

milliampere (mA). 0.001 ampere.

millisecond (ms). 0.001 second.
mnemonic. A symbol chosen to assist the human memory; for example, an abbreviation such as “mpy” for “multiply”.

mode. (1) A method of operation; for example, the binary mode, the interpretive mode, the alphanumeric mode. (2) The most frequent value in the statistical sense.

modem
(modulator-demodulator). A device that converts serial (bit by bit) digital signals from a business machine (or data communication equipment) to analog signals that are suitable for transmission in a telephone network. The inverse function is also performed by the modem on reception of analog signals.

modified frequency modulation (MFM). The process of varying the amplitude and frequency of the ‘write’ signal. MFM pertains to the number of bytes of storage that can be stored on the recording media. The number of bytes is twice the number contained in the same unit area of recording media at single density.

modulation. The process by which some characteristic of

one wave (usually high frequency) is varied in accordance with another wave or signal (usually low frequency). This technique is used in modems to make business-machine signals compatible with communication facilities.

module. (1) A program unit that is discrete and identifiable with respect to compiling, combining with other units, and loading. (2) A packaged functional hardware unit designed for use with other components.

monitor. (1) A device that observes and verifies the operation of a data processing system and indicates any significant departure from the norm. (2) Software or hardware that observes, supervises, controls, or verifies the operations of a system.

most-significant digit. The leftmost (nonzero) digit. See also high-order position.

ms. Millisecond; 0.001 second.

multiplexer. A device capable of interleaving the events of two or more activities, or capable of distributing the events of an
interleaved sequence to the respective activities.

**n.** Prefix nano; 0.000 000 001.

**nano (n).** Prefix 0.000 000 001.

**nanosecond (ns).** 0.000 000 001 second.

**negative true.** Synonym for active low.

**negative-going edge.** The edge of a pulse or signal changing in a negative direction. Synonymous with falling edge.

**ns.** Nanosecond; 0.000 000 001 second.

**NUL.** The null character.

**null character (NUL).** A control character that is used to accomplish media-fill or time-fill, and that may be inserted into or removed from, a sequence of characters without affecting the meaning of the sequence; however, the control of the equipment or the format may be affected by this character.

**offline.** Pertaining to the operation of a functional unit without the continual control of a computer.

**operand.** (1) An entity to which an operation is applied. (2) That which is operated upon. An operand is usually identified by an address part of an instruction.

**operating system.** Software that controls the execution of programs. An operating system may provide services such as resource allocation, scheduling, input/output control, and data management.

**output.** Pertaining to a device, process, or channel involved in an output process, or to the data or states involved in an output process.

**output process.** (1) The process that consists of the delivery of data from a data processing system, or from any part of it. (2) The return of information from a data processing system to an end user, including the translation of data from a machine language to a language that the end user can understand.

**parallel.** (1) Pertaining to the concurrent or simultaneous operation of two or more devices, or to the concurrent performance of two or more activities.
(2) Pertaining to the concurrent or simultaneous occurrence of two or more related activities in multiple devices or channels. (3) Pertaining to the simultaneity of two or more processes. (4) Pertaining to the simultaneous processing of the individual parts of a whole, such as the bits of a character and the characters of a word, using separate facilities for the various parts. (5) Contrast with serial.

**parameter.** (1) A variable that is given a constant value for a specified application and that may denote the application. (2) A name in a procedure that is used to refer to an argument passed to that procedure.

**parity bit.** A binary digit appended to a group of binary digits to make the sum of all the digits either always odd (odd parity) or always even (even parity).

**PEL.** Picture element.

**personal computer.** A small home or business computer that has a processor and keyboard and that can be connected to a television or some other monitor. An optional printer is usually available.

**picture element (PEL).** The smallest displayable unit on a display.

**port.** An access point for data entry or exit.

**positive true.** Synonym for active high.

**positive-going edge.** The edge of a pulse or signal changing in a positive direction. Synonymous with rising edge.

**power supply.** A device that produces the power needed to operate electronic equipment.

**printed circuit.** A pattern of conductors (corresponding to the wiring of an electronic circuit) formed on a board of insulating material.

**printed-circuit board.** A usually copper-clad plastic board used to make a printed circuit.

**priority.** A rank assigned to a task that determines its precedence in receiving system resources.

**processing unit.** A functional unit that consists
of one or more processors and all or part of internal storage.

**processor.** (1) In a computer, a functional unit that interprets and executes instructions. (2) A functional unit, a part of another unit such as a terminal or a processing unit, that interprets and executes instructions. (3) Deprecated term for processing program. (4) See microprocessor.

**program.** (1) A series of actions designed to achieve a certain result. (2) A series of instructions telling the computer how to handle a problem or task. (3) To design, write, and test computer programs.

