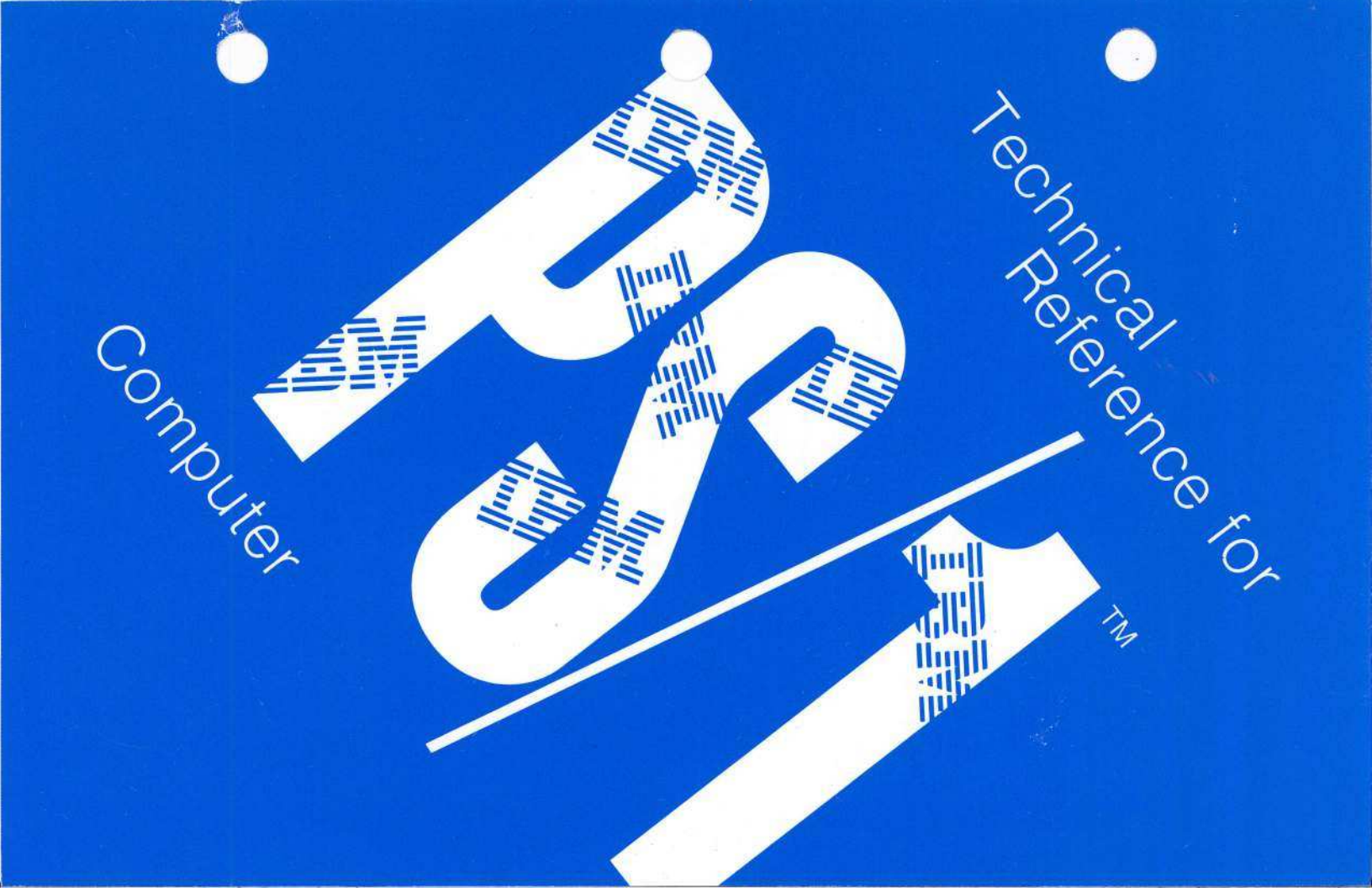

IBM
PS/2
Computer
Technical Reference for
IBM
TM

First Edition (April 1990)

Address comments about this publication to IBM Corporation, Dept. F95/035-3, 740 New Circle Road N.W., Lexington, KY 40511 USA. Information you supply may be used by IBM without obligation.

For copies of publications related to this product, call toll free 1-800-IBM-7282 in the Continental USA. In Alaska, call 1-414-633-8108. In Canada, call toll free 1-800-465-1234. Other locations, contact your point of purchase.

Preface

This technical reference provides system-specific hardware and software interface information for the IBM Personal System/1™ computer. It is intended for developers who provide hardware and software products to operate with this system.

You should understand the concepts of computer architecture and programming before using this publication.

This manual is divided into the following sections.

Section 1, "System Description" contains a description of IBM PS/1™ computer and discusses system board features, an I/O address map, and system unit specifications.

Section 2, "System Board" discusses the operation of the system microprocessor, interrupts, system timers, and speaker.

Section 3, "System Board I/O Controllers" describes the input and output interfaces of the system board. This includes the operation of the keyboard/mouse controller, video subsystem, diskette drive controller, serial port controller, parallel port controller, and the different types of memory.

Section 4, "Power Supply" contains electrical input/output specifications and theory of operation for the power supply.

Section 5, "Keyboard" contains a description and specifications for the keyboard.

Section 6, "Mouse" contains a description and specifications for the mouse.

Section 7, "Displays" describes the Color or Black and White ™PHOTO GRAPHIC Displays.

Section 8, "Drives" contains a description of the 3.5-inch 30MB fixed disk drive and controller and the 3.5-inch 1.44MB diskette drive.

Section 9, "Modem/RS-232C Interface Cards" describes the 2400 bps modem card and the RS-232C interface card.

Section 10, "80286 Microprocessor Instruction Set" contains a quick reference for the 80286 assembly instruction set.

Section 11, "Characters and Keystrokes" contains the decimal and hexadecimal values for ASCII characters.

Section 12, "Compatibility" discusses guidelines to help developers design programs, adapters, and attachments that are compatible with other IBM Personal System/2 products and IBM Personal Computer products.

Appendix, "Options" contains features purchased separately, such as the 512KB memory expansion, audio card and joystick, 5.25-inch diskette drive unit, and adapter card unit.

A Glossary, Bibliography, and Index are also provided.

Suggested Related Publications:

- *BIOS Interface Technical Reference for IBM PS/1™ Computer*
- Hardware Maintenance Service for *IBM PS/1™ Computer*
- *Using DOS 4.00*
- *DOS 4.00 Command Reference*
- *DOS 4.00 Technical Reference and Application Programming*
- *IBM Personal Computer BASIC Reference.*

Contents

Section 1. System Description 1-1
Description 1-3
System Board Features 1-4
System Board Locations 1-5
System Board Block Diagram 1-6
System I/O Address Map 1-7
Specifications 1-8

Section 2. System Board 2-1
Description 2-3
System Microprocessor 2-3
Real Address Mode 2-3
Protected Virtual Address Mode 2-4
Performance 2-4
Adapter Card Channel 2-5
Connector 2-5
Signal Description 2-7
Signal Timings 2-11
DMA Controller 2-19
Data Transfers Between Memory and I/O Devices 2-20
Byte Pointer 2-20
DMA Channels 2-20
Page Register Addresses 2-21
DMA I/O Address Map 2-23
DMA Registers 2-24
Interrupts 2-28
Non-Maskable Interrupt 2-28
Interrupt Assignments 2-29
System Timers 2-31
Channel 0 - System Timer 2-32
Channel 1 - Refresh Request Generator 2-32
Channel 2 - Tone Generation for Speaker 2-32
Timers 0, 1, and 2 2-32
Programming the System Timers 2-32
Counter Write Operations 2-33
Counter Read Operations 2-33
Registers 2-34
Counter Latch Command 2-35
System Timer Modes 2-36
Operations Common to All Modes 2-41
Speaker 2-42

Section 3. System Board I/O Controllers 3-1
Keyboard/Mouse Controller 3-3
Keyboard Password Security 3-3
Keyboard Controller Command and Status Bytes 3-5
Keyboard/Mouse Programming Considerations 3-9
Mouse/System Timings 3-10
Signals 3-12
Connector 3-13
Video Subsystem 3-14
VGA Components 3-17
BIOS ROM 3-18
Support Logic 3-18
Video Graphics Array Major Components 3-18
Modes of Operation 3-22
Display Support 3-23
Video Subsystem Programmable Option Select 3-24
Alphanumeric Modes 3-25
Graphics Modes 3-28
Video Memory Organization 3-35
Video Memory Read/Write Operations 3-50
Registers 3-53
General Registers 3-54
Sequencer Registers 3-58
CRT Controller Registers 3-64
Graphics Controller Registers 3-79
Attribute Controller Registers 3-87
VGA Programming Considerations 3-93
Video Digital-to-Analog Converter (Video DAC) 3-101
Diskette Drive Controller 3-111
Registers 3-112
Diskette Drive Controller Programming Considerations . . . 3-115
Command Format 3-118
Command Status Registers 3-128
Signal Descriptions 3-131
Connector 3-134
Serial Port Controller 3-135
Communications Application 3-136
Programmable Baud-Rate Generator 3-137
Registers 3-137
Serial Port Controller Programming Considerations 3-147
Signal Descriptions 3-147
System Board Voltage Interchange 3-148
Parallel Port Controller 3-149
Parallel Port Programmable Option Select 3-150
Parallel Port Controller Programming Considerations 3-151
Parallel Port Timing 3-154

Signal Descriptions 3-154
Connector 3-155
Memory 3-156
Read-Only Memory (ROM) 3-156
Random Access Memory (RAM) 3-156
Complementary Metal Oxide Semiconductor RAM 3-158
CMOS RAM Configuration 3-162
Miscellaneous System Ports 3-165
System Control Port B (Hex 0061) 3-165
RTC/CMOS and NMI Mask (Hex 0070) 3-166
System Control Port A (Hex 0092) 3-167

Section 4. Power Supply 4-1
Description 4-3
Outputs 4-3
Power Controller Coordination 4-3
Output Protection 4-3
Voltage Sequencing 4-4
No-Load Operation 4-4
Auto Restart 4-4
Power Good Signal 4-4
Power Card Connectors 4-5
Power/Audio Cable Connector 4-6

Section 5. Keyboard 5-1
Description 5-3
Keyboard Charts 5-3
Belgium (Dutch) Keyboard 5-6
Canada (French) Keyboard 5-7
Denmark Keyboard 5-8
Finland and Sweden Keyboard 5-9
France and Belgium (French) Keyboard 5-10
Germany Keyboard 5-11
Italy Keyboard 5-12
Netherlands Keyboard 5-13
Norway Keyboard 5-14
Portugal Keyboard 5-15
Spain Keyboard 5-16
Switzerland (French) Keyboard 5-17
Switzerland (German) Keyboard 5-18
U.K. Keyboard 5-19
U.S. and Canada (English) Keyboard 5-20
Sequential Key-Code Scanning 5-21
Buffer 5-21
Keys 5-21
Power-On Routine 5-22

Power-On Reset (POR) 5-22
Basic Assurance Test (BAT) 5-22
Commands from the System 5-23
Commands to the System 5-28
Scan Codes 5-29
Set 1 Scan Code Tables 5-29
Set 2 Scan Code Tables 5-32
Set 3 Scan Code Tables 5-36
Clock and Data Signals 5-39
Data Stream 5-39
Data Output 5-40
Data Input 5-41
Encode and Usage 5-42
Extended Functions 5-45
Shift States 5-46
Special Handling 5-48
System Reset 5-48
Break 5-48
Pause 5-48
Print Screen 5-48
System Request (SysRq) 5-49
Other Characteristics 5-49
Cables and Connectors 5-50
Specifications 5-51

Section 6. Mouse 6-1
Description 6-3
Programming Considerations 6-4
Commands 6-4
Data Report 6-9
Error Handling 6-10
Data Transmission 6-10
Mouse Device Driver Interface 6-12
Call to Device Driver 6-14
Device Driver Functions 6-15
Connector 6-32
Specifications 6-33

Section 7. Displays 7-1
Description 7-3
Vertical Modes 7-4
Signals 7-4
Display Connector 7-5
Power/Audio Cable 7-5
Specifications 7-6

Section 8. Drives 8-1
3.5-Inch 30MB Fixed Disk Drive and Controller 8-3
Description 8-3
Programming 8-5
Registers 8-6
Control Blocks 8-10
Commands 8-19
Connector 8-24
Signal Description 8-25
Specifications 8-26
3.5-Inch 1.44MB Diskette Drive 8-27
Description 8-27
Programming Considerations 8-28
Connector 8-28
Specifications 8-29

Section 9. Modem/RS-232C Interface Cards 9-1
Description 9-3
2400 bps Modem Card 9-3
Description 9-3
Modem Commands and Operation 9-6
Specifications 9-20
RS-232C Interface Card 9-21
Description 9-21
Voltage Interchange Information 9-21
RS-232C Connector 9-22

Section 10. 80286 Microprocessor Instruction Set 10-1
80286 Microprocessor Instruction Set 10-3
Data Transfer 10-3
Arithmetic 10-8
Logic 10-13
String Manipulation 10-16
Repeated String Manipulation 10-17
Control Transfer 10-18
Processor Control 10-23
Protection Control 10-25

Section 11. Characters and Keystrokes 11-1
Available Code Pages 11-3
Character Codes 11-4
Quick Reference 11-11

Section 12. Compatibility 12-1
Introduction 12-3
System Board 12-3

Diskette Drives and Controller 12-4
Fixed Disk Drives and Controller 12-5
Application Guidelines 12-6
Hardware Interrupts 12-6
Software Interrupts 12-7
High-Level Language Considerations 12-7
Assembler Language Programming Considerations 12-7
Multitasking Provisions 12-13
IBM PS/1 Computer Considerations 12-16

Appendix. Options A-1
512KB Memory Expansion A-3
Description A-3
Connector A-3
Audio Card & Joystick A-5
Function Description A-5
Joystick A-5
Digital-to-Analog Converter (DAC) A-6
Analog-to-Digital Converter (ADC) A-6
FIFO Timer A-7
Interrupt Generator A-7
Sound Generator A-7
Serial MIDI A-8
Connector Descriptions A-8
Joystick Schematic Diagram A-12
I/O Address Assignments A-13
Software Registers A-14
Signal Timings A-24
Audio Subsystem A-26
Audio Module Block Diagram A-27
Sound Generator Module Block Diagram A-28
5.25-Inch External Diskette Drive Unit A-29
Buffer Card Functional Diagram A-29
Specifications A-30
Buffer Card Connector Pin-Outs A-31
5.25-Inch Double-Sided Diskette Drive A-32
Description A-32
Interfaces A-33
Specifications A-34
5.25-Inch High Capacity Diskette Drive A-35
Description A-35
Interfaces A-36
Specifications A-37
Adapter Card Unit A-38
Description A-38
Power Supply A-38

Specifications A-40

Glossary X-1

Bibliography X-19

Index X-21

Figures

1-1. IBM Personal System/1 Computer Functional Block Diagram 1-4
1-2. System Board Block Diagram 1-6
1-3. System I/O Address Map 1-7
2-1. Pin Numbers and Signal Assignments for Adapter Card Unit Connector 2-5
2-2. DMA Channel Assignments 2-20
2-3. DMA Address Generation for Channels 0-3 2-20
2-4. DMA Address Generation for Channels 5-7 2-21
2-5. Page Register Addresses 2-21
2-6. DMA I/O Addresses for Memory Addresses, Word Counts, and Command/Status Registers 2-23
2-7. DMA Registers 2-24
2-8. Set/Clear Single Mask Bit Using 8237 Compatible Mode 2-25
2-9. DMA Mask Register Write Using 8237 Compatible Mode 2-25
2-10. 8237 Compatible Mode Register 2-26
2-11. Status Register (Read) 2-26
2-12. Command Register (Write) 2-27
2-13. DMA Channel 2 Programming Example 2-27
2-14. Interrupt Level Assignments by Priority 2-29
2-15. System Timer Block Diagram 2-31
2-16. System Timer/Counter Registers 2-34
2-17. SC - Select Counter, I/O Address Hex 0043 2-34
2-18. RW - Read/Write Counter, I/O Address Hex 0043 2-34
2-19. M - Counter Mode, I/O Address Hex 0043 2-35
2-20. Binary Coded Decimal (BCD) 2-35
2-21. Counter Latch Command 2-35
2-22. Minimum and Maximum Initial Counts, Counters 0 and 2 2-41
2-23. Speaker Tone Generation 2-42
3-1. Keyboard Controller Command Byte, Port Hex 0064 Write 3-5
3-2. Keyboard Controller Status Byte, Port Hex 0064 Read . 3-6
3-3. Command A9 Test Results 3-7
3-4. Command AB Test Results 3-8
3-5. Mouse Data Stream Bit Definitions 3-10
3-6. Receiving Data Timings 3-11
3-7. Sending Data Timings 3-12
3-8. Keyboard/Mouse Signals 3-12
3-9. Keyboard/Mouse Connectors Voltage and Signal Assignments 3-13
3-10. Video Subsystem Block Diagram 3-17

3-11. Graphics Controller Block Diagram 3-20
3-12. Attribute Controller Block Diagram 3-21
3-13. BIOS Video Modes 3-22
3-14. BIOS Double-Scan and Border Support 3-23
3-15. IBM 31.5 KHz Direct-Drive Analog Displays 3-24
3-16. Character/Attribute Format 3-26
3-17. Attribute Byte Functions 3-26
3-18. Attribute Byte Definitions 3-27
3-19. Attribute Byte Colors 3-27
3-20. PEL Format, Modes Hex 4 and 5 3-29
3-21. Video Memory Format 3-29
3-22. Color Selections, Modes Hex 4 and 5 3-29
3-23. PEL Format, Mode Hex 6 3-30
3-24. PEL Bit Definitions 3-31
3-25. Palette Colors 3-32
3-26. Attribute Byte 3-33
3-27. 256KB Video Memory Map 3-35
3-28. VGA Color Compare Operations 3-51
3-29. Data Flow for VGA Memory Write Operations 3-52
3-30. VGA Register Overview 3-53
3-31. General Register Overview 3-54
3-32. Miscellaneous Output Register 3-54
3-33. Display, Vertical Size 3-55
3-34. Clock Select 3 and 2 Bit Definitions 3-55
3-35. Input Status Register 0 3-56
3-36. Input Status Register 1 3-56
3-37. Diagnostic Bits 3-57
3-38. VGA Enable Register, Hex 03C3 3-57
3-39. Sequencer Register Overview 3-58
3-40. Sequencer Address Register 3-58
3-41. Reset Register, Index Hex 00 3-59
3-42. Clocking Mode Register, Index Hex 01 3-59
3-43. Map Mask Register, Index Hex 02 3-60
3-44. Character Map Select Register, Index Hex 03 3-61
3-45. Character Map Select A 3-61
3-46. Character Map Select B 3-62
3-47. Memory Mode Register, Index Hex 04 3-62
3-48. Memory Mode, Chain 4 3-63
3-49. CRT Controller Register Overview 3-64
3-50. CRT Controller Address Register 3-65
3-51. End Horizontal Blanking Register, Index Hex 03 3-66
3-52. Bit Values and Amount of Skew 3-67
3-53. End Horizontal Retrace Register, Index Hex 05 3-67
3-54. CRT Controller Overflow Register, Index Hex 07 3-69
3-55. Preset Row Scan Register, Index Hex 08 3-69
3-56. Maximum Scan Line Register, Index Hex 09 3-70

3-57. Cursor Start Register, Index Hex 0A 3-70
3-58. Cursor End Register, Index Hex 0B 3-71
3-59. Cursor Skew 3-71
3-60. Vertical Retrace End Register, Index Hex 11 3-73
3-61. Underline Location Register, Index Hex 14 3-75
3-62. CRTC Mode Control Register, Index Hex 17 3-76
3-63. Internal Memory Address Counter Wiring to the Output Multiplexer 3-77
3-64. CRT Controller Memory Address Mapping 3-77
3-65. Graphics Controller Register Overview 3-79
3-66. Graphics Address Register 3-79
3-67. Set/Reset Register, Index Hex 00 3-80
3-68. Enable Set/Reset Register, Index Hex 01 3-80
3-69. Color Compare Register, Index Hex 02 3-81
3-70. Data Rotate Register, Index Hex 03 3-81
3-71. Function Select Bit Definitions 3-82
3-72. Read Map Select Register, Index Hex 04 3-82
3-73. Graphics Mode Register, Index Hex 05 3-83
3-74. Write Mode Bit Definitions 3-84
3-75. Miscellaneous Register, Index Hex 06 3-84
3-76. Memory Map Bit Definitions 3-85
3-77. Color Don't Care Register, Index Hex 07 3-86
3-78. Attribute Controller Register Overview 3-87
3-79. Attribute Address Register 3-87
3-80. Palette Registers, Index Hex 00-0F 3-88
3-81. Attribute Mode Control Register, Index Hex 10 3-89
3-82. Color Plane Enable Register, Index Hex 12 3-91
3-83. Color Output Wiring 3-91
3-84. Horizontal PEL Panning Register, Index Hex 13 3-92
3-85. Image Shifting 3-92
3-86. Color Select Register, Index Hex 14 3-93
3-87. Character Table Structure 3-98
3-88. Character Pattern Example 3-99
3-89. Dual-Screen Definition 3-99
3-90. Screen Mapping within the Display Buffer Address Space 3-100
3-91. Video DAC I/O Address Usage 3-101
3-92. Display Vertical Size 3-105
3-93. Display Vertical SYNC, 350 Lines 3-106
3-94. Display Vertical SYNC, 400 Lines 3-106
3-95. Display Vertical SYNC, 480 Lines 3-107
3-96. Display Horizontal Timing, 80 Column with Border . 3-108
3-97. Display Horizontal Timing, 40/80 Column, no Border 3-109
3-98. Display Connector Signals 3-110
3-99. Status Register A (Hex 03F0) 3-112
3-100. Status Register B (Hex 03F1) 3-112

3-101. Digital Output Register (Hex 03F2) 3-113
3-102. Digital Input Register (Hex 03F7) 3-113
3-103. Configuration Control Register (Hex 03F7) 3-113
3-104. Diskette Drive Controller Status Register (Hex 03F4) 3-114
3-105. Command Symbols, Diskette Drive Controller 3-116
3-106. Read Data Command 3-119
3-107. Read Data Result 3-119
3-108. Read Deleted-Data Command 3-120
3-109. Read Deleted-Data Result 3-120
3-110. Read a Track Command 3-121
3-111. Read a Track Result 3-121
3-112. Read ID Command 3-122
3-113. Read ID Result 3-122
3-114. Write Data Command 3-123
3-115. Write Data Result 3-123
3-116. Write Deleted-Data Command 3-124
3-117. Write Deleted-Data Result 3-124
3-118. Format a Track Command 3-125
3-119. Format a Track Result 3-125
3-120. Recalibrate Command 3-125
3-121. Sense Interrupt Status Command 3-126
3-122. Sense Interrupt Status Result 3-126
3-123. Specify Command 3-126
3-124. Sense Drive Status Command 3-127
3-125. Sense Drive Status Result 3-127
3-126. Seek Command 3-127
3-127. Invalid Command Result 3-127
3-128. Status Register 0 (ST 0) 3-128
3-129. Status Register 1 (ST 1) 3-129
3-130. Status Register 2 (ST 2) 3-130
3-131. Status Register 3 (ST 3) 3-130
3-132. Diskette Drive Controller Connector Voltage and Signal Assignments 3-134
3-133. Serial Port Block Diagram 3-136
3-134. Serial Port, Data Format 3-136
3-135. Serial Port Register Addresses 3-137
3-136. Baud Rates at 1.846154 MHz 3-138
3-137. Interrupt Enable Register (Hex 3F9) 3-139
3-138. Interrupt Identification Register (Hex 3FA) 3-140
3-139. Interrupt Control Functions 3-140
3-140. Line Control Register (Hex 3FB) 3-141
3-141. Stop Bits 3-142
3-142. Word Length 3-142
3-143. Modem Control Register (Hex 3FC) 3-143
3-144. Line Status Register (Hex 3FD) 3-144
3-145. Modem Status Register (Hex 3FE) 3-146

3-146. Serial Interface Connector 3-148
3-147. Parallel Port Controller Block Diagram 3-149
3-148. Parallel Port Configuration 3-150
3-149. Parallel Port Address Assignments 3-150
3-150. Parallel Port Extended Mode Configurations 3-150
3-151. Status Port 3-152
3-152. Parallel Control Port 3-153
3-153. Parallel Port Timing Sequence 3-154
3-154. Data and Interrupt Signals 3-154
3-155. Control Signals 3-154
3-156. Parallel Port Connector Signal and Voltage Assignments 3-155
3-157. Memory Option Connector Pin Assignments 3-157
3-158. System Board Memory Map 3-157
3-159. System Board Memory Enable 3-158
3-160. RTC/CMOS RAM Address Map 3-158
3-161. Real-Time Clock (Addresses Hex 000 - 00D) 3-160
3-162. System Control Port B, Write Operations 3-165
3-163. System Control Port B, Read Operations 3-165
3-164. RTC/CMOS and NMI Mask 3-166
3-165. System Control Port A 3-167
4-1. Power Card Connectors 4-5
4-2. Power/Audio Cable Connector 4-6
5-1. Key Numbering for the 101-Key Keyboard 5-4
5-2. Key Numbering for the 102-Key Keyboard 5-5
5-3. Keyboard Commands from the System 5-23
5-4. Set All Keys Commands 5-25
5-5. Set Key Type Commands 5-25
5-6. Set/Reset Status Indicators 5-26
5-7. Typematic Rate 5-27
5-8. Keyboard Commands to the System 5-28
5-9. Keyboard Scan Codes, Set 1 (Part 1 of 5) 5-30
5-10. Keyboard Scan Codes, Set 1 (Part 2 of 5) 5-31
5-11. Keyboard Scan Codes, Set 1 (Part 3 of 5) 5-31
5-12. Keyboard Scan Codes, Set 1 (Part 4 of 5) 5-31
5-13. Keyboard Scan Codes, Set 1 (Part 5 of 5) 5-32
5-14. Keyboard Scan Codes, Set 2 (Part 1 of 5) 5-33
5-15. Keyboard Scan Codes, Set 2 (Part 2 of 5) 5-34
5-16. Keyboard Scan Codes, Set 2 (Part 3 of 5) 5-34
5-17. Keyboard Scan Codes, Set 2 (Part 4 of 5) 5-34
5-18. Keyboard Scan Codes, Set 2 (Part 5 of 5) 5-35
5-19. Keyboard Scan Codes, Set 3 5-36
5-20. Keyboard Data Stream Bit Definitions 5-40
5-21. Character Codes 5-42
5-22. Special Character Codes 5-44
5-23. Keyboard Extended Functions 5-45

5-24. Keyboard Connector Signal and Voltage Assignments 5-50
6-1. Mouse Commands 6-4
6-2. Sampling Rate 6-6
6-3. Status Request Format 6-7
6-4. Set Resolution 6-7
6-5. Set Scaling 2:1 6-8
6-6. Data Packet Report Format 6-9
6-7. Data Frame 6-12
6-8. Data Passed to Driver 6-14
6-9. Function 0: Installed Flag and Reset 6-15
6-10. Device Driver Variables 6-15
6-11. Mouse Initial Values 6-15
6-12. Function 1: Show Cursor 6-16
6-13. Function 2: Hide Cursor 6-16
6-14. Function 3: Get Position and Button Status 6-16
6-15. Function 4: Set Cursor Position 6-17
6-16. Function 5: Get Button Press Information 6-17
6-17. Function 6: Get Button Release Information 6-18
6-18. Function 7: Set Minimum and Maximum Horizontal Position 6-18
6-19. Function 8: Set Minimum and Maximum Vertical Position 6-19
6-20. Function 9: Set Graphics Cursor Block 6-20
6-21. Graphics Modes Less Than 7 6-20
6-22. Graphics Modes Greater Than 7 6-20
6-23. Function 10: Set Text Cursor 6-21
6-24. Function 11: Read Motion Counters 6-21
6-25. Function 12: Set User-Defined Subroutine Input Mask . 6-22
6-26. Call Mask Bit Definitions 6-22
6-27. Function 13: Light Pen Emulation Mode On 6-23
6-28. Function 14: Light Pen Emulation Mode Off 6-23
6-29. Function 15: Set xy/PEL Ratio 6-23
6-30. Function 16: Conditional Off 6-24
6-31. Function 19: Set Double Speed Threshold 6-24
6-32. Function 20: Swap User Interrupt Vector 6-24
6-33. Call Mask Bit Definitions 6-25
6-34. Register Information 6-25
6-35. Function 21: Query Save State Storage Requirements 6-26
6-36. Function 22: Save Mouse Driver State 6-26
6-37. Function 23: Restore Mouse Driver State 6-26
6-38. Function 24: Set User-Defined Subroutine Input Mask . 6-27
6-39. Register Conventions 6-27
6-40. Event Mask Bit Definitions 6-27
6-41. Function 25: Get User Alternate Interrupt Vector 6-28
6-42. Function 26: Set Mouse Sensitivity 6-28
6-43. Function 27: Get Mouse Sensitivity 6-29

6-44. Function 29: Set CRT Page Number 6-29
6-45. Function 30: Get CRT Page Number 6-29
6-46. Function 31: Disable Mouse Driver 6-30
6-47. Function 32: Enable Mouse Driver 6-30
6-48. Function 33: Software Reset 6-30
6-49. Function 36: Get Driver Version, Mouse Type, and IRQ Number 6-31
6-50. Mouse Connector Pin Assignments 6-32
7-1. Display Vertical Modes 7-4
7-2. Display Connector Signals 7-5
7-3. Power/Audio Cable Connector Pin Definition 7-5
8-1. Fixed Disk and Controller 8-4
8-2. Port Addresses 8-5
8-3. Attachment Status Register 8-6
8-4. Attachment Control Register 8-7
8-5. Interrupt Status Register 8-8
8-6. Attention Register 8-9
8-7. Command Control Block 8-10
8-8. Command Specify Block 8-12
8-9. Sense Summary Block 8-13
8-10. Format Control Block 8-17
8-11. Read Command 8-19
8-12. Read Check Command 8-19
8-13. Read Extended Command 8-20
8-14. Read ID Command 8-20
8-15. Recalibrate Command 8-20
8-16. Write Data Command 8-21
8-17. Write Verify Command 8-21
8-18. Write Extended Command 8-22
8-19. Format Disk Command 8-22
8-20. Seek Command 8-23
8-21. Format Track Command 8-23
8-22. Fixed Disk Connector 8-24
8-23. DC Power Specifications (for both drives) 8-26
8-24. Connector Pin Assignments 8-28
9-1. S14 Options Register 9-17
9-2. S16 Modem Test Options Register 9-17
9-3. S21 Modem Options Register 9-18
9-4. S22 Options Register 9-18
9-5. S23 Options Register 9-19
9-6. S27 Options Register 9-19
9-7. Voltage Levels 9-21
9-8. Serial Port Connector Pin Assignments 9-22
10-1. 2-Bit Register Field 10-27
10-2. 3-Bit Register Field 10-27
12-1. Diskette Drive Read, Write, and Format Capabilities . 12-4

12-2. Write and Format Head Settle Time 12-11
12-3. Functional Code Assignments 12-15
12-4. Machine Model Byte 12-17
A-1. 512KB Memory Expansion Signal Names A-4
A-2. Audio Card to System Board Voltage and Signal Assignments A-9
A-3. Buffer Card Functional Diagram A-29
A-4. Diskette Requirements A-32
A-5. Control Interface (P1/J1) A-33
A-6. DC Power Interface (P2/J2) A-33
A-7. Diskette Requirements A-35
A-8. Control Interface (P1/J1) A-36
A-9. DC Power Interface (P2/J2) A-36
A-10. I/O Channel Connectors. A-39

Section 1. System Description

Description 1-3
System Board Features 1-4
System Board Locations 1-5
System Board Block Diagram 1-6
System I/O Address Map 1-7
Specifications 1-8

Description

The IBM Personal System/1™ computer is a tabletop 80286 computer system consisting of a system unit, a ™PHOTO GRAPHIC Display (VGA), a keyboard, and a mouse. The system unit contains IBM DOS 4.01 in ROM; 512KB RAM; 1.44MB, 3.5-inch diskette drive; and a parallel printer port. Depending on the country where purchased, a 2400 bps modem or a RS-232C interface is included in the system unit. For example, U.S. machines contain a 2400 bps modem. The system unit supports either a second 1.44MB, 3.5-inch diskette drive or a 30MB fixed disk. The 30MB fixed disk is standard on some models.

The display contains the ac power switch for the entire system, a speaker with volume control, and a headphone jack (32 ohms). The system unit obtains a dc supply voltage from the display to support the system unit and its options.

Options include the following:

- 512KB RAM Card provides a total of 1MB RAM within the system unit. This memory card is standard on some models.
- Audio Card and Joystick provides digital audio record and playback, a 3-voice and noise generator, Musical Instrument Digital Interface (MIDI), and joysticks.
- 5.25-inch Diskette Unit provides either a 360KB or a 1.2MB diskette drive. Only one 3.5-inch diskette drive is supported when this option is installed.
- Adapter Card Unit supports two adapter cards up to 280mm (11 inches) long and one card up to 241mm (9.5 inches).

Figure 1-1 on page 1-4 shows a block diagram of the system unit and options.

PHOTO GRAPHIC is a trademark of the International Business Machines Corporation.

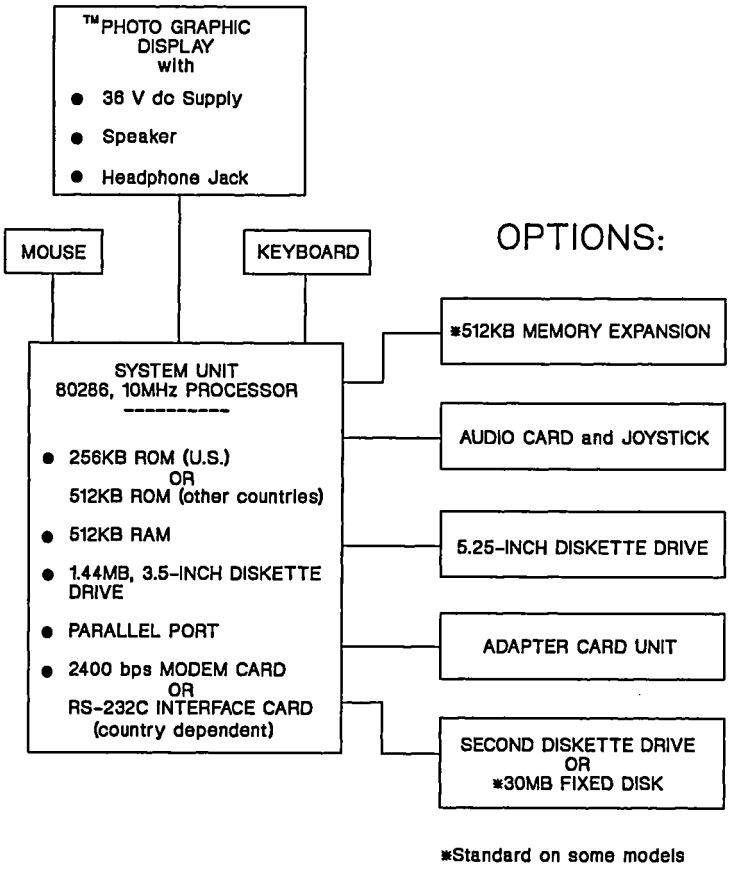


Figure 1-1. IBM Personal System/1 Computer Functional Block Diagram

System Board Features

The system board incorporates very large scale integration (VLSI) modules and surface-mount technology (SMT). The following is a list of the features on the system board:

- 80286 System Microprocessor
- Microprocessor support
  - Seven-channel direct memory access (DMA) controller
  - 15-level interrupt system
  - System clock (10 MHz)
  - Three programmable timers
- ROM, 256KB or 512KB (KB = 1024 bytes) depending on country (U.S. has 256KB)
- RAM, 512KB expandable to 1MB (MB = 1,048,576 bytes)

- 16-bit data bus
- 24-bit address bus
- Real-time clock CMOS RAM
  - Clock
  - Calendar
  - CMOS RAM
- Integrated video graphics array subsystem (VGA)
- Serial port (TTL voltage level) for internal connection to 2400 bps modem card or RS-232C interface card
- Parallel port
- Keyboard/mouse controller and connectors
- Diskette drive controller and connector
- Expansion connector for the adapter card unit
- Fixed disk drive connector
- Audio card connector
- IBM DOS 4.01 in ROM.

System Board Locations

The following shows the connector numbering of the system board.

Diskette drive connector (J1)
Serial interface connector (J2)
Mouse connector (J3)
Keyboard connector (J4)
Memory option connector (J5)
Parallel port connector (J6)
Audio card connector (J7)
Adapter card unit connector (J8)
Power card connector (J9)
Power/Audio cable connector (J10)
Fixed disk drive connector (J11)
Fan connector (J12)
Display connector (J13)

System Board Block Diagram

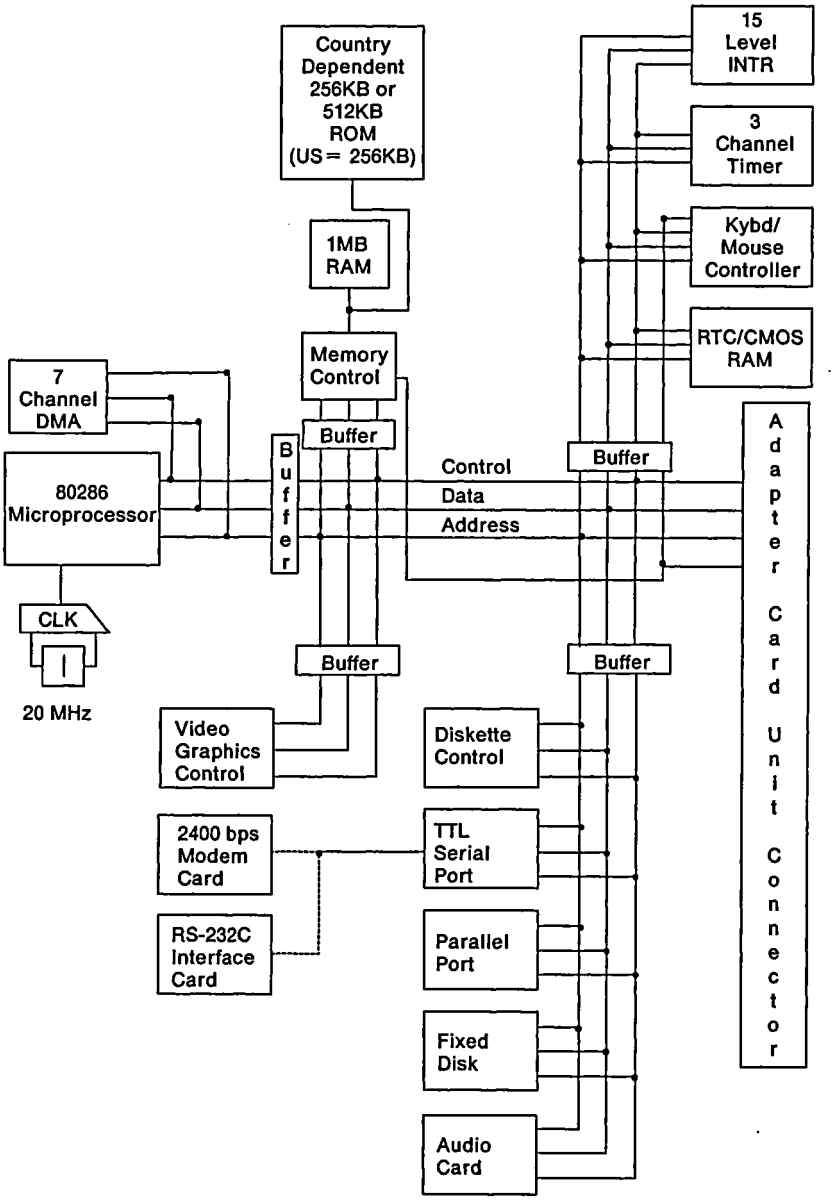


Figure 1-2. System Board Block Diagram

System I/O Address Map

Figure 1-3 shows the hexadecimal addresses for each of the system board I/O functions.

Figure 1-3. System I/O Address Map

Hex Addresses	I/O Function
0000 - 001F	DMA Controller, Channels 0-3
0020, 0021	Interrupt Controller 1
0040 - 0043	System Timers
0060	Keyboard, Mouse
0061 - 006F (odd bytes)	System Control Port B
0064	Keyboard, Mouse
0070, 0071	RTC/CMOS and NMI Mask
0081, 0082, 0083, 0087	DMA Page Registers (0 - 3)
0089, 008A, 008B, 008F	DMA Page Registers (5 - 7)
0090	Central Arbitration Control Port
0091	Card Select Feedback
0092	System Control Port A
0094	System Board Enable/Setup Register
0096	Adapter Enable/Setup Register
00A0 - 00A1	Interrupt Controller 2
00C0 - 00DE (even bytes)	DMA Controller, Channels 4-7
0100 - 0105	Programmable Option Select
x200-x207	Audio Card
0320-032F	Fixed Disk
x330-x331	Audio Card
0278 - 027A	Parallel Port 3
0378 - 037A	Parallel Port 2
03BC - 03BE	Parallel Port 1
03B4 - 03BA	Video Subsystem
03CD - 03EF	Video Subsystem
03F0 - 03F7	Diskette Drive Controller
03F8 - 03FF	Serial Port 1 (2400 bps modem or RS-232C Country dependent - U.S. and Canada have modem)

Specifications

The following are specifications for the system unit.

Size

- Width: 274 millimeters (10.8 inches)
- Depth: 351 millimeters (13.8 inches)
- Height: 82 millimeters (3.3 inches).

Weight

- Single Diskette: 4.2 kilograms (9.25 pounds)
- Fixed Disk: 4.8 kilograms (10.5 pounds).

Cables

- Keyboard: 1.8 meters (6 feet)
- Mouse: 2.1 meters (7 feet)
- Phone: 3 meters (10 feet).

Temperature

- System On: 16 to 32 degrees C (60 to 95 degrees F)
- System Off: 10 to 43 degrees C (50 to 110 degrees F)
- Shipping: -40 to 60 degrees C (-40 to 140 degrees F).

Humidity

- 8% to 80%.

Altitude

- Maximum Altitude: 2133.6 meters (7000 feet).

Heat Output

- 358 BTU/hour.

Acoustical

- Operating: 39 dBA average sound pressure at operator position
- Idling: 4.7 Bels average sound power.

Electrical

- Low Range Voltage Systems: 90 to 137 V
- High Range Voltage Systems: 180 to 265 V.

Electro-Magnetic Compatibility

- FCC Class B.

Section 2. System Board

Description 2-3
System Microprocessor 2-3
  Real Address Mode 2-3
  Protected Virtual Address Mode 2-4
  Performance 2-4
Adapter Card Channel 2-5
  Connector 2-5
  Signal Description 2-7
  Signal Timings 2-11
DMA Controller 2-19
  Data Transfers Between Memory and I/O Devices 2-20
  Byte Pointer 2-20
  DMA Channels 2-20
    Address Generation for DMA Channels 0 through 3 2-20
    Address Generation for DMA Channels 5 through 7 2-21
  Page Register Addresses 2-21
  DMA I/O Address Map 2-23
  DMA Registers 2-24
    Memory Address Register 2-24
    Transfer Count Register 2-24
    Mask Register 2-25
    Mode Register 2-26
    Status Register 2-26
Interrupts 2-28
  Non-Maskable Interrupt 2-28
  Interrupt Assignments 2-29
System Timers 2-31
  Channel 0 - System Timer 2-32
  Channel 1 - Refresh Request Generator 2-32
  Channel 2 - Tone Generation for Speaker 2-32
  Timers 0, 1, and 2 2-32
  Programming the System Timers 2-32
  Counter Write Operations 2-33
  Counter Read Operations 2-33
  Registers 2-34
    Channel 0 (Hex 0040) and Channel 2 (Hex 0042) Count Registers 2-34
    Control Byte Register - Channel 0, 1, or 2 (Hex 0043) 2-34
  Counter Latch Command 2-35
  System Timer Modes 2-36
    Mode 0 - Interrupt on Terminal Count 2-36
    Mode 1 - Hardware Retriggerable One-Shot 2-37

Mode 2 - Rate Generator 2-37
Mode 3 - Square Wave 2-38
Mode 4 - Software Retriggerable Strobe 2-40
Mode 5 - Hardware Retriggerable Strobe 2-40
Operations Common to All Modes 2-41
Speaker 2-42

Description

This section describes the system board components and how they interact. Hardware descriptions and programming information are provided to acquaint hardware designers and programmers with the operation of the system.

System Microprocessor

The 80286 microprocessor has the following:

- 10-MHz clock operation
- 24-bit address
- 16-bit data interface
- Instruction set, including string I/O
- Hardware integer multiply and divide operations
- Two operating modes
  - 8086-compatible Real Address mode
  - Protected Virtual Address mode
- 16MB (MB = 1,048,576 or 2^{20} bytes) of physical address space.

Real Address Mode

In the Real Address mode, the system microprocessor address space is a contiguous array of up to 1MB. The system microprocessor addresses memory by generating 20-bit physical addresses.

The segment portion of the pointer is interpreted as the upper 16 bits of a 20-bit segment address. The lower 4 bits of the 20-bit segment address are always 0. Therefore, segment addresses begin on multiples of 16 bytes.

All segments in the Real Address mode are 64KB in size and can be read, written, or executed. An exception or interrupt can occur if data operands or instructions attempt to wrap around the end of a segment (for example, a word with its least significant byte at offset hex FFFF and its most significant byte at hex 0000). If, in the Real Address mode, the information contained in the segment does not use the full 64KB, the unused end of the segment can be overlaid by another segment to reduce physical memory requirements.

Protected Virtual Address Mode

The Protected Virtual Address mode (Protected mode) offers extended physical and virtual memory address space, memory protection mechanisms, and new operations to support operating systems and virtual memory.

The Protected mode provides a 1-gigabyte virtual address space per task mapped into a 16MB physical address space. The virtual address space can be larger than the physical address space, because any use of an address that does not map to a physical memory location will cause a restartable exception.

As in the Real Address mode, the Protected mode uses 32-bit pointers, consisting of 16-bit selector and offset components. The selector specifies an index into a memory-resident table rather than the upper 16 bits of a real memory address. The 24-bit base address of the desired segment is obtained from the tables in memory. The 16-bit offset is added to the segment base address to form the physical address. The tables are automatically referred to by the system microprocessor whenever a segment register is loaded with a selector. All instructions that load a segment register refer to the memory-based tables without additional program support. The memory-based tables contain 8-byte values.

Performance

The 80286 microprocessor operates at 10 MHz for a clock cycle time of 100 nanoseconds. The system inserts one wait state whenever it accesses system board RAM, which results in a 300-nanosecond, 16-bit memory cycle time. The system inserts a minimum of one wait state whenever it does a system board I/O operation, which results in a minimum I/O cycle time of 300-nanoseconds. The system inserts two wait states whenever it accesses system board ROM, which results in a 400-nanosecond 16-bit ROM cycle time.

The bandwidth for memory refresh is approximately 3.0%.

The bus cycle time for memory operations is 1 microsecond for direct memory access (DMA) operations, with the DMA controller operating at 5 MHz with no additional wait states inserted. I/O operations and system board memory operations have one wait state inserted.

Adapter Card Channel

The system board has a 112-pin connector in interface with the adapter card unit. The adapter card unit contains three I/O channel connectors to support two 11-inch adapter boards and one 9.5-inch adapter board.

Connector

The adapter card unit connector on the system board contains the I/O channel signals, power supply voltages, and power supply control (-FAULT).

The I/O channel signals are designed to provide sufficient power for the adapters assuming two low-power Schottky (LS) loads per slot. IBM adapters typically use only one load per adapter.

The pin numbering and signal assignments for the system board are:

Figure 2-1 (Page 1 of 2). Pin Numbers and Signal Assignments for Adapter Card Unit Connector

Pin	Signal	Pin	Signal
1	Ground	57	+36 V
2	Ground	58	Ground
3	-I/O CH CK	59	SD15
4	RESET DRV	60	SD14
5	Ground	61	SD13
6	+5 V	62	SD12
7	IRQ9	63	SD11
8	DRQ2	64	SD10
9	-12 V	65	SD9
10	-OWS	66	SD8
11	+12 V	67	Ground
12	+12 V	68	-MEMW
13	Ground	69	-MEMR
14	I/O CH RDY	70	LA17
15	AEN	71	LA18
16	-SMEMW	72	LA19
17	-SMEMR	73	LA20
18	-IOW	74	LA21
19	-IOR	75	LA22
20	Ground	76	LA23
21	-DACK3	77	-SBHE
22	DRQ3	78	Ground
23	-DACK1	79	SA19
24	DRQ1	80	SA18
25	-REFRESH	81	SA17
26	CLK	82	SA16
27	Ground	83	Ground

Figure 2-1 (Page 2 of 2). Pin Numbers and Signal Assignments for Adapter Card Unit Connector

Pin	Signal	Pin	Signal
28	IRQ7	84	SA15
29	IRQ6	85	SA14
30	IRQ5	86	SA13
31	IRQ4	87	SA12
32	IRQ3	88	Ground
33	-DACK2	89	SA11
34	TC	90	SA10
35	BALE	91	SA9
36	Ground	92	SA8
37	OSC	93	Ground
38	-MEMCS16	94	SA7
39	-I/O CS16	95	SA6
40	IRQ15	96	SA5
41	IRQ14	97	SA4
42	-FAULT	98	Ground
43	IRQ11	99	SA3
44	IRQ10	100	SA2
45	-DACK0	101	SA1
46	DRQ0	102	SA0
47	-DACK5	103	Ground
48	DRQ5	104	SD0
49	Ground	105	SD1
50	-DACK6	106	SD2
51	DRQ6	107	SD3
52	-DACK7	108	Ground
53	DRQ7	109	SD4
54	-MASTER	110	SD5
55	Ground	111	SD6
56	+36 V	112	SD7

Signal Description

The following is a description of the I/O channel signals. All lines are TTL-compatible. The (I), (O), or (I/O) notation refers to input, output, or input and output.

SA0—SA19 (O): Address bits 0 through 19 are used to address memory and I/O devices within the system.

These 20 address lines, in addition to LA17 through LA23, allow access of up to 16MB of memory. SA0 through SA19 are gated on the system bus when BALE is high and are latched on the falling edge of BALE. These signals are generated by the microprocessor or DMA controller. They also may be driven by other microprocessors or DMA controllers that reside on the I/O channel. Only the lower 16 signals are used in I/O addressing, and all 16 should be decoded by I/O devices. SA0 is the least significant and SA19 is the most significant.

LA17—LA23 (O): These signals (unlatched) are used to address memory and I/O devices within the system. They give the system up to 16MB of addressability. These signals are valid when BALE is high. LA17 through LA23 are not latched during microprocessor cycles and therefore do not stay valid for the whole cycle. Their purpose is to generate memory decodes for one wait state memory cycles. These decodes should be latched by I/O adapters on the falling edge of BALE. These signals also may be driven by other microprocessors or DMA controllers that reside on the I/O channel.

AEN (O): The address enable signal is used to de-gate the microprocessor and other devices from the I/O channel to allow DMA transfers to take place. When this signal is active, the DMA controller has control of the address bus, data bus, and Read and Write command signals, (MEMR, SMEMR, IOR, and MEMW, SMEMW, IOW). When this signal is inactive, the microprocessor has control. This signal should be part of the adapter-select decode to prevent incorrect adapter selects during DMA operations.

BALE (O) (Buffered): The buffered address latch enable signal is provided by the bus controller and is used on the system board to latch valid addresses and memory decodes from the microprocessor.

Addresses are latched with the falling edge of BALE. BALE is forced high during DMA cycles. It is available to the I/O channel as an indicator of a valid microprocessor or DMA address when used with AEN.

CLK (O): System clock has a frequency of 10 MHz and a 50% duty cycle. This signal should only be used for synchronization. It is not intended for uses requiring a fixed frequency.

SD0—SD15 (I/O): Data bits 0 through 7 provide data bus bits 0 to 7 for the microprocessor, memory, and I/O devices.

All 8-bit devices on the I/O channel should use SD0 through SD7 for communications to the microprocessor. The 16-bit devices will use SD0 through SD15. To support 8-bit devices, the data on SD8 through SD15 will be gated to SD0 through SD7 during 8-bit transfers to these devices; 16-bit microprocessor transfers to 8-bit devices will be converted to two 8-bit transfers.

-DACK0 to -DACK3 and -DACK5 to -DACK7 (O): -DMA acknowledge 0 to 3 and 5 to 7 are used to acknowledge DMA requests (DRQ0 to DRQ3 and DRQ5 to DRQ7). They are active low.

DRQ0—DRQ3 and DRQ5—DRQ7 (I): DMA requests 0 through 3 and 5 through 7 are asynchronous channel requests used by peripheral devices and the I/O channel microprocessors to gain DMA services. They are prioritized with DRQ0 being the highest and DRQ7 being the lowest. A request is generated by bringing a DRQ line to an active level. A DRQ line must be held high until the corresponding DMA acknowledge line goes active. DRQ0 through DRQ3 will perform 8-bit DMA transfers and DRQ5 through DRQ7 will perform 16-bit transfers. DRQ4 is used on the system board and is not available on the I/O channel.

-I/O CH CK (I): The I/O channel check signal generates an NMI to provide parity information about memory or a device on the I/O channel. It is driven active to indicate an uncorrectable error and held active for at least two clock cycles.

I/O CH RDY (I): The I/O channel ready signal is normally active (ready) and is pulled inactive (not ready) by a memory or I/O device to lengthen I/O or memory cycles. It allows slower devices to attach to the I/O channel with a minimum of difficulty. Any slow device using this signal should drive it inactive immediately after detecting a valid address and a Read or Write command. For every clock cycle this signal is inactive, one wait state is added. This signal should not be held inactive longer than 17 clock cycles.

-IOR (O): The I/O read signal instructs an I/O device to drive its data onto the data bus. This signal is driven by the microprocessor or the DMA controller.

-IOW (O): The I/O write signal instructs an I/O device to read the data on the data bus. This signal is driven by the microprocessor or the DMA controller.

IRQ3–IRQ7, IRQ9–IRQ11, and IRQ14–IRQ15 (I): Interrupt request signals 3 through 7, 9 through 11, and 14 through 15 are used to signal the microprocessor that an I/O device requires attention. They are prioritized with IRQ9 through IRQ11 and IRQ14 through IRQ15 having highest priority (IRQ9 is the highest) and IRQ3 through IRQ7 having the lowest (IRQ7 is the lowest). An interrupt request is generated when an IRQ line is raised from low to high. The line must be held high until the microprocessor acknowledges the interrupt request (Interrupt Service routine). Interrupts 12 and 13 are used on the system board and are not available on the I/O channel. Interrupt 8 is used for the real-time clock.

-SMEMR (O) -MEMR (I/O): These signals instruct memory devices to drive data onto the data bus. -SMEMR is active only when the memory decode is within the low 1MB of memory space. -MEMR is active on all memory read cycles. -MEMR can be driven by any microprocessor or DMA controller in the system. -SMEMR is derived from -MEMR and the decode of the low 1MB of memory. When a microprocessor on the I/O channel wishes to drive -MEMR, it must have the address lines valid on the bus for one clock period before driving -MEMR active. Both signals are active low.

-SMEMW (O) -MEMW (I/O): These signals instruct the memory devices to store the data present on the data bus. -SMEMW is active only when the memory decode is within the low 1MB of the memory space. -MEMW is active on all memory write cycles. -MEMW can be driven by any microprocessor or DMA controller in the system. -SMEMW is derived from -MEMW and the decode of the low 1MB of memory. When a microprocessor on the I/O channel wishes to drive -MEMW, it must have the address lines valid on the bus for one system clock period before driving -MEMW active. Both signals are active low.

-REFRESH (I/O): The memory refresh signal indicates a refresh cycle and can be driven by a microprocessor on the I/O channel.

OSC (O): The oscillator signal is a high-speed clock with a 70 ns period (14.31818 MHz) and a 50% duty cycle. This signal is not synchronous with the system clock.

RESET DRV (O): The reset drive signal is used to reset or initialize system logic upon power on or during a low line-voltage.

TC (O): Terminal Count provides a pulse when the terminal count for any DMA channel is reached.

-MASTER (I): This signal is used with the DRQ line to gain control of the system. A processor or DMA controller on the I/O channel can issue a DRQ to a DMA channel in cascade mode and receive a -DACK. Upon receiving the -DACK, an I/O microprocessor can pull -MASTER low, which will allow it to control the system address, data, and control lines (a condition known as tri-state). After -MASTER is low, the I/O microprocessor must wait one system clock period before driving the address and data lines, and two clock periods before issuing a Read or Write command. If the signal is held low for more than 15 microseconds, system memory can be lost because of the lack of refresh.

-MEM CS16 (I): The MEM 16 chip select signal indicates to the system that the present data transfer is a one wait state 16-bit memory cycle. It must be derived from the decode of LA17 through LA23. -MEM CS16 should be driven with an open-collector or tri-state driver capable of sinking 20mA.

-I/O CS16 (I): The -I/O 16-bit chip select signal indicates to the system that the present data transfer is a 16-bit, one wait state, I/O cycle. It is derived from an address decode. -I/O CS16 is active low and should be driven with an open-collector or tri-state driver capable of sinking 20mA.

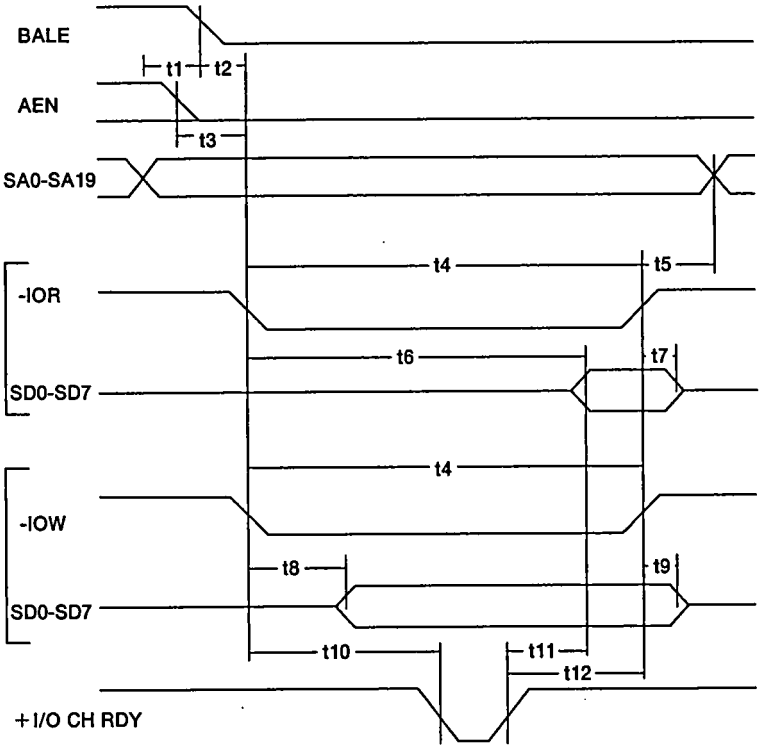
-0WS (I): The zero wait state signal tells the microprocessor that it can complete the present bus cycle without inserting any additional wait states. In order to run a memory cycle to a 16-bit device without wait states, -0WS is derived from an address decode. In order to run a memory cycle to an 8-bit device with a minimum of two wait states, -0WS should be driven active one system clock after the Read or Write command is active and gated with the address decode for the device. Memory Read and Write commands to an 8-bit device are active on the falling edge of the system clock. -0WS is active low and should be driven with an open collector or tri-state driver capable of sinking 20mA.

-SBHE (I/O): The system bus high enable signal indicates a transfer of data on the high-order byte of the data bus, SD8 - SD15. Sixteen-bit devices use SBHE to condition data bus buffers tied to SD8 through SD15.

Signal Timings

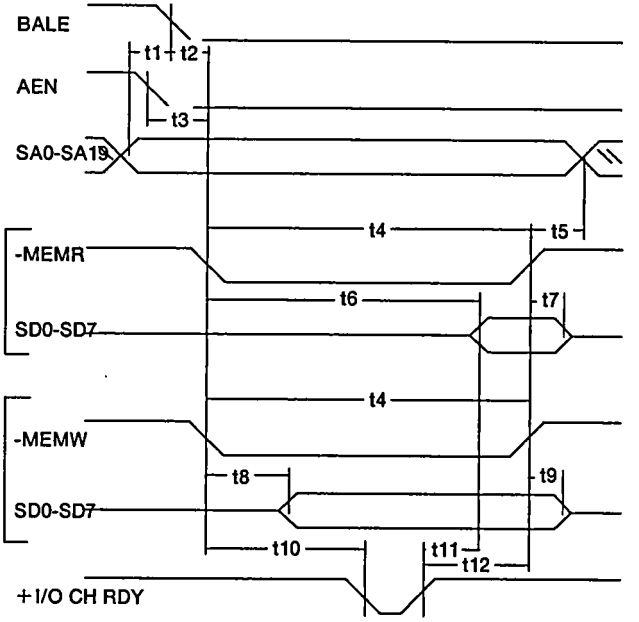
The following diagrams show the signal timings for I/O and memory operations.

8-Bit I/O Bus Cycles



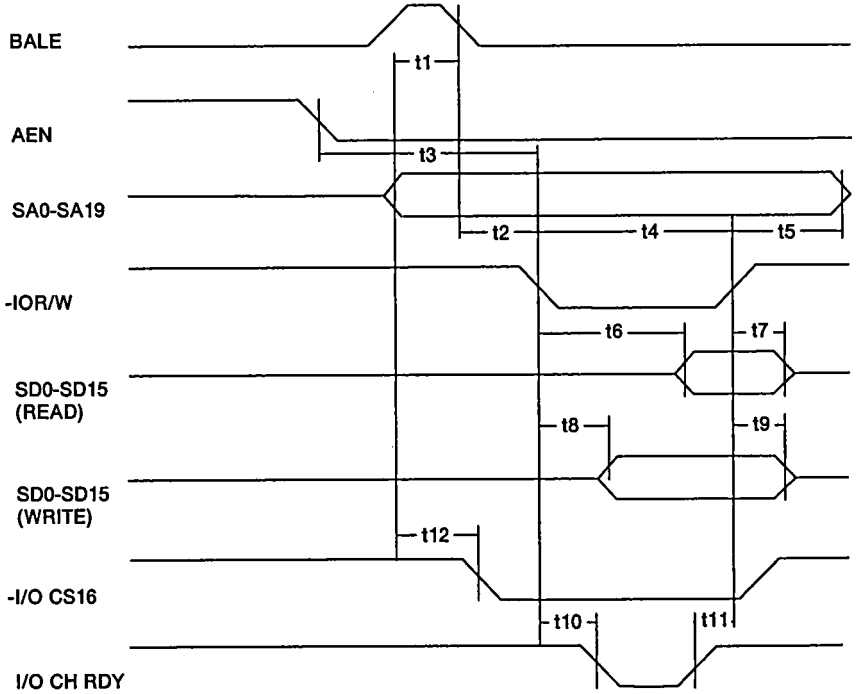
Symbol	Description	Min (ns)	Max (ns)
t1	Address valid to BALE inactive	10	
t2	BALE inactive to Command active	25	
t3	Command active from AEN inactive	150	
t4	Command pulse width	415	
t5	Address hold from Command inactive	20	
t6	Data valid from Read active		372
t7	Data hold from Read inactive	0	
t8	Data valid from Write active		55
t9	Data hold from Write inactive	15	
t10	I/O CH RDY inactive from Command active		285
t11	Read Data valid from I/O CH RDY active		0
t12	Command inactive from I/O CH RDY active	130	

8-Bit Memory Bus Cycles



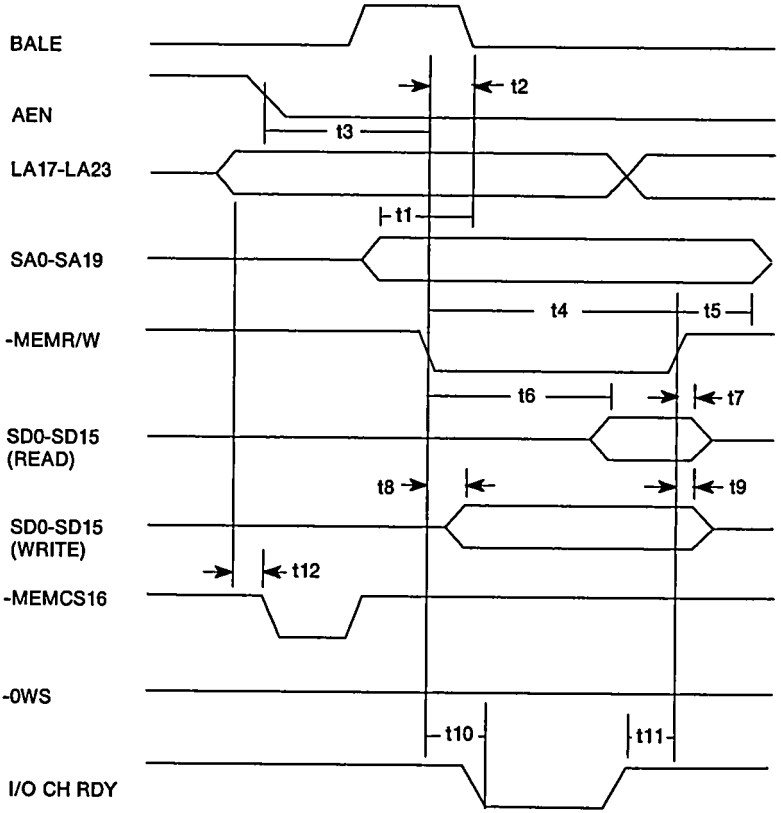
Symbol	Description	Min (ns)	Max (ns)
t1	Address valid to BALE inactive	10	
t2	BALE inactive to Command active	25	
t3	Command active from AEN inactive	150	
t4	Command pulse width	415	
t5	Address hold from Command inactive	20	
t6	Data valid from Read active		372
t7	Data hold from Read inactive	0	
t8	Data valid from Write active		55
t9	Data hold from Write inactive	15	
t10	I/O CH RDY inactive from Command active		285
t11	Read Data valid from I/O CH RDY active		0
t12	Command inactive from I/O CH RDY active	130	

16-Bit I/O Bus Cycles



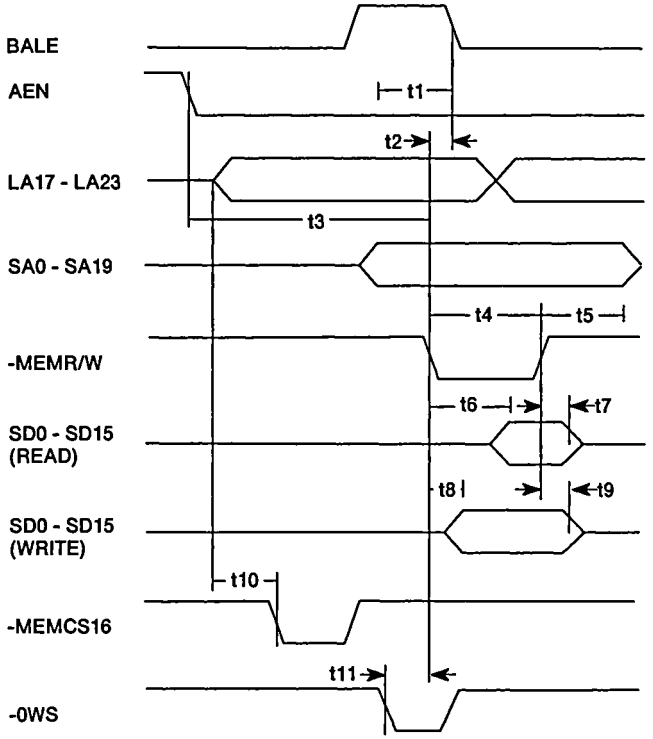
Symbol	Description	Min (ns)	Max (ns)
t1	Address valid to BALE inactive	10	
t2	BALE inactive to Command active	25	
t3	Command active from AEN inactive	150	
t4	Command pulse width	115	
t5	Address hold from Command inactive	20	
t6	Data valid from Read active		72
t7	Data hold from Read inactive	0	
t8	Data valid from Write active		55
t9	Data hold from Write inactive	15	
t10	I/O CH RDY inactive from Command active		20
t11	Command Inactive from I/O CH RDY active	130	
t12	-I/O CS16 active from SA0-SA19 valid		62

16-Bit Memory Bus Cycles (1 Wait State)



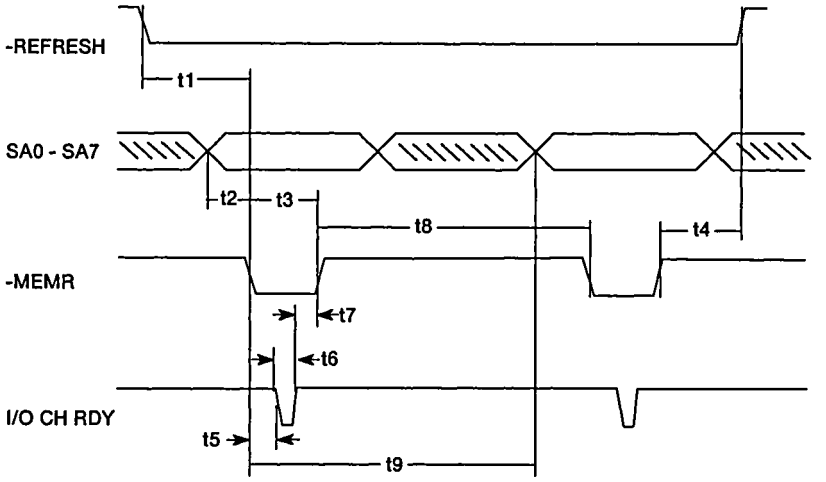
Symbol	Description	Min (ns)	Max (ns)
t1	Address valid to BALE inactive	10	
t2	Command active to BALE inactive	25	
t3	Command active from AEN inactive	150	
t4	Command pulse width	165	
t5	Address hold from Command inactive	20	
t6	Data valid from Read active		122
t7	Data hold from Read inactive	0	
t8	Data valid from Write active		35
t9	Data hold from Write inactive	15	
t10	I/O CH RDY inactive from Command active		35
t11	Command Inactive from I/O CH RDY active	130	
t12	-MEMCS16 active from LA17-LA23 valid		43

16-Bit Memory Bus Cycles (0 Wait State)



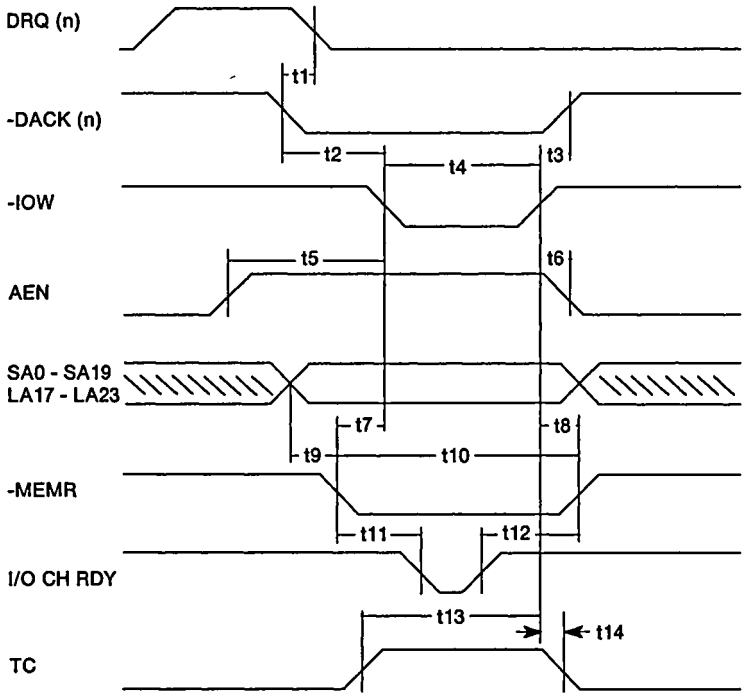
Symbol	Description	Min (ns)	Max (ns)
t1	Address valid to BALE inactive	10	
t2	Command active to BALE inactive	25	
t3	Command active from AEN inactive	150	
t4	Command pulse width	65	
t5	Address hold from Command inactive	20	
t6	Data valid from Read active		22
t7	Data hold from Read inactive	0	
t8	Data valid from Write active		35
t9	Data hold from Write inactive	15	
t10	-MEMCS16 active from LA17-LA23 valid		43
t11	-0WS active to -MEMR/W active	12	

Memory Refresh



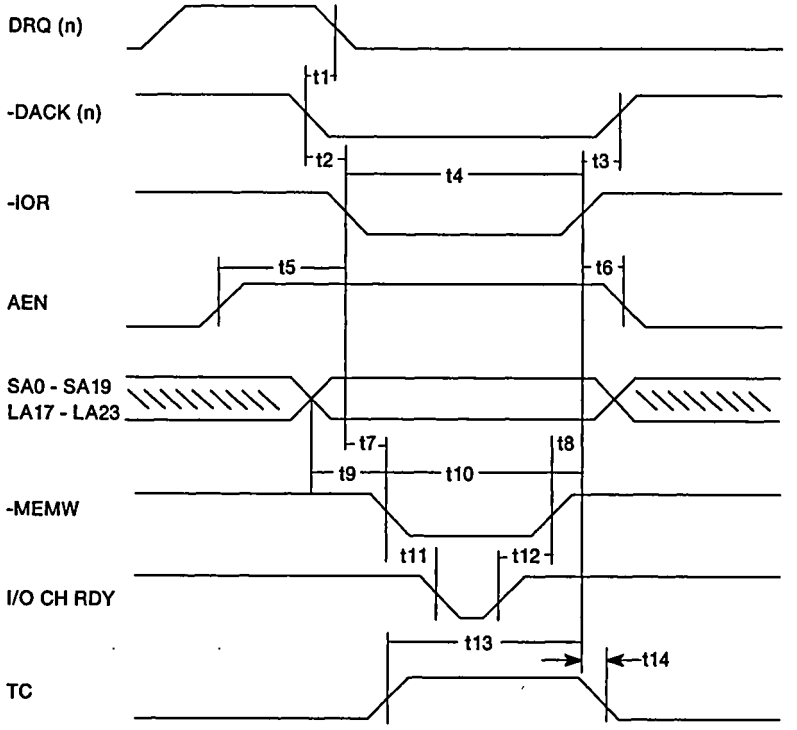
Symbol	Description	Min (ns)	Max (ns)
t1	-REFRESH active to -MEMR active	130	
t2	Address valid to -MEMR active	40	
t3	-MEMR pulse width	150	
t4	-MEMR inactive to -REFRESH inactive	10	
t5	-MEMR active to I/O CH RDY inactive		60
t6	I/O CH RDY pulse width		600
t7	-MEMR inactive from I/O CH RDY active	0	
t8	-MEMR inactive to next -MEMR active	200	
t9	Next address valid from -MEMR active	300	

DMA Read



Symbol	Description	Min (ns)	Max (ns)
t1	-DACK active to DRQ inactive	0	
t2	-DACK active to -IOW active	260	
t3	-IOW inactive to -DACK inactive	0	
t4	-IOW pulse width	360	
t5	AEN active to -IOW active	400	
t6	-IOW inactive to AEN inactive	0	
t7	-IOW active from -MEMR active	0	
t8	-IOW inactive to -MEMR inactive	0	
t9	Address valid to -MEMR active	130	
t10	-MEMR pulse width	360	
t11	-MEMR active to I/O CH RDY inactive		75
t12	-MEMR inactive from I/O CH RDY active	200	
t13	TC active setup to -IOW inactive	310	
t14	TC inactive from -IOW inactive	0	

DMA Write



Symbol	Description	Min (ns)	Max (ns)
t1	-DACK active to DRQ inactive	0	
t2	-DACK active to -IOR active	0	
t3	-IOR inactive to -DACK inactive	0	
t4	-IOR pulse width	560	
t5	AEN active to -IOR active	250	
t6	-IOR inactive to AEN inactive	0	
t7	-MEMW active from -IOR active	70	
t8	-MEMW inactive to -IOR inactive	0	
t9	Address valid to -MEMW active	130	
t10	-MEMW pulse width	330	
t11	-MEMW active to I/O CH RDY inactive		75
t12	-MEMW inactive from I/O CH RDY active	200	
t13	TC active setup to -IOR inactive	310	
t14	TC inactive from -IOR inactive	0	

DMA Controller

The direct memory access (DMA) controller allows I/O devices to transfer data directly to and from memory. This allows higher system microprocessor throughput by freeing the system microprocessor of I/O tasks.

The DMA controller is software programmable. The system microprocessor can address the DMA controller and read or modify the internal registers to define the DMA modes, transfer addresses, transfer counts, channel masks, and page registers.

The functions of the DMA controller can be grouped in two categories; program condition and DMA transfer. During program condition, the DMA registers can be programmed or read.

Program condition commences when the system microprocessor refers to the DMA controller within a specific address range. (See "DMA I/O Address Map" on page 2-23.) The DMA controller needs a minimum of 1.0 microseconds (one wait state) to complete any program command generated by the system microprocessor.

Single transfers require a minimum of 1.0 microseconds to transfer either a byte or a word.

The DMA controller supports:

- Register/Program compatibility with the IBM Personal Computer AT® DMA channels (8237 Compatible mode).
- 16MB (24-bit) address capability for memory.
- Seven independent DMA channels capable of transferring data between memory and I/O devices.
- Serial DMA operation with overlapped read and write cycles for each transfer operation.
- Each channel fixed to byte or word transfer.
- Sharing of the system bus interface and control logic.

Personal Computer AT is a registered trademark of International Business Machines Corporation.

Data Transfers Between Memory and I/O Devices

The DMA controller performs serial transfers with a minimum of five 200-nanosecond clock cycles for memory read to I/O, or I/O read to memory write operations. No burst mode or memory-to-memory transfers are supported.

Byte Pointer

A byte pointer allows 8-bit ports to access consecutive bytes of registers greater than 8 bits. These registers are the Memory Address registers (3 bytes), the Transfer Count registers (2 bytes), and the I/O Address registers (2 bytes).

DMA Channels

DMA channels 0 through 3 support 8-bit data transfers between 8-bit I/O adapters and 8-bit or 16-bit system memory. Each channel can transfer data throughout the 16MB system-address space in 64KB blocks. Logic is included on the system board containing the equivalent of two 8237 controller chips.

Figure 2-2 shows DMA channel assignments.

Figure 2-2. DMA Channel Assignments

Channel	Assignment
DRQ0	Unused
DRQ1	Unused
DRQ2	Diskette
DRQ3	Fixed Disk
DRQ4	Cascade
DRQ5	Unused
DRQ6	Unused
DRQ7	Unused

Address Generation for DMA Channels 0 through 3

Figure 2-3 shows address generation for DMA channels 0 through 3.

Figure 2-3. DMA Address Generation for Channels 0-3

Source	DMA Page Registers	Controller
Address	A23---------A16	A15---------A0

Note: The byte high enable (BHE) addressing signal is generated by inverting address line A0.

Address Generation for DMA Channels 5 through 7

DMA channel 4 is used to cascade channels 5 through 7 to the microprocessor. Channels 5, 6, and 7 support 16-bit data transfers between 16-bit I/O adapters and 16-bit system memory. These channels can transfer data throughout the 16MB system-address space in 128KB blocks. Channels 5, 6, and 7 cannot transfer data on odd-byte boundaries.

Figure 2-4 shows address generation for the DMA channels 5 through 7.

Figure 2-4. DMA Address Generation for Channels 5-7

Source	DMA Page Registers	Controller
Address	A23---------A17	A16---------A1

Note: The addressing signals, BHE and A0, are forced to a logical 0.

Page Register Addresses

Figure 2-5 shows the addresses for the page registers.

Figure 2-5. Page Register Addresses

Page Register	I/O Hex Address
DMA Channel 0	0087
DMA Channel 1	0083
DMA Channel 2	0081
DMA Channel 3	0082
DMA Channel 5	008B
DMA Channel 6	0089
DMA Channel 7	008A
Refresh	008F

Addresses for all DMA channels do not increase or decrease through page boundaries (64KB for channels 0 through 3, and 128KB for channels 5 through 7).

DMA channels 5 through 7 perform 16-bit data transfers. Access is only to 16-bit devices (I/O or memory) during the DMA cycles of channels 5 through 7. Access to the DMA controller, which controls these channels, is through I/O addresses hex 0C0 through 0DF.

All DMA memory transfers made with channels 5 through 7 must occur on even-byte boundaries. When the base address for these channels is programmed, the real address divided by 2 is the data written to the base address register. Also, when the base word count for channels 5 through 7 is programmed, the count is the number of 16-bit words to be transferred. Therefore, DMA channels 5 through 7 can transfer 65,536 words, or 128KB maximum, for any selected page of memory. These DMA channels divide the 16MB memory space into 128KB pages. When the DMA page registers for channels 5 through 7 are programmed, data bits D7 through D1 contain the most-significant 7 address bits (A23 through A17) of the desired memory space. Data bit D0 of the page registers for channels 5 through 7 is not used in the generation of the DMA memory address.

At power-on time, all internal locations, especially the mode registers, should be loaded with some valid value. This is done even if some channels are unused.

DMA I/O Address Map

Figure 2-6. DMA I/O Addresses for Memory Addresses, Word Counts, and Command/Status Registers

Address (hex)	Description	Bit Description	Byte Pointer
0000	Channel 0, Memory Address Register	00-15	Yes
0001	Channel 0, Transfer Count Register	00-15	Yes
0002	Channel 1, Memory Address Register	00-15	Yes
0003	Channel 1, Transfer Count Register	00-15	Yes
0004	Channel 2, Memory Address Register	00-15	Yes
0005	Channel 2, Transfer Count Register	00-15	Yes
0006	Channel 3, Memory Address Register	00-15	Yes
0007	Channel 3, Transfer Count Register	00-15	Yes
0008	Channel 0-3, Rd Status/Wrt Command Reg	00-07	
0009	Channel 0-3, Write Request Register	00-02	
000A	Channel 0-3, Write Single Mask Reg Bit	00-02	
000B	Channel 0-3, Mode Register (Write)	00-07	
000C	Channel 0-3, Clear Byte Pointer (Write)	N/A	
000D	Channel 0-3, Master Clear (W)/Temp (R)	00-07	
000E	Channel 0-3, Clear Mask Register (Write)	00-03	Yes
000F	Channel 0-3, Write All Mask Register Bits	00-03	Yes
0081	Channel 2, Page Table Address Register **	00-07	
0082	Channel 3, Page Table Address Register **	00-07	
0083	Channel 1, Page Table Address Register **	00-07	
0087	Channel 0, Page Table Address Register **	00-07	
0089	Channel 6, Page Table Address Register **	00-07	
008A	Channel 7, Page Table Address Register **	00-07	
008B	Channel 5, Page Table Address Register **	00-07	
008F	Channel 4, Pg Tbl Address/Refresh Register	00-07	
00C0	Channel 4, Memory Address Register	00-15	Yes
00C2	Channel 4, Transfer Count Register	00-15	Yes
00C4	Channel 5, Memory Address Register	00-15	Yes
00C6	Channel 5, Transfer Count Register	00-15	Yes
00C8	Channel 6, Memory Address Register	00-15	Yes
00CA	Channel 6, Transfer Count Register	00-15	Yes
00CC	Channel 7, Memory Address Register	00-15	Yes
00CE	Channel 7, Transfer Count Register	00-15	Yes
00D0	Channel 4-7, Read Status/Write Command Register	00-07	
00D2	Channel 4-7, Write Request Register	00-02	
00D4	Channel 4-7, Write Single Mask Register Bit	00-02	
00D6	Channel 4-7, Mode Register (Write)	00-07	
00D8	Channel 4-7, Clear Byte Pointer (Write)	N/A	
00DA	Channel 4-7, Master Clear (W)/Temp (R)	00-07	
00DC	Channel 4-7, Clear Mask Register (Write)	00-03	
00DE	Channel 4-7, Write All Mask Register Bits	00-03	

* Dependent upon the function used.

** Upper byte of Memory Address register.

DMA Registers

All system microprocessor access to the DMA must be 8-bit I/O instructions. Figure 2-7 lists the name and size of the DMA registers.

Figure 2-7. DMA Registers

Register	Size (bits)	Quantity of Registers	Allocation
Memory Address	16	8	1 per Channel
Transfer Count	16	8	1 per Channel
Page	8	8	1 per Channel
Mask	4	2	1 for Channels 7 - 4 1 for Channels 3 - 0
Mode	8	8	1 per Channel
Status	8	2	1 for Channel 7 - 4 1 for Channel 3 - 0

Memory Address Register

Each DMA channel has a 16-bit Memory Address register. This register holds the value of the address used during DMA transfers. The address is incremented or decremented after each transfer and the intermediate values of the address are stored in this register during the transfer. This register is written or read by the microprocessor in successive 8-bit bytes. Auto-initialization will restore this register to its original value.

The contents of the corresponding DMA page register is the most significant byte of the DMA memory address.

Transfer Count Register

Each DMA channel has a 16-bit Transfer Count register that is loaded by the system microprocessor. The transfer count determines the number of transfers to be executed by the DMA channel before reaching terminal count. The number of transfers is always one more than the count specifies. For example, if the count is 0, the DMA does one transfer. When the value in the register goes from hex 0000 to FFFF, a terminal count pulse is generated by the DMA channel. The Transfer Count register can be read by the system microprocessor in successive I/O bytes when the DMA controller is in the program condition. Auto-initialization will restore the register to its original value.

Mask Register

Each DMA channel has a corresponding mask bit that, when set, disables the DMA from servicing the requesting device. Each mask bit can be set or cleared by the system microprocessor. A system reset or DMA master clear sets all mask bits to 1. A Clear Mask Register command sets all mask bits to 0. This register can be programmed using the 8237 Compatible mode commands.

Figure 2-8. Set/Clear Single Mask Bit Using 8237 Compatible Mode

Bit	Function
7 - 3	Reserved (Must be set to 0)
2	0 = Clear Mask Bit 1 = Set Mask Bit
1, 0	00 = Select Channel 0 or 4 01 = Select Channel 1 or 5 10 = Select Channel 2 or 6 11 = Select Channel 3 or 7

Figure 2-9. DMA Mask Register Write Using 8237 Compatible Mode

Bit	Function
7 - 4	Reserved (Must be set to 0)
3	0 = Clear Channel 3 or 7 Mask Bit 1 = Set Channel 3 or 7 Mask Bit
2	0 = Clear Channel 2 or 6 Mask Bit 1 = Set Channel 2 or 6 Mask Bit
	0 = Clear Channel 1 or 5 Mask Bit 1 = Set Channel 1 or 5 Mask Bit
0	0 = Clear Channel 0 or 4 Mask Bit 1 = Set Channel 0 or 4 Mask Bit

Mode Register

Each DMA channel has a Mode register that identifies the type of operation that takes place when that channel is activated. The Mode register is programmed by the system microprocessor and the contents are reformatted and stored internally in the DMA controller. This register can only be written.

