IBM

Personal Computer Hardware Reference Library

Technical Reference

Options and Adapters

Volume 2

6137806

Personal Computer Hardware Reference Library

Technical Reference

Options and Adapters

Volume 2

Revised Edition (April 1984)

The following paragraph does not apply to the United Kingdom or any country where such provisions are inconsistent with local law: International Business Machines Corporation provides this manual "as is," without warranty of any kind, either expressed or implied, including, but not limited to the particular purpose. IBM may make improvements and/or changes in the product(s) and/or the program(s) described in this manual at any time.

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

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

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

The following paragraph applies only to the United States and Puerto Rico: A Reader's Comment Form is provided at the back of this publication. If the form has been removed, address comments to: IBM Corp., Personal Computer, P.O. Box 1328-C, Boca Raton, Florida 33432. IBM may use or distribute any of the information you supply in any way it believes appropriate without incurring any obligations whatever.

Federal Communications Commission Radio Frequency Interference Statement

Warning: The equipment described herein has been certified to comply with the limits for a Class B computing device, pursuant to Subpart J of Part 15 of the FCC rules. Only peripherals (computer input/output devices, terminals, printers, etc.) certified to comply with the Class B limits may be attached to the computer. Operation with non-certified peripherals is likely to result in interference to radio and TV reception. If peripherals not offered by IBM are used with the equipment, it is suggested to use shielded grounded cables with in-line filters if necessary.

CAUTION
The product described herein is equipped with a grounded plug for the user's safety. It is to be used in conjunction with a properly grounded receptacle to avoid electrical shock.

Personal Computer Hardware Reference Library

IBM Monochrome Display and Printer Adapter

6361511

IBM Monochrome Display and Printer Adapter

Contents

Introduction

The IBM Monochrome Display and Printer Adapter has two functions. The first is to provide an interface to the IBM Monochrome Display. The second is to provide a parallel interface for the IBM Printers. We will discuss this adapter by function.

Monochrome Display Adapter Function

Description

The IBM Monochrome Display and Printer Adapter is designed around the Motorola 6845 CRT Controller module. There are 4K bytes of RAM on the adapter that are used for the display buffer. This buffer has two ports to which the system unit's microprocessor has direct access. No parity is provided on the display buffer.

Two bytes are fetched from the display buffer in 553 ns, providing a data rate of 1.8M bytes/second.

The adapter supports 256 different character codes. An 8K-byte character generator contains the fonts for the character codes. The characters, values, and screen characteristics are given in "Of Characters, Keystrokes, and Colors" in your *Technical Reference* system manual.

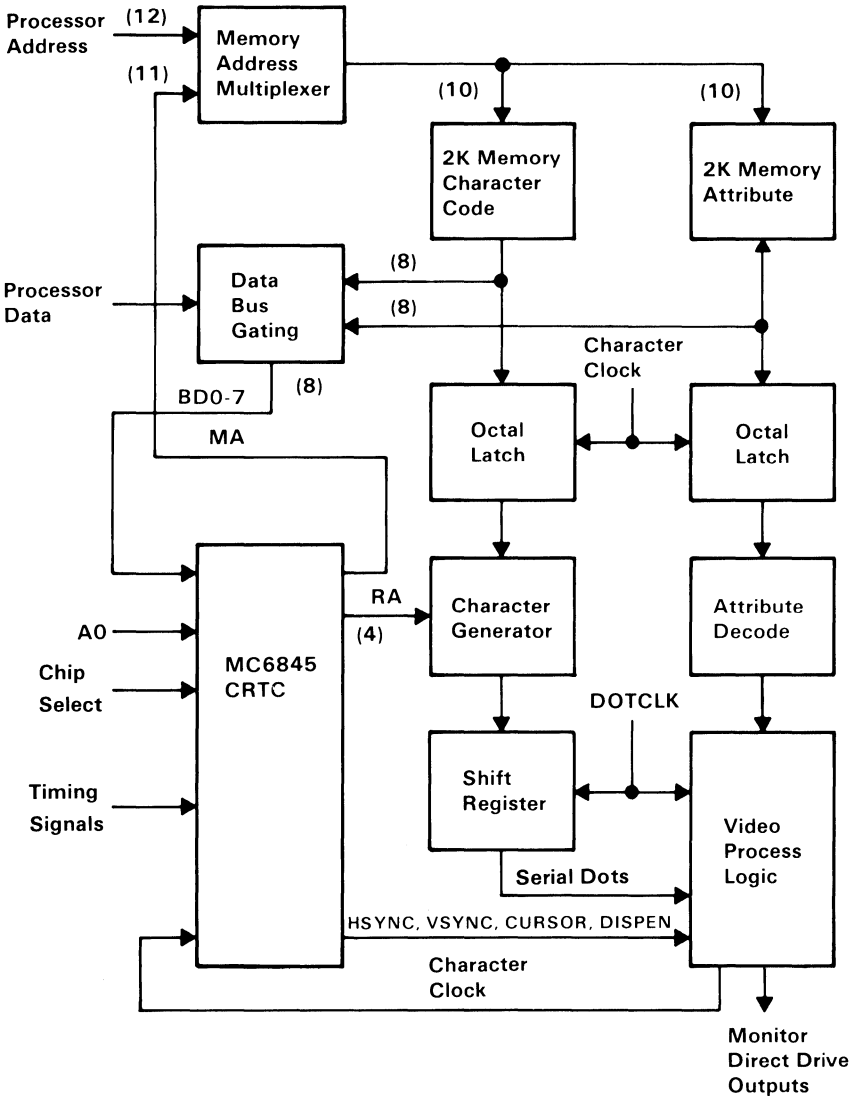
This adapter, when used with a display containing P39 phosphor, does not support a light pen.

Where possible, only one low-power Schottky (LS) load is present on any I/O slot. Some of the address bus lines have two LS loads. No signal has more than two LS loads.

Characteristics of the adapter are:

- Supports 80-character by 25-line screen
- Has direct-drive output
- Supports 9-PEL by 14-PEL character box
- Supports 7-PEL by 9-PEL character
- Has 18-kHz monitor
- Has character attributes

The following is a block diagram of the monochrome display adapter portion of the IBM Monochrome Display and Printer Adapter.



IBM Monochrome Display Adapter Block Diagram

4 Monochrome Adapter

Programming Considerations

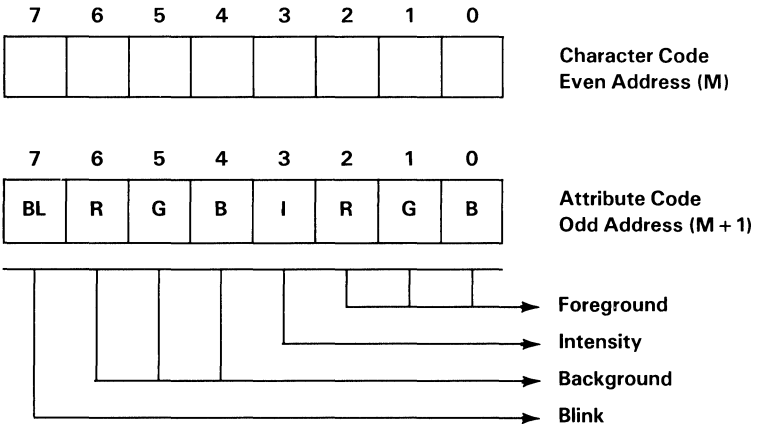
The following table summarizes the 6845 controller module's internal data registers, their functions, and their parameters. For the IBM Monochrome Display, the values must be programmed into the 6845 to ensure proper initialization of the display.

Register Number	Register File	Program Unit	IBM Monochrome Display (Address in hex)
R0	Horizontal Total	Characters	61
R1	Horizontal Displayed	Characters	50
R2	Horizontal Sync Position	Characters	52
R3	Horizontal Sync Width	Characters	F
R4	Vertical Total	Character Rows	19
R5	Vertical Total Adjust	Scan Line	6
R6	Vertical Displayed	Character Row	19
R7	Vertical Sync Position	Character Row	19
R8	Interlace Mode	---	02
R9	Maximum Scan Line Address	Scan Line	D
R10	Cursor Start	Scan Line	B
R11	Cursor End	Scan Line	C
R12	Start Address (H)	---	00
R13	Start Address (L)	---	00
R14	Cursor (H)	---	00
R15	Cursor (L)	---	00
R16	Reserved	---	--
R17	Reserved	---	--

To ensure proper initialization, the first command issued to the IBM Monochrome Display and Printer Adapter must be sent to the CRT control port 1 (hex 3B8), and must be a hex 01, to set the high-resolution mode. If this bit is not set, the system unit's microprocessor's access to the adapter must never occur. If the high-resolution bit is not set, the system unit's microprocessor will stop running.

System configurations that have both an IBM Monochrome Display and Printer Adapter, and an IBM Color/Graphics Monitor Adapter, must ensure that both adapters are properly initialized after a power-on reset. Damage to either display may occur if not properly initialized.

The IBM Monochrome Display and Printer Adapter supports 256 different character codes. In the character set are alphanumerics and block graphics. Each character in the display buffer has a corresponding character attribute. The character code must be an even address, and the attribute code must be an odd address in the display buffer.



The adapter decodes the character attribute byte as defined above. The blink and intensity bits may be combined with the foreground and background bits to further enhance the character attribute functions listed below:

Background R G B	Foreground R G B	Function
0 0 0	0 0 0	Non-Display
0 0 0	0 0 1	Underline
0 0 0	1 1 1	White Character/Black Background
1 1 1	0 0 0	Reverse Video

The 4K display buffer supports one screen of the 25 rows of 80 characters, plus a character attribute for each display character. The starting address of the buffer is hex B0000. The display buffer can be read using direct memory access (DMA); however, at least one wait state will be inserted by the system unit's microprocessor. The duration of the wait state will vary, because the microprocessor/monitor access is synchronized with the character clock on this adapter.

Interrupt level 7 is used on the parallel interface. Interrupts can be enabled or disabled through the printer control port. The interrupt is a high-level active signal.

The following table breaks down the functions of the I/O address decode for the adapter. The I/O address decode is from hex 3B0 through hex 3BF. The bit assignment for each I/O address follows:

I/O Register Address	Function
3B0	Not Used
3B1	Not Used
3B2	Not Used
3B3	Not Used
3B4	6845 Index Register
3B5	6845 Data Register
3B6	Not Used
3B7	Not Used
3B8	CRT Control Port 1
3B9	Reserved
3BA	CRT Status Port
3BB	Reserved
3BC	Parallel Data Port
3BD	Printer Status Port
3BE	Printer Control Port
3BF	Not Used

I/O Address and Bit Map

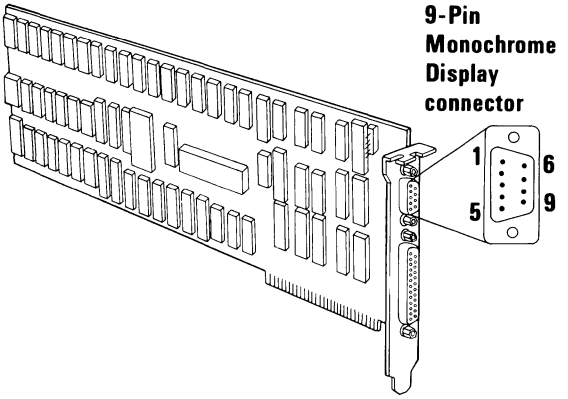
Bit Number	Function
0	+ High Resolution Mode
1	Not Used
2	Not Used
3	+ Video Enable
4	Not Used
5	+ Enable Blink
6,7	Not Used

6845 CRT Control Port 1 (Hex 3B8)

Bit Number	Function
0	+ Horizontal Drive
1	Reserved
2	Reserved
3	+ Black/White Video

6845 CRT Status Port (Hex 3BA)

Specifications



At Standard TTL Levels

IBM Monochrome Display	Ground	1	IBM Monochrome Display and Printer Adapter
	Ground	2	
	Not Used	3	
	Not Used	4	
	Not Used	5	
	+ Intensity	6	
	+ Video	7	
	+ Horizontal	8	
	– Vertical	9	

Note: Signal voltages are 0.0 to 0.6 Vdc at down level and + 2.4 to 3.5 Vdc at high level.

Connector Specifications

10 Monochrome Adapter

Printer Adapter Function

Description

The printer adapter portion of the IBM Monochrome Display and Printer Adapter is specifically designed to attach printers with a parallel-port interface, but it can be used as a general input/output port for any device or application that matches its input/output capabilities. It has 12 TTL-buffer output points, which are latched and can be written and read under program control using the microprocessor In or Out instruction. The adapter also has five steady-state input points that may be read using the microprocessor's In instructions.

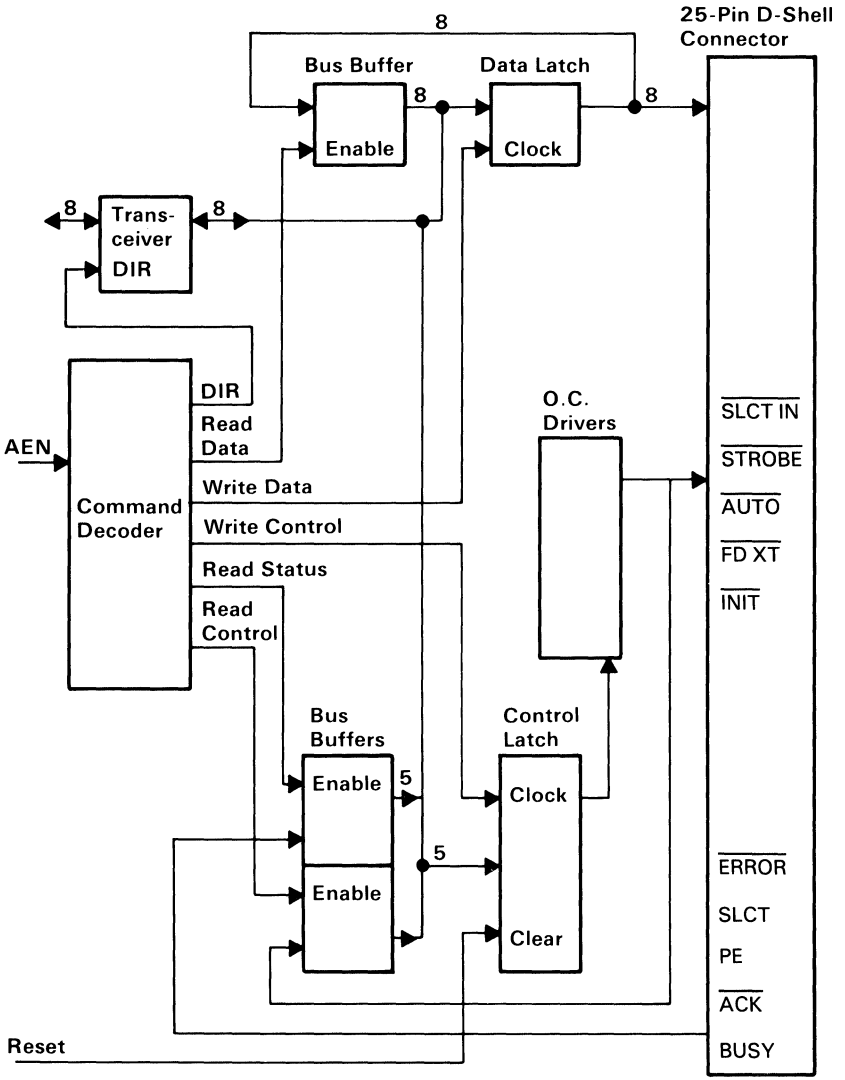
In addition, one input can also be used to create a microprocessor interrupt. This interrupt can be enabled and disabled under program control. A reset from the power-on circuit is also ORed with a program output point, allowing a device to receive a 'power-on reset' when the system unit's microprocessor is reset.

The input/output signals are made available at the back of the adapter through a right-angle, printed-circuit-board-mounted, 25-pin, D-shell connector. This connector protrudes through the rear panel of the system unit or expansion unit, where a cable may be attached.

When this adapter is used to attach a printer, data or printer commands are loaded into an 8-bit, latched, output port, and the strobe line is activated, writing data to the printer. The program then may read the input ports for printer status indicating when the next character can be written, or it may use the interrupt line to indicate "not busy" to the software.

The output ports may also be read at the card's interface for diagnostic loop functions. This allows faults to be isolated to the adapter or the attaching device.

The following is a block diagram of the printer adapter portion of the Monochrome Display and Printer Adapter.



Printer Adapter Block Diagram

Programming Considerations

The printer adapter portion of the IBM Monochrome Display and Printer Adapter responds to five I/O instructions: two output and three input. The output instructions transfer data into 2 latches whose outputs are presented on pins of a 25-pin D-shell connector.

Two of the three input instructions allow the system unit's microprocessor to read back the contents of the two latches. The third allows the system unit's microprocessor to read the real-time status from a group of pins on the connector.

A description of each instruction follows.

IBM Monochrome Display & Printer Adapter			
Output to address hex 3BC			
Bit 7 Pin 9	Bit 6 Pin 8	Bit 5 Pin 7	Bit 4 Pin 6

The instruction captures data from the data bus and is present on the respective pins. Each of these pins is capable of sourcing 2.6 mA and sinking 24 mA.

It is essential that the external device does not try to pull these lines to ground.

IBM Monochrome Display & Printer Adpater	
Output to address hex 3BE	
	Bit 4 IRQ Enable

This instruction causes the latch to capture the five least significant bits of the data bus. The four least significant bits present their outputs, or inverted versions of their outputs, to the

respective pins as shown in the previous figure. If bit 4 is written as a 1, the card will interrupt the system unit's microprocessor on the condition that pin 10 changes from high to low.

These pins are driven by open-collector drivers pulled to +5 Vdc through 4.7 kΩ resistors. They can each sink approximately 7 mA and maintain 0.8 volts down-level.