**programmable read-only memory (PROM).** A read-only memory that can be programmed by the user.

**programming language.** (1) An artificial language established for expressing computer programs. (2) A set of characters and rules with meanings assigned prior to their use, for writing computer programs.

**programming system.** One or more programming languages and the necessary software for using these languages with particular automatic data-processing equipment.

**PROM.** Programmable read-only memory.

**protocol.** (1) A specification for the format and relative timing of information exchanged between communicating parties. (2) The set of rules governing the operation of functional units of a communication system that must be followed if communication is to be achieved.

**pulse.** A variation in the value of a quantity, short in relation to the time schedule of interest, the final value being the same as the initial value.

**radio frequency (RF).** An ac frequency that is higher than the highest audio frequency. So called because of the application to radio communication.

**RAM.** Random access memory. Read/write memory.

**random access memory (RAM).** Read/write memory.

**read.** To acquire or interpret data from a storage device,
from a data medium, or from another source.

**read-only memory (ROM).** A storage device whose contents cannot be modified. The memory is retained when power is removed.

**read/write memory.** A storage device whose contents can be modified. Also called RAM.

**recoverable error.** An error condition that allows continued execution of a program.

**red-green-blue-intensity (RGBI).** The description of a direct-drive color monitor that accepts input signals of red, green, blue, and intensity.

**register.** (1) A storage device, having a specified storage capacity such as a bit, a byte, or a computer word, and usually intended for a special purpose. (2) A storage device in which specific data is stored.

**reverse video.** A form of highlighting a character, field, or cursor by reversing the color of the character, field, or cursor with its background; for example, changing a red character on a black background to a black character on a red background.

**RF.** Radio frequency.

**RF modulator.** The device used to convert the composite video signal to the antenna level input of a home TV.

**RGBI.** Red-green-blue-intensity.

**rising edge.** Synonym for positive-going edge.

**ROM.** Read-only memory.

**ROM/BIOS.** The ROM resident basic input/output system, which provides the level control of the major I/O devices in the computer system.

**RS-232C.** A standard by the EIA for communication between computers and external equipment.

**RTS.** Request to send. Associated with modem control.

**run.** A single continuous performance of a computer program or routine.

**Schottky TTL.** A version (S series) of TTL with faster
switching speed, but requiring more power. See also transistor-transistor logic and low power Schottky TTL.

sector. That part of a track or band on a magnetic drum, a magnetic disk, or a disk pack that can be accessed by the magnetic heads in the course of a predetermined rotational displacement of the particular device.

serial. (1) Pertaining to the sequential performance of two or more activities in a single device. In English, the modifiers serial and parallel usually refer to devices, as opposed to sequential and consecutive, which refer to processes. (2) Pertaining to the sequential or consecutive occurrence of two or more related activities in a single device or channel. (3) Pertaining to the sequential processing of the individual parts of a whole, such as the bits of a character or the characters of a word, using the same facilities for successive parts. (4) Contrast with parallel.

setup. (1) In a computer that consists of an assembly of individual computing units, the arrangement of interconnections between the units, and the adjustments needed for the computer to operate. (2) The preparation of a computing system to perform a job or job step. Setup is usually performed by an operator and often involves performing routine functions, such as mounting tape reels. (3) The preparation of the system for normal operation.

signal. A variation of a physical quantity, used to convey data.

software. (1) Computer programs, procedures, and rules concerned with the operation of a data processing system. (2) Contrast with hardware.

source. The origin of a signal or electrical energy.

start bit. A signal to a receiving mechanism to get ready to receive data or perform a function.

stop bit. A signal to a receiving mechanism to wait for the next signal.

storage. (1) A storage device. (2) A device, or part of a device, that can retain data. (3) The retention of data in a storage device. (4) The placement of data into a storage device.
Strobe. An instrument that emits adjustable-rate flashes of light. Used to measure the speed of rotating or vibrating objects.

Symbol. (1) A conventional representation of a concept or a representation of something by reason of relationship, association, or convention. (2) A representation of something by reason of relationship, association, or convention.

Synchronization. The process of adjusting the corresponding significant instants of two signals to obtain the desired phase relationship between these instants.

Synchronous transmission. (1) Data transmission in which the time of occurrence of each signal representing a bit is related to a fixed time frame. (2) Data transmission in which the sending and receiving devices are operating continuously at substantially the same frequency and are maintained, by means of correction, in a desired phase relationship.

Time-out. (1) A parameter related to an enforced event designed to occur at the conclusion of a predetermined elapsed time. A time-out condition can be cancelled by the receipt of an appropriate time-out cancellation signal. (2) A time interval allotted for certain operations to occur; for example, response to polling or addressing before system operation is interrupted and must be restarted.

Track. (1) The path or one of the set of paths, parallel to the reference edge on a data medium, associated with a single reading or writing component as the data medium moves past the component. (2) The portion of a moving data medium such as a drum, or disk, that is accessible to a given reading head position.