Figure 2-10. 8237 Compatible Mode Register

Bit	Function
7-6	Reserved (Must be set to 0)
5	00 = Address Increment 01 = Address Decrement
4	00 = Auto-initialize Disable 01 = Auto-initialize Enable
3, 2	00 = Verify Operation 01 = Write Operation 10 = Read Operation 11 = Reserved
1, 0	00 = Select Channel 0 or 4 01 = Select Channel 1 or 5 10 = Select Channel 2 or 6 11 = Select Channel 3 or 7

Status Register

There are two Status registers containing information about the status of the devices. This information tells which channels have reached terminal count. Bits 3 through 0 are set every time a terminal count is reached by a corresponding channel. All bits are cleared by reset or following a system microprocessor Status Read command. When these registers are written, they are Command registers as shown in Figure 2-12 on page 2-27.

Figure 2-11. Status Register (Read)

Bit	Function
7	Channel 3 or 7 Request
6	Channel 2 or 6 Request
5	Channel 1 or 5 Request
4	Channel 0 Request
3	Channel 3 or 7 Terminal Count
2	Channel 2 or 6 Terminal Count
1	Channel 1 or 5 Terminal Count
0	Channel 0 Terminal Count

Figure 2-12. Command Register (Write)

Bit	Function
7	0 = DACK Active Low 1 = DACK Active High
6	0 = DRQ Active High 1 = DRQ Active Low
5	0 = Late Write 1 = Extended Write
4	0 = Fixed Priority 1 = Rotating Priority
3	0 = Normal Timing 1 = Compressed Timing
2	0 = Controller Enable 1 = Controller Disable
1	Reserved
0	Reserved (Must be 0)

Figure 2-13 shows an example of programming DMA channel 2.

Figure 2-13. DMA Channel 2 Programming Example

Program Step	OUT to ADRS Data
Set Channel Mask Bit	(000AH) x6H
Clear Byte Pointer	(000CH) xxH
Write Memory Address	(0004H) xxH
Write Memory Address	(0004H) xxH
Write Page Table Address	(0081H) xxH
Clear Byte Pointer	(000CH) xxH
Write Register Count	(0005H) xxH
Write Register Count	(0005H) xxH
Write Mode Register	(000BH) xxH
Clear Channel 2 Mask Bit	(000AH) x2H

Values inside ( ) are addresses, x represents data.

Interrupts

The system provides 15 levels of system interrupts. Any or all of the interrupts can be masked, including the non-maskable interrupt. The system board uses the logic equivalent of two Intel 8259A interrupt controllers.

Non-Maskable Interrupt

The non-maskable interrupt (NMI) signals the system microprocessor that a channel check has occurred. The NMI masks all other interrupts and the IRET instruction restores the interrupt flag to the state it was in prior to the interrupt. A system reset resets the NMI.

A NMI request from channel check is subject to mask control from the NMI mask bit at I/O address hex 0070. This address is shared with the address for the RT/CMOS RAM. (See "RTC/CMOS RAM I/O Operations" on page 3-159.) The power-on default of the NMI mask is 1 (NMI disabled). Prior to enabling the NMI after a power-on reset (write to address hex 0070 with bit 7 equal to 0), the channel check state is initialized by the POST.

Warning: When writing to address hex 0070 to enable or disable an NMI, a read to address hex 0071 must be accessed immediately. Failure to do this can cause unreliable operation of the Real-Time Clock/CMOS RAM.

Interrupt Assignments

Figure 2-14 shows the interrupt levels and their functions. The interrupt levels are listed in the order of priority. The highest priority is the NMI and the lowest is IRQ7.

Figure 2-14. Interrupt Level Assignments by Priority

Level		Function
NMI		Channel Check
IRQ0		Timer
IRQ1		Keyboard
IRQ2		Cascade Interrupt Control to IRQ8 - IRQ15
	IRQ8	Real-Time Clock
	IRQ9	Redirect Cascade
	IRQ10	Reserved
	IRQ11	Reserved
	IRQ12	Mouse
	IRQ13	Reserved
	IRQ14	Fixed Disk
	IRQ15	Reserved
IRQ3		Serial Alternate
IRQ4		Serial Primary
IRQ5		Reserved
IRQ6		Diskette
IRQ7		Parallel Port/Audio Card

IRQ 8 through 15 are cascaded through IRQ 2.

Hardware interrupt IRQ9 is defined as the replacement interrupt level for the cascade level IRQ2. Program interrupt sharing should be implemented on IRQ2, interrupt hex 0A. The following processing occurs to maintain compatibility with the IRQ2 used by IBM Personal Computer products:

1. A device drives the interrupt request active on IRQ2 of the channel.
2. This interrupt request is mapped in hardware to IRQ9 input on the second interrupt controller.

3. When the interrupt occurs, the system microprocessor passes control to the IRQ9 (interrupt hex 71) interrupt handler.
4. This interrupt handler performs an end of interrupt (EOI) to the second interrupt controller and passes control to IRQ2 (interrupt hex 0A) interrupt handler.
5. This IRQ2 interrupt handler causes the device to reset the interrupt request prior to performing an EOI to the master interrupt controller that finishes servicing the IRQ2 request.

Note: Prior to programming the interrupt controllers, interrupts should be disabled by executing a CLI instruction. This includes the Mask register, end of interrupts, initialization control words, and operational control words.

System Timers

The system has three programmable timers: channel 0, channel 1, and channel 2. Channels 0, 1, and 2 are similar to channels 0, 1, and 2 of the IBM Personal Computer, IBM Personal Computer XT™, and the IBM Personal Computer AT.

Figure 2-15 is a block diagram of the timers.

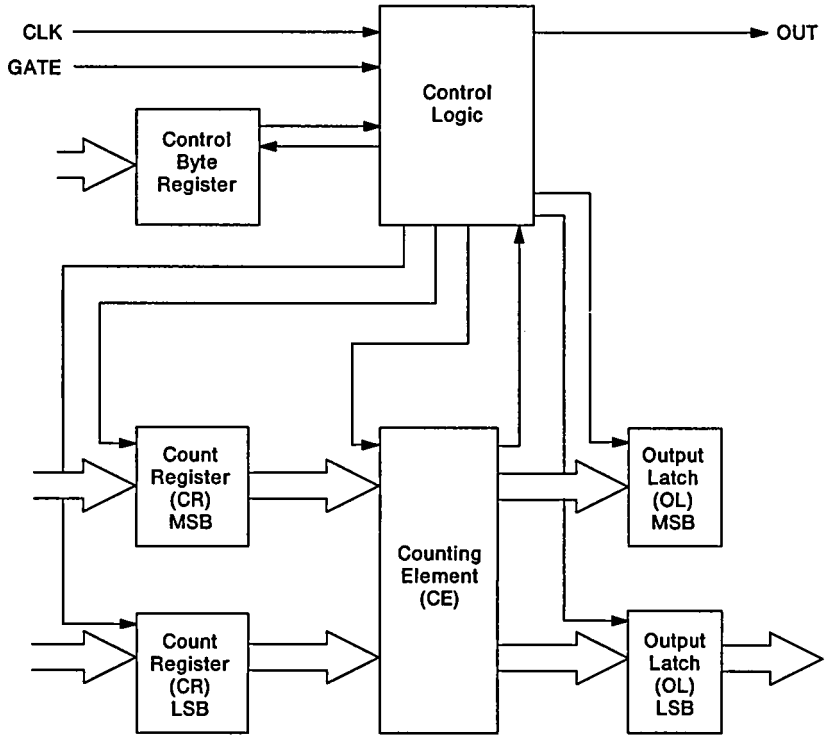


Figure 2-15. System Timer Block Diagram

Personal Computer XT is a trademark of International Business Machines Corporation.

Channel 0 - System Timer

- GATE 0 is always enabled.
- CLK IN 0 is driven by 1.190 MHz.
- CLK OUT 0 drives the interrupt controller chip, IRQ 0.

Channel 1 - Refresh Request Generator

- GATE 1 is always enabled.
- CLK IN 1 is driven by 1.190 MHz.
- CLK OUT 1 is request-refresh cycle.

  Note: Channel 1 is programmed as a rate generator to produce a 15-microsecond signal.

Channel 2 - Tone Generation for Speaker

- GATE 2 is controlled by bit 0 (PPI) of port hex 0061.
- CLK IN 2 is driven by 1.190 MHz.
- CLK OUT 2 is connected to the input port hex 0061, bit 5. CLK OUT 2 is also logically ANDed with port hex 0061, bit 1 to produce the speaker data enable signal.

Timers 0, 1, and 2

Each timer is independent. Timers 0, 1, and 2 are 16-bit down counters that can be preset. They can count in binary or binary-coded decimal (BCD).

Programming the System Timers

The system treats the programmable interval timer as an arrangement of four external I/O ports. Three ports are treated as count registers and one as control register for mode programming. Timers are programmed by writing a control word and then an initial count. All control words are written into the control word registers, which are located at I/O address hex 0043 for timers 0, 1, and 2. Initial counts are written into the count registers, not the control word registers. The format of the initial count is determined by the control word used.

The count is written to the count register. It is then transferred to the counting element, according to the mode definition. When the count is read, the data is presented at the output latch.

Counter Write Operations

The control word must be written before the initial count is written.

The count must follow the count format specified in the control word.

A new initial count can be written to the counters at any time without affecting the programmed mode of the counter. Counting is affected as described in the mode definitions. The new count must follow the programmed count format.

Counter Read Operations

The counters can be read using the Counter Latch command. (See "Counter Latch Command" on page 2-35.)

If the counter is programmed for 2-byte counts, 2 bytes must be read. The 2 bytes need not be read consecutively; read, write, or programming operations of other counters can be inserted between them.

Note: If the counters are programmed to read or write 2-byte counts, the program must not transfer control between writing the first and second byte to another routine that also reads or writes into the same counter. This will cause an incorrect count.

Registers

Figure 2-16. System Timer/Counter Registers

I/O Address (hex)	Register
0040	Read/Write Timer/Counter 0
0041	Read/Write Timer/Counter 1
0042	Read/Write Timer/Counter 2
0043	Write Control Byte for Counters 0, 1, or 2

Channel 0 (Hex 0040) and Channel 2 (Hex 0042) Count Registers

The control byte is written to address hex 0043, indicating the format of the count (least-significant byte only, most-significant byte only, or least-significant byte followed by most-significant byte). This procedure must be done before writing the count to I/O address hex 0040 for channel 0 or hex 0042 for channel 2.

Control Byte Register - Channel 0, 1, or 2 (Hex 0043)

This is a write-only register. Figure 2-17 through Figure 2-20 describe the format for the control byte (I/O address hex 0043) for counters 0, 1, and 2.

Figure 2-17. SC - Select Counter, I/O Address Hex 0043

Bit 7 SC1	Bit 6 SC0	
0	0	Select Counter 0
0	1	Select Counter 1
1	0	Select Counter 2
1	1	Reserved

Figure 2-18. RW - Read/Write Counter, I/O Address Hex 0043

Bit 5 RW1	Bit 4 RW0	
0	0	Counter Latch Command
0	1	Read/Write Counter Bits 0 - 7 only
1	0	Read/Write Counter Bits 8 - 15 only
1	1	Read/Write Counter Bits 0 - 7 first, then Bits 8 - 15

Figure 2-19. M - Counter Mode, I/O Address Hex 0043

Bit 3 M2	Bit 2 M1	Bit 1 M0	
0	0	0	Mode 0
0	0	1	Mode 1
X	1	0	Mode 2
X	1	1	Mode 3
1	0	0	Mode 4
1	0	1	Mode 5

Don't care bits (X) should be set to 0.

Figure 2-20. Binary Coded Decimal (BCD)

Bit 0 BCD	
0	Binary Counter 16 Bits
1	Binary Coded Decimal Counter (4 Decades)

Counter Latch Command

The Counter Latch command is written to the control byte register. The SC0 and SC1 bits select the counter, and bits 5 and 4 distinguish this command from a control byte. Figure 2-21 shows the format of the Counter Latch command.

Figure 2-21. Counter Latch Command

Bit	Function
7	SC1 - Specifies the counter to be latched
6	SC0 - Specifies the counter to be latched
5, 4	0 - Specifies the Counter Latch command
3 - 0	0 - Reserved

The count is latched into the selected counter output latch at the time the Counter Latch command is received. This count is held in the latch until it is read by the system microprocessor (or until the counter is reprogrammed). After the count is read by the system microprocessor, it is automatically unlatched and the output latch returns to following the counting element. Counter Latch commands do not affect the programmed mode of the counter in any way. All subsequent latch commands to a given counter, issued before the count is read, are ignored. A read cycle to the counter latch returns the value latched by the first Counter Latch command.

System Timer Modes

The following definitions are used when describing the timer modes.

CLK pulse The rising edge, then the falling edge of the CLK input to a counter.

Trigger The rising edge of the GATE input to a counter.

Counter load The transfer of a count from the counter register to the counting element.

Mode 0 - Interrupt on Terminal Count

Event counting can be done using mode 0. Counting is enabled when GATE is set to 1, and disabled when GATE is set to 0. If GATE is 1 when the control byte and initial count are written to the counter, the sequence for mode 0 is:

1. The control byte is written to the counter, and OUT goes low.
2. The initial count is written.
3. Initial count is loaded on the next CLK pulse. The count is not decremented by this CLK pulse.

   The count is decremented until the counter reaches 0. For an initial count of *n*, the counter reaches 0 after $n+1$ CLK pulses.
4. OUT goes high.

OUT remains high until a new count or new mode 0 control byte is written into the counter.

If GATE is equal to 0 when an initial count is written to the counter, it is loaded on the next CLK pulse even though counting is not enabled. After GATE enables counting, OUT goes high *n* CLK pulses later.

If a new count is written to a counter while counting, it is loaded on the next CLK pulse, and counting continues from the new count. If a 2-byte count is written to the counter, the following occurs:

1. The first byte written to the counter disables the counting. OUT goes low immediately and there is no delay for the CLK pulse.
2. When the second byte is written to the counter, the new count is loaded on the next CLK pulse. OUT goes high when the counter reaches 0.

Mode 1 - Hardware Retriggerable One-Shot

The counter is armed by writing the control byte and initial count to the counter. When a trigger occurs the counter is loaded. The sequence for mode 1 is:

1. OUT is high.
2. The control byte and initial count are written to the counter.
3. On the CLK pulse following a trigger, OUT goes low and begins the one-shot pulse.
4. The count is decremented until the counter reaches 0.
5. OUT goes high.

OUT remains high until the CLK pulse after the next trigger.

For an initial count of *n*, a one-shot pulse is *n* CLK pulses long. The one-shot pulse repeats the same count of *n* for the next triggers. OUT remains low *n* CLK pulses following any trigger. GATE does not affect OUT. The current one-shot pulse is not affected by a new count written to the counter unless the counter is retriggered. If the counter is retriggered, the new count is loaded and the one-shot pulse continues.

Note: Mode 1 is only valid on counter 2.

Mode 2 - Rate Generator

This mode causes the counter to perform a divide-by-*n* function. Counting is enabled when GATE is set to 1, and disabled when GATE is set to 0. The sequence for mode 2 is:

1. OUT is high.
2. The control byte and initial count are written to the counter.
3. Initial count is loaded on the next CLK pulse.
4. The count is decremented until the counter reaches 1.
5. OUT goes low for one CLK pulse.
6. OUT goes high.
7. The counter reloads the initial count.
8. The process is repeated.

If GATE goes low during the OUT pulse, OUT goes high. On the next CLK pulse, a trigger reloads the counter with the initial count. OUT

goes low *n* CLK pulses after the trigger. This allows the GATE input to be used to synchronize the counter.

OUT goes low *n* CLK pulses after writing the initial count. This allows software synchronization of the counter.

The current counting sequence is not affected by a new count being written to the counter. If the counter receives a trigger after a new count is written and before the end of the current count, the new count is loaded on the next CLK pulse and counting continues from the new count. If the trigger is not received by the counter, the new count is loaded following the current counting cycle.

Mode 3 - Square Wave

Mode 3 is similar to mode 2 except for the duty cycle of OUT. Counting is enabled when GATE is set to 1, and disabled when GATE is set to 0. An initial count of *n* results in OUT being a square wave. The period of the square wave is *n* CLK pulses. If OUT is low and GATE goes low, OUT goes high. On the next CLK pulse, a trigger reloads the counter with the initial count.

After writing a control byte and initial count, the counter is loaded on the next CLK pulse. This allows software synchronization of the counter.

The current counting sequence is not affected by a new count being written to the counter. If the counter receives a trigger after a new count is written, and before the end of the current count half-cycle of the square wave, the new count is loaded on the next CLK pulse, and counting continues from the new count. If the trigger is not received by the counter, the new count is loaded following the current half-cycle.

The implementation of mode 3 differs, depending on whether the count written is an odd or even number. If the count is even, the sequence for mode 3 is:

1. OUT is high.
2. The initial count is loaded on the first CLK pulse.
3. The count is decremented by 2 on succeeding CLK pulses.
4. The count is decremented until the counter reaches 0.
5. OUT changes state.
6. The counter is reloaded with the initial count.

7. The process repeats indefinitely.

If the count is odd, the sequence for mode 3 is:

1. OUT is high.
2. The initial count minus 1 is loaded on the first CLK pulse.
3. The count is decremented by 2 on succeeding CLK pulses.
4. The count is decremented until the counter reaches 0.
5. One CLK pulse after the count reaches 0, and OUT goes low.
6. The counter is reloaded with the initial count minus 1.
7. The count is decremented by 2 on succeeding CLK pulses.
8. The count is decremented until the counter reaches 0.
9. OUT goes high.
10. The process repeats indefinitely.

Mode 3, using an odd count, causes OUT to go high for a count of $(n+1)/2$ and low for a count of $(n-1)/2$.

Mode 3 can operate such that OUT is initially set low, when the control byte is written. For this condition, the sequence for mode 3 is:

1. OUT is low.
2. The count decrements to half of the initial count.
3. OUT goes high.
4. The count is decremented until the counter reaches 0.
5. OUT goes low.
6. The process repeats indefinitely.

This process results in a square wave with a period of *n* CLK pulses.

Note: If it is required that OUT be high after the control byte is written, the control byte must be written twice. This applies only to mode 3.

Mode 4 - Software Retriggerable Strobe

Counting is enabled when GATE is 1, and disabled when GATE is 0. Counting is triggered when an initial count is written. The sequence for mode 4 is:

1. OUT is high.
2. The control byte and initial count are written to the counter.
3. The initial count is loaded on the next CLK pulse. The count is not decremented by this clock pulse.
4. The count is decremented until the counter reaches 0. For an initial count of *n*, the counter reaches 0 after $n+1$ CLK pulses.
5. OUT goes low for one CLK pulse.
6. OUT goes high.

GATE should not go low one half CLK pulse before or after OUT goes low. If this occurs, OUT remains low until GATE transitions high.

If a new count is written to a counter while counting, it is loaded on the next CLK pulse. Counting then continues from the new count. If a 2-byte count is written, the following occurs:

1. The first byte written to the counter does not affect counting.
2. When the second byte is written to the counter, the new count is loaded on the next CLK pulse.

The mode 4 sequence can be retriggered by software. The period from when the new count of *n* is written to when OUT strobes low is $(n+1)$ pulses.

Mode 5 - Hardware Retriggerable Strobe

Counting is triggered by the rising edge of GATE. The sequence for mode 5 is:

1. OUT is high.
2. The control byte and initial count are written to the counter. Counting is triggered by the rising edge of GATE.
3. The counter is loaded on the next CLK pulse after the trigger. The count is not decremented by this CLK pulse.
4. The count is decremented until the counter reaches 0.
5. OUT goes low for one CLK pulse. This occurs $(n+1)$ CLK pulses after the trigger.

6. OUT goes high.

The counting sequence can be retriggered. OUT strobes low $(n+1)$ pulses after the trigger. GATE does not affect OUT.

The current counting sequence is not affected by a new count being written to the counter. If the counter receives a trigger after a new count is written and before the end of the current count, the new count is loaded on the next CLK pulse and counting continues from the new count.

Note: Mode 5 is valid only on counter 2.

Operations Common to All Modes

Control bytes written to a counter cause the control logic to reset. OUT goes to a known state. This does not take a CLK pulse.

The falling edge of the CLK pulse is when new counts are loaded and counters are decremented.

Counters do not stop when they reach 0. In modes 0, 1, 4, and 5, the counter wraps around to the highest count, and continues counting. Modes 2 and 3 are periodic; the counter reloads itself with the initial count and continues from there.

The GATE is sampled on the rising edge of the CLK pulse.

Figure 2-22 shows the minimum and maximum initial counts for the counters.

Figure 2-22. Minimum and Maximum Initial Counts, Counters 0 and 2

Mode	Min Count	Max Count
0	1	$0 = 2^{16}$ (Binary Counting) or 10^4 (BCD Counting)
1	1	$0 = 2^{16}$ (Binary Counting) or 10^4 (BCD Counting)
2	2	$0 = 2^{16}$ (Binary Counting) or 10^4 (BCD Counting)
3	2	$0 = 2^{16}$ (Binary Counting) or 10^4 (BCD Counting)
4	1	$0 = 2^{16}$ (Binary Counting) or 10^4 (BCD Counting)
5	1	$0 = 2^{16}$ (Binary Counting) or 10^4 (BCD Counting)

Speaker

The system board generates a signal to an amplifier driving the display speaker. The audio signal is carried through the power/audio cable connector from the system to the display.

The audio signal on the system board combines audio from three sources:

- System timer
- 2400 bps modem card
- Audio card.

The system timer circuit on the system board allows the speaker to be driven three different ways:

- A direct program control register bit can be toggled to generate a pulse train.
- The clock input to the timer can be modulated with a program-controlled I/O port bit.
- The output from channel 2 of the timer can be programmed to generate a waveform to the speaker.

Figure 2-23. Speaker Tone Generation

Channel 2	Tone
Gate 2	Controlled by I/O Port Bit 1
Clock In 2	1.190 MHz OSC
Clock Out 2	Used to drive speaker

All three methods can be performed simultaneously. (See "System Timers" on page 2-31.)

Section 3. System Board I/O Controllers

Keyboard/Mouse Controller 3-3
Keyboard Password Security 3-3
Keyboard Controller Command and Status Bytes 3-5
Keyboard/Mouse Programming Considerations 3-9
Mouse/System Timings 3-10
Signals 3-12
Connector 3-13
Video Subsystem 3-14
VGA Components 3-17
BIOS ROM 3-18
Support Logic 3-18
Video Graphics Array Major Components 3-18
Modes of Operation 3-22
Display Support 3-23
Video Subsystem Programmable Option Select 3-24
Alphanumeric Modes 3-25
Graphics Modes 3-28
Video Memory Organization 3-35
Video Memory Read/Write Operations 3-50
Registers 3-53
General Registers 3-54
Sequencer Registers 3-58
CRT Controller Registers 3-64
Graphics Controller Registers 3-79
Attribute Controller Registers 3-87
VGA Programming Considerations 3-93
Video Digital-to-Analog Converter (Video DAC) 3-101
Diskette Drive Controller 3-111
Registers 3-112
Diskette Drive Controller Programming Considerations . . . 3-115
Command Format 3-118
Command Status Registers 3-128
Signal Descriptions 3-131
Connector 3-134
Serial Port Controller 3-135
Communications Application 3-136
Programmable Baud-Rate Generator 3-137
Registers 3-137
Serial Port Controller Programming Considerations 3-147
Signal Descriptions 3-147
System Board Voltage Interchange 3-148
Parallel Port Controller 3-149

Parallel Port Programmable Option Select 3-150
Parallel Port Controller Programming Considerations 3-151
Parallel Port Timing 3-154
Signal Descriptions 3-154
Connector 3-155
Memory 3-156
Read-Only Memory (ROM) 3-156
Random Access Memory (RAM) 3-156
Complementary Metal Oxide Semiconductor RAM 3-158
CMOS RAM Configuration 3-162
Miscellaneous System Ports 3-165
System Control Port B (Hex 0061) 3-165
RTC/CMOS and NMI Mask (Hex 0070) 3-166
System Control Port A (Hex 0092) 3-167

Keyboard/Mouse Controller

The keyboard/mouse controller uses an Intel 8042 microprocessor. The keyboard is connected to the left-most of the two controller connectors in the rear of the system unit. This connector is dedicated to the keyboard.

The keyboard controller receives the serial data, checks the parity, translates keyboard scan codes (see "Keyboard Controller Command and Status Bytes" on page 3-5), and presents the data to the system as a byte of data at I/O address hex 0060. The interface can interrupt the system when data is available or can wait for polling from the microprocessor.

I/O address hex 0064 is the command/status port. When the system reads I/O address hex 0064, it receives status information from the keyboard controller. When the system writes to the port, the keyboard controller interprets the byte as a command.

Keyboard Password Security

The keyboard controller provides for a password security mechanism. Three commands are available regarding password operation:

A4	**Test Password Installed**
A5	**Load Security**
A6	**Enable Security.**

The system microprocessor can issue a Test Password Installed command to determine if a password is currently installed. This feature allows the controlling program to decide whether or not to write over the existing password.

The system microprocessor can issue a Load Security command and set a password in the keyboard controller at any time. Any existing password is lost, and the new password becomes the active password. The password must be installed in scan code format.

The system microprocessor must issue the Enable Security command to set the keyboard controller into Secure mode. At this point the keyboard controller does not pass any information along to the system microprocessor. The keyboard controller intercepts the keyboard data stream, continuously comparing it to the installed password pattern. Until a match is encountered, no keyboard or

mouse data is passed to the system microprocessor. When a match occurs, the state of the keyboard controller is restored and data is allowed to pass to the system microprocessor.

The password can be changed as often as the user chooses. No command to verify the installed password is provided. No commands are accepted by the keyboard controller when keyboard security is active.

Keyboard Controller Command and Status Bytes

Keyboard Controller Command Byte (Hex 0064, Write)

Figure 3-1 and Figure 3-2 on page 3-6 show the keyboard controller command and status bytes:

Figure 3-1. Keyboard Controller Command Byte, Port Hex 0064 Write

Bit	Function		
7	Reserved = 0		
6	IBM Keyboard Translate Mode	1=	Translate incoming scan codes to Scan Code Set 1*
		0=	No translation
5	Disable Mouse	1=	Drive clock low, disabling mouse interface
		0=	Drive clock high, enabling mouse interface
4	Disable Keyboard	1=	Drive clock low, disabling keyboard interface
		0=	Drive clock high, enabling keyboard interface
3	Reserved = 0		
2	System Flag		Placed in the system flag bit of the keyboard controller status register
1	Enable Mouse Interrupt	1=	Keyboard controller generates an interrupt when mouse data is written to output buffer
0	Enable Keyboard Interrupt	1=	Keyboard controller generates an interrupt when keyboard data is written to output buffer

* Used on the IBM Personal Computer and the IBM Personal Computer XT.

Keyboard Controller Status Byte (Hex 0064, Read)

Figure 3-2 shows the keyboard controller status byte:

Figure 3-2. Keyboard Controller Status Byte, Port Hex 0064 Read

Bit	Function
7	Parity Error
6	General Time Out
5	Mouse Output Buffer Full
4	Inhibit Switch
3	Command/Data
2	System Flag
1	Input Buffer Full
0	Output Buffer Full

Input Buffer

The input buffer (hex 0060) is an 8-bit write-only register. When the input buffer is written, a flag is set that indicates a data write. Data written to the input buffer is sent to the keyboard unless the keyboard controller is expecting a data byte following a keyboard controller command. Data should be written to the keyboard controller input buffer only if Input Buffer Full (bit 1) in the status byte (hex 0064) is equal to 0.

Output Buffers

The output buffer (hex 0060) is an 8-bit read-only register. When the output buffer is read, the keyboard controller uses it to send information to the system microprocessor. The information can be scan codes from the keyboard, data from a mouse, or data bytes that result from a command from the system microprocessor.

Keyboard Controller Commands

A command is a byte written to the keyboard controller through I/O address hex 0064. The following are the recognized keyboard controller commands (hex values).

20-3F **Read Keyboard Controller RAM** — Bits D5-D0 specify the address.

20 **Read Keyboard Controller Command Byte** — The keyboard controller puts the command byte in its output buffer (hex 0060).

60-7F **Write Keyboard Controller RAM** – Bits D5-D0 specify the address.

60 **Write Keyboard Controller Command Byte** – The keyboard controller puts the next byte of data written to I/O address hex 0060 in its command byte.

A4 **Test Password Installed** – This command checks for a password installed in the keyboard controller. The result, hex FA (password installed) or hex F1 (password not installed), is placed in the output buffer (I/O address hex 0060 and IRQ01).

A5 **Load Security** – This command initiates the password load procedure. The keyboard controller inputs from the data port until it detects a 0, which terminates password entry.

A6 **Enable Security** – This command enables the keyboard controller security feature. This command is valid only when a password pattern is currently loaded into the keyboard controller.

A7 **Disable Mouse Interface** – This command sets bit 5 of the keyboard controller command byte to 1, driving the clock line low which disables the mouse interface. Data is not sent or received.

A8 **Enable Mouse Interface** – This command clears bit 5 of the keyboard controller command byte to 0, which enables the mouse interface.

A9 **Interface Test** – This command causes the keyboard controller to test the mouse clock and data lines. The result is placed in the output buffer (I/O address hex 0060 and IRQ01) as shown in Figure 3-3.

Figure 3-3. Command A9 Test Results

Test Result (hex)	Meaning
00	No error was detected.
01	Mouse clock line is stuck low.
02	Mouse clock line is stuck high.
03	Mouse data line is stuck low.
04	Mouse data line is stuck high.

AA **Self Test** – This command causes the keyboard controller to perform internal diagnostic tests. Hex 55 is placed in the output buffer (I/O address hex 0060) if no errors are detected.

AB **Interface Test** – This command causes the keyboard controller to test the keyboard clock and data lines. The result is placed in the output buffer (I/O address hex 0060 and IRQ01) as shown in Figure 3-4 on page 3-8.

Figure 3-4. Command AB Test Results

Test Result (hex)	Meaning
00	No error was detected.
01	Keyboard clock line is stuck low.
02	Keyboard clock line is stuck high.
03	Keyboard data line is stuck low.
04	Keyboard data line is stuck high.

AC **Reserved**

AD **Disable Keyboard Interface** – This command sets bit 4 of the keyboard controller command byte to 1, driving the clock line low which disables the keyboard interface. Data will not be sent or received.

AE **Enable Keyboard Interface** – This command clears bit 4 of the keyboard controller command byte to 0, which enables the keyboard interface.

C0 **Read Input Port** – This command causes the keyboard controller to read its input port and place the data in the output buffer (hex 0060). This command should only be used if the output buffer is empty.

C1 **Poll Input Port Low** – Port 1 bits 0-3, in status bits 4-7.

C2 **Poll Input Port High** – Port 1 bits 4-7, in status bits 4-7.

D0 **Read Output Port** – This command causes the keyboard controller to read its output port and place the data in the output buffer (hex 0060). This command should only be used if the output buffer is empty.

D1 **Write Output Port** – The next byte of data written to I/O address hex 0060 is placed in the keyboard controller output port.

Note: Bit 0 of the keyboard controller output port is connected to System Reset. This bit should not be written low.

Note: System Control Port A should be used to set Alternate Gate A20 instead of writing to the output port. See page 3-167.

D2 **Write Keyboard Output Buffer** — The next byte written to the input buffer (hex 0060), is written to the output buffer (hex 0060) as if initiated by a device. An interrupt occurs if the interrupt is enabled in the command byte.

D3 **Write Mouse Output Buffer** — The next byte written to the input buffer (hex 0060) is written to the output buffer (hex 0060) as if initiated by a device. An interrupt occurs if the interrupt is enabled in the command byte.

D4 **Write to Mouse** — The next byte written to the input buffer (hex 0060) is transmitted to the mouse.

E0 **Read Test Inputs** — This command causes the keyboard controller to read its T0 and T1 inputs. This data is placed in the output buffer. Data bit 0 represents T0 and data bit 1 represents T1.

F0-FF **Pulse Output Port** — Bits 0 through 3 of the keyboard controller output port can be pulsed low for approximately 6 microseconds. Bits 0 through 3 of this command indicate which bits are to be pulsed. A 0 indicates that the bit should be pulsed. A 1 indicates the bit should not be modified.

Note: Bit 0 of the keyboard controller output port is connected to System Reset. Pulsing of this bit resets the system microprocessor.

Keyboard/Mouse Programming Considerations

The following are some programming considerations for the keyboard/mouse controller.

- Status register (I/O address hex 0064) can be read at any time.
- Output buffer (I/O address hex 0060) should be read only when the Output Buffer Full (bit 0) in the status register is 1.
- The Mouse Output Buffer Full (bit 5) in the status register indicates that the data in the output buffer (hex 0060) came from the mouse. This bit is valid only when the Output Buffer Full (bit 0) is 1.
- The output buffer (hex 0060) and status register (hex 0064) should be written only when the Input Buffer Full (bit 1) and the Output Buffer Full (bit 0) are 0s.
- The devices connected to the keyboard controller should be disabled before initiating a command that generates output. If output is generated, any value in the output buffer is overwritten.

Mouse/System Timings

Data transmissions to and from the mouse connector consist of an 11-bit data stream sent serially over the data line. Figure 3-5 on page 3-10 shows the function of each bit.

Figure 3-5. Mouse Data Stream Bit Definitions

Bit	Function
11	Stop Bit (always 1)
10	Parity Bit (odd parity)
9	Data Bit 7 (most-significant)
8	Data Bit 6
7	Data Bit 5
6	Data Bit 4
5	Data Bit 3
4	Data Bit 2
3	Data Bit 1
2	Data Bit 0 (least-significant)
1	Start Bit (always 0)

System Receiving Data

The following describes the typical sequence of events that takes place when the system is receiving data from the mouse. Figure 3-6 on page 3-11 shows a graphic representation of the timing relationships.

1. The mouse checks the clock line. If the line is inactive, output from the mouse is not allowed.
2. The mouse checks the data line. If the line is inactive, the mouse receives data from the system.
3. The mouse checks the clock line at intervals not exceeding 100 microseconds during transmission. If the system is holding the clock line inactive, the transmission is terminated. The system can terminate transmission anytime during the first 10 clock cycles.
4. A final check for terminated transmission is performed at least 5 microseconds after the 10th clock.
5. The system can hold the clock line inactive to inhibit the next transmission.
6. The system can set the data line inactive when it has data to transmit to the mouse. The start bit (always 0) sets the data line inactive.
7. The system raises the clock line to allow the next transmission.

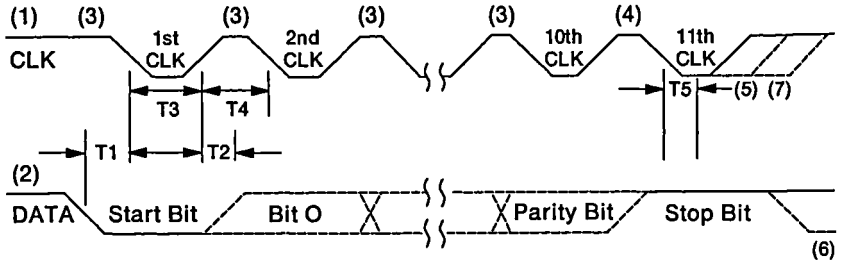


Figure 3-6. Receiving Data Timings

	Timing Parameter	Min/Max (microseconds)
T1	Time from DATA transition to falling edge of CLK	5 / 25
T2	Time from rising edge of CLK to DATA transition	5 / T4 - 5
T3	Duration of CLK inactive	30 / 50
T4	Duration of CLK active	30 / 50
T5	Time to mouse inhibit after clock 11 to ensure the mouse does not start another transmission	>0 / 50

System Sending Data

The following describes the typical sequence of events that takes place when the system is sending data to the mouse. Figure 3-7 on page 3-12 shows timing relationships.

1. The system checks for a mouse transmission in process. If a transmission is in process and beyond the 10th clock, the system must receive the data.
2. The mouse checks the clock line. If the line is inactive, an I/O operation is not allowed.
3. The mouse checks the data line. If the line is inactive, the system has data to transmit. The start bit (always 0) sets the data line inactive.
4. The mouse sets the clock line inactive. The system then places the first bit on the data line. Each time the mouse sets the clock line inactive, the system places the next bit on the data line until all bits are transmitted.
5. The mouse samples the data line for each bit while the clock line is active. Data must be stable within 1 microsecond after the rising edge of the clock line.
6. The mouse checks for a positive-level stop bit after the 10th clock. If the data line is inactive, the mouse continues to clock until the

data line becomes active, clocks the line-control bit, and at the next opportunity sends a Resend command to the system.

7. The mouse sets the data line inactive, producing the line-control bit.
8. The system can set the clock line inactive, inhibiting the mouse.

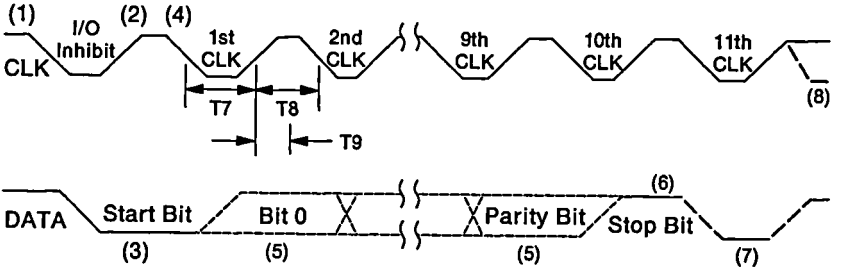


Figure 3-7. Sending Data Timings

	Timing Parameter	Min/Max (microseconds)
T7	Duration of CLK inactive	30 / 50
T8	Duration of CLK active	30 / 50
T9	Time from inactive to active CLK transition, used to time when the mouse samples DATA	5 / 25

Signals

The keyboard and mouse signals are driven by open-collector drivers pulled to 5 V dc through 4.7K ohm resistors. Figure 3-8 shows the characteristics of the signals.

Figure 3-8. Keyboard/Mouse Signals

Signal		
Sink Current	20 mA	Maximum
High-level Output Voltage	+5.0 V dc minus pullup	Minimum
Low-level Output Voltage	+0.5 V dc	Maximum
High-level Input Voltage	+2.0 V dc	Minimum
Low-level Input Voltage	+0.8 V dc	Maximum

Connector

The keyboard and the mouse connectors are 6-pin miniature DIN connectors. Figure 3-9 shows the voltages and signals assigned to the keyboard/mouse connectors.

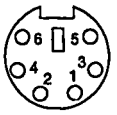


Figure 3-9. Keyboard/Mouse Connectors Voltage and Signal Assignments

Pin	I/O	Signal Name
1	I/O	Data
2	NA	Ground
3	NA	Ground
4	NA	+5 V dc
5	I/O	Clock
6	NA	Ground

Video Subsystem

The system video is generated by the IBM Video Graphics Array (VGA) and its associated circuitry. The associated circuitry consists of the video memory and a video digital-to-analog converter (DAC). The 256KB of video memory consists of four 64KB by 8 memory maps. The red, green, and blue (RGB) outputs from the video DAC drive analog displays with 31.5 KHz horizontal deflection.

All video modes available on the IBM Monochrome Display Adapter, IBM Color/Graphics Monitor Adapter, and IBM Enhanced Graphics Adapter are supported. All video modes supported by the video subsystem are available on all of the supported analog displays. Colors will be displayed as shades of gray when the monochrome analog display is used.

The modes available are:

- 640 by 480 graphics in both 2 and 16 colors
- 720 by 400 alphanumeric in both 16-color and monochrome
- 360 by 400 16-color alphanumeric
- 320 by 200 graphics with 256 colors.

In addition, all 200 line modes are double-scanned by the video subsystem and displayed as 400 lines on the display. This means that each 1-PEL-high horizontal scan line is displayed twice on the display.

The VGA interfaces with the system microprocessor and video memory. All data passes through the VGA when the system microprocessor writes to or reads from video memory. The VGA controls the arbitration for video memory between the system microprocessor and the cathode ray tube (CRT) controller function contained within the VGA. Programs do not need to wait for horizontal retrace to update the display buffer. The system microprocessor receives better performance when accessing the display buffer during non-active display times because there is less interference from the CRT controller.

Video memory addressing is controlled by the VGA. The starting address of the video memory is programmable to three different starting addresses for compatibility with previous video adapters. BIOS will program the VGA appropriately during a video mode set.

In alphanumeric modes, the system microprocessor writes ASCII character code and attribute data to the video memory maps 0 and 1, respectively. The character generator is stored in video map 2 and is loaded by BIOS during an alphanumeric video mode set. BIOS downloads the character generator data (character generator = font = character set) from system ROM. Three fonts are contained in ROM. Two fonts contain dot patterns identical to those provided by the IBM Monochrome Display Adapter, the IBM Color/Graphics Monitor Adapter, and the IBM Enhanced Graphics Adapter. The third font is an 8 x 16 character font. Up to eight 256-character fonts can be loaded into video memory map 2 at a time (the IBM Enhanced Graphics Adapter allows up to four fonts). A BIOS interface exists to load user-defined fonts. As on the IBM Enhanced Graphics Adapter, a register selects which font is actually used to form characters. Also, as on the IBM Enhanced Graphics Adapter, the intensity bit in the attribute byte may be redefined as a switch between two 256-character fonts. This allows 512 characters to be displayed on the screen at one time. See "Character Map Select Register" on page 3-61 and "RAM-Loadable Character Generator" on page 3-97.

The VGA formats the information stored in video memory into an 8-bit digital value that is sent to the video DAC. This 8-bit value allows access to a maximum of 256 registers inside the video DAC. For example, in the two-color graphics modes, only two different 8-bit values would be presented to the video DAC. In the 256-Color Graphics mode, 256 different 8-bit values would be presented to the video DAC. Each register inside the video DAC contains a color value that is selected from a choice of over 256,000 colors.

The video DAC outputs three analog color signals (red, green, and blue) that are sent to the display connector. The monochrome analog display uses only the green analog output. This output is used to determine the shade of gray that is displayed.

The video subsystem supports attachment of only 31.5 KHz direct-drive analog displays. Other IBM displays are not supported because they have digital interfaces, or have a different horizontal sweep frequency.

A BIOS call enables or disables the VGA. Disable means that the VGA will not respond to video memory or I/O reads or writes. The contents of registers and video memory are preserved with the values present when the disable is invoked. Because of this, the VGA continues to generate valid video output if it was doing so before it was disabled.

Compatibility with other hardware is best achieved by using the BIOS interface whenever possible. If an application is forced to write directly to the VGA, the following rules should be followed:

- To program address registers, all currently reserved bits should be set to 0 to maximize compatibility with other hardware.
- To program data registers, all currently reserved bits should be read out and written back unmodified to maximize compatibility with other hardware.

Previous video adapters required that the video mode used correspond to the display attached. For example, the IBM Enhanced Graphics Adapter required that the Enhanced Color Display be attached to run mode hex 3*, and required that the monochrome display be attached to run mode hex 7. All the modes supported by the VGA are supported by the IBM 31.5 KHz direct-drive analog displays. Colors are displayed as shades of gray when the monochrome analog display is connected. Circuitry on the system board detects which type of analog display is connected (color or monochrome). BIOS maps (sums) the colors into shades of gray.

Figure 3-10 on page 3-17 is a block diagram of the video subsystem on the system board.

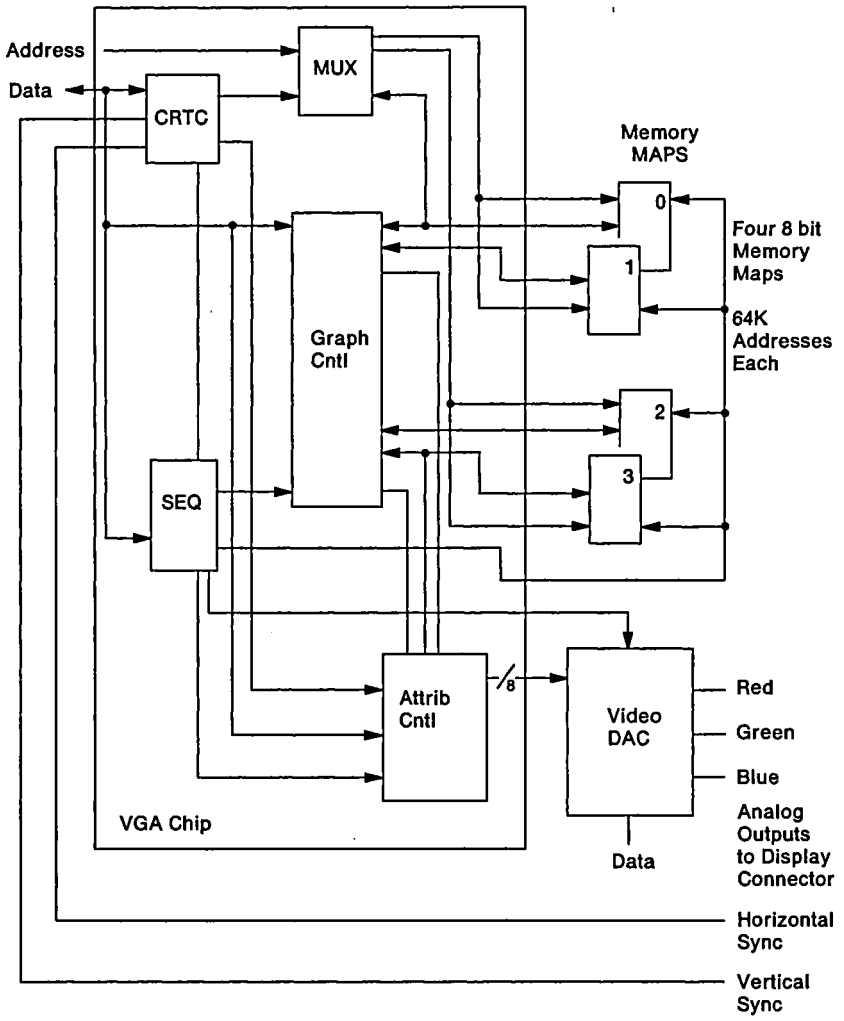


Figure 3-10. Video Subsystem Block Diagram

VGA Components

Most of the logic for the VGA is contained in one module. This module contains all the circuits necessary to generate the timing for the video memory and generates the video information that goes to the video DAC.

BIOS ROM

Software support is provided by video BIOS. Video BIOS is part of the system BIOS. BIOS is contained in the read-only memory (ROM) on the system board. This ROM BIOS contains the character generators and the control code to run the video subsystem.

Support Logic

Two clock sources (25.175 MHz and 28.322 MHz) provide the dot rate. The clock source is selected by setting a bit in a sequencer register. This is done by BIOS when a mode set is done.

The digital video output is sent to the video digital-to-analog converter (DAC), which contains a color look-up table. Three analog signals (red, green, blue) are output from the DAC and are sent to the display. (See "Attribute Controller" on page 3-21.) The sync signals to the monitor are TTL levels. The analog video signals are 0 to 0.7 volts.

The maximum number of colors displayed is 16 out of 256K, except mode hex 13, which can display 256 out of 256K. The maximum number of shades of gray is 16 out of 64, except mode hex 13, which can display 64 out of 64 shades of gray.

Video Graphics Array Major Components

The video graphics array has four major components:

- CRT controller
- Sequencer
- Graphics controller
- Attribute controller.

CRT Controller

The cathode ray tube (CRT) controller generates horizontal and vertical synchronous timings, addressing for the regenerative buffer, cursor and underline timings, and refresh addressing for the dynamic RAMs.

Sequencer

The sequencer generates basic memory timings for the dynamic RAMs and the character clock for controlling regenerative memory fetches. It allows the system microprocessor to access memory during active display intervals by inserting dedicated system microprocessor memory cycles periodically between the display memory cycles. Map Mask registers are available to protect entire memory maps from being changed.

The graphics controller is the interface between video memory and either the attribute controller during active display times or the system microprocessor during video memory reads or writes. During display times, memory data is latched and sent to the attribute controller. In all points addressable (APA) modes, the parallel memory data is converted to serial bit-plane data before being sent. In alphanumeric (A/N) modes, the parallel attribute data is sent directly. During a system microprocessor write or read to video memory, the graphics controller can perform logical operations on the memory data before it reaches video memory or the system microprocessor data bus, respectively. These operations include four logical write modes and two logical read modes. These features allow enhanced operations such as a color-compare read mode, individual bit masking during write modes, 32-bit writes in a single memory cycle, and writing to the display buffer on non-byte boundaries.

Figure 3-11 is a block diagram of the graphics controller.

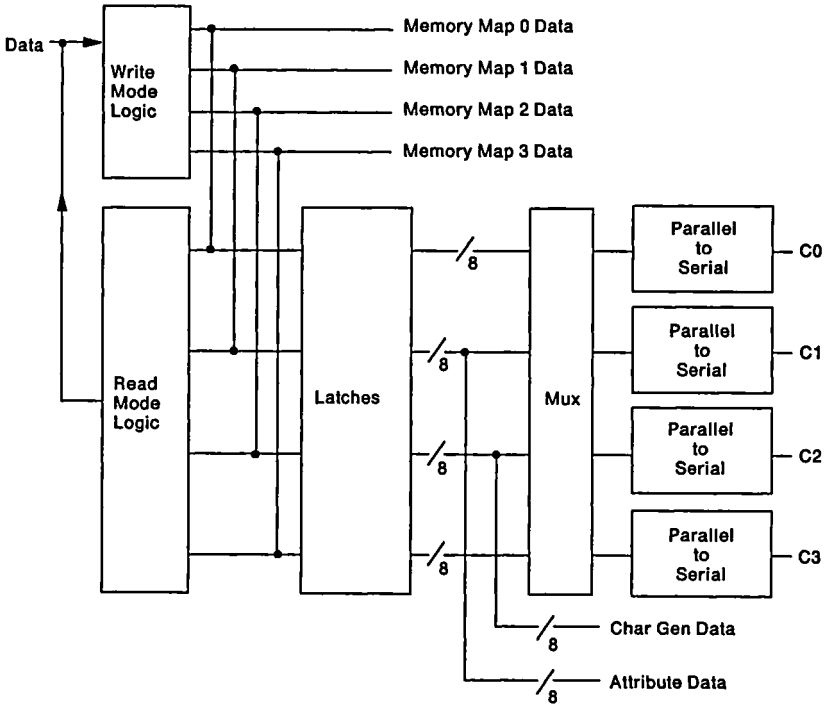


Figure 3-11. Graphics Controller Block Diagram

Attribute Controller

The attribute controller takes in data from video memory through the graphics controller and formats it for display. Incoming attribute data in A/N mode, and serialized bit-plane data in APA mode is converted to an 8-bit output digital color value. Each output color value is selected from an internal color palette of 64 possible colors (except in 256-Color mode). The output color value is sent to the integrated DAC, where it is used as an address into an 18-bit color register whose value is in turn converted to three analog color signals that drive the display. Blinking, underlining, cursor insertion, and PEL panning are also controlled by the attribute controller.

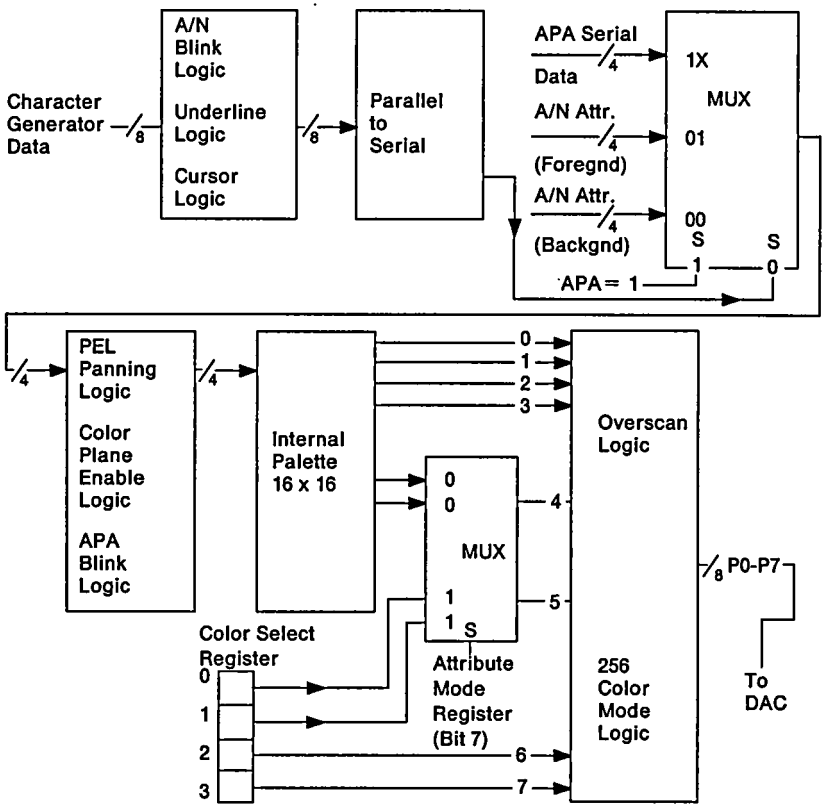


Figure 3-12. Attribute Controller Block Diagram

Modes of Operation

Figure 3-13 describes the modes supported by BIOS on IBM 31.5 KHz direct-drive analog color and monochrome displays.

Figure 3-13. BIOS Video Modes

Mode (hex)	Type	Alpha Format	Buffer Start	Box Size	Max Pgs.	Vert. Freq.	PELS	
	Colors							
0, 1	A/N	16/256K	40 x 25	B8000	8 x 8	8	70 Hz	320 x 200
2, 3	A/N	16/256K	80 x 25	B8000	8 x 8	8	70 Hz	640 x 200
0*, 1*	A/N	16/256K	40 x 25	B8000	8 x 14	8	70 Hz	320 x 3/0
2*, 3*	A/N	16/256K	80 x 25	B8000	8 x 14	8	70 Hz	640 x 350
0+, 1+	A/N	16/256K	40 x 25	B8000	9 x 16	8	70 Hz	360 x 400
2+, 3+	A/N	16/256K	80 x 25	B8000	9 x 16	8	70 Hz	720 x 400
4, 5	APA	4/256K	40 x 25	B8000	8 x 8	1	70 Hz	320 x 200
6	APA	2/256K	80 x 25	B8000	8 x 8	1	70 Hz	640 x 200
7	A/N	-	80 x 25	B0000	9 x 14	8	70 Hz	720 x 350
7+	A/N	-	80 x 25	B0000	9 x 16	8	70 Hz	720 x 400
D	APA	16/256K	40 x 25	A0000	8 x 8	8	70 Hz	320 x 200
E	APA	16/256K	80 x 25	A0000	8 x 8	4	70 Hz	640 x 200
F	APA	-	80 x 25	A0000	8 x 14	2	70 Hz	640 x 350
10	APA	16/256K	80 x 25	A0000	8 x 14	2	70 Hz	640 x 350
11	APA	2/256K	80 x 30	A0000	8 x 16	1	60 Hz	640 x 480
12	APA	16/256K	80 x 30	A0000	8 x 16	1	60 Hz	640 x 480
13	APA	256/256K	40 x 25	A0000	8 x 8	1	70 Hz	320 x 200

* Enhanced modes (EGA)
\+ Enhanced modes (VGA)

When the color display is used, each color is selected from a palette of over 256,000 colors.

When the monochrome display is used, the colors are displayed as shades of gray. Each shade of gray is selected from a palette of 64 shades.

Modes hex 0 through 6 emulate the support provided by the IBM Color/Graphics Monitor Adapter.

Modes hex 0, 2, and 4 are identical to modes hex 1, 3, and 5, respectively. On the IBM Color/Graphics Monitor Adapter there is a difference in these modes. In modes hex 0, 2, and 4, the color burst was turned off. Color burst is not provided by the video subsystem. Mode hex 3+ is the default mode with an analog color display attached to the system. Mode hex 7+ is the default mode with an analog monochrome display attached to the system.

Mode hex 7 emulates the support provided by the IBM Monochrome Display Adapter.

Modes hex 0*, 1*, 2*, 3*, D, E, F, and 10 emulate the support provided by the IBM Enhanced Graphics Adapter.

Double-scan means that each horizontal scan line is displayed twice. This is used for 200-line modes, which are displayed as 400 lines. Border support depends on the BIOS mode selected. Figure 3-14 shows double-scan and border support.

Figure 3-14. BIOS Double-Scan and Border Support

Mode (hex)	Double Scan	Border Support
0, 1	Yes	No
2, 3	Yes	Yes
0*, 1*	No	No
2*, 3*	No	Yes
0+, 1+	No	No
2+, 3+	No	Yes
4, 5	Yes	No
6	Yes	Yes
7	No	Yes
7+	No	Yes
D	Yes	No
E	Yes	Yes
F	No	Yes
10	No	Yes
11	No	Yes
12	No	Yes
13	Yes	Yes

* Enhanced modes (EGA)
\+ Enhanced modes (VGA)

Display Support

The video subsystem supports attachment of 31.5 KHz horizontal sweep frequency direct-drive analog displays. These displays have a vertical sweep frequency capability of 60 to 70 cycles per second, providing extended color and sharpness and reduced flicker in most modes. Other IBM displays are not supported because they have digital interfaces, or have a different horizontal and vertical sweep frequency. Figure 3-15 summarizes the analog display characteristics.

Figure 3-15. IBM 31.5 KHz Direct-Drive Analog Displays

Parameter	Color	Monochrome
Horizontal Scan Rate	31.5 KHz	31.5 KHz
Vertical Scan Rate	60 to 70 Hz	60 to 70 Hz
Video Bandwidth	28 MHz	28 MHz
Displayable Colors*	256/256K Maximum	64/64 Shades Gray
Maximum Horizontal Resolution	720 PELs	720 PELs
Maximum Vertical Resolution	480 PELs	480 PELs

* Controlled by Video Circuit.

Since both color and monochrome displays run at the same sweep rate, all modes work on either type. The vertical size of the display is controlled by the polarity of the vertical and horizontal sync pulses. This is done so that 350, 400, or 480 lines can be displayed without adjusting the display. (See "Display Connector Timing (SYNC Signals)" on page 3-105.)

Video Subsystem Programmable Option Select

The video subsystem supports Programmable Option Select (POS). When the POS sleep bit is set, the video subsystem does not respond to any memory or I/O reads or writes. Video is still generated if the video subsystem is programmed to do so. POS must be enabled for video subsystem operation.

The implementation of POS for the video subsystem is as follows:

- When in Setup mode (I/O address hex 0094, bit 5 equals 0), the VGA responds to a single option select byte at I/O address hex 0102 and treats the LSB (bit 0) of that byte as the VGA sleep bit. When the LSB is 0, the VGA does not respond to commands, addresses, or data on the data bus. When the LSB is 1, the VGA responds. If the VGA was set up and is generating video output when the LSB is set to 0, the output is still generated.
- The VGA responds only to address hex 0102 when in the Setup mode. No other addresses are valid at that time. Conversely, the VGA ignores address hex 0102 when in the Enabled mode (I/O address hex 0094, bit 5 equals 1), and decodes normal I/O and memory addresses.

Note: When VGA is disabled, accesses to the video DAC registers are disabled.

When the system is powered on, POST initializes and enables the video subsystem.

For information on BIOS calls to enable or disable the VGA, see *BIOS Interface Technical Reference for IBM PS/1™ Computer.*

Alphanumeric Modes

This section describes the alphanumeric modes supported by the video subsystem and BIOS. Note that the colors in this section are generated when the BIOS is used to set the mode. BIOS initializes the video subsystem and the video DAC palette to generate these colors. If the video DAC palette is changed, different colors are generated.

The alphanumeric modes are modes hex 0 - 3 and 7. The mode chart lists the variations of these modes. The data format for alphanumeric modes is the same as the data format on the IBM Color/Graphics Monitor Adapter, the IBM Monochrome Display Adapter, and the IBM Enhanced Graphics Adapter (EGA). As in the EGA, bit 3 of the attribute byte may be redefined by the Character Map Select register to act as a switch between character sets. This gives the programmer access to 512 characters at one time.

When an alphanumeric mode is selected, the BIOS transfers character patterns from the ROM to map 2. The system microprocessor stores the character data in map 0, and the attribute data in map 1. The programmer can view maps 0 and 1 as a single buffer in alphanumeric modes. The CRT controller generates sequential addresses, and fetches one character code byte and one attribute byte at a time. The character code and row scan count are combined to address map 2, which contains the character generators. The appropriate dot patterns are then sent to the palette in the attribute section, where color is assigned according to the attribute data.

Every display-character position in the alphanumeric mode is defined by 2 bytes in the display buffer. Figure 3-16 shows the 2-byte character/attribute format used in both the color/graphics and the monochrome emulation modes.

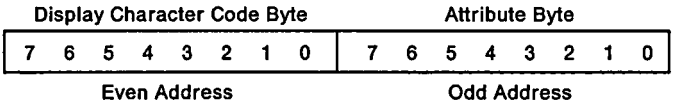


Figure 3-16. Character/Attribute Format

See Section 11, "Characters and Keystrokes" on page 11-1 for characters loaded during BIOS mode sets.

Figure 3-17 shows the functions of the attribute byte.

Figure 3-17. Attribute Byte Functions

Attribute Function	Attribute Byte 7 6 5 4 B/I R G B Background	Attribute Byte 3 2 1 0 I/CS R G B Foreground
Normal (White on Black)	B/I 0 0 0	I/CS 1 1 1
Reverse (Black on White)	B/I 1 1 1	I/CS 0 0 0
Nondisplay (Black)	B/I 0 0 0	I/CS 0 0 0
Nondisplay (White)	B/I 1 1 1	I/CS 1 1 1
Mono=Underline / Color=Blue	B/I 0 0 0	I/CS 0 0 1

I = Highlighted
B = Blinking Foreground (Character)
CS = Character select

The BIOS defaults on a mode set are blink for bit 7 (B/I) and intensity for bit 3 (I/CS).

Figure 3-18 shows the attribute byte definitions.

Figure 3-18. Attribute Byte Definitions

Bit	Color	Function
7	B/I	Blinking/Background Intensity
6	R	Red Background
5	G	Green Background
4	B	Blue Background
3	I/CS	Intensity/Character Select
2	R	Red Foreground
1	G	Green Foreground
0	B	Blue Foreground

See "Character Map Select Register" on page 3-61 and "Attribute Mode Control Register" on page 3-89.

Any other code combination produces white-on-white in the monochrome emulation mode and the following colors in the color emulation mode:

Figure 3-19. Attribute Byte Colors

I	R	G	B	Color
0	0	0	0	Black
0	0	0	1	Blue
0	0	1	0	Green
0	0	1	1	Cyan
0	1	0	0	Red
0	1	0	1	Magenta
0	1	1	0	Brown
0	1	1	1	White
1	0	0	0	Gray
1	0	0	1	Light Blue
1	0	1	0	Light Green
1	0	1	1	Light Cyan
1	1	0	0	Light Red
1	1	0	1	Light Magenta
1	1	1	0	Yellow
1	1	1	1	White (High Intensity)

Both 40-column and 80-column alphanumeric modes are supported. The 40-column alphanumeric modes (all variations of modes hex 0 and 1) have the following features:

- Display up to 25 rows of 40 characters each
- Require 2,000 bytes of read/write memory per page
- One character and one attribute for each character.

The 80-column alphanumeric modes (all variations of modes hex 2, 3, and 7) have the following features:

- Display up to 25 rows of 80 characters each
- Require 4,000 bytes of read/write memory per page
- One character and attribute for each character.

Graphics Modes

This section describes the graphics modes supported by the video subsystem and BIOS. Note that the colors in this section are generated when the BIOS is used to set the mode. BIOS initializes the video subsystem and the video DAC palette to generate these colors. If the video DAC palette is changed, different colors are generated.

320 x 200 Four-Color Graphics (Modes Hex 4 and 5)

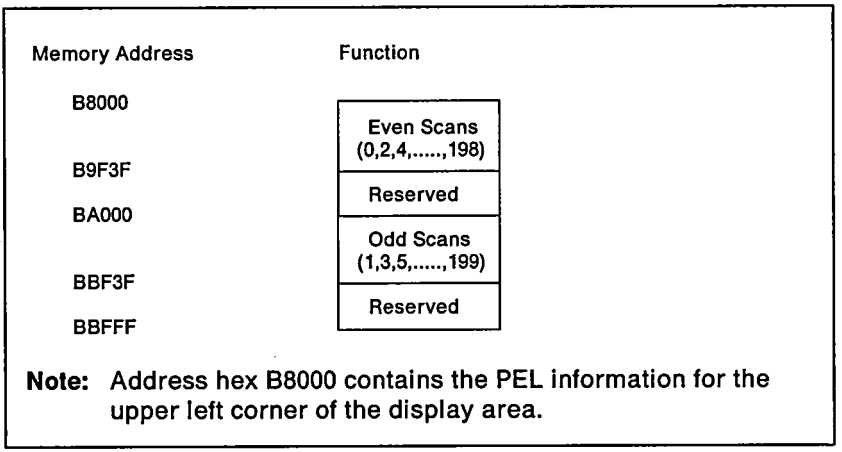
Addressing, mapping, and data format are the same as the 320-by-200-PEL mode of the IBM Color/Graphics Monitor Adapter. The display buffer is configured at hex B8000. Bit image data is stored in memory maps 0 and 1. The two bit planes (C0 and C1) are each formed from bits from both memory maps.

This mode has the following features:

- Contains a maximum of 200 rows of 320 PELs
- Selects one of four colors for each PEL
- Requires 16,000 bytes of read/write memory
- Uses memory-mapped graphics
- Double-scanned on display to 400 rows
- Formats four PELs-per-byte as shown in Figure 3-20.
- Organizes graphics memory in two banks of 8,000 bytes using the format as shown in Figure 3-21 on page 3-29.

Figure 3-20. PEL Format, Modes Hex 4 and 5

Bit	Function
7	C1 - First Display PEL
6	C0 - First Display PEL
5	C1 - Second Display PEL
4	C0 - Second Display PEL
3	C1 - Third Display PEL
2	C0 - Third Display PEL
1	C1 - Fourth Display PEL
0	C0 - Fourth Display PEL



Note: Address hex B8000 contains the PEL information for the upper left corner of the display area.

Figure 3-21. Video Memory Format

Color selection is determined as shown in Figure 3-22.

Figure 3-22. Color Selections, Modes Hex 4 and 5

C1	C0	Color Selected	
0	0	Black	
0	1	Light Cyan	* Green
1	0	Light Magenta	* Red
1	1	Intensified White	* Brown

* Selectable by a video BIOS call.

640 x 200 Two-Color Graphics (Mode Hex 6)

Addressing, mapping, and data format are the same as the 640-by-200-PEL black and white mode of the IBM Color/Graphics Monitor Adapter. The display buffer is configured at hex B8000. Bit image data is stored in memory map 0 and comprises a single bit plane (C0).

This mode has the following features:

- Contains a maximum of 200 rows of 640 PELs.
- Supports two colors only.
- Requires 16,000 bytes of read/write memory.
- Addressing and mapping procedures are the same as 320-by-200 two- and four-color graphics, but the data format is different. In this mode, each bit in memory is mapped to a PEL on the screen.
- Double-scanned on display to 400 rows.
- Formats 8 PELs per byte in the following manner:

Figure 3-23. PEL Format, Mode Hex 6

Bit	Function
7	First Display PEL
6	Second Display PEL
5	Third Display PEL
4	Fourth Display PEL
3	Fifth Display PEL
2	Sixth Display PEL
1	Seventh Display PEL
0	Eighth Display PEL

The bit definition for each PEL is 0 for Black and 1 for Intensified White.

640 x 480 Two-Color Graphics (Mode Hex 11)

This mode provides two-color graphics with the same data format as mode 6. Addressing and mapping is shown under "Video Memory Organization" on page 3-35.

The bit image data is stored in map 0 and comprises a single bit plane (C0). A sequential buffer starting at address hex A0000 is provided. Location hex A0000 contains the byte with information for the first 8 PELs; location hex A0001 contains information for the second 8 PELs, and so on. The bit definition for each PEL is 0 for Black and 1 for Intensified White.

640 x 350 Graphics (Mode Hex F)

This mode emulates the EGA graphics mode on the IBM Monochrome Display with the following attributes: black, white, blinking white, and intensified white. Resolution of 640-by-350 requires 56KB to support four attributes. Maps 0 and 2 are used in this mode. Map 0 is the video bit plane, and map 2 is the intensity bit plane. Both planes reside at address hex A0000.

Two bits, one from each bit plane, define one PEL on the screen. Figure 3-24 shows the PEL bit definitions. C0 is the video bit plane and C2 is the intensity bit plane.

Figure 3-24. PEL Bit Definitions

C2	C0	PEL Color
0	0	Black
0	1	White
1	0	Blinking White
1	1	Intensified White

The byte organization in memory is linear. The first 8 PELs on the screen are defined by the contents of memory in location hex A0000, the second 8 PELs by location hex A0001, and so on. The first PEL within any one byte is defined by bit 7 in the byte. The last PEL within the byte is defined by bit 0 in the byte.

Since both bit planes reside at address hex A0000, the user must select which plane or planes to update. This is accomplished by the Map Mask register of the sequencer. (See "Video Memory Organization" on page 3-35.)

16-Color Graphics (Modes Hex D, E, 10 and 12)

These modes support graphics in 16 colors. The bit image data is stored in all four memory maps in these modes. Each memory map contains the data for one bit plane. Each bit plane represents a color as shown below. The bit planes are denoted as C0, C1, C2, and C3, respectively.

C0 = Blue PELs
C1 = Green PELs
C2 = Red PELs
C3 = Intensified PELs

Four bits (one from each plane) define one PEL on the screen. Figure 3-25 shows the color combinations.

Figure 3-25. Palette Colors

C3	C2	C1	C0	Color
0	0	0	0	Black
0	0	0	1	Blue
0	0	1	0	Green
0	0	1	1	Cyan
0	1	0	0	Red
0	1	0	1	Magenta
0	1	1	0	Brown
0	1	1	1	White
1	0	0	0	Dark Gray
1	0	0	1	Light Blue
1	0	1	0	Light Green
1	0	1	1	Light Cyan
1	1	0	0	Light Red
1	1	0	1	Light Magenta
1	1	1	0	Yellow
1	1	1	1	Intensified White

The display buffer resides at address hex A0000. The Map Mask register is used to select any or all of the maps to be updated when a memory write to the display buffer is executed by the system microprocessor.

256-Color Graphics (Mode Hex 13)

This mode provides graphics with the capability to display 256 colors on the screen at one time.

The display buffer is linear, starting at address hex A0000, and is 64,000 bytes long. The first location contains the 8-bit color for the upper left PEL. The second byte contains the second PEL, and so on for 64,000 PELs (320-by-200). The bit image data is stored in all four memory maps and comprises four bit planes. The four bit planes are sampled twice to produce eight bit planes that address the video DAC.

The internal palette of the video subsystem is not used to select colors. It is set by BIOS and should not be changed. The external palette in the video DAC is programmed by the BIOS such that the first 16 locations contain colors that are compatible with colors of other modes (See Figure 3-26). The second 16 locations contain 16 evenly spaced gray shades. The remaining 216 locations are loaded, based on a hue-saturation-intensity model tuned to provide a usable, generic color set that covers a wide range of color values.
Figure 3-26 shows the colors that are compatible with colors of other modes.

Figure 3-26. Attribute Byte

Attribute Byte								Analog Output Color
C7	C6	C5	C4	C3	C2	C1	C0	
0	0	0	0	0	0	0	0	Black
0	0	0	0	0	0	0	1	Blue
0	0	0	0	0	0	1	0	Green
0	0	0	0	0	0	1	1	Cyan
0	0	0	0	0	1	0	0	Red
0	0	0	0	0	1	0	1	Magenta
0	0	0	0	0	1	1	0	Brown
0	0	0	0	0	1	1	1	White
0	0	0	0	1	0	0	0	Dark Gray
0	0	0	0	1	0	0	1	Light Blue
0	0	0	0	1	0	1	0	Light Green
0	0	0	0	1	0	1	1	Light Cyan
0	0	0	0	1	1	0	0	Light Red
0	0	0	0	1	1	0	1	Light Magenta
0	0	0	0	1	1	1	0	Yellow
0	0	0	0	1	1	1	1	Intensified White

The video DAC palette can be programmed from a selection of over 256,000 different colors.

This mode has the following features:

- Contains a maximum of 200 rows of 320 PELs
- Double-scanned on display to 400 rows
- Selects one of 256 colors for each PEL
- Requires 64,000 bytes of read/write memory
- Uses memory-mapped graphics
- Uses 1 byte of memory for each PEL.

Video Memory Organization

The video display buffer on the system board consists of 256KB of dynamic read/write memory configured as four 64KB video maps.

The address of the display buffer can be changed to remain compatible with other video adapters and application software. Four locations are provided. The buffer can be configured at segment address hex A0000 for a length of 128KB, at hex A0000 for a length of 64KB, at hex B0000 for a length of 32KB, or at hex B8000 for a length of 32KB.

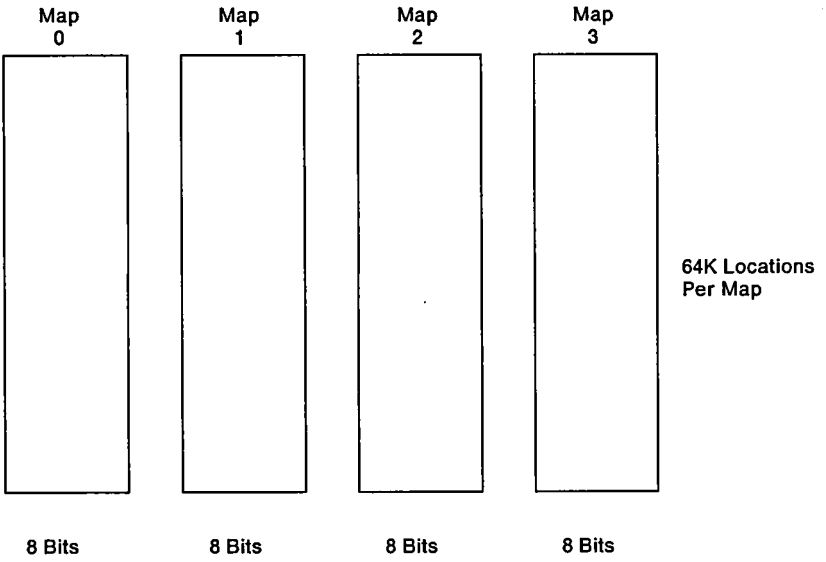


Figure 3-27. 256KB Video Memory Map

Maps 0 through 3 usually form bit planes 0 through 3:

Map 0 = Bit Plane 0

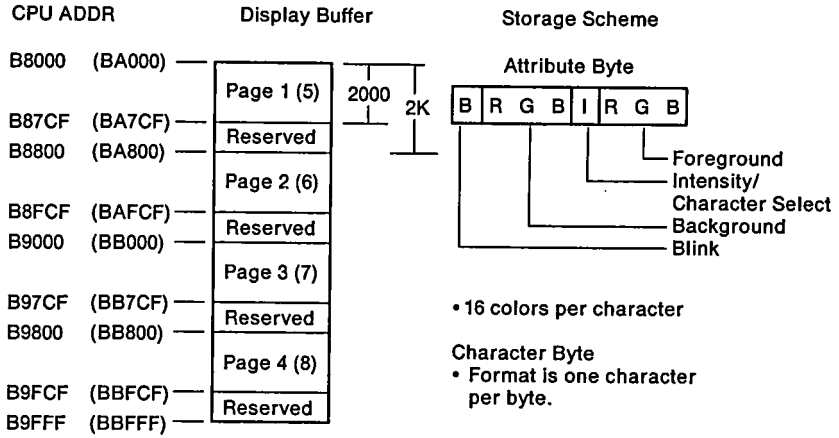
Map 1 = Bit Plane 1

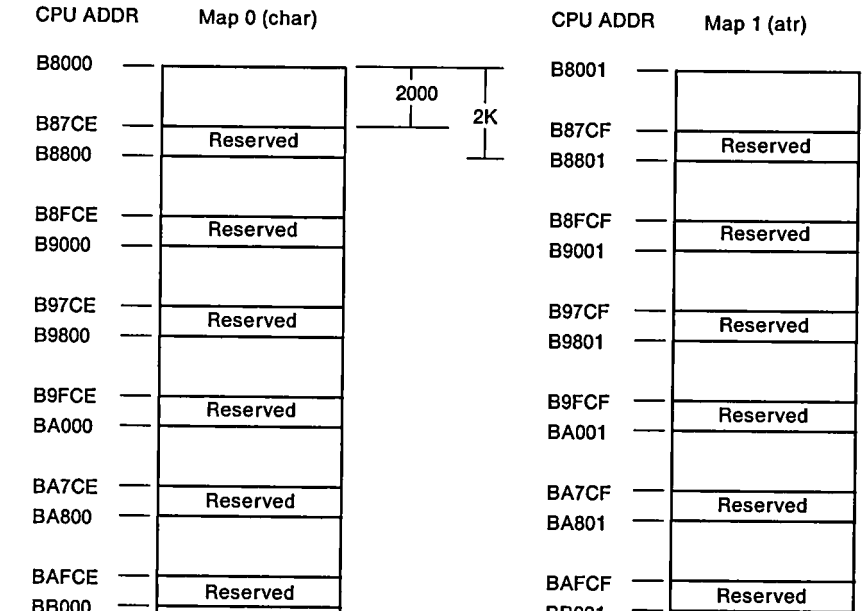
Map 2 = Bit Plane 2

Map 3 = Bit Plane 3

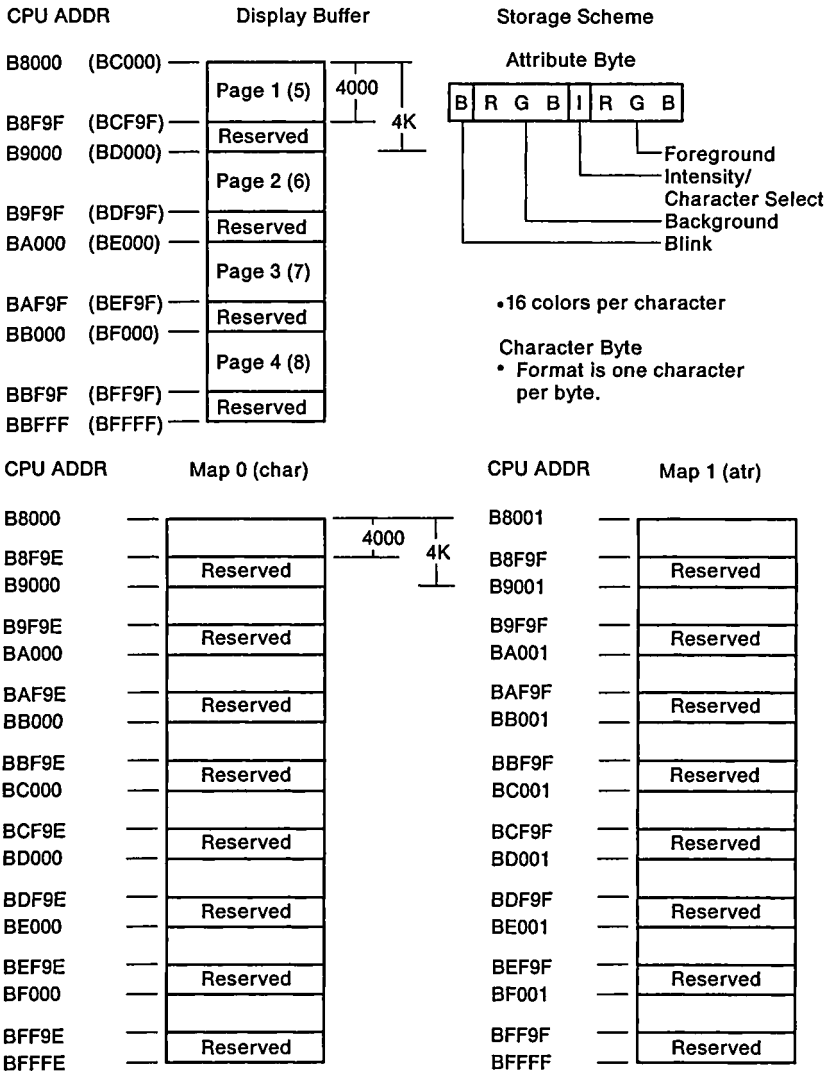
In mode hex 13, each of the four bit planes is formed with data from all four maps. The four bits are sampled twice internally to produce the eight bit values needed to select 256 colors.

Modes Hex 0, 1 (All Variations of Modes Hex 0 and 1)

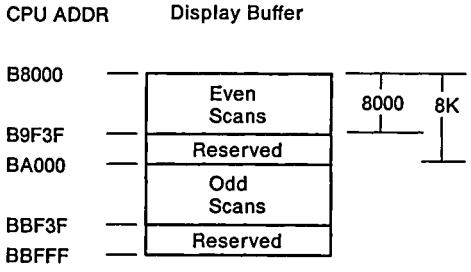


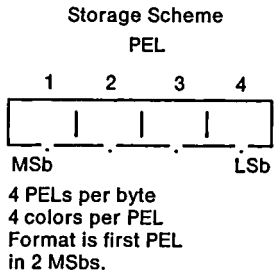


Modes Hex 2, 3 (All Variations of Modes Hex 2 and 3)

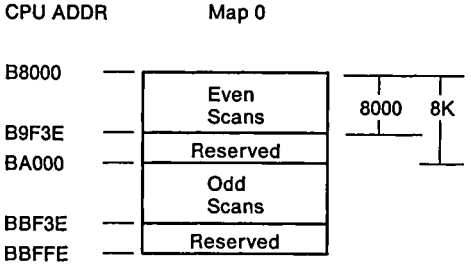


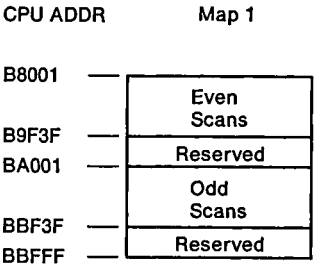
Modes Hex 4, 5



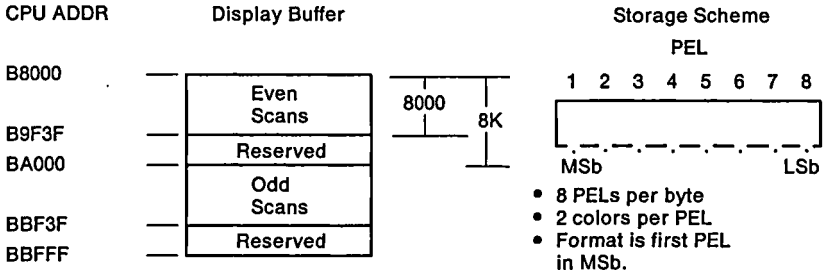


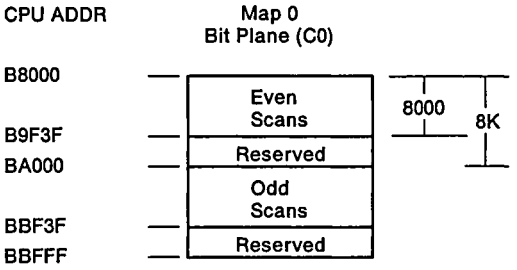
- 4 PELs per byte
- 4 colors per PEL
- Format is first PEL in 2 MSbs.



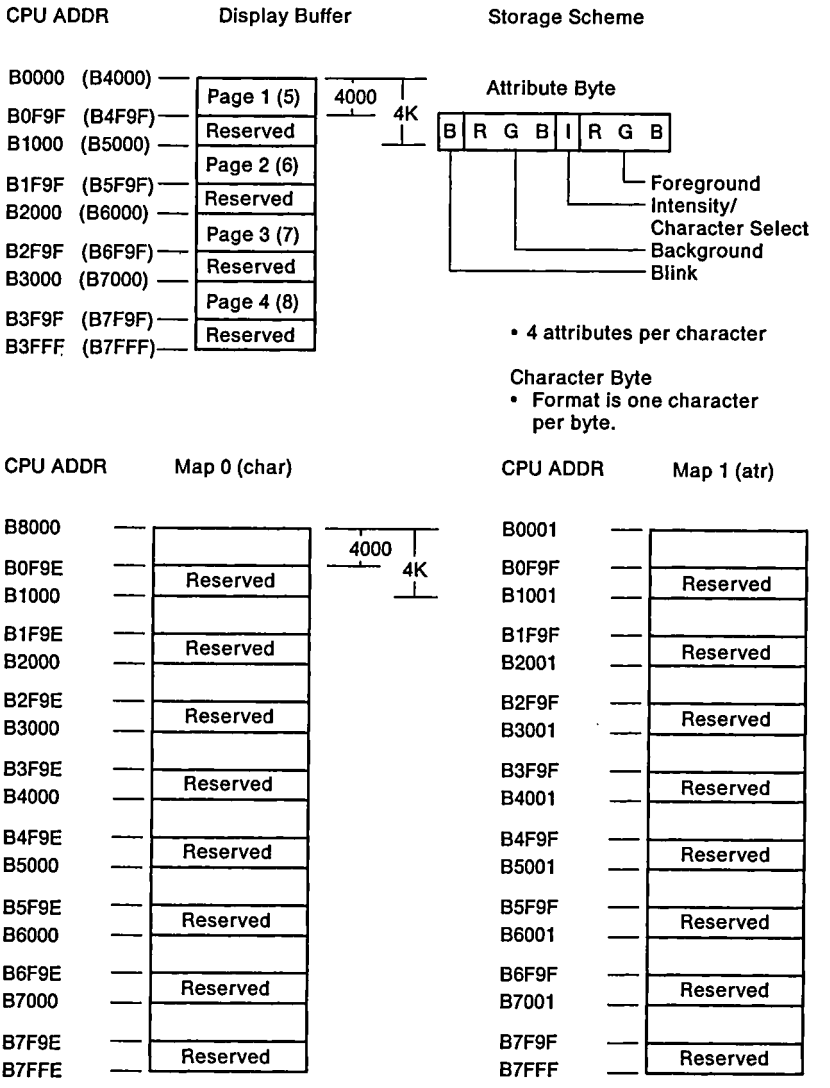


Mode Hex 6

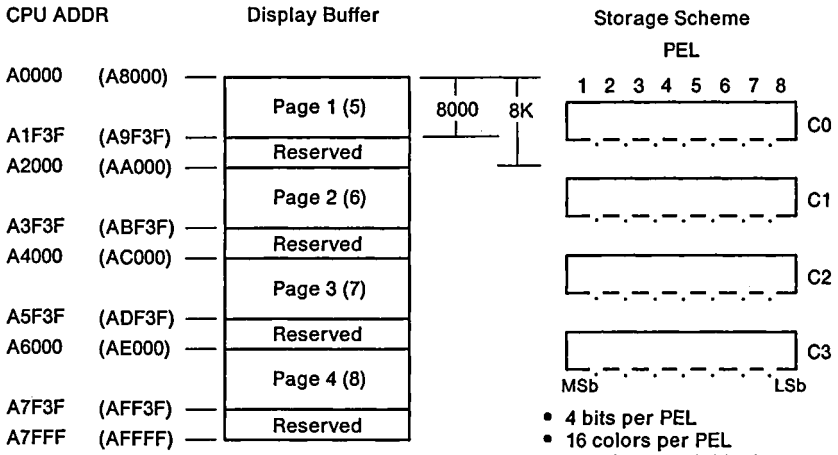

CPU ADDR
Display Buffer
Storage Scheme
PEL
1 2 3 4 5 6 7 8
B8000
Even Scans
B9F3F
Reserved
BA000
Odd Scans
BBF3F
Reserved
BBFFF
8000
8K
MSb
LSb
• 8 PELs per byte
• 2 colors per PEL
• Format is first PEL in MSb.