IBM Monochrome Display & Printer Adapter
Input from address hex 3BC

This instruction presents the system unit's microprocessor with data present on the pins associated with the output to hex 3BC. This should normally reflect the exact value that was last written to hex 3BC. If an external device should be driving data on these pins at the time of an input (in violation of usage ground rules), this data will be ORed with the latch contents.

IBM Monochrome Display & Printer Adapter
Input from address hex 3BD

This instruction presents the real-time status to the system unit's microprocessor from the pins as follows.

Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
Pin 11	Pin 10	Pin 12	Pin 13	Pin 15	—	—	—

IBM Monochrome Display & Printer Adapter
Input from address hex 3BE

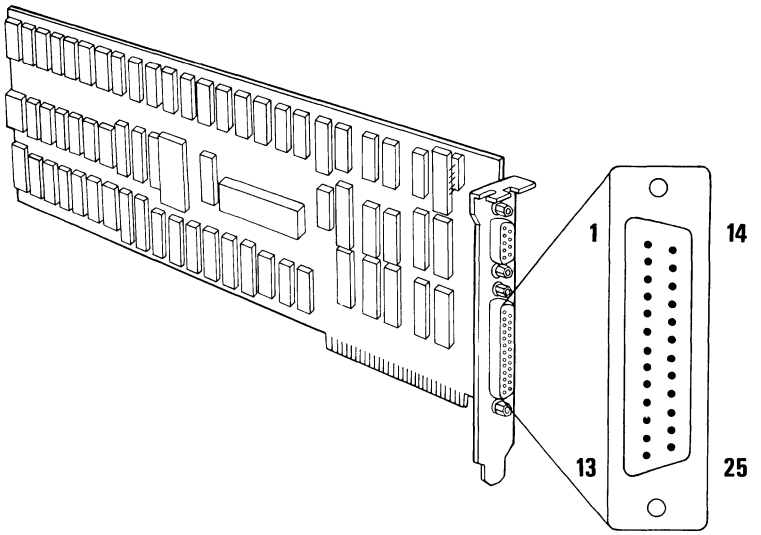
This instruction causes the data present on pins 1, 14, 16, 17, and the IRQ bit to be read by the system unit's microprocessor. In the absence of external drive applied to these pins, data read by the system unit's microprocessor will match data last written to hex 3BE in the same bit positions. Notice that data bits 0–2 are not included. If external drivers are dotted to these pins, that data will be ORed with data applied to the pins by the hex 3BE latch.

Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
			IRQ Enable	$\overline{\text{Pin 17}}$	Pin 16	$\overline{\text{Pin 14}}$	$\overline{\text{Pin 1}}$
			Por = 0	Por = 1	Por = 0	Por = 1	Por = 1

These pins assume the states shown after a reset from the system unit's microprocessor.

16 Monochrome Adapter

Specifications



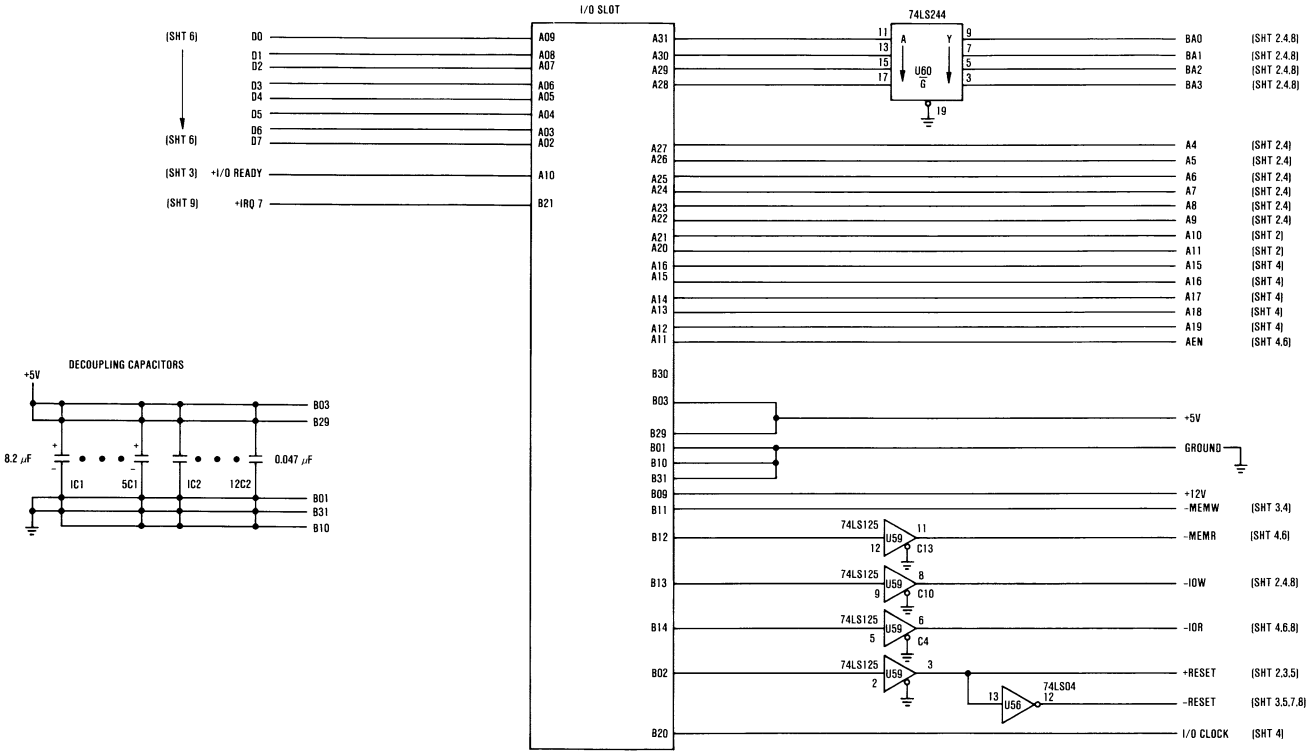
At Standard TTL Levels

Signal Name	Adapter Pin Number	
– Strobe	1	
+ Data Bit 0	2	
+ Data Bit 1	3	
+ Data Bit 2	4	
+ Data Bit 3	5	
+ Data Bit 4	6	
+ Data Bit 5	7	IBM
+ Data Bit 6	8	Monochrome
+ Data Bit 7	9	Display and
– Acknowledge	10	Printer
+ Busy	11	Adapter
+ P. End (out of paper)	12	
+ Select	13	
– Auto Feed	14	
– Error	15	
– Initialize Printer	16	
– Select Input	17	
Ground	18-25	

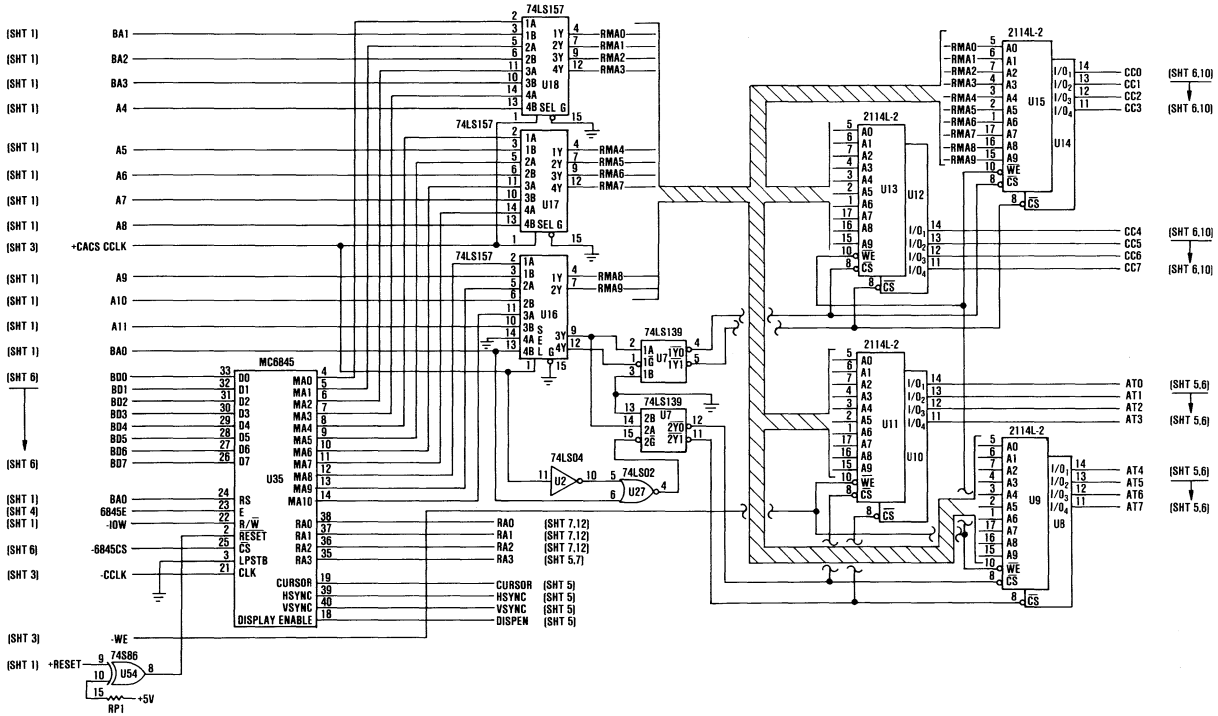
Printer

Connector Specifications

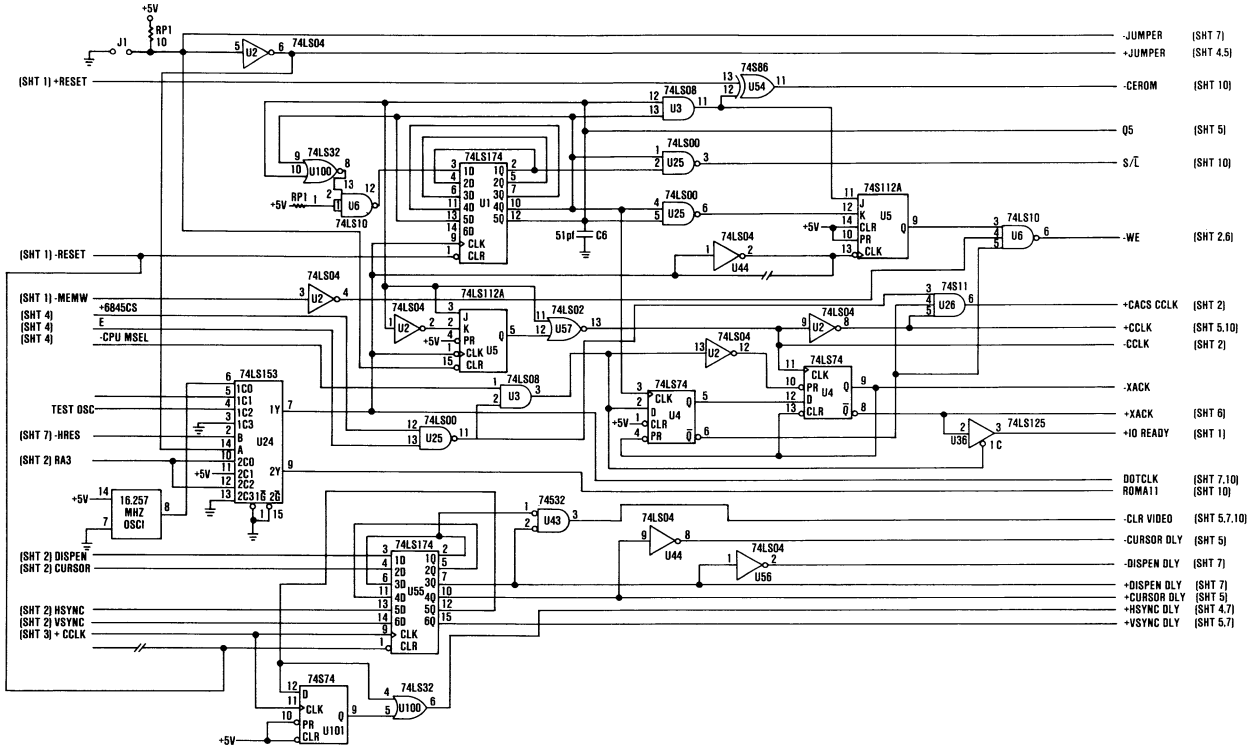
18 Monochrome Adapter



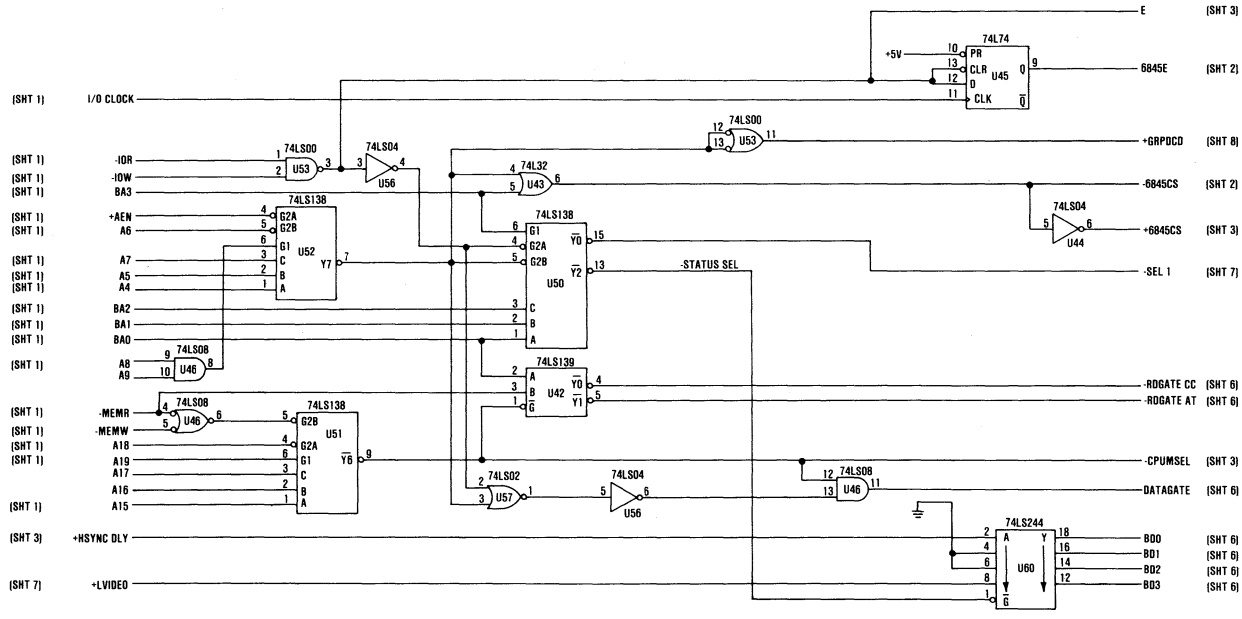
Monochrome Display Adapter (Sheet 1 of 10)



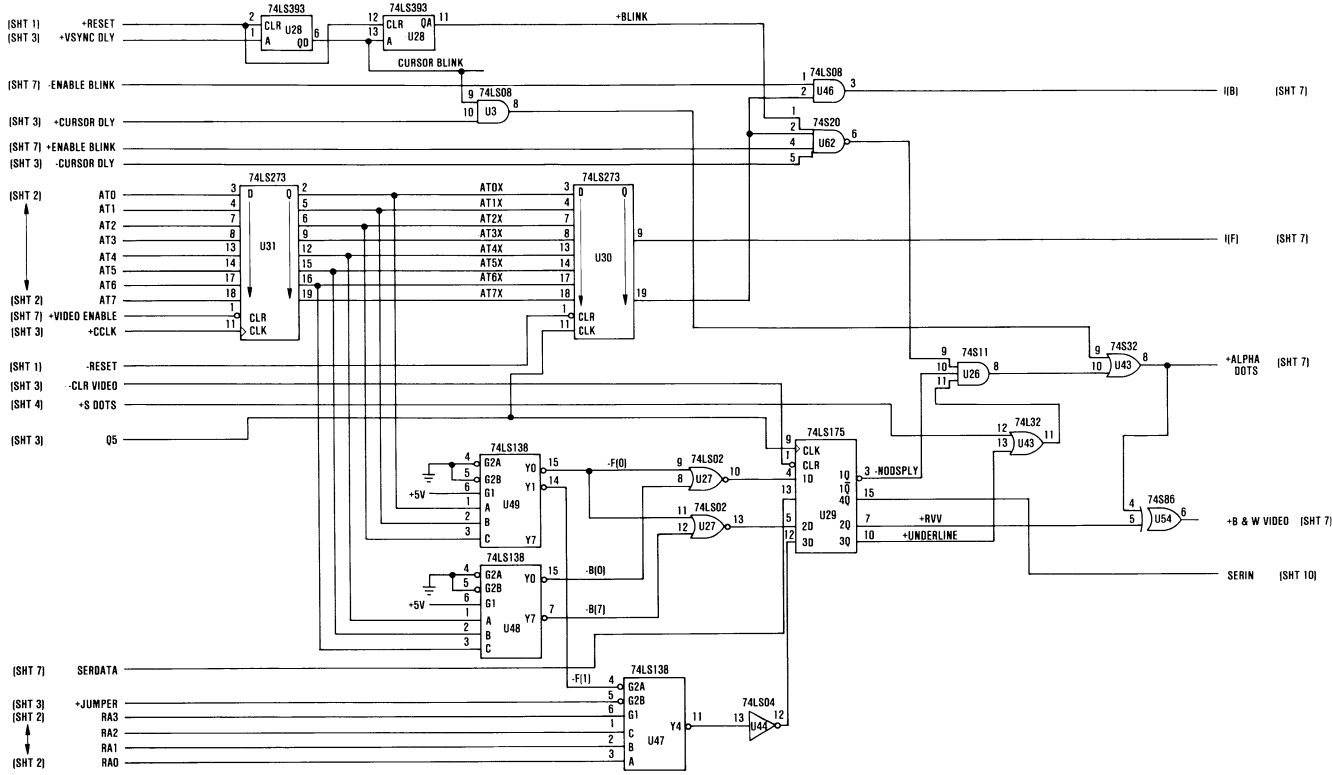
Monochrome Display Adapter (Sheet 2 of 10)