Transistor-transistor logic (TTL). A popular logic circuit family that uses multiple-emitter transistors.

Translate. To transform data from one language to another.

Transmission. (1) The sending of data from one place for reception elsewhere. (2) In ASCII and data communication, a series of characters including headings and text. (3) The dispatching of a signal, message, or other
form of intelligence by wire, radio, telephone, or other means. (4) One or more blocks or messages. For BSC and start-stop devices, a transmission is terminated by an EOT character. (5) Synonymous with data transmission.

**TTL.** Transistor-transistor logic.

**V.** Volt.

**video.** Computer data or graphics displayed on a cathode ray tube, monitor, or display.

**volt.** The basic practical unit of electric pressure. The potential that causes electrons to flow through a circuit.

**W.** Watt.

**watt.** The practical unit of electric power.

**word.** (1) A character string or a bit string considered as an entity. (2) See computer word.

**write.** To make a permanent or transient recording of data in a storage device or on a data medium.

**write precompensation.** The varying of the timing of the head current from the outer tracks to the inner tracks of the diskette to keep a constant ‘write’ signal.
Notes:
ac adapter specifications B-24
access bit, divisor latch 3-14, 3-37
acknowledge signal 3-22
adapter power line 2-44
adapter, automobile 3-90
   communications 5-10
address bits 0-7 2-35
address bits 8-16 2-35
address enable line 2-44
address latch enable 2-36
address latch enable line 2-42
address lines 2-41
alarm interrupt 2-88
alarm interrupt (BIOS) 4-21
all-points-addressable mode 2-50
alphanumeric mode 2-46, 2-50
alphanumeric mode, CRT 3-54
analog loop back test 3-52
answer command, modem 3-42
aspect ratio, LCD 5-8
aspect ratio, printer 5-11
asynchronous communications 5-10
attribute
   intensify 2-48
   reverse video 2-48
attribute byte 2-47, 2-50
attribute byte, CRT 3-54
audible signals D-4
audio control 4-36
audio control register 2-73
audio controller 2-72
   programming considerations 2-73
auto feed line 3-22
automatic power off 4-41
automobile adapter 3-90
automobile power adapter specifications B-25

basic input/output system 4-3
BASIC interrupt (BIOS) 4-17
battery charger 3-90
battery charger specifications B-25
battery input specifications B-4
battery warning D-4
BIOS
   accessing 4-3
   alarm interrupt 4-21
   bootstrap interrupt 4-17
   breakpoint interrupt 4-14
   diskette interrupt 4-13
diskette parameters interrupt 4-20
divide by zero 4-7
equipment interrupt 4-12
event post/wait interrupt 4-15
interrupt vectors 4-4
keyboard break interrupt 4-19
keyboard interrupt 4-16
lower character graphics interrupt 4-20
memory interrupt 4-12
overflow interrupt 4-8
parameters 4-3
print screen interrupt 4-8
printer interrupt (BIOS) 4-17
programming guidelines 4-26
reserved allocations 4-21
reserved areas 4-23
resident BASIC interrupt 4-17
single step interrupt 4-7
system resume interrupt 4-21
system services interrupt 4-15
time of day interrupt 4-18
timer tick interrupt 4-19
upper character graphics interrupt 4-20
usage 4-3
video initialization interrupt 4-19
video input/output interrupt 4-10
blank function 4-41
board, system 2-4
bootstrap interrupt (BIOS) 4-17
break 4-36
break command, modem 3-42
break interrupt indicator 3-17, 3-39
breakpoint interrupt (BIOS) 4-8
buffer, LCD refresh 2-57
busy signal 3-22
card select 2-36
carrier detect line 3-8
cascading of interrupts 2-18
cassette 5-7
channel timer 2-12
channel check line 2-42
channel check mask register 2-23
channel ready line 2-42
character code 2-47, 2-50, A-5
character code, CRT 3-54
character codes 4-27
characteristics, operating See specifications
charger, battery 3-90
clear byte pointer register 2-30
clear mask register 2-30
clear to send 3-18, 3-40
clear to send line 3-9
clicker control 4-36
clock (CK) signal 2-41
clock line 2-41
clock rate, system 2-11
clock, real-time 2-88
clock, sleep 2-11
clock, system 2-11
codes, extended
  keyboard 4-34
codes, power on error D-2
cold start 4-35
color display 3-90
color display
  specifications B-20
color mapping 5-8
color mapping, CRT 3-55
color operations 2-50
color operations, LCD 2-47
color set, CRT 3-59
color/graphics displays 5-9
command, internal modem
  answer 3-42
  break 3-42
  count 3-42
  DIAL 3-43
  format 3-44
  hangup 3-44
  initialize 3-45
  long response 3-46
  modem 3-47
  new 3-47
  originate 3-47
  pickup 3-47
  query 3-48
  redial 3-49
  speed 3-49
  transparent 3-50
  voice 3-51
  wait 3-51
  xmit 3-51
  ztest 3-52
common carrier
  interface 3-32
communications
adapter 5-10
communications
  interface 3-5
communications interrupt
  (BIOS) 4-14
compatibility 5-2
  hardware differences 5-6
  loops 5-5
  programming
    considerations 5-3
  stack manipulation 5-4
  timing dependencies 5-6
  unequal
    configurations 5-6
    using BIOS and
    DOS 5-3
compatibility with the IBM
  AT 5-3
composite CRT
  interface 3-61
configuration switches 5-7
configurations, unequal 5-6
connector
  common carrier 3-32
  communications
    interface 3-5
  composite CRT 3-61
  composite video 3-61
  CRT 3-60
  direct drive video 3-60
  diskette drive 2-76
  I/O channel 2-38
  LCD 2-54
  modem 3-30
  parallel printer 3-20
  RAM 2-34
  rf modulator 3-62
connector requirements
  See specifications
control modes (DMA) 5-10
control registers 2-80
  audio 2-73
  channel check
    mask 2-23
  clear byte pointer 2-30
  clear mask 2-30