CPU ADDR
Map 0
Bit Plane (C0)
B8000
Even Scans
B9F3F
Reserved
BA000
Odd Scans
BBF3F
Reserved
BBFFF
8000
8K

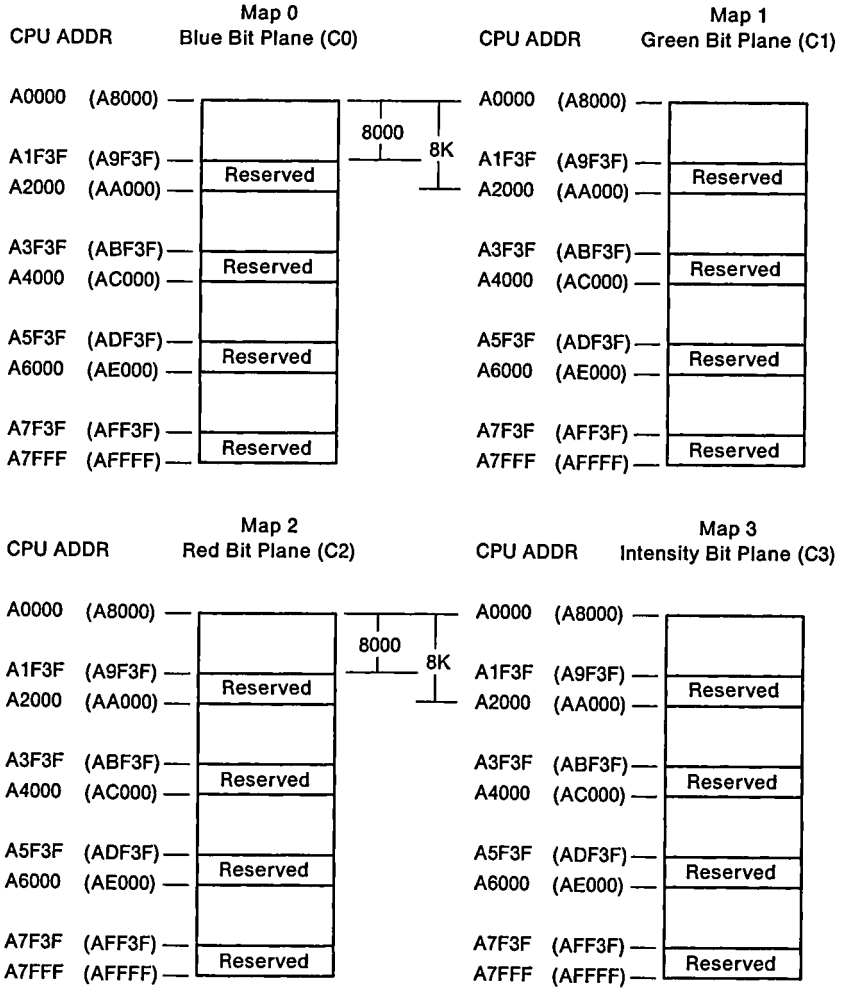
Mode Hex 7 (All Variations of Mode Hex 7)



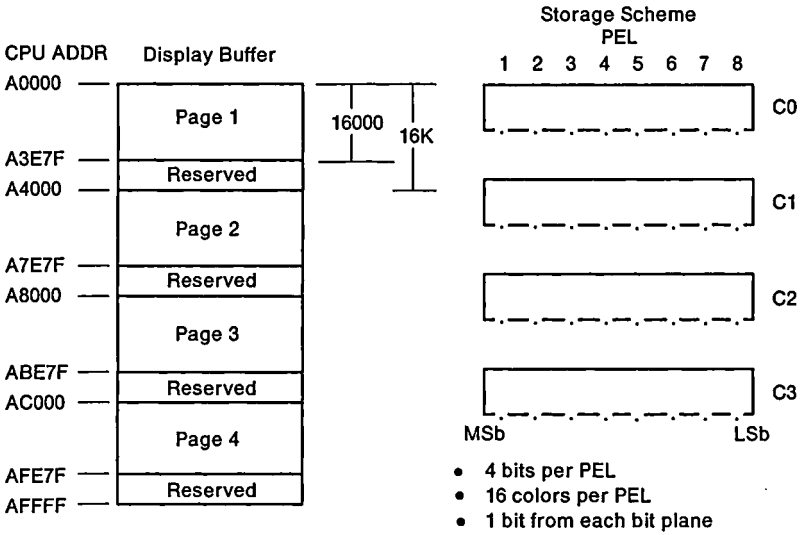
Mode Hex D



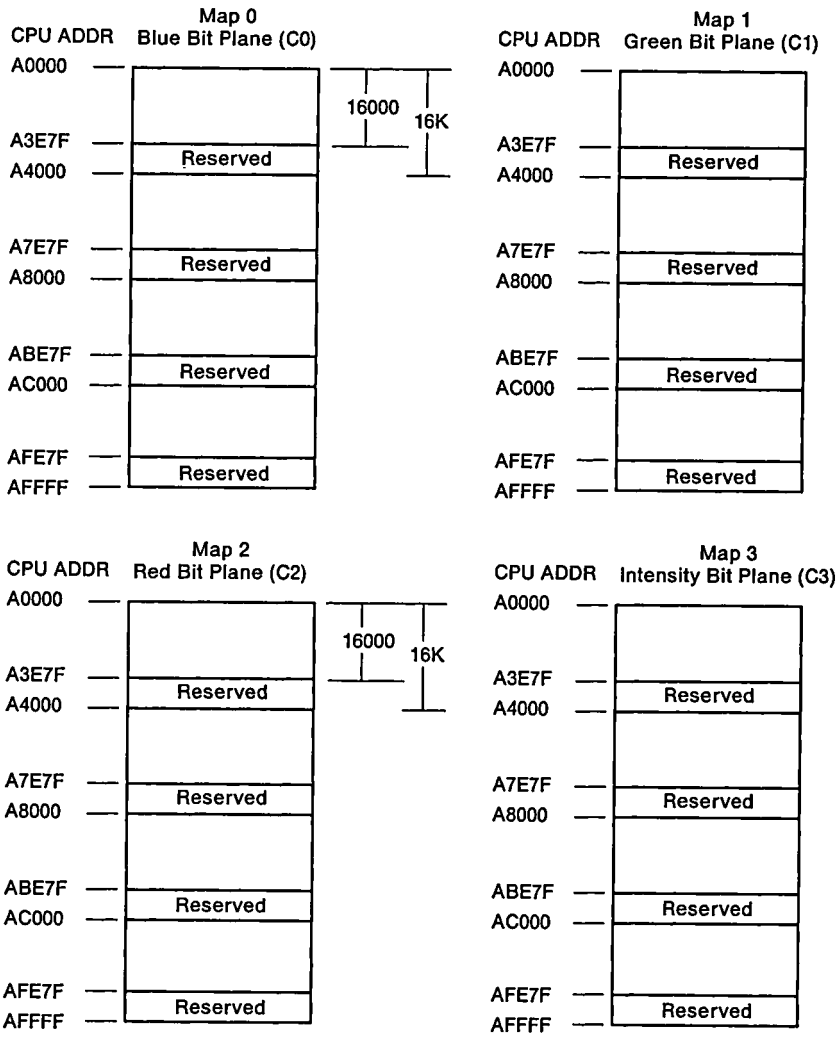
- 4 bits per PEL
- 16 colors per PEL
- 1 bit from each bit plane (C3,C2,C1,CO) per PEL
- Format is first PEL in MSb of all 4 bit planes.


Map 0
CPU ADDR
Blue Bit Plane (C0)
A0000 (A8000)
A1F3F (A9F3F)
A2000 (AA000)
A3F3F (ABF3F)
A4000 (AC000)
A5F3F (ADF3F)
A6000 (AE000)
A7F3F (AFF3F)
A7FFF (AFFFF)
Reserved
8000
8K
Map 1
CPU ADDR
Green Bit Plane (C1)
Map 2
CPU ADDR
Red Bit Plane (C2)
Map 3
CPU ADDR
Intensity Bit Plane (C3)

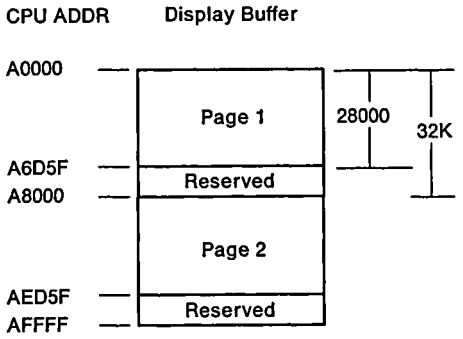
Mode Hex E

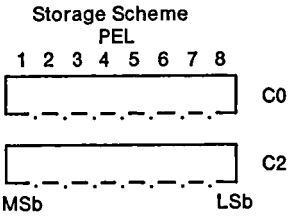


- 4 bits per PEL
- 16 colors per PEL
- 1 bit from each bit plane (C3,C2,C1,C0) per PEL
- Format is first PEL in MSb of all 4 bit planes.

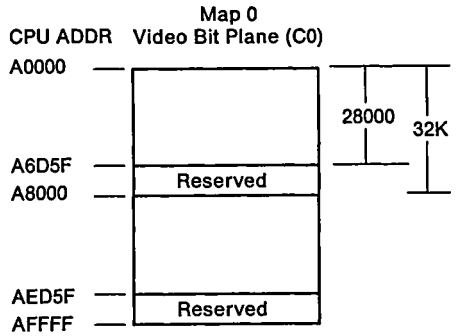

Map 0
CPU ADDR
Blue Bit Plane (C0)
A0000
A3E7F
A4000
A7E7F
A8000
ABE7F
AC000
AFE7F
AFFFF
Reserved
Reserved
Reserved
Reserved
16000
16K
Map 1
CPU ADDR
Green Bit Plane (C1)
A0000
A3E7F
A4000
A7E7F
A8000
ABE7F
AC000
AFE7F
AFFFF
Reserved
Reserved
Reserved
Reserved
Map 2
CPU ADDR
Red Bit Plane (C2)
A0000
A3E7F
A4000
A7E7F
A8000
ABE7F
AC000
AFE7F
AFFFF
Reserved
Reserved
Reserved
Reserved
16000
16K
Map 3
CPU ADDR
Intensity Bit Plane (C3)
A0000
A3E7F
A4000
A7E7F
A8000
ABE7F
AC000
AFE7F
AFFFF
Reserved
Reserved
Reserved
Reserved

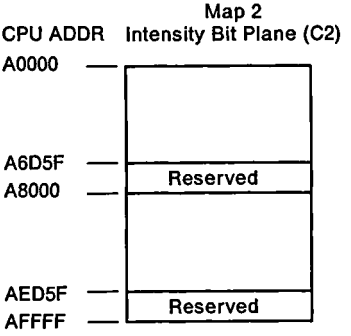
Mode Hex F



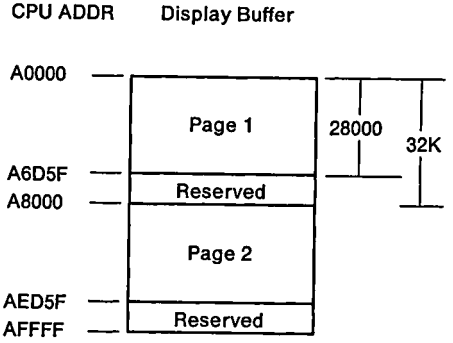


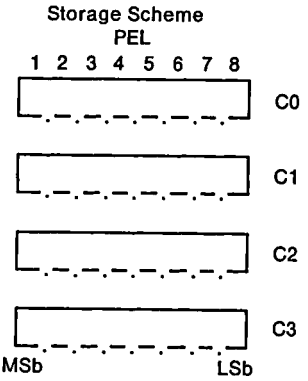
- 2 bits per PEL
- 4 attributes per PEL
- 1 bit from each bit plane (C2,C0)
- Format is first PEL in MSb of video and intensity bit planes.



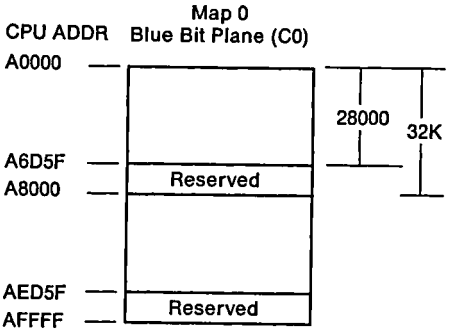


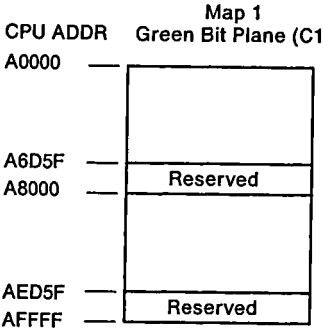
Mode Hex 10

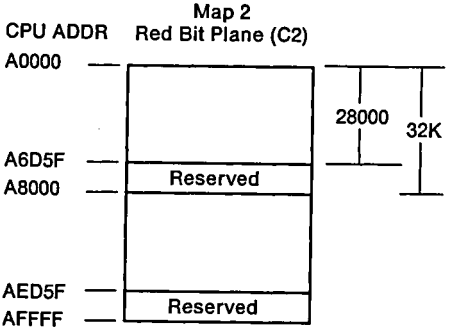


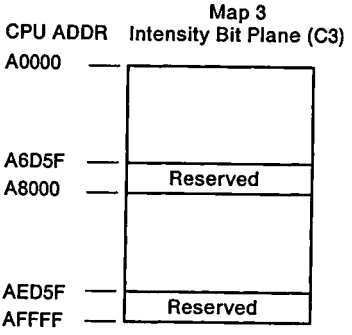


- 4 bits per PEL
- 16 colors per PEL
- 1 bit from each bit plane (C3,C2,C1,C0) per PEL
- Format is first PEL in MSb of all 4 bit planes.

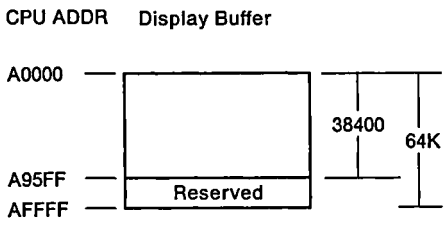




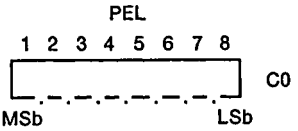




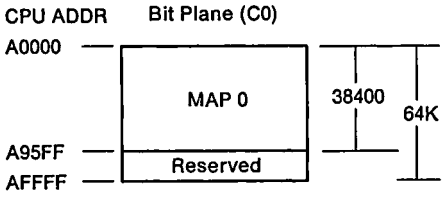
Mode Hex 11



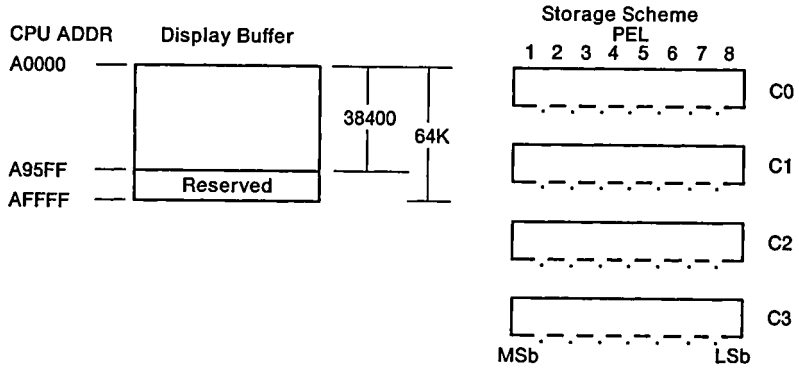
Storage Scheme



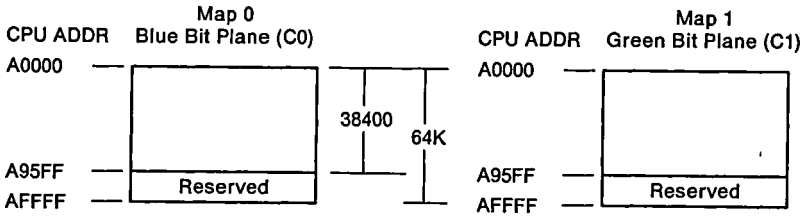
- 1 bit per PEL
- 2 attributes per PEL
- Format is first PEL in MSb position.

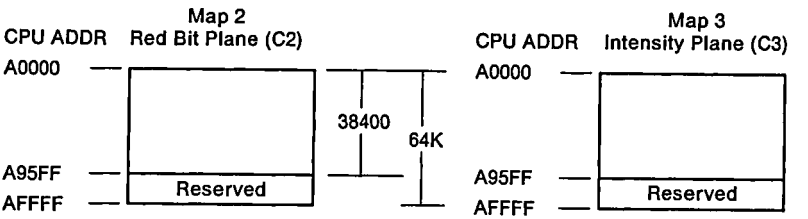


Mode Hex 12

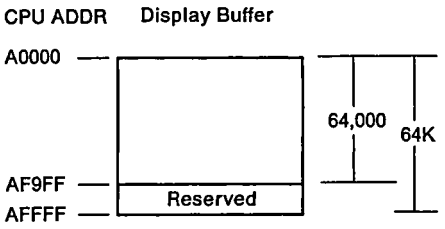


- 4 bits per PEL
- 16 colors per PEL
- 1 bit from each bit plane (C3,C2,C1,C0) per PEL
- Format is first PEL in MSb of all 4 bit planes.

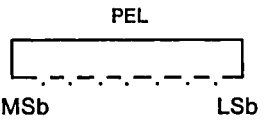




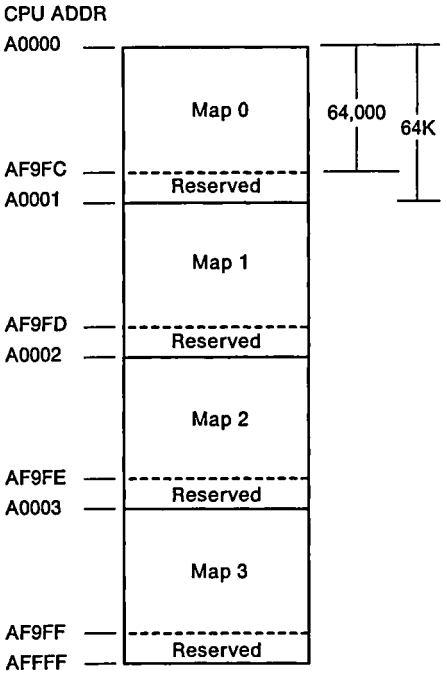
Mode Hex 13



Storage Scheme



- 8 bits per PEL
- 256 colors per PEL
- 1 PEL per byte
- Format is PEL 1 at address A0000.



Video Memory Read/Write Operations

Read Operations

There are two ways to do a video memory read. When read type 0 is selected using the Graphics Mode register, the system microprocessor video memory read returns the 8-bit value that is determined by the logical decode of the memory address, and the Read Map Select register if applicable. When read type 1 is selected using the Graphics Mode register, the 8-bit value returned is the result of the color compare operation controlled by the Color Compare and Color Don't Care registers. Figure 3-28 on page 3-51 shows the data flow for the color compare operations.

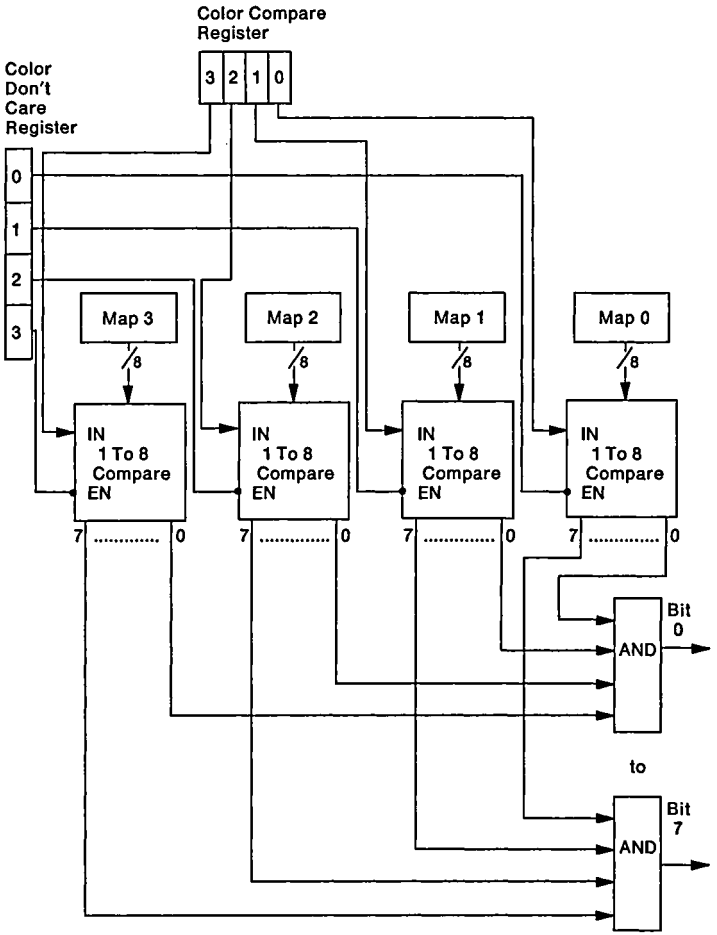


Figure 3-28. VGA Color Compare Operations

Write Operations

During system microprocessor video memory writes, the maps are enabled by the logical decode of the memory address and, depending on the video mode, of the Map Mask register. Figure 3-29 shows the data flow for a system microprocessor write operation.

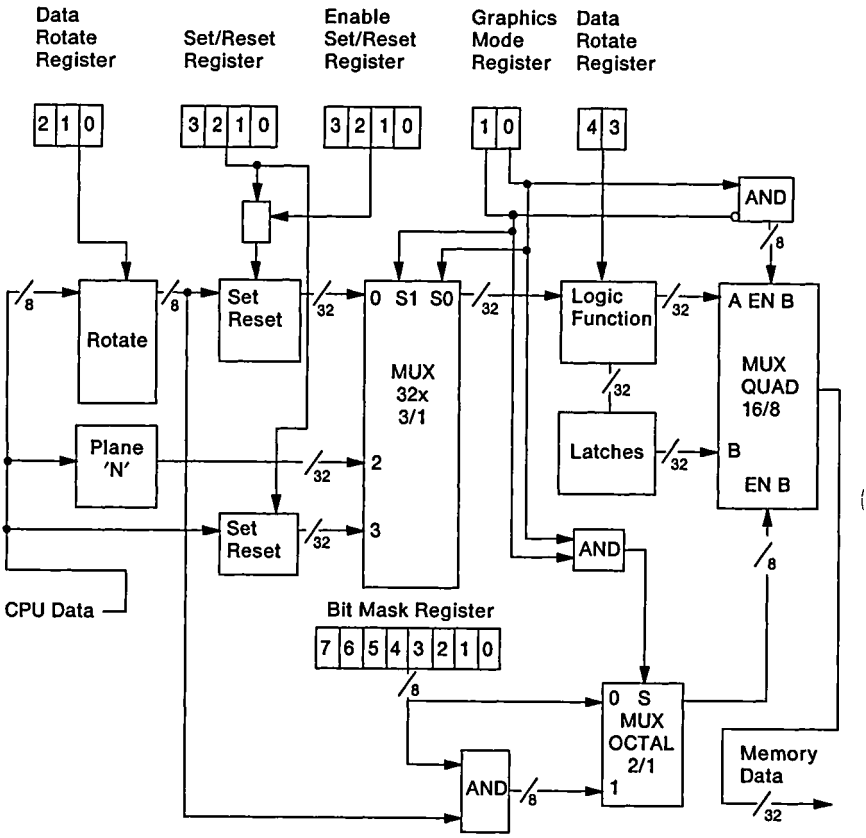


Figure 3-29. Data Flow for VGA Memory Write Operations

Note: Which maps are actually written with data is under control of the system microprocessor video memory address, which depends on the mode selected and the Map Mask register.

Registers

There are six sets of registers in the video subsystem as shown in Figure 3-30.

Figure 3-30. VGA Register Overview

	R/W	Monochrome Emulation	Color Emulation	Either
General Registers Addresses used 03BA or 03DA; 03C2; 03CA; and 03CC				
Miscellaneous Output Reg	W			03C2
	R			03CC
Input Status Register 0	RO			03C2
Input Status Register 1	RO	03BA	03DA	
Feature Control Register	W	03BA	03DA	
	R			03CA
VGA Enable Register	RW			03C3
Attribute Mode Control Registers Addresses used 03C0 - 03C1				
Address Register	RW			03C0
Other Attribute Registers	W			03C0
	R			03C1
CRT Controller Registers Addresses used 03D4 to 03D5 or 03B4 to 03B5				
Index Register	RW	03B4	03D4	
Other CRT Controller Regs.	RW	03B5	03D5	
Sequencer Registers Addresses used 03C4 to 03C5				
Address Register	RW			03C4
Other Sequencer Registers	RW			03C5
Graphics Controller Registers Addresses used 03CE to 03CF				
Address Register	RW			03CE
Other Graphics Registers	RW			03CF
Video DAC Registers Addresses used 03C6 to 03C9				
PEL Address Register	RW (Write Mode)			03C8
	WO (Read Mode)			03C7
DAC State Register	RO			03C7
PEL Data Register	RW			03C9
PEL Mask Register	RW			03C6

RO = Read-Only, RW = Read / Write, WO = Write-Only
Register addresses are in hex.

General Registers

This section contains descriptions of the following registers.

Figure 3-31. General Register Overview

Name	Read Port	Write Port	Index
Miscellaneous Output Register	03CC	03C2	-
Input Status Register 0	03C2	-	-
Input Status Register 1	03?A	-	-
Feature Control Register	03CA	03?A	-
VGA Enable Register	03C3	03C3	-

The (?) is controlled by bit 0 of the Miscellaneous Output register.

? = B in Monochrome Emulation Modes and

? = D in Color Emulation Modes

Register addresses are in hex.

Miscellaneous Output Register

This is a read/write register. A hardware reset causes all bits to be reset to 0. Read address = hex 03CC; write address = hex 03C2.

Figure 3-32. Miscellaneous Output Register

Bit	Function
7	Vertical Sync Polarity
6	Horizontal Sync Polarity
5	Page Bit For Odd/Even (Modes 0-5)
4	Reserved = 0
3	Clock Select 1
2	Clock Select 0
1	Enable RAM
0	I/O Address Select

Bit 7 When bit 7 is set to 1, negative vertical retrace is selected. When bit 7 is set to 0, positive vertical retrace is selected.

Bit 6 When bit 6 is set to 1, negative horizontal retrace is selected. When bit 6 is set to 0, positive horizontal retrace is selected.

Note: Bits 7 and 6 are selected based on the vertical size as shown in Figure 3-33.

Figure 3-33. Display, Vertical Size

Bit 7	Bit 6	Vertical Size
0	0	Reserved = 0
1	0	400 lines
0	1	350 lines
1	1	480 lines

Bit 5 Bit 5 selects between two 64K pages of memory when in the Odd/Even modes (0-5). When bit 5 is set to 1, the high page of memory is selected. When bit 5 is set to 0, the low page of memory is selected. This bit is provided for diagnostic use.

Bit 4 Reserved

Bits 3, 2 Bits 3 and 2 select the clock source according to Figure 3-34.

Figure 3-34. Clock Select 3 and 2 Bit Definitions

Bit 3	Bit 2	Function
0	0	Selects 25.175 MHz clock for 640 horizontal PELs
0	1	Selects 28.322 MHz clock for 720 horizontal PELs
1	0	Reserved
1	1	Reserved

Bit 1 When bit 1 is set to 1, Video RAM to the system microprocessor is enabled. When bit 1 is set to 0, Video RAM address decode from the system microprocessor is disabled.

Bit 0 This bit maps the CRT controller I/O addresses for IBM Monochrome or Color/Graphics Monitor Adapter emulation. When bit 0 is set to 1, the CRT controller address is set to hex 03DX and Input Status register 1 address is set to hex 03DA for IBM Color/Graphics Monitor Adapter emulation. When bit 0 is set to 0, the CRT controller address is set to hex 03BX and and Input Status register 1 address is set to hex 03BA for IBM Monochrome Adapter emulation.

Input Status Register 0

This is a read-only register. Read address = hex 03C2.

Figure 3-35. Input Status Register 0

Bit	Function
7	CRT Interrupt
6, 5	Reserved = 0
4	Switch Sense Bit
3 - 0	Reserved = 0

Bit 7 When bit 7 is set to 1, a vertical retrace interrupt is pending. When bit 7 is set to 0, the vertical retrace interrupt is cleared.

Bits 6, 5 Reserved

Bit 4 Bit 4 allows the system microprocessor to read the switch sense line and allows the power-on self-test to determine if a monochrome or color display is connected to the system.

Bits 3 - 0 Reserved

Input Status Register 1

This is a read-only register. Read address = hex 03?A.

Figure 3-36. Input Status Register 1

Bit	Function
7, 6	Reserved = 0
5	Diagnostic 0
4	Diagnostic 1
3	Vertical Retrace
2, 1	Reserved = 0
0	Display Enable

Bits 7, 6 Reserved.

Bits 5, 4 Diagnostic Usage - Bits 5 and 4 are selectively connected to two of the eight color outputs of the attribute controller. The Color Plane Enable register controls the multiplexer for the video wiring. Figure 3-37 shows the combinations available and the color output wiring.

Figure 3-37. Diagnostic Bits

Color Plane Enable Register		Input Status Register 1	
Bit 5	**Bit 4**	**Bit 5**	**Bit 4**
0	0	P2	P0
0	1	P5	P4
1	0	P3	P1
1	1	P7	P6

Bit 3 Vertical Retrace - When bit 3 is set to 1, a vertical retrace interval is occurring. When bit 3 is set to 0, video information is being displayed. Bit 3 can be programmed, through bits 5 and 4 of the Vertical Retrace End register, to interrupt the system microprocessor on interrupt level 2 at the start of the vertical retrace.

Bit 2, 1 Reserved.

Bit 0 Display Enable - When bit 0 is set to 1, a horizontal or vertical retrace interval is occurring. This bit is the real-time status of the inverted display enable signal. To avoid glitches on the display, some programs use this status bit to restrict screen updates to inactive display intervals. The video subsystem has been designed to eliminate this software requirement, so display screen updates may be made at any time.

Feature Control Register

This is a read/write register. Read address = hex 03CA; write address = hex 03?A.

All bits in this register are reserved and bit 3 must be 0.

VGA Enable Register

Figure 3-38. VGA Enable Register, Hex 03C3

Bit	Function
7 - 1	Reserved = 0
0	VGA Enable

Bits 7 - 1 Reserved

Bit 0 When bit 0 is set to 1, the video I/O and memory address decoding is enabled. When bit 0 is set to 0, the video I/O and memory address decoding is disabled.

Note: Accesses to the Video Subsystem Enable register are not affected by the VGA sleep bit (I/O port 102, bit 0), described in "Video Subsystem Programmable Option Select" on page 3-24.

Sequencer Registers

This section contains descriptions of the following registers.

Figure 3-39. Sequencer Register Overview

Name	Port (hex)	Index (hex)
Sequencer Address	03C4	-
Reset	03C5	00
Clocking Mode	03C5	01
Map Mask	03C5	02
Character Map Select	03C5	03
Memory Mode	03C5	04

Sequencer Address Register

The Sequencer Address register is a pointer register located at address hex 03C4. This register is loaded with a binary value that points to the Sequencer Data register where data is to be written. This value is referred to as *Index* in Figure 3-39.

Figure 3-40. Sequencer Address Register

Bit	Function
7 - 3	Reserved = 0
2 - 0	Sequencer Address Bits

Bits 7 - 3 Reserved

Bits 2 - 0 Bits 2 through 0 contain the binary value pointing to the register where data is to be written.

Reset Register

This is a read/write register pointed to when the value in the Sequencer Address register is hex 00. The port address for this register is hex 03C5.

Figure 3-41. Reset Register, Index Hex 00

Bit	Function
7 - 2	Reserved = 0
1	Synchronous Reset
0	Asynchronous Reset

Bits 7 - 2 Reserved.

Bit 1 Synchronous Reset - When bits 0 and 1 are both 1, the sequencer operates. When bit 1 is set to 0, the sequencer is synchronously cleared and stopped. Bit 1 must be set to 0 before changing either bit 3 or bit 0 of the Clocking Mode register (index 01), or bits 2 or 3 of the Miscellaneous Output register (hex 03C2).

Bit 0 Asynchronous Reset - When bits 0 and 1 are both 1, the sequencer operates. When bit 0 is set to 0, the sequencer is asynchronously cleared and stopped. Resetting the sequencer with this bit can cause data loss in the dynamic RAMs.

Clocking Mode Register

This is a read/write register pointed to when the value in the Sequencer Address register is hex 01. The port address for this register is hex 03C5.

Figure 3-42. Clocking Mode Register, Index Hex 01

Bit	Function
7, 6	Reserved = 0
5	Screen Off
4	Shift 4
3	Dot Clock
2	Shift Load
1	Reserved = 0
0	8/9 Dot Clocks

Bits 7, 6 Reserved.

Bit 5 Screen Off - When bit 5 is set to 1, the video screen is turned off and maximum memory bandwidth is assigned to the system microprocessor. When bit 5 is set to 0, normal screen operation is selected. The screen is blanked when this bit is set, and the sync pulses are maintained. Use this bit for rapid full-screen updates.

Bit 4 Shift 4 - When bit 4 is set to 1, the serializers are loaded every fourth character clock. This mode is useful when 32 bits are fetched per cycle and chained together in the shift registers. When bit 4 is set to 0, the video serializers are loaded every character clock.

Bit 3 Dot Clock - When bit 3 is set to 1, the master clock is divided by 2 to generate the dot clock. Dot clock divided by 2 is used for 320- and 360-horizontal-PEL modes. When bit 3 is set to 0, the dot clock is derived from the sequencer master clock input (normal). All the other timings are affected because they are derived from the dot clock.

Bit 2 Shift Load - When bit 2 is set to 1, the video serializers are reloaded every other character clock. This mode is useful when 16 bits are fetched per cycle and chained together in the shift registers. When bit 2 is set to 0, and bit 4 is set to 0, the video serializers are reloaded every character clock.

Bit 1 Reserved.

Bit 0 8/9 Dot Clocks - When bit 0 is set to 1, the sequencer generates character clocks 8 dots wide. When bit 0 is set to 0, the sequencer generates character clocks 9 dots wide. Alphanumeric modes hex 0+, 1+, 2+, 3+, 7 and 7+ are the only modes that use character clocks 9 dots wide. All other modes must use 8 dots per character clock. The 9-dot mode is for alphanumeric modes only. The ninth dot equals the eighth dot for ASCII codes hex C0 through DF. (See the enable line graphics character codes bit in "Attribute Mode Control Register" on page 3-89.)

Map Mask Register

This is a read/write register pointed to when the value in the Sequencer Address register is hex 02. The port address for this register is hex 03C5.

Figure 3-43. Map Mask Register, Index Hex 02

Bit	Function
7 - 4	Reserved = 0
3	Map 3 Enable
2	Map 2 Enable
1	Map 1 Enable
0	Map 0 Enable

A logical 1 in any of bits 3 through 0 enables the system microprocessor to write to the corresponding map. If this register is programmed with a value of 0FH, the system microprocessor can perform a 32-bit write operation with only one memory cycle. This substantially reduces the overhead on the system microprocessor during display update cycles in graphics modes. Data scrolling operations are also enhanced by setting this register to a value of 0FH and writing the display buffer address with the data stored in the system microprocessor data latches. This is a read-modify-write operation. When odd/even modes are selected, maps 0 and 1 and maps 2 and 3 should have the same map mask value. When chain 4 mode is selected, all maps should be enabled.

Character Map Select Register

This is a read/write register pointed to when the value in the Sequencer Address register is hex 03. The port address for this register is hex 03C5.

Figure 3-44. Character Map Select Register, Index Hex 03

Bit	Function
7, 6	Reserved = 0
5	Character Map Select High Bit A
4	Character Map Select High Bit B
3, 2	Character Map Select A
1, 0	Character Map Select B

Bits 3 and 2 select the portion of map 2 used to generate alphanumeric characters when attribute bit 3 is 1, according to Figure 3-45.

Figure 3-45. Character Map Select A

Bit 5 Value	Bit 3 Value	Bit 2 Value	Map Selected	Table Location
0	0	0	0	1st 8KB of Map 2
0	0	1	1	3rd 8KB of Map 2
0	1	0	2	5th 8KB of Map 2
0	1	1	3	7th 8KB of Map 2
1	0	0	4	2nd 8KB of Map 2
1	0	1	5	4th 8KB of Map 2
1	1	0	6	6th 8KB of Map 2
1	1	1	7	8th 8KB of Map 2

In alphanumeric modes, bit 3 of the attribute byte normally has the function of turning the foreground intensity on or off. This bit can be

redefined as a switch between character sets. For this feature to be enabled, the following must be true:

- Memory Mode register (index 04) bit 1 must be equal to 1
- The value of Character Map Select A does not equal the value of Character Map Select B.

If either condition is not met, the first 16KB of map 2 is used.

Bits 1 and 0 select the portion of map 2 used to generate alpha characters when attribute bit 3 is 0, according to Figure 3-46.

Figure 3-46. Character Map Select B

Bit 4 Value	Bit 1 Value	Bit 0 Value	Map Selected	Table Location
0	0	0	0	1st 8KB of Map 2
0	0	1	1	3rd 8KB of Map 2
0	1	0	2	5th 8KB of Map 2
0	1	1	3	7th 8KB of Map 2
1	0	0	4	2nd 8KB of Map 2
1	0	1	5	4th 8KB of Map 2
1	1	0	6	6th 8KB of Map 2
1	1	1	7	8th 8KB of Map 2

Memory Mode Register

This is a read/write register pointed to when the value in the Sequence Address register is hex 04. The output port address for this register is 03C5.

Figure 3-47. Memory Mode Register, Index Hex 04

Bit	Function
7 - 4	Reserved = 0
3	Chain 4
2	Odd/Even
1	Extended Memory
0	Reserved = 0

Bits 7 - 4 Reserved.

Bit 3 Chain 4 - When bit 3 is set to 1, the 2 low-order bits select the map that will be accessed as shown in Figure 3-48 on page 3-63. When bit 3 is set to 0, the system microprocessor sequentially addresses access data within a bit map using the Map Mask register.

Note: This bit controls the map selected in the graphics subsection during system microprocessor reads.

Figure 3-48. Memory Mode, Chain 4

A1	A0	Map Selected
0	0	0
0	1	1
1	0	2
1	1	3

Bit 2 Odd/Even - When bit 2 is set to 1, the system microprocessor sequentially accesses data within a bit map. When bit 2 is set to 0, even addresses access maps 0 and 2 and odd addresses access maps 1 and 3. The maps are accessed according to the value in the Map Mask register (index 02).

Bit 1 Extended Memory - When bit 1 is set to 1, more than 64KB of video memory is present. This bit must be set to allow the VGA to use the 256KB of video memory on the system board, and to enable the character map selection on the previous page.

Bit 0 Reserved.

CRT Controller Registers

This section contains descriptions of the following registers:

Figure 3-49. CRT Controller Register Overview

Name	Port (hex)	Index (hex)
CRT Controller Address	03?4	-
Horizontal Total	03?5	00
Horizontal Display Enable End	03?5	01
Start Horizontal Blanking	03?5	02
End Horizontal Blanking	03?5	03
Start Horizontal Retrace Pulse	03?5	04
End Horizontal Retrace	03?5	05
Vertical Total	03?5	06
Overflow	03?5	07
Preset Row Scan	03?5	08
Maximum Scan Line	03?5	09
Cursor Start	03?5	0A
Cursor End	03?5	0B
Start Address High	03?5	0C
Start Address Low	03?5	0D
Cursor Location High	03?5	0E
Cursor Location Low	03?5	0F
Vertical Retrace Start	03?5	10
Vertical Retrace End	03?5	11
Vertical Display Enable End	03?5	12
Offset	03?5	13
Underline Location	03?5	14
Start Vertical Blank	03?5	15
End Vertical Blank	03?5	16
CRTC Mode Control	03?5	17
Line Compare	03?5	18

? = B in Monochrome Emulation Modes
? = D in Color Emulation Modes.
This is controlled by bit 0 of the Miscellaneous Output register.

CRT Controller Address Register

The CRT Controller Address register is a pointer register located at hex 03B4 or hex 03D4. Which address is used depends on bit 0 of the Miscellaneous Output register at address hex 03C2. The CRT Controller Address register is loaded with a binary value that points to the CRT Controller Data register where data is to be written. This value is referred to as *Index* in Figure 3-49 on page 3-64. All CRT controller registers are read/write registers.

Figure 3-50. CRT Controller Address Register

Bit	Function
7	Reserved = 0
6	Reserved = 0
5	Reserved = 0
4	CRTC Index
3	CRTC Index
2	CRTC Index
1	CRTC Index
0	CRTC Index

Bits 7, 6 Reserved.

Bit 5 Bit 5 is used for chip testing and must be 0.

Bits 4 - 0 Bits 4 through 0 contain the index value of the CRT Controller Data register where data is to be written.

Horizontal Total Register

This is a read/write register pointed to when the value in the CRT Controller Address register is hex 00. The port address for this register is hex 03?5.

This register defines the total number of characters (minus 5) in the horizontal scan interval including the retrace time. The value directly controls the period of the horizontal retrace output signal. An internal horizontal character counter counts character clock inputs to the CRT controller, and all horizontal and vertical timings are based upon the Horizontal Total register. Comparators are used to compare register values with horizontal character values to provide horizontal timings.

Horizontal Display Enable End Register

This is a read/write register pointed to when the value in the CRT Controller Address register is hex 01. The port address for this register is hex 03?5.

This register defines the length of the horizontal display enable signal. It determines the number of displayed character positions per horizontal line. The value in this register is the total number of displayed characters less 1.

Start Horizontal Blanking Register

This is a read/write register pointed to when the value in the CRT Controller Address register is hex 02. The port address for this register is hex 03?5.

The horizontal blanking signal becomes active when the horizontal character counter reaches the value in this register.

End Horizontal Blanking Register

This is a read/write register pointed to when the value in the CRT Controller Address register is hex 03. The port address for this register is hex 03?5.

Figure 3-51. End Horizontal Blanking Register, Index Hex 03

Bit	Function
7	Reserved = 1
6	Display Enable Skew Control
5	Display Enable Skew Control
4 - 0	End Horizontal Blanking

This register determines when the horizontal blanking output signal becomes inactive.

Bit 7 Bit 7 is used for chip testing and must be set to 1.

Bits 6, 5 Display Enable Skew Control - Bits 6 and 5 determine the amount of display enable skew. Display enable skew control is required to provide sufficient time for the CRT controller to access the display buffer to obtain a character and attribute code, access the character generator font, and then go through the Horizontal PEL Panning register in the attribute controller. Each access requires the display enable signal to be skewed one character clock unit so that the video output is in

synchronization with the horizontal retrace and vertical retrace signals. The bit values and amount of skew are shown in Figure 3-52 on page 3-67.

Figure 3-52. Bit Values and Amount of Skew

Bit 6	Bit 5	Skew
0	0	Zero-character Clock Skew
0	1	One-character Clock Skew
1	0	Two-character Clock Skew
1	1	Three-character Clock Skew

Bits 4 - 0 End Horizontal Blanking - A value equal to the 5 least-significant bits of the horizontal character counter value at which time the horizontal blanking signal becomes inactive (logical 0). To obtain a blanking signal of width W, the following algorithm is used: Value of Start Blanking register + width of blanking signal in character clock units = 6-bit result to be programmed into the End Horizontal Blanking register. Bit 5 is located in the End Horizontal Retrace register (index hex 05).

Start Horizontal Retrace Pulse Register

This is a read/write register pointed to when the value in the CRT Controller Address register is hex 04. The port address for this register is hex 03?5.

This register is used to center the screen horizontally, and to specify the character position at which the horizontal retrace signal becomes active. The value programmed is a binary count of the character position at which the signal becomes active.

End Horizontal Retrace Register

This is a read/write register pointed to when the value in the CRT Controller Address register is hex 05. The port address for this register is hex 03?5.

Figure 3-53. End Horizontal Retrace Register, Index Hex 05

Bit	Function
7	End Horizontal Blanking, Bit 5
6, 5	Horizontal Retrace Delay
4 - 0	End Horizontal Retrace

This register specifies the character position at which the horizontal retrace pulse becomes inactive (logical 0).

Bit 7 End Horizontal Blanking, MSB - The first 5 bits are located in the End Horizontal Blanking register (index hex 03).

Bits 6, 5 Bits 6 and 5 control the skew of the horizontal retrace signal. Binary 00 equals no horizontal retrace delay. For some modes, it is necessary to provide a horizontal retrace signal that takes up the entire blanking interval. Some internal timings are generated by the falling edge of the horizontal retrace signal. To guarantee the signals are latched properly, the retrace signal is started before the end of the display enable signal, and then skewed several character clock times to provide proper screen centering.

Bits 4 - 0 A value equal to the 5 least-significant bits of the horizontal character counter value at which the horizontal retrace signal becomes inactive (logical 0). To obtain a retrace signal of width W, the following algorithm is used: Value of Start Retrace register + width of the horizontal retrace signal in character clock units = 5-bit result to be programmed into the End Horizontal Retrace register.

Vertical Total Register

This is a read/write register pointed to when the value in the CRT Controller Address register is hex 06. The port address for this register is hex 03?5.

This register contains the low-order 8 bits of a 10-bit register. The binary value represents the number of horizontal raster scans on the CRT minus 2, including vertical retrace. The value in this register determines the period of the vertical retrace signal.

Bit 8 of this register is contained in the CRT Controller Overflow register, index hex 07 bit 0.

Bit 9 of this register is contained in the CRT Controller Overflow register, index hex 07 bit 5.

CRT Controller Overflow Register

This is a read/write register pointed to when the value in the CRT Controller Address register is hex 07. The port address for this register is hex 03?5.

Figure 3-54. CRT Controller Overflow Register, Index Hex 07

Bit	Function
7	Bit 9 of the Vertical Retrace Start (Index hex 10)
6	Bit 9 of the Vertical Display Enable End (Index hex 12)
5	Bit 9 of the Vertical Total (Index hex 6)
4	Bit 8 of the Line Compare (Index hex 18)
3	Bit 8 of the Start Vertical Blanking (Index hex 15)
2	Bit 8 of the Vertical Retrace Start (Index hex 10)
1	Bit 8 of the Vertical Display Enable End (Index hex 12)
0	Bit 8 of the Vertical Total (Index hex 6)

Preset Row Scan Register

This is a read/write register pointed to when the value in the CRT Controller Address register is hex 08. The port address for this register is hex 03?5.

Figure 3-55. Preset Row Scan Register, Index Hex 08

Bit	Function
7	Reserved = 0
6	Byte Panning Control
5	Byte Panning Control
4 - 0	Starting Row Scan Count after a Vertical Retrace

This register is used for PEL scrolling.

Bit 7 Reserved.

Bits 6, 5 Bits 6 and 5 control byte panning in modes programmed as multiple shift modes. (Currently, no modes are programmed for multiple shift operation.) This is required for PEL panning operations. The PEL Panning register in the attribute section provides panning of up to 7 or 8 individual PELs. In single-byte shift modes, to pan to the next higher PEL (8 or 9), the CRT controller start address is incremented and attribute panning is reset to 0. In multiple shift modes, the byte pan bits are used as extensions to the attribute PEL Panning register. This allows panning across the width of the video output shift. For example, in the 32-bit shift mode, the byte pan and PEL panning bits provide up to 31 bits of panning capability. To pan from position 31 to 32, the CRT controller start address is incremented and PEL and byte panning is reset to 0. These bits should normally be set to 0.

Bits 4 - 0 Bits 4 through 0 specify the starting row scan count after a vertical retrace. The row scan counter increments each horizontal retrace time until a maximum row scan occurs. At maximum row scan compare time, the row scan is cleared (not preset).

Maximum Scan Line Register

This is a read/write register pointed to when the value in the CRT Controller Address register is hex 09. The port address for this register is hex 03?5.

Figure 3-56. Maximum Scan Line Register, Index Hex 09

Bit	Function
7	200 --> 400 Line Conversion
6	Bit 9 of the Line Compare (Index hex 18)
5	Bit 9 of the Start Vertical Blanking (Index hex 15)
4 - 0	Maximum Scan Line

Bit 7 When bit 7 is set to 1, 200 to 400 line conversion is performed. The clock in the row scan counter is divided by 2. This allows the older 200-line modes to be displayed as 400 lines on the display (this is double scanning, in which each line is displayed twice). When bit 7 is set to 0, the clock to the row scan counter is equal to the horizontal scan rate. Double scan is not enabled.

Bit 6 Bit 9 of the Line Compare register (index hex 18).

Bit 5 Bit 9 of the Start Vertical Blank register (index hex 15).

Bits 4 - 0 Bits 4 through 0 specify the number of scan lines per character row. The number programmed is the maximum row scan number minus 1.

Cursor Start Register

This is a read/write register pointed to when the value in the CRT Controller Address register is hex 0A. The port address for this register is hex 03?5.

Figure 3-57. Cursor Start Register, Index Hex 0A

Bit	Function
7, 6	Reserved = 0
5	Cursor Off
4 - 0	Row Scan Cursor begins

Bits 7, 6 Reserved.

Bit 5 When bit 5 is set to 1, the cursor is turned off. When bit 5 is set to 0, the cursor is turned on.

Bits 4 - 0 Bits 4 through 0 specify the row scan of a character line where the cursor is to begin. The number programmed is the starting cursor row scan minus 1.

When the Cursor Start register is programmed with a value greater than the Cursor End register, no cursor is generated.

Cursor End Register

This is a read/write register pointed to when the value in the CRT Controller Address register is hex 0B. The port address for this register is hex 03?5.

Figure 3-58. Cursor End Register, Index Hex 0B

Bit	Function
7	Reserved = 0
6, 5	Cursor Skew Control
4 - 0	Row Scan Cursor ends

Bit 7 Reserved.

Bits 6, 5 Bits 6 and 5 control the skew of the cursor signal. Cursor skew delays the cursor by the selected number of clocks. For example, a skew of 1 moves the cursor right one position on the screen.

Figure 3-59. Cursor Skew

Bit 6	Bit 5	Function
0	0	Zero-character Clock Skew
0	1	One-character Clock Skew
1	0	Two-character Clock Skew
1	1	Three-character Clock Skew

Bits 4 - 0 Cursor End - Bits 4 through 0 specify the row scan of a character line where the cursor is to end.

Start Address High Register

This is a read/write register pointed to when the value in the CRT Controller Address register is hex 0C. The port address for this register is hex 03?5.

This register contains the most-significant 8 bits of the start address. The 16-bit value, from the Start Address High and Low registers, is the first address after the vertical retrace on each screen refresh.

Start Address Low Register

This is a read/write register pointed to when the value in the CRT Controller Address register is hex 0D. The port address for this register is hex 03?5.

This register contains the least-significant 8 bits of the start address.

Cursor Location High Register

This is a read/write register pointed to when the value in the CRT Controller Address register is hex 0E. The port address for this register is hex 03?5.

This register contains the most-significant 8 bits of the cursor location.

Cursor Location Low Register

This is a read/write register pointed to when the value in the CRT Controller Address register is hex 0F. The port address for this register is hex 03?5.

This register contains the least-significant 8 bits of the cursor location.

Vertical Retrace Start Register

This is a read/write register pointed to when the value in the CRT Controller Address register is hex 10. The port address for this register is hex 03?5.

This register contains the least-significant 8 bits of the vertical retrace signal start position, programmed in horizontal scan lines. Bits 8 and 9 are in the CRT Controller Overflow register (index hex 07).

Vertical Retrace End Register

This is a read/write register pointed to when the value in the CRT Controller Address register is hex 11. The port address for this register is hex 03?5.

Figure 3-60. Vertical Retrace End Register, Index Hex 11

Bit	Function
7	Protect R0-7
6	Select 5 Refresh Cycles
5	Enable Vertical Interrupt=0
4	Clear Vertical Interrupt=0
3 - 0	Vertical Retrace End

Bit 7 When bit 7 is set to 1, writing to CRT controller registers 0 through 7 is disabled. When bit 7 is set to 0, writing to CRT controller registers is enabled. Bit 4 in CRT Controller Overflow register, index hex 07, is not protected.

Bit 6 When bit 6 is set to 1, five DRAM refresh cycles are generated per horizontal line. Selecting five refresh cycles allows use of the VGA chip with slow sweep rate displays (15.75 KHz). When bit 6 is set to 0, three refresh cycles are selected. This bit is set to 0 by BIOS during a mode set, a reset, or power-on.

Bit 5 When bit 5 is set to 0, a vertical retrace interrupt is enabled. The vertical retrace interrupt is on IRQ2. This interrupt level can be shared so the Input Status register 0, bit 7, should be checked to find out if the VGA generated the interrupt. As with bit 4, do not change the value of the other bits in this register.

Bit 4 When bit 4 is set to 0, vertical retrace interrupt is cleared. At the end of the active vertical display time, a flip-flop is set in the VGA for vertical interrupt. The output of this flip-flop goes to the system board interrupt controller. An interrupt handler has to reset this flip-flop by writing a 0 to this bit, then setting the bit to 1 so that the flip-flop does not hold interrupts inactive. *Do not change* the other bits in this register. The register is readable, so a read can be done to determine what the other bit settings are before the flip-flop is reset.

Bits 3 - 0 Bits 3 through 0 determine the horizontal scan count value when the vertical retrace signal becomes inactive. The register is programmed in units of horizontal scan lines. To obtain a vertical retrace signal of width W, the following algorithm is used: Value of Start Vertical Retrace register + width of the vertical retrace signal in horizontal scan units = 4-bit result to be programmed into the End Horizontal Retrace register.

Vertical Display Enable End Register

This is a read/write register pointed to when the value in the CRT Controller Address register is hex 12. The port address for this register is hex 03?5.

This register contains the least-significant 8 bits of a 10-bit value that defines the vertical display enable end position.

Bit 8 of this value is contained in the CRT Controller Overflow register, index hex 07, bit 1.

Bit 9 of this value is contained in the CRT Controller Overflow register, index hex 07, bit 6.

This value specifies which scan line ends the active video area of the screen. It is programmed with the total number of lines minus 1.

Offset Register

This is a read/write register pointed to when the value in the CRT Controller Address register is hex 13. The port address for this register is hex 03?5.

This register specifies the logical line width of the screen. The starting memory address for the next character row is larger than the current character row by two or four times this amount. The Offset register is programmed with a word address. Depending on the method of clocking the CRT controller, this word address is either a word or doubleword address.

Underline Location Register

This is a read/write register pointed to when the value in the CRT Controller Address register is hex 14. The port address for this register is hex 03?5.

Figure 3-61. Underline Location Register, Index Hex 14

Bit	Function
7	Reserved = 0
6	Doubleword Mode
5	Count by 4
4 - 0	Horizontal Row Scan where Underline will occur

Bit 7 Reserved.

Bit 6 When bit 6 is set to 1, memory addresses are doubleword addresses. (See "CRTC Mode Control Register" on page 3-76.)

Bit 5 When bit 5 is 1, the memory address counter is clocked with the character clock divided by 4. This bit is used when doubleword addresses are used.

Bits 4 - 0 Bits 4 through 0 specify the horizontal row scan of a character row on which an underline occurs. The value programmed is the scan line number desired minus 1.

Start Vertical Blanking Register

This is a read/write register pointed to when the value in the CRT Controller Address register is hex 15. The port address for this register is hex 03?5.

This register contains the least-significant bits of a 10-bit value. Bit 8 is in the CRTC Overflow register (index hex 07). Bit 9 is in the Maximum Scan Line register (index hex 09).

The value in these 10 bits is one less than the horizontal scan line count at which the vertical blanking signal becomes active.

End Vertical Blanking Register

This is a read/write register pointed to when the value in the CRT Controller Address register is hex 16. The port address for this register is hex 03?5.

This register specifies the horizontal scan count value when the vertical blanking signal becomes inactive. The register is programmed in units of horizontal scan lines.

To obtain a vertical blanking signal of width W, the following algorithm is used: (Value of Start Vertical Blanking register minus 1) + width of the vertical blanking signal in horizontal scan units = 8-bit result to be programmed into the End Vertical Blanking register.

CRTC Mode Control Register

This is a read/write register pointed to when the value in the CRTC Address register is hex 17. The port address for this register is hex 03?5.

Figure 3-62. CRTC Mode Control Register, Index Hex 17

Bit	Function
7	Hardware Reset
6	Word/Byte Mode
5	Address Wrap
4	Reserved = 0
3	Count By 2
2	Horizontal Retrace Select
1	Select Row Scan Counter
0	Compatibility Mode Support 0

Bit 7 When bit 7 is set to 1, horizontal and vertical retrace are enabled. When bit 7 is set to 0, horizontal and vertical retrace are cleared. This bit does not reset any other registers or outputs.

Bit 6 When bit 6 is set to 1, Byte Address mode is selected. When bit 6 is set to 0, the Word Address mode shifts all memory address counter bits down 1 bit, and the most-significant bit of the counter appears on the least-significant bit of the memory address outputs.

Bit 6 of the End Vertical Blanking register in the CRT controller also controls the addressing. When it is 0, bit 6 above has control. When it is 1, the addressing is shifted by 2 bits. (See Figure 3-63.)

Figure 3-63. Internal Memory Address Counter Wiring to the Output Multiplexer

Memory Address	Byte Address Mode	Word Address Mode	Doubleword Address Mode
MA 0/RFA 0	MA 0	MA 15 or MA 13	MA 12
MA 1/RFA 1	MA 1	MA 0	MA 13
MA 2/RFA 2	MA 2	MA 1	MA 0
MA 3/RFA 3	MA 3	MA 2	MA 1
MA 4/RFA 4	MA 4	MA 3	MA 2
MA 5/RFA 5	MA 5	MA 4	MA 3
MA 6/RFA 6	MA 6	MA 5	MA 4
MA 7/RFA 7	MA 7	MA 6	MA 5
MA 8/RFA 8	MA 8	MA 7	MA 6
MA 9	MA 9	MA 8	MA 7
MA 10	MA 10	MA 9	MA 8
MA 11	MA 11	MA 10	MA 9
MA 12	MA 12	MA 11	MA 10
MA 13	MA 13	MA 12	MA 11
MA 14	MA 14	MA 13	MA 12
MA 15	MA 15	MA 14	MA 13

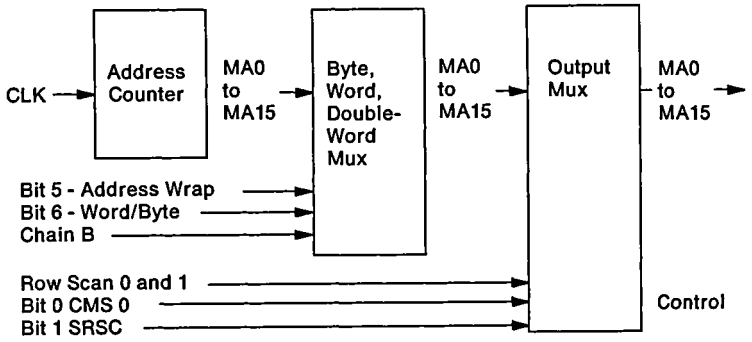


Figure 3-64. CRT Controller Memory Address Mapping

Bit 5 Address Wrap - Bit 5 selects memory address counter bit MA 13 or bit MA 15, and it appears on the MA 0 output pin in the Word Address mode. If the VGA is not in the Word Address mode, MA 0 counter output appears on the MA 0 output pin. When bit 5 is set to 1, MA 15 is selected. In odd/even mode, bit MA 15 should be selected since 256KB of video memory is installed on the system board. (Bit MA 13 is selected in applications where only 64KB memory is present. This function implements IBM Color/Graphics Monitor Adapter compatibility.)

Bit 4 Reserved.

Bit 3 Count By 2 - When bit 3 is set to 1, the memory address counter is clocked with the character clock input divided by 2. When bit 3 is set to 0, the memory address counter is clocked with the character clock input. This bit is used to create either a byte or word refresh address for the display buffer.

Bit 2 Horizontal Retrace Select - Bit 2 selects horizontal retrace or horizontal retrace divided by 2 as the clock that controls the vertical timing counter. This bit can be used to effectively double the vertical resolution capability of the CRT controller. The vertical counter has a maximum resolution of 1024 scan lines due to the 10-bit wide Vertical Total register. If the vertical counter is clocked with the horizontal retrace divided by 2, then the vertical resolution is doubled to 2048 horizontal scan lines. When bit 2 is set to 1, HRTC divided by 2 is selected. When bit 2 is set to 0, HRTC is selected.

Bit 1 Select Row Scan Counter - When bit 1 is set to 1, the MA 14 counter bit on the MA 14 output pin is selected. When bit 1 is set to 0, row scan counter bit 1 on the MA 14 output pin is selected.

Bit 0 Compatibility Mode Support - When bit 0 is set to 1, the MA 13 counter bit appears on the MA 13 output pin of the CRT controller. When bit 0 is set to 0, row scan address bit 0 is substituted for MA 13 during active display time. The CRT controller used on the IBM Color/Graphics Monitor Adapter is the 6845. The 6845 has 128 horizontal scan line address capability. To obtain 640 by 200 graphics resolution, the CRT controller is programmed for 100 horizontal scan lines with two row scan addresses per character row. Row scan address bit 0 becomes the most-significant address bit to the display buffer. Successive scan lines of the display image are displaced in memory by 8KB. This bit allows compatibility with the 6845 and color graphics APA modes of operation.

Line Compare Register

This is a read/write register pointed to when the value in the CRT Controller Address register is hex 18. The port address for this register is hex 03?5.

This register contains the least-significant 8 bits of the compare target. When the vertical counter reaches this value, the internal start of the line counter is cleared. As a result, an area of the screen is not affected by scrolling. Bit 8 of this value is in the CRT Controller Overflow register, index hex 07. Bit 9 is in the Maximum Scan Line register, index hex 09.

Graphics Controller Registers

This section contains descriptions of the following registers.

Figure 3-65. Graphics Controller Register Overview

Name	Port (hex)	Index (hex)
Graphics Address	03CE	-
Set/Reset	03CF	00
Enable Set/Reset	03CF	01
Color Compare	03CF	02
Data Rotate	03CF	03
Read Map Select	03CF	04
Graphics Mode	03CF	05
Miscellaneous	03CF	06
Color Don't Care	03CF	07
Bit Mask	03CF	08

Graphics Address Register

This is a read/write register and the port address for this register is hex 03CE.

Figure 3-66. Graphics Address Register

Bit	Function
7 - 4	Reserved = 0
3 - 0	Graphics Address

Bits 7 - 4 Reserved.

Bits 3 - 0 Bits 3 through 0 point to the other registers in the graphics section.

Set/Reset Register

This is a read/write register pointed to by the value in the Graphics Address register. This value must be hex 00 before writing can occur. The port address for this register is hex 03CF.

Figure 3-67. Set/Reset Register, Index Hex 00

Bit	Function
7 - 4	Reserved = 0
3	Set/Reset Map 3
2	Set/Reset Map 2
1	Set/Reset Map 1
0	Set/Reset Map 0

Bits 7 - 4 Reserved.

Bits 3 - 0 Bits 3 through 0 represent the value written to all 8 bits of the respective memory map when the system microprocessor does a memory write with write mode 0 selected and Set/Reset mode enabled. Set/Reset can be enabled on a map-by-map basis with separate OUT instructions to the Enable Set/Reset register, index hex 01.

Enable Set/Reset Register

This is a read/write register and is pointed to by the value in the Graphics Address register. This value must be hex 01 before writing can occur. The port address for this register is hex 03CF.

Figure 3-68. Enable Set/Reset Register, Index Hex 01

Bit	Function
7 - 4	Reserved = 0
3	Enable Set/Reset Map 3
2	Enable Set/Reset Map 2
1	Enable Set/Reset Map 1
0	Enable Set/Reset Map 0

Bits 7 - 4 Reserved.

Bits 3 - 0 Bits 3 through 0 enable the set/reset function. When enabled (bit = 1) and write mode is 0, the respective memory map is written with the value of the Set/Reset register. When write mode is 0 and Set/Reset is not enabled (bit = 0) on a map, that map is written with the value of the system microprocessor data.

Color Compare Register

This is a read/write register pointed to by the value in the Graphics Address register. This value must be hex 02 before writing can occur. The port address for this register is hex 03CF.

Figure 3-69. Color Compare Register, Index Hex 02

Bit	Function
7 - 4	Reserved = 0
3	Color Compare Map 3
2	Color Compare Map 2
1	Color Compare Map 1
0	Color Compare Map 0

Bits 7 - 4 Reserved.

Bits 3 - 0 Bits 3 through 0 represent a 4-bit color value to be compared. If the system microprocessor sets Read Type bit in the Graphics Mode register, index hex 05 and does a memory read, the data returned from the memory cycle will be a 1 in each bit position where the four maps equal the Color Compare register.

The color compare bit is the value that all bits of the corresponding map's byte are compared with. Each of the 8 bit positions of the selected byte are then compared across the four maps, and a 1 is returned in each bit position where the bits of all four maps equal their respective color compare values.

Data Rotate Register

This is a read/write register pointed to by the value in the Graphics Address register. This value must be hex 03 before writing can occur. The port address for this register is hex 03CF.

Figure 3-70. Data Rotate Register, Index Hex 03

Bit	Function
7 - 5	Reserved = 0
4	Function Select
3	Function Select
2	Rotate Count 2
1	Rotate Count 1
0	Rotate Count 0

Bits 7 - 5 Reserved.

Bits 4, 3 Data written to memory can operate logically with data already in the system microprocessor latches.

Data can be any of the choices selected by the Write Mode bits in the Graphics Mode register, index hex 05, except system microprocessor latches, which cannot be modified. If rotated data is selected, the rotate applies before the logical function. The bit functions are defined in Figure 3-71.

Figure 3-71. Function Select Bit Definitions

Bit 4	Bit 3	Function
0	0	Data unmodified.
0	1	Data ANDed with latched data.
1	0	Data ORed with latched data.
1	1	Data XORed with latched data.

Bits 2 - 0 Rotate Count - Bits 2 through 0 represent a binary encoded value of the number of positions to right-rotate the system microprocessor data bus during system microprocessor memory writes. This operation is done when the write mode is 0. To write non-rotated data, the system microprocessor must select a count of 0.

Read Map Select Register

This is a read/write register pointed to by the value in the Graphics Address register. This value must be hex 04 before writing can occur. The port address for this register is hex 03CF.

Figure 3-72. Read Map Select Register, Index Hex 04

Bit	Function
7 - 2	Reserved = 0
1	Map Select 1
0	Map Select 0

Bits 7 - 2 Reserved.

Bits 1, 0 Bits 1 and 0 represent a binary encoded value of the memory map from which the system microprocessor reads data. This register has no effect on the Color Compare Read mode. In odd/even modes the value may be 00 or 01 (10 or 11) for chained maps 0, 1 (2, 3).

Graphics Mode Register

This is a read/write register pointed to by the value in the Graphics Address register. This value must be hex 05 before writing can occur. The port address for this register is 03CF.

Figure 3-73. Graphics Mode Register, Index Hex 05

Bit	Function
7	Reserved = 0
6	256-Color Mode
5	Shift Register Mode
4	Odd/Even
3	Read Type
2	Reserved = 0
1, 0	Write Mode

Bit 7 Reserved.

Bit 6 When bit 6 is set to 1, the shift registers are loaded in a manner that supports the 256-Color mode. When bit 6 is set to 0, bit 5 is allowed to control the loading of the shift registers.

Bit 5 When bit 5 is set to 1, the shift registers in the graphics section format the serial data stream with even-numbered bits from both maps on the even-numbered maps and odd-numbered bits from both maps on the odd-numbered maps. This bit is used for modes 4 and 5.

Bit 4 Odd/Even - When bit 4 is set to 1, the odd/even addressing mode is selected, which is used for emulation of the IBM Color/Graphics Monitor Adapter compatible modes. Normally, this value follows the value of the Memory Mode register bit 2 (Odd/even) of the sequencer.

Bit 3 Read Type - When bit 3 is set to 1, the system microprocessor reads the results of the comparison of the four memory maps and the Color Compare register, index hex 02. When bit 3 is set to 0, the system microprocessor reads data from the memory map selected by the Read Map Select register, unless bit 3 (Chain 4) of the Memory Mode register equals 1. In this case, the Read Map Select register, index hex 04, has no effect.

Bit 2 Reserved.

Bits 1, 0 Write mode.

The bit functions are defined in Figure 3-74.

Figure 3-74. Write Mode Bit Definitions

Bit 1	Bit 0	Function
0	0	Each memory map is written with the system microprocessor data rotated by the number of counts in the Data Rotate register, unless Set/Reset is enabled for the map. Maps for which Set/Reset is enabled are written with 8 bits of the value contained in the Set/Reset register for that map.
0	1	Each memory map is written with the contents of the system microprocessor latches. These latches are loaded by a system microprocessor read operation.
1	0	Memory map *n* (0 through 3) is filled with 8 bits of the value of data bit *n*.
1	1	Each map is written with 8 bits of the value contained in the Set/Reset register for that map (the Enable Set/Reset register has no effect). Rotated system microprocessor data is ANDed with the Bit Mask register data to form an 8-bit value that performs the same function as the Bit Mask register does in write modes 0 and 2 (see "Bit Mask Register" on page 3-86).

The logic function specified by the Function Select bits in the Data Rotate register, index hex 03, is applied to data being written to memory following modes 0, 2, and 3 above.

Miscellaneous Register

This is a read/write register pointed to by the value in the Graphics Address register. This value must be hex 06 before writing can occur. The port address for this register is hex 03CF.

Figure 3-75. Miscellaneous Register, Index Hex 06

Bit	Function
7 - 4	Reserved = 0
3	Memory Map 1
2	Memory Map 0
1	Odd/Even
0	Graphics Mode

Bits 7 - 4 Reserved.

Bits 3, 2 Bits 3 and 2 control the mapping of the regenerative buffer into the system microprocessor address space. The bit functions are defined in Figure 3-76.

Figure 3-76. Memory Map Bit Definitions

Bit 3	Bit 2	Function
0	0	Hex A0000 for 128KB
0	1	Hex A0000 for 64KB
1	0	Hex B0000 for 32KB
1	1	Hex B8000 for 32KB

Bit 1 Odd/Even - When bit 1 is set to 1, the system microprocessor address bit 0 is replaced by a more-significant bit and odd/even maps to be selected with odd/even values of the system microprocessor A0 bit, respectively.

Bit 0 Graphics Mode - Bit 0 controls alphanumeric mode addressing. When bit 0 is set to 1, graphics mode is selected and the character generator address latches are disabled.

Color Don't Care Register

This is a read/write register and is pointed to by the value in the Graphics Address register. This value must be hex 07 before writing can occur. The port address for this register is hex 03CF.

Figure 3-77. Color Don't Care Register, Index Hex 07

Bit	Function
7 - 4	Reserved = 0
3	Map 3 = Don't Care
2	Map 2 = Don't Care
1	Map 1 = Don't Care
0	Map 0 = Don't Care

Bits 7 - 4 Reserved.

Bits 3 - 0 When any of bits 3 through 0 are set to 1, the associated map is included in the color compare cycle. When any of bits 3 through 0 are set to 0, the associated map is not included in the color compare cycle.

Bit Mask Register

This is a read/write register and is pointed to by the value in the Graphics Address register. This value must be hex 08 before writing can occur. The port address for this register is hex 03CF.

If any bit in this register is set to 1, writes are allowed to the corresponding bit in each map. When any bit is set to 0, the corresponding bit in each map will not change, provided that the location being written was the last location read by the system microprocessor.

The bit mask applies to write modes 0 and 2. To preserve bits using the bit mask, data must be latched internally by reading the location. When data is written to preserve the bits, the most current data in the latches is written in those positions. The bit mask applies to all maps simultaneously.

Attribute Controller Registers

This section contains descriptions of the following registers.

Figure 3-78. Attribute Controller Register Overview

Name	Port (hex)	Index (hex)
Address Register	03C0/1	-
Palette Registers	03C0/1	00-0F
Attribute Mode Control Register	03C0/1	10
Overscan Color Register	03C0/1	11
Color Plane Enable Register	03C0/1	12
Horizontal PEL Panning Register	03C0/1	13
Color Select Register	03C0/1	14

Each Attribute Data register is written at hex 03C0 as described below. Data is read from the registers at address hex 03C1.

Attribute Address Register

This is a read/write register. The port address is hex 03C0.

Figure 3-79. Attribute Address Register

Bit	Function
7, 6	Reserved = 0
5	Palette Address Source
4 - 0	Attribute Address

Bits 7, 6 Reserved.

Bit 5 Bit 5 must be set to 1 for normal operation of the attribute controller. This enables the video memory data to access the Palette registers. Bit 5 must be set to 0 when loading the Palette registers.

Bits 4 - 0 Bits 4 through 0 is a binary value that points to the Attribute Data register where data is to be written.

The Attribute Controller register does not have an address bit input to control selection of the address and data registers. An internal address latch controls selection of either the address or data registers. To initialize the latch, an IOR instruction is issued to the attribute controller at address hex 03BA or 03DA. This clears the latch, and selects the address register. After the address register has been loaded with an OUT to hex 03C0, the next OUT instruction to hex 03C0 loads the data register.

The latch toggles each time an OUT instruction is issued to the attribute controller. It does not toggle on IN instructions for a read to 03C1. (See "VGA Programming Considerations" on page 3-93.)

Palette Registers

These are read/write registers pointed to by the value in the Attribute Address register. This value must be hex 00 - 0F before writing can occur. The output port address for these registers is hex 03C0. The input port address is hex 03C1.

Figure 3-80. Palette Registers, Index Hex 00-0F

Bit	Function
7, 6	Reserved = 0
5	P5
4	P4
3	P3
2	P2
1	P1
0	P0

Bits 7, 6 Reserved.

Bits 5 - 0 Bits 5 through 0 allow a dynamic mapping between the text attribute or graphic color input value and the display color on the CRT. The value in these six bits selects the appropriate color.

The Palette registers should be modified only during the vertical retrace interval to avoid problems with the displayed image. These internal Palette register values are sent to the video DAC, where they in turn serve as addresses into the video DAC internal registers. (See "Attribute Controller" on page 3-21.)

Attribute Mode Control Register

This is a read/write register pointed to by the value in the Attribute Address register. This value must be hex 10 before writing can occur. The output port address for this register is hex 03C0. The input port address for this register is hex 03C1.

Figure 3-81. Attribute Mode Control Register, Index Hex 10

Bit	Function
7	P5, P4 Select
6	PEL Width
5	PEL Panning Compatibility
4	Reserved = 0
3	Select Background Intensity or Enable Blink
2	Enable Line Graphics Character Codes
1	Monochrome/Color Emulation
0	Graphics/Alphanumeric Mode

Bit 7 P5, P4 Select - When bit 7 is set to 1, P5 and P4 are bits 1 and 0 of the Color Select register. When bit 7 is set to 0, P5 and P4 are the outputs of the Palette registers. (See "Attribute Controller" on page 3-21 and "Video DAC Programming Considerations" on page 3-103.)

Bit 6 PEL Width - When bit 6 is set to 1, the video pipeline is sampled so that 8 bits are available to select a color in the 256-Color mode (hex 13). Bit 6 should be set to 0 in all other modes.

Bit 5 PEL Panning Compatibility - When bit 5 is set to 1, a successful line compare in the CRT controller forces the output of the Horizontal PEL Panning register to 0 until +VSYNC becomes active, at which time the output returns to its programmed value. This bit allows a selected portion of a screen to be panned. When bit 5 is set to 0, line compare has no effect on the output of the Horizontal PEL Panning register.

Bit 4 Reserved.

Bit 3 Enable Blink/Select Background Intensity - When bit 3 is set to 1, the blink attribute in alphanumeric modes is enabled. Bit 3 must also be set to 1 for blinking graphics modes. When bit 3 is set to 0, the background intensity of the attribute input is selected. This mode is also available on the IBM Monochrome and Color/Graphics Monitor adapters.

Bit 2 Enable Line Graphics Character Codes - When bit 2 is set to 1, the special line graphics character codes are enabled for the IBM Monochrome emulation mode. When bit 2 is set to 0, the ninth dot will be the same as the background.

When enabled, this bit forces the ninth dot of a line graphic character to be identical to the eighth dot of the character. The line graphics character codes for the Monochrome emulation mode are hex C0 through hex DF.

For character fonts that do not use the line graphics character codes in the range of hex C0 through hex DF, bit 2 should be a logical 0. Otherwise, unwanted video information is displayed on the CRT screen.

BIOS sets this bit, the correct dot clock, and other registers when a 9-dot alphanumeric mode is set.

Bit 1 Monochrome/Color Emulation - When bit 1 is set to 1, a monochrome emulation mode is set. When bit 1 is set to 0, a color emulation mode is set.

Bit 0 Graphics/Alphanumeric Mode - When bit 0 is set to 1, graphics mode is selected. When bit 0 is set to 0, alphanumeric mode is selected.

Overscan Color Register

This is a read/write register pointed to by the value in the Attribute Address register. This value must be hex 11 before writing can occur. The output port address for this register is hex 03C0. The input port address for this register is hex 03C1.

This register determines the overscan (border) color displayed on the CRT.

The border is a band of color around the perimeter of the display area. Its width is the same as one 80-column character. This border is not supported in the 40-column alphanumeric modes or the 320-PEL graphics modes, except for mode hex 13.

Color Plane Enable Register

This is a read/write register pointed to by the value in the Attribute Address register. This value must be hex 12 before writing can occur. The output port address for this register is 03C0. The input port address for this register is 03C1.

Figure 3-82. Color Plane Enable Register, Index Hex 12

Bit	Function
7, 6	Reserved = 0
5, 4	Video Status MUX
3 - 0	Enable Color Plane

Bits 7, 6 Reserved.

Bits 5, 4 Video Status MUX - Bits 5 and 4 select two of the eight color outputs to be available on the status port. Figure 3-83 shows the combinations available and the color output wiring.

Figure 3-83. Color Output Wiring

Color Plane Register		Input Status Register 1	
Bit 5	Bit 4	Bit 5	Bit 4
0	0	P2	P0
0	1	P5	P4
1	0	P3	P1
1	1	P7	P6

Bits 3 - 0 Enable Color Plane - When any of bits 3 through 0 is a 1, the respective display memory color plane is enabled.

Horizontal PEL Panning Register

This is a read/write register pointed to by the value in the Attribute Address register. This value must be hex 13 before writing can occur. The output port address for this register is hex 03C0. The input port address for this register is hex 03C1.

Figure 3-84. Horizontal PEL Panning Register, Index Hex 13

Bit	Function
7 - 4	Reserved = 0
3 - 0	Horizontal PEL Panning

Bits 7 - 4 Reserved.

Bits 3 - 0 Bits 3 through 0 select the number of PELs to shift the video data horizontally to the left. PEL panning is available in both alphanumeric (A/N) and all points addressable (APA) modes. In modes 0+, 1+, 2+, 3+, 7, and 7+ the image can be shifted a maximum of 8 PELs. In 256-Color APA mode (mode 13), the image can be shifted a maximum of 3 PELs. Further panning may be accomplished by changing the start address in the CRT controller. In all other A/N and APA modes, the image can be shifted a maximum of 7 PELs. The sequence for shifting the image is shown in Figure 3-85.

Figure 3-85. Image Shifting

PEL Panning Register Value	Number of PELs Shifted to the Left		
	0+, 1+, 2+, 3+, 7, 7+	All Other Modes	Mode 13
0	1	0	0
1	2	1	-
2	3	2	1
3	4	3	-
4	5	4	2
5	6	5	-
6	7	6	3
7	8	7	-
8	0	-	-

Color Select Register

This is a read/write register pointed to by the value in the Attribute Address register. This value must be hex 14 before writing can occur. The port address for this register is hex 03C1 for input and hex 03C0 for output.

Figure 3-86. Color Select Register, Index Hex 14

Bit	Function
7 - 4	Reserved = 0
3	S_color 7
2	S_color 6
1	S_color 5
0	S_color 4

Bits 3, 2 Bits 3 and 2 are the 2 most-significant bits of the 8-bit digital color value in all modes except 256-Color Graphics mode. In 256-Color Graphics mode, the 8-bit attribute stored in video memory becomes the 8-bit digital color value sent to the Video DAC. (See "VGA Programming Considerations.")

Bits 1, 0 Bits 1 and 0 can be used in place of the P4, P5 outputs from the Palette registers to form the 8-bit digital color value sent to the Video DAC. (See "Attribute Mode Control Register" on page 3-89.) Bits 1 and 0 are also used to rapidly switch between sets of colors in the video DAC.

VGA Programming Considerations

The following are some programming considerations for the VGA.

- Certain internal timings must be guaranteed by the user to have the CRT controller perform properly. These timings can be guaranteed by ensuring that the rules listed below are followed when programming the CRT controller.
  - The Horizontal Total register (RO) must be greater than or equal to a value of 25 (decimal).
  - The minimum positive pulse width of the HSYNC signal must be four character clock units.
  - End Horizontal Retrace register must be programmed so that the HSYNC output goes to 0 at least one character clock time before the horizontal display enable signal goes to 1.

- Vertical Retrace Start register must be a minimum of one horizontal scan line greater than the Vertical Display Enable End register which defines where the vertical display enable signal ends.

All of the above rules are satisfied when the video mode is set by the BIOS.

- When bit 5 of the Attribute Mode Control register is 1, a successful line compare (see "Line Compare Register" on page 3-79) in the CRT controller forces the output of the Horizontal PEL Panning register to 0 until VSYNC occurs. When VSYNC occurs, the output returns to the programmed value. This allows the portion of the screen indicated by the Line Compare register to be operated on by the Horizontal PEL Panning register.
- A write to the Character Map Select register becomes valid on the next whole character line. No deformed characters are displayed by changing character generators in the middle of a character scan line.
- For 256-Color 320-by-200 Graphics mode (hex 13), the attribute controller is configured so that the 8-bit attribute stored in video memory for each PEL becomes the 8-bit address (P0 - P7) into the DAC. The user should not modify the contents of the internal Palette registers when using this mode.
- The following sequence should be followed when accessing any of the Attribute Data registers pointed to by the Attribute Address register:

  1. Disable interrupts
  2. Reset read/write latch
  3. Write to Attribute Address register
  4. Read from or write to a data register
  5. Enable interrupts.

- The Color Select register in the attribute controller section may be used to rapidly switch between sets of colors in the video DAC. When bit 7 of the Attribute Mode Control register is 0, the 8-bit color value presented to the video DAC is composed of 6 bits from the internal Palette registers and bits 2 and 3 from the Color Select register. When bit 7 of the Attribute Mode Control register is 1, the 8-bit color value presented to the video DAC is composed of the lower 4 bits from the internal Palette registers and the 4 bits in the Color Select register. By changing the value in the Color Select register, the software rapidly switches between sets of colors in the video DAC. Note that BIOS does not support multiple sets of colors in the video DAC. The user must load

these colors if this function is to be used. (See "Attribute Controller" on page 3-21.) Note that the above discussion applies to all modes except 256-Color Graphics mode. In this mode, the Color Select register is not used to switch between sets of colors.

- An application that saves the video state must store the 4 bytes of information contained in the system microprocessor latches in the graphics controller subsection. These latches are loaded with 32 bits from video memory (8 bits per map) each time the system microprocessor does a read from video memory. The application needs to:

  1. Use write mode 1 (Graphics Mode register) to write the values in the latches to a location in video memory that is not part of the display buffer. The last location in the address range is a good choice.
  2. Save the values of the latches by reading them back from video memory.

  Note: If in a chain 4 or odd/even mode (Memory Mode register), it will be necessary to reconfigure the memory organization as four sequential maps prior to performing the sequence above. BIOS provides support for completely saving and restoring video state. (See *BIOS Interface Technical Reference for IBM PS/1™ Computer.*)

- The description of the Horizontal PEL Panning register includes a figure showing the number of PELs shifted left for each valid value of the Horizontal PEL Panning register and each valid video mode. Further panning beyond that shown in the figure may be accomplished by changing the start address in the CRT Controller registers (Start Address High and Start Address Low). The sequence involved in further panning would be as follows:

  1. Use the Horizontal PEL Panning register to shift the maximum number of bits to the left. (See Figure 3-84 on page 3-92 for the appropriate values.)
  2. Increment the start address.
  3. If modes 0+, 1+, 2+, 3+,7, or 7+ are not used, set the Horizontal PEL Panning register to 0. If these modes are used, set the Horizontal PEL Panning register to 8. The screen will now be shifted 1 PEL left of the position it was in at the end of step 1. Step 1 through step 3 may be repeated as desired.

- The Line Compare register (CRT Controller register, index hex 18) should be programmed with even values in 200-line modes

when used in split screen applications that scroll a second screen on top of a first screen. This is a requirement imposed by the double scan logic in the CRT Controller.

- If the Cursor Start register (CRT Controller register, index hex 0A) is programmed with a value greater than that in the Cursor End register (CRT Controller register, index hex 0B), then no cursor is displayed. A split cursor is not possible.
- In 8-dot character modes, the underline attribute produces a solid line across adjacent characters, as in the IBM Color/Graphics Monitor Adapter, Monochrome Display Adapter and the Enhanced Graphics Adapter. In 9-dot modes, the underline across adjacent characters is dashed, as in the IBM 327X display terminals. In 9-dot modes, the line graphics characters (hex C0 - DF character codes) have solid underlines.

Programming the Registers

Each of the video subsections has an address register and a number of data registers. The address register serves as a pointer to the other registers on the device. These registers are loaded by the system microprocessor by executing an OUT instruction to its I/O address with the index of the selected data register.

The data registers in each subsection are accessed through a common I/O address. They are distinguished by the pointer (index) in the address register. To write to a data register, the address register is loaded with the index of the appropriate data register, then the selected data register is loaded by executing an OUT instruction to the common I/O address.

The general registers are not accessed through an address register; they are written directly.

See "Video DAC/System Microprocessor Interface" on page 3-102 for details on accessing the video DAC.

For compatibility with the IBM Enhanced Graphics Adapter (EGA), the internal VGA palette is programmed the same as the EGA. The video DAC is programmed by BIOS so that the compatible values in the internal VGA palette produce a color compatible with that produced by EGA. Mode hex 13 (256 colors) is programmed so that the first 16 locations in the DAC produce compatible colors.

The color palette is changed when BIOS is used to load the video DAC palette for a color mode, and a monochrome display is connected to the system unit. The colors are summed to produce shades of gray that allow color applications to produce a readable screen.