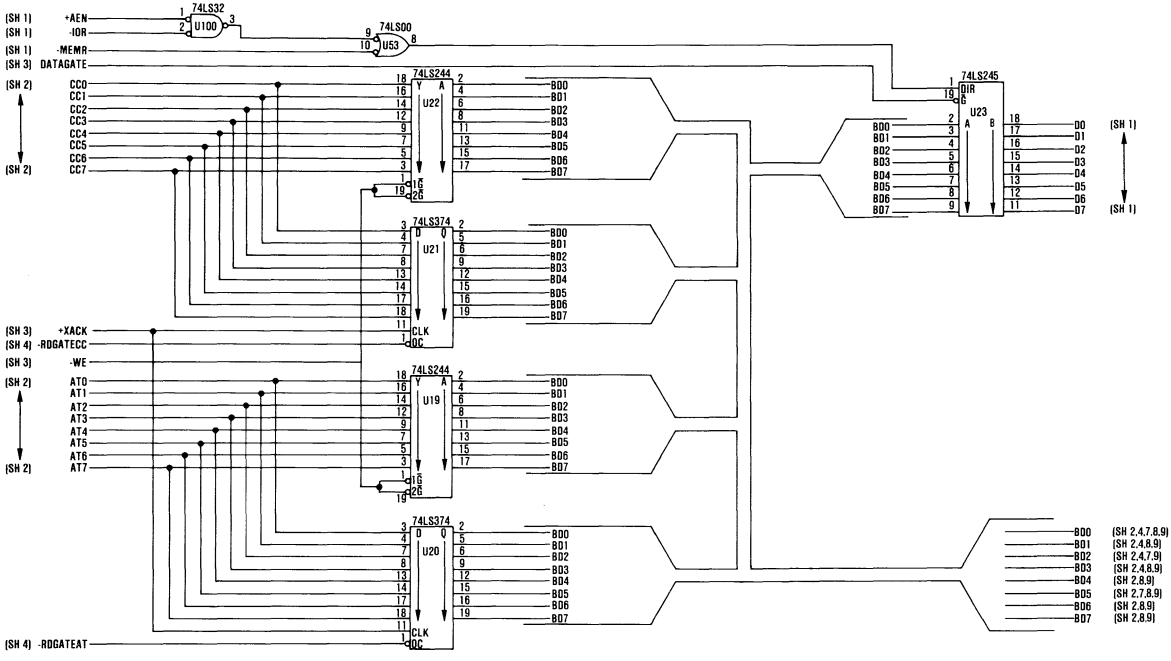
Monochrome Display Adapter (Sheet 3 of 10)



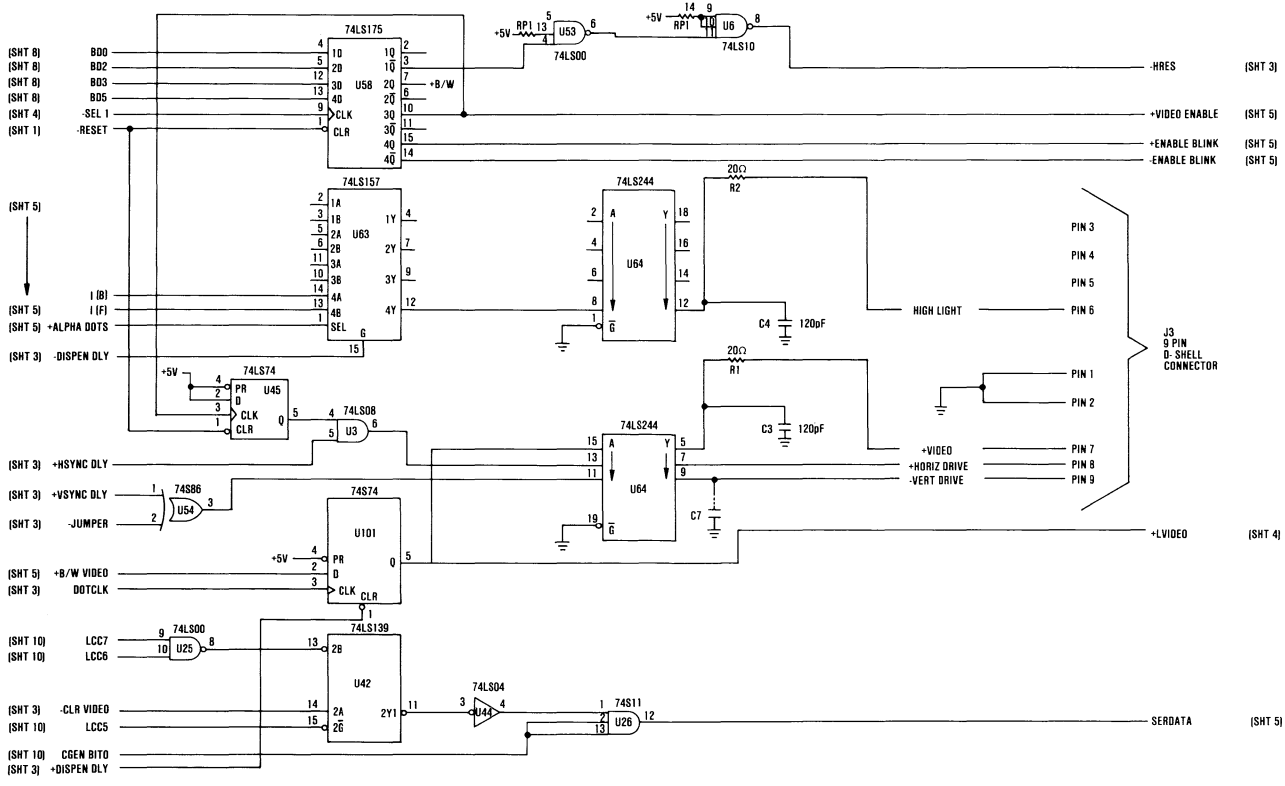
Monochrome Display Adapter (Sheet 4 of 10)



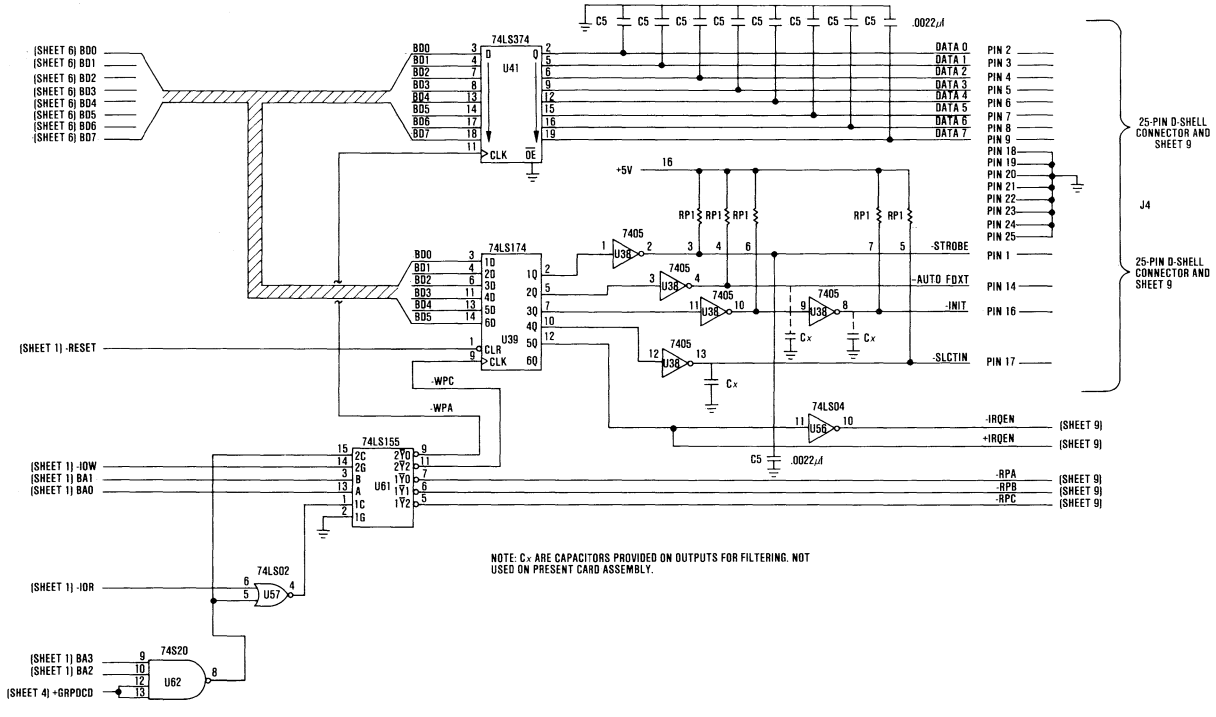
Monochrome Display Adapter (Sheet 5 of 10)



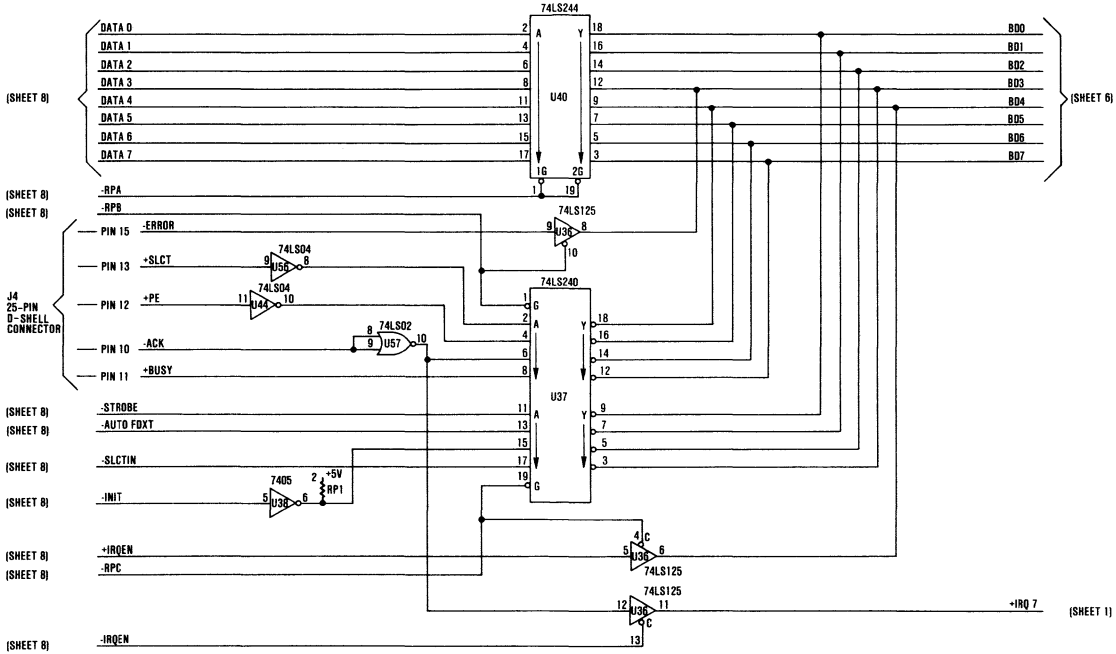
Monochrome Display Adapter (Sheet 6 of 10)



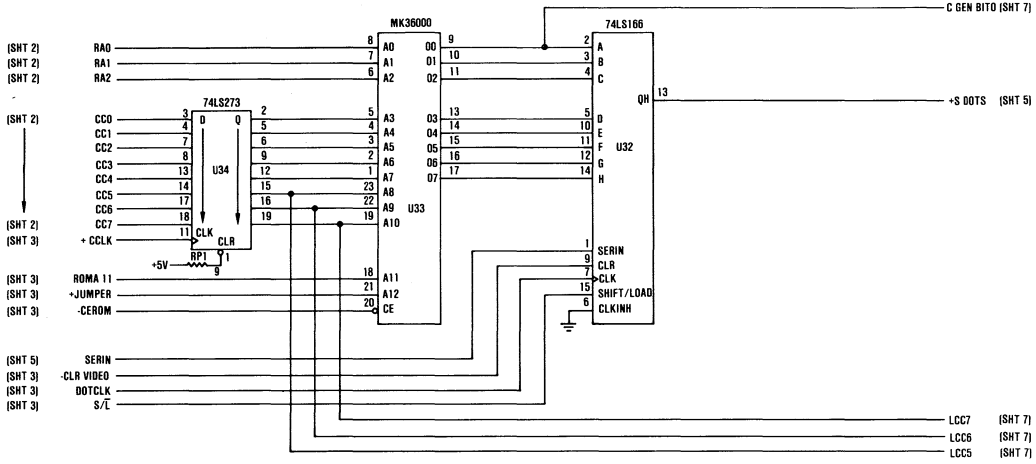
Monochrome Display Adapter (Sheet 7 of 10)



Monochrome Display Adapter (Sheet 8 of 10)



Monochrome Display Adapter (Sheet 9 of 10)



Monochrome Display Adapter (Sheet 10 of 10)

Personal Computer
Hardware Reference
Library

IBM Color/Graphics Monitor Adapter

6361509

Contents

Description

The IBM Color/Graphics Monitor Adapter is designed to attach to the IBM Color Display, to a variety of television-frequency monitors, or to home television sets (user-supplied RF modulator is required for home television sets). The adapter is capable of operating in black-and-white or color. It provides three video interfaces: a composite-video port, a direct-drive port, and a connection interface for driving a user-supplied RF modulator. A light pen interface is also provided.

The adapter has two basic modes of operation: alphanumeric (A/N) and all-points-addressable (APA) graphics. Additional modes are available within the A/N or APA graphics modes. In the A/N mode, the display can be operated in either a 40-column by 25-row mode for a low-resolution monitor or home television, or in an 80-column by 25-row mode for high-resolution monitors. In both modes, characters are defined in an 8-wide by 8-high character box and are 7-wide by 7-high, double dotted characters with one descender. Both uppercase and lowercase characters are supported in all modes.

The character attributes of reverse video, blinking, and highlighting are available in the black-and-white mode. In the color mode, 16 foreground and 8 background colors are available for each character. In addition, blinking on a per-character basis is available.

The monitor adapter contains 16K bytes of storage. As an example, a 40-column by 25-row display screen uses 1000 bytes to store character information and 1000 bytes to store attribute/color information. This means that up to eight screens can be stored in the adapter memory. Similarly, in an 80-wide by 25-row mode, four display screens can be stored in the adapter memory. The entire 16K bytes of storage in the display adapter are directly accessible by the processor, which allows maximum program flexibility in managing the screen.

In A/N color modes, it is also possible to select the color of the screen's border. One of 16 colors can be selected.

In the APA graphics mode, there are two resolutions available: a medium-resolution color graphics mode (320 PELs by 200 rows) and a high-resolution black-and-white graphics mode (640 PELs by 200 rows). In the medium-resolution mode, each picture element (PEL) may have one of four colors. The background color (Color 0) may be any of the 16 possible colors. The remaining three colors come from one of the two program-selectable palettes. One palette contains green/red/brown; the other contains cyan/magenta/white.

The high-resolution mode is available only in black-and-white because the entire 16K bytes of storage in the adapter is used to define the on or off state of the PELs.

The adapter operates in noninterlace mode at either 7 or 14 MHz, depending on the mode of the operation selected.

In the A/N mode, characters are formed from a ROS character generator. The character generator contains dot patterns for 256 different characters. The character set contains the following major groupings of characters.

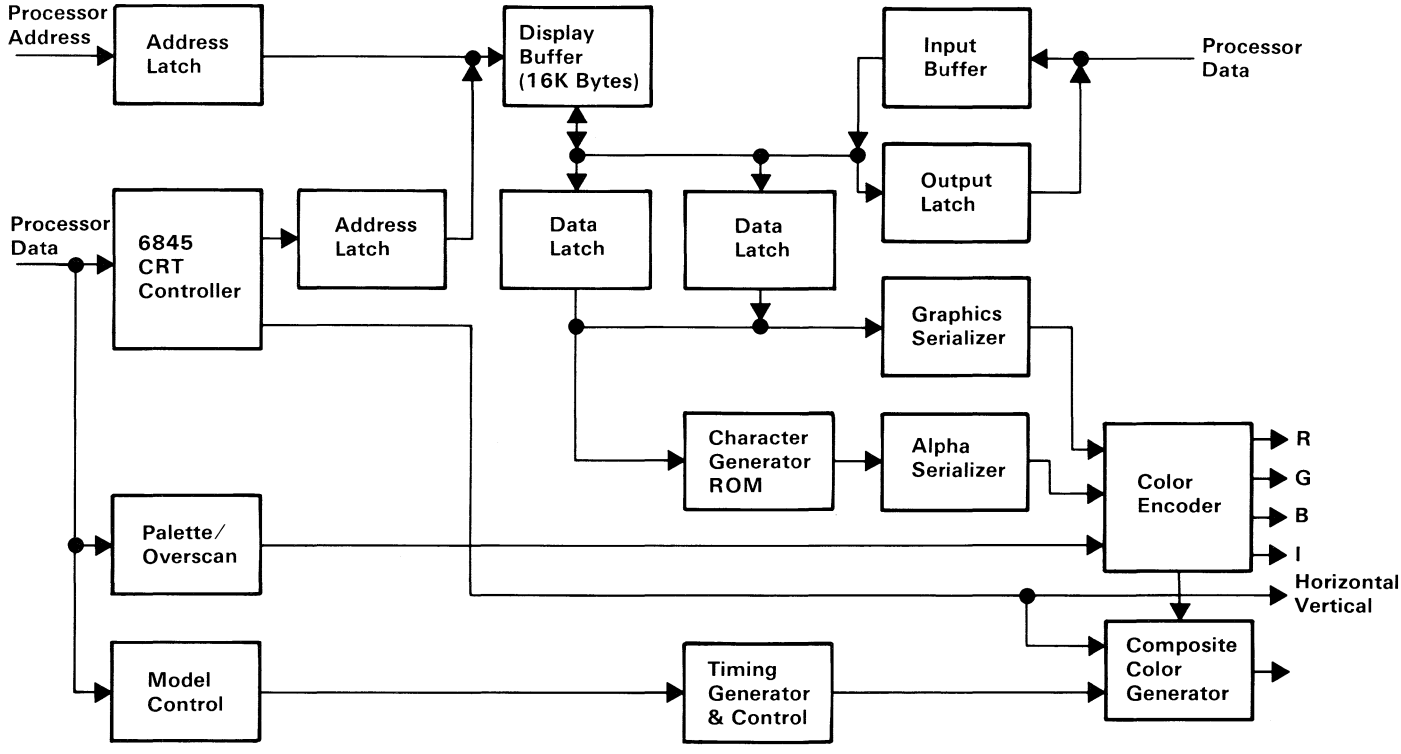
- 16 special characters for game support
- 15 characters for word-processing editing support
- 96 characters for the standard ASCII graphics set
- 48 characters for foreign-language support
- 48 characters for business block-graphics support (for the drawing of charts, boxes, and tables using single and double lines)
- 16 selected Greek characters
- 15 selected scientific-notation characters

The color/graphics monitor function is on a single adapter. The direct-drive and composite-video ports are right-angle mounted connectors on the adapter, and extend through the rear panel of

the system unit. The direct-drive video port is a 9-pin, D-shell, female connector. The composite-video port is a standard female phono jack.