Index X-25
color select 3-67
control data (LCD) 2-58
control index
(LCD) 2-58
CRT adapter 3-63
data (CRT) 3-64
data (LCD) 2-58
data register 2-85
data register (diskette controller) 2-82
diagnostics 2-22
digital input register
(diskette controller) 2-83
digital output register 2-81
diskette control 2-22,
2-80
diskette controller 2-80
divisor 3-13, 3-35
DMA 2-25
DMA command 2-26
DMA request 2-27
DMA status 2-27
DMA write single mask 2-28
I/O 2-6
I/O register 2 2-15,
2-20, 2-73
I/O register 3 2-15,
2-21
index (CRT) 3-64
index (LCD) 2-58
internal modem 3-32
interrupt diagnostics 2-21
interrupt enable 3-13,
3-35
interrupt identification 3-14,
3-36
keyboard 2-22
keyboard data 2-72
LCD 2-58
line control register 3-14, 3-36
line status register 3-16,
3-39
main status 2-82
mask, channel check 2-23
master clear 2-30
mode control (CRT) 3-66
mode control (LCD) 2-61
mode register (DMA) 2-29
modem control 3-16,
3-38
modem status 3-17, 3-40
NMI 2-20
parallel adapter 3-24
power system control 2-23
printer adapter 2-85
printer control 3-27
printer data 3-25
printer status 3-26
rate divisor latch 2-85
real-time clock 2-90
receive buffer 3-12,
3-34
register A (real-time clock) 2-91
register B (real-time clock) 2-92
register C (real-time clock) 2-92
register D (real-time clock) 2-93
serial adapter 3-11
status 2-86
status (CRT) 3-68
status (LCD) 2-62
status register (diskette controller) 2-82
status, main 2-82
system clock
color 2-21
system control 2-6
system register 3-34
system register, parallel
adapter 3-25
system register, serial
adapter 3-12
timer control 2-14
transmit buffer 3-12, 3-34
vectored interrupt
color 2-19
write all mask register
bits 2-31
control registers,
system 2-6, 2-9
controller
direct memory access
(DMA) 2-24
diskette 2-75
interrupt 2-16
keyboard 2-62
LCD 2-46
Count command,
modem 3-42
CRT adapter
alphanumeric
mode 3-54
color mapping 3-55
color select register 3-67
color sets 3-59
composite 3-61
connector 3-60
control registers 3-63
data register 3-64
direct drive 3-60
display storage 3-53
graphics mode 3-57
high resolution 3-58
I/O interface 3-60
index register 3-64
medium resolution 3-58
mode control
register 3-66
pel mapping 3-59
programming
considerations 3-63
rf modulator 3-62
status register 3-68
storage map 3-57
CRT display adapter
specifications B-15

data bits 0-7 2-35
data carrier detect 3-8
data enable 2-36
data enable line 2-44
data lines 2-41
data ready indicator 3-17, 3-39
data register 2-58, 2-85
data register, CRT 3-64
data registers (diskette
controller) 2-82
data set ready 3-17, 3-40
data set ready line 3-8
data terminal ready 3-16, 3-38
data terminal ready
line 3-10
date error D-3
delta clear to send 3-18, 3-41
delta data set ready
indicator 3-18, 3-41
delta received line signal
detector 3-18, 3-40
diagnostics control
register 2-22
diagnostics, interrupt 2-21