There are 4 bits that should not be modified (unless the sequencer is reset by setting bit 1 of the Reset register to 0). These bits are:

- Bit 3 or bit 0 of the Clocking Mode register
- Bit 3 or bit 2 of the Miscellaneous Output register.

RAM-Loadable Character Generator

The character generator is RAM-loadable and can support characters up to 32 scan lines high. Three character generators are stored within the BIOS, and one is automatically loaded into the RAM by the BIOS when an alphanumeric mode is selected. The Character Map Select register can be programmed to define the function of bit 3 of the attribute byte to be a character generator switch. This allows the user to select between any two character sets residing in map 2. This effectively gives the user access to 512 characters instead of 256. Character tables can be loaded offline. Up to eight tables can be loaded.

The structure of the character tables is described in Figure 3-87. The character generator is in map 2 and must be protected using the map mask function.

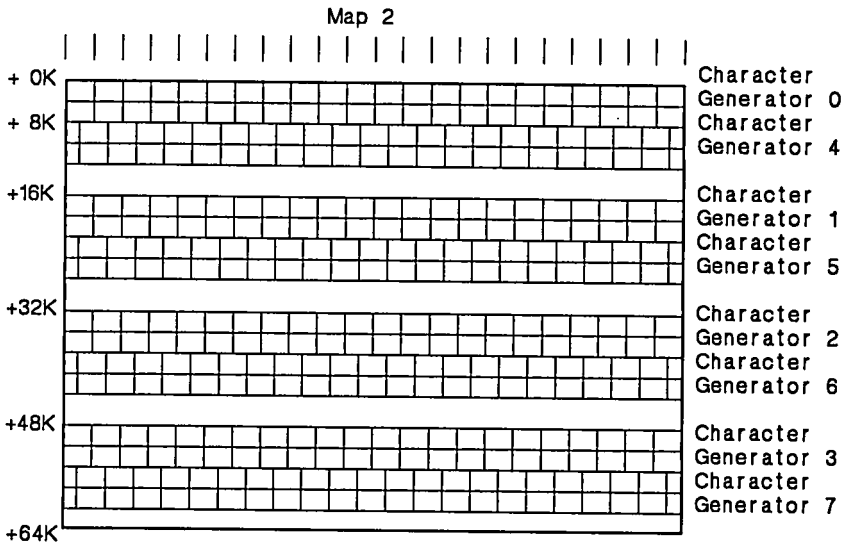


Figure 3-87. Character Table Structure

Figure 3-88 shows the structure of each character pattern. If the CRT controller is programmed to generate *n* row scans, then *n* bytes must be filled in for each character in the character generator. The example assumes eight row scans per character.

Address	Byte Image								Data
CC * 32 + 0				X	X				18H
1			X	X	X	X			3EH
2		X	X			X	X		66H
3		X	X			X	X		66H
4		X	X	X	X	X	X		7EH
5		X	X			X	X		66H
6		X	X			X	X		66H
7		X	X			X	X		66H

Figure 3-88. Character Pattern Example

Note: CC equals the value of the character code. For example, hex 41 is the character code for an ASCII A.

Creating a Split Screen

The VGA hardware supports a dual-screen display. The top portion of the screen is designated as screen A and the bottom portion of the screen is designated as screen B, as shown in Figure 3-89.

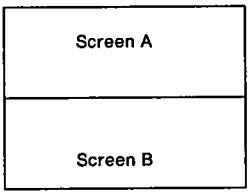


Figure 3-89. Dual-Screen Definition

Figure 3-90 shows the screen mapping for a system containing a 32KB alphanumeric storage buffer. Note that the VGA has a 32KB storage buffer in alphanumeric mode. Information displayed on screen A is defined by the Start Address High and Low registers (index hex 0CH and 0DH) of the CRT controller. Information displayed on screen B always begins at address hex 0000.

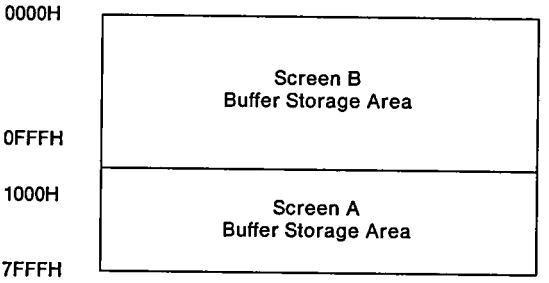


Figure 3-90. Screen Mapping within the Display Buffer Address Space

The Line Compare register (index hex 18H) of the CRT controller performs the split screen function. The CRT controller has an internal horizontal scan line counter. The CRT controller also has logic that compares the horizontal scan line counter value to the Line Compare register value and clears the memory address generator when a compare occurs. The linear address generator then sequentially addresses the display buffer starting at location 0, and each subsequent row address is determined by the 16-bit addition of the start of line latch and the Offset register.

Screen B can be smoothly scrolled onto the CRT by updating the Line Compare in synchronization with the vertical retrace signal. The information on screen B is not affected by scrolling operations that utilize the Start Address High and Low registers to scroll through the Screen A address map.

When bit 5 of the Attribute Mode Control register is 1, a successful line compare forces the output of the Horizontal PEL Panning register to 0 until vertical synchronization occurs. When VSYNC occurs, the output returns to its programmed value. This feature allows the information on screen B to remain unaffected by PEL panning operations on screen A.

Video Digital-to-Analog Converter (Video DAC)

The video DAC integrates the function of a color look-up table with three internal DACs for driving an analog display.

The size of the color look-up table is 256 by 18 bits to allow the display of 256 colors from a palette of over 256,000 possible colors. Each RGB analog output is driven by a 6-bit DAC. Each register in the color look-up table contains 6 bits each for the red, green, and blue DACs.

Figure 3-91. Video DAC I/O Address Usage
Video Digital-to-Analog Converter (DAC) Addresses in Hex

PEL Address (Write mode)	RW	03C8
PEL Address (Read mode)	WO	03C7
DAC State Register	RO	03C7
PEL Data Register	RW	03C9
PEL Mask *	RW	03C6

RO = Read-Only, RW = Read / Write, WO = Write-Only.

* This register must not be written to by application code, or destruction of the color look-up table may occur. (See "Video DAC Programming Considerations" on page 3-103.)

Video DAC Operation

The PEL address inputs (P0 - P7) and the blanking input are sampled on the rising edge of the PEL clock. After three further rising edges of the PEL clock, the analog outputs reflect the state of these inputs.

During normal operation, the PEL address inputs (P0 - P7) are used as a pointer to one of the 256 internal registers (color look-up table). The value in each register is then, in turn, converted to an analog signal for each of the three analog outputs (red, green, blue). The blanking input can also be used to force the analog outputs to 0 volts. The blanking operation is independent of the state of the PEL address inputs.

During system microprocessor accesses, the 8-bit PEL Address register acts as a pointer to the 256 internal registers. Each internal register is 18 bits wide; 6 bits each for red, green, and blue. The internal registers are accessible through the system microprocessor interface as described on the following page.

The system microprocessor interface is asynchronous with the video path. The timing of this interface is controlled by the write enable and read enable signals.

Video DAC/System Microprocessor Interface

The PEL Address register holds an 8-bit value that is used to address a location within the color look-up table. The PEL Address register may be written at two different addresses to establish a read or write mode, respectively. Once the PEL Address register has been written and an access has been made to a location in the color look-up table, the PEL Address register automatically increments and further accesses may occur to successive locations.

Each time the PEL Address register is written at address hex 03C8, it identifies that a write sequence will occur. A write sequence consists of three successive byte writes to the PEL Data register at address hex 03C9. The least-significant 6 bits of each byte are concatenated to form the value placed in the 18-bit Data register. The order is red byte first, then green, and finally blue. Once the third byte has been written, the value in the Data register is written to the location pointed to by the PEL Address register. The order of events for a write cycle is:

1. Write the PEL Address register at hex 03C8.
2. Three bytes are written to the PEL Data register at hex 03C9.
3. The contents of the PEL Data register are transferred to the location in the color look-up table pointed to by the PEL Address register.
4. The PEL Address register auto-increments by 1.
5. Go to step 2.

Each time the PEL Address register is written at address hex 03C7, it identifies that a read sequence will occur. A read sequence consists of three successive byte reads from the PEL Data register at address hex 03C9. The least-significant 6 bits of each byte taken from the PEL Data register contain the corresponding color value. The order is red byte first, then green, and finally blue. The order of events for a read cycle is:

1. Write the PEL Address register at hex 03C7.
2. The contents of the location in the color look-up table pointed to by the PEL Address register are transferred to the PEL Data register.

3. The PEL Address register auto-increments by one.
4. Three bytes are read back from the PEL Data register at hex 03C9.
5. Go to step 2.

If the PEL Address register is written during either a read or write cycle, a mode is initialized and the unfinished cycle is aborted. The effects of writing the PEL Data register during a read cycle or reading the PEL Data register during a write cycle are undefined and may change the look-up table contents.

A read from address hex 03C7 returns 0's in bit positions 0 and 1 if the video DAC is currently in a read mode. A read from address hex 03C7 returns 1's in bit positions 0 and 1 if the video DAC is currently in a write mode.

Reads from the PEL Address register at hex 03C8 or the DAC State register at hex 03C7 do not interfere with read or write cycles and may occur at any time.

Video DAC Programming Considerations

As explained above, the effects of writing the PEL Data register during a read cycle or reading the PEL Data register during a write cycle are undefined and may change the look-up table contents. Therefore, the following sequence must be followed to ensure the color look-up table integrity during accesses to it:

1. Write address to the PEL Address register.
2. Disable Interrupts.
3. Write or read three bytes of data.
4. Go to step 3. Repeat this step for the desired number of locations.
5. Enable interrupts.

Note: The above sequence assumes that any interrupting process will return the video DAC in the correct mode (write or read). If this is not the case, the sequence shown below should be followed:

1. Disable interrupts.
2. Write address to PEL Address register.
3. Write or read three bytes of data.
4. Go to step 2. Repeat this step for the desired number of locations.
5. Enable interrupts.

There is a timing requirement on the minimum amount of time that must separate the trailing edge of one Read or Write command to the video DAC and the leading edge of the next Read or Write command. The minimum separation is 240 nanoseconds. Software must ensure that the 240-nanosecond separation exists between two successive accesses to the video DAC. Assembly language programs can meet this requirement by placing a JMP $+2 instruction between successive accesses to the video DAC.

To prevent *snow* on the screen, an application reading or writing the Video DAC register should ensure that the blanking input to the video DAC is asserted. This can be accomplished either by restricting data transfers to retrace intervals (use Input Status register 1 to determine when retrace is occurring) or by using the screen off bit located in the Clocking Mode register of the sequencer subsection.

Note: BIOS provides read and write interfaces to the video DAC.

The PEL Mask register (hex 03C6) must not be written by application code, or destruction of the color look-up table may result. This register is correctly initialized to hex FF by BIOS during a video mode set.

Display Connector Timing (SYNC Signals)

BIOS sets the VGA registers to generate the video modes. The video modes are shown in Figure 3-13 on page 3-22. All of these modes are 70 Hz vertical retrace except for modes 11 and 12. These two modes are 60 Hz vertical retrace. The VGA generates timings that are within the specifications for the supported displays using these modes.

The analog displays operate from 60 to 70 Hz vertical retrace frequency. The following timing diagrams represent only the vertical frequencies set by BIOS.

Note: The vertical size of the display is encoded using the polarity of the SYNC signals as shown in Figure 3-92. (See "Miscellaneous Output Register" on page 3-54.) The polarity of the SYNC pulse refers to whether it is a positive pulse (with respect to ground) or a negative pulse (with respect to +5 V dc).

Figure 3-92. Display Vertical Size

VSYNC Polarity	HSYNC Polarity	Vertical Size
+	+	Reserved
-	+	350 lines
+	-	400 lines
-	-	480 lines

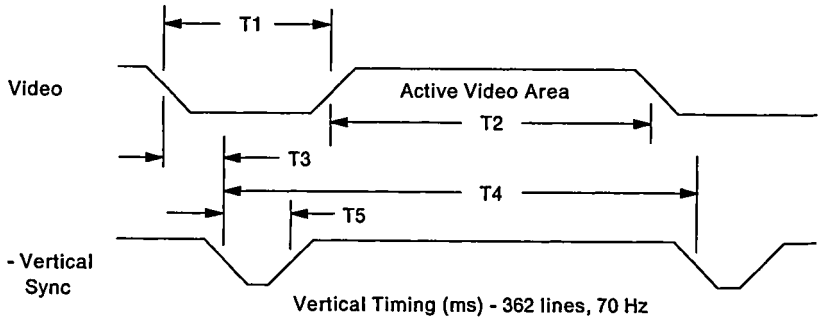


Figure 3-93. Display Vertical SYNC, 350 Lines

Signal	Time
T1	2.765 milliseconds
T2	11.504 milliseconds
T3	0.985 milliseconds
T4	14.268 milliseconds
T5	0.064 milliseconds

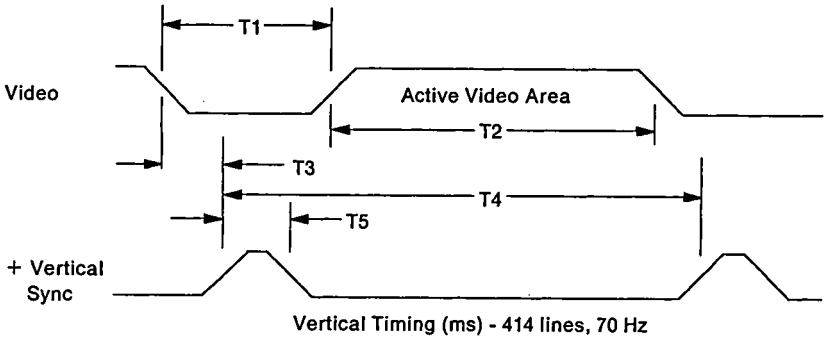


Figure 3-94. Display Vertical SYNC, 400 Lines

Signal	Time
T1	1.112 milliseconds
T2	13.156 milliseconds
T3	0.159 milliseconds
T4	14.268 milliseconds
T5	0.064 milliseconds

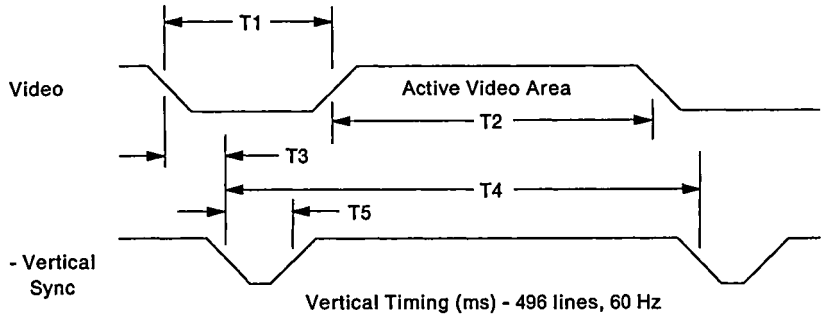
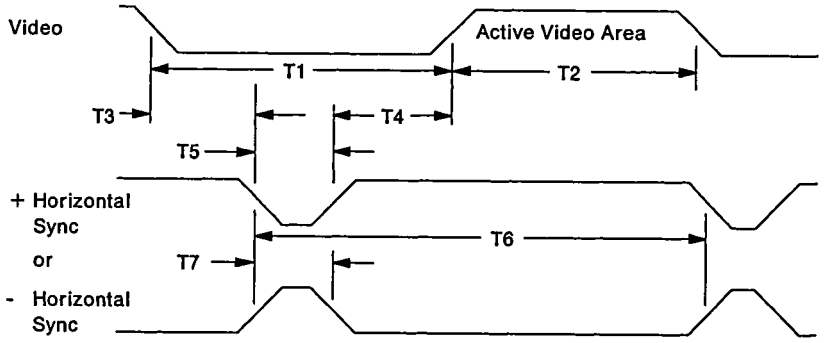


Figure 3-95. Display Vertical SYNC, 480 Lines

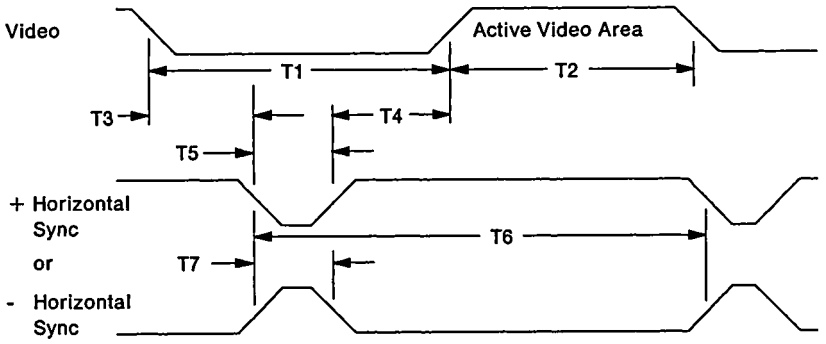
Signal	Time
T1	0.922 milliseconds
T2	15.762 milliseconds
T3	0.064 milliseconds
T4	16.683 milliseconds
T5	0.064 milliseconds



Horizontal Timing (μs) – 80 column with border

Figure 3-96. Display Horizontal Timing, 80 Column with Border

Signal	Time
T1	5.720 microseconds
T2	26.058 microseconds
T3	0.318 microseconds
T4	1.589 microseconds
T5	3.813 microseconds
T6	31.778 microseconds
T7	3.813 microseconds



Horizontal Timing (μs) – 40/80 column, no border

Figure 3-97. Display Horizontal Timing, 40/80 Column, no Border

Signal	Time
T1	6.356 microseconds
T2	25.422 microseconds
T3	0.636 microseconds
T4	1.907 microseconds
T5	3.813 microseconds
T6	31.778 microseconds
T7	3.813 microseconds

Display Connector

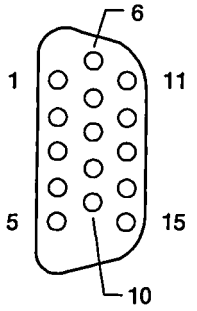


Figure 3-98. Display Connector Signals

Pin	Signal
1	Red Video
2	Green Video
3	Blue Video
4	NC
5	Ground
6	Ground (Analog)
7	Ground (Analog)
8	Ground (Analog)
9	NC
10	Ground
11	NC
12	NC
13	Horizontal SYNC
14	Vertical SYNC
15	NC

Diskette Drive Controller

The diskette drive controller and connector are on the system board.

Note: This section is included as a guide to compatibility for existing software. New software should access the diskette drive controller through BIOS.

The Intel 82077AA diskette controller interfaces with the diskette drives. The controller operates in the IBM PS/2® Model 30 compatibility mode. The following are supported by the system board:

- Two drives
- 1.44MB, 3.5-inch drive accepting either 1MB or 2MB unformatted media
- 360KB, 5.25-inch drive
- 1.2MB, 5.25-inch drive
  - using both 1.2MB and 360KB formats
  - operating at 500K bps at high density and 300K bps at low density.

Precompensation of 125 nanoseconds is provided for all cylinders.

The diskette drive controller reads and writes both high- and low-density media.

Warning: Sixteen-bit operations to the video subsystem can cause a diskette overrun in the 1.44MB mode because data width conversions may require more than 12 microseconds. If an overrun occurs, BIOS returns an error code and the operation should be retried.

Warning: The diskette controller does not check if the media supports the density selected. Low density media (360KB or 720KB) will not be reliably formatted to the high density (1.2MB or 1.44MB) and loss of data may result. High density media will not be reliably formatted to the low density and loss of data may result. The diskette should be formatted at the correct density.

PS/2 is a registered trademark of the International Business Machines Corporation.

Registers

Status Register A

This is a read-only register that shows the status of the corresponding signals. The input port address is hex 03F0.

Figure 3-99. Status Register A (Hex 03F0)

Bit	Function
7	IRQ6
6	DRQ2
5	Step (latched)
4	Track 0
3	-Head 1 Select
2	Index
1	Write Protect
0	-Direction

Status Register B

This is a read-only register that shows the status of signals between the diskette drive and the controller. The input port address is hex 03F1.

Figure 3-100. Status Register B (Hex 03F1)

Bit	Function
7	-Drive 2 Installed
6	-Drive Select 1
5	-Drive Select 0
4	Write Data (latched)
3	Read Data (latched)
2	Write Enable (latched)
1, 0	Reserved

Digital Output Register

This is a write-only register used to control drive motors, drive selection, and feature enable. All bits are cleared by a Reset. The output port address is hex 03F2.

Figure 3-101. Digital Output Register (Hex 03F2)

Bit	Function
7, 6	Reserved
5	Motor Enable 1
4	Motor Enable 0
3	DMA and Interrupt Enable
2	-Controller Reset
1, 0	Drive Select (0 or 1)*

* 00 = drive 0, 01 = drive 1

Digital Input Register

This is a read-only register used to sense the state of the diskette change signal and for diagnostic purposes. The input port address is hex 03F7.

Figure 3-102. Digital Input Register (Hex 03F7)

Bit	Function
7	-Diskette Change
6 - 4	Reserved
3	DMA enable
2	No Write Pre Comp
1, 0	Data Rate Select (0 or 1)*

* 00 = 500K bps, 10 = 250K bps, 01 = 300K bps

Configuration Control Register

This is a write-only register used to set the transfer rate. The output port address is hex 03F7.

Figure 3-103. Configuration Control Register (Hex 03F7)

Bit	Function
7 - 3	Reserved = 0
2	No write Pre Comp
1, 0	Data Rate Select (0 or 1)*

* 00 = 500K bps, 10 = 250K bps, 01 = 300K bps

Diskette Drive Controller Status Register

This is a read-only register used to facilitate the transfer of data between system microprocessor and the controller. The input port address is hex 03F4.

Figure 3-104. Diskette Drive Controller Status Register (Hex 03F4)

Bit	Function
7	Request for Master
6	Data Input/Output
5	Non-DMA Mode
4	Diskette Controller Busy
3	Reserved
2	Reserved
1	Drive 1 Busy
0	Drive 0 Busy

Bit 7 When bit 7 is set to 1, the Data register is ready for transfer with the system microprocessor.

Bit 6 When bit 6 is set to 1, the data transfer is from the diskette drive controller to the system microprocessor. When bit 6 is set to 0, the data transfer is from the system microprocessor to the diskette drive controller.

Bit 5 When bit 5 is set to 1, the diskette drive controller is in the non-DMA mode.

Bit 4 When bit 4 is set to 1, a Read or Write command is being executed.

Bit 3, 2 Reserved.

Bit 1 When bit 1 is set to 1, diskette drive 1 is in the seek mode.

Bit 0 When bit 0 is set to 1, diskette drive 0 is in the seek mode.

Data Registers, Hex 03F5

Address hex 03F5 is the address/data register for several registers in a stack with only one register presented to the data bus at a time. These registers store data, commands, and parameters, and provide diskette-drive status information. Data bytes are passed through the address/data register to program or obtain results after a command.

Diskette Drive Controller Programming Considerations

Each command is initiated by a multibyte transfer from the system microprocessor, and the result can also be a multibyte transfer back to the system microprocessor. Because of this multibyte interchange of information between the diskette drive controller and the microprocessor, each command is considered to consist of three phases:

Command Phase: The system microprocessor issues a series of writes to the diskette drive controller that direct it to perform a specific operation.

Execution Phase: The controller performs the specified operation.

Result Phase: After completion of the operation, status and other housekeeping information is made available from the system microprocessor through a sequence of Read commands.

The following is a summary of the commands that are issued to the diskette drive controller.

- Read Data command
- Read Deleted-Data command
- Read Track command
- Read ID command
- Write Data command
- Write Deleted-Data command
- Format Track command
- Recalibrate command
- Sense Interrupt Status command
- Specify command
- Sense Drive Status command
- Seek command.

Figure 3-105 shows the symbols used in the format of the individual commands. (See "Command Format" on page 3-118 for the individual commands.)

Figure 3-105 (Page 1 of 2). Command Symbols, Diskette Drive Controller

Symbol	Name	Description
C	Cylinder Number	C contains the current or selected cylinder number in binary notation.
D	Data	D contains the data pattern to be written to a sector.
D7-D0	Data Bus	An 8-bit data bus in which D7 is the most-significant bit and D0 is the least-significant bit.
DTL	Data Length	When N is 00, DTL is the data length to be read from or written to a sector.
EOT	End of Track	The final sector number on a cylinder.
GPL	Gap Length	The length of gap 3 (spacing between sectors excluding the voltage-controlled oscillator synchronous field).
H	Head Address	The head number, either 0 or 1, as specified in the ID field.
HD	Head	The selected head number, 0 or 1 (H=HD in all command words).
HLT	Head Load Time	The head load time in the selected drive (2 to 254 milliseconds in 2-millisecond increments).
HUT	Head Unload Time	The head unload time after a read or write operation (16 to 240 milliseconds in 16-millisecond increments).

Figure 3-105 (Page 2 of 2). Command Symbols, Diskette Drive Controller

Symbol	Name	Description
MF	FM or MFM Mode	A 0 selects FM mode and a 1 selects MFM.
MT	Multitrack	A 1 selects multitrack operation (both HD0 and HD1 will be read or written).
N	Number	The number of data bytes written in a sector.
NCN	New Cylinder Number	The new cylinder number for a seek operation.
ND	Non-Data Mode	This indicates an operation in the non-data mode.
PCN	Present Cylinder Number	The cylinder number at the completion of a Sense Interrupt Status command (present position of the head).
R	Record	The sector number to be read or written.
SC	Sector	The number of sectors per cylinder.
SK	Skip	This stands for skip deleted-data address mark.
SRT	Step Rate	The stepping rate for the diskette drive (1 to 16 milliseconds in 1-millisecond increments).
ST 0 - ST 3	Status 0-Status 3	Four registers that store status information after a command is executed.
STP	Scan Test	If STP is 1, the data in contiguous sectors is compared with the data sent by the system microprocessor during a scan operation. If STP is 2, then alternate sectors are read and compared.
US0 - US1	Unit Select	The selected drive number encoded the same as bits 0 and 1 of the Digital Output register.

Command Format

This section contains the format of the commands that are issued to the diskette drive controller.

The following abbreviations are used in the figures on the following pages. An X is used to indicate a *don't care* condition.

MT = Multitrack
MF = MFM mode
SK = Skip deleted-data address mark
HD = Head number
USX = Unit select
SRT = Diskette stepping rate
HUT = Head unload time
HLT = Head load time
ND = Non-data mode.

Read Data Command Format

Command Phase

Figure 3-106. Read Data Command

	7	6	5	4	3	2	1	0
Byte 0	MT	MF	SK	0	0	1	1	0
Byte 1	X	X	X	X	X	HD	US1	US0
Byte 2	Cylinder Number (C)							
Byte 3	Head Address (H)							
Byte 4	Sector Number (R)							
Byte 5	Number of Data Bytes in Sector (N)							
Byte 6	End of Track (EOT)							
Byte 7	Gap Length (GPL)							
Byte 8	Data Length (DTL)							

Result Phase

Figure 3-107. Read Data Result

	7	6	5	4	3	2	1	0
Byte 0	Status Register 0 (ST 0)							
Byte 1	Status Register 1 (ST 1)							
Byte 2	Status Register 2 (ST 2)							
Byte 3	Cylinder Number (C)							
Byte 4	Head Address (H)							
Byte 5	Sector Number (R)							
Byte 6	Number of Data Bytes in Sector (N)							

Read Deleted-Data Command Format

Command Phase

Figure 3-108. Read Deleted-Data Command

	7	6	5	4	3	2	1	0
Byte 0	MT	MF	SK	0	1	1	0	0
Byte 1	X	X	X	X	X	HD	US1	US0
Byte 2	Cylinder Number (C)							
Byte 3	Head Address (H)							
Byte 4	Sector Number (R)							
Byte 5	Number of Data Bytes in Sector (N)							
Byte 6	End of Track (EOT)							
Byte 7	Gap Length (GPL)							
Byte 8	Data Length (DTL)							

Result Phase

Figure 3-109. Read Deleted-Data Result

	7	6	5	4	3	2	1	0
Byte 0	Status Register 0 (ST 0)							
Byte 1	Status Register 1 (ST 1)							
Byte 2	Status Register 2 (ST 2)							
Byte 3	Cylinder Number (C)							
Byte 4	Head Address (H)							
Byte 5	Sector Number (R)							
Byte 6	Number of Data Bytes in Sector (N)							

Read a Track Command Format

Command Phase

Figure 3-110. Read a Track Command

	7	6	5	4	3	2	1	0
Byte 0	0	MF	0	0	0	0	1	0
Byte 1	X	X	X	X	X	HD	US1	US0
Byte 2	Cylinder Number (C)							
Byte 3	Head Address (H)							
Byte 4	Sector Number (R)							
Byte 5	Number of Data Bytes in Sector (N)							
Byte 6	End of Track (EOT)							
Byte 7	Gap Length (GPL)							
Byte 8	Data Length (DTL)							

Result Phase

Figure 3-111. Read a Track Result

	7	6	5	4	3	2	1	0
Byte 0	Status Register 0 (ST 0)							
Byte 1	Status Register 1 (ST 1)							
Byte 2	Status Register 2 (ST 2)							
Byte 3	Cylinder Number (C)							
Byte 4	Head Address (H)							
Byte 5	Sector Number (R)							
Byte 6	Number of Data Bytes in Sector (N)							

Read ID Command Format

Command Phase

Figure 3-112. Read ID Command

	7	6	5	4	3	2	1	0
Byte 0	0	MF	0	0	1	0	1	0
Byte 1	X	X	X	X	X	HD	US1	US0

Result Phase

Figure 3-113. Read ID Result

	7	6	5	4	3	2	1	0
Byte 0	Status Register 0 (ST 0)							
Byte 1	Status Register 1 (ST 1)							
Byte 2	Status Register 2 (ST 2)							
Byte 3	Cylinder Number (C)							
Byte 4	Head Address (H)							
Byte 5	Sector Number (R)							
Byte 6	Number of Data Bytes in Sector (N)							

Write Data Command Format

Command Phase

Figure 3-114. Write Data Command

	7	6	5	4	3	2	1	0
Byte 0	MT	MF	0	0	0	1	0	1
Byte 1	X	X	X	X	X	HD	US1	US0
Byte 2	Cylinder Number (C)							
Byte 3	Head Address (H)							
Byte 4	Sector Number (R)							
Byte 5	Number of Data Bytes in Sector (N)							
Byte 6	End of Track (EOT)							
Byte 7	Gap Length (GPL)							
Byte 8	Data Length (DTL)							

Result Phase

Figure 3-115. Write Data Result

	7	6	5	4	3	2	1	0
Byte 0	Status Register 0 (ST 0)							
Byte 1	Status Register 1 (ST 1)							
Byte 2	Status Register 2 (ST 2)							
Byte 3	Cylinder Number (C)							
Byte 4	Head Address (H) Bits 7 - 0							
Byte 5	Sector Number (R)							
Byte 6	Number of Data Bytes in Sector (N)							

Write Deleted-Data Command Format

Command Phase

Figure 3-116. Write Deleted-Data Command

	7	6	5	4	3	2	1	0
Byte 0	MT	MF	0	0	1	0	0	1
Byte 1	X	X	X	X	X	HD	US1	US0
Byte 2	Cylinder Number (C)							
Byte 3	Head Address (H)							
Byte 4	Sector Number (R)							
Byte 5	Number of Data Bytes in Sector (N)							
Byte 6	End of Track (EOT)							
Byte 7	Gap Length (GPL)							
Byte 8	Data Length (DTL)							

Result Phase

Figure 3-117. Write Deleted-Data Result

	7	6	5	4	3	2	1	0
Byte 0	Status Register 0 (ST 0)							
Byte 1	Status Register 1 (ST 1)							
Byte 2	Status Register 2 (ST 2)							
Byte 3	Cylinder Number (C)							
Byte 4	Head Address (H)							
Byte 5	Sector Number (R)							
Byte 6	Number of Data Bytes in Sector (N)							

Format a Track Command Format

Command Phase

Figure 3-118. Format a Track Command

	7	6	5	4	3	2	1	0
Byte 0	0	MF	0	0	1	1	0	1
Byte 1	X	X	X	X	X	HD	US1	US0
Byte 2	Number of Data Bytes in Sector (N)							
Byte 3	Sectors per Cylinder (SC)							
Byte 4	Gap Length (GPL)							
Byte 5	Data (D)							

Result Phase

Figure 3-119. Format a Track Result

	7	6	5	4	3	2	1	0
Byte 0	Status Register 0 (ST 0)							
Byte 1	Status Register 1 (ST 1)							
Byte 2	Status Register 2 (ST 2)							
Byte 3	Reserved							
Byte 4	Reserved							
Byte 5	Reserved							
Byte 6	Reserved							

Recalibrate Command Format

Command Phase

Figure 3-120. Recalibrate Command

	7	6	5	4	3	2	1	0
Byte 0	0	0	0	0	0	1	1	1
Byte 1	X	X	X	X	X	0	US1	US0

Result Phase: This command has no result phase.

Sense Interrupt Status Command Format

Command Phase

Figure 3-121. Sense Interrupt Status Command

	7	6	5	4	3	2	1	0
Byte 0	0	0	0	0	1	0	0	0

Result Phase

Figure 3-122. Sense Interrupt Status Result

	7	6	5	4	3	2	1	0
Byte 0	Status Register 0 (ST 0)							
Byte 1	Present Cylinder Number (PCN)							

Specify Command Format

Command Phase

Figure 3-123. Specify Command

	7	6	5	4	3	2	1	0
Byte 0	0	0	0	0	0	0	1	1
Byte 1	SRT	SRT	SRT	SRT	HUT	HUT	HUT	HUT
Byte 2	HLT	HLT	HLT	HLT	HLT	HLT	HLT	ND

Result Phase: This command has no result phase.

Sense Drive Status Command Format

Command Phase

Figure 3-124. Sense Drive Status Command

	7	6	5	4	3	2	1	0
Byte 0	0	0	0	0	0	1	0	0
Byte 1	X	X	X	X	X	HD	US1	US0

Result Phase

Figure 3-125. Sense Drive Status Result

	7	6	5	4	3	2	1	0
Byte 0	Status Register 3 (ST 3)							

Seek Command Format

Command Phase

Figure 3-126. Seek Command

	7	6	5	4	3	2	1	0
Byte 0	0	0	0	0	1	1	1	1
Byte 1	X	X	X	X	X	HD	US1	US0
Byte 2	New Cylinder Number for Seek (NCN)							

Result Phase: This command has no result phase.

Invalid Command Format

Result Phase: The following status byte is returned to the system microprocessor when an invalid command has been received.

Figure 3-127. Invalid Command Result

	7	6	5	4	3	2	1	0
Byte 0	Status Register 0 (ST 0)							

Command Status Registers

This section contains definitions of status registers ST 0 through ST 3.

Status Register 0

Figure 3-128 shows the bit definitions of status register ST 0.

Figure 3-128. Status Register 0 (ST 0)

Bit	Function
7, 6	Interrupt Code (IC) 00 = Normal Termination of Command (NT) - The command was completed and properly executed. 01 = Abrupt Termination of Command (AT) - The execution of the command was started but not successfully completed. 10 = Invalid Command Issue (IC) - The issued command was never started. 11 = Reserved (Abnormal Termination)
5	Seek End (SE) - Set to 1 when the diskette drive completes the Seek command.
4	Equipment Check (EC) - Set to 1 if the -track 0 signal fails to occur after 80 step pulses (Recalibrate command).
3	Set to 0.
2	Head Address (HD) - Indicates the state of the head at interrupt.
1	Unit select 1 (US 1) - Indicates drive 1 is at interrupt.
0	Unit select 0 (US 0) - Indicates drive 0 is at interrupt.

Status Register 1

Figure 3-129 shows the bit definitions of status register ST 1.

Figure 3-129. Status Register 1 (ST 1)

Bit	Function
7	End of Cylinder (EN) - Set to 1 when the controller tries to gain access to a sector beyond the final sector of a cylinder.
6	Reserved - This bit is always set to 0.
5	Data Error (DE) - Set to 1 when the controller detects a CRC error in either the ID field or the data field.
4	Overrun (OR) - Set to 1 if the controller is not serviced by the main system within a certain time limit during data transfers.
3	Reserved - This bit is always set to 0.
2	No Data (ND) - Set to 1 if the controller cannot find the sector specified in the ID register during the execution of a Read Data or Read Deleted-Data command. This flag is also set to 1 if the controller cannot read the ID field without an error during the execution of a Read ID command or if the starting sector cannot be found during the execution of a Read a Track command.
1	Not Writable (NW) - Set to 1 if the controller detects a write-protect signal from the diskette drive during execution of a Write Data, Write Deleted-Data, or Format a Track command.
0	Missing Address Mark (MA) - Set to 1 if the controller cannot detect the ID address mark. At the same time, the MD of Status register 2 is set to 1.

Status Register 2

Figure 3-130 shows the bit definitions of status register ST 2.

Figure 3-130. Status Register 2 (ST 2)

Bit	Function
7	Reserved - This bit is always set to 0.
6	Control Mark (CM) - This flag is set to 1 if the controller encounters a sector that has a deleted-data address mark during execution of a Read Data command.
5	Data Error in Data Field (DD) - Set to 1 if the controller detects an error in the data.
4	Wrong Cylinder (WC) - This flag is related to ND (no data) bit. When the contents of C on the medium are different from that stored in the ID register, this flag is set.
3	Set to 0.
2	Set to 0.
1	Bad Cylinder (BC) - Related to ND; when the contents of C on the medium are different from that stored in the ID register, and the contents of C is hex FF, this flag is set to 1.
0	Missing Address Mark in Data Field (MD) - Set to 1 if the controller cannot find a data address mark or a deleted-data address mark when data is read from the medium.

Status Register 3

Figure 3-131 shows the bit definitions of status register ST 3.

Figure 3-131. Status Register 3 (ST 3)

Bit	Function
7	Reserved.
6	Write Protect (WP) - Status of the -write-protect signal from the diskette drive.
5	Reserved.
4	Track 0 (T0) - Status of the -track 0 signal from the diskette drive.
3	Reserved.
2	Head Address (HD) - Status of the -head 1 select signal to the diskette drive.
1, 0	Status of DS1, DS0 pins.

Signal Descriptions

The diskette drive controller interface signal sequences and timings are compatible with the industry standard 5.25-inch diskette interface. All interface signals are TTL compatible at the driver/receiver, both in the rise and fall times as well as the interface levels.

The following paragraphs describe the interface to the diskette drive.

Output Signals

All output signals to the diskette drive operate between 5V dc and ground and must meet the following specifications at the input to the drives:

- The inactive level is +2.0V dc minimum.
- The active level is +0.8V dc maximum.

The following are descriptions of the diskette drive controller output signals:

-High Density Select: Signal is active low when a drive's high density mode is selected. The polarity of this signal is consistent with 1.44MB, 3.5-inch drives. Signal is low for 2MB mode (diskettes are formatted for 1.44MB capacity). Signal is high for 1MB mode (diskettes are formatted for 720KB capacity).

On 5.25-inch drives, the drive requires a high for enabling the 1.2MB mode; a low, for the low density 360KB mode. The signal needs to be inverted when interfacing with 5.25-inch drives.

-Drive Select 0 - 1: The drive select signals enable or disable all drive interface signals except -motor enable. When a drive select signal is active, the drive is enabled. When it is inactive, all controlled inputs are ignored, and all drive outputs are disabled.

-Motor Enable 0 - 1: When this signal is activated, along with the proper drive select signals, the spindle starts to turn. There must be a 500-millisecond delay after -motor enable 0 or -motor enable 1 becomes active before a read, write, or seek operation. When inactive, this signal causes the spindle motor to decelerate and stop.

-Direction: When this signal is active, -step moves the heads toward the drive spindle. When this signal is inactive, -step moves the heads away from the drive spindle. This signal is stable for 1 microsecond before and 1 microsecond after the trailing edge of the step pulse.

Note: After a direction change, a 15-millisecond delay is required before the next step pulse.

-Step: A 1-microsecond active pulse of this signal causes the read/write heads to move one track. The state of -direction at the trailing edge of -step determines the direction of motion.

Note: Before a read or write operation, a 15-millisecond seek settle time must be allowed.

-Write Data: A 250-nanosecond (minimum) pulse of this signal causes a bit to be written if -write enable is active. Data written on the diskette will have 125-nanosecond write precompensation.

-Write Enable: When active, this signal enables the write current circuits, and -write data controls the writing of information. Motor start and head settle times must be observed before this line becomes active.

-Head 1 Select: Making this signal active selects the upper head; otherwise, the lower head is selected.

Input Signals

All inputs from the drive can sink 4.0 mA at the active (low) level and must meet the following specifications at the drive output:

- The inactive level is +3.7V dc minimum.
- The active level is +0.4V dc maximum.

Note: The controller interfaces directly with the outputs of 3.5-inch drives with tri-state, totem pole outputs. The IBM PS/1 computer 5.25-inch diskette drive unit contains buffer circuits to isolate the 5.25-inch interface from the 3.5-inch interface.

-Index: When the drive senses the index, it generates an active pulse of at least 1 millisecond on this line. This signal is gated to the interface only when the drive is selected.

-Track 0: This signal is active when the read/write head is on track 0. Track 0 is determined by a sensor, not a track counter.

The 3.5-inch drive is able to seek to track 0 under control of the system even if there is no diskette inserted at the time. This is required so the system software can determine how many drives are attached to the system.

Software selects each drive and attempts to recalibrate that drive to track 0. The track 0 signal is used to determine whether or not each drive is installed in the system.

-Write Protect: When active, this signal indicates that a diskette is write-protected.

-Read Data: Each bit detected provides a 250-nanosecond active pulse on this line for the 250,000-bit rate or a 125-nanosecond pulse for the 500,000-bit rate.

-Drive Type 1: When signal is low and drive is selected, a 1.44MB drive is assumed. If high when selected, either a 5.25-inch drive is present or no drive is installed.

-Diskette Change: This signal is active at power-on and latched inactive when a diskette is present, the drive is selected, and a step pulse occurs. This signal is activated when the diskette is removed from the drive. The presence of a diskette is determined by a media sensor.

The signal is used in detection of the drive type. If "drive type 1" signal is not active (not a 3.5-inch drive), then a 1.2MB 5.25-inch drive is assumed if the diskette change signal is active at power-on. Otherwise, a 360KB, 5.25-inch drive is assumed if Track 0 is active after a recalibration.

-Drive 2 Installed: This signal determines the presence of a second drive in the system. The signal is derived by the cable configuration which has the signal open to drive A and connected to drive B. If drive B is present, the signal is grounded. If high (wire open), BIOS assumes that no drive is present in position B. The signal is connected to the Intel 82077AA controller.

Power Sequencing

The write gate signal is turned off and kept off before power is turned on or off. The read/write heads return to track 0 when the system power is turned on.

Connector

The system board has a single 2-by-17-pin connector for the attachment of one or two diskette drives. Signals and data are transmitted to and from the drives through a 34-wire cable.

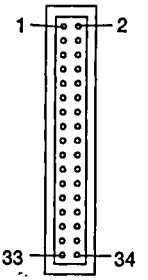


Figure 3-132 shows the voltages and signals assigned to the diskette drive connector.

Figure 3-132. Diskette Drive Controller Connector Voltage and Signal Assignments

Pin No.	I/O	Signal Name	Pin No.	I/O	Signal Name
1	I	-Drive 2 Installed	2	O	-High Density Select
3	NA	+5V dc	4	I	-Drive Type 1
5	NA	Signal Ground	6	NA	+12V dc
7	NA	Signal Ground	8	I	-Index
9	I	Reserved	10	O	-Motor Enable 1
11	NA	Signal Ground	12	O	-Drive Select 0
13	NA	Signal Ground	14	O	-Drive Select 1
15	NA	Signal Ground	16	O	-Motor Enable 0
17	NA	Signal Ground	18	O	-Direction
19	NA	Signal Ground	20	O	-Step
21	NA	Signal Ground	22	O	-Write Data
23	NA	Signal Ground	24	O	-Write Enable
25	NA	Signal Ground	26	I	-Track 0
27	NA	Signal Ground	28	I	-Write Protect
29	NA	Signal Ground	30	I	-Read Data
31	NA	Signal Ground	32	O	-Head 1 Select
33	NA	Signal Ground	34	I	-Diskette Change

Serial Port Controller

The serial port is controlled by a VL16C451A serial communications controller. The controller provides a TTL level interface which connects to a 2400 bps internal modem card or to a RS-232C interface card. It is programmable and supports asynchronous communications. The controller automatically adds and removes start, stop, and parity bits. A programmable baud-rate generator allows operation from 50 baud to 19,200 baud. The port supports 5-, 6-, 7- and 8-bit characters with 1, 1.5, or 2 stop bits. A prioritized interrupt system controls transmit, receive, error, and line status as well as data-set interrupts.

The VL16C451A controller is functionally compatible to the NS16450 controller. To programs, the VL16C451A appears to be identical to the serial portion of the IBM Personal Computer AT Serial/Parallel Adapter.

The serial port controller provides the following functions:

- Full double buffering in the character mode, eliminating the need for precise synchronization
- False-start bit detection
- Line-break generation and detection
- Modem control functions:
  - Clear to send (CTS)
  - Request to send (RTS)
  - Data set ready (DSR)
  - Data terminal ready (DTR)
  - Ring indicator (RI)
  - Data carrier detect (DCD).

Figure 3-133 on page 3-136 is a block diagram of the serial port controller.

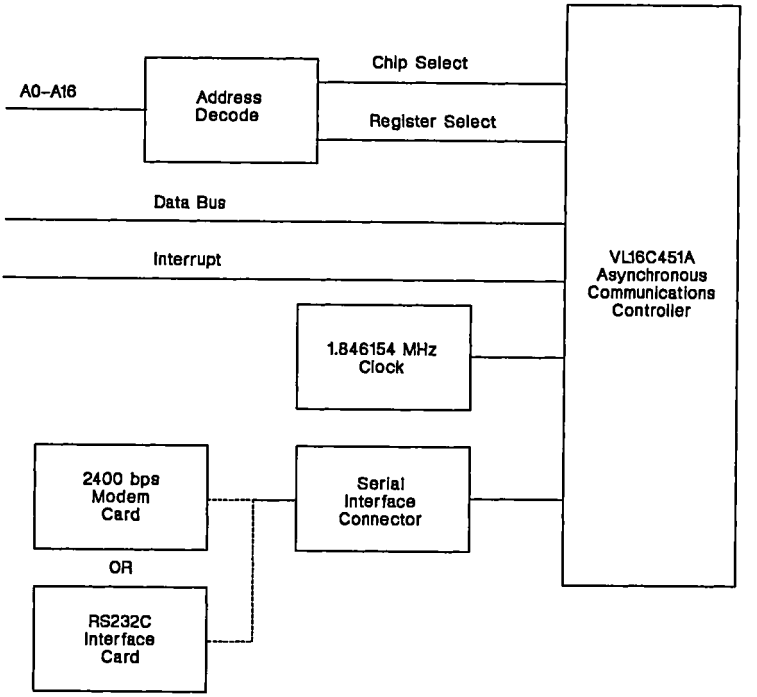


Figure 3-133. Serial Port Block Diagram

Communications Application

The serial output port is addressed as serial output port 1 (Serial 1) and interrupts as level 4 (IRQ4). To allow the controller to send interrupts to the interrupt controller, bit 3 of the Modem Control register must be set to 1. At this point, any interrupts allowed by the Interrupt Enable register will cause an interrupt.

Figure 3-134 shows the serial port data format.

Mark-ing	Start Bit	D0	D1	D2	D3	D4	D5	D6	D7	Parity Bit	Stop Bit

Figure 3-134. Serial Port, Data Format

Data bit 0 is the first bit to be sent or received. The controller automatically inserts the start bit, the correct parity bit (if programmed to do so), and the stop bits (1, 1.5, or 2, depending on the command in the Line Control register).

Programmable Baud-Rate Generator

The controller has a programmable baud-rate generator that can divide the clock input (1.846154 MHz) by any divisor from 1 to 65,535. The output frequency of the baud-rate generator is the baud rate multiplied by 16. Two 8-bit latches store the divisor in a 16-bit binary format. These divisor latches must be loaded during setup to ensure desired operation of the baud-rate generator. When either of the divisor latches is loaded, a 16-bit baud counter is immediately loaded. This procedure prevents long counts on the first load.

Registers

The controller has a number of accessible registers. The system programmer may gain access to or control any of the controller registers through the system microprocessor. These registers are used to control the controller operations and to transmit and receive data.

This section contains descriptions of the following registers:

Figure 3-135. Serial Port Register Addresses

DLAB State *	Port Address (hex)	R/W	Register
0	03F8	W	Transmitter Holding Register
0	03F8	R	Receiver Buffer Register
1	03F8	R/W	Divisor Latch, LSB
1	03F9	R/W	Divisor Latch, MSB
0	03F9	R/W	Interrupt Enable Register
X	03FA	R	Interrupt Identification Register
X	03FB	R/W	Line Control Register
X	03FC	R/W	Modem Control Register
X	03FD	R	Line Status Register
X	03FE	R	Modem Status Register
X	03FF	R/W	Scratch Register

* The DLAB state is controlled by bit 7 of the Line Control register.

Transmitter Holding Register

This register contains the character to be sent. Bit 0 is the least-significant bit and the first bit sent serially.

Receiver Buffer Register

This register contains the received character. Bit 0 is the least-significant bit and the first bit received serially.

Divisor Latch Register (LSB and MSB)

The Divisor Latch registers are used to program the baud rate generator. The values in these two registers form the divisor of the clock input (1.846154 MHz), which establishes the desired baud rate.

Figure 3-136 shows the use of the baud-rate generator with a frequency of (1.846154 MHz), For baud rates of 19,200 and below, the error obtained is minimal.

Note: In no case should the baud rate be greater than 19,200 baud with the RS-232C interface card or greater than 2,400 baud with the 2400 bps modem card.

Figure 3-136. Baud Rates at 1.846154 MHz

Desired Baud Rate	Standard Divisor Used to Generate 16x Clock (Decimal)	(Hex)	Percent of Error Difference Between Desired and Actual
50	2304	0900	0.160
75	1536	0600	0.160
110	1047	0417	0.186
134.5	857	0359	0.102
150	768	0300	0.160
300	384	0180	0.160
600	192	00C0	0.160
1200	96	0060	0.160
1800	64	0040	0.160
2000	58	003A	-0.531
2400	48	0030	0.160
3600	32	0020	0.160
4800	24	0018	0.160
7200	16	0010	0.160
9600	12	000C	0.160
19200	6	0006	0.160

Interrupt Enable Register (IER)

This 8-bit register allows the four types of controller interrupts to separately activate the chip-interrupt output signal. The interrupt system can be totally disabled by clearing bits 0 through 3 of the Interrupt Enable register. Similarly, by setting the appropriate bits of this register to 1, selected interrupts can be enabled. Disabling the interrupts inhibits the chip-interrupt output signal from the controller. All other system functions operate normally, including the setting of the Line Status and Modem Status registers.

Figure 3-137. Interrupt Enable Register (Hex 3F9)

Bit	Function
7 - 4	Reserved = 0
3	Modem Status Interrupt
2	Receiver Line Status Interrupt
1	Transmitter Holding Register Empty Interrupt
0	Received Data Available Interrupt

Bits 7 - 4 Reserved. Bits 7 through 4 are always cleared to 0.

Bit 3 When bit 3 is set to 1, the modem status interrupt is enabled.

Bit 2 When bit 2 is set to 1, the receiver line status interrupt is enabled.

Bit 1 When bit 1 is set to 1, the transmitter holding register empty interrupt is enabled.

Bit 0 When bit 0 is set to 1, the received data available interrupt is enabled.

Interrupt Identification Register

In order to minimize programming overhead during data character transfers, the controller prioritizes interrupts into four levels:

- Priority 1 - Receiver line status
- Priority 2 - Received data available
- Priority 3 - Transmitter holding register empty
- Priority 4 - Modem status.

The Interrupt Identification register stores information about a pending interrupt. When this register is addressed, the pending interrupt with the highest priority is held and no other interrupts are acknowledged until the system microprocessor services that interrupt.

Figure 3-138. Interrupt Identification Register (Hex 3FA)

Bit	Function
7 - 3	Reserved = 0
2	Interrupt ID, Bit 1
1	Interrupt ID, Bit 0
0	Interrupt Pending = 0

Bits 7 - 3 Reserved. Bits 7 through 3 are always set to 0.

Bits 2, 1 Bits 2 and 1 identify the pending interrupt with the highest priority, as shown in Figure 3-139.

Bit 0 When bit 0 is set to 1, no interrupt is pending, and polling (if used) continues. When bit 0 is set to 0, an interrupt is pending, and the contents of this register can be used as a pointer to the appropriate interrupt service routine.

Bit 0 can be used in either hard-wired, prioritized, or polled conditions to indicate that an interrupt is pending.

Bits 2 - 0 select interrupt control functions as shown in Figure 3-139.

Figure 3-139. Interrupt Control Functions

Bits 2 1 0	Priority	Type	Cause	Interrupt Reset Control
0 0 1	-	None	None	-
1 1 0	Highest	Receiver Line Status	Overrun, Parity, or Framing Error or Break Interrupt	Read the Line Status Register
1 0 0	Second	Received Data Available	Data in Receiver Buffer	Read the Receiver Buffer Register
0 1 0	Third	Transmitter Holding Register Empty	Transmitter Holding Register is Empty	Read Interrupt Identification Register or Write to Transmitter Holding Register
0 0 0	Fourth	Modem Status	Change in Signal Status from modem	Read the Modem Status Register

Line Control Register

This register programs the format of asynchronous communications.

Figure 3-140. Line Control Register (Hex 3FB)

Bit	Function
7	Divisor Latch Access Bit
6	Set Break
5	Stick Parity
4	Even Parity Select
3	Parity Enable
2	Number of Stop Bits
1	Word Length Select, Bit 1
0	Word Length Select, Bit 0

Bit 7 When bit 7 is set to 1, access is gained to the divisor latches of the baud-rate generator during a read or write operation. When bit 7 is set to 0, access is gained to the Receiver Buffer, Transmitter Holding, or Interrupt Enable registers.

Bit 6 When bit 6 is set to 1, set break is enabled, serial output is forced to the spacing state and remains there regardless of other transmitter activity. When bit 6 is set to 0, set break is disabled.

Bit 5 When bits 5 through 3 are set to 1's, the parity bit is sent and checked as a logical 0. When bits 5 and 3 are both set to 1's, and bit 4 is set to 0, the parity bit is sent and checked as a logical 1.

Bit 4 When bits 4 and 3 are set to 1's, an even number of logical 1's are transmitted and checked in the data word bits and parity bit. When bit 4 is set to 0, and bit 3 is set to 1, an odd number of logical 1's are transmitted and checked in the data word bits and parity bit.

Bit 3 When bit 3 is set to 1, a parity bit is generated (transmit data) or checked (receive data) between the last data word bit and stop bit. (The parity bit is used to produce an even or odd number of 1's when the data word bits and the parity bit are summed.)

Bit 2 Bits 2 through 0 specify the number of stop bits in each serial character that is sent or received as shown in Figure 3-141.

Figure 3-141. Stop Bits

Bit 2	Word Length *	Number of Stop Bits
0	N/A	1
1	5 Bits	1-1/2
1	6 Bits	2
1	7 Bits	2
1	8 Bits	2

* Word length is specified by bits 1 and 0 in this register.

Bits 1, 0 Bits 1 and 0 specify the number of bits in each serial character that is sent or received. Word length is selected as shown in Figure 3-142.

Figure 3-142. Word Length

Bits 1 0	Word Length
0 0	5 Bits
0 1	6 Bits
1 0	7 Bits
1 1	8 Bits

Modem Control Register

This register controls the data exchange with the modem, data set, or peripheral device emulating a modem.

Figure 3-143. Modem Control Register (Hex 3FC)

Bit	Function
7 - 5	Reserved = 0
4	Loop Test
3	Out 2
2	Out 1
1	Request to Send (RTS)
0	Data Terminal Ready (DTR)

Bits 7 - 5 Reserved. Bits 7 through 5 are always set to 0.

Bit 4 Bit 4 provides a loopback feature for diagnostic testing of the serial port. When bit 4 is set to 1:

- Transmitter serial output is set to the marking state.
- Receiver serial input is disconnected.
- Output of the Transmitter Shift register is *looped back* to the Receiver Shift register input.

  Note: The Transmitter and Receiver Shift registers are not accessible VL16C451 registers.

- The modem control inputs (CTS, DSR, DCD, and RI) are disconnected.
- The modem control outputs (DTR, RTS, OUT 1, and OUT 2) are internally connected to the four modem control inputs.
- The modem control output pins are forced inactive.

When the serial port is in the diagnostic mode, transmitted data is immediately received. This feature allows the system microprocessor to verify the transmit-data and receive-data paths of the serial port.

When the serial port is in the diagnostic mode, the receiver and transmitter interrupts are fully operational. The modem control interrupts are also operational, but their sources are the lower 4 bits of the Modem Control register instead of the four modem control input signals. The interrupts are still controlled by the Interrupt Enable register.

Bit 3 Bit 3 controls the OUT 2 signal which is an auxiliary user-programmed interrupt enable signal. OUT 2 is used to control the interrupt signal to the channel. When bit 3 is set to 1, the interrupt is enabled. When bit 3 is set to 0, the interrupt is disabled.

Bit 2 Bit 2 controls the OUT 1 signal which is an auxiliary user-programmed output signal. When bit 2 is set to 1, OUT 1 is forced active. When bit 2 is set to 0, OUT 1 is forced inactive.

Bit 1 Bit 1 controls the RTS modem control output signal. When bit 1 is set to 1, RTS is forced active. When bit 1 is set to 0, RTS is forced inactive.

Bit 0 Bit 0 controls the DTR modem control output signal. When bit 0 is set to 1, DTR is forced active. When bit 0 is set to 0, DTR is forced inactive.

Line Status Register

This register provides the system microprocessor with status information about the data transfer.

Figure 3-144. Line Status Register (Hex 3FD)

Bit	Function
7	Reserved = 0
6	Transmitter Shift Register Empty (TSRE)
5	Transmitter Holding Register Empty (THRE)
4	Break Interrupt (BI)
3	Framing Error (FE)
2	Parity Error (PE)
1	Overrun Error (OR)
0	Data Ready (DR)

Bit 7 Reserved. Bit 7 is always set to 0.

Bit 6 Bit 6 is set to 1 when both the Transmitter Holding register and the Transmitter Shift register are empty. Bit 6 is set to 0 when either the Transmitter Holding register or the Transmitter Shift register contains a data character.

Bit 5 Bit 5 indicates that the serial port controller is ready to accept a new character for transmission. Bit 5 is set to 1 when a character is transferred from the Transmitter Holding register into the Transmitter Shift register. Bit 5 is set to 0 when the system microprocessor loads the Transmitter Holding register.

Bit 5 also causes the controller to issue an interrupt to the system microprocessor when bit 1 in the Interrupt Enable register is set to 1.

Bit 4 Bit 4 is set to 1 when the received data input is held in the spacing state for longer than a full word transmission time (that is, the total time of start bit + data bits + parity + stop bits).

Note: Bits 1 through 4 are the error conditions that produce a receiver line status interrupt whenever any of the corresponding conditions are detected.

Bit 3 Bit 3 is set to 1 when the stop bit, following the last data bit or parity bit, is at a spacing level. This indicates that the received character did not have a valid stop bit.

Bit 2 Bit 2 is set to 1 when a parity error is detected (the received character does not have the correct even or odd parity, as selected by the even-parity-select bit in the Line Control register). Bit 2 is set to 0 when the system microprocessor reads the contents of the Line Status register.

Bit 1 When bit 1 is set to 1, it indicates that data in the Receiver Buffer register was not read by the system microprocessor before the next character was transferred into the Receiver Buffer register, destroying the previous character. Bit 1 is set to 0 when the system microprocessor reads the contents of the Line Status register.

Bit 0 Bit 0 is set to 1 when a complete incoming character has been received and transferred into the Receiver Buffer register. Bit 0 is set to 0 when the system microprocessor reads the Receiver Buffer register.

Modem Status Register

This register provides the current state of the control lines from the modem (or external device) to the system microprocessor. In addition, bits 3 through 0 of this register provide change information. These 4 bits are set to 1 whenever a control input from the modem changes state. They are set to 0 whenever the system microprocessor reads this register.

Figure 3-145. Modem Status Register (Hex 3FE)

Bit	Function
7	Data Carrier Detect
6	Ring Indicate
5	Data Set Ready
4	Clear to Send
3	Delta Data Carrier Detect
2	Trailing Edge Ring Indicate
1	Delta Data Set Ready
0	Delta Clear to Send

Bit 7 Bit 7 is the inverted DTR modem control input signal. If bit 4 of the Modem Control register is set to 1, this bit is equivalent to bit 3 in the Modem Control register.

Bit 6 Bit 6 is the inverted RI modem control input signal. If bit 4 of the Modem Control register is set to 1, this bit is equivalent to bit 2 in the Modem Control register.

Bit 5 Bit 5 is the inverted DSR modem control input signal. If bit 4 of the Modem Control register is set to 1, this bit is equivalent to bit 0 in the Modem Control register.

Bit 4 Bit 4 is the inverted CTS modem control input signal. If bit 4 of the Modem Control register is set to 1, this bit is equivalent to bit 1 in the Modem Control register.

Bit 3 When bit 3 is set to 1, it indicates that DTR has changed state since the last time it was read by the system microprocessor.

Note: Whenever bit 0, 1, 2, or 3 is set to 1, a modem status interrupt is generated.

Bit 2 When bit 2 is set to 1, it indicates that RI has changed from an active condition to an inactive condition.

Bit 1 When bit 1 is set to 1, it indicates that DSR has changed state since the last time it was read by the system microprocessor.

Bit 0 When bit 0 is set to 1, it indicates that CTS has changed state since the last time it was read by the system microprocessor.

Scratch Register

This register does not control the serial port in any way. It can be used by the system microprocessor to temporarily hold data.

Serial Port Controller Programming Considerations

The serial port uses the VL16C451A serial communications controller. The VL16C451A is functionally compatible with the NS16450 and appears identical to software as the IBM Personal Computer AT serial port on the serial/parallel adapter. See "Hardware Interrupts" on page 12-6 for additional programming considerations.

The serial port must be configured as Serial 1 (COM1:) using the system configuration utilities.

Signal Descriptions

Modem Control Input Signals

The following are system board input signals from the modem or RS-232C interface card to the controller. Bits 7 through 4 in the Modem Status register indicate the condition of these signals. Bits 3 through 0 in the Modem Status register monitor these signals to indicate when the modem changes state.

Clear to Send (CTS): When active, this signal indicates that the modem is ready for the serial port to transmit data.

Data Set Ready (DSR): When active, this signal indicates that the modem or data set is ready to establish the communications link and transfer data with the controller.

Ring Indicate (RI): When active, this signal indicates that the modem or data set detected a telephone ringing signal.

Received Level Signal Detect (RLSD): When active, this signal indicates that the modem or data set detected a data carrier.

Modem Control Output Signals

The following are controller output signals. They are all set inactive upon a master reset operation. These signals are controlled by bits 3 through 0 in the Modem Control register.

Data Terminal Ready (DTR): When active, this signal informs the modem or data set that the controller is ready to communicate.

Request to Send (RTS): When active, this signal informs the modem or data set that the controller is ready to send data.

Output 1 (OUT 1): This signal is pulled high.

Output 2 (OUT 2): User-programmed output. This signal controls interrupts to the system.

System Board Voltage Interchange

The system board provides a serial interface connector with TTL level signals for connection to either a 2400 bps modem card or to a RS-232C interface card. Included for the connector are +5V dc, +12V dc, and −12V dc.

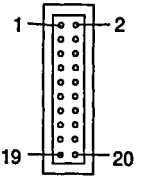


Figure 3-146. Serial Interface Connector

See Connector J2 on page 9-5 for signal definition.

Parallel Port Controller

The parallel port allows the attachment of devices that transfer 8 bits of parallel data at standard TTL levels. It has a 25-pin, D-shell connector. This port may be addressed as parallel port 1, 2, or 3.

The parallel port is compatible with previous IBM Personal Computer parallel port implementations. The primary function of the parallel port is to attach a printer with a parallel interface to the system. The parallel port has an Extended mode that allows support of bidirectional input and output. The port also supports edge-triggered interrupts and a readable interrupt-pending status. The interrupt level is common with the Audio Card.

Figure 3-147 is a block diagram of the parallel port controller.

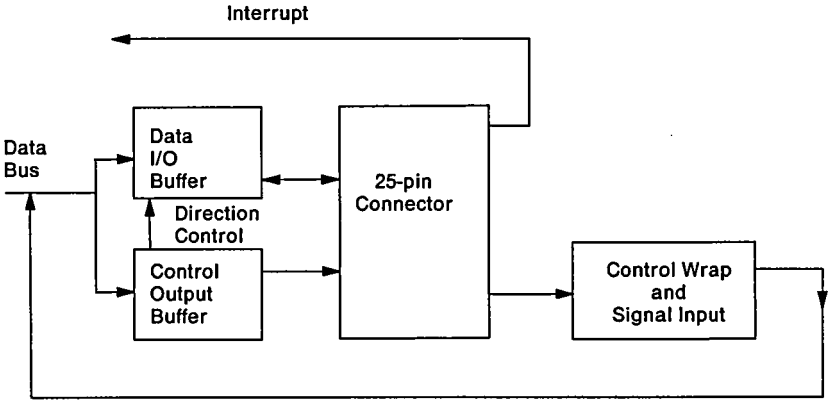


Figure 3-147. Parallel Port Controller Block Diagram

Parallel Port Programmable Option Select

The parallel port can be configured to three different address spaces previously used in IBM Personal Computer products. These addresses are selected by placing the system board in Setup; clear bit 7 of port hex 94. Perform an I/O write to port hex 0102. Bits 6 and 5 in port hex 0102 are used to select the address spaces shown in Figure 3-148.

When setup is complete, be sure to set bit 7 of port hex 94.

Figure 3-148. Parallel Port Configuration

Bit 6	Bit 5	Function
0	0	Parallel 1 LPT1:
0	1	Parallel 2 LPT2:
1	0	Parallel 3 LPT3:

The address assignments for each configuration are shown in Figure 3-149.

Figure 3-149. Parallel Port Address Assignments

Port	Data Address (hex)	Status Address (hex)	Parallel Control Address (hex)
Parallel 1	03BC	03BD	03BE
Parallel 2	0378	0379	037A
Parallel 3	0278	0279	027A

Parallel Port Extended Mode

The Extended mode option of the parallel port is selected through Programmable Option Select. With the system board in Setup, the Extended mode is selected by writing a 0 to bit 7 of I/O address hex 0102. The Extended mode makes the parallel port an 8-bit parallel and bidirectional interface. Figure 3-150 shows the possible configurations for the parallel port in the Extended mode.

Figure 3-150. Parallel Port Extended Mode Configurations

Port Mode	Port Direction	POS Mode Bit	Parallel Control Direction Bit	System Reset
Extended	Write	0	0	1
Extended	Write	0	0	0
Extended	Read	0	1	0
Compatible	Write	1	NA	0

Parallel Port Controller Programming Considerations

The following are some considerations for programming the parallel port controller.

The interface responds to five I/O instructions: two output and three input. In the Compatible mode, the output instructions transfer data into two latches whose outputs are presented on the pins of the D-shell connector. In the Extended mode, the 8-bit data latch output to the D-shell connector is controlled by bit 5 in the Parallel Control port.

In the Compatible mode, two of the three input instructions allow the processor to read back the contents of the two latches. In the Extended mode, the read-back of the 8-bit data in the Data Address is controlled by bit 5 in the Parallel Control port. The third input instruction allows the system microprocessor to read the real-time status of a group of pins on the connector.

The Extended mode can be used by externally attached equipment.

During POST, the parallel port is configured as an output port. POST status information is written to this port during the power-on initialization as well as initialization after a reset from the keyboard (Ctrl, Alt, Delete).

The following is a detailed description of each interface port instruction. For specific signal timing parameters, refer to the specifications for the equipment connected to the parallel port connector.

Data Address Port

The Data Address port is the 8-bit data port for both the Compatible and Extended modes. For the Compatible mode, a write operation to this port immediately presents data to the connector pins. A read operation from this port in the Compatible mode produces the data that was last written to it.

In the Extended mode, a write operation to this port latches the data but it is only presented to the connector pins if the direction bit was set to write in the Parallel Control port. A read operation in the Extended mode produces either:

- The data previously written if the direction bit in the Parallel Control port is set to write.

- The data on the connector pins from another device if the direction bit is set to read.

Status Port

The Status port is a read-only port in either mode. A read operation to this port presents the system microprocessor with the interrupt pending status of the interface as well as the real-time status of the connector pins as shown in Figure 3-151. An interrupt is pending when the interrupt status bit is set to 0.

Figure 3-151. Status Port

Port Bit	Port Data
7	-Busy
6	-Acknowledge (-ACK)
5	Paper End (PE)
4	Select (SLCT)
3	-Error
2	-IRQ Status
1, 0	Reserved

Bit 7 Bit 7 represents the state of the -BUSY signal. When this signal is active, the printer is busy and cannot accept data.

Bit 6 Bit 6 represents the current state of the printer -ACK signal. When bit 6 is set to 0, the printer has received a character and is ready to accept another.

Bit 5 Bit 5 represents the current state of the printer PE signal. When bit 5 is set to 1, the printer has detected the end of the paper.

Bit 4 Bit 4 represents the current state of the SLCT signal. When bit 4 is set to 1, the printer has been selected.

Bit 3 Bit 3 represents the current state of the printer -ERROR signal. When bit 3 is set to 0, the printer has encountered an error condition.

Bit 2 When bit 2 is set to 0, the printer has acknowledged the previous transfer using the -ACK signal.

Bits 1, 0 Reserved.

Parallel Control Port

The Parallel Control port is a read/write port. A write operation to this port latches the 6 least-significant data bits of the bus. The sixth bit corresponds to the direction control bit and is only applicable in the Extended mode. The remaining 5 bits are compatible with previous implementations as shown in Figure 3-152. A read operation to the Parallel Control port sends the system microprocessor the data that was last written to it, with the exception of the write-only direction bit.

Figure 3-152. Parallel Control Port

Port Bit	Port Data
7, 6	Reserved
5	Direction
4	IRQ EN
3	Pin 17 (SLCT IN)
2	Pin 16 (-INIT)
1	Pin 14 (AUTO FD XT)
0	Pin 1 (STROBE)

Bits 7, 6 Reserved.

Bit 5 Bit 5 controls the direction of the data port. For more information on the use of this bit, see Figure 3-150 on page 3-150. This is a write-only bit.

Bit 4 When bit 4 is set to 1, an interrupt occurs when the -ACK signal changes from active to inactive. This bit enables the parallel port interrupt.

Bit 3 Bit 3 controls the SLCT IN signal. When bit 3 is set to 1, the printer is selected.

Bit 2 Bit 2 controls the -INIT printer signal. When bit 2 is set to 0, the printer starts.

Bit 1 Bit 1 controls the AUTO FD XT signal. When bit 1 is set to 1, the printer will automatically line feed after each line is printed.

Bit 0 Bit 0 controls the STROBE signal to the printer. When bit 0 is set to 1, data is clocked into the printer.

Parallel Port Timing

The timing for the parallel port depends on the timing of the devices connected to the port. Figure 3-153 shows the sequence for typical parallel port signal timing.

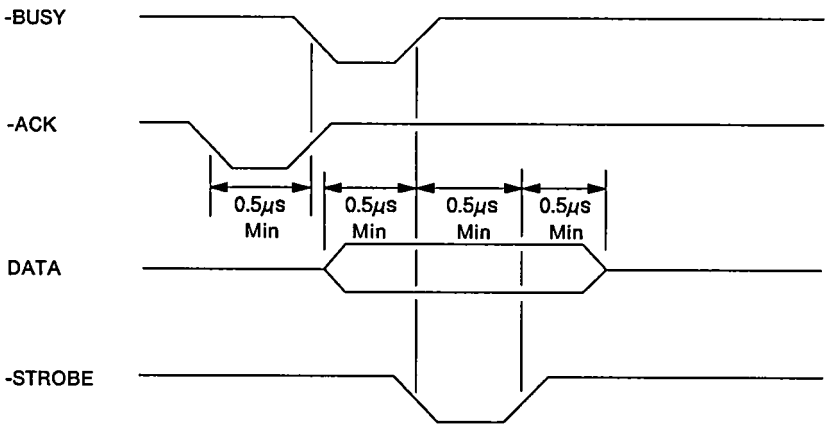


Figure 3-153. Parallel Port Timing Sequence

Signal Descriptions

Figure 3-154 and Figure 3-155 list characteristics of the data, interrupt, and control signals.

Figure 3-154. Data and Interrupt Signals		
Sink Current	12 mA	Maximum
Source Current	2 mA	Maximum
High-level Output Voltage	2.4V dc	Minimum
Low-level Output Voltage	0.5V dc	Maximum

Pins 1, 14, 16, and 17 are driven by open collector drivers pulled to 5 V dc through 4.7 K ohm resistors.

Figure 3-155. Control Signals		
Sink Current	10 mA	Maximum
Source Current	0.2 mA	Maximum
High-level Output Voltage	5.0V dc minus pullup	Minimum
Low-level Output Voltage	0.5V dc	Maximum

Connector

The parallel port connector is a standard 25-pin D-shell connector. The D0 - D7 lines on the connector are driven by drivers capable of sourcing 2 mA and sinking 12 mA.

Figure 3-156 show the voltages and signals assigned to the parallel port connector.

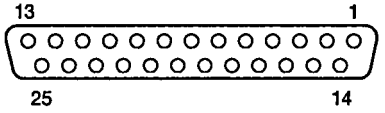


Figure 3-156. Parallel Port Connector Signal and Voltage Assignments

Pin No.	I/O	Signal Name	Pin No.	I/O	Signal Name
1	I/O	-STROBE	14	O	-AUTO FD XT
2	I/O	Data Bit 0	15	I	-ERROR
3	I/O	Data Bit 1	16	O	-INIT
4	I/O	Data Bit 2	17	O	-SLCT IN
5	I/O	Data Bit 3	18	NA	Ground
6	I/O	Data Bit 4	19	NA	Ground
7	I/O	Data Bit 5	20	NA	Ground
8	I/O	Data Bit 6	21	NA	Ground
9	I/O	Data Bit 7	22	NA	Ground
10	I	-ACK	23	NA	Ground
11	I	BUSY	24	NA	Ground
12	I	PE	25	NA	Ground
13	I	SLCT			

Memory

The system has the following types of memory:

- Read-Only Memory (ROM)
- Random Access Memory (RAM)
- Complementary Metal Oxide Semiconductor RAM (CMOS RAM).

Read-Only Memory (ROM)

The system board ROM subsystem consists of either two 128K-by-8-bit modules or four 128K-by-8-bit modules arranged in 16 bit words. 512KB of ROM are addressed from F80000 to FFFFFF. Addresses FE0000 to FFFFFF are also addressed as 0E0000 to 0FFFFF.

Random Access Memory (RAM)

The system board RAM starts at address hex 000000 of the 16MB address space. 512KB of RAM are soldered on the system board. An additional 512KB may be added by installing a memory card into connector on the system board. The RAM operates with one wait state. The system board memory is not parity checked. Memory-refresh requests occur once every 15 microseconds.

RAM can be disabled in 128KB blocks, from 0KB to 640KB (five blocks). If I/O memory exists in any block, the RAM will be automatically disabled by the POST.

The 512KB memory card connects to a 2-by-20 card edge connector on the system board.

Figure 3-157 shows the pin definitions for the RAM module connectors.

Figure 3-157. Memory Option Connector Pin Assignments

Pin	I/O	Signal	Pin	I/O	Signal
1	I/O	Data Bit 0	2	I/O	Data Bit 1
3	I/O	Data Bit 2	4	I/O	Data Bit 3
5	N/A	Ground	6	I/O	Data Bit 4
7	I/O	Data Bit 5	8	I/O	Data Bit 6
9	I/O	Data Bit 7	10	N/A	Ground
11	I/O	Data Bit 8	12	I/O	Data Bit 9
13	I/O	Data Bit 10	14	I/O	Data Bit 11
15	N/A	Ground	16	I/O	Data Bit 12
17	I/O	Data Bit 13	18	I/O	Data Bit 14
19	I/O	Data Bit 15	20	N/A	Ground
21	N/A	+5V dc	22	I	-Second Bank
23	I	1Mb Chip Detect	24	O	-Row Address Strobe 1
25	O	-Row Address Strobe 0	26	O	-Write Strobe
27	O	-Column Address Strobe High	28	O	-Column Address Strobe Low
29	N/A	Ground	30	O	Address Bit 9
31	O	Address Bit 8	32	O	Address Bit 7
33	O	Address Bit 6	34	O	Address Bit 5
35	N/A	Ground	36	O	Address Bit 4
37	O	Address Bit 3	38	O	Address Bit 2
39	O	Address Bit 1	40	O	Address Bit 0

Figure 3-158 shows the mapping for the memory locations on the system board.

Figure 3-158. System Board Memory Map

Hex Range	Function
000000 to 09FFFF	640KB System Board RAM *
0A0000 to 0BFFFF	Video RAM
0C0000 to 0DFFFF	128KB I/O Expansion Adapters
0E0000 to 0FFFFF	128KB System Board ROM (also mapped in hex FE0000 to FFFFFF)
100000 to 15FFFF	384KB System Board RAM with 512KB RAM option
160000 to F7FFFF	I/O Channel Expansion Memory Addresses
F80000 to FFFFFF	512KB System Board ROM

* A 256-byte portion of this RAM is reserved as a BIOS data area. A 1K portion of this RAM is reserved as an extended BIOS data area. See *BIOS Interface Technical Reference for IBM PS/1™ Computer* for details.

Note: With the system board in setup, the system board memory can be disabled by writing port address hex 104 as shown in the following figure.

Figure 3-159. System Board Memory Enable

Bits	Function	Hex Range
7 - 5	Reserved	
4	Enable Fifth 128KB Bank	080000 - 09FFFF
3	Enable Fourth 128KB Bank	060000 - 07FFFF
2	Enable Third 128KB Bank	040000 - 05FFFF
1	Enable Second 128KB Bank	020000 - 03FFFF
0	Enable First 128KB Bank	000000 - 01FFFF

Complementary Metal Oxide Semiconductor RAM

The real-time clock and 64 bytes of nonvolatile RAM are contained on the real-time clock/complementary metal oxide semiconductor (RTC/CMOS) RAM chip. The internal clock circuitry uses 14 bytes of this memory, and the rest is allocated to configuration and system status information.

Figure 3-160 shows the RTC/CMOS RAM bytes and their addresses.

Figure 3-160. RTC/CMOS RAM Address Map

Address (hex)	Function
000 - 00D	Real-Time Clock Information
00E	Diagnostic Status Byte
00F	Shut Down Status Byte
010	Diskette Drive Byte
011	Fixed Disk Type Byte
012 - 014	Reserved
015 - 016	Low and High Base Memory Bytes
017 - 018	Low and High Memory Expansion Bytes
019 - 031	Reserved
032 - 033	Configuration CRC Bytes
034 - 036	Reserved
037	Date Century Byte
038 - 03F	Reserved

RTC/CMOS RAM I/O Operations

When performing I/O operations to the RTC/CMOS RAM addresses (see Figure 3-160 on page 3-158 and the following pages), interrupts should be inhibited to avoid having interrupt routines change the CMOS address register before data is read or written. Port hex 0070 should be left to point to status register D of the RTC/CMOS.

I/O operations to the RTC/CMOS RAM addresses require the following sequence:

1. OUT to port hex 0070 with the RTC/CMOS address to be written.

   Note: The NMI mask bit resides in port hex 0070. (See "Non-Maskable Interrupt" on page 2-28.)

2. JMP $+2 for I/O delay.
3. OUT to port hex 0071 with the data to be written.

Reading the RTC/CMOS RAM requires the following sequence:

1. OUT to port hex 0070 with the CMOS address to be read.
2. IN from port hex 0071, and the data read is returned in the AL register.

Warning: When writing port hex 0070, a read or write to port hex 0071 must be accessed immediately. Failure to do this can cause intermittent failures and unreliable operation of the RTC/CMOS RAM.

Real-Time Clock

Figure 3-161 shows bit definitions for the real-time clock bytes (addresses hex 000 - 00D).

Figure 3-161. Real-Time Clock (Addresses Hex 000 - 00D)

Address (hex)	Function	Byte Number
000	Seconds	0
001	Second Alarm	1
002	Minutes	2
003	Minute Alarm	3
004	Hours	4
005	Hour Alarm	5
006	Day Of Week	6
007	Date Of Month	7
008	Month	8
009	Year	9
00A	Status Register A	10
00B	Status Register B	11
00C	Status Register C	12
00D	Status Register D	13

Note: INT hex 1A is the BIOS call to read and set the time and date. It initializes registers A, B, C, and D.

Status Register A

Bit 7 Update in Progress (UIP)— When bit 7 is set to 1, it indicates that the time update cycle is in progress. When bit 7 is set to 0, it indicates that the current date and time are available to read. Reading the date or time when this bit is set to 1 may result in incorrect values.

Bits 6 - 4 22-Stage Divider (DV2 through DV0)— These three divider-selection bits identify which time-base frequency is being used. The system initializes the stage divider to 010, which selects a 32.768 KHz time base. This is the only stage divider supported by the system microprocessor for proper timekeeping.

Bits 3 - 0 Rate Selection Bits (RS3 through RS0)— These bits allow the selection of a divider output frequency. The system initializes the rate selection bits to 0110, which selects a 1.024 KHz square-wave output frequency and a 976.562 microsecond periodic interrupt rate.

Status Register B

Bit 7 Set— When bit 7 is set to 1, any update cycle in progress is aborted and the program can initialize the 14 time-bytes without any further updates occurring until a 0 is written to this bit. When bit 7 is set to 0, the cycle is updated normally by advancing the counts at one per second.

Bit 6 Periodic Interrupt Enable (PIE)— When bit 6 is set to 1, it enables the interrupt to occur at a rate specified by the rate and divider bits in register A. When bit 6 is set to 0, the interrupt is disabled. Bit 6 is a read/write bit that the system initializes to a 0.

Bit 5 Alarm Interrupt Enable (AIE)— When bit 5 is set to 1, the alarm interrupt is enabled. When bit 5 is set to 0, the alarm interrupt is disabled. The system initializes this bit to 0.

Bit 4 Update-Ended Interrupt Enabled (UIE)— When bit 4 is set to 1, the update-ended interrupt is enabled. When bit 4 is set to 0, the update-ended interrupt is disabled. The system initializes this bit to 0.

Bit 3 Square Wave Enabled (SQWE)— When bit 3 is set to 1, the square-wave frequency, as set by the rate selection bits in Status register A, is enabled. When bit 3 is set to 0, square wave is disabled. The system initializes this bit to 0.

Bit 2 Date Mode (DM)— When bit 2 is set to 1, the time and date are updated using a binary format. When bit 2 is set to 0, the time and date are updated using a binary-coded-decimal (BCD) format. The system initializes this bit to 0.

Bit 1 24/12— When bit 1 is set to 1, the hours byte is in 24-hour mode. When bit 1 is set to 0, the hours byte is in 12-hour mode. The system initializes this bit to 1.

Bit 0 Daylight Savings Enabled (DSE)— When bit 0 is set to 1, daylight savings time is enabled. When bit 0 is set to 0, daylight savings time is disabled (reverts to standard time). The system initializes this bit to 0.

Status Register C

Bits 7 - 4 IRQF, PF, AF, UF— These read-only flag bits are affected when the PIE, AIE, and UIE interrupts (bits 6 - 4) are enabled with Status Register B.

Bits 3 - 0 Reserved.

Status Register D

Bit 7 Valid RAM Bit (VRB)— This bit is read-only and indicates the condition of the contents of the CMOS RAM through the power sense pin. A low state of the power sense pin indicates the real-time clock has lost its power. When bit 7 is set to 1, there is power on the real-time clock. When bit 7 is set to 0, the real-time clock has lost power.

Bits 6 - 0 Reserved.

CMOS RAM Configuration

The following shows the bit definitions for the CMOS configuration bytes (addresses hex 00E - 03F).

Diagnostic Status Byte (Hex 00E)

Bit 7 Real-Time Clock Power— When bit 7 is set to 1, the real-time clock chip has lost power. When bit 7 is set to 0, the real-time clock chip has not lost power.

Bit 6 Configuration Record and Checksum Status— When bit 6 is set to 1, the checksum is incorrect. When bit 6 is set to 0, the checksum is correct. The checksum is used to verify the bytes from hex 10 to hex 33.

Bit 5 Incorrect Configuration— This is a check, at power-on time, of the equipment byte of the configuration record. When bit 5 is set to 1, the configuration information is incorrect. Power-on checks require at least one diskette drive installed (bit 0 of the equipment byte set to 1). When bit 5 is set to 0, the configuration information is correct.

Bit 4 Memory Size Miscompare— When bit 4 is set to 1, the power-on check determines that the memory size is different from the memory size in the configuration record. When bit 4 is set to 0, the memory size is the same.

Bit 3 Reserved

Bit 2 Time Status Indicator (POST validity check)— When bit 2 is set to 1, the time is invalid. When bit 2 is set to 0, the time is valid.

Bit 1 Reserved.

Bit 0 Reserved.

Shut Down Status Byte (Hex 00F)

The bits in this byte are defined by the power-on diagnostics.

Diskette Drive Type Byte (Hex 010)

Bits 7 - 4 Type of first diskette drive:

0000 = No drive present
0001 = Reserved
0010 = Reserved
0011 = Reserved
0100 = High-density diskette drive (1.44MB).

Note: 0101 through 1111 are reserved.

Bits 3 - 0 Type of second diskette drive installed:

0000 = No drive present
0001 = 360KB 5.25-inch diskette drive
0010 = 1.2MB 5.25-inch diskette drive
0011 = Reserved
0100 = High-density diskette drive (1.44MB).

Note: 0101 through 1111 are reserved.

Fixed Disk Type Byte (Hex 011)

This byte defines the type of fixed disk drive (drive C) installed. Hex 00 indicates a fixed disk drive is not present.

Reserved (Hex 012 to 014)

Base Memory Bytes (Hex 015 and 016)

These bytes define the amount of memory installed below the 640KB address space.

The hex value from these bytes represents the number of 1KB blocks of base memory. For example, hex 0280 is equal to 640KB. Byte hex 015 is the least-significant byte of the base memory size. Byte hex 016 is the most-significant byte of the base memory size.

Memory Expansion Bytes (Hex 017 and 018)

These bytes define the amount of memory installed above the 1MB address space.

The hex value from these bytes represents the number of 1KB blocks of expansion memory. For example, hex 0800 is equal to 2048KB. Byte hex 017 is the least-significant byte of the expansion memory size. Byte hex 018 is the most-significant byte of the expansion memory size.

Reserved (Hex 019 through 031)

Configuration CRC Bytes (Hex 032 and 033)

These bytes contain the cyclic-redundancy-check (CRC) data for bytes hex 010 through hex 031 of the 64-byte CMOS. Byte hex 032 is the most-significant byte of the configuration CRC. Byte hex 033 is the least-significant byte of the configuration CRC.