The display adapter uses a Motorola 6845 CRT Controller device. This adapter is highly programmable with respect to raster and character parameters. Therefore, many additional modes are possible with programming of the adapter.

On the following page is a block diagram of the Color/Graphics Monitor Adapter.



Color/Graphics Monitor Adapter Block Diagram

Controller

The controller is a Motorola 6845 Cathode Ray Tube (CRT) Controller. It provides the necessary interface to drive the raster-scan CRT.

Mode Set Register

The mode set register is a general-purpose, programmable, I/O register. It has I/O ports that may be individually programmed. Its function in this adapter is to provide mode selection and color selection in the medium-resolution color-graphics mode.

Display Buffer

The display buffer resides in the processor-address space, starting at address hex B8000. It provides 16 bytes of dynamic read/write memory. A dual-ported implementation allows the processor and the graphics control unit access to this buffer. The processor and the control unit have equal access to this buffer during all modes of operation, except in the high-resolution alphanumeric mode. In this mode, only the processor should have access to this buffer during the horizontal-retrace intervals. While the processor may write to the required buffer at any time, a small amount of display interference will result if this does not occur during the horizontal-retrace intervals.

Character Generator

A ROS character generator is used with 8K bytes of storage that cannot be read from or written to under program control. This is a general-purpose ROS character generator with three character fonts. Two character fonts are used on the Color/Graphics Monitor Adapter: a 7-high by 7-wide double-dot font and a 7-high by 5-wide single-dot font. The font is selected by a jumper (P3). The single-dot font is selected by inserting the jumper; the double-dot font is selected by removing the jumper.

Timing Generator

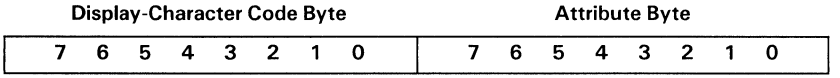
This generator produces the timing signals used by the 6845 CRT Controller and by the dynamic memory. It also solves the processor/graphic controller contentions for access to the display buffer.

Composite Color Generator

This generator produces base-band-video color information.

Alphanumeric Mode

Every display character position in the alphanumeric mode is defined by two bytes in the regen buffer (a part of the monitor adapter), not the system memory. Both the Color/Graphics Monitor Adapter and the Monochrome Display and Printer Adapter use the following 2-byte character-attribute format.

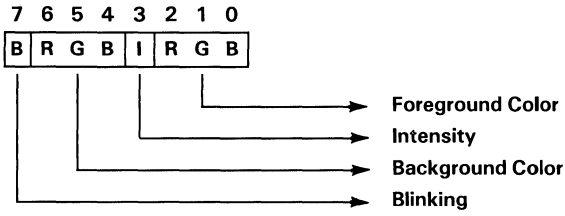


The following table shows the functions of the attribute byte.

Attribute Function	Attribute Byte							
	7	6	5	4	3	2	1	0
	B	R	G	B	I	R	G	B
	FG	Background			Foreground			
Normal	B	0	0	0	I	1	1	1
Reverse Video	B	1	1	1	I	0	0	0
Nondisplay (Black)	B	0	0	0	I	0	0	0
Nondisplay (White)	B	1	1	1	I	1	1	1

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

The definitions of the attribute byte are in the following table.



In the alphanumeric mode, the display can be operated in either a low-resolution mode or a high-resolution mode.

The low-resolution alphanumeric mode has the following features:

- Supports home color televisions or low-resolution monitors.
- Displays up to 25 rows of 40 characters each.
- Has a ROS character generator that contains dot patterns for a maximum of 256 different characters.
- Requires 2,000 bytes of read/write memory (on the adapter).
- Has an 8-high by 8-wide character box.
- Has two jumper-controlled character fonts available: a 7-high by 5-wide single-dot character font with one descender, and a 7-high by 7-wide double-dotted character font with one descender.
- Has one character attribute for each character.

The high-resolution alphanumeric mode has the following features:

- Supports the IBM Color Display or other color monitor with direct-drive capability.
- Supports a black-and-white composite-video monitor.
- Displays up to 25 rows of 80 characters each.

- Has a ROS display generator that contains dot patterns for a maximum of 256 characters.
- Requires 4,000 bytes of read/write memory (on the adapter).
- Has an 8-high by 8-wide character box.
- Has two jumper-controlled character fonts available: a 7-high by 5-wide single-dot character font with one descender, and a 7-high by 7-wide double-dot character font with one descender.
- Has one character attribute for each character.

The Color/Graphics Monitor Adapter will change foreground and background colors according to the color value selected in the attribute byte. The following figure shows the color values for the various red, green, blue, and intensity bit settings.

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

Note: Not all Monitors recognize the intensity (I) bit.

Graphics Modes

The Color/Graphics Monitor Adapter has three graphics modes: low-resolution, medium-resolution, and high-resolution color graphics. However, only medium- and high-resolution graphics are supported in ROM. The following figure shows these modes.

Mode	Horizontal (PELs)	Vertical (Rows)	Number of Colors Available (Includes Background Color)
Low Resolution	160	100	16 (Includes black-and-white)
Medium Resolution	320	200	4 Colors Total 1 of 16 for Background and 1 of Green, Red, or Brown or 1 of Cyan, Magenta, or White
High Resolution	640	200	Black-and-white only

Low-Resolution Color/Graphics Mode

The low-resolution mode supports home televisions or color monitors. This mode, not supported in ROM, has the following features:

- Contains a maximum of 160 PELs by 100 rows, with each PEL being 2-high by 2-wide.
- Specifies 1 of 16 colors for each PEL by the I, R, G, and B bits.
- Requires 16,000 bytes of read/write memory on the adapter.
- Uses memory-mapped graphics.

Medium-Resolution Color/Graphics Mode

The medium-resolution mode supports home televisions or color monitors. It has the following features:

- Contains a maximum of 320 PELs by 200 rows, with each PEL being 1-high by 1-wide.

- Preselects 1 of 4 colors for each PEL.
- Requires 16,000 bytes of read/write memory on the adapter.
- Uses memory-mapped graphics.
- Formats 4 PELs per byte in the following manner:

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

- Organizes graphics storage into two banks of 8,000 bytes, using the following format:

Memory Address (in hex)	Function
B8000	Even Scans (0,2,4,...198) 8,000 bytes
B9F3F	Not Used
BA000	Odd Scans (1,3,5...199) 8,000 Bytes
BBF3F	Not Used
BBFFF	

Address hex B8000 contains the PEL instruction for the upper-left corner of the display area.

- Color selection is determined by the following logic:

C1	C0	Function
0	0	Dot takes on the color of 1 of 16 preselected background colors
0	1	Selects first color of preselected Color Set 1 or Color Set 2
1	0	Selects second color of preselected Color Set 1 or Color Set 2
1	1	Selects third color of preselected Color Set 1 or Color Set 2

C1 and C0 select 4 of 16 preselected colors. This color selection (palette) is preloaded in an I/O port.

The two color sets are:

Color Set 1	Color Set 2
Color 1 is Green	Color 1 is Cyan
Color 2 is Red	Color 2 is Magenta
Color 3 is Brown	Color 3 is White

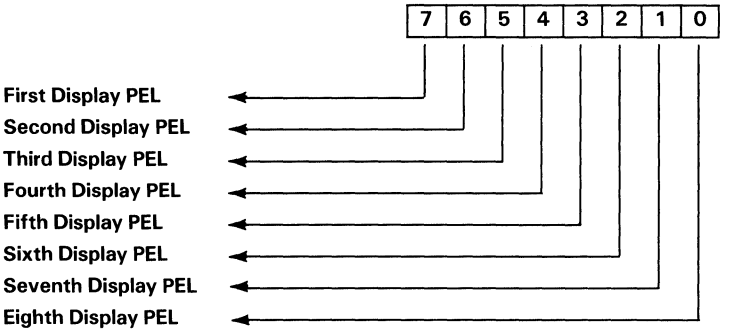
The background colors are the same basic 8 colors defined for low-resolution graphics, plus 8 alternate intensities defined by the intensity bit, for a total of 16 colors, including black and white.

High-Resolution Black-and-White Graphics Mode

The high-resolution mode supports color monitors. This mode has the following features:

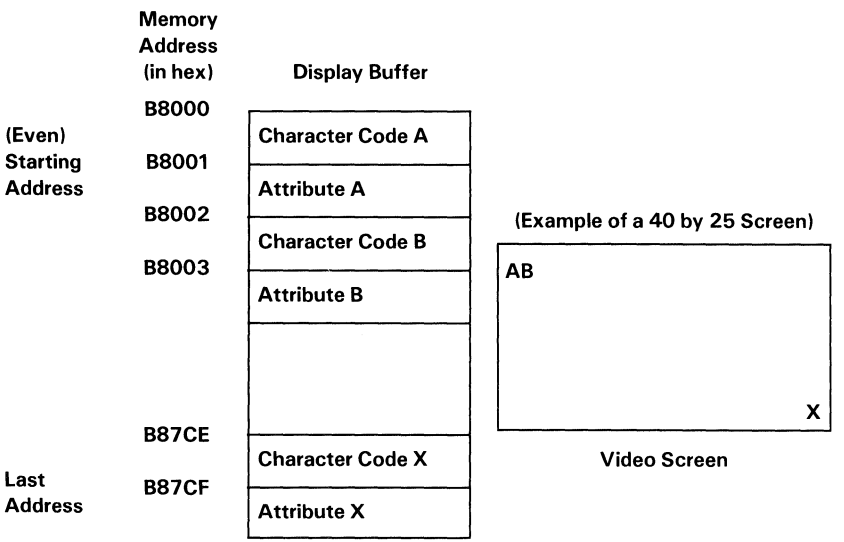
- Contains a maximum of 640 PELs by 200 rows, with each PEL being 1-high by 1-wide.
- Supports black-and-white only.
- Requires 16,000 bytes of read/write memory on the adapter.
- Addressing and mapping procedures are the same as medium-resolution color/graphics, but the data format is different. In this mode, each bit in memory is mapped to a PEL on the screen.

- Formats 8 PELs per byte in the following manner:



Basic Operations

In the alphanumeric mode, the adapter fetches character and attribute information from its display buffer. The starting address of the display buffer is programmable through the CRT controller, but it must be an even address. The character codes and attributes are then displayed according to their relative positions in the buffer. The following addresses will produce an "AB" in the upper-left corner of a 40 by 25 screen and an "X" in the lower-right corner.



The processor and the display control unit have equal access to the display buffer during all of the operating modes, except the high-resolution alphanumeric mode. During this mode, the processor gains access to the display buffer during the vertical retrace time. If it does not, the display will be affected with random patterns as the microprocessor is using the display buffer. In the alphanumeric mode, the characters are displayed from a pre-stored ROM character generator that contains the dot patterns for all of the displayable characters.

In the graphics mode, the displayed dots and colors, up to 16K bytes, are fetched from the display buffer.

Programming Considerations

Programming the Mode Control and Status Register

The following I/O devices are defined on the Color/Graphics Monitor Adapter.

Hex Address	A9	A8	A7	A6	A5	A4	A3	A2	A1	A0	Function of Register
3D8	1	1	1	1	0	1	1	0	0	0	Mode Control Register (D0)
3D9	1	1	1	1	0	1	1	0	0	1	Color Select Register (D0)
3DA	1	1	1	1	0	1	1	0	1	0	Status Register (D1)
3DB	1	1	1	1	0	1	1	0	1	1	Clear Light Pen Latch
3DC	1	1	1	1	0	1	1	1	0	0	Preset Light Pen Latch
3D4	1	1	1	1	0	1	0	Z	Z	0	6845 Index Register
3D5	1	1	1	1	0	1	0	Z	Z	1	6845 Data Register

Z = don't care condition

Programming the 6845 CRT Controller

The controller has 19 internal accessible registers, which are used to define and control a raster-scan CRT display. One of these registers, the index register, is used as a pointer to the the other 18 registers. It is a write-only register, which is loaded from the processor by executing an 'out' instruction to I/O address hex 3D4. The five least-significant bits of the I/O bus are loaded into the index register.

In order to load any of the other 18 registers, the index register is first loaded with the necessary pointer, then the data register is

loaded with the information to be placed in the selected register. The data register is loaded from the processor by executing an 'out' instruction to I/O address hex 3D5.

The table on the next page defines the values that must be loaded into the 6845 CRT Controller registers to control the different modes of operation supported by the attachment.

Address Register	Register Number	Register Type	Units	I/O	40 by 25 Alpha-numeric	80 by 25 Alpha-numeric	Graphic Modes
0	R0	Horizontal Total	Character	Write Only	38	71	38
1	R1	Horizontal Displayed	Character	Write Only	28	50	28
2	R2	Horizontal Sync Position	Character	Write Only	2D	5A	2D
3	R3	Horizontal Sync Width	Character	Write Only	0A	0A	0A
4	R4	Vertical Total	Character Row	Write Only	1F	1F	7F
5	R5	Vertical Total Adjust	Scan Line	Write Only	06	06	06
6	R6	Vertical Displayed	Character Row	Write Only	19	19	64
7	R7	Vertical Sync Position	Character Row	Write Only	1C	1C	70
8	R8	Interlace Mode	-	Write Only	02	02	02
9	R9	Maximum Scan Line Address	Scan Line	Write Only	07	07	01
A	R10	Cursor Start	Scan Line	Write Only	06	06	06
B	R11	Cursor End	Scan Line	Write Only	07	07	07
C	R12	Start Address (H)	-	Write Only	00	00	00
D	R13	Start Address (L)	-	Write Only	00	00	00
E	R14	Cursor Address (H)	-	Read/ Write	XX	XX	XX
F	R15	Cursor Address (L)	-	Read/ Write	XX	XX	XX
10	R16	Light Pen (H)	-	Read Only	XX	XX	XX
11	R17	Light Pen (L)	-	Read Only	XX	XX	XX

Note: All register values are given in hexadecimal

6845 Register Description

Color-Select Register

The color-select register is a 6-bit output-only register. Its I/O address is hex 3D9, and it can be written to using a processor 'out' command. The following are the bit definitions for this register.

Bit 0 Selects blue border color in 40 by 25 alphanumeric mode.
Selects blue background color (C0-C1) in 320 by 200 graphics mode.
Selects blue foreground color in 640 by 200 graphics mode.

Bit 1 Selects green border color in 40 by 25 alphanumeric mode.
Selects green background color (C0-C1) in 320 by 200 graphics mode.
Selects green foreground color in 640 by 200 graphics mode.

Bit 2 Selects red border color in 40 by 25 alphanumeric mode.
Selects red background color (C0-C1) in 320 by 200 graphics mode.
Selects red foreground color in 640 by 200 graphics mode.

Bit 3 Selects intensified border color in 40 by 25 alphanumeric mode.
Selects intensified background color (C0-C1) in 320 by 200 graphics mode.
Selects intensified foreground color in 640 by 200 graphics mode.

Bit 4 Selects alternate, intensified set of colors in the graphics mode.
Selects background colors in the alphanumeric mode.

Bit 5 Selects active color set in 320 by 200 graphics mode.

When bit 5 is set to 1, colors are determined as follows:

C1	C0	Set Selected
0	0	Background (Defined by bits 0-3 of port hex 3D9)
0	1	Cyan
1	0	Magenta
1	1	White

When bit 5 is set to 0, colors are determined as follows:

C1	C0	Set Selected
0	0	Background (Defined by bits 0-3 of port hex 3D9)
0	1	Green
1	0	Red
1	1	Brown

Bit 6 Not used

Bit 7 Not used

Mode-Control Register

The mode-control register is a 6-bit output-only register. Its I/O address is hex 3D8, and it can be written to using a processor 'out' command. The following are bit definitions for this register.

Bit 0 A 1 selects 80 by 25 alphanumeric mode.
A 0 selects 40 by 25 alphanumeric mode.

Bit 1 A 1 selects 320 by 200 graphics mode.
A 0 selects alphanumeric mode.