Index X-27
See timing diagrams
DIAL command, modem 3-43
digital input register 2-83
digital-output register 2-81
direct drive video specifications B-16
direct memory access 2-24
direction line 2-78
diskette controller 2-75
data registers 2-82
digital input register 2-83
digital output register 2-81
programming considerations 2-80
registers 2-80
status register 2-82
drive 2-74
carrier 2-76
interface 2-76
drive interface 2-76
specifications B-6, B-7
diskette changed line 2-79
diskette control register 2-22, 2-80
diskette drive 2-74
diskette drive considerations 5-12
diskette drive sensors 2-75
diskette drive specifications B-6
diskette interrupt (BIOS) 4-13
diskette parameters interrupt (BIOS) 4-20
diskette specifications B-7
display color 3-90
monochrome 3-89
display enable (LCD) 2-55
display storage (CRT) 3-53
display storage (LCD) 2-46
display storage map 2-52
display storage map, CRT 3-57
displays, color/graphics 5-9
divide by zero interrupt 4-7
divisor register 3-13, 3-35
DMA acknowledge line 2-43
DMA controller acknowledge line 2-43
channels 5-10
clear byte pointer register 2-30
clear mask 2-30
command register 2-26
control modes 5-10
controller registers 2-25
master clear 2-30
memory read timing C-20
memory write timing C-22
mode register 2-29
request lines 2-43
request register 2-27
status register 2-27
write all mask register bits 2-31
write single mask register 2-28
DMA from I/O to memory timing C-10
DMA from memory to I/O timing C-8
DMA request lines 2-43
drive oscillator (LCD) 2-55
drive select line 2-78
EIA drivers 3-8
enable clock line
(LCD) 2-56
environmental specifications
  See specifications
equipment interrupt
  (BIOS) 4-12
error codes, power on D-2
error line 3-23
error, date D-3
error, time D-3
event post/wait interrupt
  (BIOS) 4-15
extended keyboard
codes 4-34

failures, features D-3
failures, system unit D-2
feature failures D-3
font storage area 2-46
fonts, RAM 5-8
foreground 2-48
background 2-48
foreground, CRT 3-54
background, CRT 3-54
format command,
  modem 3-44
frame error indicator 3-17, 3-39
frame start line (LCD) 2-56
functional units 2-3

generator, rate 3-13, 3-35
graphics mode, CRT 3-57
graphics operations 2-50

hangup command,
  modem 3-44
hardware differences 5-6
hardware interrupt
  levels 2-17
hardware interrupts
  keyboard port
    loaded 4-9
    level 0 4-9
    level 1 4-9
    level 2 4-9
    level 3 4-9
    level 4 4-9
    level 5 4-10
    level 6 4-10
    level 7 4-10
head select line 2-78
high resolution
  CRT 3-58
  LCD 2-52
high resolution mapping,
  CRT 3-58
  high z line 3-31

I/O address map 2-7
I/O addresses 2-6
I/O channel
  audio 2-72
  communications 3-5
  composite CRT 3-61
  connector 2-38
  CRT adapter 3-60
  direct drive 3-60
diskette drive 2-76
interface 2-37
internal modem 3-29
parallel adapter 3-19
parallel printer 3-20
power supply 2-93
printer 2-84
rf modulator 3-62
serial adapter
specifications B-4
I/O channel check line 2-42
I/O channel check mask register 2-23
I/O channel ready line 2-42
I/O channel
specifications B-4
I/O memory read line 2-43
I/O read line 2-42
I/O read timing,
processor C-12
I/O register 2 2-15, 2-20,
2-73
I/O register 3 2-15, 2-21
I/O registers 2-6, 2-8
I/O timing, DMA C-10
I/O timing, DMA from memory C-8
I/O write line 2-43
I/O write timing,
processor C-14
identification byte 5-12
index line 2-79
index register 2-58
index register, CRT 3-64
initialization command words (ICWs) 2-18
initialize command,
modem 3-45
initialize line 3-23
input requirements
See specifications
intensity, LCD 5-9
interface, printer 2-84
internal modem
answer command 3-42
break command 3-42
commands 3-41
connector 3-30, 3-32
control registers 3-32
count command 3-42
DIAL command 3-43
format command 3-44
hangup command 3-44
initialize command 3-45
long response
command 3-46
modem command 3-47
new command 3-47
originate command 3-47
pickup command 3-47
programming
considerations 3-32
query command 3-48
redial command 3-49
speed command 3-49
system board
interface 3-29
transparent
command 3-50
voice command 3-51
wait command 3-51
xmit command 3-51
ztest command 3-52
internal modem
specifications B-22
interrupt cascading 2-18
interrupt control
registers 2-19
interrupt controller 2-16
interrupt diagnostics 2-21
interrupt enable
register 3-13, 3-35
interrupt identification 3-14, 3-36
interrupt identification
register 3-14, 3-36