Reserved (Hex 034 through 036)

Date Century Byte (Hex 037)

This byte contains the BCD value for the century (BIOS interface to read and set).

Reserved (Hex 038 through 03F)

Miscellaneous System Ports

Ports hex 0061, 0070, and 0092 contain information that is used for system control.

System Control Port B (Hex 0061)

Port B is accessed by I/O read or write operations to I/O address hex 0061. Figure 3-162 shows the bit definitions.

Figure 3-162. System Control Port B, Write Operations

Bit	Function
7 - 4	Reserved
3	-Enable Channel Check
2	-Enable Parity Check
1	Speaker Data Enable
0	Timer 2 Gate to Speaker

Figure 3-163. System Control Port B, Read Operations

Bit	Function
7	Parity Check
6	Channel Check
5	Timer 2 Output
4	Toggles with each Refresh Request
3	Enable Channel Check
2	Enable Parity Check
1	Speaker Data Enable
0	Timer 2 Gate to Speaker

Bit 7 When bit 7 is set to 1, a parity check has occurred on a read operation.

Note: System board RAM does not support parity.

Bit 6 When bit 6 is set to 1, a channel check has occurred on a read operation.

Bit 5 Bit 5 returns the condition of timer/counter 2 output signal on a read operation.

Bit 4 Bit 4 toggles for each refresh request on a read operation.

Bit 3 When bit 3 is set to 0, channel check is enabled on a write. The result of the last write operation to this bit is returned on a read operation. The system initializes this bit to 1 during a power-on reset.

Bit 2 When bit 2 is set to 0, parity check is enabled on a write. The result of the last write operation to this bit is returned on a read operation. The system initializes this bit to 1 during a power-on reset.

Note: System board RAM does not support parity.

Bit 1 When bit 1 is set to 1, speaker data is enabled on a write. The result of the last write operation to this bit is returned on a read operation. The system initializes this bit to 0 during a power-on reset.

Bit 0 When bit 0 is set to 1, timer 2 gate is enabled on a write. The result of the last write operation to this bit is returned on a read operation. When bit 0 is set to 0, the gate is disabled on a write.

RTC/CMOS and NMI Mask (Hex 0070)

Figure 3-164. RTC/CMOS and NMI Mask

Bit	Function
7	Non-maskable Interrupt (NMI)
6	Reserved
5 - 0	RTC/CMOS RAM

Bit 7 When bit 7 is cleared to 0, NMI is enabled. When bit 7 is set to 1, NMI is masked off. This bit is set to 1 during a power-on reset. This bit is write-only. See "Interrupts" on page 2-28 for more information about the NMI.

Bit 6 Reserved.

Bits 5 - 0 See "RTC/CMOS RAM I/O Operations" on page 3-159.

Note: Port hex 0071 is used with port hex 0070 to read and write to the CMOS RAM and the NMI Mask register.

System Control Port A (Hex 0092)

Port hex 0092 supports alternate system microprocessor reset, PASS A20, and CMOS security lock. Figure 3-165 shows the bit definitions for port hex 0092.

Figure 3-165. System Control Port A

Bit	Function
7 - 4	Reserved
3	Security Lock Latch
2	Reserved = 0
1	Alternate Gate A20
0	Reserved

Bits 7 - 4 Reserved.

Bit 3 When bit 3 is set to 1, the 8-byte password is electrically locked in the secured area of CMOS. This read/write bit is set by POST, and it can only be cleared by turning the system power off and then turning the power on.

Note: System board and POST do not support password.

Bit 2 Reserved.

Bit 1 When bit 1 is set to 1, the A20 address bit is active when the microprocessor is in Real Address mode. When bit 1 is set to 0, A20 is inactive in Real Address mode. This read/write bit is set to 0 during a system reset.

Bit 0 Reserved.

Section 4. Power Supply

Description 4-3
Outputs 4-3
  Power Controller Coordination 4-3
  Output Protection 4-3
  Voltage Sequencing 4-4
  No-Load Operation 4-4
  Auto Restart 4-4
  Power Good Signal 4-4
Power Card Connectors 4-5
Power/Audio Cable Connector 4-6

Description

The power supply function is distributed among the elements of the system. The display regulates ac line voltage to a 30−40V dc bulk, which is input to the system and adapter card units. The system and adapter card units each contain power cards that regulate the bulk dc into the voltages required by the system logic.

The system and adapter card unit power controllers receive 30−40V dc input (36 V nominal) from the display, at 2.9A maximum. This voltage is supplied from the power/audio cable connector.

Outputs

The power supply provides three voltages: +5.1, +12, and -12V dc to the system unit and +5, +12, -5, and -12 to the I/O channel in the adapter card unit. "Adapter Card Channel" on page 2-5 provides additional information for voltage and power requirements.

Power Controller Coordination

In a system which has an adapter card unit, the system and adapter card unit power cards are linked by the open collector -FAULT signal (Active low). If either card experiences a fault condition (overcurrent, overvoltage), it latches its outputs off and asserts -FAULT, causing the other card to also latch off.

Output Protection

A short circuit placed on any dc output (between outputs or between an output and dc return) latches all dc outputs into a shut down state with no damage to the power supply.

If an overvoltage fault occurs (internal to the power supply), the supply latches all dc outputs into a shut down state with no damage to the power supply.

If either of these shut down states is entered, the power supply returns to normal operation only after the fault has been removed and the power switch has been turned off for at least three seconds.

Voltage Sequencing

At power-on, the output voltages track within 50 milliseconds of each other when measured at the 50% points.

No-Load Operation

The power supply is capable of operation with no load on the outputs. The outputs may regulate between the undervoltage and overvoltage limits.

Auto Restart

If the power supply outputs drop out of regulation due to an ac line outage, the power supply automatically restarts (generates output voltages) when ac power returns.

Power Good Signal

A power good signal indicates proper operation of the power supply and is active (high) during normal operation. The power good signal can sink 3 milliamps.

The power supply provides the power good signal to indicate proper operation of the power supply and to reset system logic. At power-on, this signal has a turn-on delay of at least 200 milliseconds but not greater than 600 milliseconds, after all voltages have reached minimum regulation levels. At power-off, this signal will go inactive within 100 milliseconds and be at a low level before the +5V dc output reaches the undervoltage sense level. If the power good signal goes inactive due to an ac line outage, it is regenerated as described above when ac power returns.

Power Card Connectors

Figure 4-1 shows the signals and voltages assigned to the power card connectors.

Figure 4-1. Power Card Connectors

System Unit Power Card		Adapter Card Unit Power Card	
Pin	**Signal**	**Pin**	**Signal**
1	Ground	1	Ground
2	Ground	2	Ground
3	+36 V	3	-FAULT
4	+36 V	4	+36 V
5	-12 V	5	+36 V
6	Ground	6	Ground
7	+12 V	7	Ground
8	+12 V	8	+5REF
9	-FAULT	9	+5 V
10	Ground	10	+5 V
11	+5 V	11	+5 V
12	+5 V	12	+5 V
13	+5 V	13	+5 V
14	+5 V	14	-12 V
15	Ground	15	-5 V
16	Power Good	16	Power Good
17	Ground	17	Ground

Power/Audio Cable Connector

Figure 4-2 shows the signals and voltage assigned to the power/audio cable connector for the display.

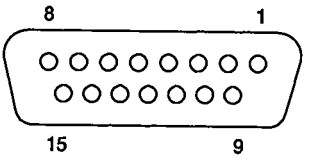


Figure 4-2. Power/Audio Cable Connector

Pin	Signal	Pin	Signal
1	+36 V		
2	+36 V	9	+36 V
3	Ground	10	Ground
4	Reserved	11	Ground
5	Reserved	12	Reserved
6	Audio	13	Reserved
7	Ground	14	Ground
8	Audio	15	Ground

Section 5. Keyboard

Description 5-3
Keyboard Charts 5-3
- Belgium (Dutch) Keyboard 5-6
- Canada (French) Keyboard 5-7
- Denmark Keyboard 5-8
- Finland and Sweden Keyboard 5-9
- France and Belgium (French) Keyboard 5-10
- Germany Keyboard 5-11
- Italy Keyboard 5-12
- Netherlands Keyboard 5-13
- Norway Keyboard 5-14
- Portugal Keyboard 5-15
- Spain Keyboard 5-16
- Switzerland (French) Keyboard 5-17
- Switzerland (German) Keyboard 5-18
- U.K. Keyboard 5-19
- U.S. and Canada (English) Keyboard 5-20

Sequential Key-Code Scanning 5-21
- Buffer 5-21
- Keys 5-21

Power-On Routine 5-22
- Power-On Reset (POR) 5-22
- Basic Assurance Test (BAT) 5-22

Commands from the System 5-23
Commands to the System 5-28
Scan Codes 5-29
- Set 1 Scan Code Tables 5-29
- Set 2 Scan Code Tables 5-32
- Set 3 Scan Code Tables 5-36

Clock and Data Signals 5-39
- Data Stream 5-39
- Data Output 5-40
- Data Input 5-41

Encode and Usage 5-42
Extended Functions 5-45
- Shift States 5-46

Special Handling 5-48
- System Reset 5-48
- Break 5-48
- Pause 5-48
- Print Screen 5-48
- System Request (SysRq) 5-49

Other Characteristics 5-49
Cables and Connectors 5-50
Specifications 5-51

Description

The IBM PS/1 computer keyboard has 101 keys for the U.S. and Canada (English) models. For other countries' models, it has 102 keys.

At system power-on, the keyboard monitors the signals on the "clock" and "data" lines and establishes its line protocol. A bidirectional serial interface in the keyboard converts the clock and data signals and sends this information to and from the keyboard through the keyboard cable.

Keyboard Charts

The 101-/102-key keyboard is shown in the following charts:

- Belgium (Dutch)
- Canada (French)
- Denmark
- Finland and Sweden
- France and Belgium (French)
- Germany
- Italy
- Netherlands
- Norway
- Portugal
- Spain
- Switzerland (French)
- Switzerland (German)
- U.K.
- U.S. and Canada (English).

The charts are shown in alphabetical order on the following pages. On the actual keyboard, nomenclature is on the top and the front of the keybuttons. In this manual, nomenclature for the front of the key is shown on the top of the key, in the lower right corner.

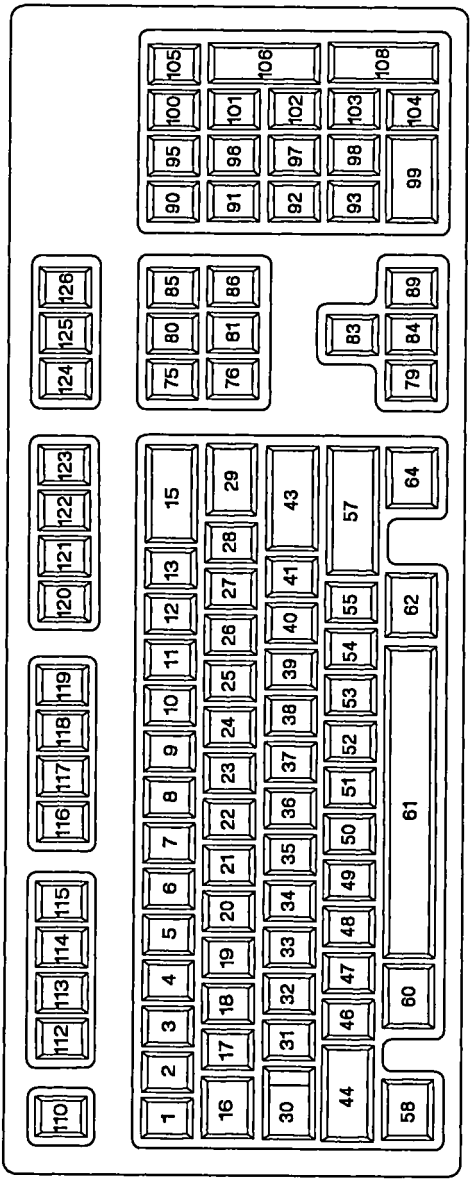


Figure 5-1. Key Numbering for the 101-Key Keyboard

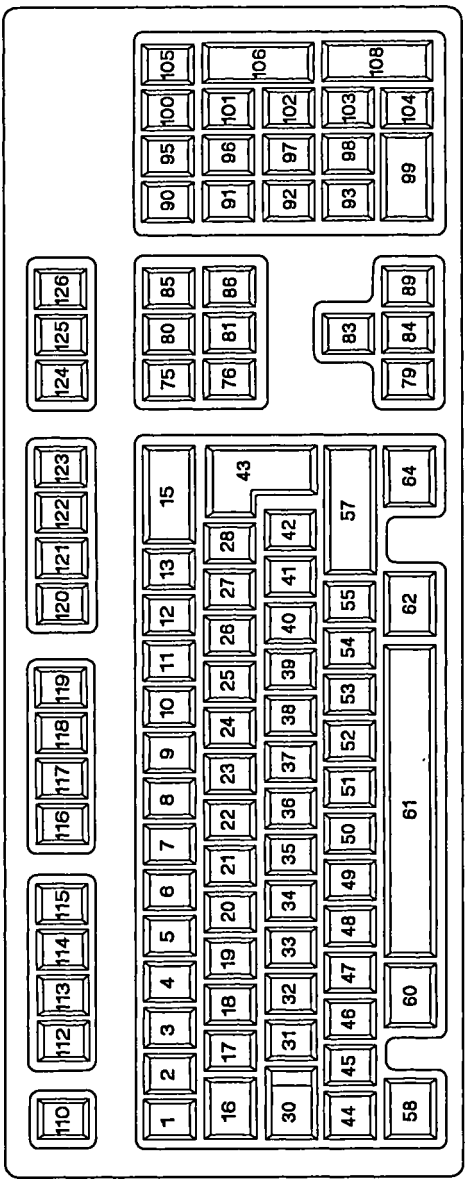
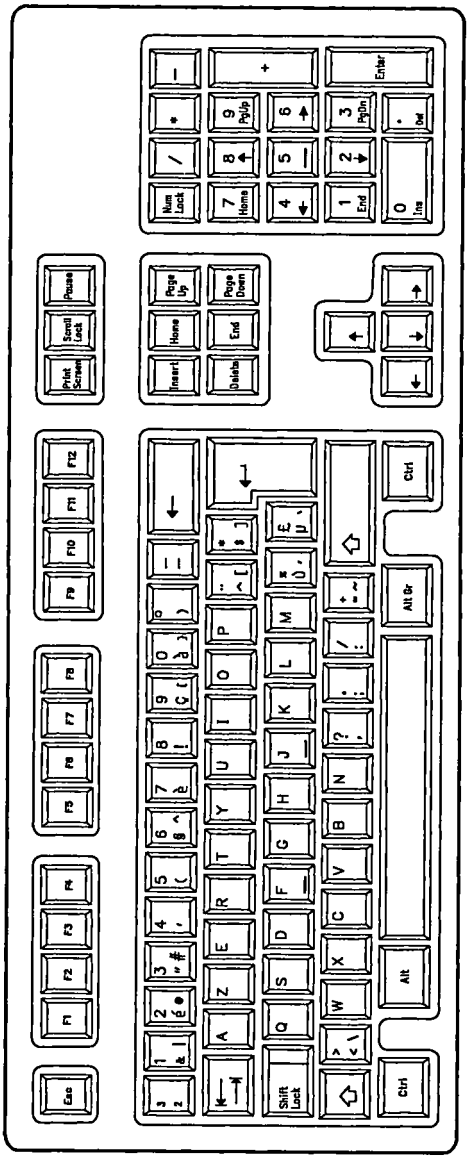
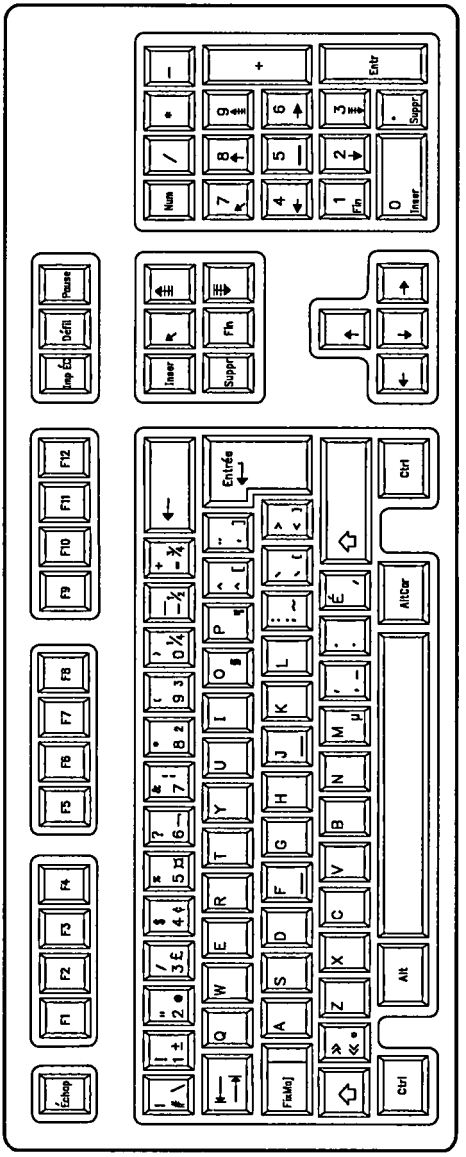


Figure 5-2. Key Numbering for the 102-Key Keyboard

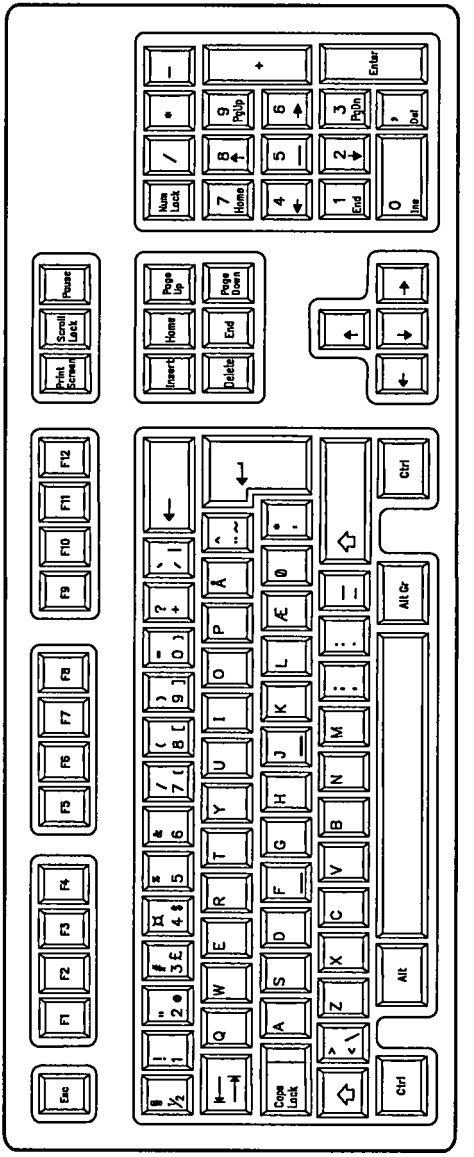
Belgium (Dutch) Keyboard



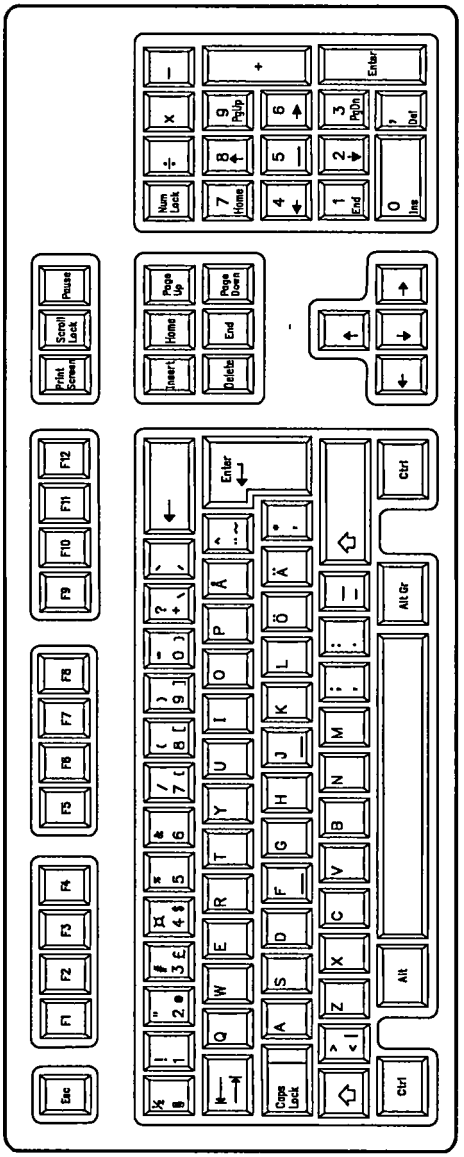
Canada (French) Keyboard



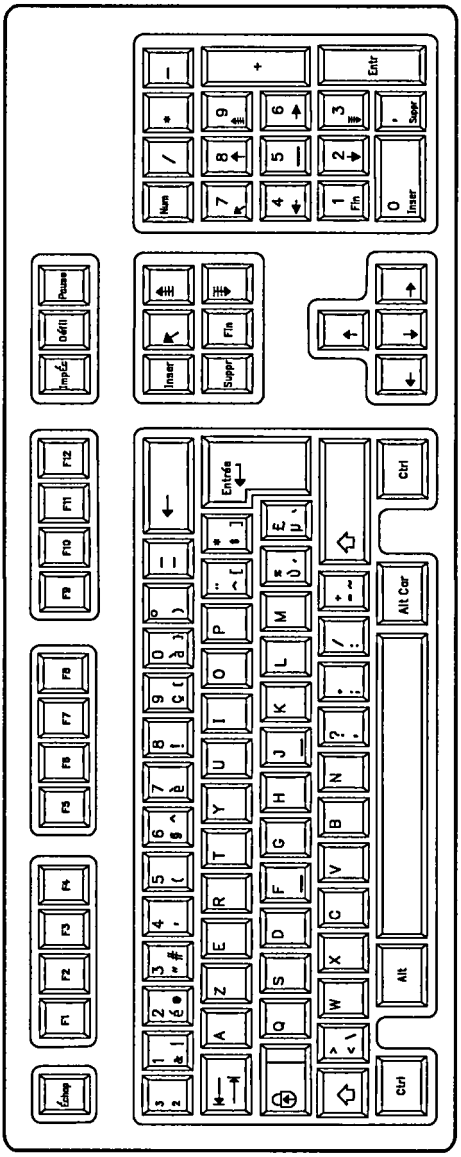
Denmark Keyboard



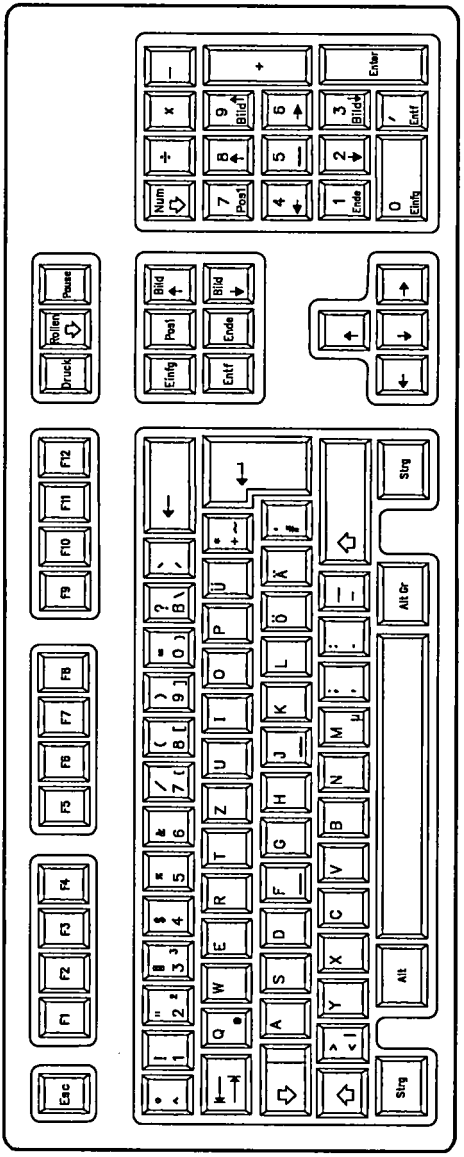
Finland and Sweden Keyboard



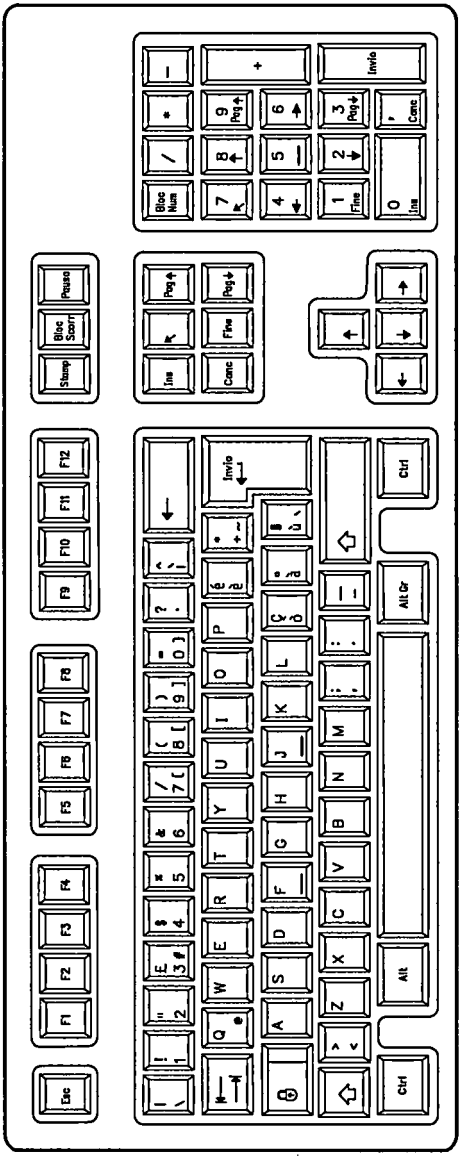
France and Belgium (French) Keyboard



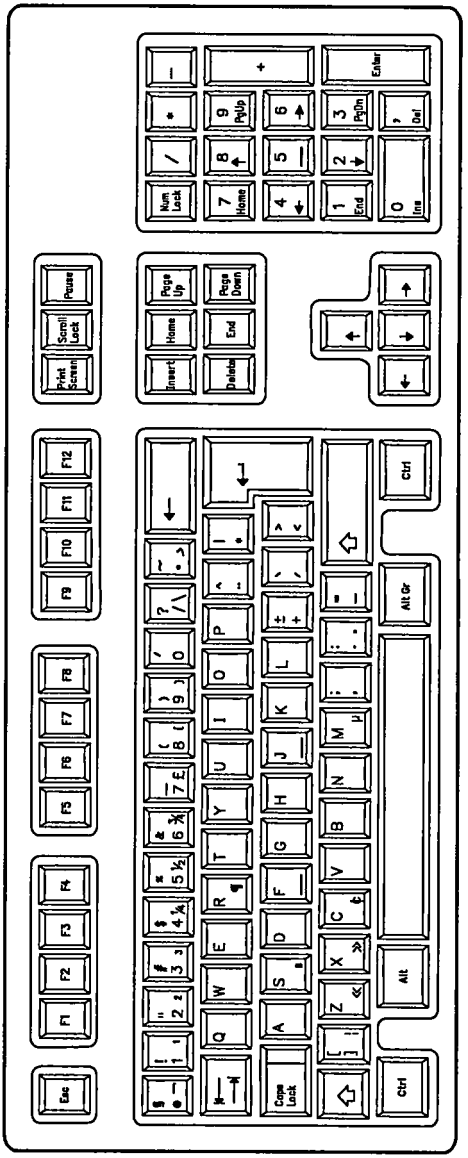
Germany Keyboard



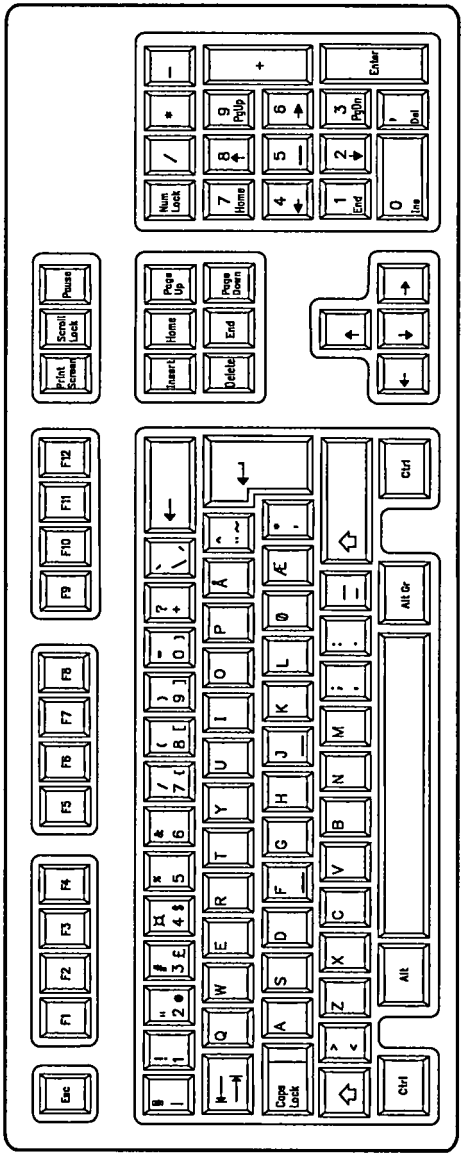
Italy Keyboard



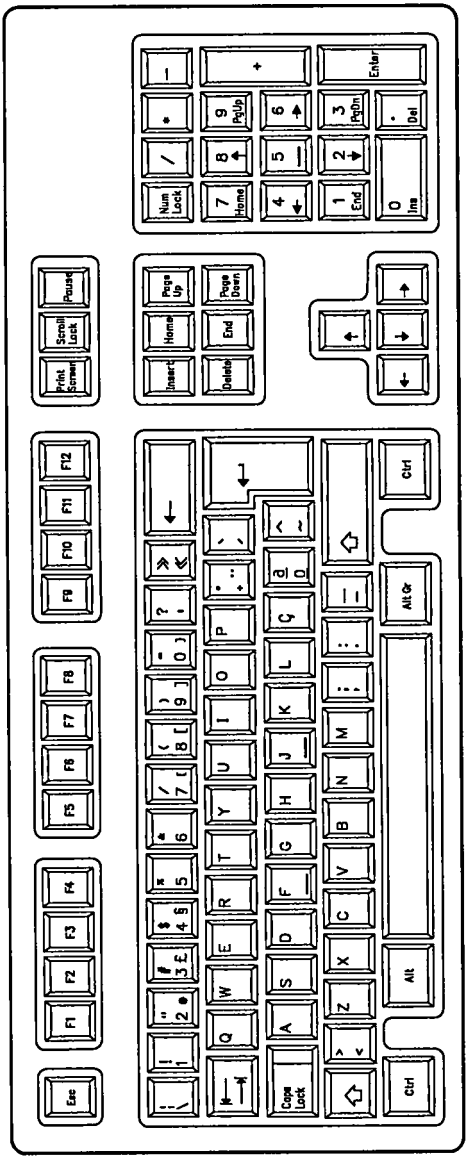
Netherlands Keyboard



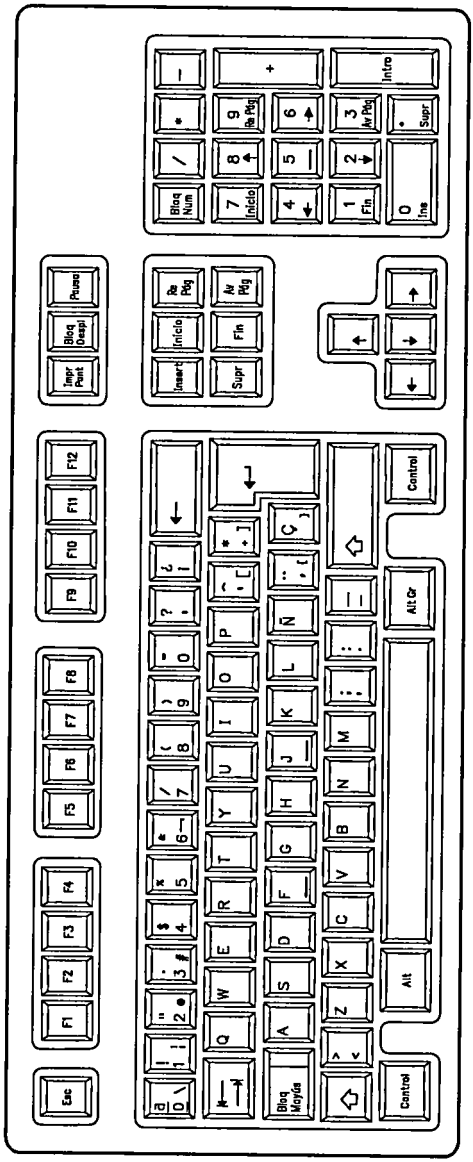
Norway Keyboard



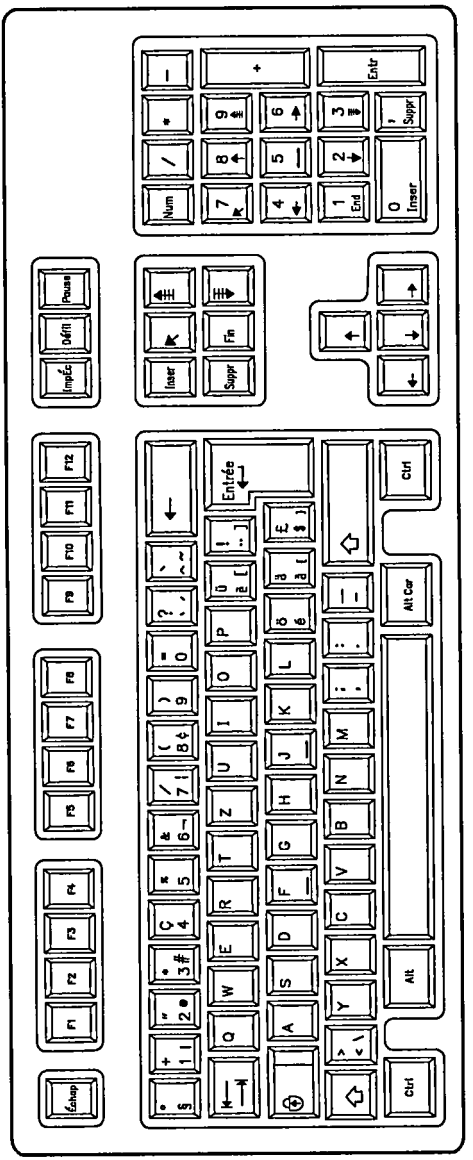
Portugal Keyboard



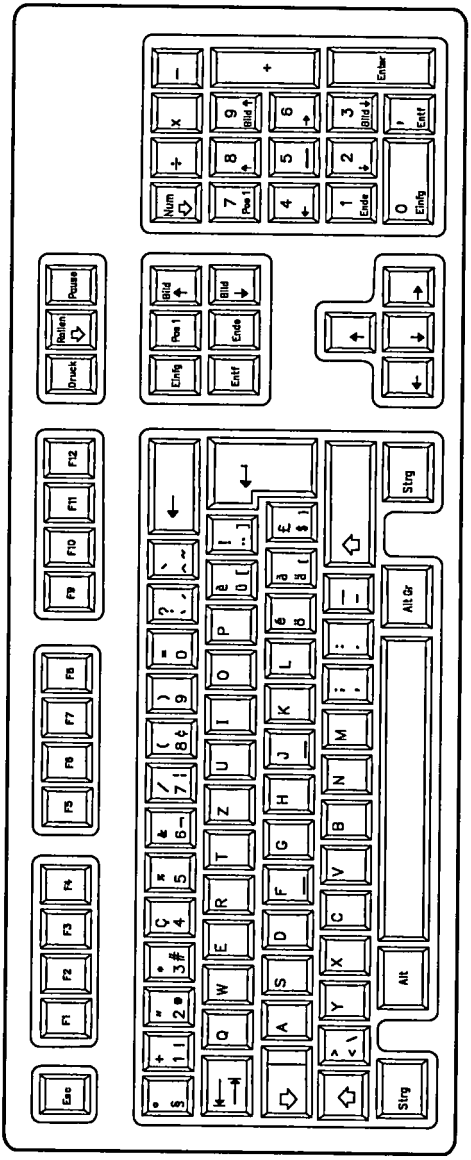
Spain Keyboard



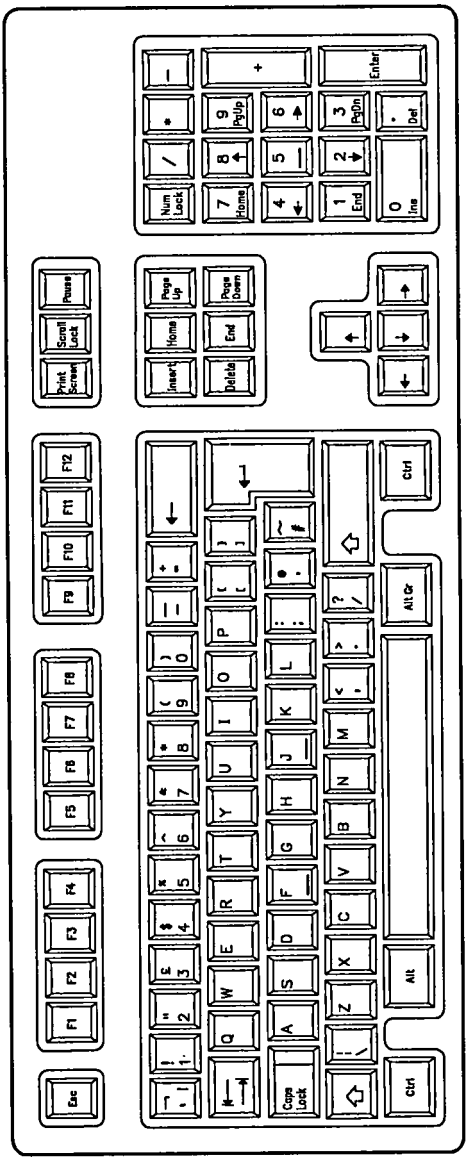
Switzerland (French) Keyboard



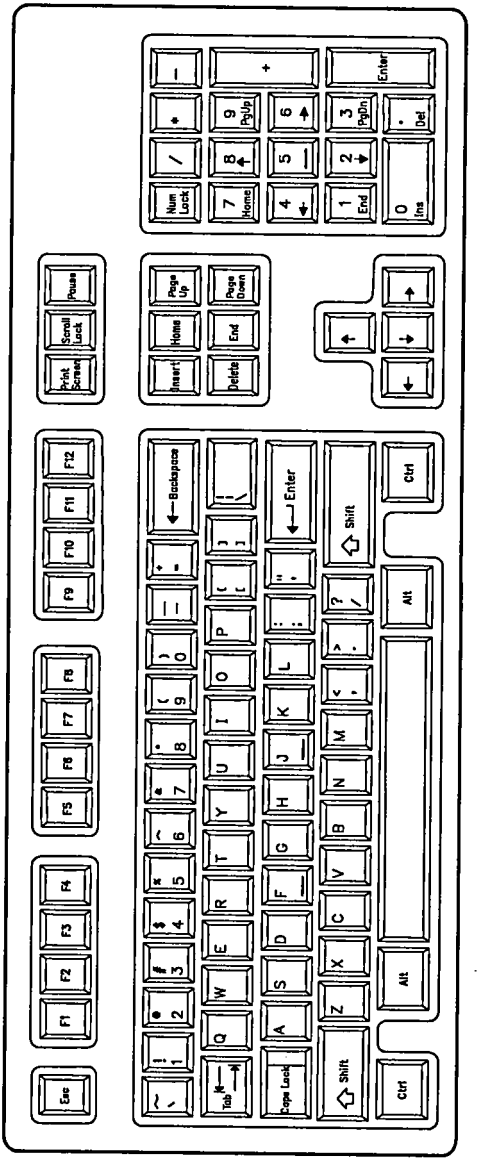
Switzerland (German) Keyboard



U.K. Keyboard


Esc
F1
F2
F3
F4
F5
F6
F7
F8
F9
F10
F11
F12
Print Screen
Scroll Lock
Pause
Insert
Home
Page Up
Delete
End
Page Down
Num Lock
Caps Lock
Ctrl
Alt
Alt Gr
Enter
Del
Ins
PgUp
PgDn

U.S. and Canada (English) Keyboard



Sequential Key-Code Scanning

The keyboard detects all keys pressed, and sends each scan code in the correct sequence. The keyboard stores the scan codes in its buffer until serviced by the system.

Buffer

A 17-byte first-in-first-out (FIFO) buffer in the keyboard stores the scan codes until the system is ready to receive them. A buffer-overrun condition occurs when more than 16 bytes are placed in the keyboard buffer. An overrun code replaces the 17th byte. If more keys are pressed before the system allows keyboard output, the additional data is lost.

When the keyboard is allowed to send data, the bytes in the buffer are sent as in normal operation, and new data entered is detected and sent. Response codes do not occupy a buffer position.

If keystrokes generate a multiple-byte sequence, the entire sequence must fit into the available buffer space or the keystroke is discarded and a buffer-overrun condition occurs.

Keys

With the exception of the Pause key, all keys are make/break. The make scan code of a key is sent to the keyboard controller when the key is pressed. The break scan code is sent when the key is released.

Additionally, except for the Pause key, all keys are typematic. When a key is pressed and held down, the keyboard sends the make code for that key, delays 500 milliseconds ±20%, and begins sending make codes for that key at a rate of 10.9 characters per second ±20%. The typematic rate and delay can be modified. (See "Set Typematic Rate/Delay (Hex F3)" on page 5-26.)

If two or more keys are held down, only the last key pressed repeats at the typematic rate. Typematic operation stops when the last key pressed is released, even if other keys are still held down. If a key is pressed and held down while keyboard transmission is inhibited, only the first make code is stored in the buffer. This prevents buffer overflow as a result of typematic action.

Note: Scan code set 3 allows key types to be changed by the system. (See "Set 3 Scan Code Tables" on page 5-36 for the default settings.)

Power-On Routine

The following activities take place when power is first applied to the keyboard.

Power-On Reset (POR)

A "power-on reset" (POR) signal occurs a minimum of 150 milliseconds and a maximum of 2.0 seconds from the time power is applied to the keyboard.

Basic Assurance Test (BAT)

The basic assurance test (BAT) consists of a keyboard processor test, a read-only memory (ROM) checksum, and a random-access memory (RAM) test. During the BAT, activity on the "clock" and "data" lines is ignored. The LEDs are turned on at the beginning and off at the end of the BAT. The BAT takes a minimum of 300 milliseconds and a maximum of 500 milliseconds. This is in addition to the time required for the POR.

Upon satisfactory completion of the BAT, the keyboard sends the BAT completion code (hex AA) and keyboard scanning begins. If a BAT failure occurs, the keyboard sends the BAT failure code (hex FC) and the keyboard is disabled pending command input. Completion codes are sent between 450 milliseconds and 2.5 seconds after POR, and between 300 and 500 milliseconds after a Reset command is acknowledged.

Immediately following POR, the keyboard monitors the signals on the keyboard "clock" and "data" lines and sets the line protocol.

Commands from the System

Figure 5-3 shows the commands that the system may send and their hexadecimal values.

Figure 5-3. Keyboard Commands from the System

Command	Hex Value
Set/Reset Status Indicators	ED
Echo	EE
Invalid Command	EF
Select Alternate Scan Codes	F0
Invalid Command	F1
Read ID	F2
Set Typematic Rate/Delay	F3
Enable	F4
Default Disable	F5
Set Default	F6
Set All Keys - Typematic	F7
- Make/Break	F8
- Make	F9
- Typematic Make/Break	FA
Set Key Type - Typematic	FB
- Make/Break	FC
- Make	FD
Resend	FE
Reset	FF

These commands can be sent to the keyboard at any time. The keyboard responds within 20 milliseconds, except when performing the BAT or executing a Reset command.

The following commands are in alphabetical order. They have different meanings when issued by the keyboard. (See "Commands to the System" on page 5-28.)

Default Disable (Hex F5): The Default Disable command resets all conditions to the power-on default state. The keyboard responds with Acknowledge, clears its output buffer, sets the default key types (scan code set 3 operation only) and typematic rate/delay, and clears the last typematic key. The keyboard stops scanning, and waits for the next command.

Echo (Hex EE): The Echo command is a diagnostic aid. When the keyboard receives this command, it responds with Echo and if the keyboard was previously enabled continues scanning.

Enable (Hex F4): When the Enable command is sent, the keyboard responds with Acknowledge, clears its output buffer, clears the last typematic key, and starts scanning.

Invalid Command (Hex EF and F1): Hex EF and hex F1 are invalid commands and are not supported. If one of these is sent, the keyboard responds with a Resend and continues in its prior scanning state. No other activities occur.

Read ID (Hex F2): The Read ID command requests identification information from the keyboard. The keyboard responds with Acknowledge, discontinues scanning, and sends the two keyboard ID bytes. The second byte must follow completion of the first by no more than 500 microseconds. After sending the second ID byte, the keyboard continues scanning.

Resend (Hex FE): The Resend command is sent when the system detects an error in any transmission from the keyboard. It is sent only after a keyboard transmission and before the system allows the next keyboard output. The keyboard responds with the previous output again (unless the previous output was Resend, in which case the keyboard sends the last byte before the Resend command).

Reset (Hex FF): The Reset command starts a program reset and a keyboard internal self-test. The keyboard responds with Acknowledge and ensures that the system accepts the Acknowledge before executing the command. The system signals acceptance of Acknowledge by raising the "clock" and "data" lines for a minimum of 500 microseconds. The keyboard is disabled from the time it receives the Reset command until Acknowledge is accepted, or until another command is sent that overrides the previous command.

Following acceptance of Acknowledge, the keyboard is reinitialized and performs the BAT. After sending the BAT completion code, the keyboard defaults to scan code set 2.

Select Alternate Scan Codes (Hex F0): The Select Alternate Scan Codes command instructs the keyboard to select one of three sets of scan codes. The keyboard responds with Acknowledge and clears both the output buffer and the typematic key (if one is active). The system then sends the option byte and the keyboard responds with another Acknowledge. An option byte value of hex 01 selects scan code set 1, hex 02 selects set 2, and hex 03 selects set 3.

An option byte value of hex 00 causes the keyboard to respond with Acknowledge and send a byte telling the system which scan code set is currently in use.

After establishing the new scan code set, the keyboard returns to the scanning state it was in before receiving the Select Alternate Scan Codes command.

Set All Keys (Hex F7, F8, F9, FA)
The Set All Keys commands instruct the keyboard to set all the keys to the type listed in Figure 5-4.

Figure 5-4. Set All Keys Commands

Hex Value	Command
F7	Set All Keys - Typematic
F8	Set All Keys - Make/Break
F9	Set All Keys - Make
FA	Set All Keys - Typematic Make/Break

The keyboard responds with Acknowledge, clears its output buffer, sets all keys to the type indicated by the command, and continues scanning (if it was previously enabled). Although these commands can be sent using any scan code set, they affect only the operation of scan code set 3.

Set Default (Hex F6): The Set Default command resets all conditions to the power-on default state. The keyboard responds with Acknowledge, clears its output buffer, sets the default key types (scan code set 3 operation only) and typematic rate/delay, clears the last typematic key, and continues scanning (if it was previously enabled).

Set Key Type (Hex FB, FC, FD): The Set Key Type commands instruct the keyboard to set all the keys to the type listed in Figure 5-5.

Figure 5-5. Set Key Type Commands

Hex Value	Command
FB	Set Key Type - Typematic
FC	Set Key Type - Make/Break
FD	Set Key Type - Make

The keyboard responds with Acknowledge, clears its output buffer, and prepares to receive key identification. Key identification is accomplished by the system identifying each key by its scan code value as defined in scan code set 3. Only scan code set 3 values are

valid for key identification. The type of each identified key is set to the value indicated by the command.

These commands can be sent using any scan code set, but affect only the operation of scan code set 3.

Set/Reset Status Indicators (Hex ED): Three status indicators on the PC Enhanced keyboard— Caps Lock, Num Lock, and Scroll Lock— are accessible by the system. The keyboard activates or deactivates these indicators when it receives a valid command-code sequence from the system. The command sequence begins with the Set/Reset Status command. The keyboard responds Acknowledge, discontinues scanning, and waits for the option byte from the system. The bit assignments for this option byte are as listed in Figure 5-6.

Figure 5-6. Set/Reset Status Indicators

Bit	Function
7 - 3	Reserved (must be 0s)
2	Caps Lock Indicator
1	Num Lock Indicator
0	Scroll Lock Indicator

If the bit for an indicator is set to 1, the indicator is turned on. If the bit is set to 0, the indicator is turned off.

The keyboard responds to the option byte with Acknowledge, sets the indicators and, if the keyboard was previously enabled, continues scanning. The state of the indicators will reflect the bits in the option byte and can be activated or deactivated in any combination. If another command is received in place of the option byte, execution of the Set/Reset Mode Indicators command is stopped, with no change to the indicator states, and the new command is processed.

Immediately after power-on, the lights default to the off state. If the Set Default and Default Disable commands are received, the lights remain in the state they were in before the command was received.

Set Typematic Rate/Delay (Hex F3): The Set Typematic Rate/Delay command changes the typematic rate and delay. The keyboard responds Acknowledge, stops scanning, and waits for the system to issue the rate/delay value byte. The keyboard responds to the rate/delay value byte with another Acknowledge, sets the rate and delay to the values indicated, and continues scanning (if it was previously enabled). Bit 7 is always 0. Bits 6 and 5 set the delay. The delay is equal to 1 plus the binary value of bits 6 and 5, multiplied by 250 milliseconds ±20%. Bits 4, 3, 2, 1, and 0 set the rate.

The period (interval from one typematic output to the next) is determined by the following equation:

Period = (8 + A) X (2^B) X 0.00417 seconds.
where:
A = binary value of bits 2, 1, and 0.
B = binary value of bits 4 and 3.

The typematic rate (make codes per second) is the inverse of the period.

Figure 5-7. Typematic Rate

Bit	Typematic Rate ± 20%	Bit	Typematic Rate ± 20%
00000	30.0	10000	7.5
00001	26.7	10001	6.7
00010	24.0	10010	6.0
00011	21.8	10011	5.5
00100	20.0	10100	5.0
00101	18.5	10101	4.6
00110	17.1	10110	4.3
00111	16.0	10111	4.0
01000	15.0	11000	3.7
01001	13.3	11001	3.3
01010	12.0	11010	3.0
01011	10.9	11011	2.7
01100	10.0	11100	2.5
01101	9.2	11101	2.3
01110	8.6	11110	2.1
01111	8.0	11111	2.0

The default values for the system keyboard are as follows:

Typematic rate = 10.9 characters per second ± 20%.

Delay = 500 milliseconds ± 20%.

The execution of this command stops without change to the existing rate if another command is received instead of the rate/delay value byte.

Commands to the System

Figure 5-8 shows the commands that the keyboard may send to the system and their hexadecimal values.

Figure 5-8. Keyboard Commands to the System

Command	Hex Value
Key Detection Error/Overrun	00 (Code Sets 2 and 3)
Keyboard ID (Enhanced)	83AB
BAT Completion Code	AA
BAT Failure Code	FC
Echo	EE
Acknowledge (ACK)	FA
Resend	FE
Key Detection Error/Overrun	FF (Code Set 1)

The commands the keyboard sends to the system are described below, in alphabetical order. They have different meanings when issued by the system. (See "Commands from the System" on page 5-23.)

Acknowledge (Hex FA): The keyboard sends Acknowledge to any valid input other than an Echo or Resend command. If the keyboard is interrupted while sending Acknowledge, it discards Acknowledge and accepts and responds to the new command.

BAT Completion Code (Hex AA): The keyboard sends the BAT completion code following satisfactory completion of BAT. Any other code indicates a failure of the keyboard.

BAT Failure Code (Hex FC): The keyboard sends the BAT failure code if a BAT failure occurs. The keyboard then discontinues scanning and waits for a system response or reset.

Echo (Hex EE): The keyboard sends Echo in response to an Echo command.

Keyboard ID: The Keyboard ID consists of two bytes, hex 83AB (PC Enhanced keyboard). The keyboard responds to the read ID with Acknowledge, discontinues scanning, and sends the two ID bytes. The least-significant byte is sent first, followed by the most-significant byte. Following the output of the keyboard ID, the keyboard begins scanning.

Key Detection Error (Hex 00 or FF): The keyboard sends the key detection error if conditions in the keyboard make it impossible to identify a switch closure. If the keyboard is using scan code set 1, the code is hex FF. For sets 2 and 3, the code is hex 00.

Overrun (Hex 00 or FF): An overrun character is placed in the keyboard buffer and replaces the last code when the buffer capacity has been exceeded. The code is sent to the system when it reaches the top of the buffer. If the keyboard is using scan code set 1, the code is hex FF. For sets 2 and 3, the code is hex 00.

Resend (Hex FE): The keyboard issues a Resend command following receipt of an invalid input or any input with incorrect parity. If the system sends nothing to the keyboard, no response is required.

Scan Codes

The following tables list the key numbers of the three scan code sets and their hexadecimal values. The system defaults to scan code set 2, but can be switched to set 1 or set 3. (See "Select Alternate Scan Codes (Hex F0)" on page 5-24.)

Set 1 Scan Code Tables

In scan code set 1, each key is assigned a base scan code and, in some cases, extra codes to generate artificial shift states in the system. The typematic scan codes are identical to the base scan code for each key.

Figure 5-9 on page 5-30 shows the keys and the scan codes they send, regardless of any shift states in the keyboard or the system. Refer to "Keyboard Charts" on page 5-3 to determine the character associated with each key number.

Figure 5-9. Keyboard Scan Codes, Set 1 (Part 1 of 5)

Key Number	Make Code	Break Code	Key Number	Make Code	Break Code
1	29	A9	47	2D	AD
2	02	82	48	2E	AE
3	03	83	49	2F	AF
4	04	84	50	30	B0
5	05	85	51	31	B1
6	06	86	52	32	B2
7	07	87	53	33	B3
8	08	88	54	34	B4
9	09	89	55	35	B5
10	0A	8A	57	36	B6
11	0B	8B	58	1D	9D
12	0C	8C	60	38	B8
13	0D	8D	61	39	B9
15	0E	8E	62	E0 38	E0 B8
16	0F	8F	64	E0 1D	E0 9D
17	10	90	90	45	C5
18	11	91	91	47	C7
19	12	92	92	4B	CB
20	13	93	93	4F	CF
21	14	94	96	48	C8
22	15	95	97	4C	CC
23	16	96	98	50	D0
24	17	97	99	52	D2
25	18	98	100	37	B7
26	19	99	101	49	C9
27	1A	9A	102	4D	CD
28	1B	9B	103	51	D1
29 **	2B	AB	104	53	D3
30	3A	BA	105	4A	CA
31	1E	9E	106	4E	CE
32	1F	9F	108	E0 1C	E0 9C
33	20	A0	110	01	81
34	21	A1	112	3B	BB
35	22	A2	113	3C	BC
36	23	A3	114	3D	BD
37	24	A4	115	3E	BE
38	25	A5	116	3F	BF
39	26	A6	117	40	C0
40	27	A7	118	41	C1
41	28	A8	119	42	C2
42 ***	2B	AB	120	43	C3
43	1C	9C	121	44	C4
44	2A	AA	122	57	D7
45 ***	56	D6	123	58	D8
46	2C	AC	125	46	C6

** Key 29 on U.S. keyboard only.
*** Keys 42 and 45 on all but U.S. keyboards.

The remaining keys send a series of codes dependent on the state of the shift keys (Ctrl, Alt, and Shift), and the state of Num Lock (On or Off). Because the base scan code is identical to that of another key, an extra code (hex E0) has been added to the base code to make it unique.

The following charts show the make/break code using the left Shift key. If the right Shift key is used, substitute its make/break code for that of the left Shift key.

Figure 5-10. Keyboard Scan Codes, Set 1 (Part 2 of 5)

Key No.	Base Case, or Shift + Num Lock Make/Break	Shift Case Make/Break	Num Lock on Make/Break
75	E0 52	E0 AA E0 52	E0 2A E0 52
	/E0 D2	/E0 D2 E0 2A	/E0 D2 E0 AA
76	E0 53	E0 AA E0 53	E0 2A E0 53
	/E0 D3	/E0 D3 E0 2A	/E0 D3 E0 AA
79	E0 4B	E0 AA E0 4B	E0 2A E0 4B
	/E0 CB	/E0 CB E0 2A	/E0 CB E0 AA
80	E0 47	E0 AA E0 47	E0 2A E0 47
	/E0 C7	/E0 C7 E0 2A	/E0 C7 E0 AA
81	E0 4F	E0 AA E0 4F	E0 2A E0 4F
	/E0 CF	/E0 CF E0 2A	/E0 CF E0 AA
83	E0 48	E0 AA E0 48	E0 2A E0 48
	/E0 C8	/E0 C8 E0 2A	/E0 C8 E0 AA
84	E0 50	E0 AA E0 50	E0 2A E0 50
	/E0 D0	/E0 D0 E0 2A	/E0 D0 E0 AA
85	E0 49	E0 AA E0 49	E0 2A E0 49
	/E0 C9	/E0 C9 E0 2A	/E0 C9 E0 AA
86	E0 51	E0 AA E0 51	E0 2A E0 51
	/E0 D1	/E0 D1 E0 2A	/E0 D1 E0 AA
89	E0 4D	E0 AA E0 4D	E0 2A E0 4D
	/E0 CD	/E0 CD E0 2A	/E0 CD E0 AA

Figure 5-11. Keyboard Scan Codes, Set 1 (Part 3 of 5)

Key No.	Scan Code Make/Break	Shift Case Make/Break
95	E0 35/E0 B5	E0 AA E0 35/E0 B5 E0 2A

Figure 5-12. Keyboard Scan Codes, Set 1 (Part 4 of 5)

Key No.	Scan Code Make/Break	Ctrl Case, Shift Case Make/Break	Alt Case Make/Break
124	E0 2A E0 37 /E0 B7 E0 AA	E0 37/E0 B7	54/D4

Figure 5-13. Keyboard Scan Codes, Set 1 (Part 5 of 5)

Key No.	Make Code	Ctrl Key Pressed
126 *	E1 1D 45 E1 9D C5	E0 46 E0 C6

* This key is not typematic. All associated scan codes occur on the make of the key.

Set 2 Scan Code Tables

In scan code set 2, each key is assigned a unique 8-bit make scan code, which is sent when the key is pressed. Each key also sends a break code when the key is released. The break code consists of 2 bytes, the first of which is the break code prefix (hex F0). The second byte is the same as the make scan code for that key. The typematic scan code for a key is the same as the make code.

The following charts show the keys and the scan codes the keys send, regardless of any shift states in the keyboard or the system. Refer to"Keyboard Charts" on page 5-3 to determine the character associated with each key number.

Figure 5-14. Keyboard Scan Codes, Set 2 (Part 1 of 5)

Key Number	Make Code	Break Code	Key Number	Make Code	Break Code
1	0E	F0 0E	47	22	F0 22
2	16	F0 16	48	21	F0 21
3	1E	F0 1E	49	2A	F0 2A
4	26	F0 26	50	32	F0 32
5	25	F0 25	51	31	F0 31
6	2E	F0 2E	52	3A	F0 3A
7	36	F0 36	53	41	F0 41
8	3D	F0 3D	54	49	F0 49
9	3E	F0 3E	55	4A	F0 4A
10	46	F0 46	57	59	F0 59
11	45	F0 45	58	14	F0 14
12	4E	F0 4E	60	11	F0 11
13	55	F0 55	61	29	F0 29
15	66	F0 66	62	E0 11	E0 F0 11
16	0D	F0 0D	64	E0 14	E0 F0 14
17	15	F0 15	90	77	F0 77
18	1D	F0 1D	91	6C	F0 6C
19	24	F0 24	92	6B	F0 6B
20	2D	F0 2D	93	69	F0 69
21	2C	F0 2C	96	75	F0 75
22	35	F0 35	97	73	F0 73
23	3C	F0 3C	98	72	F0 72
24	43	F0 43	99	70	F0 70
25	44	F0 44	100	7C	F0 7C
26	4D	F0 4D	101	7D	F0 7D
27	54	F0 54	102	74	F0 74
28	5B	F0 5B	103	7A	F0 7A
29 **	5D	F0 5D	104	71	F0 71
30	58	F0 58	105	7B	F0 7B
31	1C	F0 1C	106	79	F0 79
32	1B	F0 1B	108	E0 5A	E0 F0 5A
33	23	F0 23	110	76	F0 76
34	2B	F0 2B	112	05	F0 05
35	34	F0 34	113	06	F0 06
36	33	F0 33	114	04	F0 04
37	3B	F0 3B	115	0C	F0 0C
38	42	F0 42	116	03	F0 03
39	4B	F0 4B	117	0B	F0 0B
40	4C	F0 4C	118	83	F0 83
41	52	F0 52	119	0A	F0 0A
42 ***	5D	F0 5D	120	01	F0 01
43	5A	F0 5A	121	09	F0 09
44	12	F0 12	122	78	F0 78
45 ***	61	F0 61	123	07	F0 07
46	1A	F0 1A	125	7E	F0 7E

** Key 29 on U.S. keyboards only.
*** Keys 42 and 45 on all but U.S. keyboards.

The remaining keys send a series of codes dependent on the state of the shift keys (Ctrl, Alt, and Shift), and the state of Num Lock (On or Off). Because the base scan code is identical to that of another key, an extra code (hex E0) has been added to the base code to make it unique.

The following charts show the make/break code using the left Shift key. If the right Shift key is used, substitute its make/break code for that of the left Shift key.

Figure 5-15. Keyboard Scan Codes, Set 2 (Part 2 of 5)

Key No.	Base Case, or Shift + Num Lock Make/Break	Shift Case Make/Break	Num Lock on Make/Break
75	E0 70	E0 F0 12 E0 70	E0 12 E0 70
	/E0 F0 70	/E0 F0 70 E0 12	/E0 F0 70 E0 F0 12
76	E0 71	E0 F0 12 E0 71	E0 12 E0 71
	/E0 F0 71	/E0 F0 71 E0 12	/E0 F0 71 E0 F0 12
79	E0 6B	E0 F0 12 E0 6B	E0 12 E0 6B
	/E0 F0 6B	/E0 F0 6B E0 12	/E0 F0 6B E0 F0 12
80	E0 6C	E0 F0 12 E0 6C	E0 12 E0 6C
	/E0 F0 6C	/E0 F0 6C E0 12	/E0 F0 6C E0 F0 12
81	E0 69	E0 F0 12 E0 69	E0 12 E0 69
	/E0 F0 69	/E0 F0 69 E0 12	/EO F0 69 E0 F0 12
83	E0 75	E0 F0 12 E0 75	E0 12 E0 75
	/E0 F0 75	/E0 F0 75 E0 12	/E0 F0 75 E0 F0 12
84	E0 72	E0 F0 12 E0 72	E0 12 E0 72
	/E0 F0 72	/E0 F0 72 E0 12	/E0 F0 72 E0 F0 12
85	E0 7D	E0 F0 12 E0 7D	E0 12 E0 7D
	/E0 F0 7D	/E0 F0 7D E0 12	/E0 F0 7D E0 F0 12
86	E0 7A	E0 F0 12 E0 7A	E0 12 E0 7A
	/E0 F0 7A	/E0 F0 7A E0 12	/E0 F0 7A E0 F0 12
89	E0 74	E0 F0 12 E0 74	E0 12 E0 74
	/E0 F0 74	/E0 F0 74 E0 12	/E0 F0 74 E0 F0 12

Figure 5-16. Keyboard Scan Codes, Set 2 (Part 3 of 5)

Key No.	Scan Code Make/Break	Shift Case Make/Break
95	E0 4A/E0 F0 4A	E0 F0 12 4A/E0 12 F0 4A

Figure 5-17. Keyboard Scan Codes, Set 2 (Part 4 of 5)

Key No.	Scan Code Make/Break	Ctrl Case, Shift Case Make/Break	Alt Case Make/Break
124	E0 12 E0 7C /E0 F0 7C E0 F0 12	E0 7C/E0 F0 7C	84/F0 84

Figure 5-18. Keyboard Scan Codes, Set 2 (Part 5 of 5)

Key No.	Make Code	Ctrl Key Pressed
126 *	E1 14 77 E1 F0 14 F0 77	E0 7E E0 F0 7E

* This key is not typematic. All associated scan codes occur on the make of the key.

Set 3 Scan Code Tables

In scan code set 3, each key is assigned a unique 8-bit make scan code, which is sent when the key is pressed. Each key also sends a break code when the key is released. The break code consists of 2 bytes, the first of which is the break code prefix (hex F0). The second byte is the same as the make scan code for that key. The typematic scan code for a key is the same as the make code. With this scan code set, each key sends only one scan code, and no keys are affected by the state of any other keys.

The following charts show the keys and the scan codes they send, regardless of any shift states in the keyboard or the system. Refer to"Keyboard Charts" on page 5-3 to determine the character associated with each key number.

Figure 5-19 (Page 1 of 3). Keyboard Scan Codes, Set 3

Key Number	Make Code	Break Code	Default Key State
1	0E	F0 0E	Typematic
2	16	F0 16	Typematic
3	1E	F0 1E	Typematic
4	26	F0 26	Typematic
5	25	F0 25	Typematic
6	2E	F0 2E	Typematic
7	36	F0 36	Typematic
8	3D	F0 3D	Typematic
9	3E	F0 3E	Typematic
10	46	F0 46	Typematic
11	45	F0 45	Typematic
12	4E	F0 4E	Typematic
13	55	F0 55	Typematic
15	66	F0 66	Typematic
16	0D	F0 0D	Typematic
17	15	F0 15	Typematic
18	1D	F0 1D	Typematic
19	24	F0 24	Typematic
20	2D	F0 2D	Typematic
21	2C	F0 2C	Typematic
22	35	F0 35	Typematic
23	3C	F0 3C	Typematic
24	43	F0 43	Typematic
25	44	F0 44	Typematic
26	4D	F0 4D	Typematic
27	54	F0 54	Typematic
28	5B	F0 5B	Typematic
29 **	5C	F0 5C	Typematic
30	14	F0 14	Make/Break
31	1C	F0 1C	Typematic
32	1B	F0 1B	Typematic
33	23	F0 23	Typematic
34	2B	F0 2B	Typematic
35	34	F0 34	Typematic

Figure 5-19 (Page 2 of 3). Keyboard Scan Codes, Set 3

Key Number	Make Code	Break Code	Default Key State
36	33	F0 33	Typematic
37	3B	F0 3B	Typematic
38	42	F0 42	Typematic
39	4B	F0 4B	Typematic
40	4C	F0 4C	Typematic
41	52	F0 52	Typematic
42 ***	53	F0 53	Typematic
43	5A	F0 5A	Typematic
44	12	F0 12	Make/Break
45 ***	13	F0 13	Typematic
46	1A	F0 1A	Typematic
47	22	F0 22	Typematic
48	21	F0 21	Typematic
49	2A	F0 2A	Typematic
50	32	F0 32	Typematic
51	31	F0 31	Typematic
52	3A	F0 3A	Typematic
53	41	F0 41	Typematic
54	49	F0 49	Typematic
55	4A	F0 4A	Typematic
57	59	F0 59	Make/Break
58	11	F0 11	Make/Break
60	19	F0 19	Make/Break
61	29	F0 29	Typematic
62	39	F0 39	Make only
64	58	F0 58	Make only
75	67	F0 67	Make only
76	64	F0 64	Typematic
79	61	F0 61	Typematic
80	6E	F0 6E	Make only
81	65	F0 65	Make only
83	63	F0 63	Typematic
84	60	F0 60	Typematic
85	6F	F0 6F	Make only
86	6D	F0 6D	Make only
89	6A	F0 6A	Typematic
90	76	F0 76	Make only
91	6C	F0 6C	Make only
92	6B	F0 6B	Make only
93	69	F0 69	Make only
95	77	F0 77	Make only
96	75	F0 75	Make only
97	73	F0 73	Make only
98	72	F0 72	Make only
99	70	F0 70	Make only
100	7E	F0 7E	Make only
101	7D	F0 7D	Make only
102	74	F0 74	Make only
103	7A	F0 7A	Make only
104	71	F0 71	Make only
105	84	F0 84	Make only
106	7C	F0 7C	Typematic
108	79	F0 79	Make only
110	08	F0 08	Make only

Figure 5-19 (Page 3 of 3). Keyboard Scan Codes, Set 3

Key Number	Make Code	Break Code	Default Key State
112	07	F0 07	Make only
113	0F	F0 0F	Make only
114	17	F0 17	Make only
115	1F	F0 1F	Make only
116	27	F0 27	Make only
117	2F	F0 2F	Make only
118	37	F0 37	Make only
119	3F	F0 3F	Make only
120	47	F0 47	Make only
121	4F	F0 4F	Make only
122	56	F0 56	Make only
123	5E	F0 5E	Make only
124	57	F0 57	Make only
125	5F	F0 5F	Make only
126	62	F0 62	Make only

** Key 29 on U.S. keyboards only.

*** Keys 42 and 45 on all but U.S. keyboards.

Clock and Data Signals

The keyboard and system communicate over the "clock" and "data" lines. The source of each of these lines is an open-collector device on the keyboard that allows either the keyboard or the system to force a line to an inactive (low) level. When no communication is occurring, the "clock" line is at an active (high) level. The state of the "data" line is held active (high) by the keyboard.

When the system sends data to the keyboard, it forces the "data" line to an inactive level and allows the "clock" line to go to an active level.

An inactive signal will have a value of at least 0 V dc, but not greater than +0.7 V dc. A signal at the inactive level is a logical 0. An active signal will have a value of at least +2.4 V dc, but not greater than +5.5 V dc. A signal at the active level is a logical 1. Voltages are measured between a signal source and the dc ground.

When the keyboard sends data to or receives data from the system, it generates the "clock" signal to time the data. The system can prevent the keyboard from sending data by forcing the "clock" line to an inactive level; the "data" line may be active or inactive during this time.

During the BAT, the keyboard allows the "clock" and "data" lines to go to an active level.

Data Stream

Data transmissions to and from the keyboard consist of an 11-bit data stream (mode 2) sent serially over the "data" line. Figure 5-20 on page 5-40 shows the functions of the bits.

Figure 5-20. Keyboard Data Stream Bit Definitions

Bit	Function
11	Stop Bit (always 1)
10	Parity Bit (odd parity)
9	Data Bit 7 (most-significant)
8	Data Bit 6
7	Data Bit 5
6	Data Bit 4
5	Data Bit 3
4	Data Bit 2
3	Data Bit 1
2	Data Bit 0 (least-significant)
1	Start Bit (always 0)

Note: Mode 1 is a 9-bit data stream that does not have a parity bit or stop bit, and the start bit is always 1.

Data Output

When the keyboard is ready to send data, it first checks for a keyboard-inhibit or system request-to-send status on the "clock" and "data" lines. If the "clock" line is inactive (low), data is stored in the keyboard buffer. If the "clock" line is active (high) and the "data" line is inactive (request-to-send), data is stored in the keyboard buffer, and the keyboard receives system data.

If the "clock" and "data" lines are both active, the keyboard sends the start bit (0), 8 data bits, the parity bit, and the stop bit (1). Data will be valid before the trailing edge and beyond the leading edge of the clock pulse. During transmission, the keyboard checks the "clock" line for an active level at least every 60 milliseconds. If the system lowers the "clock" line from an active level after the keyboard starts sending data, a condition known as line contention occurs, and the keyboard stops sending data. If line contention occurs before the leading edge of the 10th clock signal (parity bit), the keyboard buffer returns the "clock" and "data" lines to an active level. If line contention does not occur by the 10th clock signal, the keyboard completes the transmission. Following line contention, the system can request the keyboard to resend the data.

Following a transmission, the system can inhibit the keyboard until the system processes the input, or until it requests that a response be sent.

Data Input

When the system is ready to send data to the keyboard, it first checks to see if the keyboard is sending data. If the keyboard is sending, but has not reached the 10th clock signal, the system can override the keyboard output by forcing the keyboard "clock" line to an inactive (low) level. If the keyboard transmission is beyond the 10th clock signal, the system must receive the transmission.

If the keyboard is not sending, or if the system elects to override the output from the keyboard, the system forces the keyboard "clock" line to an inactive level for more than 60 microseconds while preparing to send data. When the system is ready to send the start bit (the "data" line will be inactive), it allows the "clock" line to go to an active (high) level.

The keyboard checks the state of the "clock" line at intervals of no more than 10 milliseconds. If a system request-to-send is detected, the keyboard counts 11 bits. After the 10th bit, the keyboard checks for an active level on the "data" line, and if the line is active, forces it inactive, and counts one more bit. This action signals the system that the keyboard has received its data. Upon receipt of this signal, the system returns to a ready state, in which it can accept keyboard output, or goes to the inhibited state until it is ready.

If the keyboard "data" line is found at an inactive level following the 10th bit, a framing error has occurred, and the keyboard continues to count until the "data" line becomes active. The keyboard then makes the "data" line inactive and sends a Resend.

Each system command or data transmission to the keyboard requires a response from the keyboard before the system can send its next output. The keyboard will respond within 20 milliseconds unless the system prevents keyboard output. If the keyboard response is invalid or has a parity error, the system sends the command or data again. However, the two byte commands require special handling. If Set Typematic Rate/Delay (hex F3), Select Alternate Scan Codes (hex F0), or Set/Reset Mode Indicators (hex ED) has been sent and acknowledged, and the value byte has been sent but the response is invalid or has a parity error, the system will resend both the command and the value byte.

Encode and Usage

The keyboard routine, provided by IBM in the ROM BIOS, is responsible for converting the keyboard scan codes into *extended ASCII*. The extended ASCII codes returned by the ROM routine are mapped to the U.S. English keyboard chart. Some operating systems can make provisions for alternate keyboard charts by providing an interrupt replacement routine that resides in the RAM. This section discusses only the ROM routine.

Extended ASCII encompasses 1-byte character codes with possible values of 0 to 255, an extended code for certain extended keyboard functions, and functions handled within the keyboard routine or through interrupts.

The character codes are passed through the BIOS keyboard routine to the system or application program. A -1 means the combination is suppressed in the keyboard routine. The codes are returned in the AL register. See Section 11, "Characters and Keystrokes" for the codes.

Figure 5-21 (Page 1 of 3). Character Codes

Key	Base Case	Uppercase	Ctrl	Alt
1	`	~	-1	(*)
2	1	!	-1	(*)
3	2	@	Null(000) (*)	(*)
4	3	#	-1	(*)
5	4	$	-1	(*)
6	5	%	-1	(*)
7	6	^	RS(030)	(*)
8	7	&	-1	(*)
9	8	*	-1	(*)
10	9	(	-1	(*)
11	0	)	-1	(*)
12	-	_	US(031)	(*)
13	=	+	-1	(*)
15	Backspace (008)	Backspace (008)	Del(127)	(*)
16	Tab (009)	Back-tab (*)	(*)	(*)
17	q	Q	DC1(017)	(*)
18	w	W	ETB(023)	(*)
19	e	E	ENQ(005)	(*)
20	r	R	DC2(018)	(*)
21	t	T	DC4(020)	(*)
22	y	Y	EM(025)	(*)
23	u	U	NAK(021)	(*)
24	i	I	HT(009)	(*)

Figure 5-21 (Page 2 of 3). Character Codes

Key	Base Case	Uppercase	Ctrl	Alt
25	o	O	SI(015)	(*)
26	p	P	DLE(016)	(*)
27	[	{	Esc(027)	(*)
28	]	}	GS(029)	(*)
29	\	|	FS(028)	(*)
30 Caps Lock	-1	-1	-1	-1
31	a	A	SOH(001)	(*)
32	s	S	DC3(019)	(*)
33	d	D	EOT(004)	(*)
34	f	F	ACK(006)	(*)
35	g	G	BEL(007)	(*)
36	h	H	BS(008)	(*)
37	j	J	LF(010)	(*)
38	k	K	VT(011)	(*)
39	l	L	FF(012)	(*)
40	;	:	-1	(*)
41	'	"	-1	(*)
43	CR(013)	CR(013)	LF(010)	(*)
44 Shift (Left)	-1	-1	-1	-1
46	z	Z	SUB(026)	(*)
47	x	X	CAN(024)	(*)
48	c	C	ETX(003)	(*)
49	v	V	SYN(022)	(*)
50	b	B	STX(002)	(*)
51	n	N	SO(014)	(*)
52	m	M	CR(013)	(*)
53	,	<	-1	(*)
54	.	>	-1	(*)
55	/	?	-1	(*)
57 Shift (Right)	-1	-1	-1	-1
58 Ctrl (Left)	-1	-1	-1	-1
60 Alt (Left)	-1	-1	-1	-1
61	Space	Space	Space	Space
62 Alt (Right)	-1	-1	-1	-1
64 Ctrl (Right)	-1	-1	-1	-1
90 Num Lock	-1	-1	-1	-1
95	/	/	(*)	(*)
100	*	*	(*)	(*)
105	-	-	(*)	(*)
106	+	+	(*)	(*)
108	Enter	Enter	LF(010)	(*)
110	Esc	Esc	Esc	(*)
112	Null (*)	Null (*)	Null (*)	Null(*)
113	Null (*)	Null (*)	Null (*)	Null(*)
114	Null (*)	Null (*)	Null (*)	Null(*)
115	Null (*)	Null (*)	Null (*)	Null(*)

Figure 5-21 (Page 3 of 3). Character Codes

Key	Base Case	Uppercase	Ctrl	Alt
116	Null (*)	Null (*)	Null (*)	Null(*)
117	Null (*)	Null (*)	Null (*)	Null(*)
118	Null (*)	Null (*)	Null (*)	Null(*)
119	Null (*)	Null (*)	Null (*)	Null(*)
120	Null (*)	Null (*)	Null (*)	Null(*)
121	Null (*)	Null (*)	Null (*)	Null(*)
122	Null (*)	Null (*)	Null (*)	Null(*)
123	Null (*)	Null (*)	Null (*)	Null(*)
125 Scroll Lock	-1	84/85 key NumLk	-1	-1
126	Pause(**)	Pause(**)	Break(**)	Pause(**)

(*) Refer to Extended Functions in this section.
(**) Refer to Special Handling in this section.

Figure 5-22 lists the keys that have meaning only in Num Lock, Shift, or Ctrl states.

The Shift key temporarily reverses the current Num Lock state.

Figure 5-22. Special Character Codes

Key	Num Lock	Base Case	Alt	Ctrl
91	7	Home (*)	-1	Clear Screen
92	4	← (*)	-1	Reverse Word(*)
93	1	End (*)	-1	Erase to EOL(*)
96	8	↑ (*)	-1	(*)
97	5	(*)	-1	(*)
98	2	↓ (*)	-1	(*)
99	0	Ins	-1	(*)
101	9	PgUp (*)	-1	Top of Text and Home
102	6	→ (*)	-1	Advance Word (*)
103	3	PgDn (*)	-1	Erase to EOS (*)
104	.	Del (*,**)	(**)	(**)
105	-	SysRq	-1	-1
106	+	+ (*)	-1	-1

(*) Refer to "Extended Functions" in this section.
(**) Refer to "Special Handling" in this section.

Extended Functions

For certain functions that cannot be represented by a standard ASCII code, an extended code is used. A character code of 00H (null) is returned in AL. This indicates that the system or application program should examine a second code, which will indicate the actual function. Usually, this second code is the scan code of the primary key that was pressed. This code is returned in AH.

Figure 5-23 is a list of the extended codes and their functions.

Figure 5-23 (Page 1 of 2). Keyboard Extended Functions

Second Code	Function
1	Alt Esc
3	Null Character
14	Alt Backspace
15	Back-tab
16-25	Alt Q, W, E, R, T, Y, U, I, O, P
26-28	Alt [ ] ↵
30-38	Alt A, S, D, F, G, H, J, K, L
39-41	Alt ; ' `
43	Alt \
44-50	Alt Z, X, C, V, B, N, M
51-53	Alt , . /
55	Alt Keypad *
59-68	F1 to F10 Function Keys (Base Case)
71	Home
72	↑ (Cursor Up)
73	Page Up
74	Alt Keypad -
75	← (Cursor Left)
76	Center Cursor
77	→ (Cursor Right)
78	Alt Keypad +
79	End
80	↓ (Cursor Down)
81	Page Down
82	Ins (Insert)
83	Del (Delete)
84-93	Shift F1 to F10
94-103	Ctrl F1 to F10
104-113	Alt F1 to F10
114	Ctrl Print Screen (Start/Stop Echo to Printer)
115	Ctrl ← (Reverse Word)
116	Ctrl → (Advance Word)
117	Ctrl End (Erase to End of Line-EOL)
118	Ctrl PgDn (Erase to End of Screen-EOS)
119	Ctrl Home (Clear Screen and Home)
120-131	Alt 1, 2, 3, 4, 5, 6, 7, 8, 9, 0, -, = keys 2-13
132	Ctrl PgUp (Top 25 Lines of Text and Cursor Home)
133-134	F11, F12

Figure 5-23 (Page 2 of 2). Keyboard Extended Functions

Second Code	Function
135-136	Shift F11, F12
137-138	Ctrl F11, F12
139-140	Alt F11, F12
141	Ctrl Up/8
142	Ctrl Keypad -
143	Ctrl Keypad 5
144	Ctrl Keypad +
145	Ctrl Down/2
146	Ctrl Ins/0
147	Ctrl Del/.
148	Ctrl Tab
149	Ctrl Keypad /
150	Ctrl Keypad *
151	Alt Home
152	Alt Up
153	Alt Page Up
155	Alt Left
157	Alt Right
159	Alt End
160	Alt Down
161	Alt Page Down
162	Alt Insert
163	Alt Delete
164	Alt Keypad /
165	Alt Tab
166	Alt Enter

Shift States

Most shift states are handled within the keyboard routine and are not apparent to the system or application program. In any case, the current status of active shift states is available by calling a software interrupt in the BIOS keyboard routine. The following keys result in altered shift states:

Shift: This key temporarily shifts keys 1 through 13, 16 through 29, 31 through 41, and 46 through 55, to uppercase (base case if in Caps Lock state). Also, the Shift temporarily reverses the Num Lock or non-Num Lock state of keys 91 through 93, 96, 98, 99, and 101 through 104 on a PC Enhanced keyboard. If in Num Lock state, the Shift key temporarily invokes the cursor functions of keys 8 through 10, 12, 13, 23 through 25, 37 through 40, 52, and 54 on a Space Saving keyboard.

Control (Ctrl): This key temporarily shifts keys 3, 7, 12, 15 through 29, 31 through 39, 43, 46 through 52, 75 through 89, 91 through 93, 95 through 108, 112 through 124, and 126 to the Ctrl state. The Ctrl key is also used with the Alt and Delete keys to cause the system reset

function and with the Pause/Break key to cause the break function. The system reset and break functions are described under "Special Handling" on page 5-48.

Alternate (Alt): This key temporarily shifts keys 1 through 29, 31 through 43, 46 through 55, 75 through 89, 95, 100, and 105 through 124 to the Alt state. The Alt key is also used with the Ctrl and Delete keys to cause a system reset.

The Alt key also allows the user to enter any character code from 1 to 255. The user holds down the Alt key and types the decimal value of the desired character on the numeric keypad. The Alt key is then released. If the number is greater than 255, a modulo-256 value is used. This value is interpreted as a character code and is sent through the keyboard routine to the system or application program. Alt is handled in the keyboard routine.

Caps Lock: This key shifts keys 17 through 26, 31 through 39, and 46 through 52 to uppercase. When Caps Lock is pressed again, it reverses the action. Caps Lock is handled in the keyboard routine. When Caps Lock is pressed, it changes the Caps Lock Mode indicator. If the indicator was on, it will go off; if it was off, it will go on.

Scroll Lock or ScrLk: When interpreted by appropriate application programs, this key indicates that the cursor control keys will cause windowing over the text rather than moving the cursor. When the Scroll Lock (ScrLk) key is pressed again, it reverses the action. The keyboard routine simply records the current shift state of the key. It is the responsibility of the application program to perform the function. When Scroll Lock is pressed, it changes the Scroll Lock Mode indicator. If the indicator was on, it will go off; if it was off, it will go on.

Num Lock or NumLk: For the PC Enhanced keyboard, this key shifts keys 91 through 93, 96 through 99, and 101 through 104 to uppercase. When Num Lock is pressed again, it reverses the action.

For the Space Saving keyboard, this key shifts keys 8 through 10, 12, 13, 37 through 40, 23 through 25, and 52 through 54. The NumLk for this keyboard is Shift plus the ScrLk key.

Num Lock for both keyboards is handled in the keyboard routine.

Shift Key Priorities and Combinations: If combinations of the Alt, Ctrl, and Shift keys are pressed and only one is valid, the priority is as follows: the Alt key is first, the Ctrl key is second, and the Shift key is third. The only valid combination is Alt and Ctrl, which is used in the system reset function.

Special Handling

System Reset

The combination of any Alt, Ctrl, and Delete keys results in the keyboard routine that starts a system reset or restart. System reset is handled by BIOS.

Break

The combination of the Ctrl and Pause/Break keys results in the keyboard routine signaling INT 1BH. The extended characters (AL) = hex 00 and (AH) = hex 00 are also returned.

Pause

The Pause key causes the keyboard interrupt routine to loop, waiting for any character or function key to be pressed. This provides a method of temporarily suspending an operation, such as listing or printing, and then resuming the operation. The method is not apparent to either the system or the application program. The keystroke used to resume operation is discarded. Pause is handled in the keyboard routine.

Print Screen

The Print Screen key results in an interrupt invoking the print-screen routine. This routine works in the alphanumeric or graphics mode, with unrecognizable characters printing as blanks.

System Request (SysRq)

When the SysRq (Alt and Print Screen) key is pressed, a hex 8500 is placed in AX, and an INT 15H is executed. When the SysRq key is released, a hex 8501 is placed in AX, and another INT 15H is executed. If an application uses the key, the following steps must be performed:

1. Save the previous address.
2. Overlay interrupt vector hex 15.
3. Check AH for a value of hex 85:

   If yes, process may begin.
   If no, go to previous address.

The application program must preserve the values in all registers, except AX, upon return. A system request is handled in the keyboard routine.

Other Characteristics

The keyboard routine does its own buffering. If a key is pressed when the buffer is full, the key is ignored and the alarm is activated.

The keyboard routine also suppresses the typematic action of the following keys: Ctrl, Shift, Alt, Num Lock (NumLk), Scroll Lock (ScrLk), Caps Lock, and Ins (Insert).

During each INT 09H from the keyboard, an INT 15H, function (AH) = hex 4F is generated by the BIOS after the scan code is read from the keyboard adapter. The scan code is passed in the (AL) register with the carry flag set. This is to enable an operating system to intercept each scan code prior to its being handled by the INT 09H routine, and have a chance to change or act on the scan code. If the carry flag is changed to 0 on return from INT 15H, the scan code will be ignored by the interrupt handler.