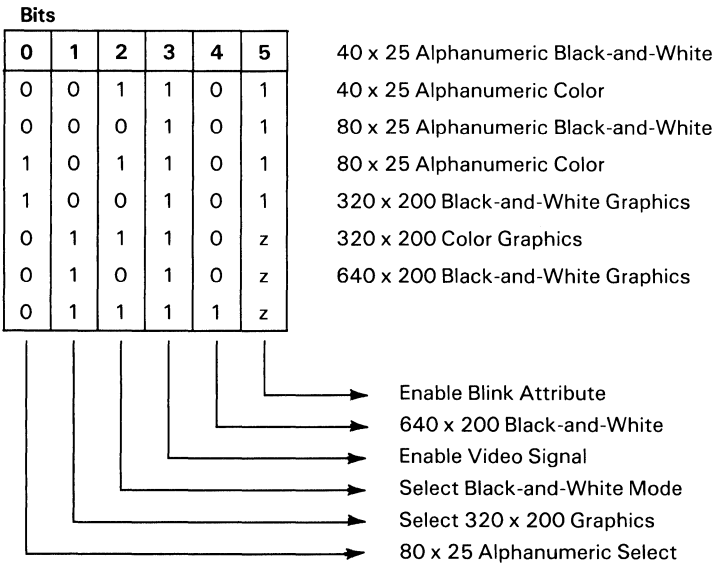
Bit 2 A 1 selects black-and-white mode.
A 0 selects color mode.

Bit 3 A 1 enables the video signal. The video signal is disabled when changing modes.

Bit 4 A 1 selects the high-resolution (640 by 200) black-and-white graphics mode. One of eight colors can be selected on direct-drive monitors in this mode by using register hex 3D9.

Bit 5 A 1 will change the character background intensity to the blinking attribute function for alphanumeric modes. When the high-order attribute is not selected, 16 background colors or intensified colors are available. This bit is set to 1 to allow the blinking function.

Mode-Control Register Summary



z = don't care condition

Note: The low-resolution (160 by 100) mode requires special programming and is set up as the 40 by 25 alphanumeric mode.

Status Register

The status register is a 4-bit read-only register. Its I/O address is hex 3DA, and it can be read using the processor 'in' instruction. The following are bit definitions for this register.

Bit 0 A 1 indicates that a regen-buffer memory access can be made without interfering with the display.

Bit 1 A 1 indicates that a positive-going edge from the light pen has set the light pen's trigger. This trigger is reset when power is turned on and may also be cleared by a processor 'out' command to hex address 3DB. No specific data setting is required; the action is address-activated.

Bit 2 The light pen switch is reflected in this bit. The switch is not latched or debounced. A 0 indicates that the switch is on.

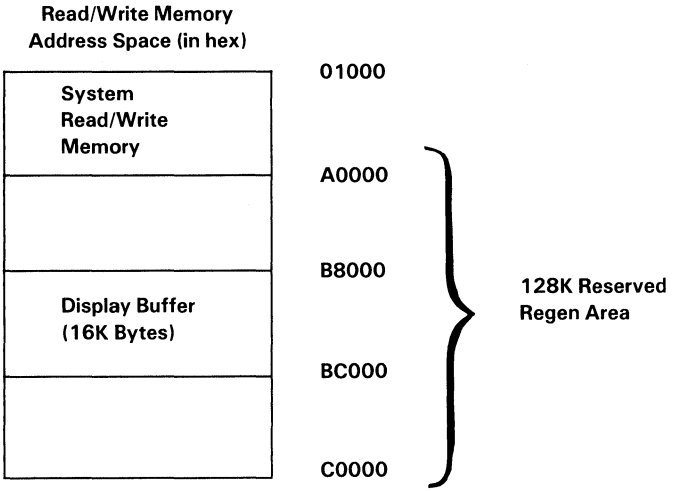
Bit 3 A 1 indicates that the raster is in a vertical retrace mode. Screen-buffer updating can be performed at this time.

Sequence of Events for Changing Modes

1. Determine the mode of operation.
2. Reset the video-enable bit in the mode-control register.
3. Program the 6845 CRT Controller to select the mode.
4. Program the mode-control and color-select registers including re-enabling the video.

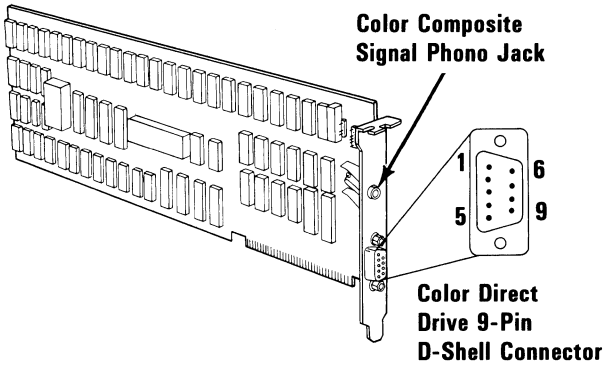
Memory Requirements

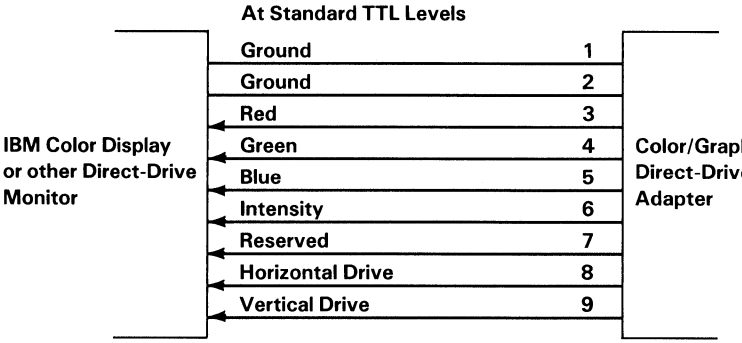
The memory used by this adapter is self–contained. It consists of 16K bytes of memory without parity. This memory is used as both a display buffer for alphanumeric data and as a bit map for graphics data. The regen buffer's address starts at hex B8000.

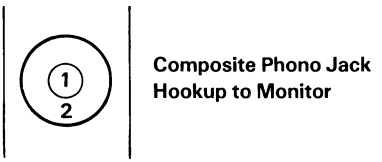


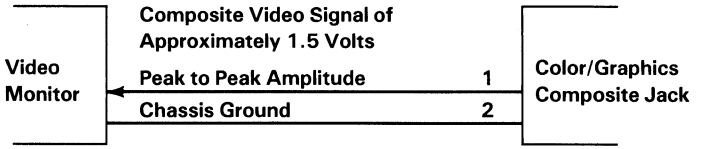
Specifications

The following pages contain card and connector specifications for the IBM Color/Graphics Monitor Adapter.

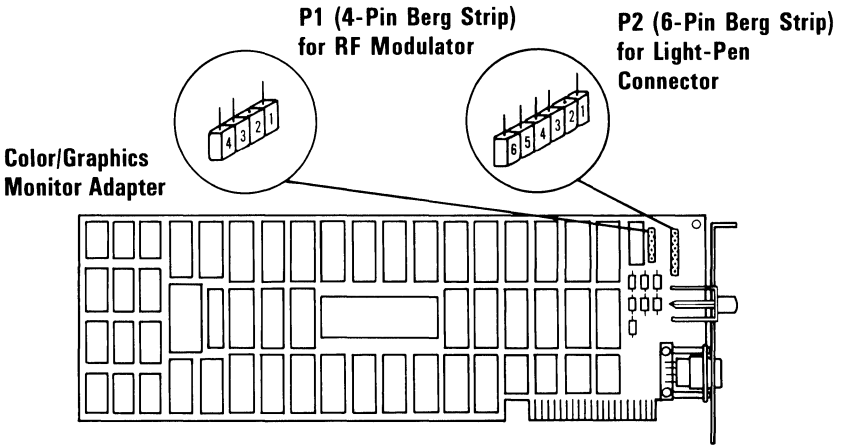


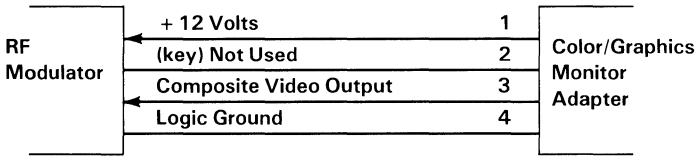




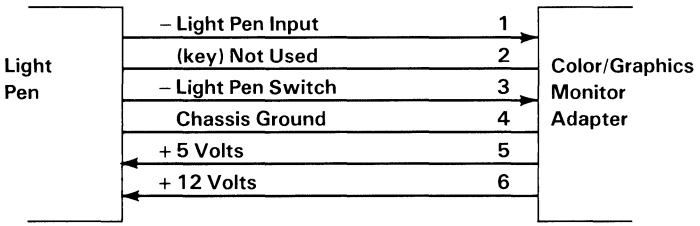


Connector Specifications (Part 1 of 2)





RF Modulator Interface

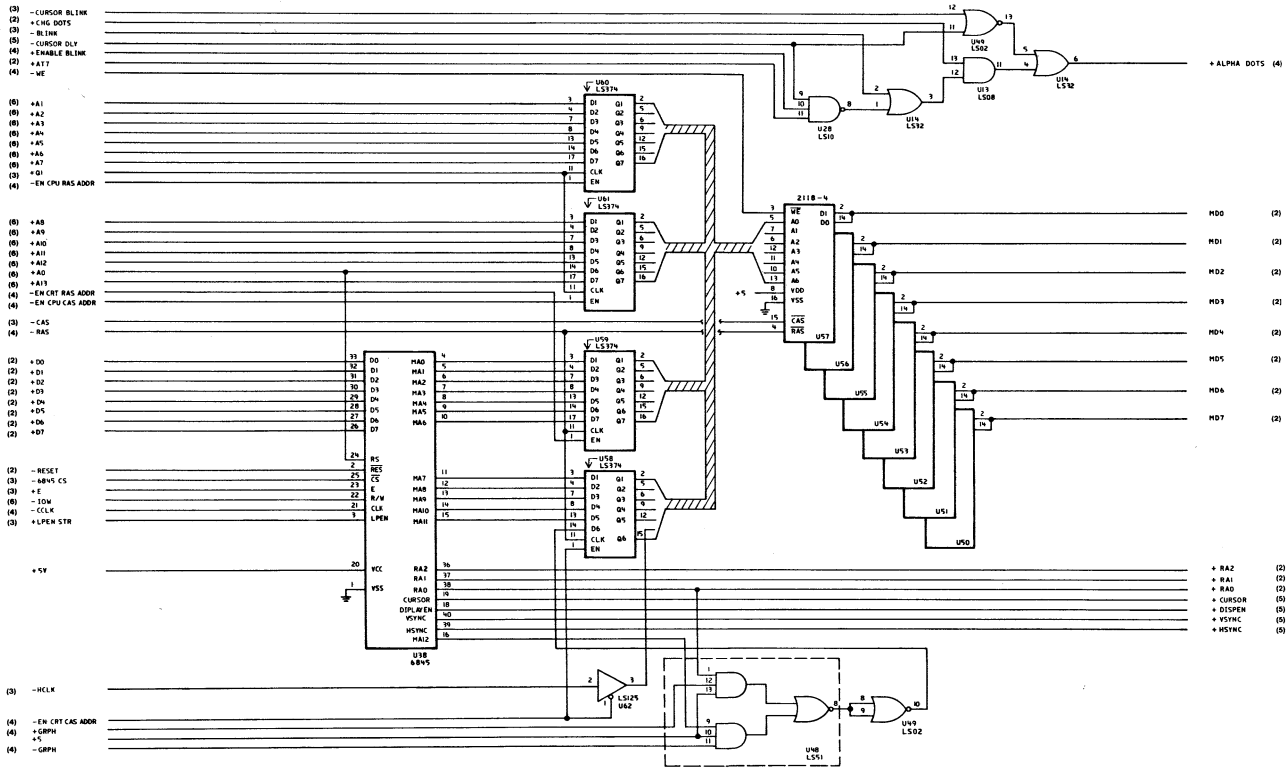


Light Pen Interface

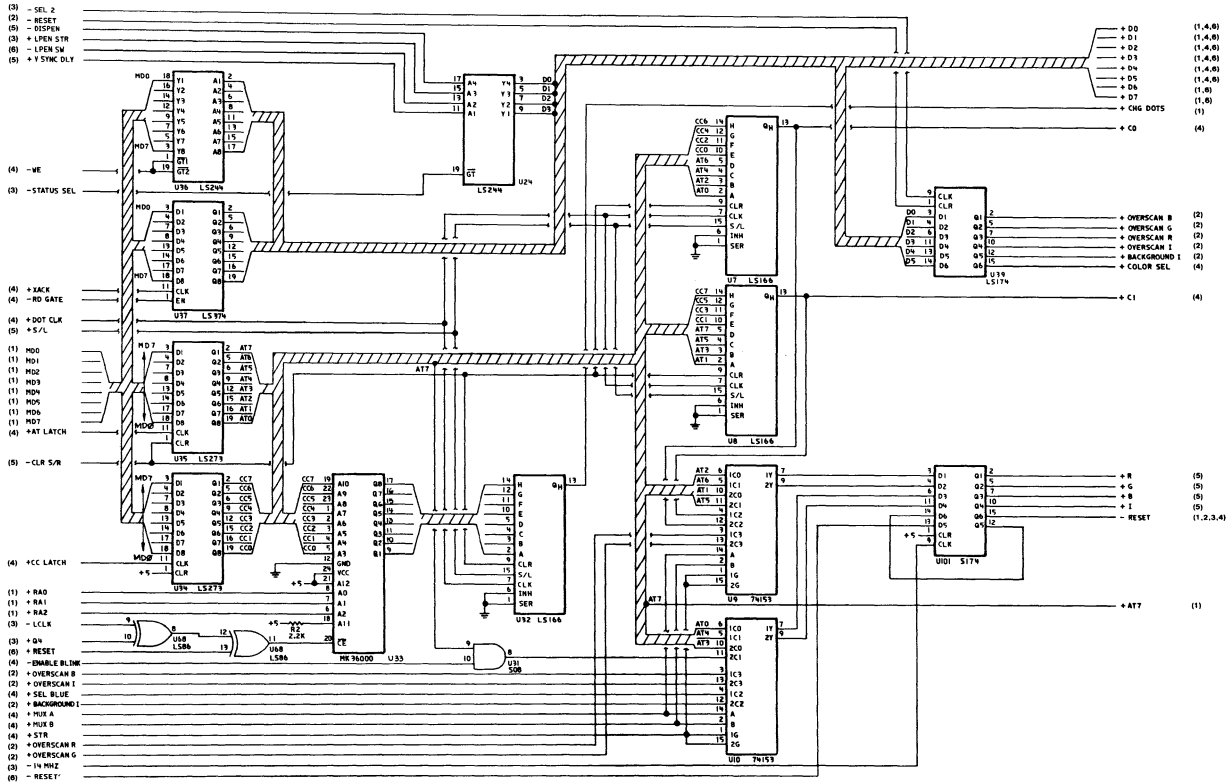
Connector Specifications (Part 2 of 2)

Logic Diagrams

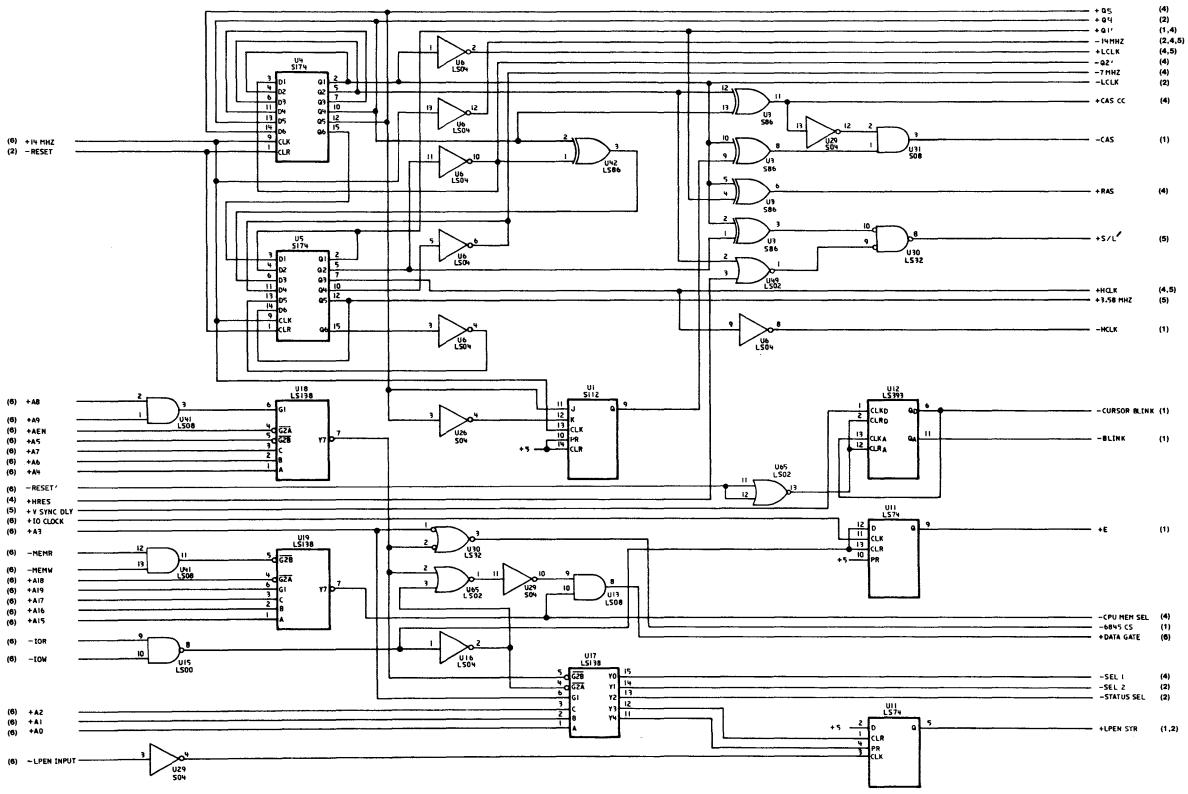
The following pages contain the logic diagrams for the IBM Color/Graphics Monitor Adapter.