X-30 Index
interrupt levels 2-16, 2-17
interrupt pending
indicator 3-14, 3-36
interrupt request line 2-42
interrupt vectors 4-4
interrupt vectors (BIOS) 4-4
interrupt, alarm 2-88
interrupt, nonmaskable 4-7
interrupt, periodic 2-88
interrupt, update ended 2-88
interrupts
alarm (BIOS) 4-21
BASIC (BIOS) 4-17
BIOS interrupt vectors 4-4
bootstrap (BIOS) 4-17
breakpoint (BIOS) 4-8
communications (BIOS) 4-14
diskette (BIOS) 4-13
diskette parameters (BIOS) 4-20
divide by zero (BIOS) 4-7
equipment (BIOS) 4-12
event post/wait (BIOS) 4-15
keyboard (BIOS) 4-16
keyboard break (BIOS) 4-19
keyboard port loaded 4-9
level 0 (hardware) 4-9
level 1 (hardware) 4-9
level 2 (hardware) 4-9
level 3 (hardware) 4-9
level 4 (hardware) 4-9
level 5 (hardware) 4-10
level 6 (hardware) 4-10
level 7 (hardware) 4-10
lower character graphics (BIOS) 4-20
memory (BIOS) 4-12
nonmaskable (NMI) 2-16
overflow (BIOS) 4-8
print screen (BIOS) 4-8
printer interrupt (BIOS) 4-17
real-time clock 2-88
reserved allocations 4-21
single step (BIOS) 4-7
software 4-26
system resume (BIOS) 4-21
system services (BIOS) 4-15
time of day (BIOS) 4-18
timer tick (BIOS) 4-19
upper character graphics (BIOS) 4-20
vectored 2-16
video initialization (BIOS) 4-19
video input/output (BIOS) 4-10

introduction 1-1

keyboard
character codes 4-27
codes, extended 4-34
controller 2-62
data register 2-72
extended codes 4-34
keyboard encoding 4-27
keyboard interrupt (BIOS) 4-16
keyboard usage 4-27
layout 2-63, 2-64, 2-65, 2-66
programming
considerations
scan code mapping 2-68
special handling 4-35
keyboard break interrupt (BIOS) 4-19
keyboard considerations 5-12
keyboard control
register 2-22
keyboard data register 2-72
keyboard encoding 4-27
keyboard interrupt (BIOS) 4-16
keyboard layout 4-33
keyboard port loaded interrupt 4-9
keyboard usage 4-27

LCD
layout, keyboard 4-33

LCD
all-points-addressable 2-50
color operations 2-50
connector 2-54
control data
registers 2-58
control index
register 2-58
control registers 2-58
controller 2-46
data register 2-58
display 2-45
display storage 2-46
graphics operations 2-50
high resolution 2-52
index register 2-58
interface 2-54
interface timing C-24
medium resolution 2-53
mode control
register 2-61
Operations
monochrome 2-47
pel mapping 2-53
programming
considerations 2-57
refresh buffer 2-57
specifications B-5
status register 2-62
storage map 2-52

LCD aspect ratio 5-8
LCD intensity 5-9
LCD sense line 2-56
LCD specifications B-5
LCD switching 5-7
level-0 interrupt (hardware) 4-9
level-1 interrupt (hardware) 4-9
level-2 interrupt (hardware) 4-9
level-3 interrupt (hardware) 4-9
level-4 interrupt (hardware) 4-9
level-5 interrupt (hardware) 4-10
level-6 interrupt (hardware) 4-10
level-7 interrupt (hardware) 4-10
line
acknowledge line 2-43
adapter power 2-44
address 2-41
address bits 0–7 2-35
address bits 8–16 2-35
address enable 2-44
address latch
  enable 2-36, 2-42
card select 2-36
carrier detect 3-8
channel check 2-42
channel ready 2-42
clear to send 3-9
clock 2-41
data 2-41
data bit 0-7 2-35
data enable 2-36, 2-44
data set ready 3-8
data terminal ready 3-10
direction 2-78
diskette changed 2-79
display enable
  (LCD) 2-55
drive oscillator
  (LCD) 2-55
drive select 2-78
head select 2-78
high Z 3-31
index 2-79
interrupt request 2-4
interrupt request, serial adapter
memory card select 2-36
memory read 2-43
memory write 2-36, 2-43
motor enable 2-78
power adapter
  active 2-44
power enable 2-44, 2-45
printer busy 2-45
printer error 2-45
printer power 2-45
RAM enable 2-36
read 2-42
read data 2-79
receive data 3-9
request lines 2-43
request to send 3-9
reset 2-42
reset, serial adapter
  ring indicator 3-10
scan data (LCD) 2-55
select, head 2-78
sense line (LCD) 2-56
serial adapter
sleep clock 2-41
start frame (LCD) 2-56
step 2-78
terminal count 2-43
track 0 2-79
transmit data 2-44, 3-9
write 2-43
write data 2-79
write enable 2-79
write protect 2-79
X enable clock
  (LCD) 2-56
X shift clock
  (LCD) 2-56
Y shift clock
  (LCD) 2-56
line control register 3-14,
  3-36
line status register 3-16,
  3-39
liquid crystal display 2-46
long response command,
  modem 3-46
loops 5-5
low battery warning D-3
lower character graphics
  interrupt (BIOS) 4-20