Cables and Connectors

The keyboard cable connects to the system with a 6-pin miniature DIN connector. Figure 5-24 shows the pin configuration and signal assignments.

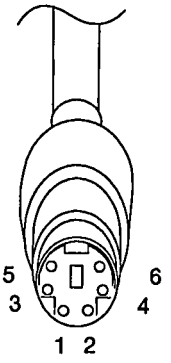


Figure 5-24. Keyboard Connector Signal and Voltage Assignments

DIN Connector Pins	Signal Name
1	+Data
2	Reserved
3	Ground
4	+5.0 V dc
5	+Clock
6	Reserved
Shield	Frame Ground

Specifications

The following are specifications for the keyboard.

	Keyboard	
Power Requirements	+5 V dc ± 10% 275 milliamperes	
Size	**Millimeters**	**Inches**
• Length:	450	17.7
• Depth:	158	6.2
• Height:	39	2.3
Legs extended	55	2.2
Weight	**Kilograms**	**Pounds**
	0.93	2.05

Section 6. Mouse

Description 6-3
Programming Considerations 6-4
  Commands 6-4
  Data Report 6-9
  Error Handling 6-10
  Data Transmission 6-10
Mouse Device Driver Interface 6-12
  Call to Device Driver 6-14
  Device Driver Functions 6-15
Connector 6-32
Specifications 6-33

Description

The Mouse is a cursor-positioning device that uses a ball and two encoders to indicate *x* and *y* movement to the system. Two push-button switches transmit their states directly to the system. The Mouse is connected to the system with a 1.8-meter (6-foot), shielded, four-conductor cable. The ball is removable for cleaning.

The following are descriptions of the four modes of operation:

Mode	Description
Reset	In this mode a self-test is initiated during power-on or by a Reset command. Upon satisfactory completion of the diagnostics, a completion code (hex AA) and an ID Code (hex 00) are transmitted to the system. The following defaults are set: sampling rate of 100 reports per second, linear scaling, stream mode, resolution of four counts per millimeter, and the Mouse is disabled. Any commands sent before the transmission of the completion code and the ID byte are ignored. The Mouse sends an error code of hex FC followed by an ID code of hex 00 immediately following a failure to complete the diagnostics. At this time, the Mouse is disabled and awaits further command input from the system.
Stream	In this mode, a data report is transmitted to the system if a switch is pressed or released, or if at least one count of movement has been detected. The maximum rate of transfer is the programmed sample rate. No transmissions occur if the Mouse is motionless unless a switch is operated, in which case, the incremental movement report is zero.
Remote	In this mode, data is transmitted only in response to a Read Data command.
Wrap	In this mode, any byte of data sent by the system, except hex EC and hex FF, is returned by the Mouse.

Programming Considerations

This section describes the Mouse commands and discusses the interface.

Commands

The ACK (hex FA) is always the first response to any valid input received from the system other than a Set Wrap Mode or Resend command, and supersedes all other Mouse output. If an interruption occurs during output of an ACK, the Mouse discards the ACK and accepts and responds to the new command. The ACK response is considered part of a special protocol between the system and the Mouse, and is not stored in any buffered internal memory. It is transmitted when required, then discarded.

The following lists all the valid commands.

Figure 6-1. Mouse Commands

Hex Code	Command
FF	Reset
FE	Resend
F6	Set Default
F5	Disable
F4	Enable
F3	Set Sampling Rate
F2	Read Device Type
F0	Set Remote Mode
EE	Set Wrap Mode
EC	Reset Wrap Mode
EB	Read Data
EA	Set Stream Mode
E9	Status Request
E8	Set Resolution
E7	Set Scaling 2:1
E6	Reset Scaling

The following describes valid commands:

Hex Code	Description
FF	**Reset:** This command causes the Mouse to enter the reset mode and do an internal self-test.
FE	**Resend:** Any time the Mouse receives an invalid command, it returns a Resend command to the system. The Mouse continues operating in the state it was in before receiving the invalid command. The count accumulators are cleared after receiving any command other than a Resend. The system sends this command when it detects an error in any transmission from the Mouse. The Resend command is sent following the Mouse transmission and before the system enables the interface, allowing the next output. When the Mouse receives a Resend command, it retransmits its last packet of data. If the last packet was a Resend command, it transmits the packet just prior to the Resend command. In the Stream mode, if a Resend command is received by the Mouse immediately following a 3-byte data packet transmission to the system, the Mouse resends the 3-byte data packet prior to clearing the count accumulators. This command also is described in "Error Handling" on page 6-10.
F6	**Set Default:** This command reinitializes all conditions to the power-on default state. Following receipt of this command, the Mouse sets up for a sampling rate of 100 reports per second, linear scaling, Stream mode, four counts per millimeter resolution, and disabled. No further action occurs until another command is sent from the system.
F5	**Disable:** This command is used in the Stream mode to stop transmissions initiated from the Mouse. It responds to all other commands while disabled. If the Mouse is in the Stream mode, it must be disabled before sending it any command that requires a response.
F4	**Enable:** This command is used in the Stream mode to begin transmissions.

F3, XX **Set Sampling Rate:** In the Stream mode, this command sets the sampling rate to the value indicated by the byte shown in the following figure.

Figure 6-2. Sampling Rate

Second Byte (in Hex)	Sample Rate
0A	10 per Second
14	20 per Second
28	40 per Second
3C	60 per Second
50	80 per Second
64	100 per Second
C8	200 per Second

F2 **Read Device Type:** This command always receives a response of hex 00.

F0 **Set Remote Mode:** This command sets the Remote mode. Data values are reported only in response to a Read Data command.

EE **Set Wrap Mode:** This command sets the Wrap mode. This mode remains until hex FF or hex EC is received.

EC **Reset Wrap Mode:** This command resets the Wrap mode. The Mouse returns to the previous mode of operation after receiving this command.

Note: If the Mouse is in the Stream mode and the Wrap mode is entered, the Reset Mode command causes the Mouse to reenter the Stream Mode in a disabled state.

EB **Read Data:** This command requests that all data defined in the data packet format be transmitted. This command is executed in either remote or stream mode. The data is transmitted even if there has been no movement since the last report or the switch status is unchanged. Following a Read Data command, the accumulators are cleared after a data transmission.

EA **Set Stream Mode:** This command sets the Stream mode.

E9 **Status Request:** When this command is issued by the system, the Mouse responds with a 3-byte status report as follows.

Figure 6-3. Status Request Format

Byte	Bit	Description
1	7	Reserved
	6	0 = Stream Mode, 1 = Remote Mode
	5	0 = Disabled, 1 = Enabled
	4	0 = Scaling 1:1, 1 = Scaling 2:1
	3	Reserved
	2	1 = Left Button Pressed
	1	Reserved
	0	1 = Right Button Pressed
2	7 - 0	Current Resolution Setting
3	7 - 0	Current Sampling Rate

E8, XX **Set Resolution:** The Mouse provides four resolutions selected by the second byte of this command as follows.

Figure 6-4. Set Resolution

Second Byte (in Hex)	Resolution (Counts per mm)
00	1
01	2
02	4
03	8

E7 **Set Scaling 2:1:** Scaling is used to provide a course or fine tracking response. At the end of a sample interval in the Stream mode, the current *x* and *y* data values are converted to new values. The sign bits are not involved in this conversion. The relationship between the input and output counts follows.

Figure 6-5. Set Scaling 2:1

Input	Output
0	0
1	1
2	1
3	3
4	6
5	9
N (≥6)	2.0 x N

2:1 scaling is performed only in Stream mode. In response to a Read Data command, the current value before conversion is sent.

E6 **Reset Scaling:** This command restores scaling to 1:1.

Data Report

When operating in Stream mode, a data report is sent at the end of a sample interval if a button remains pressed or is released during the interval, or if at least one count of Mouse movement has occurred since the last report.

If a button is pressed during a sample interval, it is reported as pressed at the end of the interval. If a button remains pressed, no further reports are transmitted until it is released unless there is further Mouse movement to report. When movement is to be reported and there has been no change in the button status during the last interval, the buttons are reported in their current state (1 = pressed, 0 = not pressed). If a button is pressed and released during a sample interval, it is reported as pressed at the end of the interval. Any transmission reporting button status change can also include travel data.

In the Remote mode, a data report is sent in response to a Read Data command. The buttons are reported in their current state at the time of transmission.

The following data report format is valid for both the Stream and Remote modes and is three bytes long.

Figure 6-6. Data Packet Report Format

Byte	Bit	Description
1	7	*y* Data Overflow 1 = Overflow
	6	*x* Data Overflow 1 = Overflow
	5	*y* Data sign 1 = Negative
	4	*x* Data sign 1 = Negative
	3	Reserved
	2	Reserved
	1	Right Button Status 1 = Pressed
	0	Left Button Status 1 = Pressed
2	7 - 0	*X* Data
3	7 - 0	*y* Data

The data values are in binary and the least significant bit (LSB) indicates 0.25 millimeter of movement when operating with linear scaling at four counts per millimeter resolution. Negative values of *x* and *y* data are expressed in twos complement where 0.25 millimeter movement in the negative direction is expressed with all bits set to 1 and the sign bit set to 1. The full movement number is a 9-bit twos-complement number.

The count accumulators do not wrap around. If a count during a sample period is greater than the format allows, the maximum count is reported and the overflow bit for that coordinate is set. After a transmission, the accumulators are cleared to 0.

Error Handling

The Mouse issues a Resend command (hex FE) following receipt of an invalid input or any input with incorrect polarity. If two invalid inputs are received in succession, an error code of hex FC is sent to the system.

Following a system transmission, a response is sent within 25 milliseconds if the system requires a response or if an error is detected in the transmission. If the Mouse is in the Stream mode, the system disables the Mouse before issuing any command requiring a response. When a command requiring a response is issued by the system, another command should not be issued until either the response is received or 25 milliseconds has elapsed. No more than four non-response commands should be sent in succession.

Data Transmission

During a data transmission, CLK is used to clock serial data. The Mouse generates the clocking signal when sending data to and receiving data from the system. The system requests the Mouse receive system data output by forcing the DATA line low and allowing CLK to go high.

Communication is bi-directional using the CLK and DATA signal lines. The signal for each of these lines comes from open collector devices, allowing either the Mouse or the system to drive a line low. During a non-transmission state, CLK and DATA are both held high.

Data Output: When the Mouse is ready to transmit data, it must first check for its own Inhibit or system request-to-send status on the CLK and DATA lines. If CLK is low (inhibit status), data is continuously updated and no transmissions are started. If CLK is high and DATA is low (request-to-send), data is updated. Data is received from the system and no transmissions are started by the Mouse until CLK and DATA are both high.

If CLK and DATA are both high, the Mouse proceeds to output zero start-bits, eight data bits, a parity bit, and a stop bit if a transmission is required. Data is valid prior to the falling edge of CLK and beyond

the rising edge of CLK. During transmission, the Mouse checks for line contention by checking for a low level on CLK at intervals not to exceed 100 milliseconds. Contention occurs when the system drives CLK low to inhibit the Mouse output after the Mouse has started a transmission. If this occurs before the rising edge of the tenth clock (parity bit), the Mouse internally stores the data packet in the its buffer and returns DATA and CLK to a high level. If the contention does not occur by the tenth clock, the transmission is completed.

Following a transmission, the system inhibits the Mouse by holding CLK low until it can service the input or until the system receives a request to send a response from the Mouse. The system raises CLK to allow the next transmission.

Data Input: When the system is ready to send data, it first checks to see if the Mouse is transmitting data. If the Mouse is transmitting, the system can override the output by driving CLK low prior to the tenth clock. If the Mouse transmission is beyond the tenth clock, the system receives the data.

If the Mouse is not transmitting or if the system chooses to override the output, the system drives CLK low for a period of not less than 100 microseconds while preparing for output. When the system is ready to output 0 start bit (DATA line is low), it allows CLK to go to an active level. The Mouse checks for this state not to exceed every 10 milliseconds.

If request-to-send is detected, the Mouse clocks in 11 bits. Following the tenth clock, the Mouse checks for DATA being high, and if found, then drives DATA low (line control bit), and clocks once more. This signals the system to return to the ready state when it can accept input or go to the Inhibit mode until ready. If DATA is low following clock 10, a framing error has occurred and the Mouse continues to clock until DATA is high, then clocks the line control bit and requests a resend.

For each system command or data transmission that requires a response, the system waits for the Mouse to respond before sending its next output. The response must be within 20 milliseconds, unless the system inhibits the Mouse output or inhibits the data transmissions from the system that require a response.

If the system initiates a command or data transmission and the response is invalid or has a parity error, the system resends the command or data. If after two retries the response is still invalid or has a parity error, the system resets the Mouse.

Figure 6-7. Data Frame

Bit	Function
Start Bit	Always 0
0	Least Significant Bit (LSB)
1 - 6	Data Bits 1-6
7	Most Significant Bit (MSB)
Parity Bit	Odd Parity
Stop Bit	Always 1

Mouse Device Driver Interface

The function of the device driver is to allow the Mouse to operate with applications that use the interface designed by Microsoft® . The device driver can only be loaded using DOS commands and cannot be loaded as a DOS device driver from the CONFIG.SYS file.

Applications access the device driver by issuing an interrupt hex 33. The device driver determines which function to perform by the value in the AX register. Parameters are passed from the calling application to the device driver in the BX, CX and DX registers (Function 16 also uses the SI and DI registers). High-level programming languages can access the device driver by making a call to the entry point of the device driver.

To make a call from a BASIC program you must:

- Assign the offset and segment address of the software to a pair of integer variables in your program. The entry offset and segment address are in memory. To get these values, insert the following statements into your program:

```
10 DEF SEG=0
20 MSEG=256*PEEK(51*4+3)+PEEK(51*4+2)
30 MOUSE=256*PEEK(51*4+1)+PEEK(51*4)
40 IF MSEG OR MOUSE THEN 60
50 PRINT "Mouse Driver not found":END
60 DEF SEG=MSEG
70 IF PEEK (MOUSE)=207 THEN 50 '207 is IRET
80 'Mouse driver is there, continue
```

  Be sure these statements appear before making any calls to Mouse functions.

Microsoft is a registered trademark of the Microsoft Corporation.

- Use the CALL statement to make the call. The statement should have the form:

  ```
  CALL MOUSE (M1%,M2%,M3%,M4%)
  ```

  where MOUSE is the variable containing the entry offset of the Mouse software and M1%, M2%, M3%, and M4% are the names of the integer variables you have chosen for parameters in this call (constants and non-integer variables are not allowed). All four parameters must appear in the CALL statement, even if no value is assigned to one or more of them.

  Set the values for M1%, M2%, M3% and M4% to the AX, BX, CX, and DX registers described in the device driver functions.

  To ensure the variables are integer variables, use the percent sign (%) as part of all the variable names. You may also use the DEFINT statement at the beginning of your program. For example, the statement

  ```
  10 DEFINT A-Z
  ```

  defines all variables as integer. If this statement appears at the beginning of the program, the variable names do not need to include the percent sign.

  An application gets status by continuously polling the device driver for Mouse movement and button position. The device driver also supports the ability of an application to hook in a subroutine to be called whenever a Mouse event occurs. Status is passed to the subroutine when it is called from the device driver.

To interface with the device driver from an assembler language program, set up the registers specified in the following figures according to the desired function and issue an interrupt hex 33 as specified in the following figures.

Call to Device Driver

The device driver receives data from the Mouse whenever the Mouse is moved or whenever a button position changes. During initial setup of the Mouse and the device driver, a pointer to a subroutine in the device driver is passed to BIOS through INT 15 AH = C2 AL = 7. The subroutine is called whenever Mouse movement is detected or a button position changes.

The following data is passed to the device driver subroutine on the stack.

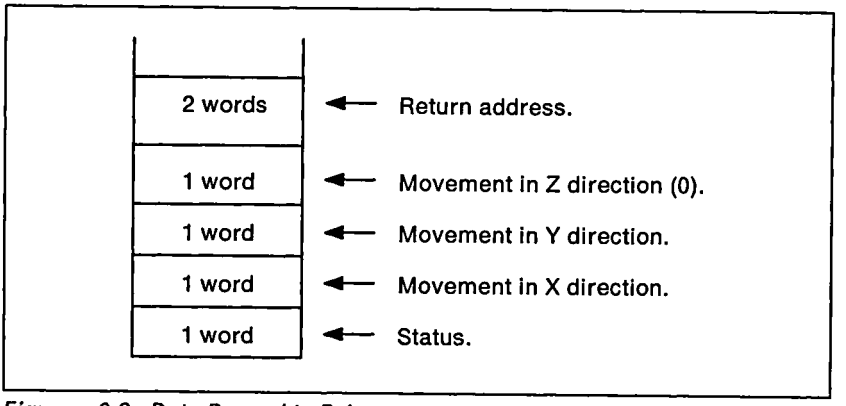


Figure 6-8. Data Passed to Driver

The data is read from the stack and processed. While the data is being processed, no more data will be sent by the Mouse BIOS. If a device driver function is currently in progress the data is lost.

Device Driver Functions

Function 0: Installed Flag and Reset: This function initializes variables and initializes the Mouse if it is attached. If the Mouse is not attached, the device driver sends an unsuccessful return code to the calling program.

Figure 6-9. Function 0: Installed Flag and Reset

Input		Output	
Register	**Value**	**Register**	**Value**
AX	0	AX	Mouse Status 0 = Unsuccessful -1 = Successful
		BX	Number of Buttons 2 If successful 0 Otherwise

The following device driver variables are initialized.

Figure 6-10. Device Driver Variables

Description	**Value**
xy Movements	0
Button Presses	0
Button Releases	0
Cursor Position at Last Button Press	0
Cursor Position at Last Button Release	0
Cursor Position	Screen Center
Conditional Off Region	Hex 07FFF
Scaling Factor (Horizontal)	8
Scaling Factor (Vertical)	16
Text Cursor	Inverted Box
Graphics Cursor	Arrow
Hot Spot	(-1,-1)
Min/Max Cursor Position	Depends on Video Mode

The Mouse is initialized to the following values.

Figure 6-11. Mouse Initial Values

Description	**Value**
Sample Rate	100 Reports per Second
Resolution	8 Counts per mm
Data Package Size	3 Bytes
Scaling	2:1

Function 1: Show Cursor: The internal cursor flag is incremented; if it is equal to 0, the cursor is displayed. This function always increments the cursor flag. Upon Reset the flag is set to -1, the first show will display a cursor. Additional show calls will increment the cursor flag to positive values.

Note: Hide Cursor calls should be paired with Show Cursor calls.

Figure 6-12. Function 1: Show Cursor

Input		Output	
Register	**Value**	**Register**	**Value**
AX	1	—	—

Function 2: Hide Cursor: The internal cursor flag is decremented and the cursor is removed from the screen if it is displayed. For every Hide Cursor function call made, a Show Cursor function call must be made to increment the cursor flag.

Figure 6-13. Function 2: Hide Cursor

Input		Output	
Register	**Value**	**Register**	**Value**
AX	2	—	—

Function 3: Get Position and Button Status: The current horizontal and vertical cursor positions and the button status are returned to the calling program. Both horizontal and vertical cursor positions, are returned as "virtual screen coordinates." They might not map directly into the video (graphics) mode.

Figure 6-14. Function 3: Get Position and Button Status

Input		Output	
Register	**Value**	**Register**	**Value**
AX	3	BX	Button Status Bit 1 1 = Right Button Pressed 0 = Right Button Released Bit 0 1 = Left Button Pressed 0 = Left Button Released
		CX	Cursor Position (Horizontal)
		DX	Cursor Position (Vertical)

Function 4: Set Cursor Position: This function sets the horizontal and vertical cursor positions.

Figure 6-15. Function 4: Set Cursor Position

Input		Output	
Register	**Value**	**Register**	**Value**
AX	4	—	—
CX	New Horizontal Coordinate	—	—
DX	New Vertical Coordinate	—	—

Function 5: Get Button Press Information: This function returns the following information, for the specified Mouse button, to the calling program:

- The status of the button
- The number of times the button was pressed since the last call to this function
- The cursor position of the last button pressed.

Figure 6-16. Function 5: Get Button Press Information

Input		Output	
Register	**Value**	**Register**	**Value**
AX	5	AX	Button Status
BX	Button (0 = Left, 1 = Right)	BX	Count of Button Presses
		CX	Cursor (Horizontal) at Last Press
		DX	Cursor (Vertical) at Last Press

Function 6: Get Button Release Information: This function returns the following information, for the specified Mouse button, to the calling program:

- The status of the button
- The number of times the button was released since the last call to this function
- The cursor position of the last button release.

Figure 6-17. Function 6: Get Button Release Information

Input		Output	
Register	**Value**	**Register**	**Value**
AX	6	AX	Button Status
BX	Button (0 = Left, 1 = Right)	BX	Count of Button Releases
		CX	Cursor (Horizontal) at Last Release
		DX	Cursor (Vertical) at Last Release

Function 7: Set Minimum and Maximum Horizontal Position: This function sets the minimum and maximum values on the x-axis. If the minimum value specified is larger than the maximum value, the values are exchanged.

If the maximum value is larger than allowed, the value is set to the maximum value allowed. This also applies to the minimum value. The minimum and maximum values are defined by the graphic mode.

Figure 6-18. Function 7: Set Minimum and Maximum Horizontal Position

Input		Output	
Register	**Value**	**Register**	**Value**
AX	7	—	—
CX	Minimum Position	—	—
DX	Maximum Position	—	—

Function 8: Set Minimum and Maximum Vertical Position: This function sets the minimum and maximum values on the *y*-axis. If the minimum value specified is larger than the maximum value, the values are exchanged.

If the maximum value is larger than allowed, the value is set to the maximum value allowed. This also applies to the minimum value. The minimum and maximum values are defined by the graphic mode.

Figure 6-19. Function 8: Set Minimum and Maximum Vertical Position

Input		Output	
Register	**Value**	**Register**	**Value**
AX	8	—	—
CX	Minimum Position	—	—
DX	Maximum Position	—	—

Function 9: Set Graphics Cursor Block: This function is used to define the screen mask, cursor mask, and hot spot for the graphics cursor. The cursor masks are 16-bit by 16-bit matrixes used to determine the shape and color of the graphics cursor. In a 640 by XXX (XXX can be 200, 350, or 480) graphics mode, one bit in each mask represents a PEL. In a 320 by XXX (XXX can be 200 or 350) graphics mode, two horizontal bits represent a PEL.

Figure 6-20. Function 9: Set Graphics Cursor Block

Input		Output	
Register	**Value**	**Register**	**Value**
AX	9	—	—
BX	Horizontal Cursor Hot Spot (-16 to 16)	—	—
CX	Vertical Cursor Hot Spot (-16 to 16)	—	—
DX	Pointer to Screen and Cursor Masks	—	—

The graphics cursor is defined as follows for graphics modes less than mode 7.

Figure 6-21. Graphics Modes Less Than 7

Screen Mask	Cursor Mask	Result On Screen
0	0	0
0	1	1
1	0	Unchanged
1	1	Inverted

The graphics cursor is defined as follows for graphics modes greater than mode 7.

Figure 6-22. Graphics Modes Greater Than 7

Screen Mask	Cursor Mask	Result On Screen
0	0	Black
0	1	White
1	0	Unchanged
1	1	Unchanged

Function 10: Set Text Cursor: This function defines the cursor and screen mask for the text cursor and defines the text cursor as the hardware or software cursor.

Figure 6-23. Function 10: Set Text Cursor

Input		Output	
Register	**Value**	**Register**	**Value**
AX	10	—	—
BX	Cursor Select 0 = Software text cursor 1 = Hardware text cursor	—	—
CX	Screen Mask Value or Scan Line Start	—	—
DX	Cursor Mask Value or Scan Line Stop	—	—

Function 11: Read Motion Counters: This function returns the horizontal (*x*) and vertical (*y*) counts since the last time this function was called. The *x* and *y* are units of distance approximately equal to 0.5 millimeter (0.02 inch). A positive count indicates movement to the right or down. A negative count indicates movement to the left or up. The count is always within the range of -32768 to 32767.

Figure 6-24. Function 11: Read Motion Counters

Input		Output	
Register	**Value**	**Register**	**Value**
AX	11	CX	Count (Horizontal)
		DX	Count (Vertical)

Function 12: Set User-Defined Subroutine Input Mask: This function allows the calling program to set up a subroutine that will be called by the device driver when a condition in the mask occurs.

Figure 6-25. Function 12: Set User-Defined Subroutine Input Mask

Input		Output	
Register	**Value**	**Register**	**Value**
AX	12	—	—
CX	Call Mask	—	—
DX	Address Offset to Subroutine	—	—

The Mouse hardware interrupts automatically stop execution of the program and call the specified subroutine when one or more of the conditions defined by the Call Mask occur. When the subroutine finishes, the program continues execution from the point of interruption.

The Call Mask is an integer value that defines conditions that cause an interrupt. Each bit in the Call Mask corresponds to a specific condition, as shown in the following.

Figure 6-26. Call Mask Bit Definitions

Mask Bit	Condition
15 - 5	Not Used
4	Right Button Released
3	Right Button Pressed
2	Left Button Released
1	Left Button Pressed
0	Cursor Position Changes

To enable the subroutine, set the corresponding Call Mask bit to 1. To disable the subroutine, set the corresponding bit to 0. Function 0 automatically disables all interrupts.

Note: Before the program ends, restore initial values of the Call Mask and the subroutine address.

Function 13: Light Pen Emulation Mode On: When the Light Pen Emulation mode is turned on, the Mouse emulates a light pen. The cursor position represents the light pen position; pressing both buttons represents the light pen being pressed down.

Figure 6-27. Function 13: Light Pen Emulation Mode On

Input		Output	
Register	**Value**	**Register**	**Value**
AX	13	—	—

Function 14: Light Pen Emulation Mode Off: This function disables light pen emulation.

Figure 6-28. Function 14: Light Pen Emulation Mode Off

Input		Output	
Register	**Value**	**Register**	**Value**
AX	14	—	—

Function 15: Set xy/PEL Ratio: This function sets the horizontal and vertical ratios of Mouse movement to cursor movement (8 PEL). The default value for the horizontal ratio is 8 *xy* to 8 PEL. This ratio moves the cursor all the way across the screen when the Mouse moves 81 millimeters (3.2 inch). The default value for the vertical ratio is 16 *xy* to 8 PEL. This ratio moves the cursor all the way down the screen when the Mouse moves 50.1 millimeters (2.0 inch).

Figure 6-29. Function 15: Set xy/PEL Ratio

Input		Output	
Register	**Value**	**Register**	**Value**
AX	15	—	—
CX	*x* per 8 PEL Ratio (Horizontal)	—	—
DX	*y* per 8 PEL Ratio (Vertical)	—	—

Function 16: Conditional Off: This function defines an area on the screen that erases the cursor. Function 1 (Show Cursor) must be called after calling this function.

Figure 6-30. Function 16: Conditional Off

Input		Output	
Register	**Value**	**Register**	**Value**
AX	16	—	—
CX	Upper *x* Screen Coordinate	—	—
DX	Upper *y* Screen Coordinate	—	—
SI	Lower *x* Screen Coordinate	—	—
DI	Lower *y* Screen Coordinate	—	—

Function 19: Set Double Speed Threshold: This function sets a threshold speed (in *xy*'s per second), which if exceeded, causes the cursor to move twice the distance on the screen.

Figure 6-31. Function 19: Set Double Speed Threshold

Input		Output	
Register	**Value**	**Register**	**Value**
AX	19	—	—
DX	Threshold Speed in Movements per Second	—	—

Function 20: Swap User Interrupt Vector: This function sets new values for the Call Mask and the subroutine address and returns the values previously specified.

Figure 6-32. Function 20: Swap User Interrupt Vector

Input		Output	
Register	**Value**	**Register**	**Value**
AX	20	—	—
CX	New User Interrupt Event Mask	CX	Old User Interrupt Event Mask
ES:DX	New User Interrupt Vector	ES:DX	Old User Interrupt Vector

The Mouse hardware interrupts automatically stop execution of the program and call the specified subroutine when one or more of the conditions defined by the Call Mask occur. When the subroutine finishes, the program continues execution from the point of interruption.

The Call Mask is an integer value that defines conditions that cause an interrupt. Each bit in the Call Mask corresponds to a specific condition, as shown in the following.

Figure 6-33. Call Mask Bit Definitions

Mask Bit	Condition
15 - 5	Not Used
4	Right Button Released
3	Right Button Pressed
2	Left Button Released
1	Left Button Pressed
0	Cursor Position Changes

When the program makes a call to the subroutine, it loads the following information into the microprocessor registers.

Figure 6-34. Register Information

Register	Information
AX	Condition Mask (Similar to the Call Mask except a bit is set only if the condition has occurred)
BX	Button State (see Figure 14)
CX	Cursor Coordinate (Horizontal)
DX	Cursor Coordinate (Vertical)
DI	Horizontal Counts
SI	Vertical Counts

Note: The DS register contains the Mouse device driver data segments. The subroutine is responsible for setting the DS register as needed.

To enable the subroutine, set the corresponding Call Mask bit to 1 and pass the mask in the CX register. To disable the subroutine, set the corresponding bit to 0 and pass the mask in the CX register. Function 0 automatically disables all interrupts.

Note: Before the program ends, restore the initial values of the Call Mask and the subroutine address.

Function 21: Query Save State Storage Requirements: This function returns the size of the buffer required to store the current state of the Mouse device driver. It is used with functions 22 and 23 to temporarily interrupt a program using the Mouse and execute another program also using the Mouse.

Figure 6-35. Function 21: Query Save State Storage Requirements

Input		Output	
Register	**Value**	**Register**	**Value**
AX	21	—	—
—	—	BX	Buffer Size Required for Functions 22 and 23

Function 22: Save Mouse Driver State: This function saves the current Mouse device driver state in a buffer allocated by the program. It is used with functions 21 and 23 to temporarily interrupt a program using the Mouse and execute another program that also using the Mouse.

Before calling Function 22, the program should call Function 21 to determine the buffer size required for saving the Mouse device driver state, then allocate the appropriate amount of memory.

Figure 6-36. Function 22: Save Mouse Driver State

Input		Output	
Register	**Value**	**Register**	**Value**
AX	22	—	—
ES:DX	Pointer to the Buffer	—	—

Function 23: Restore Mouse Driver State: This function restores the last Mouse device driver state saved by Function 22. It is used with functions 21 and 22 to temporarily interrupt a program that also uses the Mouse. To restore the Mouse device driver state saved by function 22, call Function 23 at the end of the interrupt program.

Figure 6-37. Function 23: Restore Mouse Driver State

Input		Output	
Register	**Value**	**Register**	**Value**
AX	23	—	—
ES:DX	Pointer to the Buffer	—	—

Function 24: Set Alternate Mouse User Subroutine: This function allows the calling program to set up a subroutine that will be called by the device driver when a condition in the Event Mask occurs. This function is similar to Function 12 except the Event Mask can also include a combination of certain keystrokes. Up to three routines can be defined by calling this function. After the third call an error is returned.

Figure 6-38. Function 24: Set User-Defined Subroutine Input Mask

Input		Output	
Register	**Value**	**Register**	**Value**
AX	12	—	—
CX	Call Mask	—	—
DX	Address Offset to Subroutine	—	—

The register used when entering user subroutines are shown in the following figure.

Figure 6-39. Register Conventions

Register	Definition
AX	Condition Mask
BX	Button State (see Figure 14)
CX	Cursor Coordinate (Horizontal)
DX	Cursor Coordinate (Vertical)
SI	Delta *x* and *y* Movement (Horizontal)
DI	Delta *x* and *y* Movement (Vertical)

Note: The DS register contains the Mouse device driver data segments. The subroutine is responsible for setting the DS register as needed.

Bit definitions for the Event Mask are shown in the following figure.

Figure 6-40. Event Mask Bit Definitions

Bits	Definition
15 - 8	Reserved
7	Alt Key Pressed During Button Event
6	Ctrl Key Pressed During Button Event
5	Shift Key Pressed During Button Event
4	Right Button Released
3	Right Button Pressed
2	Left Button Released
1	Left Button Pressed
0	Cursor Moved

Note: When bits 7 - 5 are set, the corresponding shift state must be active to allow other events to cause the user subroutine to be called.

Function 25: Get User Alternate Interrupt Vector: This function returns a pointer to the subroutine defined by Function 24. The Event Mask condition in the CX register must match the Event Mask used to define the subroutine in Function 24. If no match is found, a 0 is returned in the CX register.

Figure 6-41. Function 25: Get User Alternate Interrupt Vector

Input		Output	
Register	**Value**	**Register**	**Value**
AX	25	—	—
—	—	BX:DX	User Interrupt Vector (or Undefined)
CX	User Interrupt Event Mask Shift Bits	CX	User Interrupt Event Mask (0 if No Match)

Function 26: Set Mouse Sensitivity: This function sets the mouse to cursor-movement sensitivity by defining the number of *x*'s and *y*'s that are equal to a single PEL. It also sets the double-speed threshold for the mouse. The sensitivity value must be in the range of 1 to 100, where a value of 1 specifies an *xy*-to-PEL ratio of 1:1. The default value is 50. These values are not reset by a Mouse Reset function call. If a value greater than 100 is passed in the BX or CX register, the number is truncated to 100.

Figure 6-42. Function 26: Set Mouse Sensitivity

Input		Output	
Register	**Value**	**Register**	**Value**
AX	26	—	—
BX	Horizontal *x* and *y* Coordinates per PEL	—	—
CX	Vertical *x* and *y* Coordinates per PEL	—	—
DX	Double Speed Threshold	—	—

Function 27: Get Mouse Sensitivity: This function returns the value set by Function 26.

Figure 6-43. Function 27: Get Mouse Sensitivity

Input		Output	
Register	**Value**	**Register**	**Value**
AX	27	—	—
—	—	BX	Horizontal *x* and *y* Coordinates per PEL
—	—	CX	Vertical *x* and *y* Coordinates per PEL
—	—	DX	Double-Speed Threshold

Function 29: Set CRT Page Number: This function specifies which CRT page the Mouse cursor will be displayed on.

Figure 6-44. Function 29: Set CRT Page Number

Input		Output	
Register	**Value**	**Register**	**Value**
AX	29	—	—
BX	CRT Page for Cursor Display	—	—

Function 30: Get CRT Page Number: This function returns the number of the CRT page the Mouse cursor will be displayed on.

Figure 6-45. Function 30: Get CRT Page Number

Input		Output	
Register	**Value**	**Register**	**Value**
AX	30	—	—
—	—	BX	CRT Page for Cursor Display

Function 31: Disable Mouse Driver: This function restores all interrupt vectors (except the interrupt hex 33 vector) used by the Mouse device driver to their values before the Mouse device driver was installed. The value returned in ES:BX can be used to restore the value of interrupt hex 33 vector. The Mouse device driver uses interrupt hex 10. It also uses interrupt hex 71 for systems with the 8086 microprocessor, or interrupt hex 74 for systems with the 80286 or 80386 microprocessor. AX is set to -1 if the Mouse device driver is unable to restore one or more of the vectors it is using.

Figure 6-46. Function 31: Disable Mouse Driver

Input		Output	
Register	**Value**	**Register**	**Value**
AX	31	—	—
—	—	AX	31 if Disable Was Successful -1 if Disable Was Unsuccessful
—	—	ES:BX	Previous Function 31 Vector

Function 32: Enable Mouse Driver: This function reinstalls the interrupt vector values used by the Mouse device driver. The pointer to the Mouse device driver is reinstalled through interrupt hex 15.

Note: This function will rechain any vectors unchained as a result of calls made to Function 32.

Figure 6-47. Function 32: Enable Mouse Driver

Input		Output	
Register	**Value**	**Register**	**Value**
AX	32	—	—

Function 33: Software Reset: This function is identical to Function 0, except that the Mouse is not reset.

Figure 6-48. Function 33: Software Reset

Input		Output	
Register	**Value**	**Register**	**Value**
AX	33	—	—
—	—	AX	-1 if the Mouse Driver is Installed 33 if the Mouse Driver is Not Installed
—	—	BX	2 if AX = -1

Function 36: Get Driver Version, Mouse Type, and IRQ Number: This function gets the version number of the Mouse driver, the type of Mouse it requires, and the number of the interrupt-request (IRQ) type. For example, a function call 36 to version 6.10 would return the value 0610 (binary coded decimal) in the BX register.

The Mouse type is contained in the CH register. A value of 1 indicates a bus Mouse, a value of 4 indicates a Personal System/2® Mouse.

The value for the interrupt-request type is contained in the CL register.

Figure 6-49. Function 36: Get Driver Version, Mouse Type, and IRQ Number

Input		Output	
Register	**Value**	**Register**	**Value**
AX	36	BH	Major Version Number
—	—	BL	Minor Version Number
—	—	CH=1	Bus Mouse
—	—	CH=2	Serial Mouse
—	—	CH=4	Personal System/2 Mouse
—	—	CL=0	Personal System/2 Value

Video - Supported Modes: The device driver supports all VGA (video graphics array) video modes. In graphics modes hex 0D through hex 13, the cursor is displayed only in black or white. (See Function 9 on page 6-20 for more detail about graphics cursors).

All VGA registers that are altered by the device driver during cursor displaying and erasing are returned to the state they were in when the device driver code was called. Modes hex 0D through hex 12 map video using bit planes.

Personal System/2 is a registered trademark of the International Business Machines Corporation.

Connector

The Mouse is connected to the system by a 6-pin connector. Figure 6-50 shows the pin configuration and signal assignments.

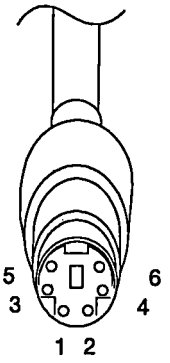


Figure 6-50. Mouse Connector Pin Assignments

Pin	Signal
1	Data
2	No Connection
3	Ground
4	+ 5 V dc
5	Clock
6	No Connection

Specifications

The following are specifications for the Mouse.

Programmable Resolution: (counts per millimeter) 1, 2, 4 (default), or 8.

Programmable Sampling Rate: (reports per second) 10, 20, 30, 40, 60, 80, 100 (default), or 200.

Data Modes: Stream (default), Remote, or Wrap.

Scaling: 1:1, 2:1.

Power: +5 V dc, ±10%, 70 milliamperes (maximum).

Maximum Tracking Speed: ≥ 200 millimeters per second.

Size	**Millimeters**	**Inches**
• Length:	110	4.3
• Depth:	66	2.6
• Heigth:	32	1.3

Weight	**Kilograms**	**Pounds**
	1.23	0.5

Section 7. Displays

Description 7-3
Vertical Modes 7-4
Signals 7-4
Display Connector 7-5
Power/Audio Cable 7-5
Specifications 7-6

Description

The IBM PS/1 computer will have either a Color or Black and White ™PHOTO GRAPHIC Display.

The IBM Color PHOTO GRAPHIC Display is a non-interlaced, VGA compatible color, direct-drive analog display with externally selected vertical modes. The display has a power switch, power-on indicator, headphone jack, and controls for brightness, contrast, and audio, and comes with a tilt leg. The IBM Black and White PHOTO GRAPHIC Display has the same description except with shades of gray rather than color.

Models

- 104—125V ac 50/60 Hz, Northern Hemisphere
- 200—240V ac 50 Hz, Northern Hemisphere.

Characteristics

- Vertical addressability of 350, 400, or 480 lines
- Self-test with a white screen test pattern
- 75-ohm direct-drive analog video input, 0.0—0.7V dc
- 305 millimeter (12 inch) cathode ray tube
- Etched, anti-glare faceplate
- Phosphor type: P-22 (Color), WD (Black and White)
- Horizontal deflection rate of 31.5 kHz, ± 0.5 kHz
- Horizontal blanking time of 5.7 microseconds
- Vertical deflection rate of 60—70 Hz
- Vertical blanking time of 0.88 millisecond
- Automatic degaussing (Color only)
- 1.1 meters (4 feet) display cable with a miniature 15-pin D-shell connector
- 1.1 meters (4 feet) display power/audio cable with a 15-pin D-shell connector
- 1.8 meters (6 feet) detachable AC line cord for the 200—240V ac version
- 1.8 meters (6 feet) nondetachable AC line cord for the 104—125V ac version
- Miniature 32 ohm headphone jack: stereo (Color), monaural (Black and White)
- Speaker with volume control.

PHOTO GRAPHIC is a trademark of the International Business Machines Corporation.

Vertical Modes

The display monitors the polarity of the synchronization pulses from the video controller and selects the number of scan lines according to the following figure.

Figure 7-1. Display Vertical Modes

VSYNC Polarity	HSYNC Polarity	Data Scan Lines	Data and Border Scan Lines
+	+	Reserved	Reserved
+	−	350	362
−	+	400	414
−	−	480	496

Signals

The display receives the video signals from a current source with a 150-ohm termination. The signal input impedance is 75 ohms. The video signals have a range of 0.0 − 0.7V dc. The vertical synchronization pulse width is 63.556 microseconds. The horizontal synchronization pulse width is 3.813 microseconds. The self-test signal, when open, enables the full raster test. The audio signal input has a range of 0 − 50 mV rms. The audio output is produced in the headphone or speaker. The display will supply a 36V output (30 − 40V dc range) with the maximum power output not to exceed 90 watts.

Display Connector

The following figure shows the pin numbers and signal assignments for the signal cable miniature 15-pin D-shell connector.

Figure 7-2. Display Connector Signals

Pin	Signal (Color)	Signal (Black and White)
1	Video red	No pin
2	Video green	Video
3	Video blue	No pin
4	Reserved	No pin
5	Self test	Self test
6	Video red return	Orientation pin
7	Video green return	Video return
8	Video blue return	No pin
9	Reserved	No pin
10	Ground	Ground
11	Monitor sense (Ground)	Monitor sense (Open)
12	Monitor sense (Open)	Monitor sense (Ground)
13	Horizontal sync	Horizontal sync
14	Vertical sync	Vertical sync
15	Reserved	No pin

Power/Audio Cable

The following figure shows the pin numbers and signal assignments for the power cable 15-pin D-shell connector.

Figure 7-3. Power/Audio Cable Connector Pin Definition

Pin	Signal (Color)	Signal (Black and White)
1	+ 36 volts	+ 36 volts
2	+ 36 volts	+ 36 volts
3	Ground	Ground
4	Reserved	Reserved
5	Reserved	Reserved
6	Audio left channel	Audio
7	Audio return	Audio return
8	Audio right channel	Reserved
9	+ 36 volts	+ 36 volts
10	Ground	Ground
11	Ground	Ground
12	Reserved	Reserved
13	Reserved	Reserved
14	Ground	Ground
15	Audio return	Reserved

Specifications

The following are specifications for the displays.

Size

- Width: 320 millimeters (12.6 inches)
- Depth: 430 millimeters (16.9 inches)
- Height: 280 millimeters (11.0 inches).

Weight

- 12.7 kilograms (28 pounds) - Color
- 10 kilograms (22 pounds) - Black and White.

Cables

- AC Line Cord: 1.8 meters (6 feet)
- Display: 1.2 meters (4 feet)
- Power/Audio: 1.2 meters (4 feet).

Operational

- Temperature: 15 to 35 degrees C (60 to 95 degrees F)
- Humidity: 8% to 80%
- Altitude: 0 to 2134 meters (0 to 7000 feet).

Section 8. Drives

3.5-Inch 30MB Fixed Disk Drive and Controller 8-3
Description 8-3
Fixed Disk Controller 8-4
Programming 8-5
Registers 8-6
Data Register, Hex 320 8-6
Attachment Status Register, Hex 322 8-6
Attachment Control Register, Hex 322 8-7
Interrupt Status Register, Hex 324 8-8
Attention Register, Hex 324 8-9
Control Blocks 8-10
Command Control Block 8-10
Command Specify Block 8-12
Sense Summary Block 8-13
Status Bytes 8-14
Format Control Block 8-17
Commands 8-19
Connector 8-24
Signal Description 8-25
Specifications 8-26
3.5-Inch 1.44MB Diskette Drive 8-27
Description 8-27
Programming Considerations 8-28
Signal Levels 8-28
Connector 8-28
Specifications 8-29

3.5-Inch 30MB Fixed Disk Drive and Controller

Description

This section describes the 30MB fixed disk drive for the IBM PS/1 computer. The drive is attached to a dedicated I/O channel connector on the system board through a single flat cable. This channel supplies all the necessary power and control signals.

The drive and controller is buffered on the I/O bus and use the system's direct memory access (DMA) for fixed-disk-drive data transfers. When enabled, an interrupt request to the system occurs on the interrupt line, IRQ14. The interrupt controller then causes an interrupt hex 76.

The drive and controller provide automatic 48-bit error checking and correction (ECC).

The device-level control for the fixed disk is in the system BIOS.

The last cylinder on the fixed disk drive is reserved for diagnostic use. The diagnostic write test destroys any data on this cylinder.

Important: IBM provides a BIOS interface to insulate the programmer from hardware dependences. IBM recommends that all applications use the BIOS interface, or the operating system interface, to prevent incompatibilities caused by differences in hardware.

Fixed Disk Controller

The disk controller has five registers that may be accessed by the system microprocessor: two status registers, a data register, and two control registers.

The two status registers contain the status information of the disk controller and can be accessed at any time. These registers are read-only and indicate the status of data transfers between the microprocessor and disk controller.

The data register (consisting of several registers in a stack with only one register presented to the data bus) stores data, commands, and parameters and provides the disk controller's sense information. Data bytes are read from, or written to the data register in order to program or obtain the results after a particular command.

The two control registers initiate the transfer of commands, data, and sense information through the data registers and enable DMA and interrupt requests. The controller-select signal (-DISK CS) is generated by writing to port hex 32x.

The following is a block diagram of the disk drive and controller.

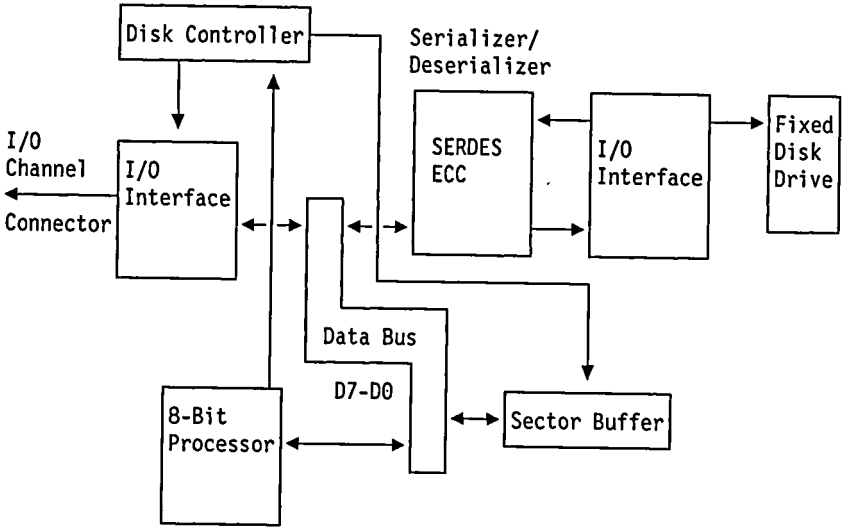


Figure 8-1. Fixed Disk and Controller

Programming

Before enabling the DMA controller or the interrupt controller, the request enable bits in the Attachment Control register should be set to enable the requests. This prevents erroneous DMA transfers or interrupts.

Because the system has a dedicated I/O channel for the fixed disk, the I/O address decoding is done on the system board. The system activates the disk address signal (-DISK CS) using the 16 most-significant address bits; the 3 least-significant address bits select the various registers.

Because the drive and controller logic are integrated and the controller is dedicated to the one drive type, the controller has no switch settings or drive select requirements.

The drive and controller operate using a control block architecture that minimizes system interaction. Control blocks can be transferred to the drive by the microprocessor or DMA controller. These transfers and data transfers are initiated through the Attention register. Completion of the operation is signaled through the two status registers and the system interrupt.

The following figure is a table of the I/O ports and their function.

Figure 8-2. Port Addresses

R/W	Port Address	Function
Read	320	Read data from drive
Write		Write data to drive
Read	322	Read Attachment Status register
Write		Write Attachment Control register
Read	324	Read Interrupt Status register
Write		Write Attention register

Registers

Data Register, Hex 320

The data register is read/write and transfers data between the controller and the system. This register consists of two register stacks. A sector of data is transferred into the first stack. While that data is transferred out of the first stack, the second stack receives the second sector of data.

Attachment Status Register, Hex 322

This register is read-only and contains status information on the present state of the controller. The following are bit descriptions.

Figure 8-3. Attachment Status Register

Bit	Function
7 - 5	Reserved
4	Data Request
3	Direction
2	Busy
1	Interrupt Request
0	Transfer Enable

Bit 7 - 5 Reserved.

Bit 4 Data Request: This bit is set by the controller to request a data transfer. This bit does not cause an interrupt, so it must be polled by the microprocessor before an I/O Read or Write to the data register.

Bit 3 Direction: This bit indicates the direction of the data transfer associated with bit 4. When set, the direction is from the controller.

Bit 2 Busy: When set to 1, this bit indicates that the controller is processing a command through the Attention register.

Bit 1 Interrupt Request: This bit is set to indicate that a command has been completed and the content of the Interrupt Status register is valid. If the interrupt is enabled, this bit causes an interrupt. This bit is cleared when the register is read.

Bit 0 Transfer Enable: When set, this bit indicates that data is being transferred.

Attachment Control Register, Hex 322

The Attachment Control register controls the fixed-disk interrupt and DMA channel, and resets the drive. The write address is hex 322.

Figure 8-4. Attachment Control Register

Bit	Function
7	Reset
6 - 2	Reserved = 0
1	Interrupt Enable
0	DMA Enable

Bit 7 Reset: When set to 1, this bit causes a hardware reset. The sequence of the reset is:

1. Write this register with this bit set.
2. Write this register with all bits clear.
3. Write this register with bits set as desired.

Bits 6 - 2 Must be written as 0.

Bit 1 Interrupt Enable: When set to 1, this bit enables the interrupt signal to the system.

Bit 0 DMA Enable: When set, this bit enables DMA requests to the system. When clear, the microprocessor performs the transfers.

Interrupt Status Register, Hex 324

At the end of all commands from the microprocessor, the disk controller returns completion status information to this register. This byte informs the system if an error occurred during the execution of the command. The following shows the format of this byte.

Figure 8-5. Interrupt Status Register

Bit	Function
7	Termination Error
6	Invalid Command
5	Command Reject
4 - 2	Reserved
1	ERP Invoked
0	Equipment Check

Bit 7 Termination Error: When set, this bit indicates a severe error has occurred. The specific error is indicated by other bits in this register or in the Sense Summary Block described later in this section.

Bit 6 Invalid Command: When set, this bit indicates that the Command Control Block, Command Specify Block, or the Attention register contains an invalid command or parameter.

Bit 5 Command Reject: When set, this bit indicates the controller cannot execute the command. This bit is also set if a transaction other than a request for a Sense Summary Block is initiated before a required reset.

Bits 4 - 2 Reserved.

Bit 1 ERP Invoked: When set, this bit shows an error has occurred during command execution and error recovery procedures (ERP) have been done. If the procedures were not successful, the termination error bit is set. This bit is used to track soft errors.

Bit 0 Equipment Check: When set, this bit indicates an internal hardware error has occurred. This error is cleared by resetting the drive.

Attention Register, Hex 324

The system uses this register to initiate all transactions with the drive. The write address is hex 324. The bits are set according to the type of transaction and are defined as follows:

Figure 8-6. Attention Register

Bit	Function
7	CCB Request
6	CSB Request
5	SSB Request
4	Data Request
3 - 0	Reserved = 0

Bit 7 Command Control Block: When set, this bit signals the drive that a Command Control Block is pending transfer. The drive then requests the control block through the data request bit of the Attachment Status register.

Bit 6 Command Specify Block: When set, this bit signals the drive that a command specify block is pending transfer. The drive then requests the specify block.

Bit 5 Sense Summary Block: When set, this bit signals a request for a Sense Summary Block. The drive then sends the summary block.

Bit 4 Data Request: When set, this bit indicates that the system is ready to start a data transfer.

Bits 3 - 0 Must be written as 0.

Control Blocks

Command Control Block

The system specifies the operation by sending the 6-byte command control block to the controller. It can be sent through a DMA or I/O operation. The figure below shows the format of the command control block (CCB) and defines the bytes that make up the CCB.

Figure 8-7. Command Control Block

	7	6	5	4	3	2	1	0
Byte 0	Command Code				ND	AS	0	EC
Byte 1	Head Number				0	0	Cyl High	
Byte 2	Cylinder Low							
Byte 3	Sector Number							
Byte 4	0	0	0	0	0	0	1	0
Byte 5	Number of Sectors							

Byte 0

Bit 7 - 4 The codes and their commands are explained later in this section.

Bit 3 When set, this bit indicates that the data transfer is only between the disk drive and the sector buffer. No data is transferred to the system. When this option is specified, only one sector is transferred to prevent overrun errors.

Bit 2 When set, this bit instructs the drive to seek the cylinder specified in the command block. When clear, the head positioning mechanism must be positioned to the desired cylinder using the Seek command before any other operation.

Bit 1 Must be written as 0.

Bit 0 When set, this bit indicates that data fields or sectors are processed using error checking and correction (ECC) rather than cyclic redundancy checking (CRC).

Byte 1

Bits 7 - 4 These four bits define the head number selected. Because there are only two heads, the three most-significant bits are always 0.

Bits 3, 2 Must be written as 0.

Bits 1, 0 These are the two most-significant bits of the cylinder number.

Byte 2 Cylinder Low: This byte contains the eight least-significant bits of the cylinder number.

Byte 3 Sector Number: This byte specifies the sector number being processed. For multiple sector operations, it specifies the starting sector number.

Byte 4 Sector Size: This byte contains the encoded value for the sector size. The value is always hex 02 to indicate 512 bytes.

Byte 5 Number of Sectors: This byte specifies the number of sectors that are to be processed. The system must provide for the correct amount of data transferred because the controller does not interrupt the system between sectors.

Command Specify Block

The command specify block (CSB) controls the error recovery procedures of the drive. The drive assumes the default parameters following power on and reset. These default values can be overridden by issuing a command specify block after initialization of the drive. All 14 bytes of the CSB are not used by the drive and controller; however, all bytes must be sent.

The command specify block is described in the following.

Figure 8-8. Command Specify Block

	7	6	5	4	3	2	1	0
Byte 0	En	0	0	0	Retries			
Byte 1 to Byte 13	Reserved = 0							

Byte 0 Control

Bit 7 When set, this bit causes the drive to correct an ECC error in the sector buffer. The default is ECC enabled.

Bits 6 - 4 Must be written as 0.

Bits 3 - 0 These bits specify the number of retries that the drive performs on an operation before reporting the error. The default is 15 retries.

Sense Summary Block

The sense summary block contains the current status of the drive. The information in the summary block is updated after each command is completed, after an error, or before the block is transferred.

If the Interrupt Status register shows an error (bit 1 set), the system requests 14 bytes of sense data by setting bit 5 in the Attention register. The format for the summary block is:

Figure 8-9. Sense Summary Block

	7	6	5	4	3	2	1	0
Byte 0	-R	SE	0	WF	CE	0	0	T0
Byte 1	EF	ET	AM	BT	WC	0	0	ID
Byte 2	0	RR	RG	DS	Hd Sel State			
Byte 3	Cylinder Low							
Byte 4	DS	Cyl High		0	Hd Number			
Byte 5	Sector Number							
Byte 6	Sector Size (hex 02)							
Byte 7	Hd Number				0	0	Cyl High	
Byte 8	Cylinder Low							
Byte 9	Number of Sectors Corrected							
Byte 10	Number of Retries							
Byte 11	Command Syndrome							
Byte 12	Drive Type Identifier							
Byte 13	Reserved							

Status Bytes

The first three bytes contain status information on the drive and controller signals.

Status Byte 0

Bit 7 When set, this bit indicates that the drive is not ready.

Bit 6 When set, this bit indicates that the drive has completed a seek operation.

Bit 5 Reserved.

Bit 4 When set, this bit indicates that the drive has shown a write-fault condition.

Bit 3 When set, this bit indicates that the controller has received a request to step beyond the limit of the last cylinder. The park option of the Seek command does not cause this bit to be set.

When the CCB specifies a cylinder beyond the limit, no step operation is done and the heads do not move. When the CCB started at a valid cylinder but involved multiple track operations that exceeded the limit, the heads are at the limit.

Bits 2, 1 Reserved.

Bit 0 When set, this bit indicates the heads are at track 0.

Status Byte 1

Bit 7 This bit indicates the type of field (ID or data) that caused the uncorrectable error. When set, this bit indicates an ID field.

Bit 6 When set, this bit indicates a CRC or uncorrectable ECC error occurred. The field is identified by bit 7.

Bit 5 When set, this bit indicates that the address mark for the data field of the requested sector was not found.

Bit 4 When set, this bit indicates an ID field with all bits set was detected.

Bit 3 When set, this bit indicates that the cylinder bytes read did not match the cylinder requested in the CCB.

Bit 2, 1 Reserved.

Bit 0 When set, this bit indicates an ID match was not found and the complete track was searched.

Status Byte 2

Bit 7 Reserved.

Bit 6 When set, this bit indicates that the drive needs to be reset.

Bit 5 When set, this bit indicates that the read or write retry corrected the error condition.

Bit 4 When set, this bit indicates that the defective sector bit in the ID field is set to 1.

Bits 3 - 0 These bits reflect the state of the head select signals to the drive.

Bytes 3 through 6 Last ID: These bytes represent the cylinder, head, sector number, and sector size of the last ID field processed. The most-significant bit of byte 4 is set if the operation was to a defective sector.

Bytes 7 and 8 Present ID: These bytes represent the true cylinder and head numbers of the present head location in the same format as bytes 1 and 2 of the CCB.

Byte 9 NSC: This byte contains the number of sectors corrected by the ECC before transfer.

Byte 10 Retry Tally: This byte contains the number of retries attempted during the last operation. When this byte reaches the maximum value, the controller halts operation until it is reset. This value, 255, does not depend on the value allowed for any one sector.

Byte 11 Command Syndrome: This byte shows the progress of the controller through the last command. It allows the system to monitor the controller and determine if a reset is needed. When the transfer of the control block is started, the value is set to hex 00. The progress indicated by this byte is:

1. Set to hex 01 after the control block is successfully transferred.
2. Set to hex 02 when the command is valid and the drive is ready.
3. Set to hex 03 when the head is in the correct track. The most-significant four bits (high nibble) are then used to indicate the successful stages of the data transfer.

   Bit 7 A sector was transferred between the system and the sector buffer.
   Bit 6 A sector was transferred between the controller and the sector buffer.
   Bit 5 An error was detected and error recovery procedures have been started.
   Bit 4 The controller has completed the operation and is now not busy.
4. When the transfer is complete, the low nibble equals hex 4 and the high nibble is unchanged.

Byte 12 Drive Type Identifier: This byte contains an 8-bit identifier code for the fixed disk drive installed. The system can read this code to set the drive type using the drive configuration table.

Byte 13: This bit is reserved.

Format Control Block

The format control block (FCB) specifies the ID data used in formatting the track. It is used by the Format Track and Format Disk commands and contains five bytes for each sector formatted on that track.

When the Format Disk command is used, the control block contains the sector information of all sectors for head 0, cylinder 0. The drive will use the same block to format the rest of the disk and automatically increment the head number and cylinder number for the remaining tracks. The sector numbers, sector size, and the fill byte will be the same for each track.

The drive formats the sector IDs on the disk in the same order as they are specified in the control block. Therefore, sector interleaving is accomplished by filling in the control block with the desired interleave. For example, when formatting 17 sectors per track with an interleave of 2, the control block has the first 5 bytes with a sector number of 1, the second with a sector number of 10, the third with a sector number of 2, and continuing until all 17 sectors for that track are defined.

The format for the format control block is described in the following. The five bytes are repeated for each sector on the track. The control block must contain an even number of bytes. If an odd number of sectors are being formatted, an additional byte is sent with all bits 0.

Figure 8-10. Format Control Block

	7	6	5	4	3	2	1	0
Byte 0	Hd Number				0	DS	Cyl High	
Byte 1	Cylinder Low							
Byte 2	Sector Number							
Byte 3	0	0	0	0	0	0	1	0
Byte 4	Fill Byte							

Sector Information: The five bytes of sector information contains the same basic information for Format Disk and Format Track.

Byte 0: This byte contains the head select bits (7 through 4), the defective sector bit (2), and the two most-significant bits for the cylinder number (1 and 0).

Byte 1: This byte contains the low-order bits of the cylinder number.

Byte 2: This byte contains the sector number written in the ID field for that sector.

Byte 3: This byte specifies the sector size.

Byte 4: This byte contains the fill character for each sector. This is the value that is written into each byte in the data field of the formatted sector. To identify an entire track as defective, the system performs a Format Track command with all five bytes of the FCB for each sector set to hex FF.

Commands

The commands to the controller are in the first byte of the CCB. The command determines the number of bytes and the particular bits used, even though the CCB always contains the full six bytes.

Read Data: This command is the normal Read command. The number of sectors specified is read from the disk and transferred to the system. Up to 255 sectors can be specified in one CCB. The controller automatically increments the head and cylinders as necessary to satisfy the sector count. If this feature is used, the sectors must be numbered the same for all tracks. The CCB for the Read Data command is:

Figure 8-11. Read Command

	7	6	5	4	3	2	1	0
Byte 0	0	0	0	1	ND	AS	0	E
Byte 1	Hd Number				0	0	Cyl High	
Byte 2	Cylinder Low							
Byte 3	Sector Number							
Byte 4	0	0	0	0	0	0	1	0
Byte 5	Number of Sectors							

Read Check: This command reads the data from the disk without transferring it to the system. It serves to verify the readability of the data and the correctness of the CRC or ECC field. Multiple sectors, up to 255, can be tested in this manner. This command is similar to the Read Data command with the No Data option specified. The difference is that the Read Check command can specify more than one sector.

Figure 8-12. Read Check Command

	7	6	5	4	3	2	1	0
Byte 0	0	0	1	0	0	AS	0	E
Byte 1	Hd Number				0	0	Cyl High	
Byte 2	Cylinder Low							
Byte 3	Sector Number							
Byte 4	0	0	0	0	0	0	1	0
Byte 5	Number of Sectors							

Read Extended: This command reads and transfers the specified sector along with the associated CRC or ECC bytes to the system. Six bytes of error checking are transferred with the sector data regardless of the error checking used. When using this command, the controller does not check the data for errors.

Figure 8-13. Read Extended Command

	7	6	5	4	3	2	1	0
Byte 0	0	0	1	1	0	AS	0	E
Byte 1	Hd Number				0	0	Cyl High	
Byte 2	Cylinder Low							
Byte 3	Sector Number							
Byte 4	0	0	0	0	0	0	1	0
Byte 5	X	X	X	X	X	X	X	X

Read ID: This command reads the ID from the first sector encountered on the specified head and cylinder. This command is used to verify the current head position on the drive.

Note: Because of the freedom allowed in the Format Track command, it is possible to format a track with IDs containing different cylinder values than the actual cylinder.

Figure 8-14. Read ID Command

	7	6	5	4	3	2	1	0
Byte 0	0	1	0	1	0	AS	0	0
Byte 1	Hd Number				0	0	Cyl High	
Byte 2	Cylinder Low							
Byte 3 to Byte 5	X	X	X	X	X	X	X	X

Recalibrate: This command moves the heads to cylinder 0. If the operation fails, the termination error bit is set in the Interrupt Status register.

Figure 8-15. Recalibrate Command

	7	6	5	4	3	2	1	0
Byte 0	1	0	0	0	0	0	0	0
Byte 1 to Byte 5	X	X	X	X	X	X	X	X

Write Data: This command is the normal Write command. Data is written on the fixed disk in consecutive sectors, beginning with the sector number in byte 3. Up to 255 sectors can be processed with each command. When more than one track of data is being transferred, the controller automatically increments the head and cylinder number as necessary.

There are two items to consider when performing multiple track transfers:

1. Because the controller adjusts the head and cylinder number automatically, all cylinders must have the same sector numbering.
2. Because the system is not interrupted until the entire operation is completed or an error occurs, the system must be sure to transfer the stated amount of data.

Figure 8-16. Write Data Command

	7	6	5	4	3	2	1	0
Byte 0	1	0	0	1	ND	AS	0	E
Byte 1	Hd Number				0	0	Cyl High	
Byte 2	Cylinder Low							
Byte 3	Sector Number							
Byte 4	0	0	0	0	0	0	1	0
Byte 5	Number of Sectors							

Write Verify: This command is a combination of two commands, Write Data and Read Check. After the data is written, the controller reads the data for ECC errors. For multiple track operations, the read check is performed on the track boundary to prevent unnecessary track steps.

Figure 8-17. Write Verify Command

	7	6	5	4	3	2	1	0
Byte 0	1	0	1	0	ND	AS	0	E
Byte 1	Hd Number				0	0	Cyl High	
Byte 2	Cylinder Low							
Byte 3	Sector Number							
Byte 4	0	0	0	0	0	0	1	0
Byte 5	Number of Sectors							

Write Extended: This command is similar to the Read Extended command. The data and the six bytes of error checking are written to the specified sector. The drive does not error check. If the error checking option is CRC, then only the first two bytes are written but all six bytes must be sent.

Figure 8-18. Write Extended Command

	7	6	5	4	3	2	1	0
Byte 0	1	0	1	1	0	AS	0	E
Byte 1	Hd Number				0	0	Cyl High	
Byte 2	Cylinder Low							
Byte 3	Sector Number							
Byte 4	0	0	0	0	0	0	1	0
Byte 5	X	X	X	X	X	X	X	X

Format Disk: This command writes the entire disk with the filler and ID information supplied in the format control block. This control block contains the format for head 0 of cylinder 0. The controller then increments the head and cylinder numbers as needed until the disk is completely formatted.

The number of sectors specified in byte 5 is the number of sectors per track.

Figure 8-19. Format Disk Command

	7	6	5	4	3	2	1	0
Byte 0	1	1	0	1	0	0	0	E
Byte 1	X	X	X	X	X	X	X	X
Byte 2	X	X	X	X	X	X	X	X
Byte 3	X	X	X	X	X	X	X	X
Byte 4	0	0	0	0	0	0	1	0
Byte 5	Number of Sectors							

Seek: This command causes the drive to position the heads at the specified cylinder. The position is not verified, even though the Present Cylinder Number fields are updated in the sense summary block. In all Read and Write commands, an implicit Seek command can be performed by choosing the Auto Seek option.

When the park option bit is set (bit 0 of byte 0), the controller positions the heads over the landing zone. No cylinder or head information is needed because it is assumed to be the last cylinder, therefore, bytes 1 and 2 become "don't care."

Figure 8-20. Seek Command

	7	6	5	4	3	2	1	0
Byte 0	1	1	1	0	0	0	0	P
Byte 1	Hd Number				0	0	Cyl High	
Byte 2	Cylinder Low							
Byte 3 to Byte 5	X	X	X	X	X	X	X	X

Format Track: This command writes a single track with ID and filler information. This information is supplied by the system in the FCB. In contrast to the Format Disk command, Format Track command allows the system to format defective or other non-standard tracks.

Figure 8-21. Format Track Command

	7	6	5	4	3	2	1	0
Byte 0	1	1	1	1	0	AS	0	E
Byte 1	Hd Number				0	0	Cyl High	
Byte 2	Cylinder Low							
Byte 3	X	X	X	X	X	X	X	X
Byte 4	0	0	0	0	0	0	1	0
Byte 5	Number of Sectors							

Connector

The fixed disk drive and controller connector and interface specifications are shown below.

Pin numbers are consistent with those numbers on the system board and cable connector.

Figure 8-22. Fixed Disk Connector

Pin	I/O	Signal	Pin	I/O	Signal
44	I	RESET DRV	43	O	-DISK Installed
42	I/O	D0	41	N/A	Ground
40	I/O	D1	39	N/A	Ground
38	I/O	D2	37	N/A	Ground
36	I/O	D3	35	N/A	Ground
34	I/O	D4	33	N/A	Ground
32	I/O	D5	31	N/A	Ground
30	I/O	D6	29	N/A	Ground
28	I/O	D7	27	N/A	Ground
26	I	-IOR	25	N/A	Ground
24	I	-IOW	23	N/A	Ground
22	I	-DISK CS	21	N/A	Ground
20	I	A0	19	N/A	Ground
18	I	A1	17	N/A	Ground
16	I	A2	15	N/A	+5
14	I	A3	13	I/O	+5
12	I/O	-DACK3	11	N/A	Ground
10	I/O	DRQ3	9	N/A	Ground
8	O	IRQ 14	7	N/A	Ground
6	O	IO CH RDY	5	I	+12
4	N/A	Spare	3	I	+12
2	N/A	Spare	1	I	+12

Signal Description

The following lines are used by the drive and controller:

A0 – A3	These four address bits are used to select the specific register within the controller. The other address lines are not needed because address decoding is done on the system board.
D0 – D7	Positive 8-bit data bus over which data and status information is passed between the system and the controller.
-IOR	When active, the "-I/O read" signal instructs the controller to write its data onto the data bus.
-IOW	When active, the "-I/O write" signal instructs the controller to read the data on the data bus.
RESET DRV	When active, the "reset driver" signal forces the controller to a reset state. The controller initializes when RESET DRV goes inactive.
IRQ 14	This signal is made active by the controller after status information is returned. This signal is enabled and disabled through the Attachment Control register.
IO CH RDY	This signal is used to lengthen I/O and DMA operations. The controller makes the signal inactive to add wait states.
DRQ 3	This signal is activated by the controller when data is available for transfer under DMA control and remains active until the "DMA acknowledge" signal (-DACK 3) becomes active. It is enabled and disabled through the Attachment Control register.
-DACK 3	This signal is active in response to a DMA request.
-DISK CS	The address decode logic for the fixed disk is on the system board. It is enabled through the Planar Control register. When the logic is enabled, the "-disk card select" signal goes active on a valid decode of A4 through A19 equal to hex 032x.
-DISK Installed	When active, this signal indicates that the drive and controller are installed.

Specifications

There are two 30MB fixed disk drives and controllers. Each system's read-only memory (ROM) contains a parameters table. If a fixed disk drive is supported by a system, the drive type will be listed on the parameters table.

The following are specifications for the 30MB fixed disk drives and controllers.

	Drive Type 35	**Drive Type 38**
Formatted Capacity		
• Bytes/Sector	512	512
• Sectors/Track	33	36
• Cylinders	921	845
• Heads	2	2
Rotational Speed	3600 rmp	3700 rpm
Transfer Rate	10.2M bps	10.8M bps
Access Time*		
• Track to Track	8 ms	9 ms
• Average (transverse 1/3 cylinder)	19 ms	21 ms
• Maximum	40 ms	40 ms
Interleave	4:1	4:1

* Nominal enivronmental and voltage conditions.

Figure 8-23. DC Power Specifications (for both drives)

• Nominal Supply	+5 V	+12 V
• Tolerance	±5%	±8%
• Supply Current		
Idle, R/W Mode	0.4 A max	0.6 A max
Seeking	0.4 A max	0.8 A max
Startup	0.4 A max	1.2 A max

3.5-Inch 1.44MB Diskette Drive

Description

This section describes the 3.5-inch 1.44MB diskette drive option. This drive is a direct-access device containing a spindle drive, head positioning mechanism, and read/write/erase logic. A drive-in-use indicator lights when the drive is selected.

The 3.5-inch 1.44MB diskette drive accepts either 1MB or 2MB capacity diskettes with 80 tracks on each side. The 3.5-inch 1.44MB diskette drive is identified externally by a "1.44" printed on its diskette eject button.

The spindle mechanism spins the diskette at a constant speed of 300 revolutions per minute (rpm). A sensor generates an index signal once per rotation of the spindle motor. The two read/write heads are positioned over the desired track of the diskette by a stepper motor. One step of the stepper motor results in a one-track linear movement of the read/write heads. An optical sensor generates a signal when the heads are over track 0.

During a write operation, data is provided to the drive in modified frequency modulation (MFM) coded form by a diskette drive controller.

If the diskette is write-protected, the drive's write-protect sensor inhibits the write operation.

Programming Considerations

The following describes the interface to the diskette drive:

Signal Levels

All input signals operate between +5 V dc and ground with the following definitions:

- The inactive level is +2.0 V dc minimum
- The active level is +0.8 V dc maximum.

All outputs from the drive can sink 4.0 mA at the active (low) level. All interface signals are CMOS compatible.

Connector

The interface is divided into three categories: Control, Data, and DC power. These signals are all provided through a 34-pin header connector. Even-numbered pins are located on the top row and odd-numbered pins are located on the bottom row (closest to the PC board). Pins are located on 2.54-millimeter centers.

Figure 8-24. Connector Pin Assignments

Pin	I/O	Signal	Pin	I/O	Signal
1	N/A	Signal Ground	2	I	-High Density Select
3	I	+5 V	4	O	-Drive Type 1
5	N/A	Signal Ground	6	I	+12 V
7	N/A	Signal Ground	8	O	-Index
9	N/A	Signal Ground	10	N/A	-Open
11	N/A	Signal Ground	12	I	-Drive Select
13	N/A	Signal Ground	14	N/A	-Open
15	N/A	Signal Ground	16	I	-Motor Enable
17	N/A	Signal Ground	18	I	-Direction
19	N/A	Signal Ground	20	I	-Step
21	N/A	Signal Ground	22	I	-Write Data
23	N/A	Signal Ground	24	I	-Write Enable
25	N/A	Signal Ground	26	O	-Track 0
27	N/A	Signal Ground	28	O	-Write Protect
29	N/A	Signal Ground	30	O	-Read Data
31	N/A	Signal Ground	32	I	-Head 1 Select
33	N/A	Signal Ground	34	O	-Diskette Change

Specifications

The following are specifications for the 3.5-inch diskette drive.

Size

- Width: 102 millimeters (4.0 inches)
- Depth: 150 millimeters (5.9 inches)
- Height: 25.4 millimeters (1.0 inch).

Weight 0.45 kilograms (1.0 pound).

Voltage

- +12 V dc
- +5 V dc.

Media

- 3.5-inch ANSI compatible
- Media Capacity

	1MB	2MB
– Unformatted	500KB per side 6.25KB per track	1MB per side 12.50KB per track
– Formatted	720KB (360KB per side) 4.5KB per track.	1.44MB (720KB per side) 9KB per track.

Track Density 135 tracks per inch.

Tracks 80.

Number of Heads 2.

Transfer Rate

- 3.5-inch 720KB diskette: 250,000 bps (MFM)
- 3.5-inch 1.44MB diskette: 500,000 bps (MFM).

Access Time

- Track-to-Track: 3 milliseconds
- Seek Settle Time: 15 milliseconds
- Motor Start Time: 500 milliseconds (maximum).

Disk Speed 300 rpm ±1.5%.

Section 9. Modem/RS-232C Interface Cards

Description 9-3
2400 bps Modem Card 9-3
Description 9-3
Modem Commands and Operation 9-6
AT (Attention) Command Description 9-6
AT Commands 9-8
Result (Return) Codes 9-14
Register Summary 9-15
Bit-Mapped Registers 9-17
Specifications 9-20
Dialer Type 9-20
Pulse Dialing 9-20
DTMF Dialing 9-20
Modulation 9-20
Transmitter Characteristics 9-20
Receiver Characteristics 9-20
Analog Interface 9-20
RS-232C Interface Card 9-21
Description 9-21
Voltage Interchange Information 9-21
RS-232C Connector 9-22

Description

The system board supports the attachment of either a 2400 bps modem card or an RS-232C interface card. Either card attaches to the serial interface connector. Only one or the other is supported dependent on the country. For example, U.S. machines contain the 2400 bps modem.

2400 bps Modem Card

Description

The 2400 bits per second (bps) modem card is a serial communications adapter that provides an interface between the system and a voice-grade communication line for the transmission and reception of digital data.

The card is connected to the system board through the serial interface connector. Its features are as follows:

- Asynchronous data transfer at 0–300, 1200, and 2400 bps
- Automatic tone or pulse dialing
- Automatic or manual originate and answer operation
- Automatic speed and type identification of the calling modem
- Dial tone and busy tone detection
- All configuration options software selectable
- Direct connection to a telephone line using a standard modular telephone jack.

Compatibility with modems conforming to these protocols:

CCITT V.22 bis	1200 bps and 2400 bps asynchronous
CCITT V.22 A and B	1200 bps asynchronous
BELL 212A	1200 bps asynchronous
BELL 103	0 to 300 bps anisochronous.

Attention (AT) command set, compatible with the Hayes[1] command set, controls modem configuration and operation.

The modem has fixed line equalization to compensate for the normal variations in line quality with the addition of adaptive equalization at 2400 bps.

The modem card is registered as the "IBM 2400 BPS Modem" with the FCC to comply with Part 68 of the FCC Regulations. Any modification voids FCC and other certifications of regulatory compliance.

1 Trademark of Hayes Microcomputer Products, Inc.

Connector J1:

A RJ11C phone jack is used to interconnect to telephone lines.

Pin	Signal Name	Definition
1	Not used	
2	Not used	
3	TIP	Phone input/output
4	Ring	Phone input/output
5	Not used	
6	Not used	

Connector J2:

A 2 x 10 shrouded keyed male connector and flat cable assembly are used to interconnect to the system board.

Pin	Name	Definition
1	+12 volts	Power to modem.
2	AUDIOMOD	Audio from modem to speaker.
3	+12 volts	Power to modem.
4	RESERVED	Reserved.
5	-12 volts	Power to modem.
6	RTS*	Request to Send indicates that the planar is ready to send data.
7	+5 volts	Power to modem.
8	DTR*	Data Terminal Ready indicates that the planar is ready to send data.
9	+5 volts	Power to modem.
10	SOUT*TXD	Serial data to modem.
11	Ground	Power to modem.
12	CTS*	Clear to Send indicates that the modem is ready to send data to the planar.
13	RESET*	System reset from planar.
14	RLSD*	Received Level Signal Detect is an indication that the modem has received a carrier.
15	Ground	Power to modem.
16	RI*	Ring Indicate is used when the modem detects a valid ring signal.
17	Ground	Power to modem.
18	DSR*	Data Set Ready is a status from the modem to indicate that it is ready to transmit or receive data.
19	Ground	Power to modem.
20	SIN*RXD	Serial data from modem.

Connector J3:

A 4-pin male connector is used to provide power to the 5.25-inch diskette drive option. The modem card only provides capacitive decoupling for these signals:

Pin	Name	Definition
1	+12 volts	Power to 5.25-inch diskette drive
2	Ground	Power to 5.25-inch diskette drive
3	Ground	Power to 5.25-inch diskette drive
4	+5 volts	Power to 5.25-inch diskette drive

Modem Commands and Operation

The modem has three states of operation: the command state, the data or online state, and the idle state. In normal operation, the modem is in either the command state, waiting to receive a command from the user, or in the data state, performing data transmission and reception.

After a system power on or reset, the modem enters the idle state. When the modem in the idle state receives the 2-character sequence AT from the user, AT serves as the "wakeup" code and the modem enters the command state. In the command state, the modem waits to receive a command from the user.

Appropriate commands make the modem enter the data or online state. The data state is the normal online communication state of the modem. Data is transmitted and received in the data state. If an escape sequence (the characters + + +) is detected, the modem returns to the command state.

When in the data state, the modem is in either the originate mode or the answer mode. In the originate mode, the modem can originate or dial calls. In the answer mode, the modem can receive incoming calls.

AT (Attention) Command Description

This modem is controlled and configured by the AT (Attention) commands. Each command consists of the following elements:

- The 2-character sequence AT
- A command
- A command parameter
- A carriage return (Enter).

The 2-character sequence AT must be uppercase. The command may be uppercase or lowercase. Spaces are ignored. A command is not performed until a carriage return (Enter) is received. For example, to enter the command "Answer," type **ATA** or **ATa** and press Enter.

Some commands do not have parameters. Any missing parameter in a command is assigned the value zero, which may be a valid parameter for the command. The sequence AT followed by Enter causes the modem to enter the command state. That is, AT without a command serves as the "wakeup" code, and OK appears on the screen.

The modem queues commands in a 40-character command line. The command line begins with AT and can contain several commands. A separator is not required between the commands.

The command line format is:

AT *command* [*parameter*] *command* [*parameter*]...Enter

When a carriage return is received, which terminates the line, the commands are performed in the order in which they are sent to the modem. After the commands are performed, an OK is returned to the computer. If more than 40 characters including spaces are sent to the modem, an error occurs and all commands must be reentered.

Notes:

1. Allow 3 seconds for the transition between command and data states. This pause allows for the modem operation to complete.
2. Data Terminal Ready (DTR) in the modem control register must be active in order to send commands to the modem.
3. The electronics used in this modem have a general purpose internal microcode that provides capabilities not supported by this design. An OK may be returned in response to a command but the modem will not respond.
4. Supplementry information concerning the AT command definitions may be found in "89024 Reference Manual." See "Bibliography" on page X-19.
5. The different modem hardware/software levels can be identified by using the ATI1 Command. For example, 194 = Level 1.6; 123 = Level 2.0.

AT Commands

Command	Command Name and Function
A	**Answer** This command takes the modem "off hook" and into answer mode. The modem does not wait for a ring on the line. If no carrier is received after the wait specified in Register S7, the modem returns to the command mode.
A/	**Re-execute** The command A/ with no preceding AT command and no carriage return causes the most recent command line to be re-executed.
B	**BELL/CCITT Handshake Default** B0 CCITT standards (Supported in Level 2.0) B1 BELL standards. (default) This command selects between the CCITT and BELL standards for 300 and 1200 bps data rate connections.

Command	Command Name and Function
D	**Dial (D n...n)** This command takes the modem "off hook" and dials the characters contained in the parameter, and controls the modem during dialing. If **D** is entered with no parameters, the modem goes "off hook" without dialing. The parameter may consist of a combination of these characters: **0 1 2 3 4 5 6 7 8 9 A B C D # * ( ) - T P R W , ; ! @ . / S** • **0 1 2 3 4 5 6 7 8 9 A B C D # *** = Dialed characters. • **( ) - . /** = Punctuation, including spaces, used to increase readability. • **T** = Switches modem to tone dialing. An example using tone dialing T is: ATDT 555-1234 CR. The speed for tone dial is set by Register S11. The command AT = attention, D = dial, T = tone, 555-1234 = dialed characters, and CR = carriage return or Enter. • **P** = Switches modem to pulse dialing (default). An example is: ATDP 555-1234 CR. • **R** = Puts modem in answer mode. The R is placed at the end of the dial string and is used for communicating with originate-only modems. An example is: ATDT 555-1234R CR. • **W** = Wait for dial tone. The W causes the modem to wait for a dial tone for the time specified in Register S7. If no dial tone is detected after this time, the modem hangs up and indicates that no dial tone was received. An example showing this command with a 9 to dial outside a local telephone network is: ATDT 9W555-1234 CR. • **,** = Modem pause (Default is 2 seconds). The comma causes the modem to pause for a specified period. An example is: ATDT 9,555-1234 CR. • **;** = Returns the modem to command state. The semicolon is placed at the end of a dial string to put the modem in a state that accepts commands after dialing. An example is: ATDT 555-1234;CR. • **! FLASH** = Causes the modem to go "on hook" for half a second. FLASH is used for transferring calls. • **@** = Wait for quiet answer. The @ causes the modem to wait for the time specified in Register S7 for at least one valid ringback, followed by 5 seconds of silence before executing the next character in the dialing string. • S = n. Dial stored number *not supported.*

Command	Command Name and Function
E	**Echo** **E0** = Disable **E1** = Enable (default). This command controls whether characters sent to the modem by the computer while in the command state are echoed back to the computer and displayed on the screen.
H	**On/Off Hook** **H0** = On hook (default) **H1** = Off hook. This command controls the telephone switch "hook." The H0 command forces the modem to go "on hook." The H1 command forces the modem "off hook."
I	**Identification Code** **I0** = Returns the product code (249) **I1** = Returns the checksum (194 or 123) **I2** = Returns OK. This command supplies identification information about the modem.
L	**Speaker Volume** **L0** = Low volume **L1** = Low volume **L2** = Medium volume (default) **L3** = High volume. This command controls the volume of the signal to the speaker. The local volume control on the monitor provides additional volume control.
M	**Speaker** **M0** = Disable speaker **M1** = Enable speaker until carrier acknowledged (default) **M2** = Speaker always on when "off hook." **M3** = Speaker is on until carrier is detected, but off while dialing. The command controls the computer speaker.
O	**Online** **O** = Online. This command forces data mode after an established call has been interrupted by the escape sequence. If "on hook," this command puts the modem "off hook" and in the originate mode. **O1** = Online and retrain. This command causes the modem to return to the online state and initiate a retrain sequence (in 2400 bps only).

Command	Command Name and Function
Q	**Result Codes** **Q0** = Returned (default) **Q1** = Not returned. This command controls whether result codes are returned to the computer after a command is performed. The result code can be either descriptive words or digit codes. Two other commands also control the result codes (see the commands **V** and **X**). See "Result (Return) Codes" on page 9-14.
Sr?	**Read Register** **r** = 0 - 27 This command causes the content of the register r to be read. See "Register Summary" on page 9-15.
Sr = n	**Set Register** **r** = 0 - 27 **n** = 0 - 255. This command causes the register r to be set to value n. See "Register Summary" on page 9-15.
V	**Verbose/Terse Result Codes** **V0** = Terse **V1** = Verbose (default). This command controls the type of result codes returned to the computer from the modem. V0 causes the result codes to be sent as digits. V1 causes the result codes to be sent as words.
X	**Basic/Extended Result Code Set** **X0** = Connect message, blind dial, no timeout, no busy detect **X1** = Full messages, blind dial, no timeout, no busy detect **X2** = Full messages, dial tone dialing, dial tone timeout, no busy detect **X3** = Full messages, blind dial, no timeout, busy detect **X4** = Full messages, dial tone dialing, dial tone timeout, busy detect (default). This command selects the result code to be returned. See "Result (Return) Codes" on page 9-14.
Y	**Enable Long Space Disconnect** **Y0** = Disable option (default) **Y1** = Enable option. This command sets the modem to disconnect when a space of 1.6 seconds duration is received from the remote modem.

Command	Command Name and Function
Z	**Software Reset** This command resets the modem to its default settings and leaves the modem in the command state. Any commands on the command line after the **Z** command are not performed because the reset clears the command buffer. If the modem is online, the reset breaks the connection. The application should wait 2 seconds before issuing the next command to the modem.
&Cn	**Carrier Signal Control CXR** **Received Level Signal Detect (RLSD)** **&C0** = Sets RLSD bit on—carrier forced on (default in Level 1.6) **&C1** = Allows RLSD to track actual carrier (default in Level 2.0). This command causes the modem to detect and track the carrier signal.
&Dn	**n = 0..3** (Register S25) **0** = DTR ignored **1** = 1 to 0 of DTR toggles command state (default in Level 1.6) **2** = 1 to 0 of DTR toggles on hook (default in Level 2.0) Disabled auto answer, enter command state. **3** = 1 to 0 of DTR toggles reset state.
&F	Fetch Factory Configuration Profile.
&Gn	**n = 0..2** **0** = No guard tone (default) **1** = 550 Hz guard tone **2** = 1800 Hz guard tone.
&Jn	**n = 0,1** *Not supported* **0** = RJ-11/RJ-41/RJ-45S (default) **1** = RJ-12/RJ-13.
&Ln	**n = 0,1** *Not supported* **0** = Dialup lines (default) **1** = Leased lines.
&Mn	**n = 0..3** *Not supported* **0** = Asynchronous (default) **1** = Sync mode 1 async dial **2** = Sync mode 2 DTR dial **3** = Sync mode 3 manual dial.