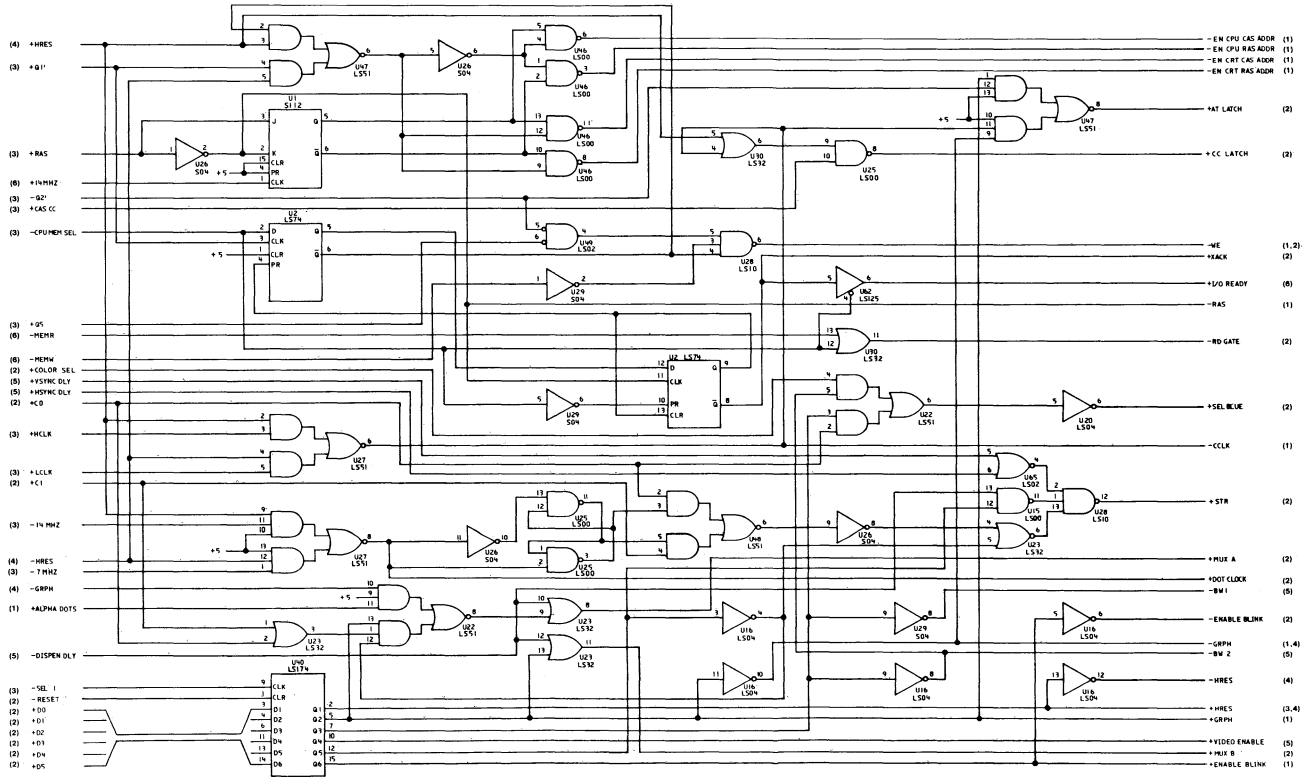
Color/Graphics Monitor Adapter (Sheet 1 of 6)



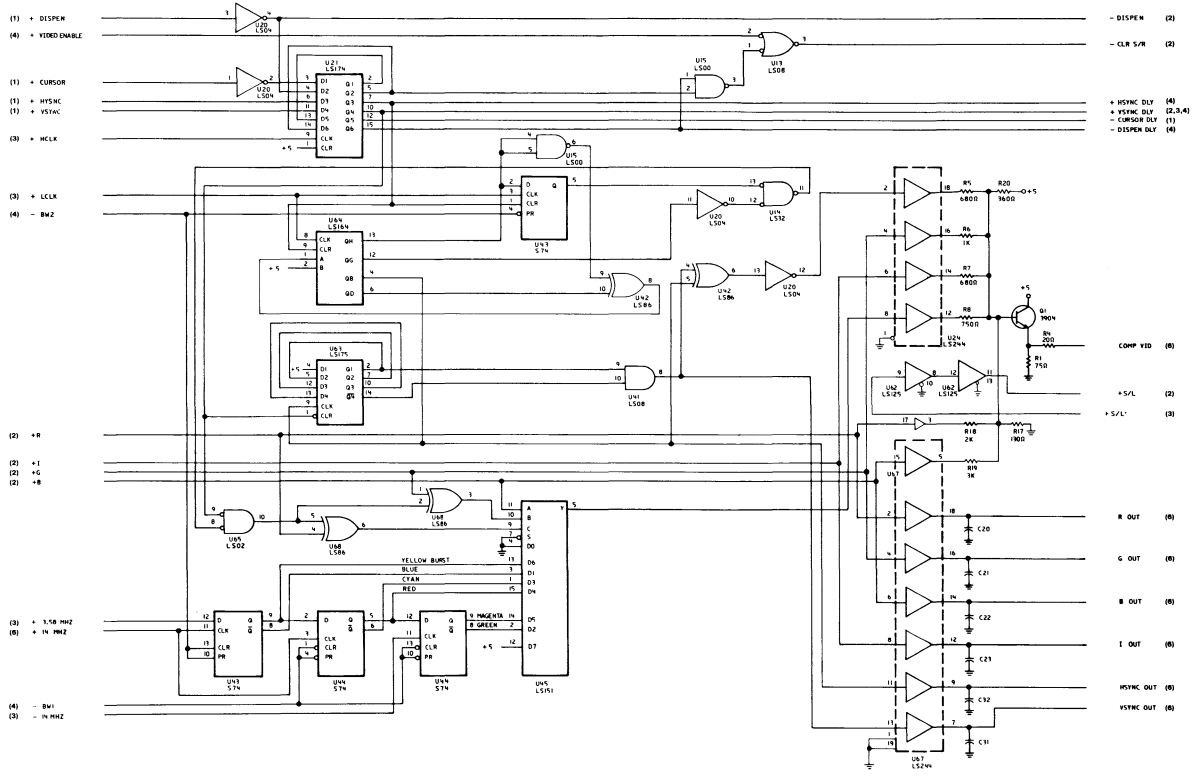
Color/Graphics Monitor Adapter (Sheet 2 of 6)



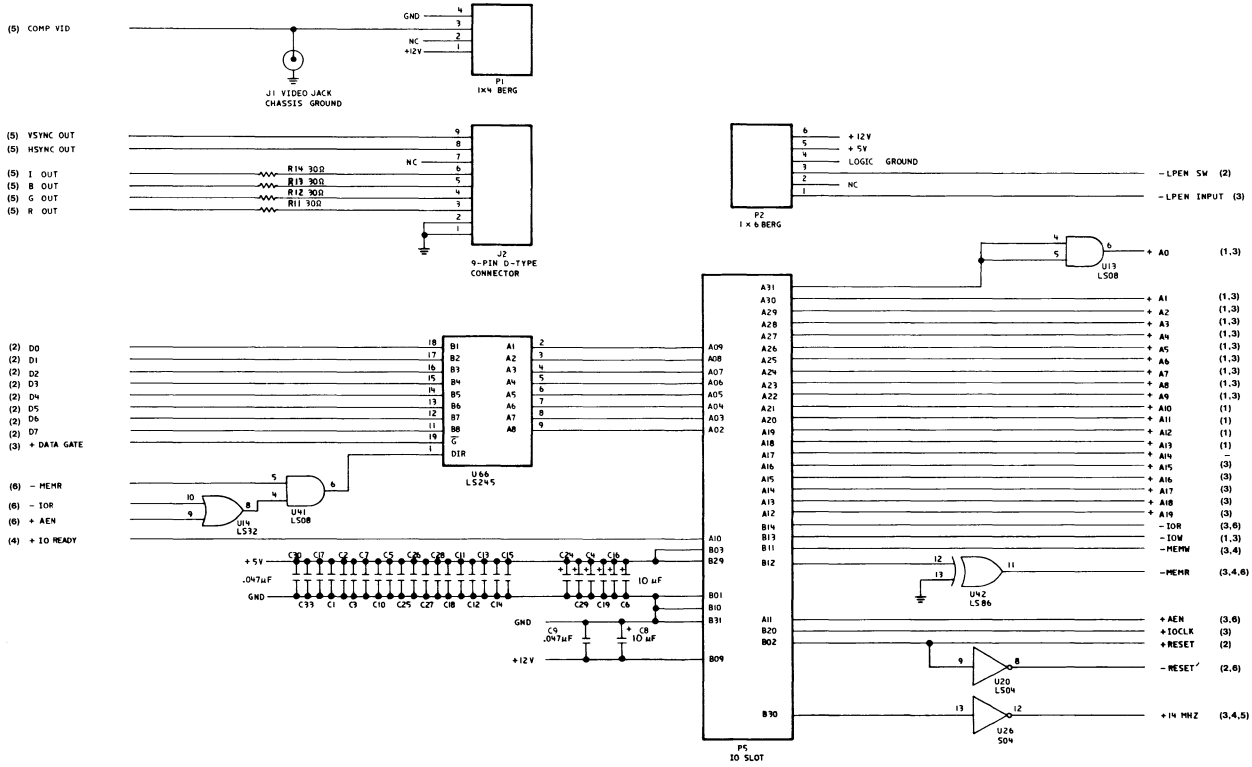
Color/Graphics Monitor Adapter (Sheet 3 of 6)



Color/Graphics Monitor Adapter (Sheet 4 of 6)



Color/Graphics Monitor Adapter (Sheet 5 of 6)



Color/Graphics Monitor Adapter (Sheet 6 of 6)

Index

A

B

C

D

G

H

L

M

P

R

S

T

Personal Computer

IBM Enhanced Graphics Adapter

Contents

Description

The IBM Enhanced Graphics Adapter (EGA) is a graphics controller that supports both color and monochrome direct drive displays in a variety of modes. In addition to the direct drive port, a light pen interface is provided. Advanced features on the adapter include bit-mapped graphics in four planes and a RAM (Random Access Memory) loadable character generator. Design features in the hardware substantially reduce the software overhead for many graphics functions.

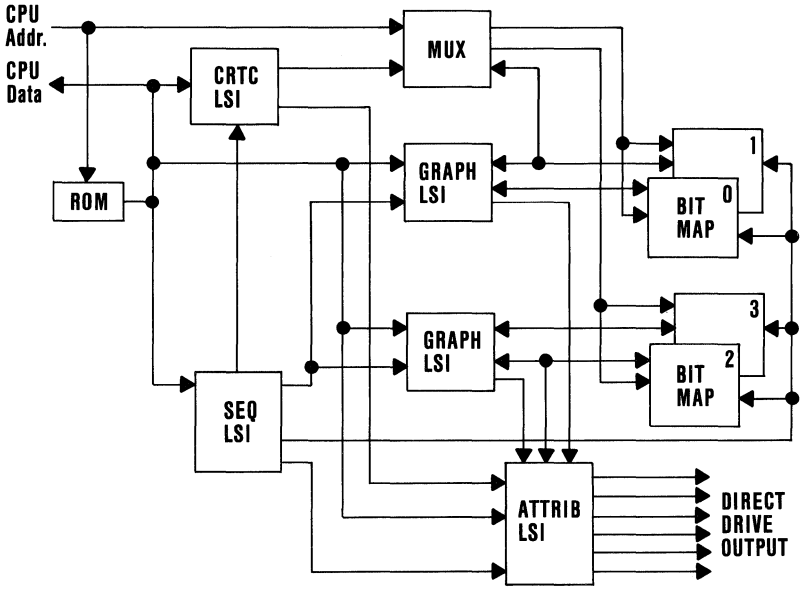
The Enhanced Graphics Adapter provides Basic Input Output System (BIOS) support for both alphanumeric (A/N) modes and all-points-addressable (APA) graphics modes, including all modes supported by the Monochrome Display Adapter and the Color/Graphics Monitor Adapter. Other modes provide APA 640x350 pel graphics support for the IBM Monochrome Display, full 16 color support in both 320x200 pel and 640x200 pel resolutions for the IBM Color Display, and both A/N and APA support with resolution of 640x350 for the IBM Enhanced Color Display. In alphanumeric modes, characters are formed from one of two ROM (Read Only Memory) character generators on the adapter. One character generator defines 7x9 characters in a 9x14 character box. For Enhanced Color Display support, the 9x14 character set is modified to provide an 8x14 character set. The second character generator defines 7x7 characters in an 8x8 character box. These generators contain dot patterns for 256 different characters. The character sets are identical to those provided by the IBM Monochrome Display Adapter and the IBM Color/Graphics Monitor Adapter.

The adapter contains 64K bytes of storage configured as four 16K byte bit planes. Memory expansion options are available to expand the adapter memory to 128K bytes or 256K bytes.

The adapter is packaged on a single 13-1/8 inch (333.50 mm) card. The direct drive port is a right-angle mounted connector at the rear of the adapter and extends through the rear panel of the system unit. Also on the card are five large scale integration (LSI) modules custom designed for this controller.

Located on the adapter is a feature connector that provides access to internal functions through a 32-pin berg connector. A separate 64-pin connector provides an interface for graphics memory expansion.

The following is a block diagram of the Enhanced Graphics Adapter:



Enhanced Graphics Adapter Block Diagram

Major Components

CRT Controller

The CRT (Cathode Ray Tube) Controller (CRTC) 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 processor to access memory during active display intervals by inserting dedicated processor memory cycles periodically between the display memory cycles. Map mask registers are available to protect entire memory maps from being changed.

Graphics Controller

The Graphics Controller directs the data from the memory to the attribute controller and the processor. In graphics modes, memory data is sent in serialized form to the attribute chip. In alpha modes the memory data is sent in parallel form, bypassing the graphics controller. The graphics controller formats the data for compatible modes and provides color comparators for use in color painting modes. Other hardware facilities allow the processor to write 32 bits in a single memory cycle, (8 bits per plane) for quick color presetting of the display areas, and additional logic allows the processor to write data to the display on non-byte boundaries.

Attribute Controller

The Attribute Controller provides a color palette of 16 colors, each of which may be specified separately. Six color outputs are

available for driving a display. Blinking and underlining are controlled by this chip. This chip takes data from the display memory and formats it for display on the CRT screen.

Display Buffer

The display buffer on the adapter consists of 64K bytes of dynamic read/write memory configured as four 16K byte video bit planes. Two options are available for expanding the graphics memory. The Graphics Memory Expansion Card plugs into the memory expansion connector on the adapter, and adds one bank of 16K to each of the four bit planes, increasing the graphics memory to 128K bytes. The expansion card also provides DIP sockets for further memory expansion. Populating the DIP sockets with the Graphics Memory Module Kit adds two additional 16K banks to each bit plane, bringing the graphics memory to its maximum of 256K bytes.

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

BIOS

A read-only memory (ROM) Basic Input Output System (BIOS) module on the adapter is linked to the system BIOS. This ROM BIOS contains character generators and control code and is mapped into the processor address at hex C0000 for a length of 16K bytes.

Support Logic

The logic on the card surrounding the LSI modules supports the modules and creates latch buses for the CRT controller, the

processor, and character generator. Two clock sources (14 MHz and 16 MHz) provide the dot rate. The clock is multiplexed under processor I/O control. The four I/O registers on the card are not part of the LSI devices.

Modes of Operation

IBM Color Display

The following table describes the modes supported by BIOS on the IBM Color Display:

Mode #	Type	Colors	Alpha Format	Buffer Start	Box Size	Max. Pages	Resolution
0	A/N	16	40x25	B8000	8x8	8	320x200
1	A/N	16	40x25	B8000	8x8	8	320x200
2	A/N	16	80x25	B8000	8x8	4/8/8	640x200
3	A/N	16	80x25	B8000	8x8	4/8/8	640x200
4	APA	4	40x25	B8000	8x8	1	320x200
5	APA	4	40x25	B8000	8x8	1	320x200
6	APA	2	80x25	B8000	8x8	1	640x200
D	APA	16	40x25	A0000	8x8	2/4/8	320x200
E	APA	16	80x25	A0000	8x8	1/2/4	640x200

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

Modes 0, 2, and 5 are identical to modes 1, 3, and 4, respectively, at the adapter's direct drive interface.

The "MAX. PAGES" fields for modes 2, 3, D, and E indicate the number of pages supported when 64K, 128K, or 256K bytes of graphics memory is installed, respectively.

IBM Monochrome Display

The following table describes the modes supported by BIOS on the IBM Monochrome Display.

Mode #	Type	Colors	Alpha Format	Buffer Start	Box Size	Max. Pages	Resolution
7	A/N	4	80x25	B0000	9x14	4/8	720x350
F	APA	4	80x25	A0000	8x14	1/2	640x350

The "MAX. PAGES" fields for modes 7 and F indicate the number of pages supported when either 64K or greater than 64K of graphics memory is installed, respectively.

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

IBM Enhanced Color Display

The Enhanced Graphics Adapter supports attachment of the IBM Enhanced Color Display. The IBM Enhanced Color Display is capable of running at the standard television frequency of 15.75 KHz as well as running 21.85 KHz. The table below summarizes the characteristics of the IBM Enhanced Color Display:

Parameter	TV Frequency	High Resolution
Horiz Scan Rate	15.75 KHz.	21.85 KHz.
Vertical Scan Rate	60 Hz.	60 Hz.
Video Bandwidth	14.318 MHz.	16.257 MHz.
Displayable Colors	16 Maximum	16 or 64
Character Size	7 by 7 Pels	7 by 9 Pels
Character Box Size	8 by 8 Pels	8 by 14 Pels
Maximum Resolution	640x200 Pels	640 by 350 Pels
Alphanumeric Modes	0,1,2,3	0,1,2,3
Graphics Modes	4,5,6,D,E	10

In the television frequency mode, the IBM Enhanced Color Display displays information identical in color and resolution to the IBM Color Display.