main status register 2-82
  color mapping,
  CRT 3-55
CRT 3-57  
memory to I/O timing,  
DMA C-8  
memory write  2-36  
memory write line  2-43  
memory write timing,  
DMA C-22  
memory write timing,  
processor C-6  
mode control register  
(CRT) 3-66  
mode control register  
(LCD) 2-61  
mode control register  
(serial printer) 2-87  
mode control (serial printer) 2-87  
mode, sleep 4-40  
modem command,  
modem 3-47  
modem control register 3-16, 3-38  
modem status register 3-17, 3-40  
modem, internal 3-28  
monochrome display 3-89  
monochrome display specifications B-18  
monochrome displays 5-9  
motor enable 2-78  

N  

new command, modem 3-47  
NMI control registers 2-20  
nonmaskable interrupt 4-7  
nonmaskable interrupts (NMI) 2-16  

O  

operating characteristics  
See specifications
operation command words (ICWs) 2-18
operations
  color 2-50
  graphics 2-50
  high resolution 2-52
  high resolution, CRT 3-58
  medium resolution 2-53
  medium resolution, CRT 3-58
  monochrome 2-47
originate command, modem 3-47
output 1 3-16, 3-38
output 2 3-16, 3-38
overflow interrupt (BIOS) 4-8
overrun error indicator 3-17, 3-39

[Page]

paper end signal 3-22
parallel adapter
  I/O interface 3-19
  printer 3-20
programming considerations 3-24
parallel port specifications B-10
parity error indicator 3-17, 3-39
pause 4-35
pel mapping 2-53
pel mapping, CRT 3-59
periodic interrupt 2-88
physical description
  See specifications
pickup command, modem 3-47
portable printer 3-68
portable printer specifications B-11
power adapter active line 2-44
power adapter input specifications B-3
power enable line 2-44
power off (suspend) 4-38
power-off function 4-41
power on (resume) 4-39
power on error codes D-2
power requirements
  See specifications
power supply 2-93
power supply interface 2-93
power system control register 2-23
print screen 4-35
print screen interrupt (BIOS) 4-8
printer adapter control registers 2-85
printer aspect ratio 5-11
printer busy line 2-45
printer control register 3-27
printer data bits 3-22
printer data register 3-25
printer error 2-45
printer interface 2-84
  mode control register 2-87
  printer interrupt (BIOS) 4-17
  programming considerations 2-85
printer interrupt (BIOS) 4-17
printer power enable line 2-45
printer power line 2-45
printer status register 3-26
printer, portable 3-68

Index X-35
printers 5-11
processor 2-10
processor I/O read
timing C-12
processor I/O write
timing C-14
processor memory read
timing C-4
processor memory write
timing C-6
processor read timing C-16
processor write timing C-18
programming considerations
audio controller 2-73
CRT adapter 3-63
diskette controller 2-80
internal modem 3-32
interrupt controller 2-18
LCD 2-57
parallel adapter 3-24
printer interface 2-85
programming
considerations
real-time clock 2-89
serial adapter 3-11
special
considerations 5-3
system clock 2-11
timer 2-12
programming guidelines,
BIOS 4-26
RAM connector 2-34
RAM enable 2-36
RAM fonts 5-8
rate divisor latch 2-85
rate generator 3-13, 3-35
read data line 2-79
read-only memory 2-32,
2-33
read timing, DMA C-20
read timing, processor C-16
read timing, processor
I/O C-12
read timing, processor
memory C-4
real-time clock 2-88
control registers 2-90
programming
considerations 2-89
register A 2-91
register B 2-92
register C 2-92
register D 2-93
real-time clock control
registers 2-90
receive buffer, serial
adapter 3-12, 3-34
receive data line 3-9
received line signal
detect 3-8, 3-17, 3-40
redial command,
modem 3-49
refresh buffer, LCD 2-57
register A (real-time
clock) 2-91
register B (real-time
clock) 2-92
register C (real-time
clock) 2-92
register D (real-time
clock) 2-93
request to send line 3-9
reserved areas 4-21
reserved areas. 4-23