Command	Command Name and Function
&Pn	**n = 0,1** **0** = 39% make/61% break (default) U.S. **1** = 33% make/67% break. U.K.
&Rn	**n = 0,1** *Not supported* **0** = CTS responds to RTS synchronous (default) **1** = Ignore RTS synchronous; CTS always on in async.
&Sn	**n = 0,1** **0** = DSR on (default) **1** = DSR to CCITT V.22 bis V.22.
&Tn	**n = 0..8** **0** = Terminate test **1** = Local analog loopback **2** = *Not supported* **3** = Local digital loopback **4** = Remote digital loopback (default) **5** = Inhibit remote digital loopback **6** = Local testing **7** = Remote digital loopback with self test CCITT V.54 **8** = Local analog loopback with self test CCITT V.54.
&W	Write configuration to NVRAM. Not supported
&Xn	**n = 0..2** *Not supported* **0** = Use transmit clock (default) **1** = Use external transmit clock **2** = Use data carrier clock.
&Zn	**n = 0..9, ABCD#*TPRW!@;** *Not supported* Store telephone number in NVRAM.
+ + +	**Escape Sequence** The escape sequence (three consecutive escape characters preceded and followed by the pause set in register S12) changes the modem from the data state to the command state. The escape character is selectable (see register S2). The plus sign, ASCII 43 (hex 2B), is the default. ASCII 28 (hex 1C) or ASCII 29 (hex 1D) may also be used.

Result (Return) Codes

The set of result codes that are returned can be selected (see the command **X**). Digit codes are followed by a carriage return character. Word responses are preceded and followed by a carriage return character and a line feed character.

Digit Code	Word	Description
0	OK	Command line executed without errors.
1	CONNECT	Carrier detected at 300 bps.
2	RING	Ringing signal detected (incoming).
3	NO CARRIER	Carrier lost or never connected.
4	ERROR	Error in command line • Invalid command • Invalid parameter in command statement • Command line exceeds 40 characters.
5	CONNECT 1200	Carrier detected at 1200 bps.
6	NO DIALTONE	No dial tone detected within timeout.
7	BUSY	Busy signal detected.
8	NO ANSWER	No answer within timeout period.
9	CONNECT 600	*Not supported.*
10	CONNECT 2400	Carrier detected at 2400 bps.

Register Summary

Register	Range	Default	Function
S0	0–255 Rings	0	Number of rings before modem automatically answers. A value of 0 (default) disables automatic answering.
S1	0–255 Rings	0	Number of rings that have occurred. Resets to 0 after eight seconds with no rings (read-only register).
S2	0–255 ASCII 0–127 128–255	043	Escape code character. Escape code characters enabled. Escape code characters disabled.
S3	0–255 ASCII	013	Command line terminator. Default is the carriage return character.
S4	0–255 ASCII	010	Line feed character. This character follows the carriage return character in a full-word return code. Default is the line feed character.
S5	0–255 ASCII	008	Backspace character. This character erases the last character in the command line being entered. The default is the backspace character.
S6	0–255 seconds	2	Wait time for dial tone. This is the delay between going "off hook" and beginning to dial if the result code set is X0, X1, or X3. The dial tone is assumed to be present after the delay has passed. With values below 2, the delay is 2 seconds. Dialing begins when the dial tone is detected if the result code set is X2 or X4. If no dial tone is detected within 5 seconds, the modem goes "on hook" and responds with the result NO DIAL TONE.
S7	0–255 seconds	30	Wait time for carrier after dialing.

Register	Range	Default	Function
S8	0–255 seconds	2	Length of pause caused by comma in the dialed characters.
S9	0–255 secs/10	6	Carrier valid delay. To be detected, the carrier must be present for this time.
S10	0–255 secs/10	14	Time between loss of carrier and going "on hook." Setting of 255 disables carrier-loss disconnect.
S11	50–255 secs/1000	100	Dialing tones duration and spacing. *Not supported* in Level 1.6.
S12	20–255 secs/20	50	Escape sequence guard time. The dialed character stream must be silent for this time before and after the escape sequence to enter the command state from the data state.
S13			Not used.
S14			Bit-mapped register.
S15			Not used.
S16			Modem test options.
S17			Not used.
S18	0–255 seconds	0	Test timer.
S19			Not used.
S20			Not used.
S21			Bit-mapped register.
S22			Bit-mapped register.
S23			Bit-mapped register.
S24			Not used.
S25			Delay to DTR.
	1–255 secs	5sec	Synchronous mode M1.
	0–2.55 secs	50ms	Asynchronous mode M2.
	0–2.55 secs	50ms	Synchronous mode M3.
S26	0–2.55 secs	1	RTS to CTS delay.
S27			Bit-mapped register.

Bit-Mapped Registers

Figure 9-1. S14 Options Register

Register	Bit	Value	Description
S14	0		Not used.
	1	0	Local echo disabled.
		1	Local echo enabled (default).
	2	0	Result codes enabled (default).
		1	Result codes off.
	3	0	Result codes digits.
		1	Result codes words (default).
	4	0	Smart mode (default).
		1	Dumb mode.
	5	0	Touch tone.
		1	Pulse (default).
	6		Not used.
	7	0	Answer.
		1	Originate (default).

Figure 9-2. S16 Modem Test Options Register

Register	Bit	Value	Description
S16	0	0	Local analog loop disabled (default).
		1	Local analog loop enabled.
	1		Not used.
	2	0	Local digital loop disabled (default).
		1	Local digital loop enabled.
	3	0	Loopback off (default).
		1	Loopback on.
	4	0	Disable remote digital loop (default).
		1	Initiate remote digital loop.
	5	0	Disable expanded remote digital loopback with self-test (default).
		1	Initiate expanded remote digital loopback with self-test.
	6	0	Disable local analog loopback with self-test (default).
		1	Initiate local analog loopback with self-test.
	7		Not used.

Figure 9-3. S21 Modem Options Register

Register	Bit	Value	Description
S21	0	0	RJ11/RJ14/RJ45 (default).
		1	RJ12/RJ13. *Not supported.*
	1		Not used.
	2	0	CTS follows RTS (default).
		1	CTS follows carrier signal.
	4 3	0 / 0	Modem ignores DTR.
		0 / 1	Command state on-to-off DTR (default in Level 1.6).
		1 / 0	Hangs on on-to-off DTR (default in Level 2.0).
		1 / 1	Initialize on on-to-off DTR.
	5	0	RLSD always on (default in Level 1.6).
		1	RLSD indicates valid carrier (default in Level 2.0).
	6	0	DSR always on (default).
		1	Modem off hook in data mode.
	7	0	Disable long space disconnect (default).
		1	Enable long space disconnect.

Figure 9-4. S22 Options Register

Register	Bit	Value	Description
S22	1 0	0 / 0	Undefined.
		0 / 1	Low speaker volume.
		1 / 0	Medium speaker volume (default).
		1 / 1	High speaker volume.
	3 2	0 / 0	Speaker disabled.
		0 / 1	Speaker on until carrier (default).
		1 / 0	Speaker always on.
		1 / 1	Speaker on until carrier but off during dial.
	6 5 4	0 / 0 / 0	2400 result codes, blind dial.
		0 / 0 / 1	Undefined.
		0 / 1 / 0	Undefined.
		0 / 1 / 1	Undefined.
		1 / 0 / 0	Full codes, blind dial.
		1 / 0 / 1	Full codes, dial tone wait.
		1 / 1 / 0	Full, blind, busy detect.
		1 / 1 / 1	Full, wait, busy (default).
	7	0 = 39/61	Make / break (default).
		1 = 33/67	Make / break.

Figure 9-5. S23 Options Register

Register	Bit	Value	Description
S23	0	0	Ignore remote loopback request.
		1	Accept remote loopback request (default).
	2 1	0 / 0	0 to 300 bps.
		0 / 1	600 bps. *Not supported.*
		1 / 0	1200 bps.
		1 / 1	2400 bps (default).
	3		Undefined.
	5 4	0 / 0	Even parity (default).
		0 / 1	Space parity.
		1 / 0	Odd parity.
		1 / 1	Mark / no parity.
	7 6	0 / 0	No guard tones (default).
		0 / 1	550 Hz guard tone.
		1 / 0	1800 Hz guard tone.
		1 / 1	Not used.

Figure 9-6. S27 Options Register

Register	Bit	Value	Description
S27	1 0	0 / 0	Asynchronous (default).
		0 / 1	Sync mode 1, async dial.
		1 / 0	Sync mode 2, DTR dial.
		1 / 1	Sync mode 3, manual dial.
	2	0	Dialup line (default).
		1	Leased line.
	3		Undefined.
	5 4	0 / 0	EIA pin 15 clock source (default).
		0 / 1	EIA pin 24 clock source.
		1 / 0	Clock from receive carrier.
		1 / 1	Not used.
	6	0	CCITT V.22 bis / V.22.
		1	BELL 212A (default).
	7		Not used.

Specifications

Dialer Type

- Pulse dialing
- Dual Tone Multifrequency (DTMF).

Pulse Dialing

- Rate: 10 pulses per second
- Duty Cycle: Relay make/break 39 ms/61 ms, interdigit delay 785 ms.

DTMF Dialing

Modulation

V.21 and Bell 103:	Frequency shift keying (FSK)
V.22 and 212 Mode:	Four-level phase shift keying (PSK)
V.22 bis:	16-point QAM

Transmitter Characteristics

The transmit level in either 103 mode or 212 mode is -11 dbm, ±2dbm.

Receiver Characteristics

- Carrier Detect Threshold

  Carrier detect threshold off: −48 dbm
  Carrier detect threshold on: −43 dbm
  Maximum input level: −9 dbm.

- Incoming Ring Detection. The modem detects as a ring this range of "on hook" voltages: 40 to 130 volts RMS at 16 to 68 Hz.

Analog Interface

The modem is provided with one RJ-11 connector. Permissive telephone line connection is made through a detachable 3,048 mm (10 foot) cable terminated at both ends with a modular six-position plug. The plug connects to a USOC RJ11C wall outlet.

- USOC RJ 11C

Both the telephone line and a voice telephone handset can be connected to the modem with a T-adapter plugged into the jack of the wall outlet. A telephone handset is required only for manually originating and answering calls.

Note: Picking up the handset while the modem is processing data causes errors in data transmission or termination of the connection.

RS-232C Interface Card

Description

This card attaches to the serial interface connector on the system board and provides RS-232C (CCITT V.24) voltages to a 25-pin, D-shell connector.

See Figure 3-146 on page 3-148 for pin assignments for connection of the card to the system board.

Voltage Interchange Information

The signal is considered in the marking condition when the voltage on the interchange circuit, measured at the interface point, is more negative than -3V dc with respect to signal ground. The signal is considered in the spacing condition when the voltage is more than +3V dc with respect to signal ground. The region between +3V dc and -3V dc is defined as the transition region and considered an invalid level. A voltage that is more negative than -15V dc or more positive than +15V dc is considered and invalid level.

Figure 9-7. Voltage Levels

Interchange Voltage	Binary State	Signal Condition	Interface Control Function
Positive Voltage	0	Spacing	On
Negative Voltage	1	Marking	Off

RS-232C Connector

The interface uses the standard D-shell connector and pin assignments defined for RS-232C. The voltage levels are EIA only. Current loop interface is not supported.

Figure 9-8 shows the pin assignments for the serial port in a communications environment.

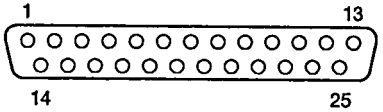


Figure 9-8. Serial Port Connector Pin Assignments

Pin No.	I/O	Signal Name	Pin No.	I/O	Signal Name
1	NA	Not Connected	13	NA	Not Connected
2	O	Transmit Data	14	NA	Not Connected
3	I	Receive Data	15	NA	Not Connected
4	O	Request to Send	16	NA	Not Connected
5	I	Clear to Send	17	NA	Not Connected
6	I	Data Set Ready	18	NA	Not Connected
7	NA	Signal Ground	19	NA	Not Connected
8	I	Received Level Signal Detect	20	O	Data Terminal Ready
			21	NA	Not Connected
9	NA	Not Connected	22	I	Ring Indicate
10	NA	Not Connected	23	NA	Not Connected
11	O	Data Terminal Ready	24	NA	Not Connected
12	NA	Not Connected	25	NA	Not Connected

Section 10. 80286 Microprocessor Instruction Set

80286 Microprocessor Instruction Set 10-3
- Data Transfer 10-3
- Arithmetic 10-8
- Logic 10-13
- String Manipulation 10-16
- Repeated String Manipulation 10-17
- Control Transfer 10-18
- Processor Control 10-23
- Protection Control 10-25

80286 Microprocessor Instruction Set

Data Transfer

MOV = move

Register to Register/Memory

1000100w	mod reg r/m

Register/Memory to Register

1000101w	mod reg r/m

Immediate to Register/Memory

1100011w	mod 000 r/m	data	data if w = 1

Immediate to Register

1011w reg	data	data if w = 1

Memory to Accumulator

1010000w	addr-low	addr-high

Accumulator to Memory

1010001w	addr-low	addr-high

Register/Memory to Segment Register

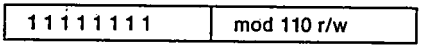
10001110	mod 0 reg r/m

Segment Register to Register/Memory

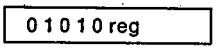
10001100	mod 0 reg r/m

PUSH = Push

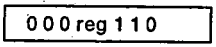
Memory



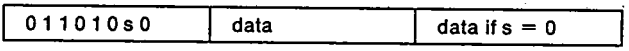
Register



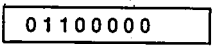
Segment Register



Immediate



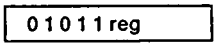
PUSHA = Push All



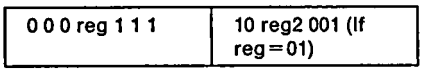
POP = Pop

Register/Memory

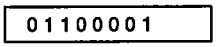
Register



Segment Register

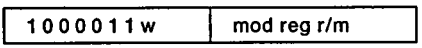


POPA = Pop All

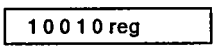


XCHG = Exchange

Register/Memory with Register



Register With Accumulator



IN = Input From

Fixed Port

1110010w	port number

Variable Port

1110110w

OUT = Output To

Fixed Port

1110011w	port number

Variable Port

1110111w

XLAT = Translate Byte to AL

11010111

LEA = Load EA to Register

10001101	mod reg r/m

LDS = Load Pointer to DS

11000101	mod reg r/m mod ≠ 11

LES = Load Pointer to ES

11000100	mod reg r/m mod ≠ 11

LAHF = Load AH into Flags

10011111

SAHF = Store AH Into Flags

10011110

PUSHF = Push Flags

10011100

POPF = Pop Flags

10011101

Arithmetic

ADD = Add

Register/Memory with Register to Either

000000dw	mod reg r/m

Immediate to Register/Memory

100000sw	mod 000 r/m	data	data if sw = 01

Immediate to Accumulator

0000010w	data	data if w = 1

ADC = Add with Carry

Register/Memory with Register to Either

000100dw	mod reg r/m

Immediate to Register/Memory

100000sw	mod 010 r/m	data	data if sw = 01

Immediate to Accumulator

0001010w	data	data if w = 1

INC = Increment

Register/Memory

1111111w	mod 000 r/m

Register

01000reg

SUB = Subtract

Register/Memory with Register to Either

0 0 1 0 1 0 dw	mod reg r/m

Immediate from Register/Memory

1 0 0 0 0 0 sw	mod 101 r/m	data	data if sw = 01

Immediate from Accumulator

0 0 1 0 1 1 0 w	data	data if w = 1

SBB = Subtract with Borrow

Register/Memory with Register to Either

0 0 0 1 1 0 dw	mod reg r/m

Immediate from Register/Memory

1 0 0 0 0 0 sw	mod 011 r/m	data	data if sw = 01

Immediate from Accumulator

0 0 0 1 1 1 0 w	data	data if w = 1

DEC = Decrement

Register/Memory

1 1 1 1 1 1 1 w	reg 001 r/m

Register

0 1 0 0 1 reg

CMP = Compare

Register/Memory with Register

0 0 1 1 1 0 dw	mod reg r/m

Register with Register/Memory

0 0 1 1 1 0 dw	mod reg r/m

Immediate with Register/Memory

1 0 0 0 0 0 sw	mod 111 r/m	data	data if sw = 01

Immediate with Accumulator

0 0 1 1 1 1 0 w	data	data if w = 1

NEG = Change Sign

1 1 1 1 0 1 1 w	mod 011 r/m

AAA = ASCII Adjust for Add

0 0 1 1 0 1 1 1

DAA = Decimal Adjust for Add

0 0 1 0 0 1 1 1

AAS = ASCII Adjust for Subtract

0 0 1 1 1 1 1 1

DAS = Decimal Adjust for Subtract

0 0 1 0 1 1 1 1

MUL = Multiply (Unsigned)

1 1 1 1 0 1 1 w	mod 100 r/m

IMUL = Integer Multiply (Signed)

1 1 1 1 0 1 1 w	mod 100 r/m

IIMUL = Integer Immediate Multiply (Signed)

0 1 1 0 1 0 s 1	mod reg r/m	data	data if s = 0

DIV = Divide (Unsigned)

1111011w	mod 110 r/m

IDIV = Integer Divide (Signed)

1111011w	mod 111 r/m

AAM = ASCII Adjust for Multiply

11010100	00001010

AAD = ASCII Adjust for Divide

11010101	00001010

CBW = Convert Byte to Word

10011000

CWD = Convert Word to Double Word

10011001

Logic

Shift/Rotate Instructions

Register/Memory by 1

1 1 0 1 0 0 0 w	mod TTT r/m

Register/Memory by CL

1 1 0 1 0 0 1 w	mod TTT r/m

Register/Memory by Count

1 1 0 0 0 0 0 w	mod TTT r/m	count

TTT	Instruction
0 0 0	ROL
0 0 1	ROR
0 1 0	RCL
0 1 1	RCR
1 0 0	SHL/SAL
1 0 1	SHR
1 1 1	SAR

AND = And

Register/Memory and Register to Either

0 0 1 0 0 0 d w	mod reg r/m

Immediate to Register/Memory

1 0 0 0 0 0 0 w	mod 100 r/m	data	data if w = 1

Immediate to Accumulator

0 0 1 0 0 1 0 w	data	data if w = 1

TEST = AND Function to Flags; No Result

Register/Memory and Register

1000010w	mod reg r/m

Immediate Data and Register/Memory

1111011w	mod 000 r/m	data	data if w = 1

Immediate Data and Accumulator

1010100w	data	data if w = 1

OR = Or

Register/Memory and Register to Either

000010dw	mod reg r/m

Immediate to Register/Memory

1000000w	mod 001 r/m	data	data if w = 1

Immediate to Accumulator

0000110w	data	data if w = 1

XOR = Exclusive OR

Register/Memory and Register to Either

001100dw	mod reg r/m

Immediate to Register/Memory

1000000w	mod 110 r/m	data	data if w = 1

Immediate to Accumulator

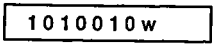
0011010w	data	data if w = 1

NOT = Invert Register/Memory

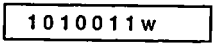
1111011w	mod 010 r/m

String Manipulation

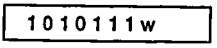
MOVS = Move Byte Word



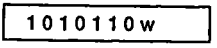
CMPS = Compare Byte Word



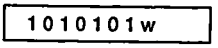
SCAS = Scan Byte Word



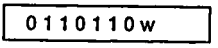
LODS = Load Byte/Word to AL/AX



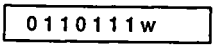
STOS = Store Byte/Word from AL/AX



INS = Input Byte/Word from DX Port



OUTS = Output Byte/Word to DX Port



Repeated String Manipulation

REP/REPNE, REPZ/REPNZ = Repeat String

Repeat Move String

11110011	1010010w

Repeat Compare String (z/Not z)

1111001z	1010011w

Repeat Scan String (z/Not z)

1111001z	1010111w

Repeat Load String

11110011	1010110w

Repeat Store String

11110011	1010101w

Repeat Input String

11110011	0110110w

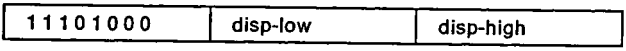
Repeat Output String

11110011	0110111w

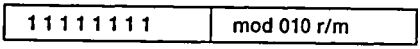
Control Transfer

CALL = Call

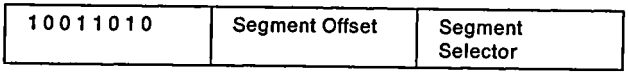
Direct Within Segment



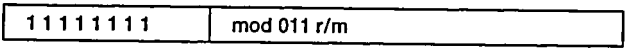
Register/Memory Indirect Within Segment



Direct Intersegment

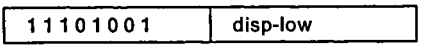


Indirect Intersegment

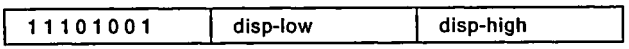


JMP = Unconditional Jump

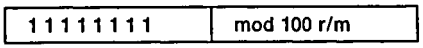
Short



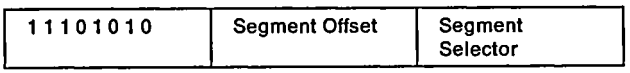
Direct within Segment



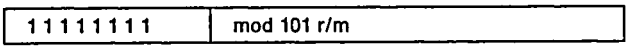
Register/Memory Indirect Within Segment



Direct Intersegment

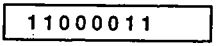


Indirect Intersegment

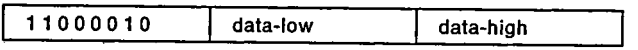


RET = Return from Call

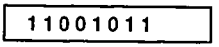
Within Segment



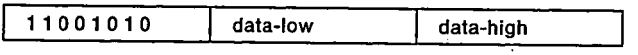
Within Segment Adding Immediate to SP



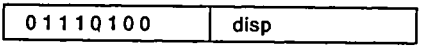
Intersegment



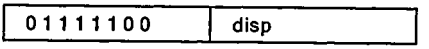
Intersegment Adding Immediate to SP



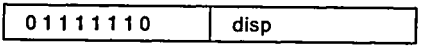
JE/JZ = Jump on Equal/Zero



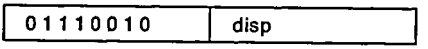
JL/JNGE = Jump on Less/Not Greater or Equal



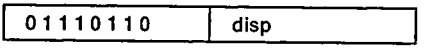
JLE/JNG = Jump on Less or Equal/Not Greater



JB/JNAE = Jump on Below/Not Above or Equal



JBE/JNA = Jump on Below or Equal/Not Above



JP/JPE = Jump on Parity/Parity Even

01111010	disp

JO = Jump on Overflow

01110000	disp

JS = Jump on Sign

01111000	disp

JNE/JNZ = Jump on Not Equal/Not Zero

01110101	disp

JNL/JGE = Jump on Not Less/Greater or Equal

01111101	disp

JNLE/JG = Jump on Not Less or Equal/Greater

01111111	disp

JNB/JAE = Jump on Not Below/Above or Equal

01110011	disp

JNBE/JA = Jump on Not Below or Equal/Above

01110111	disp

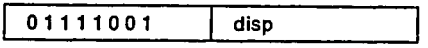
JNP/JPO = Jump on Not Parity/Parity Odd

01111011	disp

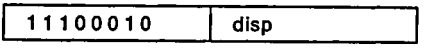
JNO = Jump on Not Overflow

01110001	disp

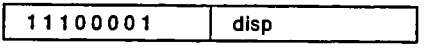
JNS = Jump on Not Sign



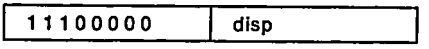
LOOP = Loop CX Times



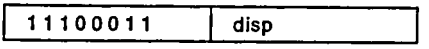
LOOPZ/LOOPE = Loop while Zero/Equal



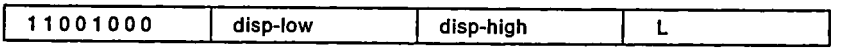
LOOPNZ/LOOPNE = Loop while Not Zero/Not Equal



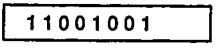
JCXZ = Jump on CX Zero



ENTER = Enter Procedure

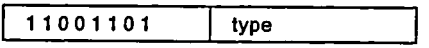


LEAVE = Leave Procedure

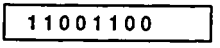


INT = Interrupt

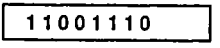
Type Specified



Type 3



INTO = Interrupt on Overflow



IRET = Interrupt Return

11001111

BOUND = Detect Value Out of Range

01100010	mod reg r/m

Processor Control

CLC = Clear Carry

11111000

CMC = Complement Carry

11110101

STC = Set Carry

11111001

CLD = Clear Direction

11111100

STD = Set Direction

11111101

CLI Clear Interrupt Enable Flag

11111010

STI = Set Interrupt Enable Flag

11111011

HLT = Halt

11110100

WAIT = Wait

10011011

LOCK = Bus Lock Prefix

11110000

CTS = Clear Task Switched Flag

00001111	00000110

ESC = Processor Extension Escape

11011TTT	mod LLL r/m

Protection Control

LGDT = Load Global Descriptor Table Register

00001111	00000001	mod 010 r/m

SGDT = Store Global Descriptor Table Register

00001111	00000001	mod 000 r/m

LIDT = Load Interrupt Descriptor Table Register

00001111	00000001	mod 011 r/m

SIDT = Store Interrupt Descriptor Table Register

00001111	00000001	mod 001 r/m

LLDT = Load Local Descriptor Table Register from Register/Memory

00001111	00000000	mod 010 r/m

SLDT = Store Local Descriptor Table Register to Register/Memory

00001111	00000000	mod 000 r/m

LTR = Load Task Register from Register/Memory

00001111	00000000	mod 011 r/m

STR = Store Task Register to Register/Memory

00001111	00000000	mod 001 r/m

LMSW = Load Machine Status Word from Register/Memory

00001111	00000001	mod 110 r/m

SMSW = Store Machine Status Word

00001111	00000001	mod 100 r/m

LAR = Load Access Rights from Register/Memory

00001111	00000010	mod reg r/m

LSL = Load Segment Limit from Register/Memory

00001111	00000011	mod reg r/m

ARPL = Adjust Requested Privilege Level from Register/Memory

01100011	mod reg r/m

VERR = Verify Read Access

00001111	00000000	mod 100 r/m

VERW = Verify Write Access

00001111	00000000	mod 101 r/m

The effective address (EA) of the memory operand is computed according to the mod and r/m fields:

If mod = 11, then r/m is treated as a reg field.
If mod = 00, then disp = 0, disp-low and disp-high are absent.
If mod = 01, then disp = disp-low sign-extended to 16 bits, disp-high is absent.
If mod = 10, then disp = disp-high:disp-low.

If r/m = 000, then EA = (BX) + (SI) + DISP
If r/m = 001, then EA = (BX) + (DI) + DISP
If r/m = 010, then EA = (BP) + (SI) + DISP
If r/m = 011, then EA = (BP) + (DI) + DISP
If r/m = 100, then EA = (SI) + DISP
If r/m = 101, then EA = (DI) + DISP
If r/m = 110, then EA = (BP) + DISP
If r/m = 111, then EA = (BX) + DISP.

DISP follows the second byte of the instruction (before data if required).

Note: An exception to the above statements occurs when mod=00 and r/m=110, in which case EA = disp-high; disp-low.

Segment Override Prefix

0 0 1 reg 1 1 0

The 2-bit and 3-bit reg fields are defined as follows:

Figure 10-1. 2-Bit Register Field

reg	Segment Register	reg	Segment Register
00	ES	10	SS
01	CS	11	DS

Figure 10-2. 3-Bit Register Field

16-bit (w = 1)	8-bit (w = 0)
000 AX	000 AL
001 CX	001 CL
010 DX	010 DL
011 BX	011 BL
100 SP	100 AH
101 BP	101 CH
110 SI	110 DH
111 DI	111 BH

The physical addresses of all operands addressed by the BP register are computed using the SS segment register. The physical addresses of the destination operands of the string primitive operations (those addressed by the DI register) are computed using the ES segment, which cannot be overridden.

Section 11. Characters and Keystrokes

Available Code Pages 11-3
Character Codes 11-4
Quick Reference 11-11

Available Code Pages

DOS allows switching among several pairs of code pages, depending on country, as shown in the table below.

Country	Country Code	Code Pages	Keyboard ID	Keyboard Layout
Belgium (French-speaking)	032	850, 437	120	FR
Belgium (Dutch-speaking)	032	850, 437	120	FR
Canada (French-speaking)	002	850, 863	058	CF
Canada (English-speaking)	001	850, 437	103	US
Denmark	045	850, 865	159	DK
Finland	358	850, 437	153	SU
France	033	850, 437	120	FR
Germany	049	850, 437	129	GR
Italy	039	850, 437	142	IT
Netherlands	031	850, 437	143	NL
Norway	047	850, 865	155	NL
Portugal	351	850, 860	163	PO
Spain	034	850, 437	172	SP
Sweden	046	850, 437	153	SV
Switzerland (French-speaking)	041	850, 437	150	SF
Switzerland (German-speaking)	041	850, 437	000	SG
United Kingdom	044	850, 437	166	UK
United States	001	437, 850	103	US

Character Codes

This section provides charts showing the decimal values, hexadecimal values, and keystrokes for each character. The notes referred to in the charts are located following the charts.

The graphic characters in the charts are those found in Code Page 437, the U.S. default. They are used as an example. Other code pages are shown in the "Quick Reference" beginning on page 11-11. The keystrokes in the charts cause the active code page characters to be displayed.

Many programs use some hexadecimal values for control functions and do not allow the assigned graphic characters in the charts to be displayed. Hexadecimal values having standard control functions are indicated with a note.

Value		As Characters		
Hex	Dec	Symbol	Keystrokes	Notes
00	0	Blank (Null)	Ctrl 2	
01	1	☺	Ctrl A	
02	2	☻	Ctrl B	
03	3	♥	Ctrl C	
04	4	♦	Ctrl D	
05	5	♣	Ctrl E	
06	6	♠	Ctrl F	
07	7	•	Ctrl G	Note 7
08	8	◘	Ctrl H, Backspace, Shift Backspace	Note 7
09	9	○	Ctrl I Tab	Note 7
0A	10	◙	Ctrl J, Ctrl ↵	Note 7
0B	11	♂	Ctrl K	Note 7
0C	12	♀	Ctrl L	Note 7
0D	13	♪	Ctrl M, ↵, Shift ↵	Note 7
0E	14	♫	Ctrl N	Note 7
0F	15	☼	Ctrl O	Note 7
10	16	►	Ctrl P	Note 7
11	17	◄	Ctrl Q	Note 7
12	18	↕	Ctrl R	Note 7
13	19	‼	Ctrl S	Note 7
14	20	¶	Ctrl T	Note 7
15	21	§	Ctrl U	
16	22	▬	Ctrl V	
17	23	↨	Ctrl W	

Value		As Characters		
Hex	Dec	Symbol	Keystrokes	Notes
18	24	↑	Ctrl X	Note 7
19	25	↓	Ctrl Y	
1A	26	→	Ctrl Z	
1B	27	←	Ctrl [Esc, Shift Esc, Ctrl Esc	Note 7
1C	28	∟	Ctrl /	
1D	29	↔	Ctrl]	
1E	30	▲	Ctrl 6	
1F	31	▼	Ctrl -	
20	32	Blank Space	Space Bar, Shift, Space, Ctrl Space, Alt Space	
21	33	!	!	Shift
22	34	"	"	Shift
23	35	#	#	Shift
24	36	$	$	Shift
25	37	%	%	Shift
26	38	&	&	Shift
27	39	'	'	Shift
28	40	(	(	Shift
29	41	)	)	
2A	42	*	*	Note 1
2B	43	+	+	Shift
2C	44	,	,	
2D	45	-	-	
2E	46	.	.	Note 2

Value		As Characters		
Hex	Dec	Symbol	Keystrokes	Notes
2F	47	/	/	
30	48	0	0	Note 3
31	49	1	1	Note 3
32	50	2	2	Note 3
33	51	3	3	Note 3
34	52	4	4	Note 3
35	53	5	5	Note 3
36	54	6	6	Note 3
37	55	7	7	Note 3
38	56	8	8	Note 3
39	57	9	9	Note 3
3A	58	:	:	Shift
3B	59	;	;	
3C	60	<	<	Shift
3D	61	=	=	
3E	62	>	>	Shift
3F	63	?	?	Shift
40	64	@	@	Shift
41	65	A	A	Note 4
42	66	B	B	Note 4
43	67	C	C	Note 4
44	68	D	D	Note 4
45	69	E	E	Note 4
46	70	F	F	Note 4
47	71	G	G	Note 4
48	72	H	H	Note 4
49	73	I	I	Note 4
4A	74	J	J	Note 4

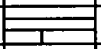
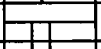
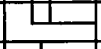
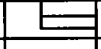
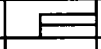
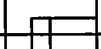
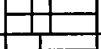
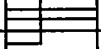
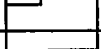
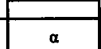
Value		As Characters		
Hex	Dec	Symbol	Keystrokes	Notes
4B	75	K	K	Note 4
4C	76	L	L	Note 4
4D	77	M	M	Note 4
4E	78	N	N	Note 4
4F	79	O	O	Note 4
50	80	P	P	Note 4
51	81	Q	Q	Note 4
52	82	R	R	Note 4
53	83	S	S	Note 4
54	84	T	T	Note 4
55	85	U	U	Note 4
56	86	V	V	Note 4
57	87	W	W	Note 4
58	88	X	X	Note 4
59	89	Y	Y	Note 4
5A	90	Z	Z	Note 4
5B	91	[	[	
5C	92	\	\	Note 4
5D	93	]	]	
5E	94	^	^	Shift
5F	95	_	_	Shift
60	96	`	`	
61	97	a	a	Note 5
62	98	b	b	Note 5
63	99	c	c	Note 5
64	100	d	d	Note 5
65	101	e	e	Note 5
66	102	f	f	Note 5

Value		As Characters		
Hex	Dec	Symbol	Keystrokes	Notes
67	103	g	g	Note 5
68	104	h	h	Note 5
69	105	i	i	Note 5
6A	106	j	j	Note 5
6B	107	k	k	Note 5
6C	108	l	l	Note 5
6D	109	m	m	Note 5
6E	110	n	n	Note 5
6F	111	o	o	Note 5
70	112	p	p	Note 5
71	113	q	q	Note 5
72	114	r	r	Note 5
73	115	s	s	Note 5
74	116	t	t	Note 5
75	117	u	u	Note 5
76	118	v	v	Note 5
77	119	w	w	Note 5
78	120	x	x	Note 5
79	121	y	y	Note 5
7A	122	z	z	Note 5
7B	123	{	{	Shift
7C	124	¦	¦	Shift
7D	125	}	}	Shift
7E	126	~	~	Shift
7F	127	⌂	Ctrl Backspace	

Value		As Characters		
Hex	Dec	Symbol	Keystrokes	Notes
80	128	Ç	Alt 128	Note 6
81	129	ü	Alt 129	Note 6
82	130	é	Alt 130	Note 6
83	131	â	Alt 131	Note 6
84	132	ä	Alt 132	Note 6
85	133	à	Alt 133	Note 6
86	134	å	Alt 134	Note 6
87	135	ç	Alt 135	Note 6
88	136	ê	Alt 136	Note 6
89	137	ë	Alt 137	Note 6
8A	138	è	Alt 138	Note 6
8B	139	ï	Alt 139	Note 6
8C	140	î	Alt 140	Note 6
8D	141	ì	Alt 141	Note 6
8E	142	Ä	Alt 142	Note 6
8F	143	Å	Alt 143	Note 6
90	144	É	Alt 144	Note 6
91	145	æ	Alt 145	Note 6
92	146	Æ	Alt 146	Note 6
93	147	ô	Alt 147	Note 6
94	148	ö	Alt 148	Note 6
95	149	ò	Alt 149	Note 6
96	150	û	Alt 150	Note 6
97	151	ù	Alt 151	Note 6
98	152	ÿ	Alt 152	Note 6
99	153	Ö	Alt 153	Note 6
9A	154	Ü	Alt 154	Note 6

Value		As Characters		
Hex	Dec	Symbol	Keystrokes	Notes
9B	155	¢	Alt 155	Note 6
9C	156	£	Alt 156	Note 6
9D	157	¥	Alt 157	Note 6
9E	158	Pt	Alt 158	Note 6
9F	159	ƒ	Alt 159	Note 6
A0	160	á	Alt 160	Note 6
A1	161	í	Alt 161	Note 6
A2	162	ó	Alt 162	Note 6
A3	163	ú	Alt 163	Note 6
A4	164	ñ	Alt 164	Note 6
A5	165	Ñ	Alt 165	Note 6
A6	166	ª	Alt 166	Note 6
A7	167	º	Alt 167	Note 6
A8	168	¿	Alt 168	Note 6
A9	169	⌐	Alt 169	Note 6
AA	170	¬	Alt 170	Note 6
AB	171	½	Alt 171	Note 6
AC	172	¼	Alt 172	Note 6
AD	173	¡	Alt 173	Note 6
AE	174	<<	Alt 174	Note 6
AF	175	>>	Alt 175	Note 6
B0	176	⣿	Alt 176	Note 6
B1	177	▒	Alt 177	Note 6
B2	178	▓	Alt 178	Note 6
B3	179	│	Alt 179	Note 6
B4	180	┤	Alt 180	Note 6
B5	181	╡	Alt 181	Note 6
B6	182	╢	Alt 182	Note 6

Value		As Characters		
Hex	Dec	Symbol	Keystrokes	Notes
B7	183	╖	Alt 183	Note 6
B8	184	╕	Alt 184	Note 6
B9	185	╣	Alt 185	Note 6
BA	186	║	Alt 186	Note 6
BB	187	╗	Alt 187	Note 6
BC	188	╝	Alt 188	Note 6
BD	189	╜	Alt 189	Note 6
BE	190	╛	Alt 190	Note 6
BF	191	┐	Alt 191	Note 6
C0	192	└	Alt 192	Note 6
C1	193	┴	Alt 193	Note 6
C2	194	┬	Alt 194	Note 6
C3	195	├	Alt 195	Note 6
C4	196	─	Alt 196	Note 6
C5	197	┼	Alt 197	Note 6
C6	198	╞	Alt 198	Note 6
C7	199	╟	Alt 199	Note 6
C8	200	╚	Alt 200	Note 6
C9	201	╔	Alt 201	Note 6
CA	202	╩	Alt 202	Note 6
CB	203	╦	Alt 203	Note 6
CC	204	╠	Alt 204	Note 6
CD	205	═	Alt 205	Note 6
CE	206	╬	Alt 206	Note 6
CF	207	╧	Alt 207	Note 6
D0	208	╨	Alt 208	Note 6

Value		As Characters		
Hex	Dec	Symbol	Keystrokes	Notes
D1	209		Alt 209	Note 6
D2	210		Alt 210	Note 6
D3	211		Alt 211	Note 6
D4	212		Alt 212	Note 6
D5	213		Alt 213	Note 6
D6	214		Alt 214	Note 6
D7	215		Alt 215	Note 6
D8	216		Alt 216	Note 6
D9	217		Alt 217	Note 6
DA	218		Alt 218	Note 6
DB	219		Alt 219	Note 6
DC	220		Alt 220	Note 6
DD	221		Alt 221	Note 6
DE	222		Alt 222	Note 6
DF	223		Alt 223	Note 6
EO	224	α	Alt 224	Note 6
E1	225	β	Alt 225	Note 6
E2	226	Γ	Alt 226	Note 6
E3	227	π	Alt 227	Note 6
E4	228	Σ	Alt 228	Note 6
E5	229	σ	Alt 229	Note 6
E6	230	μ	Alt 230	Note 6
E7	231	τ	Alt 231	Note 6
E8	232	Φ	Alt 232	Note 6
E9	233	θ	Alt 233	Note 6
EA	234	Ω	Alt 234	Note 6
EB	235	δ	Alt 235	Note 6

Value		As Characters		
Hex	Dec	Symbol	Keystrokes	Notes
EC	236	∞	Alt 236	Note 6
ED	237	φ	Alt 237	Note 6
EE	238	∈	Alt 238	Note 6
EF	239	∩	Alt 239	Note 6
F0	240	≡	Alt 240	Note 6
F1	241	±	Alt 241	Note 6
F2	242	≥	Alt 242	Note 6
F3	243	≤	Alt 243	Note 6
F4	244	⌠	Alt 244	Note 6
F5	245	⌡	Alt 245	Note 6
F6	246	÷	Alt 246	Note 6
F7	247	≈	Alt 247	Note 6
F8	248	°	Alt 248	Note 6
F9	249	∙	Alt 249	Note 6
FA	250	·	Alt 250	Note 6
FB	251	√	Alt 251	Note 6
FC	252	n	Alt 252	Note 6
FD	253	2	Alt 253	Note 6
FE	254	■	Alt 254	Note 6
FF	255	BLANK	Alt 255	Note 6

Notes

1. Asterisk (*) can be typed in the Shift state by pressing the 8 key.
2. Period (.) can be typed by pressing the . key or, in Num Lock state, by pressing the Del key on the numeric pad.
3. Numeric characters 0-9 can be typed by pressing the numeric keys on the top row of the keyboard or, in Num Lock state, by pressing the numeric keys in the keypad portion of the keyboard.
4. Uppercase alphabetical characters (A-Z) can be typed by pressing the character key in the Shift state or the Caps Lock state.
5. Lowercase alphabetical characters (a-z) can be typed by pressing the character key in the normal state or in Caps Lock and Shift state combined.
6. The three digits are typed on the numeric keypad while holding down the Alt key. Character codes 001-255 may be entered in this fashion.
7. Standard control functions are assigned to hexadecimal values as follows:

07 = Bel	0F = Shift In
08 = Backspace	11 = Device Control 1
09 = Horizontal Tab	12 = Device Control 2
0A = Line Feed	13 = Device Control 3
0B = Vertical Tab	14 = Device Control 4
0C = Form Feed	18 = Cancel
0D = Carriage Return	1B = Escape
0E = Shift Out	20 = Space

Quick Reference

Code Page 437

1 → 2 ↓	0-	1-	2-	3-	4-	5-	6-	7-	8-	9-	A-	B-	C-	D-	E-	F-
-0		▲		0	@	P	`	p	Ç	É	á	▒	└	╨	α	≡
-1	☺	▼	!	1	A	Q	a	q	ü	æ	í	▓	┴	╤	β	±
-2	☻	↕	"	2	B	R	b	r	é	Æ	ó	▓	┬	╥	Γ	≥
-3	♥	‼	#	3	C	S	c	s	â	ô	ú	▓	├	╙	π	≤
-4	♦	¶	$	4	D	T	d	t	ä	ö	ñ	┤	─	╘	Σ	⌠
-5	♣	§	%	5	E	U	e	u	à	ò	Ñ	╡	┼	╒	σ	⌡
-6	♠	▬	&	6	F	V	f	v	å	û	ª	╢	╞	╓	µ	÷
-7	•	↨	'	7	G	W	g	w	ç	ù	º	╖	╟	╫	τ	≈
-8	◘	↑	(	8	H	X	h	x	ê	ÿ	¿	╕	╚	╪	Φ	°
-9	○	↓	)	9	I	Y	i	y	ë	Ö	⌐	╣	╔	┘	Θ	∙
-A	◙	→	*	:	J	Z	j	z	è	Ü	¬	║	╩	┌	Ω	·
-B	♂	←	+	;	K	[	k	{	ï	¢	½	╗	╦	█	δ	√
-C	♀	∟	,	<	L	\	l	|	î	£	¼	╝	╠	▄	∞	ⁿ
-D	♪	↔	-	=	M	]	m	}	ì	¥	¡	╜	═	▌	φ	²
-E	♫	▲	.	>	N	^	n	~	Ä	Pt	«	╛	╬	▐	ε	■
-F	☼	▼	/	?	O	_	o	⌂	Å	ƒ	»	┐	╧	▀	∩	

1 → 2 ↓	0-	1-	2-	3-	4-	5-	6-	7-		8-	9-	A-	B-	C-	D-	E-	F-
-0		►		0	@	P	`	p		Ç	É	á	▒	└	ð	Ó	-
-1	☺	◄	!	1	A	Q	a	q		ü	æ	í	▓	┴	Ð	ß	±
-2	☻	↕	"	2	B	R	b	r		é	Æ	ó	▓	┬	Ê	Ô	=
-3	♥	!!	#	3	C	S	c	s		â	ô	ú	│	├	Ë	Ò	¾
-4	♦	¶	$	4	D	T	d	t		ã	ö	ñ	┤	─	È	õ	¶
-5	♣	§	%	5	E	U	e	u		à	ò	Ñ	Á	┼	ı	Õ	§
-6	♠	▬	&	6	F	V	f	v		å	û	ª	Â	ã	Í	µ	÷
-7	•	↨	'	7	G	W	g	w		ç	ù	º	À	Ã	Î	þ	¸
-8	◘	↑	(	8	H	X	h	x		ê	ÿ	¿	©	╚	Ï	Þ	°
-9	○	↓	)	9	I	Y	i	y		ë	Ö	®	╣	╔	┘	Ú	¨
-A	◙	→	*	:	J	Z	j	z		è	Ü	¬	║	╩	┌	Û	•
-B	♂	←	+	;	K	[	k	{		ï	ø	½	╗	╦	▉	Ù	1
-C	♀	∟	,	<	L	\	l	|		î	£	¼	╝	╠	▬	ý	3
-D	♪	↔	-	=	M	]	m	}		ì	Ø	¡	¢	═	¦	Ý	2
-E	♫	▲	.	>	N	^	n	~		Ä	×	«	¥	╬	Ì	¯	■
-F	☼	▼	/	?	O	_	o	⌂		Å	ƒ	»	┐	¤	▬	´	

Code Page 860 (Portuguese)

1 → 2 ↓	0-	1-	2-	3-	4-	5-	6-	7-		8-	9-	A-	B-	C-	D-	E-	F-
-0		▲		0	@	P	`	p		Ç	É	á	▒	└	╨	α	≡
-1	☺	▼	!	1	A	Q	a	q		ü	À	í	▒	┴	╤	β	±
-2	☻	↕	"	2	B	R	b	r		é	È	ó	▒	┬	╥	Γ	≥
-3	♥	‼	#	3	C	S	c	s		â	ô	ú	─	├	╙	π	≤
-4	♦	¶	$	4	D	T	d	t		ä	õ	ñ	┤	─	╘	Σ	⌠
-5	♣	§	%	5	E	U	e	u		à	ò	Ñ	╡	┼	╒	σ	⌡
-6	♠	▬	&	6	F	V	f	v		Á	Ú	ª	╢	╞	╓	µ	÷
-7	•	↨	'	7	G	W	g	w		ç	ù	º	╖	╟	╫	τ	≈
-8	◘	↑	(	8	H	X	h	x		ê	Ì	¿	╕	╚	╪	Φ	°
-9	○	↓	)	9	I	Y	i	y		Ê	Õ	⌐	╣	╔	┘	Θ	•
-A	◙	→	*	:	J	Z	j	z		è	Ü	¬	║	╩	┌	Ω	·
-B	♂	←	+	;	K	[	k	{		Ï	¢	½	╗	╦	█	δ	√
-C	♀	∟	,	<	L	\	l	|		Ô	£	¼	╝	╠	▄	∞	ⁿ
-D	♪	↔	-	=	M	]	m	}		ì	Ù	¡	╜	═	▌	φ	²
-E	♫	▲	.	>	N	^	n	~		Ã	Pt	«	╛	╬	▐	ε	■
-F	☼	▼	/	?	O	_	o	⌂		Â	Ó	»	┐	╧	▀	∩	

Code Page 863 (Canadian-French)

1 → 2 ↓	0-	1-	2-	3-	4-	5-	6-	7-		8-	9-	A-	B-	C-	D-	E-	F-
-0		▲		0	@	P	`	p		Ç	É	⁻	▒	└	╨	α	≡
-1	☺	▼	!	1	A	Q	a	q		ü	È	´	▓	┴	╤	β	±
-2	☻	↔	"	2	B	R	b	r		é	Ê	ó	▒	┬	╥	Γ	≥
-3	♥	‼	#	3	C	S	c	s		â	ô	ú	│	├	╙	π	≤
-4	♦	¶	$	4	D	T	d	t		Â	Ë	ñ	┤	─	╘	Σ	⌠
-5	♣	§	%	5	E	U	e	u		à	Ï	Ñ	╡	┼	╒	σ	⌡
-6	♠	▬	&	6	F	V	f	v		¶	û	ª	╢	╞	╓	µ	÷
-7	•	↨	'	7	G	W	g	w		ç	ù	º	╖	╟	╫	τ	≈
-8	◘	↑	(	8	H	X	h	x		ê	¤	¿	╕	╚	╪	Φ	°
-9	○	↓	)	9	I	Y	i	y		ë	Ô	⌐	╣	╔	┘	Θ	∙
-A	◙	→	*	:	J	Z	j	z		è	Ü	¬	║	╩	┌	Ω	·
-B	♂	←	+	;	K	[	k	{		ï	¢	½	╗	╦	█	δ	√
-C	♀	∟	,	<	L	\	l	|		î	£	¼	╝	╠	▄	∞	ⁿ
-D	♪	↔	-	=	M	]	m	}		=	Ù	¾	╜	═	▌	φ	²
-E	♫	▲	.	>	N	^	n	~		Â	Û	«	╛	╬	▐	ε	■
-F	☼	▼	/	?	O	_	o	⌂		§	ƒ	»	┐	╧	▀	∩	

Code Page 865 (Nordic)

1 → / 2 ↓	0-	1-	2-	3-	4-	5-	6-	7-		8-	9-	A-	B-	C-	D-	E-	F-
-0		▲		0	@	P	`	p		Ç	É	á	▒	└	╨	α	≡
-1	☺	▼	!	1	A	Q	a	q		ü	æ	í	▒	┴	╤	β	±
-2	☻	↔	"	2	B	R	b	r		é	Æ	ó	▒	┬	╥	Γ	≥
-3	♥	‼	#	3	C	S	c	s		â	ô	ú	▓	├	╙	π	≤
-4	♦	¶	$	4	D	T	d	t		ä	ö	ñ	│	─	╘	Σ	⌠
-5	♣	§	%	5	E	U	e	u		à	ò	Ñ	╡	┼	╒	σ	⌡
-6	♠	▬	&	6	F	V	f	v		å	û	ª	╢	╞	╓	µ	÷
-7	•	↨	'	7	G	W	g	w		ç	ù	º	╖	╟	╫	τ	≈
-8	◘	↑	(	8	H	X	h	x		ê	ÿ	¿	╕	╚	╪	Φ	°
-9	○	↓	)	9	I	Y	i	y		ë	Ö	⌐	╣	╔	┘	Θ	•
-A	◙	→	*	:	J	Z	j	z		è	Ü	¬	║	╩	┌	Ω	·
-B	♂	←	+	;	K	[	k	{		ï	ø	½	╗	╦	█	δ	√
-C	♀	∟	,	<	L	\	l	|		î	£	¼	╝	╠	▄	∞	ⁿ
-D	♪	↔	-	=	M	]	m	}		ì	Ø	¡	╜	═	▌	φ	²
-E	♫	▲	.	>	N	^	n	~		Ä	Pt	«	╛	╬	▐	ε	■
-F	☼	▶	/	?	O	_	o	⌂		Å	ƒ	¤	┐	╧	▀	∩	

Section 12. Compatibility

Introduction 12-3
System Board 12-3
Diskette Drives and Controller 12-4
Fixed Disk Drives and Controller 12-5
Application Guidelines 12-6
  Hardware Interrupts 12-6
  Software Interrupts 12-7
  High-Level Language Considerations 12-7
  Assembler Language Programming Considerations 12-7
    Opcodes 12-8
    80286 Microprocessor Anomalies 12-9
    ROM BIOS and Operating System Function Calls 12-10
    Hardware Compatibility 12-12
  Multitasking Provisions 12-13
    Interfaces 12-13
    Classes 12-14
    Time Outs 12-15
  IBM PS/1 Computer Considerations 12-16
    ROM Drive 12-16
    National Language Support 12-16
    Machine-Sensitive Programs 12-17

Introduction

This section discusses the major system differences between the IBM Personal Computer and IBM Personal System/2 product lines. Also included are programming considerations that must be taken into account when designing application programs for the IBM PS/1 computer.

System Board

The IBM PS/1 computer system board uses an 80286 microprocessor.

The 80286 system microprocessor and general architecture of the IBM PS/1 computer have created some fundamental differences between these systems and other systems. These differences must be taken into consideration when designing programs exclusively for the IBM PS/1 computer or programs compatible across the IBM Personal Computer and IBM Personal System/2 product lines. Programming considerations are discussed in "Application Guidelines" on page 12-6.

Diskette Drives and Controller

The following figure shows the read, write, and format capabilities for each type of diskette drive used by the IBM PS/1 computer.

Figure 12-1. Diskette Drive Read, Write, and Format Capabilities

Diskette Drive Type	160/180KB Mode	320/360KB Mode	720KB Mode	1.44MB Mode	1.2MB Mode
5.25-Inch 360KB Diskette Drive:					
Double-Sided	R W F	R W F	—	—	—
5.25-Inch 1.2MB Diskette Drive:					
1.2MB Drive	RWF	RWF	—	—	RWF
3.5-Inch Diskette Drive:					
1.44MB Drive	—	—	R W F	R W F	—

R-Read W-Write F-Format

Notes:

1. 5.25-inch diskette drives are supported by means of a 5.25-inch diskette drive option.
2. 5.25-inch diskettes designed for the 1.2MB mode cannot be used in either a 160/180KB or a 320/360KB diskette drive.
3. Diskettes written or formatted as 160/180KB or 320/360KB on a 5.25-inch 1.2MB drive may not work reliably when returned to a 5.25-inch 360KB diskette drive.

Warning: Sixteen-bit operations to the video subsystem can cause a diskette overrun in the 1.44MB mode because data width conversions may require more than 12 microseconds. If an overrun occurs, BIOS returns an error code and the operation should be retried.

Copy Protection

The following methods of copy protection may not work on systems using the 5.25-inch high-capacity diskette drive or the 3.5-inch 1.44MB diskette drive.

- Bypassing BIOS Routines
  - Track Density: The 5.25-inch high-capacity diskette drive records tracks at a density of 96 tracks per inch (TPI). This drive has to double-step in the 48 TPI mode that is handled by BIOS.

  - Data Transfer Rate: BIOS selects the proper data transfer rate for the media being used.
  - Diskette Parameters Table: Copy protection, which creates its own Diskette Parameters Table, may not work on these drives.

- Diskette Drive Controls
  - Rotational Speed: The time between two events on a diskette is a function of the diskette drive.
  - Access Time: Diskette BIOS routines must set the track-to-track access time for the different types of drives used in the system.
  - Diskette Change Signal: Copy protection may not be able to reset this signal.
- Write Current Control—Copy protection that uses write current control will not work because the controller selects the proper write current for the media being used.

Fixed Disk Drives and Controller

Reading from and writing to the fixed disk drive is initiated in the same way as with IBM Personal Computer products; however, new functions are supported. Detailed information about specific fixed disk drives and fixed disk adapters is available in "3.5-Inch 30MB Fixed Disk Drive and Controller," in Section 8.

Application Guidelines

Use the following information to develop application programs for the IBM PS/1 computer. Whenever possible, BIOS should be used as an interface to hardware in order to provide maximum compatibility and portability of applications across systems.

Hardware Interrupts

IBM PS/1 computer hardware interrupts are edge-triggered. On edge-sensitive interrupt systems, the interrupt controller clears its internal interrupt-in-progress latch when the interrupt routine sends an End of Interrupt (EOI) command to the controller. The EOI is sent whether the incoming interrupt request to the controller is active or inactive.

Note: Designers may want to limit the number of devices sharing an interrupt level for performance and latency considerations.

Hardware interrupt IRQ9 is defined as the replacement interrupt level for the cascade level IRQ2. Program interrupt sharing should be implemented on IRQ2, interrupt hex 0A. The following processing occurs to maintain compatibility with the IRQ2 used by IBM Personal Computer products:

1. A device drives the interrupt request active on IRQ2 of the channel.
2. This interrupt request is mapped in hardware to IRQ9 input on the second interrupt controller.
3. When the interrupt occurs, the system microprocessor passes control to the IRQ9 (interrupt hex 71) interrupt handler.
4. This interrupt handler performs an end of interrupt (EOI) to the second interrupt controller and passes control to IRQ2 (interrupt hex 0A) interrupt handler.
5. This IRQ2 interrupt handler, when handling the interrupt, causes the device to reset the interrupt request prior to performing an EOI to the master interrupt controller that finishes servicing the IRQ2 request.

Software Interrupts

With the advent of software interrupt sharing, software interrupt routines must *serially connect* interrupts. Each routine must check the function value and if it is not in the range of function calls for that routine, it must transfer control to the next routine in the chain. Because software interrupts are initially pointed to 0:0, before serially connecting it is necessary to check for this case. If the next routine is pointed to 0:0 and the function call is out of range, the appropriate action is to set the carry flag and do a RET 2 to indicate an error condition.

High-Level Language Considerations

The IBM-supported languages of IBM C, BASIC, FORTRAN, COBOL, and Pascal are the best choices for writing compatible programs.

If a program uses specific features of the hardware, that program may not be compatible with all IBM Personal Computer and IBM Personal System/2 products. Specifically, the use of assembler language subroutines or hardware-specific commands (for example, In, Out, Peek, and Poke) must follow the assembler language rules. See "Assembler Language Programming Considerations" on page 12-7.

Any program that requires precise timing information should obtain it through an operating system or language interface; for example, TIME$ in BASIC. If greater precision is required, the assembler techniques in "Assembler Language Programming Considerations" are available. The use of programming loops may prevent a program from being compatible with other IBM Personal Computer products, IBM Personal System/2 products, and software.

Assembler Language Programming Considerations

This section describes fundamental differences between the systems in the Personal Computer and Personal System/2 product lines that may affect program development.

Opcodes

The following opcodes work differently on systems using the 80286 microprocessor than they do on systems using the 8088 or 8086 microprocessor.

- PUSH SP

  The 80286 microprocessor pushes the current stack pointer; the 8088 and 8086 microprocessors push the new stack pointer, that is, the value of the stack pointer after the PUSH SP instruction is completed.

- Single-step interrupt (when TF=1) on the interrupt instruction (Opcode hex CC, CD):

  The 80286 microprocessor does not perform a single-step interrupt on the INT instruction. The 8088 and 8086 microprocessors do perform a single-step interrupt on the INT instruction.

- The divide error exception (interrupt 0):

  The 80286 microprocessor pushes the CS:IP of the instruction that caused the exception; the 8088 and 8086 microprocessors push the CS:IP of the instruction following the instruction that caused the exception.

- Shift counts for the 80286 microprocessor:

  Shift counts are masked to 5 bits. Shift counts greater than 31 are treated as mod 32. For example, a shift count of 36 shifts the operand four places.

- Multiple lockout instructions:

  There are several microprocessor instructions that, when executed, lock out external bus signals. DMA requests are not honored during the execution of these instructions. Consecutive instructions of this type prevent DMA activity from the start of the first instruction to the end of the last instruction. To allow for necessary DMA cycles, as required by the diskette controller in a multitasking system, multiple lock-out instructions must be separated by a JMP SHORT $+2.

- Consecutive I/O commands:

  Consecutive I/O commands to the same I/O ports do not permit enough recovery time for some I/O adapters. To ensure enough time, a JMP SHORT $+2 must be inserted between IN/OUT instructions to the same I/O adapters.

  Note: A MOV AL,AH type instruction does not allow enough recovery time. An example of the correct procedure follows:

```
OUT  IO_ADD,AL
JMP  SHORT $+2
MOV  AL,AH
OUT  IO_ADD,AL
```

- I/O commands followed by an STI instruction:

  I/O commands followed immediately by an STI instruction do not permit enough recovery time for some system board and channel operations. To ensure enough time, a JMP SHORT $+2 must be inserted between the I/O command and the STI instruction.

  Note: A MOV AL,AH type instruction does not allow enough recovery time. An example of the correct procedure follows:

```
OUT  IO_ADD,AL
JMP  SHORT $+2
MOV  AL,AH
STI
```

80286 Microprocessor Anomalies

In the Protected mode, when any of the null selector values (0000H, 0001H, 0002H, 0003H) are loaded into the DS or ES registers with a MOV or POP instruction or a task switch, the 80286 always loads the null selector 0000H into the corresponding register.

The following describes the operation of all 80286 microprocessor parts:

- Instructions longer than 10 bytes (instructions using multiple redundant prefixes) generate exception #13 (General Purpose Exception) in both the Real Address mode and Protected mode.

- If the second operand of an ARPL instruction is a null selector, the instruction generates an exception #13.

ROM BIOS and Operating System Function Calls

For maximum portability, programs should perform all I/O operations through operating system function calls. In environments where the operating system does not provide the necessary programming interfaces, programs should access the hardware through ROM BIOS function calls, if permissible.

- In some environments, program interrupts are used to access these functions. This practice removes the absolute addressing from the program. Only the interrupt number is required.
- In systems using the 80286 microprocessor, IRQ 9 is redirected to INT hex 0A (hardware IRQ 2). This ensures that hardware designed to use IRQ 2 will operate in these systems. See "Hardware Interrupts" on page 12-6 for more information.
- The system can mask hardware sensitivity. Device drivers can be installed to replace the ROM BIOS with the same programming interface for new devices.
- In cases where BIOS provides parameter tables, such as for video or diskette, a program can substitute new parameter values by building a new copy of the table and changing the vector to point to that table. However, the program should copy the current table, using the current vector, and then modify those locations in the table that need to be changed. In this way, the program does not inadvertently change any values that should be left the same.
- The diskette parameters table pointed to by INT hex 1E consists of 11 parameters required for diskette operation. It is recommended that the values supplied in ROM be used. If it becomes necessary to modify any of the parameters, build another parameter block and modify the address at INT hex 1E (0:78) to point to the new block.

  The parameters were established to allow the IBM PS/1 computer to operate the 3.5-inch 1.44MB diskette drive, the 5.25-inch high-capacity diskette drive (96 tracks per inch), and the 5.25-inch double-sided diskette drive (48 tracks per inch).

  The gap length parameter is not always retrieved from the parameter block. The gap length used during diskette read, write, and verify operations is derived from within diskette BIOS. The gap length for format operations is still obtained from the parameter block.

Note: Special considerations are required for format operations. Refer to the diskette section of *BIOS Interface Technical Reference for the IBM PS/1™ Computer* for the required details.

If a parameter block contains a head settle time parameter value of 0 milliseconds, and a write or format operation is being performed, the following minimum head settle times are enforced.

Figure 12-2. Write and Format Head Settle Time

Drive Type	Head Settle Time
5.25-Inch Diskette Drives:	
Double-Sided (48 TPI)	20 milliseconds
High-Capacity (96 TPI)	15 milliseconds
3.5-Inch Diskette Drives:	
1.44MB	15 milliseconds

Read and verify operations use the head settle time provided by the parameter block.

If a parameter block contains a motor start wait parameter of less than 500 milliseconds for a write or verify operation, diskette BIOS enforces a minimum time of 500 milliseconds. Read and write operations use the motor start time provided by the parameter block.

- Programs may be designed to reside on either 5.25-inch or 3.5-inch diskettes. Since not all programs are operating system dependent, the following procedure can be used to determine the type of media inserted into a diskette drive.

  1. Verify Track 0, Head 0, Sector 1 (1 sector). This allows diskette BIOS to determine if the format of the media is a recognizable type.

     If the verify operation fails, issue the reset function (AH=0) to diskette BIOS and try the operation again. If another failure occurs, the media needs to be formatted or is defective.

  2. Verify Track 0, Head 0, Sector 15 (1 sector).

     If the verify operation fails, either a 5.25-inch 360KB or 3.5-inch 720KB diskette is installed. The type can be determined by verifying Track 78, Head 1, Sector 1 (1 sector). A successful verification of Track 78 indicates a 3.5-inch 720KB diskette is installed. A verification failure indicates a 5.25-inch (48 TPI) diskette is installed.

3. Verify Track 0, Head 0, Sector 18 (1 sector). If the verify operation fails, a 5.25-inch high-capacity diskette is installed. A successful verification indicates that a 3.5-inch 1.44MB diskette is installed.

Hardware Compatibility

The IBM PS/1 computer maintains many of the interfaces used by the IBM Personal Computer AT. In most cases, command and status organization of these interfaces are maintained.

The functional interfaces for the IBM PS/1 computer are compatible with the following interfaces:

- Intel 8259 interrupt controllers (with edge triggering).
- Intel 8254 timers driven from 1.190 MHz (timer 0, 1, and 2).
- Intel 8237 DMA controller address/transfer counters, page registers and status fields only. The Command and Request registers are not supported. The rotate and mask functions are not supported. The Mode register is partially supported.
- NS16450 serial port.
- Intel 8088, 8086, and 80286 microprocessors.
- Intel 8272 diskette drive controller.
- Motorola MC146818A Real Time Clock command and status (RTC/CMOS/RAM reorganized).
- Intel 8042 keyboard port at address hex 0060 and 0064.

  Note: Use the new interface described in "System Control Port A (Hex 0092)" on page 3-167 to change the status of the A20 address line. Use the model and submodel bytes to determine that the program is running on an IBM PS/1 computer.

- Display modes supported by the IBM Monochrome Display and Printer Adapter, IBM Color/Graphics Monitor Adapter, and the IBM Enhanced Graphics Display Adapter.
- Parallel printer ports (Parallel 1, Parallel 2, and Parallel 3) in Compatibility mode.

Multitasking Provisions

The BIOS contains a feature to assist multitasking implementation. Hooks are provided for a multitasking dispatcher. Whenever a busy (wait) loop occurs in the BIOS, a hook is provided for the program to break out of the loop. Also, whenever BIOS services an interrupt, a corresponding wait loop exits and another hook is provided. Thus a program can be written that employs most of the device driver code. The following is valid only in the Real Address mode and must be taken by the code to allow this support:

- The program is responsible for the serialization of access to the device driver. The BIOS code is not reentrant.
- The program is responsible for matching corresponding Wait and Post calls.

Warning: Sixteen-bit operations to the video subsystem can cause a diskette overrun in the 1.44MB mode because data width conversions may require more than 12 microseconds. If an overrun occurs, BIOS returns an error code and the operation should be retried.

Interfaces

There are four interfaces used by the multitasking dispatcher:

Startup: First, the startup code hooks INT 15H. The dispatcher is responsible to check for function codes of (AH)= hex 90 or 91. The Wait and Post sections describe these codes. The dispatcher must pass all other functions to the previous user of INT 15H. This can be done by a JMP or a CALL. If the function code is hex 90 or 91, the dispatcher should do the appropriate processing and return by the IRET instruction.

Serialization: It is up to the multitasking system to ensure that the device driver code is used serially. Multiple entries into the code can result in serious errors.

Wait: When the BIOS is about to enter a busy loop, it first issues an INT 15H with a function code of hex 90 in AH. This signals a wait condition. At this point, the dispatcher should save the task status and dispatch another task. This allows overlapped execution of tasks when the hardware is busy. The following is an outline of the code that has been added to the BIOS to perform this function.

```
MOV AX, 90XXH          ; wait code in AH and
                       ; type code in AL
INT 15H                ; issue call
JC  TIMEOUT            ; optional: for time-out or
                       ; if carry is set, time-out
                       ; occurred
NORMAL TIMEOUT LOGIC ; normal time-out
```

Post: Whenever the BIOS has set an interrupt flag for a corresponding busy loop, an INT 15H occurs with a function code of hex 91 in AH. This signals a post condition. At this point, the dispatcher should set the task status to *ready to run* and return to the interrupt routine. The following is an outline of the code added to BIOS that performs this function.

```
MOV AX, 91XXH          ; post code AH and
                       ; type code AL
INT 15H                ; issue call
```

Classes

The following types of wait loops are supported:

- The class for hex 0 to 7F is serially reusable. This means that for the devices that use these codes, access to the BIOS must be restricted to only one task at a time.
- The class for hex 80 to BF is reentrant. There is no restriction on the number of tasks that can access the device.
- The class for hex C0 to FF is noninterrupt. There is no corresponding interrupt for the wait loop. Therefore, it is the responsibility of the dispatcher to determine what satisfies this condition to exit from the loop.

Function Code Classes

Type Code (AL)	Description
00H->7FH	Serially reusable devices; the operating system must serialize access.
80H->0BFH	Reentrant devices; ES:BX is used to distinguish different calls (multiple I/O calls are allowed simultaneously).

0C0H->0FFH Wait-only calls; there is no complementary post for these waits--these are time out only. Times are function-number dependent.

Function Code Assignments: The following are specific assignments for the IBM PS/1 computer BIOS. Times are approximate. They are grouped according to the classes described under "Function Code Classes."

Figure 12-3. Functional Code Assignments

Type Code (AL)	Time Out	Description
00H	Yes (12 seconds)	Fixed Disk
01H	Yes (2 seconds)	Diskette
02H	No	Keyboard
0FCH	Yes (20 seconds)	Fixed Disk Reset
0FDH	Yes (500-ms Read/Write)	Diskette Motor Start
0FEH	Yes (20 seconds)	Printer

The asynchronous support has been omitted. The serial and parallel controllers generate interrupts, but BIOS does not support them in the interrupt mode. Therefore, the support should be included in the multitasking system code if that device is to be supported.

Time Outs

To support time outs properly, the multitasking dispatcher must be aware of time. If a device enters a busy loop, generally it should remain there for a specific amount of time before indicating an error. The dispatcher should return to the BIOS wait loop with the carry bit set if a time out occurs.

IBM PS/1 Computer Considerations

ROM Drive

The IBM PS/1 computer contains IBM DOS 4.01 in ROM. This appears to the user as an extra drive (C: on a single diskette system and D: on a fixed disk system). This ROM drive contains the DOS kernel files (IBMBIO and IBMDOS), COMMAND.COM, an AUTOEXEC.BAT, a CONFIG.SYS, ROMSHELL.COM (which implements the four quadrant screen), and various support programs. The ROM drive is implemented as an installable file system by integrating a version of IFSFUNC into IBMBIO. Normally, IFSFUNC is loaded to run a Local Area Network (LAN). Consequently, IFSFUNC cannot be loaded when the system is booted from ROM which prohibits running a network. In fact, the ROM drive appears to be a network drive to the rest of the system.

A CUSTOMIZ program is included with the software allowing the user to customize the way the system starts. When changing items on the CUSTOMIZ menu, battery-backed CMOS RAM bits are modified. When the system is started, the BIOS looks at these customization bits to determine whether to boot from the ROM drive, the diskette drive, or the fixed disk.

When booting from a diskette or fixed disk, the ROM drive does not exist. When the ROM drive is booted, IBMBIO and COMMAND.COM look at the customization bits to determine whether to read the CONFIG.SYS and AUTOEXEC.BAT from ROM, from diskette, or from fixed disk. After the AUTOEXEC.BAT is processed, COMMAND.COM tries to load ROMSHELL.COM if SHELLSTB.COM has been installed in the CONFIG.SYS. Otherwise, the DOS prompt is displayed.

From the ROMSHELL, the user can get to the DOS prompt by pressing **Shift + F9**. In addition, pressing **Alt + SysRq** from the ROMSHELL causes a conventional PC boot from the diskette or fixed disk, regardless of the state of the CUSTOMIZ bits. This is useful when a "one-time" boot is required.

National Language Support

National Language Support (NLS) is implemented by having a unique ROM for each country. In addition to the files mentioned above, the ROM drive contains NLSFUNC.EXE, MODE.COM, KEYB.COM, KEYBOARD.SYS, COUNTRY.SYS, DISPLAY.SYS, and EGA.CPI. The AUTOEXEC.BAT and CONFIG.SYS contain the necessary statements to load the appropriate code page and keyboard for a given country.

Machine-Sensitive Programs

Programs can select machine-specific features, but they must first identify the machine and model type. IBM has defined methods for uniquely determining the specific machine type. The location of the machine model bytes can be found through INT 15H function code (AH) = hex C0. The model byte for the IBM PS/1 computer is shown in the following figure.

Figure 12-4. Machine Model Byte

Model Byte	Sub-Model Byte	Product Name
FC	0B	IBM PS/1 computer

See *IBM Personal System/2 and Personal Computer BIOS Interface Technical Reference* for a listing of model bytes for other IBM products.

Appendix. Options

512KB Memory Expansion A-3
Description A-3
Connector A-3
Audio Card & Joystick A-5
Function Description A-5
Joystick A-5
Digital-to-Analog Converter (DAC) A-6
Analog-to-Digital Converter (ADC) A-6
FIFO Timer A-7
Interrupt Generator A-7
Sound Generator A-7
Serial MIDI A-8
Connector Descriptions A-8
Audio Card Connector A-8
Microphone Connector (3/32 Inch [2.5mm] Phone Plug) . A-9
MIDI Connector (6 Pin Mini-DIN Plug) A-10
MIDI IN, OUT, THRU/OUT Connectors (5 Pin DIN Plug) . . . A-10
Joystick Connector (15 Pin Female D-Shell) A-11
Joystick Schematic Diagram A-12
I/O Address Assignments A-13
Software Registers A-14
Signal Timings A-24
Audio Subsystem A-26
Audio Module Block Diagram A-27
Sound Generator Module Block Diagram A-28
5.25-Inch External Diskette Drive Unit A-29
Buffer Card Functional Diagram A-29
Description A-30
Specifications A-30
Buffer Card Connector Pin-Outs A-31
5.25-Inch Double-Sided Diskette Drive A-32
Description A-32
Interfaces A-33
Specifications A-34
5.25-Inch High Capacity Diskette Drive A-35
Description A-35
Interfaces A-36
Specifications A-37
Adapter Card Unit A-38
Description A-38
Power Supply A-38
Specifications A-40

512KB Memory Expansion

Description

The 512KB memory expansion card provides 512KB of additional memory on the system board. When added to the system board, the card provides a total of 640KB of conventional memory below the 1MB address space and 384KB of extended memory above 1MB address space.

The card contains four 256K x 4 dynamic memory chips organized as 256K x 16 words. Memory speed is 120 nanoseconds. Card size is 2.4 inches x 1.55 inches (61.0 mm x 39.4 mm).

Connector

The card has 40 gold edge tabs, 20 on the front and 20 on the back for connection to the system board memory expansion connector.

Front Pin	Signal Name	Back Pin	Signal Name
1	MD0	40	MA0
2	MD1	39	MA1
3	MD2	38	MA2
4	MD3	37	MA3
5	Ground	36	MA4
6	MD4	35	Ground
7	MD5	34	MA5
8	MD6	33	MA6
9	MD7	32	MA7
10	Ground	31	MA8
11	MD8	30	Reserved
12	MD9	29	Ground
13	MD10	28	-CASL
14	MD11	27	-CASH
15	Ground	26	-WE
16	MD12	25	Reserved
17	MD13	24	-RAS1
18	MD14	23	Reserved
19	MD15	22	Ground
20	Ground	21	+5 V

Figure A-1. 512KB Memory Expansion Signal Names

Signal Name	Definition
-CASH	Column Address Strobe (High Byte)
-CASL	Column Address Strobe (Low Byte)
MA0-MA8	Multiplexed Address Inputs
MD0-MD15	Data In and Data Out
-RAS1	Row Address Strobe
-WE	Write Enable

Audio Card & Joystick

Function Description

The Audio card has several functions:

- Joystick Interface
- 8 Bit Digital-to-Analog Converter (DAC)
- 8 Bit Analog-to-Digital Converter (ADC)
- Sound Generator
- Serial Musical Instrument Digital Interface (MIDI).

All functions share a common interrupt (Level 7). Two Application Specific Integrated Circuits (ASIC) are used to provide some address decoding and interface to several functions.

Joystick

The Joystick interface is compatible with that used in the original IBM PC Game Control Adapter. This adapter allows two joysticks with buttons to be used as a game interface.

The position of each joystick axis is measured by triggering a pulse generator. Four pulse generators are triggered by writing to IO port 201H. After triggering a pulse, software determines the length of the pulse by reading port 201H.

In addition to pulse generators, there are four digital inputs. These are used to read the buttons on the joysticks. When the buttons are pressed, the digital inputs read as 0.

By setting a bit in the control register of the ASICs (Bit 6 of port 202H), the joystick is in the auto trigger mode. This mode automatically re-triggers the pulse generators when all pulse inputs have gone to their inactive (low) state. The value of the pulse width is timed in hardware and yields a value of 0 to 255. The axis of each joystick is simultaneously timed. The axes are latched and read at any time in ports 204H, 205H, 206H, and 207H. This hardware timing of the pulses occurs only in the automatic mode of the joystick. If for any reason the pulse is longer than 255 microseconds, the returned value will be 0FFH (255). The conversion result is latched and read until a new pulse width conversion is completed.

Digital-to-Analog Converter (DAC)

The digital-to-analog converter has a 2K byte, first-in-first-out (FIFO) register. A byte is shifted from the FIFO to the DAC when the FIFO timer reloads. Data is written to the FIFO via port 200H. When the FIFO timer reloads, a byte of data is read from the FIFO and latched onto the Data Out (DO) bus. A flag indicates almost empty status for the FIFO. This flag is programmable from 0 to 1K by writing to port 204H. The 8-bit value is multiplied by 4 and added to the read address of the FIFO controller. If this matches the write address of the FIFO controller, an almost empty flag is set to 1. This flag is only valid when this condition exists. If another read or write occurs, the flag is cleared. This flag generates an internal interrupt or, if enabled, is used to generate an external off-chip interrupt. Two additional flags indicate FIFO full and FIFO empty. If the FIFO is full, any additional attempted writing of data results in lost data. Both full and empty bits may be read via the status register (202H).

Data from the FIFO is sent to an external DAC and generates audio. This output is filtered and AC coupled to other system signals before being sent to the display power amplifier.

Analog-to-Digital Converter (ADC)

The analog-to-digital converter is used to digitize analog voltages. The A/D converter converts a voltage to a digital value provided the input voltage is between ± 380 millivolts and is changing at less than 7.42 millivolts per microsecond (mV/µS). Any input greater than ± 380 millivolts is clipped. Any signal changing faster than 7.42 mV/µS is rate limited to 7.42 mV/µS. The ADC is formed using a DAC, comparator, and up/down counter clocked at 2.5 MHz. This results in a digital low pass filter with a cut-off at 4.88 KHz. Higher frequency signals pass if they are of lower amplitude.

The value of the up/down counter is latched on reload of the FIFO timer. This allows the conversion rate to be set by writing to the FIFO reload register. This latched value is read at port 200H.

FIFO Timer

The FIFO Timer is clocked at 1 MHz and is used to control the rate of data for both audio in and out. The value of the timer is written to port 203H and is read via port 203H. If the timer counts up from 0 and is reset to 0, then the count value is equal to the reload value. A pulse is generated on overflow and is used to latch data into the ADC latch and to read data out of the FIFO. The time between reloads is one cycle longer than the value written to the reload register. A reload value of 0 stops the FIFO timer and holds it in the 0 state. The maximum time is 256 microsecond. The reload register is not affected by a reset. It must be initialized at POR.

Interrupt Generator

An interrupt can be generated to allow interrupt-driven operation of the audio subsystem. This interrupt is disabled after reset.

Joystick interrupts are enabled by setting bit 3 of the control register to 1. When enabled, an interrupt is generated after each conversion of the joysticks. If enabled, the interrupt can be watched via bit 5 of the status register (202H). The interrupt can be reset by setting bit 3 back to 0. This interrupt generates an external interrupt only if the external interrupt enable bit is set (bit 0 of control 202H).

An interrupt from the almost empty flag is available if bit 1 of the control register is set. This interrupt status is read in bit 1 of the status register.

An interrupt is generated at all times from the FIFO overflow bit to indicate ADC data is ready. Its status is available on bit 4 of the status register. Any read of ADC data (200H) will reset this interrupt.

The external interrupt enable bit is used to allow an external interrupt to be generated from any three internal interrupt sources. If the external interrupt is enabled, software must read the status register to determine the cause of the interrupt.

Sound Generator

The sound generator can generate three voices plus noise. This part is write only. Port 205H is used to write to the sound generator.

Serial MIDI

The Musical Instrument Digital Interface (MIDI) is used to transmit and receive MIDI information. The MIDI uses a serial current loop to send and receive binary data. The data stream is 8 bits, no parity, one stop and one start bit. This 10-bit stream is processed at 31.25K baud. This rate results in a bit time of 32 microsecond per bit and 320 microsecond per byte. This interface is hardware compatible with the *MIDI 1.0 Specification.*

Connector Descriptions

Audio Card Connector

The audio card has a single 2-by-17-pin connector for attachment to the system board.

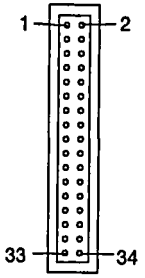


Figure A-2 on page A-9 shows the voltages and signals assigned to the audio card connector.

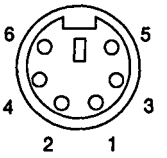
Figure A-2. Audio Card to System Board Voltage and Signal Assignments

Pin No.	I/O	Signal Name	Pin No.	I/O	Signal Name
1	O	IRQ 7	2	NA	Ground
3	NA	Ground	4	I	CLK
5	I	SA9	6	I	SA7
7	NA	Ground	8	I	SA8
9	I	-IOR	10	I	SA6
11	I	-IOW	12	I	SA5
13	I	SA4	14	I	AEN
15	I	SA3	16	NA	Ground
17	I	SA2	18	I	SA1
19	I	SA0	20	I	+12 Volts
21	I/O	SD0	22	I	-12 Volts
23	I/O	SD2	24	I/O	SD1
25	I/O	SD3	26	I	+5 Volts
27	I/O	SD4	28	O	MODOUT
29	I/O	SD5	30	I	RESET DRV
31	I/O	SD6	32	I/O	SD7
33	O	AUDOUT	34	I	MODIN

Microphone Connector (3/32 Inch [2.5mm] Phone Plug)

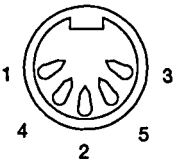
Center contact - Microphone
Outside contact - Ground

MIDI Connector (6 Pin Mini-DIN Plug)



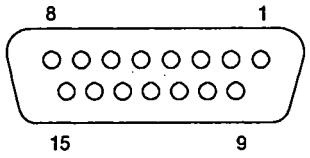
Pin	Signal Name/Function
1	MIDI THRU/OUT +
2	MIDI THRU/OUT –
3	MIDI OUT –
4	MIDI IN +
5	MIDI OUT +
6	MIDI IN –

MIDI IN, OUT, THRU/OUT Connectors (5 Pin DIN Plug)



Pin	Signal Name/Function
1	No Connect
2	Shield
3	No Connect
4	MIDI +
5	MIDI –

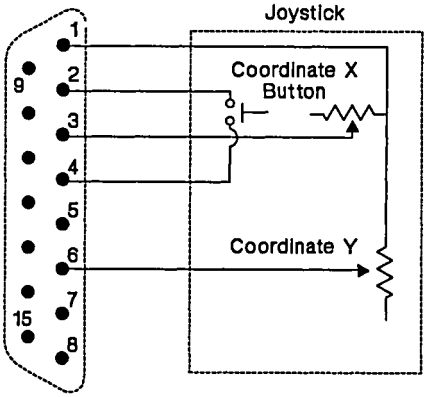
Joystick Connector (15 Pin Female D-Shell)



Pin	Signal Name/Function
1	+ 5 Volts
2	Button 4 (Stick A Button 1)
3	Position 0 (Stick A Coordinate X)
4	Ground
5	Ground
6	Position 1 (Stick A Coordinate Y)
7	Button 5 (Stick A Button 2)
8	+ 5 Volts
9	+ 5 Volts
10	Button 6 (Stick B Button 1)
11	Position 2 (Stick B Coordinate X)
12	Ground
13	Position 3 (Stick B Coordinate Y)
14	Button 7 (Stick B Button 2)
15	+ 5 Volts

Joystick Schematic Diagram

15-PIN MALE D-SHELL CONNECTOR



Note: Potentiometer for Coordinates X and Y has a range of 0 to 145 k-ohms. Button is normally open; closed when pressed.

I/O Address Assignments

I/O Address (hex)	Read Function
200	Read Analog to Digital Converter Data
201	Read Joystick and buttons
202	Read Control Register
203	Read FIFO Timer reload value
204	Joystick (X Axis Stick A) P0
205	Joystick (Y Axis Stick A) P1
206	Joystick (X Axis Stick B) P2
207	Joystick (Y Axis Stick B) P3
330	Read to MIDI TXD Register
331	Read to MIDI IER Register
332	Read to MIDI IIR Register
335	Read to MIDI LSR Register
	Write Function
200	Write to Digital to Analog Converter
201	Starts Joystick conversions
202	Write to Control Register
203	Write FIFO Timer reload value
204	Write almost empty value
205	Write to Sound Generator
330	Write to MIDI TXD Register
331	Write to MIDI IER Register
332	Write to MIDI IIR Register
335	Write to MIDI LSR Register

Software Registers

I/O Address 200H

MSB LSB

D7	D6	D5	D4	D3	D2	D1	D0

Bit	Read	Write
D7 – 7 MSB	ADC Input	DAC Output
D6 – 6	ADC Input	DAC Output
D5 – 5	ADC Input	DAC Output
D4 – 4	ADC Input	DAC Output
D3 – 3	ADC Input	DAC Output
D2 – 2	ADC Input	DAC Output
D1 – 1	ADC Input	DAC Output
D0 – 0 LSB	ADC Input	DAC Output

A read of port 200H reads the data latch for the input ADC. This data was latched on the previous overflow of the FIFO timer. This read also clears the ADC Data Ready bit.

A write to port 200H writes data to the FIFO for the output DAC. If the FIFO is full, this data is lost. If the FIFO is empty, the data is output to the DAC on the next overflow of the FIFO timer.

I/O Address 201H

MSB LSB

D7	D6	D5	D4	D3	D2	D1	D0

Bit	Read	Write
D7 – 7 MSB	Button 7 (B2)	See Note
D6 – 6	Button 6 (B1)	See Note
D5 – 5	Button 5 (A2)	See Note
D4 – 4	Button 4 (A1)	See Note
D3 – 3	Stick 3 (BY)	See Note
D2 – 2	Stick 2 (BX)	See Note
D1 – 1	Stick 1 (AY)	See Note
D0 – 0 LSB	Stick 0 (AX)	See Note

Note: A write to port 201 causes all stick inputs to go high for a value specified by the equation TIME = 24.2 microsecond + 0.011(r) microsecond. (Manual mode only.) Button inputs are not affected.

Port 201 emulates the existing IBM PC Game Control Adapter. A write to this port causes the pulse generators to be triggered.

Button pins are TTL level non-inverting inputs.

When in the auto joystick mode, the pulse generators are continuously triggered.