In the high resolution mode, the adapter provides enhanced alphanumeric character support. This enhanced alphanumeric support consists of transforming the 8 by 8 character box into an 8 by 14 character box, and providing 16 colors out of a palette of 64 possible display colors. Display colors are changed by altering the programming of the color palette registers in the Attribute Controller. In alphanumeric modes, any 16 of 64 colors are displayable. The screen resolution is 320x350 for modes 0 and 1, and 640x350 for modes 2 and 3.

The resolution displayed on the IBM Enhanced Color Display is selected by the switch settings on the Enhanced Graphics Adapter.

The Enhanced Color Display is compatible with all modes listed for the IBM Color Display. The following table describes additional modes supported by BIOS for the IBM Enhanced Color Display:

Mode #	Type	Colors	Alpha Format	Buffer Start	Box Size	Max. Pages	Resolution
0*	A/N	16/64	40x25	B8000	8x14	8	320x350
1*	A/N	16/64	40x25	B8000	8x14	8	320x350
2*	A/N	16/64	80x25	B8000	8x14	4/8	640x350
3*	A/N	16/64	80x25	B8000	8x14	4/8	640x350
10	APA	4/16 16/64	80x25	A0000	8x14	1/2	640x350

* Note that modes 0, 1, 2, and 3 are also listed for IBM Color Display support. BIOS provides enhanced support for these modes when an Enhanced Color Display is attached.

The values in the "COLORS" field indicate 16 colors of a 64 color palette or 4 colors of a sixteen color palette.

In modes 2, 3, and 10, the dual values for the "COLORS" field and the "MAX. PAGES" field indicate the support provided when either 64K or greater than 64K of graphics memory is installed, respectively.

Basic Operations

Alphanumeric Modes

The data format for alphanumeric modes on the Enhanced Graphics Adapter is the same as the data format on the IBM Color/Graphics Monitor Adapter and the IBM Monochrome Display Adapter. As an added function, bit three 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. This function is valid only when memory has been expanded to 128K bytes or more.

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

Graphics Modes

320x200 Two and Four Color Graphics (Modes 4 and 5)

Addressing, mapping and data format are the same as the 320x200 pel mode of the Color/Graphics Monitor Adapter. The display buffer is configured at hex B8000. Bit image data is stored in bit planes 0 and 1.

640x200 Two Color Graphics (Mode 6)

Addressing, mapping and data format are the same as the 640x200 pel black and white mode of the Color/Graphics

Monitor Adapter. The display buffer is configured at hex B8000. Bit image data is stored in bit plane 0.

640x350 Monochrome Graphics (Mode F)

This mode supports graphics on the IBM Monochrome Display with the following attributes: black, video, blinking video, and intensified video. Resolution of 640x350 requires 56K bytes to support four attributes. By chaining maps 0 and 1, then maps 2 and 3 together, two 32K bit planes can be formed. This chaining is done only when necessary (less than 128K of graphics memory). The first map is the video bit plane, and the second map is the intensity bit plane. Both planes reside at hex address A0000.

Two bits, one from each bit plane, define one picture element (pel) on the screen. The bit definitions for the pels are given in the following table. The video bit plane is denoted by C0 and the Intensity Bit Plane is denoted by C2.

C2	C0	Pixel Color	Valid Attributes
0	0	Black	0
0	1	Video	3
1	0	Blinking Video	C
1	1	Intensified Video	F

The byte organization in memory is sequential. The first eight pels on the screen are defined by the contents of memory in location A000:0H, the second eight pels by location A000:1H, 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.

Monochrome graphics works in odd/even mode, which means that even CPU addresses go into even bit planes and odd CPU addresses go into odd bit planes. Since both bit planes reside at address A0000, the user must select which plane or planes he desires to update. This is accomplished by the map mask register of the sequencer. (See the table above for valid attributes).

16/64 Color Graphics Modes (Mode 10)

These modes support graphics in 16 colors on either a medium or high resolution monitor. The memory in these modes consists of using all four bit planes. 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. The color combinations are illustrated in the following table:

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	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 A0000. The map mask register of the sequencer is used to select any or all of the bit planes to be updated when a memory write to the display buffer is executed by the CPU.

Color Mapping

The Enhanced Graphics Adapter supports 640x350 Graphics for both the IBM Monochrome and the IBM Enhanced Color

Displays. Four color capability is supported on the EGA without the Graphics Memory Expansion Card (base 64 KB), and sixteen colors are supported when the Graphics Memory Expansion Card is installed on the adapter (128 KB or above). This section describes the differences in the colors displayed depending upon the graphics memory available. Note that colors 0H, 1H, 4H, and 7H map directly regardless of the graphics memory available.

Character Attribute	Monochrome	Mode 10H 64KB	Mode 10H >64KB
00H*	Black	Black	Black
01H*	Video	Blue	Blue
02H	Black	Black	Green
03H	Video	Blue	Cyan
04H*	Blinking	Red	Red
05H	Intensified	White	Magenta
06H	Blinking	Red	Brown
07H*	Intensified	White	White
08H	Black	Black	Dark Gray
09H	Video	Blue	Light Blue
0AH	Black	Black	Light Green
0BH	Video	Blue	Light Cyan
0CH	Blinking	Red	Light Red
0DH	Intensified	White	Light Magenta
0EH	Blinking	Red	Yellow
0FH	Intensified	White	Intensified White

* Graphics character attributes which map directly regardless of the graphics memory available.

Registers

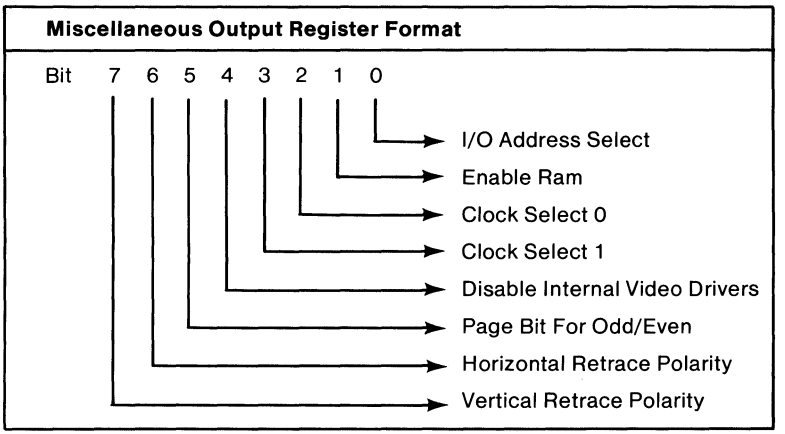
External Registers

This section contains descriptions of the registers of the Enhanced Graphics Adapter that are not contained in an LSI device.

Name	Port	Index
Miscellaneous Output Register	3C2	-
Feature Control Register	3?A	-
Input Status Register 0	3C2	-
Input Status Register 1	3?2	-
? = B in Monochrome Modes	? = D in Color Modes	

Miscellaneous Output Register

This is a write-only register. The processor output port address is hex 3C2. A hardware reset causes all bits to reset to zero.



Bit 0 3BX/3DX CRTC I/O Address—This bit maps the CRTC I/O addresses for IBM Monochrome or Color/Graphics Monitor Adapter emulation. A logical 0 sets CRTC addresses to 3BX and Input Status Register 1 's address to 3BA for Monochrome emulation. A logical 1 sets CRTC

addresses to 3DX and Input Status Register 1's address to 3DA for Color/Graphics Monitor Adapter emulation.

Bit 1 Enable RAM—A logical 0 disables RAM from the processor; a logical 1 enables RAM to respond at addresses designated by the Control Data Select value programmed into the Graphics Controllers.

Bit 2-Bit 3 Clock Select—These two bits select the clock source according to the following table:

Bits 3 2	
0 0-	Selects 14 MHz clock from the processor I/O channel
0 1-	Selects 16 MHz clock on-board oscillator
1 0-	Selects external clock source from the feature connector.
1 1-	Not used

Bit 4 Disable Internal Video Drivers—A logical 0 activates internal video drivers; a logical 1 disables internal video drivers. When the internal video drivers are disabled, the source of the direct drive color output becomes the feature connector direct drive outputs.

Bit 5 Page Bit For Odd/Even—Selects between two 64K pages of memory when in the Odd/Even modes (0,1,2,3,7). A logical 0 selects the low page of memory; a logical 1 selects the high page of memory.

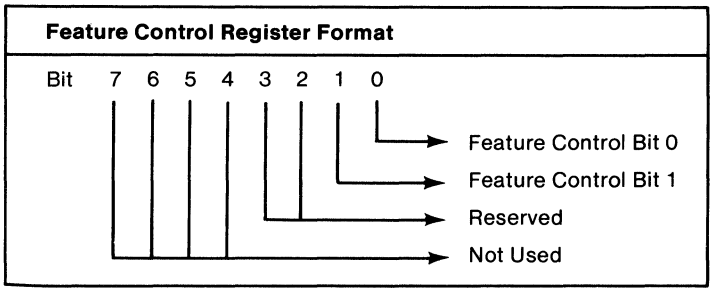
Bit 6 Horizontal Retrace Polarity—A logical 0 selects positive horizontal retrace; a logical 1 selects negative horizontal retrace.

Bit 7 Vertical Retrace Polarity—A logical 0 selects positive vertical retrace; a logical 1 selects

negative vertical retrace. The IBM Monochrome display requires a negative vertical retrace polarity.

Feature Control Register

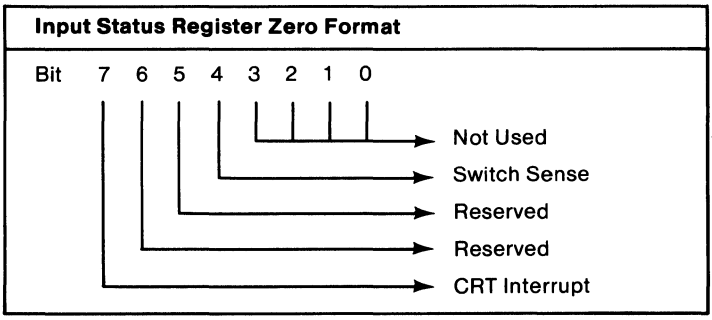
This is a write-only register. The processor output register is hex 3BA or 3DA.



Bits 0 and 1 Feature Control Bits—These bits are used to convey information to the feature connector. The output of these bits goes to the FEAT 0 (pin 19) and FEAT 1 (pin 17) of the feature connector.

Input Status Register Zero

This is a read-only register. The processor input port address is hex 3C2.



Bit 4 Switch Sense—When set to 1, this bit allows the processor to read the four configuration switches on the board. The setting of the CLKSEL field determines which switch is being read. The switch configuration can be determined by reading byte 40:88H in RAM.

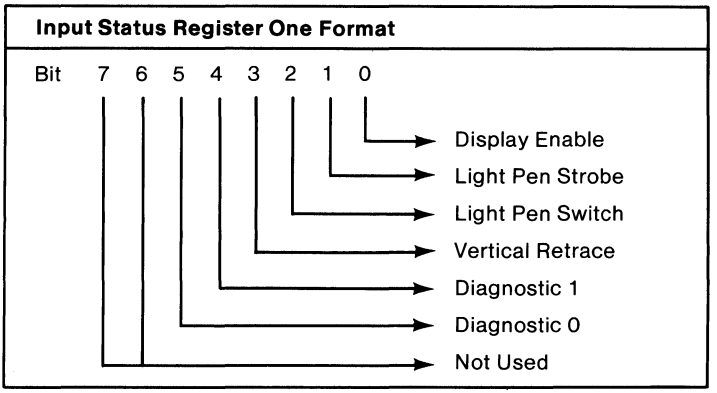
Bit 3: Switch 4 ; Logical 0 = switch closed
Bit 2: Switch 3 ; Logical 0 = switch closed
Bit 1: Switch 2 ; Logical 0 = switch closed
Bit 0: Switch 1 ; Logical 0 = switch closed

Bits 5 and 6 Feature Code—These bits are input from the Feat (0) and Feat (1) pins on the feature connector.

Bit 7 CRT Interrupt—A logical 1 indicates video is being displayed on the CRT screen; a logical 0 indicates that vertical retrace is occurring.

Input Status Register One

This is a read-only register. The processor port address is hex 3BA or hex 3DA.



Bit 0 Display Enable—Logical 0 indicates the CRT raster is in a horizontal or vertical retrace interval. This bit is the real time status of the display enable signal. Some programs use this status bit to restrict screen updates to inactive display intervals. The Enhanced Graphics Adapter does not require the CPU to update the screen buffer during inactive display intervals to avoid glitches in the display image.

Bit 1 Light Pen Strobe—A logical 0 indicates that the light pen trigger has not been set; a logical 1 indicates that the light pen trigger has been set.

Bit 2 Light Pen Switch—A logical 0 indicates that the light pen switch is closed; a logical 1 indicates that the light pen switch is open.

Bit 3 Vertical Retrace—A logical 0 indicates that video information is being displayed on the CRT screen; a logical 1 indicates the CRT is in a vertical retrace interval. This bit can be programmed to interrupt the processor on interrupt level 2 at the start of the vertical retrace. This is done through bits 4 and 5 of the Vertical Retrace End Register of the CRTC.

Bits 4 and 5 Diagnostic Usage—These bits are selectively connected to two of the six color outputs of the

Attribute Controller. The Color Plane Enable register controls the multiplexer for the video wiring. The following table illustrates the combinations available and the color output wiring.

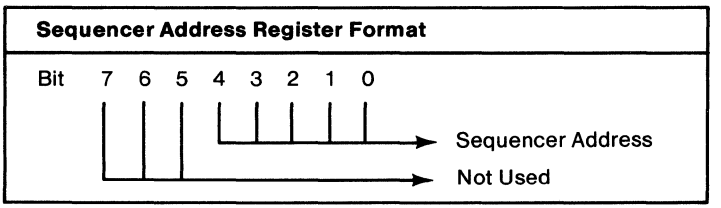
Color Plane Register		Input Status Register One	
Bit 5	**Bit 4**	**Bit 5**	**Bit 4**
0	0	Red	Blue
0	1	Secondary Blue	Green
1	0	Secondary Red	Secondary Green
1	1	Not Used	Not Used

Sequencer Registers

Name	Port	Index
Address	3C4	-
Reset	3C5	00
Clocking Mode	3C5	01
Map Mask	3C5	02
Character Map Select	3C5	03
Memory Mode	3C5	04

Sequencer Address Register

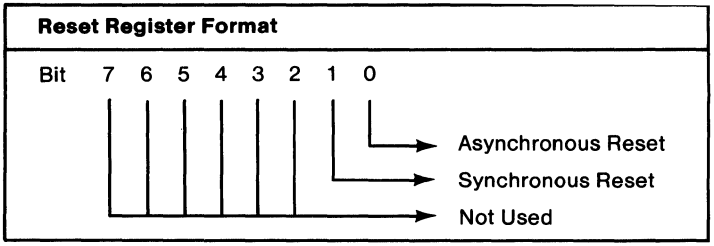
The Address Register is a pointer register located at address hex 3C4. This register is loaded with a binary value that points to the sequencer data register where data is to be written. This value is referred to as "Index" in the table above.



Bit 0-Bit 3 Sequencer Address Bits—A binary value pointing to the register where data is to be written.

Reset Register

This is a write-only register pointed to when the value in the address register is hex 00. The output port address for this register is hex 3C5.

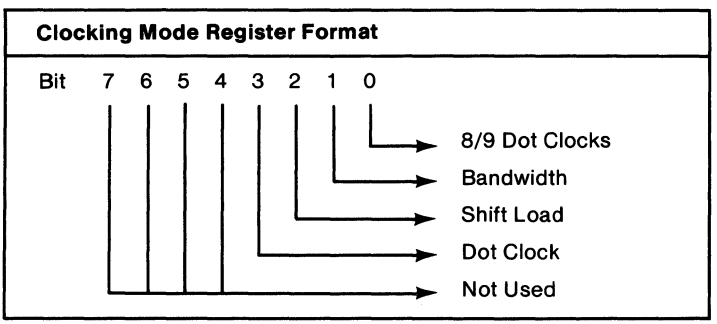


Bit 0 Asynchronous Reset—A logical 0 commands the sequencer to asynchronous clear and halt. All outputs are placed in the high impedance state when this bit is a 0. A logical 1 commands the sequencer to run unless bit 1 is set to zero. Resetting the sequencer with this bit can cause data loss in the dynamic RAMs.

Bit 1 Synchronous Reset—A logical 0 commands the sequencer to synchronous clear and halt. Bits 1 and 0 must both be ones to allow the sequencer to operate. Reset the sequencer with this bit before changing the Clocking Mode Register, if memory contents are to be preserved.

Clocking Mode Register

This is a write-only register pointed to when the value in the address register is hex 01. The output port address for this register is hex 3C5.



Bit 0 8/9 Dot Clocks—A logical 0 directs the sequencer to generate character clocks 9 dots wide; a logical 1 directs the sequencer to generate character clocks 8 dots wide. Monochrome alphanumeric mode (07H) is the only mode that uses character clocks 9 dots wide. All other modes must use 8 dots per character clock.

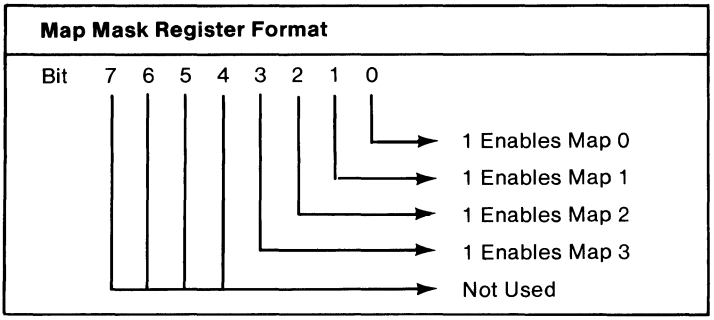
Bit 1 Bandwidth—A logical 0 makes CRT memory cycles occur on 4 out of 5 available memory cycles; a logical 1 makes CRT memory cycles occur on 2 out of 5 available memory cycles. Medium resolution modes require less data to be fetched from the display buffer during the horizontal scan time. This allows the CPU greater access time to the display buffer. All high resolution modes must provide the CRTC with 4 out of 5 memory cycles in order to refresh the display image.