query command,
modem 3-48

RAM 2-33
specifications B-8

X-36 Index
reserved interrupt allocations 4-21
reset line 2-42
resident BASIC interrupt (BIOS) 4-17
resume 4-39
reverse video 2-48
rf modulator interface 3-62
rf modulator specifications B-17
ring indicator 3-17, 3-40
ring indicator detector 3-18, 3-40
ring indicator line 3-10
ROM 2-32
memory map 2-32

shift clock line (LCD) 2-56
signal requirements
See specifications
signals, audible D-4
single step interrupt (BIOS) 4-7
sleep 4-40
sleep clock 2-11, 2-41
sleep clock line 2-41
sleep mode 2-11
speaker 2-72
special handling 4-35
special programming considerations 5-3
specifications
automobile power adapter B-25
battery charger B-25
battery input B-4
color display B-20
composite video B-16
cRT display adapter B-15
direct drive video B-16
diskette B-7
diskette drives B-6
I/O channel B-4
internal modem B-22
LCD B-5
monochrome display B-18
parallel port B-10
portable printer B-11
power adapter input B-3
RAM B-8
rf modulator B-17
serial port B-10
serial/parallel adapter B-9
system unit B-2
speed command, modem 3-49
stack manipulation 5-4

scan data line (LCD) 2-55
select in line 3-23
select line, head 2-78
selected line 3-22
sense line, LCD 2-56
sensors, diskette drive 2-75
serial adapter communications interface 3-5
control registers 3-11
I/O channel
interrupt request, serial adapter 3-5
programming considerations 3-11
system interface 3-3
serial data 2-84
serial port specifications B-10
serial/parallel adapter specifications B-9
shift clock line 2-56

Index X-37
stand by condition  4-38  
start frame line (LCD)  2-56  
status register  2-86  
status register (CRT)  3-68  
status register (diskette controller)  2-82  
status register (LCD)  2-62  
status register, main  2-82  
step line  2-78  
storage area, font  2-46  
storage map  2-6  
strobe pulse  3-21  
supply, power  2-93  
suspend  4-38  
switches, configuration  5-7  
synchronous communications  5-10  
system board  2-4, 2-5  
system clock  2-11, 5-7  
system clock control  2-21  
system clock rate  2-11  
system cold start  4-35  
system control registers  2-6, 2-9  
system memory  2-6  
system memory map  2-6  
system register, modem  3-34  
system register, parallel adapter  3-25  
system register, serial adapter  3-12  
system request  4-36  
system resume  4-39  
system resume interrupt (BIOS)  4-21  
system services interrupt (BIOS)  4-15  
system test  4-35  
system timer  2-12  
system unit failures  D-2  
system unit specifications  B-2  
terminal count line  2-43  
test, system  4-35  
time error  D-3  
time of day  5-7  
time of day interrupt (BIOS)  4-18  
time out  4-41  
timer  2-13  
timer control word  2-14  
timer registers  2-13  
timer tick interrupt (BIOS)  4-19  
timer values  2-12  
timer, system  2-12  
timers  5-7  
timing dependencies  5-6  
timing diagram  C-8  
timing diagrams  
  DMA from I/O to memory  C-10  
  DMA from memory to I/O  C-8  
  DMA read  C-20  
  DMA write  C-22  
  LCD interface  C-24  
  processor I/O read  C-12  
  processor I/O write  C-14  
  processor memory read  C-4  
  processor memory write  C-6  
  processor read  C-16  
  processor write  C-18  
track 0 line  2-79  
transmit buffer  3-12, 3-34  
transmit data line  2-44, 3-9
transmitter empty indicator 3-16, 3-39
transmitter holding register empty indicator 3-16, 3-39
transparent command, modem 3-50

write enable line 2-79
write protect line 2-79
write timing, DMA C-22
write timing, processor C-18
write timing, processor I/O C-14
write timing, processor memory C-6

unequal configurations 5-6
update ended interrupt 2-88
upper character graphics interrupt (BIOS) 4-20

X enable clock line (LCD) 2-56
X shift clock line (LCD) 2-56
xmit command, modem 3-51

Y shift clock (LCD) 2-56

ztest command, modem 3-52

wait command, modem 3-51
wait on external event 4-40
warning, low battery D-3
weights See specifications

write all mask register bits register 2-31
write data line 2-79

80C88 microprocessor 2-10

Index X-39
Notes:

X-40 Index
Reader's Comment Form

IBM PC Convertible
Technical Reference
Volume 1

Your comments assist us in improving our publication; they are an important part of the input used for revisions.

IBM may use and distribute any of the information you supply in any way it believes appropriate without incurring any obligation whatever. You may, of course, continue to use the information you supply.

Please do not use this form for questions regarding setup, operation, or program support or for requests for additional publications. Instead, contact your authorized IBM PC Convertible dealer in your area.

Comments:
IBM CORPORATION
DEPARTMENT 95H
11400 BURNET ROAD
AUSTIN, TEXAS 78758