I/O Address 202H

MSB LSB

RIO	JM	JIE	ADR	FF	FE	IR	AIE

Bit	Read	Write
RIO – 7 MSB	RIN1 Bit	ROUT Bit
JM – 6	RIN0 Bit	0 – Manual Joystick 1 – Auto Mode
JIE – 5	Joystick Int	0 – Joystick Int Disabled 1 – Enabled
ADR – 4	ADC Data Rdy	Reserved
FF – 3	FIFO Full	Reserved
FE – 2	FIFO Empty	0 – Modem Audio Disabled 1 – Enabled
IR – 1	Almost Empty Int	0 – AE Int Disabled 1 – Enabled
AIE – 0 LSB	Ext Int Enable	0 – Ext Int Disabled 1 – Enabled

Note: Default value for this port is 00H after a RESET.

Bit 7, the RIN0, RIN1, and ROUT bits, are reserved.

Bit 6 is used to select the manual or auto trigger mode. A read will read the value of the RIN0 Bit.

Bit 5 is used to enable interrupts from the joystick.

Bit 4 is read only and is used to indicate that ADC data has been latched due to FIFO timer overflow. A read of the data from port 200H resets this bit.

Bit 3 is read only and is used to indicate that the FIFO is full. Any additional write to port 200H results in lost data.

Bit 2 is used to indicate that the FIFO is empty. A write to this bit controls the modem audio enable bit.

Bit 1 is used to allow interrupts from the almost empty flag. The interrupt flag is cleared by writing a 0 then a 1 to this bit.

Bit 0 is the External interrupt control bit. A 1 enables interrupt generation from any of the three internal interrupt sources.

I/O Address 203H

MSB LSB

T7	T6	T5	T4	T3	T2	T1	T0

Bit	Read	Write
T7 – 7 MSB	FIFO Timer	Timer Reload Value
T6 – 6	FIFO Timer	Timer Reload Value
T5 – 5	FIFO Timer	Timer Reload Value
T4 – 4	FIFO Timer	Timer Reload Value
T3 – 3	FIFO Timer	Timer Reload Value
T2 – 2	FIFO Timer	Timer Reload Value
T1 – 1	FIFO Timer	Timer Reload Value
T0 – 0 LSB	FIFO Timer	Timer Reload Value

Note: Timer clocked at a 1-microsecond rate.

This port controls the timer reload value for the FIFO timer. This timer is used for both ADC and DAC timing.

I/O Address 204H

MSB LSB

P7	P6	P5	P4	P3	P2	P1	P0

Bit	Read	Write
P7 – 7 MSB	Stick 0 Position	Almost empty Bit 9
P6 – 6	Stick 0 Position	Almost empty Bit 8
P5 – 5	Stick 0 Position	Almost empty Bit 7
P4 – 4	Stick 0 Position	Almost empty Bit 6
P3 – 3	Stick 0 Position	Almost empty Bit 5
P2 – 2	Stick 0 Position	Almost empty Bit 4
P1 – 1	Stick 0 Position	Almost empty Bit 3
P0 – 0 LSB	Stick 0 Position	Almost empty Bit 2

This port is used to read the X position of Stick A. A write to this port controls the value of the almost empty register. A value of 0 generates an almost empty when there are 0 bytes left, a 1 generates an almost empty when 4 bytes are left, and so on.

I/O Address 205H (Write)

MSB LSB

0	0	F9	F8	F7	F6	F5	F4

This is the form for the second byte of all voice generators. After sending the byte to select the register and low 4 bits of the voice, the second byte is sent to adjust the high 6 bits of the 10 voice register.

MSB LSB

1	0	0	0	F3	F2	F1	F0

This is the first byte of data to select Voice Register 1 of the Sound Generator. This selects Voice 1 and sets the low 4 bits of the Voice 1 frequency. The frequency is determined by 125 KHz/F, where F is 0 to 1023.

MSB LSB

1	0	0	1	A3	A2	A1	A0

```
A3  A2  A1  A0

 0   0   0   1 -  2 dB Attenuation
 0   0   1   0 -  4 dB Attenuation
 0   1   0   0 -  8 dB Attenuation
 1   0   0   0 - 16 dB Attenuation
 1   1   1   1 - OFF - No Output
```

This controls the attenuation of Voice 1.

MSB LSB

1	0	1	0	F3	F2	F1	F0

This is the first byte of data to select Voice Register 2 of the Sound Generator. This selects Voice 2 and sets the low 4 bits of the Voice 2 frequency. The frequency is determined by 125 KHz/F, where F is 0 to 1023. The second byte determines the other 6 bits of the voice register.

MSB LSB

1	0	1	1	A3	A2	A1	A0

A3 A2 A1 A0

0 0 0 1 - 2 dB Attenuation
0 0 1 0 - 4 dB Attenuation
0 1 0 0 - 8 dB Attenuation
1 0 0 0 - 16 dB Attenuation
1 1 1 1 - OFF - No Output

This controls the attenuation of Voice 2.

MSB LSB

1	1	0	0	F3	F2	F1	F0

This is the first byte of data to select Voice Register 3 of the Sound Generator. This selects Voice 3 and sets the low 4 bits of the Voice 3 frequency. The frequency is determined by 125 KHz/F, where F is 0 to 1023. The second byte determines the other 6 bits of the voice register.

MSB LSB

1	1	0	1	A3	A2	A1	A0

A3 A2 A1 A0

0 0 0 1 - 2 dB Attenuation
0 0 1 0 - 4 dB Attenuation
0 1 0 0 - 8 dB Attenuation
1 0 0 0 - 16 dB Attenuation
1 1 1 1 - OFF - No Output

This controls the attenuation of Voice 3.

MSB LSB

1	1	1	0	0	FB	N0	N1

FB Noise Type

0 - Periodic Noise
1 - White Noise

N0	N1	Shift Rate
0	0	N/1024
0	1	N/2048
1	0	N/4096
1	1	Voice Generator 3 Output

This selects the shift rate and feedback for the Noise Generator.

MSB LSB

1	1	1	1	A3	A2	A1	A0

A3	A2	A1	A0	
0	0	0	1	- 2 dB Attenuation
0	0	1	0	- 4 dB Attenuation
0	1	0	0	- 8 dB Attenuation
1	0	0	0	- 16 dB Attenuation
1	1	1	1	- OFF - No Output

This controls the attenuation of the Noise Generator.

I/O Address 205H (Read)

MSB LSB

P7	P6	P5	P4	P3	P2	P1	P0

Bit	Read
P7 – 7 MSB	Stick 1 Position
P6 – 6	Stick 1 Position
P5 – 5	Stick 1 Position
P4 – 4	Stick 1 Position
P3 – 3	Stick 1 Position
P2 – 2	Stick 1 Position
P1 – 1	Stick 1 Position
P0 – 0 LSB	Stick 1 Position

This port is used to read the Y position of Stick A.

I/O Address 206H

MSB LSB

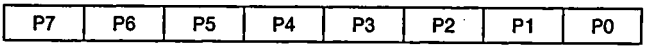
P7	P6	P5	P4	P3	P2	P1	P0

Bit	Read	Write
P7 – 7 MSB	Stick 2 Position	Reserved
P6 – 6	Stick 2 Position	Reserved
P5 – 5	Stick 2 Position	Reserved
P4 – 4	Stick 2 Position	Reserved
P3 – 3	Stick 2 Position	Reserved
P2 – 2	Stick 2 Position	Reserved
P1 – 1	Stick 2 Position	Reserved
P0 – 0 LSB	Stick 2 Position	Reserved

This port is used to read the X position of Stick B.

I/O Address 207H

MSB LSB

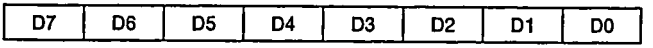


Bit	Read	Write
P7 – 7 MSB	Stick 3 Position	Reserved
P6 – 6	Stick 3 Position	Reserved
P5 – 5	Stick 3 Position	Reserved
P4 – 4	Stick 3 Position	Reserved
P3 – 3	Stick 3 Position	Reserved
P2 – 2	Stick 3 Position	Reserved
P1 – 1	Stick 3 Position	Reserved
P0 – 0 LSB	Stick 3 Position	Reserved

This port is used to read the Y position of Stick B.

I/O Address 330H

MSB LSB



Bit	Read	Write
D7 – 7 MSB	Rx Data Bit 7	Tx Data Bit 7
D6 – 6	Rx Data Bit 6	Tx Data Bit 6
D5 – 5	Rx Data Bit 5	Tx Data Bit 5
D4 – 4	Rx Data Bit 4	Tx Data Bit 4
D3 – 3	Rx Data Bit 3	Tx Data Bit 3
D2 – 2	Rx Data Bit 2	Tx Data Bit 2
D1 – 1	Rx Data Bit 1	Tx Data Bit 1
D0 – 0 LSB	Rx Data Bit 0	Tx Data Bit 0

This port is used to read and write MIDI data.

I/O Address 331H

MSB LSB

D7	D6	D5	D4	D3	D2	D1	D0

Bit	Read/Write
D7 – 7 MSB	Reserved
D6 – 6	Reserved
D5 – 5	Reserved
D4 – 4	0 – MIDI Out 1 – MIDI Thru
D3 – 3	Reserved
D2 – 2	Reserved
D1 – 1	Tx Data Interrupt
D0 – 0 LSB	Rx Data Interrupt

This port is used to control the interrupt enables for the MIDI. Interrupts will be routed to IRQ 7.

I/O Address 332H

MSB LSB

D7	D6	D5	D4	D3	D2	D1	D0

Bit	Read Only
D7 – 7 MSB	Reserved
D6 – 6	Reserved
D5 – 5	Reserved
D4 – 4	Reserved
D3 – 3	Reserved
D2 – 2	Reserved
D1 – 1	Reserved
D0 – 0 LSB	Interrupt Pending

This port is used to determine interrupt status.

I/O Address 335H

MSB LSB

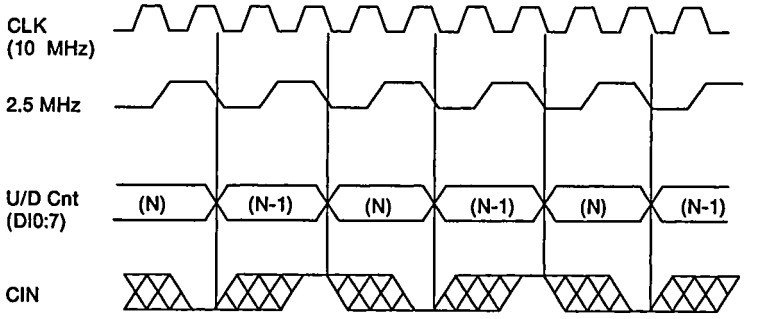
D7	D6	D5	D4	D3	D2	D1	D0

Bit	Read Only
D7 – 7 MSB	Reserved
D6 – 6	Reserved
D5 – 5	TxD Empty
D4 – 4	Reserved
D3 – 3	Framing Error
D2 – 2	Reserved
D1 – 1	Overrun Error
D0 – 0 LSB	Rx Data Available

This port is used to determine the condition of the transmit and receive data buffers.

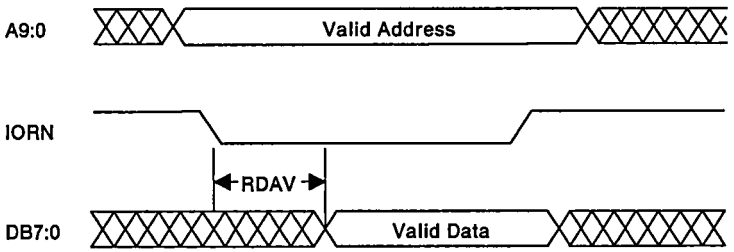
Signal Timings

Input ADC Timing (Steady State)



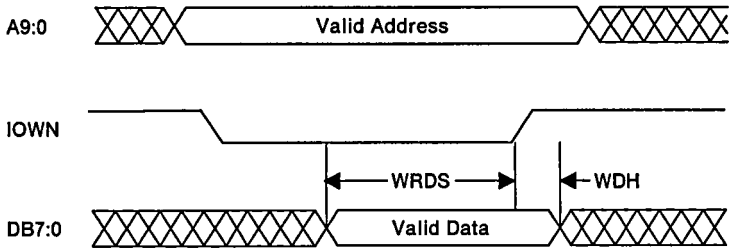
Note: CIN sampled prior to the up/down counter changes.

Bus Read Timing



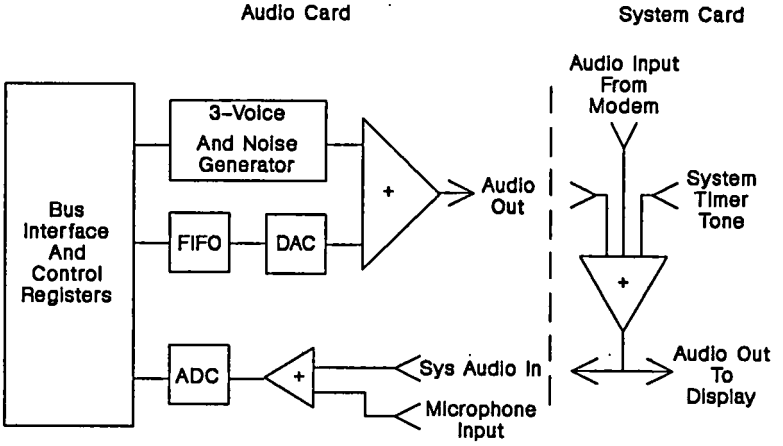
Timing		Min	Max (ns)
RDAV	RD active to valid read data		80

Bus Write Timing

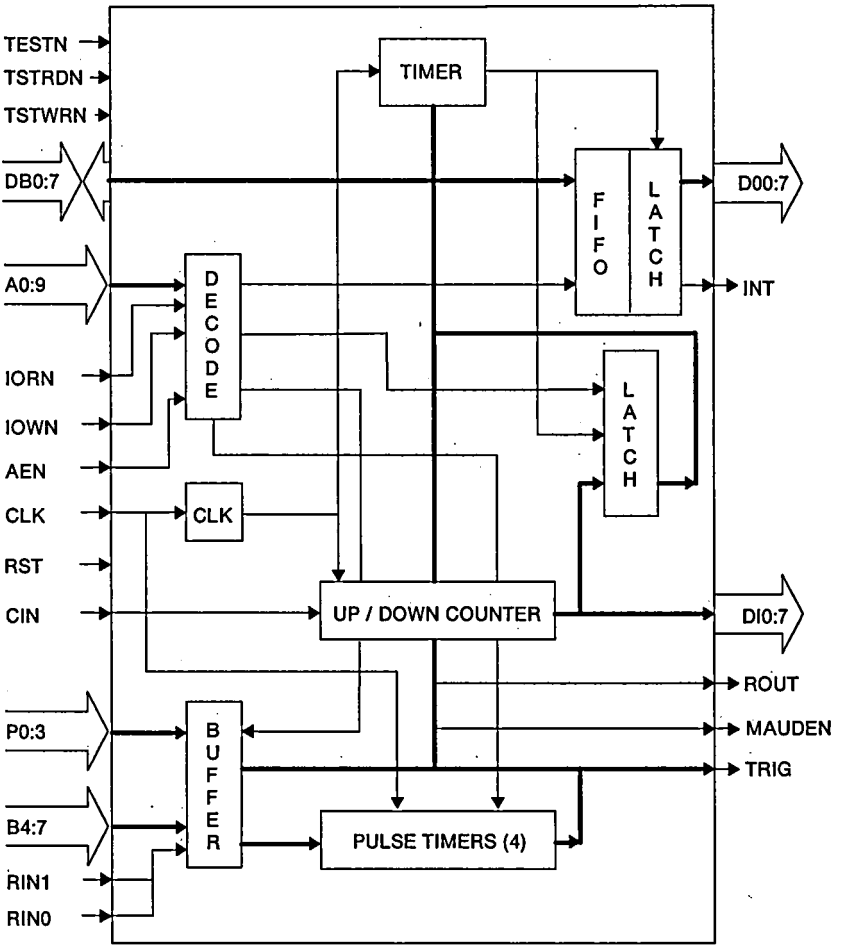


Timing		Min (ns)	Max (ns)
WRDS	Data setup prior to WR active	100	
WDH	Data hold after Write		15

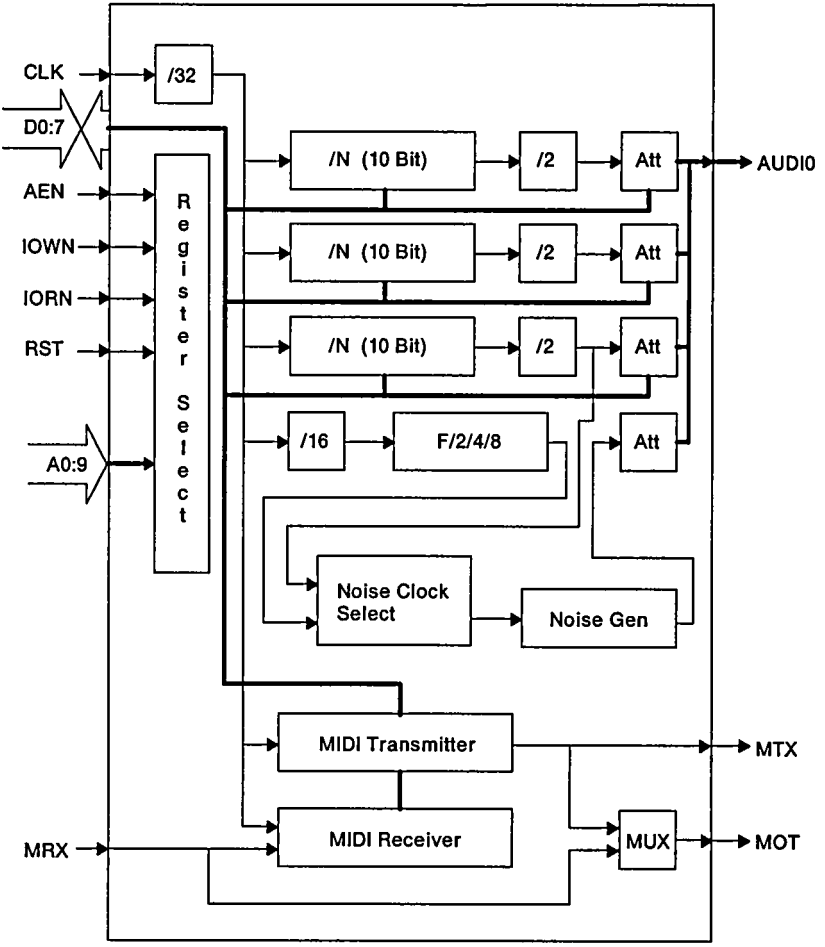
Audio Subsystem



Audio Module Block Diagram



Sound Generator Module Block Diagram



5.25-Inch External Diskette Drive Unit

The 5.25-inch External Diskette Drive Unit attaches to the bottom of the IBM PS/1 computer system unit. The 5.25-inch external diskette drive unit contains a logic card, a 5.25-inch diskette drive and the necessary cables to electrically connect the unit to the system unit.

The logic card buffers the interface signals coming from and going to the system unit. The buffer card has a flat cable attached which connects to either a 360KB Double Sided Diskette Drive or a 1.2MB High Capacity Diskette Drive, depending on the option purchased.

A thirty-four conductor flat ribbon cable connects the buffer card to the system unit. A separate four conductor cable provides the 5.25-inch external diskette drive with +5 and +12 volt power from the system unit.

Once installed, the 5.25-inch external diskette drive becomes the system drive B, which is non-bootable.

Buffer Card Functional Diagram

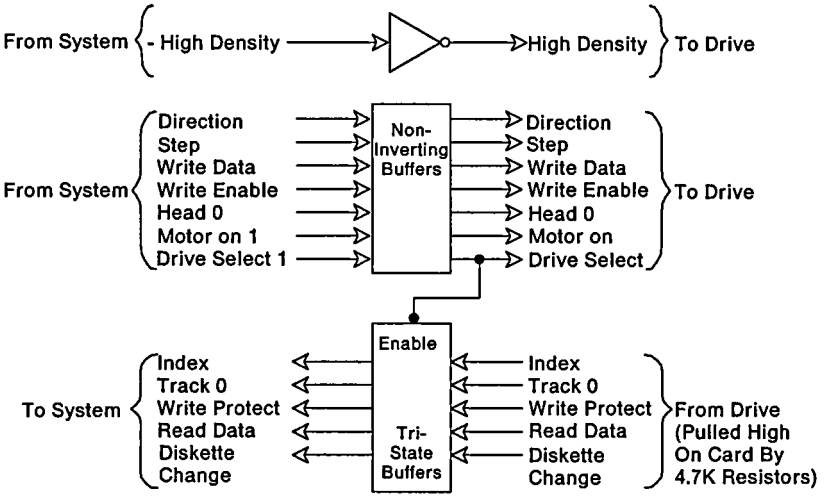


Figure A-3. Buffer Card Functional Diagram

Description

The buffer card provides high-current open-collector buffers for signals passing from the system unit to the diskette drive, inverting the high density select signal. The card also provides pull-up resistors and tri-state buffers for signals received from the drive.

A 2-by-17 pin header (J1) connects the buffer card to the system unit. A 2-by-17 card-edge connector (J2) attaches to the diskette drive.

Note: High density select has no effect when a 360KB diskette drive is present. Also, with a 360KB diskette drive, -diskette change is always high (inactive).

Specifications

The following are specifications for the 5.25-inch external diskette drive unit:

Size

- Width: 274 millimeters (10.8 inches)
- Depth: 320 millimeters (12.6 inches)
- Height: 64 millimeters (2.5 inches).

Weight

- 3.4 kilograms (7.5 pounds).

Buffer Card Connector Pin-Outs

J1			
Pin	**Signal**	**Pin**	**Signal**
1	Drive 2 installed (Gnd)	2	-High Density Select
3	+5 V	4	Reserved (N/C)
5	Ground	6	Reserved (N/C)
7	Ground	8	-Index
9	Reserved (N/C)	10	-Motor on 1
11	Ground	12	-Drive Select 0
13	Ground	14	-Drive Select 1
15	Ground	16	-Motor on 0
17	Ground	18	-Direction
19	Ground	20	-Step
21	Ground	22	-Write Data
23	Ground	24	-Write Enable
25	Ground	26	-Track 0
27	Ground	28	-Write Protect
29	Ground	30	-Read Data
31	Ground	32	Head 0
33	Ground	34	-Diskette Change

J2			
Pin	**Signal**	**Pin**	**Signal**
1	Ground	2	-High Density Select
3	Ground	4	Reserved (N/C)
5	Ground	6	Reserved (N/C)
7	Ground	8	-Index
9	Ground	10	Reserved (N/C)
11	Ground	12	-Drive Select
13	Ground	14	Reserved (N/C)
15	Ground	16	-Motor on
17	Ground	18	-Direction
19	Ground	20	-Step
21	Ground	22	-Write Data
23	Ground	24	-Write Enable
25	Ground	26	-Track 0
27	Ground	28	-Write Protect
29	Ground	30	-Read Data
31	Ground	32	Head 0
33	Ground	34	-Diskette Change

5.25-Inch Double-Sided Diskette Drive

Description

The IBM PS/1 computer 5.25-inch Double-Sided Diskette Drive is a direct-access device that can store 360KB of data on a dual-sided 5.25-inch diskette. All data format and access control is in the system. The following figure describes the type of diskette required by this drive.

Figure A-4. Diskette Requirements

Characteristic	Requirement
Certification	Double-sided 48 TPI (tracks per inch) 40 tracks per surface Soft Sector
Recording Density	5,876 bits per inch
Media Coercivity	300 to 350 Oersteds
Jacket	Standard 5.25-inch

Interfaces

The diskette drive has two types of interfaces: control and dc power. The following figure shows the signals and pin assignments for the control interfaces.

Figure A-5. Control Interface (P1/J1)

Signal	I/O	Signal Pin	Ground Pin
Reserved	-	2	1
Reserved	-	4	3
-Drive Select 3	I	6	5
-Index	O	8	7
-Drive Select 0	I	10	9
-Drive Select 1	I	12	11
-Drive Select 2	I	14	13
-Motor On	I	16	15
-Direction Select	I	18	17
-Step	I	20	19
-Write Data	I	22	21
-Write Gate	I	24	23
-Track 00	O	26	25
-Write Protect	O	28	27
-Read Data	O	30	29
-Side 1 Select	I	32	31
Reserved	-	34	33

All signals operate between +5 V dc and ground with the following definitions:

	Outputs	Inputs
Inactive Level:	+2.5 to +5.25 V dc	+2.5 to +5.25 V dc
Active Level:	0.0 to +0.4 V dc	0.0 to +0.8 V dc

All outputs from the drive can sink 40 mA at the active level. The system provides pull-up registers.

Following are the signals and pin assignments for the dc power interface.

Figure A-6. DC Power Interface (P2/J2)

Signal	Pin
+ 12 V dc	1
+ 12 V dc Return	2
+ 5 V dc Return	3
+ 5 V dc	4

Specifications

The following are physical specifications for the 5.25-inch double-sided diskette drive.

Size

- Width: 146 millimeters (5.8 inches)
- Depth: 203.2 millimeters (8.0 inches)
- Height: 41 millimeters (1.6 inches).

Weight

- 1.6 kilograms (3.52 pounds).

Electrical

- 11 Watts (typical).

The following are performance specifications for the 5.25-inch double-sided diskette drive:

Capacity unformatted	500KB
Capacity formatted	
9 sectors per track	360KB
Recording density	5,876 bits per inch
Track density	48 TPI (tracks per inch)
Cylinders	40
Heads	2
Encoding method	MFM
Rotational speed	300 RPM ±1.5%
Transfer rate	250K bits per second
Latency (average)	100 milliseconds
Access time:	
Average	81 milliseconds
Track to track	6 milliseconds
Settling time	15 milliseconds
Head load time	0 milliseconds
Motor start time	500 milliseconds

5.25-Inch High Capacity Diskette Drive

Description

The IBM PS/1 computer 5.25-inch High Capacity Diskette Drive is a direct-access device that can store 1.2MB of data on a dual-sided 5.25-inch diskette. All data format and access control is in the system. The following figure describes the type of high-density diskette required by this drive. Diskettes which meet these specifications may not be used in either a 160/180KB or a 320/360KB diskette drive.

Figure A-7. Diskette Requirements

Characteristic	Requirement
Certification	Double sided
	96 TPI (tracks per inch)
	80 tracks per surface
	Soft Sector
Recording Density	9,646 bits per inch
Media Coercivity	600 to 650 Oersteds
Jacket	Standard 5.25-inch

Note: This drive also can read diskettes formatted for a 320/360KB dual-sided drive or a 160/180KB single-sided drive.

Interfaces

The diskette drive has two types of interface: control and dc power. The following figure shows the signals and pin assignments for the control interfaces.

Figure A-8. Control Interface (P1/J1)

Signal Name	I/O	Signal Pin	Ground Pin
-Reduced Write	I	2	1
Reserved	-	4	3
-Drive Select 3	I	6	5
-Index	O	8	7
-Drive Select 0	I	10	9
-Drive Select 1	I	12	11
-Drive Select 2	I	14	13
-Motor On	I	16	15
-Direction Select	I	18	17
-Step	I	20	19
-Write Data	I	22	21
-Write Gate	I	24	23
-Track 00	O	26	25
-Write Protect	O	28	27
-Read Data	O	30	29
-Side 1 Select	I	32	31
-Diskette Change	O	34	33

All signals operate between +5 V dc and ground with the following definitions:

	Outputs	Inputs
Inactive Level:	+2.5 to +5.25 V dc	+2.5 to +5.25 V dc
Active Level:	0.0 to +0.4 V dc	0.0 to +0.8 V dc

All outputs from the drive can sink 40 mA at the active level. The system provides pull-up registers.

Following are the signals and pin assignments for the dc power interface.

Figure A-9. DC Power Interface (P2/J2)

Signal	Pin
+ 12 V dc	1
+ 12 V dc Return	2
+ 5 V dc Return	3
+ 5 V dc	4

Specifications

The following are physical specifications for the 5.25-inch high capacity diskette drive.

Size

- Width: 146 millimeters (5.8 inches)
- Depth: 203.2 millimeters (8.0 inches)
- Height: 41 millimeters (1.6 inches).

Weight

- 1.6 kilograms (3.52 pounds).

Electrical

- 11 Watts (typical).

The following are performance specifications for the 5.25-inch high capacity diskette drive:

Capacity unformatted	1604KB
Capacity formatted	
15 sectors per track	1.2MB
Recording density	9,646 bits per inch
Track density	96 TPI (tracks per inch)
Cylinders	80
Heads	2
Encoding method	MFM
Rotational speed	360 RPM
Transfer rate	500K bits per second
Latency (average)	83 milliseconds
Access time:	
Average	91 milliseconds
Track to track	3 milliseconds
Settling time	18 milliseconds
Head load time	50 milliseconds
Motor start time (including head load time)	750 milliseconds

Adapter Card Unit

Description

The Adapter Card Unit attaches to the top of the system unit, allowing the attachment of up to three adapter cards to the system. Two cards up to 280 mm (11 inches) long and one card up to 241 mm (9.5 inches) long can be installed. The adapter card unit consists of an expansion card, a power card, a brushless 12V dc fan, and a metal cover set.

The expansion card attaches to the adapter card unit connector on the system board. The card carries system bus signals between the system unit and the three 2 by 49 adapter card connectors. Also, the fan and the power card connect to the expansion card. The I/O Connector assignments are shown in Figure A-10.

The power card receives 30 – 40V bulk dc (36V nominal) from the system unit (via the expansion card) and generates + 5V dc at 6A maximum. The card uses ac/dc switching technology operating at a frequency of approximately 1 MHz. The card also receives –12V and generates –5V at 70 mA maximum using a linear regulator. See Section 4, "Power Supply" for details.

Power Supply

Four voltage levels are provided for adapter cards. The total current available for each voltage in the adapter card unit is as follows:

- + 5V dc (+ 5%, – 3%) at 4.2 A
- – 5V dc (+ 10%, – 10%) at 0.60 A
- + 12V dc (+ 5%, – 3%) at 0.66 A
- – 12V dc (+ 10%, – 10%) at 0.27 A.

The total power used by the cards must be less than 30 watts.

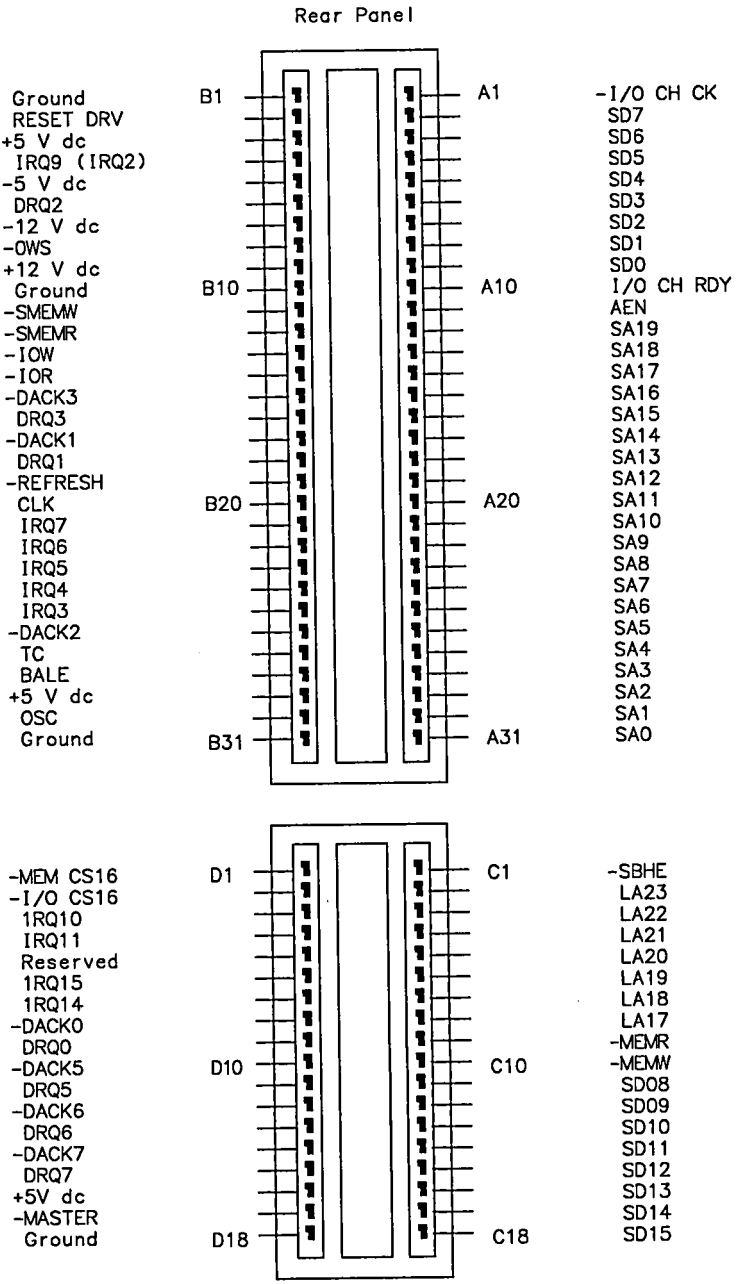


Figure A-10. I/O Channel Connectors.

Specifications

The following are specifications for the Adapter Card Unit:

Size

- Width: 274 millimeters (10.8 inches)
- Depth: 320 millimeters (12.6 inches)
- Height: 69 millimeters (2.7 inches).

Weight

- 2.3 kilograms (5 pounds).

Glossary

This glossary includes terms and definitions from *IBM Vocabulary for Data Processing, Telecommunications, and Office Systems*, GC20-1699.

μ. Prefix micro; 0.000 001.

μs. Microsecond; 0.000 001 second.

A. Ampere.

ac. Alternating current.

accumulator. A register in which the result of an operation is formed.

active high. Designates a signal that has to go high to produce an effect.

active low. Designates a signal that has to go low to produce an effect.

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.

algorithm. A finite set of well-defined rules for the solution of a problem in a finite number of steps.

all points addressable (APA). Mode in which all points of a displayable image can be controlled by the user.

alphameric. Synonym for alphanumeric.

alphanumeric (A/N). Pertaining to a character set that contains letters, digits, and usually other characters, such as punctuation marks. Synonymous with alphameric.

alternating current (ac). A current that periodically reverses its direction of flow.

American National Standard Code for Information Interchange (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 gate. A logic gate in which the output is 1 only if all inputs are 1.

APA. All points addressable.

ASCII. American National Standard Code for Information Interchange.

assemble. To translate a program expressed in an assembler language into a computer language.

assembler. A computer program used to assemble.

assembler language. A computer-oriented language whose instructions are usually in one-to-one correspondence with computer instructions.

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 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.

block. (1) A string of records, a string of words, or a character string formed for technical or logic reasons to be treated as an entity. (2) A set of things, such as words, characters, or digits, treated as a unit.

block-check character (BCC). In cyclic redundancy checking, a character that is transmitted by the sender after each message block and is compared with a block-check character computed by the receiver to determine if the transmission was successful.

bps. Bits per second.

BSC. Binary synchronous communications.

bus master. A device that arbitrates for ownership of the channel and, upon winning ownership, issues address and control signals to a slave.

buffer. (1) An area of storage that is temporarily reserved for use in performing an input or 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.

capacitor. An electronic circuit component that stores an electric charge.

CAS. Column address strobe.

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 using a dot matrix. (2) Synonymous with monitor.

CCITT. International Telegraph and Telephone Consultative Committee.

Celsius (C). A temperature scale. Contrast with Fahrenheit (F).

central processing unit (CPU). Term for processing unit.

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.

check key. A group of characters, derived from and appended to a data item, that can be used to detect errors in the data item during processing.

clipping. In computer graphics, removing parts of a display image that lie outside a window.

closed circuit. A continuous unbroken circuit; that is, one in which current can flow. Contrast with open circuit.

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.

collector. An element in a transistor toward which current flows.

column address strobe (CAS). A signal that latches the column addresses in a memory chip.

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.

complement. A number that can be derived from a specified number by subtracting it from a second specified number.

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.

computer instruction code. A code used to represent the instructions in an instruction set. Synonymous with machine code.

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, refers to a hardware configuration or a software configuration. (2) The devices and programs that make up a system, subsystem, or network.

conjunction. Synonym for AND operation.

contiguous. Touching or joining at the edge or boundary; adjacent.

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.

control storage. A portion of storage that contains microcode.

coordinate space. In computer graphics, a system of Cartesian coordinates in which an object is defined.

cps. Characters per second.

CPU. Central processing unit.

CRC. Cyclic redundancy check.

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 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.

cyclic redundancy check (CRC). (1) A redundancy check in which the check key is generated by a cyclic algorithm. (2) A system of error checking performed at both the sending and receiving station after a block-check character has been accumulated.

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 transmission. Synonym for transmission.

dB. Decibel.

dc. Direct current.

decibel. (1) A unit that expresses the ratio of two power levels on a logarithmic scale. (2) A unit for measuring relative power.

Deutsche Industrie Norm (DIN). (1) German Industrial Standard. (2) The committee that sets German industrial standards.

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.

DIN. Deutsche Industrie Norm.

DIN connector. One of the connectors specified by the DIN committee.

DIP. Dual in-line package.

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.

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 device for storing data on and retrieving data from a diskette.

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.

display element. In computer graphics, a basic graphic element that can be used to construct a display image; for example, a dot, a line segment, a character.

display image. In computer graphics, a collection of display elements or display groups that are represented together at any one time in a display space.

display surface. In computer graphics, that medium on which display images are shown; for example, the entire screen of a cathode ray tube.

DMA. Direct memory access.

DMA controller. A device that does NOT arbitrate for ownership of the channel but tracks the arbitration process. If programmed to support the winning arbitration level, the DMA controller issues address and control signals to move data to or from the DMA slave that won ownership of the bus through arbitration.

dot-matrix character generator. In computer graphics, a character generator that generates character images composed of dots.

DSR. Data set ready. Associated with modem control.

DTR. Data terminal ready, associated with modem control.

dual in-line package (DIP). A widely used container for an integrated circuit. DIPs have pins in

two parallel rows. The pins are spaced 1/10 inch apart.

duplex. (1) In data communication, a simultaneous two-way independent transmission in both directions. (2) Contrast with half-duplex.

dynamic memory. RAM using transistors and capacitors as the memory elements. This memory requires a refresh (recharge) cycle every few milliseconds.

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.

EIA. Electronic Industries Association.

electromagnet. Any device that exhibits magnetism only while an electric current flows through it.

enable. To initiate the operation of a circuit or device.

end of block (EOB). A code that marks the end of a block of data.

end of file (EOF). An internal label, immediately following the last record of a file, signaling the end of that file. It may include control totals for comparison with counts accumulated during processing.

error checking and correction (ECC). The detection and correction of all single-bit errors, plus the detection of double-bit and some multiple-bit errors.

ESC. 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 interpreted according to a different code or according to a different coded character set.

ETB. End-of-transmission-block.

ETX. End-of-text.

F. Fahrenheit.

Fahrenheit (F). A temperature scale. Contrast with Celsius (C).

falling edge. Synonym for negative-going edge.

FCC. Federal Communications Commission.

fetch. To locate and load a quantity of data from storage.

field. (1) In a record, a specified area used for a particular category of data. (2) In a data base, the smallest unit of data that can be referred to.

FIFO (first-in-first-out). A queuing technique in which the next item to be retrieved is the item that has been in the queue for the longest time.

fixed disk drive. A unit consisting of nonremovable magnetic disks, and a device for storing data on and retrieving data from the disks.

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.

flip-flop. A circuit or device containing active elements, capable of assuming either one of two stable states at a given time.

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.

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.

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.

highlighting. In computer graphics, emphasizing a given display group by changing its attributes relative to other display groups in the same display field.

housekeeping. Operations or routines that do not contribute directly to the solution of the problem but do contribute directly to the operation of the computer.

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.

immediate instruction. An instruction that contains an operand for the operation specified, rather than an address of the operand.

index register. A register with contents that can be used to modify an operand address during the execution of computer instructions.

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 can 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 as *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.

intensity. In computer graphics, the amount of light emitted at a display point.

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.

irrecoverable error. An error that makes recovery impossible without the use of recovery techniques external to the computer program or run.

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.

LED. Light-emitting diode.

light-emitting diode (LED). A semiconductor device that gives off visible or infrared light when activated.

load. In programming, to enter data into storage or working registers.

look-up table (LUT). (1) A technique for mapping one set of values into a larger set of values. (2) In computer graphics, a table that assigns a color value (red, green, blue intensities) to a color index.

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.

LUT. Look-up table.

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.

machine code. The machine language used for entering text and program instructions onto the recording medium or into storage and which is subsequently used for processing and printout.

machine language. (1) A language that is used directly by a machine. (2) Deprecated term for computer instruction code.

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, a word, an item of data, or 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.

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.

microcode. (1) One or more microinstructions. (2) A code, representing the instructions of an instruction set.

microinstruction. (1) An instruction of microcode. (2) A basic or elementary machine instruction.

microprocessor. An integrated circuit that accepts coded instructions for execution; the instructions may be entered, integrated, or stored internally.

microsecond (μs). 0.000,001 second.

MIDI. Musical instrument digital interface.

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. Synonym for cathode ray tube display (CRT display).

most-significant digit. The leftmost (non-zero) 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.

musical instrument digital interface (MIDI). An electronic interface to instruments and synthesizers to

allow recording and playback of music.

n. Prefix nano; 0.000 000 001.

NAND. A logic operator having the property that if P is a statement, Q is a statement, R is a statement,..., then the NAND of P, Q ,R,... is true if at least one statement is false, false if all statements are true.

NAND gate. A gate in which the output is 0 only if all inputs are 1.

nano (n). Prefix 0.000 000 001.

nanosecond (ns). 0.000 000 001 second.

non-return-to-zero (inverted) recording (NRZI). Deprecated term for non-return-to-zero change-on-ones recording.

NOR. A logic operator having the property that if P is a statement, Q is a statement, R is a statement,..., then the NOR of P, Q, R,... is true if all statements are false, false if at least one statement is true.

NOR gate. A gate in which the output is 0 only if at least one input is 1.

NOT. A logical operator having the property that if P is a statement, then the NOT of P is true if P is false, false if P is true.

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 can 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.

odd-even check. Synonym for parity check.

offline. Pertaining to the operation of a functional unit without the continual control of a computer.

open circuit. (1) A discontinuous circuit; that is, one that is broken at one or more points and, consequently, cannot conduct current. Contrast with closed circuit. (2) Pertaining to a no-load condition; for example, the open-circuit voltage of a power supply.

open collector. A switching transistor without an internal connection between its collector and the voltage supply. A connection from the collector to the voltage supply is made through an external (pull-up) resistor.

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 or output control, and data management.

OR. A logic operator having the property that if P is a statement, Q is a statement, R is a statement,..., then the OR of P, Q, R,...is true if at least one statement is true, false if all statements are false.

OR gate. A gate in which the output is 1 only if at least one input is 1.

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.

overcurrent. A current of higher than specified value.

overrun. Loss of data because a receiving device is unable to accept data at the rate it is transmitted.

overvoltage. A voltage of higher than specified value.

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).

parity check. (1) A redundancy check that uses a parity bit.
(2) Synonymous with odd-even check.

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.

polling. (1) Interrogation of devices for purposes such as to avoid contention, to determine operational status, or to determine readiness to send or receive data. (2) The process whereby stations are invited, one at a time, to transmit.

port. An access point for data entry or exit.

POS. Programmable option select.

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 option select (POS). A method by which device options are initialized in a device-resident register by the power-on self-test. This method replaces the setting of switches or jumpers.

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.

propagation delay. (1) The time necessary for a signal to travel from one point on a circuit to another. (2) The time delay between a signal change at an input and the corresponding change at an output.

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.

RAM. Random access memory. Read/write memory.

random access memory (RAM). Read/write memory.

RAS. Row address strobe.

raster. In computer graphics, a predetermined pattern of lines that provides uniform coverage of a display space.

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.

redundancy check. A check that depends on extra characters attached to data for the detection of errors. See cyclic redundancy check.

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.

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.

row address strobe (RAS). A signal that latches the row address in a memory chip.

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.

saturation. In computer graphics, the purity of a particular hue. A color is said to be saturated when at least one primary color (red, green, or blue) is completely absent.

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.

SDLC. Synchronous Data Link Control.

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.

short circuit. A low-resistance path through which current flows, rather than through a component or circuit.

signal. A variation of a physical quantity, used to convey data.

sink. A device or circuit into which current drains.

slave. A device that recognizes its address on the channel and responds to control signals by executing the specified Read or Write command.

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.

square wave. An alternating or pulsating current or voltage whose waveshape is square.

start bit. (1) A signal to a receiving mechanism to get ready to receive data or perform a function. (2) In a start-stop system, a signal preceding a character or block that prepares the receiving device for the reception of the code elements.

start-of-text (STX). A transmission control character that precedes a text and may be used to terminate the message heading.

stop bit. (1) A signal to a receiving mechanism to wait for the next signal. (2) In a start-stop system, a signal following a character or block that prepares the receiving device for the reception of a subsequent character or block.

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.

STX. Start-of-text.

symbol. (1) A conventional representation of a concept. (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.

syntax. (1) The relationship among characters or groups of characters, independent of their meanings or the manner of their interpretation and use. (2) The structure of expressions in a language. (3) The rules governing the structure of a language. (4) The relationships among symbols.

text. In ASCII and data communication, a sequence of characters treated as an entity if preceded and terminated by one STX and one ETX transmission control character, respectively.

time out. (1) A parameter related to an enforced event designed to occur at the conclusion of a predetermined elapsed time. (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.

vector. In computer graphics, a directed line segment.

video. Computer data or graphics displayed on a cathode ray tube, monitor, or display.

virtual space. In computer graphics, a space in which the coordinates of the display elements are expressed in terms of user coordinates.

volt. The basic practical unit of electric pressure. The potential that causes electrons to flow through a circuit.

W. Watt (W).

watt. The practical unit of electric power.

window. (1) In computer graphics, a predefined part of the virtual space. (2) In computer graphics, the visible area of a viewplane mapped into a viewport.

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.

Bibliography

- Microprocessor and Peripheral Handbook
  - INTEL Corporation.*210844.XXX*
- Introduction to the iAPX 286
  - INTEL Corporation.*210308.XXX*
- iAPX 286 Operating Systems Writer's Guide
  - INTEL Corporation.*121960.XXX*
- iAPX 286 Programmer's Reference Manual
  - INTEL Corporation.*210498.XXX*
- iAPX 286 Hardware Reference Manual
  - INTEL Corporation.*210760.XXX*
- National Semiconductor Corporation. *NS16550*
- Motorola Microprocessor's Data Manual
  - Motorola Inc. *Series B*
- 89024 Reference Manual
  - INTEL Corporation.*296235.XXX*

Index

A

AAA instruction 10-11
AAD instruction 10-12
AAM instruction 10-12
AAS instruction 10-11
acknowledge (ACK) command 5-28
adapter card unit A-38
ADC instruction 10-8
ADD instruction 10-8
address generation for DMA channels 0 through 3 2-20
address generation for DMA channels 5 through 7 2-21
address maps
  DMA 2-23
  memory 3-157
  RTC/CMOS RAM 3-158
  system 1-7
address register 3-58
addresses 8-5
AEN (address enable) 2-7
alphanumeric modes 3-25
alternate key 5-47
alternate mouse user subroutine 6-27
analog interface 9-20
AND instruction 10-13
application guidelines 12-6
applications
  communications 3-136
  compatibility guidelines 12-6
arithmetic instructions 10-8
ARPL instruction 10-26
ASCII characters 11-5
ASCII, extended 5-42
assembler language programming considerations 12-7
assignments
  function code 12-15
  interrupt 2-29
asynchronous communications 3-135
AT commands 9-8
AT (attention) command description 9-6
attachment control register 8-7
attachment status register 8-6
attention register 8-9
attribute address register 3-87
attribute controller 3-21
attribute controller registers 3-87
attribute mode control register 3-89
audio
  description 2-42
  speaker 2-42
  subsystem 2-42
audio card & joystick A-5
auto restart 4-4
auxiliary device
  controller 3-3
  controller commands 3-6
  controller programming 3-9
  data stream bit definitions 3-10
  mouse/system timings 3-10
  sending and receiving data 3-10

B

BALE (address latch enable), I/O channel 2-7
base memory bytes (RTC/CMOS) 3-164
BASIC 12-7
basic assurance test (BAT) 5-22
BASIC call 6-12
BAT completion code command 5-28
BAT failure code 5-28
baud-rate generator 3-137, 3-138
Belgium (Dutch) keyboard 5-6
Belgium (French) keyboard 5-10
BIOS
  BIOS video modes 3-22
  bypassing 12-4

BIOS *(continued)*
compatibility 12-10
function calls 12-10
interface to hardware 12-6
parameter tables 12-10
BIOS/ROM 3-15, 3-18
bit mask register 3-86
bit-mapped registers 9-17
block diagrams
attribute controller, video 3-21
fixed disk and controller 8-4
graphics controller 3-20
parallel port controller 3-149
serial port 3-135
system board 1-6
timer 2-31
video subsystem 3-17
BOUND instruction 10-23
break code 5-21
break key 5-48
buffers
keyboard 5-21
overrun condition 5-21
bus card 2-5
busy loop 12-14
buttons
description 6-3
press information 6-17
release information 6-18
status 6-16
bypassing BIOS 12-4

C

cables and connectors 5-50
call driver, BASIC 6-12
CALL instruction 10-18
call to device driver 6-14
calls, function number 6-15
Canada (English) keyboard 5-20
Canada (French) keyboard 5-7
caps lock key 5-47
CBW instruction 10-12
channel
channel check enable 3-165
channel check latch 3-165
channel check (-I/O CH CK) 2-8
channel *(continued)*
channel ready (I/O CH RDY) 2-8
channel 0 - system timer 2-32
channel 0 count register 2-34
channel 1 - refresh request generator 2-32
channel 2 - tone generation 2-32
channel 2 count register 2-34
DMA 2-23
character codes 5-42, 11-4
character generator 3-61, 3-90
character generator, RAM-loadable 3-97
character map select register 3-61
characters 11-1, 11-5
charts, keyboard 5-3
classes, wait loop 12-14
CLC instruction 10-23
CLD instruction 10-23
CLI instruction 10-23
CLK 2-8
clock
keyboard 5-39
RTC/CMOS 3-158
clock and data signals 5-39
clock (CLK) 2-8
clocking mode register 3-59
CMC instruction 10-23
CMOS RAM 3-158
CMP instruction 10-10
CMPS instruction 10-16
COBOL 12-7
code pages
available 11-3
437 11-11
850 11-12
860 11-13
863 11-14
865 11-15
codes
BAT completion command 5-28
break 5-21
character 5-42
extended, keyboard 5-45
machine identification 12-17
machine-sensitive 12-17

codes *(continued)*
make 5-21
make/break 5-32
overrun 5-21
response 5-21
scan 5-29
scan, set 1 5-29
scan, set 2 5-32
scan, set 3 5-36
select alternate scan command 5-24
set all keys command 5-25
set default command 5-25
set key type command 5-25
set typematic rate/delay 5-26
set/reset status indicators command 5-26
codes, scan
set 1 5-29
set 2 5-32
set 3 5-36
color and monochrome modes 3-22
color compare register 3-81
color don't care register 3-86
color graphics
color graphics modes 640 x 350 3-31
modes 320 x 200, 4 color 3-28
modes 640 x 480, two-color 3-30
modes 640-by-200, 2 color 3-30
color plane enable register 3-91
color select register 3-93
colors 11-1
combinations, shift key 5-48
command format 3-118
command specify block 8-12
command status registers 3-128
command summary (diskette) 3-115
command syndrome 8-16
commands
acknowledge (ACK) 5-28
BAT completion code 5-28
BAT failure code 5-28
command symbols (diskette) 3-116
commands *(continued)*
controller 8-3
counter latch 2-33, 2-35
default disable 5-23
disable 6-5
diskette drive controller 3-115, 3-119
echo 5-23, 5-28
enable 5-24, 6-5
end-of-interrupt 12-6
format disk 8-22
format track 8-23
from the system, keyboard 5-23
invalid 5-24
I/O 12-9
key detection error 5-29
keyboard ID 5-28
keyboard/mouse controller 3-6
mouse 6-4
overrun 5-29
password 3-3
read check 8-19
read data 6-6, 8-19
read device type 6-6
read ID 5-24
resend 5-24, 5-29, 6-5
reset 5-24, 6-5
reset scaling 6-8
reset wrap mode 6-6
seek 8-23
select alternate scan codes 5-24
set all keys 5-25
set default 5-25, 6-5
set key type 5-25
set remote mode 6-6
set resolution 6-7
set sampling rate 6-6
set scaling 2:1 6-8
set stream mode 6-6
set typematic rate/delay 5-26
set wrap mode 6-6
set/reset status indicators 5-26
status request 6-7
write data 8-21
write extended 8-22
write verify 8-21

communications
application 3-136
asynchronous 3-135
serial 3-135
compatibility
application guidelines 12-6
assembler language programming considerations 12-7
BIOS 12-10
CRT controller 3-78
diskette 12-4
diskette drive 3-111
diskette drives and controller 12-4
DMA 2-19
hardware 12-12
hardware interfaces 12-12
hardware interrupts 12-6
high-level language considerations 12-7
interrupt levels 2-29
microprocessor 12-3
opcodes 12-8
parallel port controller 3-149
serial port 3-147
software interrupts 12-7
system 12-3
system board 12-3
components, of VGA 3-18
conditional off 6-24
condition, wait 12-13
configuration
CMOS RAM 3-162
RTC/CMOS RAM 3-158
configuration control register (diskette) 3-113
configuration CRC bytes (RTC/CMOS) 3-164
connectors
adapter card unit 2-5
diskette drive 3-134, 8-28
display 3-110, 7-5
fixed disk 8-24
headphone 7-3
interposer 3-134
keyboard 5-50
connectors *(continued)*
keyboard/mouse 3-13
memory option 3-157
mouse 6-32
parallel port 3-155
power card 4-5
power supply 4-5
power/audio 7-5
RS-232C 9-22
serial port 9-22
video (15-pin) 3-110
Consecutive I/O commands 12-9
control block
command control 8-10
command specify 8-12
format control 8-17
sense summary 8-13
control byte register - channel 0, 1, or 2 2-34
control key 5-46
control registers 2-32
control transfer instructions 10-18
controllers
diskette drive 3-111
DMA 2-19
fixed disk 8-4
interrupt 2-28
keyboard 3-3
keyboard/mouse 3-3
memory refresh 2-4
parallel port 3-149
serial port 3-135
VL16C451A 3-135
copy protection 12-4
count registers 2-32
counter latch command 2-35
counters
See also timers
counter latch command 2-33, 2-35
rate generator 2-37
read operations 2-33
registers 2-34
system timer modes 2-36
timers 0, 1, and 2 2-32
write operations 2-33

CRT controller
address register 3-65
compatibility 3-78
description 3-18
mode control register 3-76
overflow register 3-68
registers 3-64
ctrl state 5-44
CTS instruction 10-24
cursor
set graphics block 6-20
set position 6-17
set text 6-21
show and hide 6-16
cursor end register 3-71
cursor location high register 3-72
cursor location low register 3-72
cursor start register 3-70
CWD instruction 10-12

D

DAA instruction 10-11
DAS instruction 10-11
data
address port (parallel) 3-151
bits (D0-D7) 2-8
input, keyboard 5-41
lines 6-10
modes 6-33
output, keyboard 5-40
register 8-6
registers diskette 3-114
report 6-9
rotate register 3-81
stream 3-10, 5-39
transfer instructions 10-3
transfers 2-20
transmission 6-10
transmission, input 6-11
transmission, output 6-10
width conversions 3-111
data and clock signals 5-39
date century byte
(RTC/CMOS) 3-164
DEC instruction 10-10
default disable command 5-23
defaults 6-5
delay, typematic 5-21
Denmark keyboard 5-8
description
audio 2-42
diskette drive controller 3-111
displays 7-3
DMA controller 2-19
drives 8-3
fixed disk drive 8-3
fixed disk drive controller 8-3
keyboard 5-3
keyboard/mouse controller 3-3
memory 3-156
microprocessor 2-3
mouse 6-3
parallel controller 3-149
photo displays 7-3
power supply 4-3
RS-232C interface card 9-21
serial controller 3-135
system board 1-4
system timers 2-31
video subsystem 3-14
2400 bps modem 9-3
3.5-inch diskette drive 8-27
device driver functions 6-15
device driver interface 6-12
diagnostic status byte
(RTC/CMOS) 3-162
diagrams
block, parallel port
controller 3-149
block, serial port 3-135
system board block 1-6
timer block 2-31
dialer type 9-20
digital input register
(diskette) 3-113
digital output register
(diskette) 3-113
direct drive analog displays 3-23
direct memory access
byte pointer 2-20
channels 2-23
clear mask 2-25

direct memory access *(continued)*
compatibility 2-19
controller 2-19
controller operations 2-20
data transfers 2-20
description 2-19
I/O address map 2-23
mask register 2-25
memory address registers 2-24
microprocessor access 2-24
mode register 2-26
program condition 2-19
registers 2-24
set mask 2-25
status register 2-26
transfer 2-19
transfer count registers 2-24
disable command 6-5
disable mouse driver 6-30
diskette change signal 12-5
diskette compatibility 12-4
diskette drive
compatibility 12-4
controller 3-111
gap length parameter 12-10
head settle time 12-11
programming considerations 8-28
unit A-29
3.5-inch 8-27
5.25-inch double-sided A-32
5.25-inch high capacity A-35
diskette drive controller
command format 3-119
command phase 3-115
command status registers 3-128
command summary 3-116, 3-119
command symbols 3-116
commands 3-115, 3-119
compatibility 12-4
connector 3-134
copy protection 12-4
description 3-111
digital output register 3-113
diskette drive controller 3-115
execution phase 3-115
diskette drive controller *(continued)*
interposer board 3-111
interrupt level 2-29
multibyte transfer 3-115
precompensation 3-111
registers 3-112
result phase 3-115
signal descriptions 3-131
signals 3-134
support 3-111
voltages 3-134
diskette drive controller status register 3-114
diskette drive type byte (RTC/CMOS) 3-163
diskette write current 12-5
display connector timing (SYNC signals) 3-105
display connector (15-pin) 3-110
display support 3-23
displays 3-23
displays, PHOTO GRAPHIC
characteristics 7-3
description 7-3
display connector 7-5
models 7-3
power/audio cable 7-5
signals 7-4
specifications 7-6
vertical modes 7-4
display, vertical gain 3-54, 3-105
DIV instruction 10-12
divide error exception 12-8
divisor latch access bit (DLAB) 3-141
divisor latch registers (serial) 3-138
DMA
See also direct memory access
channels 2-20
controller 2-19
read 2-17
registers 2-24
request 1 to 3 (DRQ1—DRQ3) 2-8
write 2-18

drive analog displays 3-23
drives
  description 8-3
  programming 8-5
DTMF dialing 9-20

E

earphone jack 1-3
echo command 5-23, 5-28
edge-sensitive interrupts 12-6
enable command 6-5
enable command, keyboard 5-24
enable mouse driver 6-30
enable set/reset register 3-80
encode, keyboard 5-42
end horizontal blanking
  register 3-66
end horizontal retrace
  register 3-67
end vertical blanking 3-76
end vertical blanking register 3-75
end-of-interrupt command 12-6
ENTER instruction 10-22
error handling 6-5, 6-10
error recovery 6-10
ESC instruction 10-24
exception, divide error 12-8
extended ASCII 5-42
extended codes, keyboard 5-45

F

feature control 3-57
FIFO (first-in-first-out) 5-21
Finland keyboard 5-9
fixed disk drive and controller
  compatability 12-5
  controller 8-4
  interrupt level 2-29
  power card connectors 4-5
  specifications 8-26
fixed disk drive and controller,
  3.5-inch 30MB 8-3
fixed disk type bytes
  (RTC/CMOS) 3-163
format a track command
  format 3-125
format control block 8-17
format disk 8-22
format track 8-23
format, data report 6-9
FORTRAN 12-7
frame 6-11
framing error 6-11
France keyboard 5-10
full-screen updates 3-59
function calls 6-15
function calls, operating
  system 12-10
function calls, ROM BIOS 12-10
function code assignments 12-15
function codes, multitasking 12-14

G

general registers 3-54
Germany keyboard 5-11
graphics
  address register 3-79
  controller 3-19
  controller registers 3-79
  cursor block 6-20
  modes 3-28
graphics controller
graphics mode register 3-83
graphics modes
  16-color graphics 3-32
  256-color graphics mode (mode
    hex 13) 3-33
  320 x 200 four-color graphics
    (modes hex 4 and 5) 3-28
  640 x 200 two color graphics
    (mode hex 6) 3-30
  640 x 350 graphics (mode hex
    F) 3-31
  640 x 480 two-color graphics
    (mode hex 11) 3-30

H

hardware compatibility 12-12

hardware interrupts 12-6
headphone jack 1-3
hide cursor 6-16
high-level language considerations 12-7
HLT instruction 10-24
horizontal display enable end register 3-66
horizontal PEL panning register 3-92
horizontal position 6-18
horizontal total register 3-65

I

IBM C 12-7
IDIV instruction 10-12
IIMUL instruction 10-11
IMUL instruction 10-11
IN instruction 10-6
INC instruction 10-8
initialization 6-14
input
  buffer 3-6
  data transmission 6-11
  mask 6-22
  signals 3-132
  signals, diskette drive 8-28
  status register 0 3-56
  status register 1 3-56
INS instruction 10-16
installed flag 6-15
installed flag and reset 6-15
instructions
  arithmetic 10-8
  control transfer 10-18
  data transfer 10-3
  logic 10-13
  processor control 10-23
  protection control 10-25
  repeated string manipulation 10-17
  rotate 10-13
  shift 10-13
  string manipulation 10-16
INT instruction 10-22
interfaces, multitasking 12-13
interface, mouse device driver 6-12
interface, programming 6-15
interposer board 3-111
interrupt
  assignments 2-29
  audio card and joystick 2-29
  cascade interrupt control 2-29
  controller 2-28, 8-3
  description 2-28
  edge-sensitive 12-6
  hardware 12-6
  hex 33 6-12, 6-13
  IRQ 0 - IRQ 15 2-29
  IRQ 0 reset 3-165
  levels 2-29
  NMI enable 3-166
  NMI reset 2-28
  non-maskable (NMI) 2-28
  redirect cascade 2-29
  request 2-9, 8-3
  reserved levels 2-29
  serial port 3-139
  sharing 12-6
  single-step 12-8
  software 12-7
  writing port hex 0070 2-28
interrupt enable register (serial) 3-139
interrupt identification register (serial) 3-139
interrupt status register 8-8
INTO instruction 10-22
invalid command 5-24
invalid command format 3-127
IRET instruction 10-23
IRQ 2 12-10
IRQ 9 12-10
Italy keyboard 5-12
I/O
  address map 1-7
  address map, DMA 2-23
  addresses 8-5
  CH CK 2-8
  commands and STI instructions 12-9

I/O *(continued)*
CS16 (I) 2-10
I/O channel
-memory refresh 2-9
-MEMR, -MEMW 2-9
BALE (address latch enable) 2-7
channel check (-I/O CH CK) 2-8
channel ready (I/O CH RDY) 2-8
CLK 2-8
oscillator (OSC) 2-9
read (-IOR) 2-8
Reset Drive (RESET DRV) 2-9
signals 2-5
terminal count (TC) 2-10
timing 2-11
write (-IOW) 2-9

J

jack, headphone 1-3
JBE/JNA instruction 10-20
JB/JNAE instruction 10-20
JCXZ instruction 10-22
JE/JZ instruction 10-20
JLE/JNG instruction 10-20
JL/JNGE instruction 10-20
JMP instruction 10-19
JNBE/JA instruction 10-21
JNB/JAE instruction 10-21
JNE/JNZ instruction 10-21
JNLE/JG instruction 10-21
JNL/JGE instruction 10-21
JNO instruction 10-21
JNP/JPO instruction 10-21
JNS instruction 10-22
JO instruction 10-21
JP/JPE instruction 10-21
JS instruction 10-21

K

key detection error command 5-29
key encode 5-42
key-code scanning 5-21
keyboard
acknowledge (ACK) command 5-28
keyboard *(continued)*
alternate key 5-47
basic assurance test (BAT) 5-22
BAT completion code command 5-28
BAT failure code 5-28
break key 5-48
buffer 5-21
cabling and connectors 5-50
caps lock key 5-47
character codes 5-42
charts
Belgium (Dutch) 5-6
Belgium (French) 5-10
Canada (English) 5-20
Canada (French) 5-7
Denmark 5-8
Finland 5-9
France 5-10
Germany 5-11
Italy 5-12
Netherlands 5-13
Norway 5-14
Portugal 5-15
Spain 5-16
Sweden 5-9
Switzerland (French) 5-17
Switzerland (German) 5-18
U.K. 5-19
U.S. (English) 5-20
commands from the system 5-23
command/status port 3-3
connectors 3-13, 5-50
control key 5-46
controller 3-3
controller commands 3-6
controller programming 3-9
ctrl state 5-44
data input 5-41
data output 5-40
data stream 5-39
default disable command 5-23
description 5-3
echo command 5-23, 5-28
enable command 5-24
extended codes 5-45

keyboard *(continued)*
ID command 5-28
interrupt level 2-29
invalid command 5-24
key combinations 5-48
key detection error command 5-29
key encode 5-42
key numbering 5-4
key positions 5-4
key priorities 5-48
layouts 5-3, 5-4
num lock state 5-44
number lock key 5-47
overrun command 5-29
password 3-3
pause key 5-48
power on routine (POR) 5-22
print screen key 5-48
read ID command 5-24
resend command 5-24, 5-29
reset command 5-24
routine 5-49
scan-codes 5-29
scroll lock key 5-47
select alternate scan codes command 5-24
set all keys command 5-25
set default command 5-25
set key type command 5-25
set typematic rate/delay 5-26
set/reset status indicators command 5-26
shift key 5-46
shift state 5-44, 5-46
signals 5-39
specifications 5-51
system request key 5-49
system reset 5-48
keyboard ID command 5-28
keyboard/auxiliary device controller
command byte 3-5
commands 3-6
command/status port 3-3
connectors 3-13
description 3-3
driver 3-12
keyboard/auxiliary device controller *(continued)*
input buffer 3-6
output buffers 3-6
password commands 3-3
signals 3-12
status byte 3-6
keys
num lock 5-21
pause 5-21
sequential key-code scanning 5-21
typematic 5-21
keystrokes 11-1

L

LAHF instruction 10-6
LAR instruction 10-26
layouts
key positions 5-4
keyboard 5-3
101-key keyboard 5-4
102-key keyboard 5-5
LA17—LA23 (O) 2-7
LDS instruction 10-6
LEA instruction 10-6
LEAVE instruction 10-22
LES instruction 10-6
LGDT instruction 10-25
LIDT instruction 10-25
light pen emulation 6-23
line
compare register 3-79
contention, keyboard 5-40
control register (serial) 3-141
graphics character 3-89
protocol, keyboard 5-22
status register (serial) 3-144
LLDT instruction 10-25
LMSW instruction 10-25
locations
system board connectors 1-5
LOCK instruction 10-24
LODS instruction 10-16
logic instructions 10-13

LOOP instruction 10-22
LOOPNZ/LOOPNE
instruction 10-22
LOOPZ/LOOPE instruction 10-22
loop, busy 12-14
LSL instruction 10-26
LTR instruction 10-25

M

machine model byte 12-17
machine-sensitive programs 12-17
make/break codes 5-21
map mask register 3-60
mask register (DMA) 2-25
MASTER (I) 2-10
maximum scan line register 3-70
MEM CS16 (I) 2-10
memory
address registers (DMA) 2-24
description 3-156
disable 3-156
disabling 3-157
DMA controller 2-19
expansion bytes
(RTC/CMOS) 3-164
I/O 3-157
map 3-157
memory module
packages 3-156
mode register 3-62
module package
connectors 3-157
module packages 3-156
RAM access 2-4
random access (RAM) 3-156
read only (ROM) 3-156
read (-MEMR) 2-9
refresh bandwidth 2-4
refresh controller 2-4
refresh (-MREF) 2-9
ROM access 2-4
RTC/CMOS RAM 3-158
setup 3-160
size miscompare 3-162
system board 3-157
video 3-157
memory *(continued)*
wait states 2-4
write (-MEMW) 2-9
memory expansion, 512KB A-3
microprocessor
access to DMA 2-24
compatibility 12-3
data transfer instructions 10-3
description 2-3
exception 2-3
features 2-3
interrupt 2-3
opcodes 12-8
performance 2-4
protected mode 2-4
real address mode 2-3, 3-167
segments 2-3
speed 2-4
80286 anomalies 12-9
MIDI A-8
miscellaneous register 3-84
mode
alphanumeric 3-25, 3-40
chart 3-22
color and monochrome 3-22
graphics 3-28
keyboard data stream 5-23
protected 2-4
real address 2-3
register (DMA) 2-26
vertical 7-4
video 6-31
mode hex
D 3-41
E 3-43
F 3-45
F, 640 x 350 graphics 3-31
0, 1 3-36
10 3-46
10, D, E, and 12, 16-color
graphics modes 3-32
11 3-47
11, 640 x 480 two-color
graphics 3-30
12 3-48
13 3-49
13, 256-color graphics
mode 3-33

mode hex *(continued)*
2, 3 3-37
4 and 5, 320 x 200 four-color graphics 3-28
4, 5 3-38
6 3-39
6, 640 x 200 two color graphics 3-30
7 (all variations of mode hex 7) 3-40
model byte 12-17
modem
card, 2400 bps 9-3
description 9-3
commands and operation 9-6
control function 3-135
control inputs 3-143
control interrupts 3-143
control outputs 3-143
control register (serial) 3-143
RS-232C interface cards 9-1
description 9-3
status register (serial) 3-145
modes of operation, VGA 3-22
modulation 9-20
mouse 6-1
commands 6-4
operation modes 6-3
programming considerations 6-4
specifications 6-33
mouse device driver interface 6-12
mouse/system timings 3-10
MOV instruction 10-3
MOVS instruction 10-16
MUL instruction 10-11
multiple lockout instructions 12-8
multitasking
device driver code 12-13
function codes 12-14
hooks 12-13
interfaces 12-13
provisions 12-13
serialization 12-13
startup 12-13

N

national language support 12-16
NEG instruction 10-11
Netherlands keyboard 5-13
no-load operation 4-4
non-maskable interrupt (NMI) 2-28
Norway keyboard 5-14
NOT instruction 10-15
num lock state 5-44
number lock key 5-47

O

offset register 3-74
opcodes 12-8
operating system 12-7
operating system function calls 12-10
operational characteristics 6-33
operational modes 6-3
OR instruction 10-14
organization, video memory 3-35
oscillator (OSC), I/O channel 2-9
OUT instruction 10-6
output 6-10
output buffers 3-6
output signals 3-131
outputs, keyboard 5-29
outputs, power supply 4-3
OUTS instruction 10-16
overrun command 5-29
overscan color register 3-90
overvoltage fault 4-3

P

page register addresses 2-21
palette registers 3-88
parallel control port (parallel) 3-153
parallel port controller
address assignments 3-150
bidirectional interface 3-150
block diagram 3-149
compatibility 3-149
compatible mode 3-151

parallel port controller *(continued)*
connector 3-155
data address port 3-151
description 3-149
extended mode 3-150
interrupt level 2-29
parallel control port 3-153
power-on initialization 3-151
programmable option select 3-150
programming 3-151
signals 3-154
status port 3-152
timing 3-154
parameters
gap length 12-10
tables 12-10
parity check enable 3-166
Pascal 12-7
password
keyboard 3-3
pause key 5-48
PEL, xy ratio 6-23
performance
microprocessor 2-4
PHOTO GRAPHIC display 7-3
POP instruction 10-5
POPA instruction 10-5
POPF instruction 10-7
port addresses 8-5
Portugal keyboard 5-15
POS
See programmable option select
position, minimum and maximum 6-18
position, status 6-16
POST 12-14
See also power-on self-test
power 6-33
power controller coordination 4-3
power good signal 4-4
power sequencing 3-133
power supply
adapter card unit A-38
auto restart 4-4
connectors 4-5
description 4-3
power supply *(continued)*
display, dc 7-4
no load operation 4-4
output protection 4-3
outputs 4-3
overvoltage fault 4-3
power good signal 4-4
system unit 4-3
power-on reset (POR) 5-22
power-on routine, keyboard
power-on self-test
power/audio cable connector 4-6
preset row scan register 3-69
print screen key 5-48
printer port
See parallel port controller
priorities, shift key 5-48
processor control instructions 10-23
programmable option select
parallel port controller 3-150
video subsystem setup 3-24
programmable resolution 6-33
programmable sampling rate 6-33
programming
assembler language considerations 12-7
diskette drive 8-28
diskette drive controller 3-115
DMA 2-27
fixed disk drives 8-5
interrupt controllers 2-30
keyboard/mouse 3-9
languages 12-7
parallel port controller 3-151
programming the registers 3-96
serial port controller 3-147
system timers 2-32
video graphics array 3-93
video registers 3-96
protected virtual address mode 2-4
protection control instructions 10-25
protection, power supply 4-3
protocol, keyboard 5-22
provisions, multitasking 12-13

pulse dialing 9-20
PUSH Instruction 10-4
PUSH SP 12-8
PUSHA Instruction 10-4
PUSHF Instruction 10-7

Q

query save state storage requirements 6-26
quick reference charts 11-11
quick reference, character set 11-11

R

RAM-loadable character generator 3-97
random access memory (RAM) 3-156
rate generator 2-37
rate, sampling 6-6
rate, typematic 5-21, 5-26
  to the system 5-28
read
  check 8-19
  data 6-6, 8-19
  data command format 3-119
  deleted-data command format 3-120
  device type 6-6
  extended 8-20
  ID 8-20
  ID command format 3-122
  ID command, keyboard 5-24
  I/O channel 2-8
  map select register 3-82
  memory (-MEMR) 2-9
  motion counters 6-21
  read only memory (ROM) 3-156
  track command format 3-121
read motion counters
ready (I/O CH RDY) 2-8
real address mode 2-3
real-time clock 3-160
recalibrate 8-20
recalibrate command format 3-125
receiver buffer register (serial) 3-138
receiver characteristics 9-20
recovery 6-10
refresh bandwidth 2-4
refresh controller 2-4
refresh request 3-165
register address 8-5
  attachment status 8-6
  data register 8-6
register summary 9-15
registers
  attribute controller 3-87
  command status (diskette) 3-128
  CRT controller 3-64
  digital output 3-113
  DMA 2-24
  general, video 3-54
  graphics controller 3-79
  miscellaneous output, video 3-54
  sequencer, video 3-58
  serial port controller 3-137
  status registers (RTC/CMOS) 3-160
  VGA enable 3-57
  video 3-53
remote mode 6-3, 6-6
repeated string manipulation instructions 10-17
report, data 6-9
REP/REPNE, REPZ/REPNZ instructions 10-17
request, status 6-7
resend command 5-24, 5-29, 6-5
reset
  command 5-24, 6-5
  drive signal (RESET DRV) 2-9
  mode 6-3
  power-on (POR) 5-22
  register 3-58
  scaling 6-8
  system 5-48
  wrap mode 6-6

resolution 6-7
restart, auto 4-4
restore mouse driver state 6-26
result (return) codes 9-14
RET instruction 10-20
ROM BIOS function calls 12-10
rotate instructions 10-13
routines
  basic assurance test (BAT) 5-22
  keyboard 5-49
  power-on 5-22
RS-232C interface card 9-21
RTC/CMOS RAM
  addresses 3-158
  bit definitions 3-162
  configuration 3-162
  I/O operations 3-159
  password 3-167
  RAM addresses 3-158
  real time clock bytes 3-160
  secured area 3-167
  status register A 3-160
  status register B 3-161
  status register C 3-162
  status register D 3-162
  writing port hex 0070 3-159

S

SAHF instruction 10-7
sampling rate 6-3
save mouse driver state 6-26
SA0—SA19 (O) 2-7
SBB instruction 10-9
SBHE (I/O) 2-10
scaling 6-8, 6-33
scan codes
  keyboard 5-29
  set 1 5-29
  set 2 5-32
  set 3 5-36
scanning, key-code 5-21
SCAS instruction 10-16
scratch register (serial) 3-146
screen off bit 3-59
scroll lock key 5-47
SD0—SD15 (I/O) 2-8
sector information 8-18
security, keyboard password 3-3
seek 8-23
seek command format 3-127
select alternate scan codes
  command 5-24
sense drive status command
  format 3-127
sense interrupt status command
  format 3-126
sense summary block 8-13
sequencer
  address register 3-58
  clocking mode register 3-59
  description 3-19
  map mask register 3-60
  memory mode register 3-62
  registers 3-58
  reset register 3-58
sequencing, output voltage 4-4
sequential key-code scanning 5-21
serial port controller
  baud-rate generator 3-137,
    3-138
  block diagram 3-135
  communications
    applications 3-136
  compatibility 3-147
  connector 9-22
  data format 3-136
  data speed 3-138
  description 3-135
  diagnostic capabilities 3-143
  diagnostic mode 3-143
  FE (framing error) 3-145
  interrupt level 2-29
  interrupts 3-139
  output port 1 (Serial 1) 3-136
  programming 3-147
  register addresses 3-137
  registers 3-137
  signals 3-147
  word length 3-142
serialization, multitasking 12-13
set
  all keys command 5-25

set *(continued)*
alternate mouse user subroutine 6-27
CRT page number 6-29
cursor position 6-17
default 6-5
default command 5-25
disable mouse driver 6-30
double speed threshold 6-24
enable mouse driver 6-30
get CRT page number 6-29
get driver version, mouse type, and IRQ # 6-31
get mouse sensitivity 6-29
get user alternate interrupt vector 6-28
horizontal position 6-18
input mask 6-22
key type command 5-25
mouse sensitivity 6-28
query save state storage requirements 6-26
remote mode 6-6
resolution 6-7
restore mouse driver state 6-26
sampling rate 6-6
save mouse driver state 6-26
scaling 2:1 6-8
software reset 6-30
stream mode 6-6
swap user interrupt vector 6-24
text cursor 6-21
typematic rate/delay 5-26
vertical position 6-20
wrap mode 6-6
xy/PEL ratio 6-23
1 scan code tables 5-29
2 scan code tables 5-32
3 scan code tables 5-36
setup 6-14
See also configuration
parallel port controller 3-150
set/reset register 3-80
set/reset status indicators command, keyboard 5-26
SGDT instruction 10-25
shift
counts 12-8
instructions 10-13
key 5-46
key priorities 5-48
state 5-44, 5-46
show cursor 6-16
shut down status byte (RTC/CMOS) 3-163
SIDT instruction 10-25
signal description 2-7
signal timings 2-11
signals
clock, keyboard 5-39
control, parallel port 3-154
data input, keyboard 5-41
data output, keyboard 5-40
data stream, keyboard 5-39
data, keyboard 5-39
diskette change 12-5
diskette drive 8-28
diskette drive controller 3-131, 3-134
keyboard/mouse controller 3-12, 3-13
parallel port 3-154, 3-155
photo display 7-4
power good signal 4-4
power supply 4-5
power-on reset 5-22
serial port controller 3-147, 9-22
SYNC signals (video) 3-105
single-step interrupt 12-8
SLDT instruction 10-25
sleep bit, VGA 3-24
SMSW instruction 10-25
software interrupts 12-7
software reset 6-30
Spain keyboard 5-16
speaker
display 7-3, 7-4
I/O Controllers 3-1
speaker data enable 3-166
special handling 5-48
specifications
diskette drive 8-29
displays 7-6

specifications *(continued)*
fixed disk drive 8-26
fixed disk drive controller 8-26
keyboard 5-51
system unit 1-8
specify block 8-12
specify command format 3-126
speed threshold 6-24
split screen, creating 3-99
square wave 2-38
start address high register 3-72
start address low register 3-72
start horizontal blanking register 3-66
start horizontal retrace pulse register 3-67
start vertical blanking register 3-75
startup, multitasking 12-13
status
port (parallel) 3-152
register A (diskette) 3-112
register A (RTC/CMOS) 3-160
register B (diskette) 3-112
register B (RTC/CMOS) 3-161
register C (RTC/CMOS) 3-162
register D (RTC/CMOS) 3-162
register 0 (diskette) 3-128
register 1 (diskette) 3-129
register 2 (diskette) 3-130
register 3 (diskette) 3-130
register (DMA) 2-26
registers 8-6
request 6-7
STC instruction 10-23
STD instruction 10-23
STI instruction 10-23, 12-9
STOS instruction 10-16
STR instruction 10-25
stream mode 6-3, 6-6
stream, data 5-39
string manipulation instructions 10-16
SUB instruction 10-9
subsystems
audio 2-42, A-26
diskette drive 3-111
keyboard/mouse 3-3
subsystems *(continued)*
memory 3-156
parallel port 3-149
serial port 3-135
video 3-14
support logic 3-18
supported modes 6-31
swap user interrupt vector 6-24
Sweden keyboard 5-9
Switzerland (French) keyboard 5-17
Switzerland (German) keyboard 5-18
sync polarity 3-54, 3-105
system
address map 1-7
clock (CLK) 2-8
compatibility 12-3
control port A 3-167
control port B 3-165
description 1-1
ports, miscellaneous 3-165
receiving data 3-10
request key 5-49
reset 2-28, 5-48
sending data 3-11
timer 2-32
timer modes 2-36
timers 2-31
unit 1-3
unit specifications 1-8
system board
block diagram 1-6
compatibility 12-3
connector locations 1-5
diskette drive controller 3-111
features 1-4
I/O address map 1-7
keyboard/mouse controller 3-3
locations 1-5
memory 3-156
memory map 3-157
memory option connector 3-157
parallel controller 3-149
power supply connectors 4-5
serial port controller 3-135
voltage interchange 3-148

T

tables
  extended codes, keyboard 5-45
  parameter 12-10
  scan code set 1 5-29, 5-30
  scan code set 2 5-32
  scan code set 3 5-36
terminal count (TC) 2-10
TEST instruction 10-14
test, basic assurance (BAT) 5-22
text cursor 6-21
time outs 12-15
time status indicator 3-163
timer block diagram 2-31
timer modes
  mode 0 - interrupt on terminal count 2-36
  mode 1 - hardware retriggerable one-shot 2-37
  mode 2 - rate generator 2-37
  mode 3 - square wave 2-38
  mode 4 - software retriggerable strobe 2-40
  mode 5 - hardware retriggerable strobe 2-40
  operations common to all 2-41
timers
  See also counters
  channel 0 - system timer 2-32
  channel 2 - tone generation 2-32
  control registers 2-32
  count register 2-32
  DMA operation 2-17
  interrupt level 2-29
  I/O channel 2-11
  programming of 2-32
  registers 2-34
  square wave 2-38
  system 2-31
  system timer modes 2-36
timing
  display connector (SYNC signals) 3-105
  mouse/system 3-10
  parallel port controller 3-154
timing *(continued)*
  video 3-105
tone generation 2-32
tracking speed 6-33
transfer count registers (DMA) 2-24
transmission 6-10
transmitter characteristics 9-20
transmitter holding register (serial) 3-138
typematic delay 5-21
typematic keys 5-21
typematic rate 5-21, 5-26

U

underline location register 3-74
updates, full-screen 3-59
U.K. keyboard 5-19
U.S. (English) keyboard 5-20

V

valid RAM bit 3-162
VERR instruction 10-26
vertical
  display enable end register 3-74
  gain 3-54
  interrupt 3-73
  modes 7-4
  position 6-19
  retrace end register 3-73
  retrace start register 3-72
  size 3-105
  total register 3-68
VGA components
  attribute controller 3-21
  CRT controller 3-18
  enable 3-57
  enable register 3-57
  graphics controller 3-19
  sequencer 3-19
  sleep bit 3-24
VGA (video graphics array) 6-31
video
  block diagram, attribute controller 3-21

video *(continued)*
block diagram, graphics controller 3-20
block diagram, subsystem 3-17
digital-to-analog converter 3-101
display support 3-23
memory read operations 3-50
memory write operations 3-52
modes 3-22
programmable option select 3-24
video DAC
device operation 3-101
display connector 3-110
display connector timing (SYNC signals) 3-105
programming considerations 3-103
system microprocessor interface 3-102
video graphics array
BIOS ROM 3-18
components 3-17, 3-18
description 3-14
programming 3-93
support logic 3-18
video memory organization
mode hex D 3-41
mode hex E 3-43
mode hex F 3-45
mode hex 10 3-46
mode hex 11 3-47
mode hex 12 3-48
mode hex 13 3-49
mode hex 6 3-39
mode hex 7 3-40
modes hex 0, 1 3-36
modes hex 2, 3 3-37
modes hex 4, 5 3-38
video registers
attribute address 3-87
attribute mode control 3-89
bit mask 3-86
character map select 3-61
clocking mode 3-59
color compare 3-81
video registers *(continued)*
color don't care 3-86
color plane enable 3-91
color select 3-93
CRT controller address 3-65
CRT controller overflow 3-68
CRTC mode control 3-76
cursor end 3-71
cursor location high 3-72
cursor location low 3-72
cursor start 3-70
data rotate 3-81
enable set/reset 3-80
end horizontal blanking 3-66
end horizontal retrace 3-67
end vertical blanking 3-75
feature control 3-57
graphics address 3-79
graphics mode 3-83
horizontal display enable end 3-66
horizontal PEL panning 3-92
horizontal total 3-65
input status register 0 3-56
input status register 1 3-56
line compare 3-79
map mask 3-60
maximum scan line 3-70
memory mode 3-62
miscellaneous 3-84
miscellaneous output 3-54
offset 3-74
overscan color 3-90
palette 3-88
preset row scan 3-69
programming 3-96
read map select 3-82
reset 3-58
sequencer address 3-58
set/reset 3-80
start address high 3-72
start address low 3-72
start horizontal blanking 3-66
start horizontal retrace pulse 3-67
start vertical blanking 3-75
underline location 3-74

video registers *(continued)*
vertical display enable end 3-74
vertical retrace end 3-73
vertical retrace start 3-72
vertical total 3-68
VGA enable 3-57
voltages
diskette drive controller 3-134
interchange information 9-21
keyboard and mouse controller 3-13
overvoltage fault 4-3
parallel port 3-155
power supply output 4-3
sequencing 4-4
serial port controller 9-22

W

wait condition 12-13
WAIT instruction 10-24
wait loop classes 12-14
wait state 2-4
wait state, 0WS (I) 2-10
wrap mode 6-3, 6-6
write
current, diskette 12-5
data 8-21
data command format 3-123
deleted-data command format 3-124
extended 8-22
I/O channel (-IOW) 2-9
memory command (-MEMW) 2-9
verify 8-21

X

XCHG instruction 10-5
XLAT instruction 10-6
XOR instruction 10-15
xy/PEL ratio 6-23

Numerics

80286 anomalies 12-9
80286 microprocessor 2-3
80286 microprocessor instruction set 10-1
80286 performance 2-4

International Business Machines Corporation
Printed in USA
P/N 64F0498
4/90