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

Bit 3 Dot Clock—A logical 0 selects normal dot clocks derived from the sequencer master clock input. When this bit is set to 1, the master clock will be divided by 2 to generate the dot clock. All the other timings will be stretched since they are derived from the dot clock. Dot clock divided by two is used for 320x200 modes (0, 1, 4, 5) to provide a pixel rate of 7 MHz, (9 MHz for mode D).

Map Mask Register

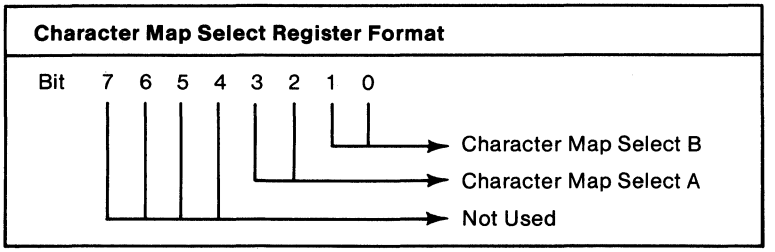
This is a write-only register pointed to when the value in the address register is hex 02. The output port address for this register is hex 3C5.



Bit 0-Bit 3 Map Mask—A logical 1 in bits 3 through 0 enables the processor to write to the corresponding maps 3 through 0. If this register is programmed with a value of 0FH, the CPU can perform a 32-bit write operation with only one memory cycle. This substantially reduces the overhead on the CPU 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 CPU 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.

Character Map Select Register

This is a write-only register pointed to when the value in the address register is hex 03. The output port address for this register is 3C5.



Bit 0-Bit 1 Character Map Select B—Selects the map used to generate alpha characters when attribute bit 3 is a 0, according to the following table:

Bits 1	0	Map Selected	Table Location
Value			
0	0	0	1st 8K of Plane 2 Bank 0
0	1	1	2nd 8K of Plane 2 Bank 1
1	0	2	3rd 8K of Plane 2 Bank 2
1	1	3	4th 8K of Plane 2 Bank 3

Bit 2-Bit 3 Character Map Select A—Selects the map used to generate alpha characters when attribute bit 3 is a 1, according to the following table:

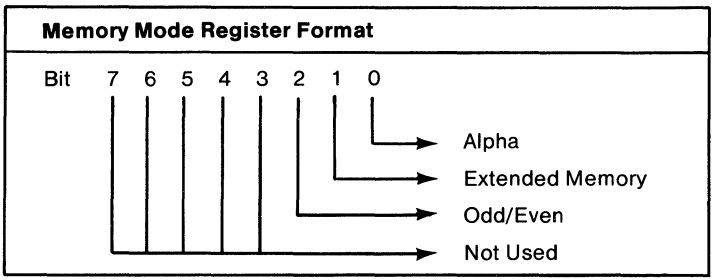
Bits 3	2	Map Selected	Table Location
Value			
0	0	0	1st 8K of Plane 2 Bank 0
0	1	1	2nd 8K of Plane 2 Bank 1
1	0	2	3rd 8K of Plane 2 Bank 2
1	1	3	4th 8K of Plane 2 Bank 3

In alphanumeric modes, bit 3 of the attribute byte normally has the function of turning the foreground intensity on or off. This bit however may be redefined as a switch between character sets. This function is enabled when there is a difference between the value in Character Map Select A and the value in Character Map Select B. Whenever these two values are the same, the character select function is disabled. The memory mode register bit 1 must be a 1 (indicates the memory extension card is installed in the unit) to enable this function; otherwise, bank 0 is always selected.

128K of graphics memory is required to support two character sets. 256K supports four character sets. Asynchronous reset clears this register to 0. This should be done only when the sequencer is reset.

Memory Mode Register

This is a write-only register pointed to when the value in the address register is hex 04. The processor output port address for this register is 3C5.



Bit 0 Alpha—A logical 0 indicates that a non-alpha mode is active. A logical 1 indicates that alpha mode is active and enables the character generator map select function.

Bit 1 Extended Memory—A logical 0 indicates that the memory expansion card is not installed. A logical 1 indicates that the memory expansion card is installed and enables access to the extended memory through address bits 14 and 15.

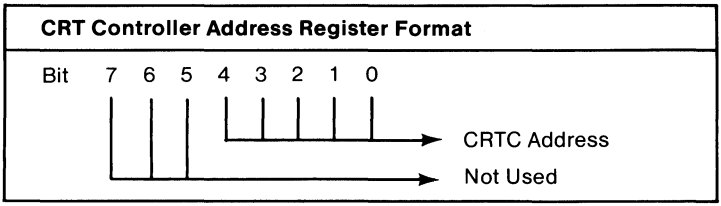
Bit 2 Odd/Even—A logical 0 directs even processor addresses to access maps 0 and 2, while odd processor addresses access maps 1 and 3. A logical 1 causes processor addresses to sequentially access data within a bit map. The maps are accessed according to the value in the map mask register.

CRT Controller Registers

Name	Port	Index
Address Register	3?4	-
Horizontal Total	3?5	00
Horizontal Display End	3?5	01
Start Horizontal Blank	3?5	02
End Horizontal Blank	3?5	03
Start Horizontal Retrace	3?5	04
End Horizontal Retrace	3?5	05
Vertical Total	3?5	06
Overflow	3?5	07
Preset Row Scan	3?5	08
Max Scan Line	3?5	09
Cursor Start	3?5	0A
Cursor End	3?5	0B
Start Address High	3?5	0C
Start Address Low	3?5	0D
Cursor Location High	3?5	0E
Cursor Location Low	3?5	0F
Vertical Retrace Start	3?5	10
Light Pen High	3?5	10
Vertical Retrace End	3?5	11
Light Pen Low	3?5	11
Vertical Display End	3?5	12
Offset	3?5	13
Underline Location	3?5	14
Start Vertical Blank	3?5	15
End Vertical Blank	3?5	16
Mode Control	3?5	17
Line Compare	3?5	18
? = B in Monochrome Modes and D in Color Modes		

CRT Controller Address Register

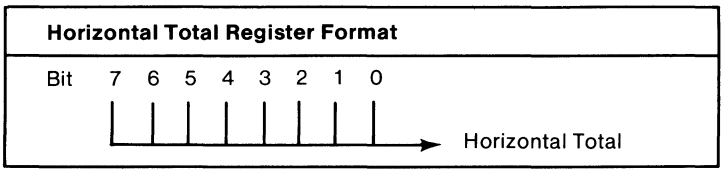
The Address register is a pointer register located at hex 3B4 or hex 3D4. If an IBM Monochrome Display is attached to the adapter, address 3B4 is used. If a color display is attached to the adapter, address 3D4 is used. This register is loaded with a binary value that points to the CRT Controller data register where data is to be written. This value is referred to as "Index" in the table above.



Bit 0-Bit 4 CRT Controller Address Bits—A binary value pointing to the CRT Controller register where data is to be written.

Horizontal Total Register

This is a write-only register pointed to when the value in the CRT Controller address register is hex 00. The processor output port address for this register is hex 3B5 or hex 3D5.

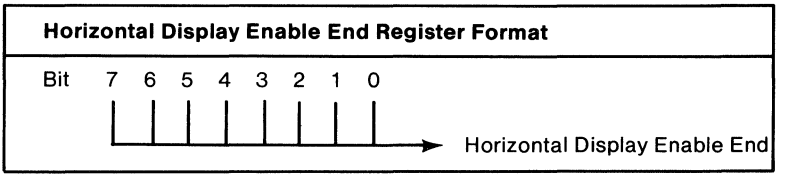


This register defines the total number of characters in the horizontal scan interval including the retrace time. The value directly controls the period of the horizontal retrace output signal. An internal horizontal character counter counts character clock inputs to the CRT Controller, and all horizontal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

Bit 0-Bit 7 Horizontal Total—The total number of characters less 2.

Horizontal Display Enable End Register

This is a write-only register pointed to when the value in the CRT Controller address register is hex 01. The processor output port address for this register is hex 3B5 or hex 3D5.

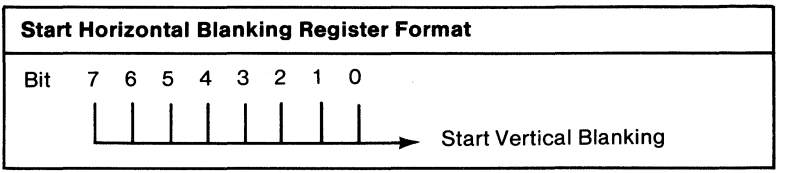


This register defines the length of the horizontal display enable signal. It determines the number of displayed character positions per horizontal line.

Bit 0-Bit 7 Horizontal display enable end —A value one less than the total number of displayed characters.

Start Horizontal Blanking Register

This is a write-only register pointed to when the value in the CRT Controller address register is hex 02. The processor output port address for this register is hex 3B5 or hex 3D5.

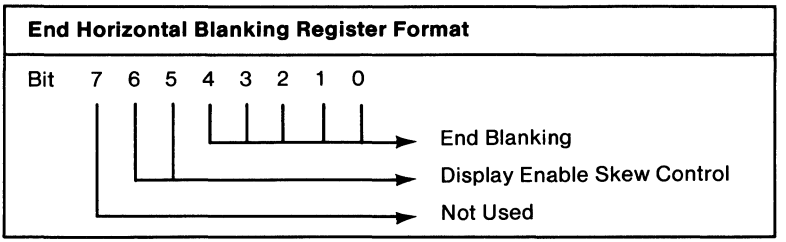


This register determines when the horizontal blanking output signal becomes active. The row scan address and underline scan line decode outputs are multiplexed on the memory address outputs and cursor outputs respectively during the blanking interval. These outputs are latched external to the CRT Controller with the falling edge of the BLANK output signal. The row scan address and underline signals remain on the output signals for one character count beyond the end of the blanking signal.

Bit 0-Bit 7 Start Horizontal Blanking—The horizontal blanking signal becomes active when the horizontal character counter reaches this value.

End Horizontal Blanking Register

This is a write-only register pointed to when the value in the CRT Controller address register is hex 03. The processor output port address for this register is hex 3B5 or hex 3D5.



This register determines when the horizontal blanking output signal becomes inactive. The row scan address and underline scan line decode outputs are multiplexed on the memory address outputs and the cursor outputs respectively during the blanking interval. These outputs are latched external to the CRT Controller with the falling edge of the BLANK output signal. The row scan address and underline signals remain on the output signals for one character count beyond the end of the blanking signal.

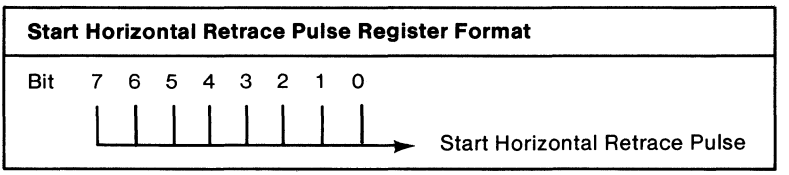
Bit 0-Bit 4 End Horizontal Blanking—A value equal to the five least significant bits of the horizontal character counter value at which time the horizontal blanking signal becomes inactive (logical 0). To obtain a blanking signal of width W, the following algorithm is used: Value of Start Blanking Register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

Bit 5-Bit 6 Display Enable Skew Control—These two bits 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 and vertical retrace signals. The bit values and amount of skew are shown in the following table:

Bits 6 5	
0 0	Zero character clock skew
0 1	One character clock skew
1 0	Two character clock skew
1 1	Three character clock skew

Start Horizontal Retrace Pulse Register

This is a write-only register pointed to when the value in the CRT Controller address register is hex 04. The processor output port address for this register is hex 3B5 or hex 3D5.

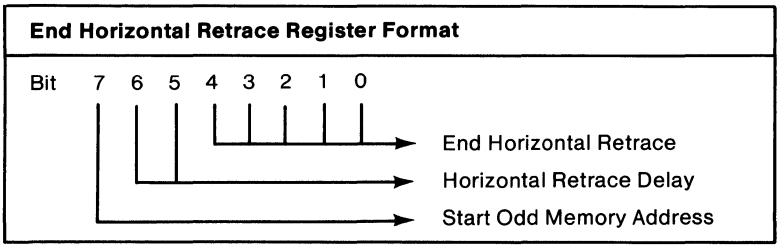


This register is used to center the screen horizontally, and to specify the character position at which the Horizontal Retrace Pulse becomes active.

Bit 0-Bit 7 Start Horizontal Retrace Pulse—The value programmed is a binary count of the character position number at which the signal becomes active.

End Horizontal Retrace Register

This is a write-only register pointed to when the value in the CRT Controller address register is hex 05. The processor output port address for this register is hex 3B5 or hex 3D5.



This register specifies the character position at which the Horizontal Retrace Pulse becomes inactive (logical 0).

Bit 0-Bit 4 End Horizontal Retrace—A value equal to the five least significant bits of the horizontal character counter value at which time 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 horizontal retrace signal in character clock units = 5-bit result to be programmed into the End Horizontal Retrace Register.

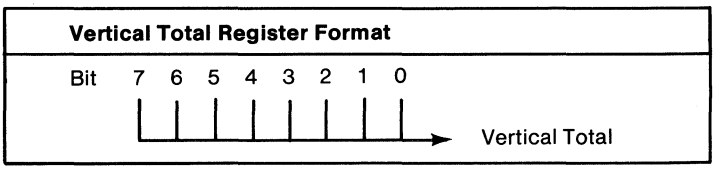
Bit 5-Bit 6 Horizontal Retrace Delay—These bits control the skew of the horizontal retrace signal. Binary 00 equals no Horizontal Retrace Delay. For some modes, it is necessary to provide a horizontal retrace signal that takes up the entire blanking interval. Some internal timings are generated by the 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 the proper screen centering.

Bit 7 Start Odd/Even Memory Address—This bit controls whether the first CRT memory address output after a horizontal retrace begins with an even or an odd address. A logical 0 selects even addresses; a logical 1 selects odd addresses. This bit is used for horizontal pel panning applications. Generally, this bit should be set to a logical 0.

Vertical Total Register

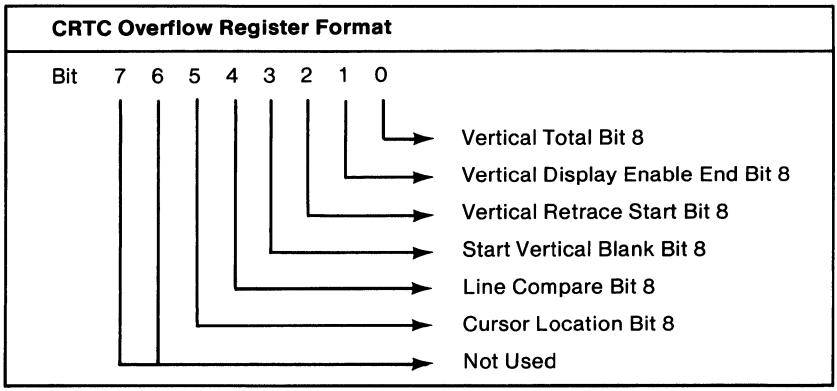
This is a write-only register pointed to when the value in the CRT Controller address register is hex 06. The processor output port address for this register is hex 3B5 or 3D5.



Bit 0-Bit 7 Vertical Total—This is the low-order eight bits of a nine-bit register. The binary value represents the number of horizontal raster scans on the CRT screen, 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 hex 07 bit 0.

CRT Controller Overflow Register

This is a write-only register pointed to when the value in the CRT Controller Address Register is hex 07. The processor output port address for this register is hex 3B5 or hex 3D5.



Bit 0 Vertical Total—Bit 8 of the Vertical Total register (index hex 06).

Bit 1 Vertical Display Enable End—Bit 8 of the Vertical Display Enable End register (index hex 12).

Bit 2 Vertical Retrace Start—Bit 8 of the Vertical Retrace Start register (index hex 10).

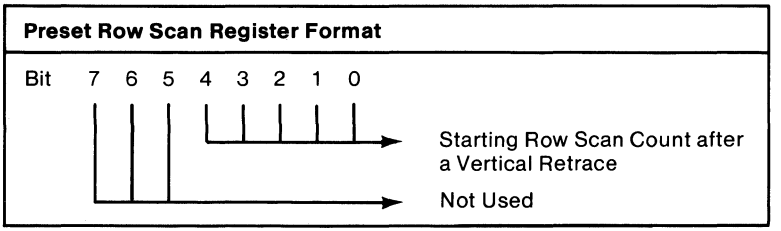
Bit 3 Start Vertical Blank—Bit 8 of the Start Vertical Blank register (index hex 15).

Bit 4 Line Compare—Bit 8 of the Line Compare register (index hex 18).

Bit 5 Cursor Location—Bit 8 of the Cursor Location register (index hex 0A).

Preset Row Scan Register

This is a write-only register pointed to when the value in the CRT Controller address register is hex 08. The processor output port address for this register is hex 3B5 or hex 3D5.

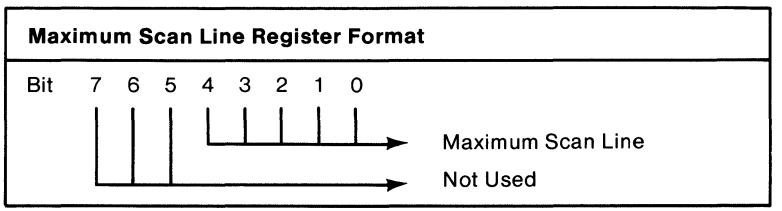


This register is used for pel scrolling.

Bit 0-Bit 4 Preset Row Scan (Pel Scrolling)—This register specifies 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 write-only register pointed to when the value in the CRT Controller address register is hex 09. The processor output port address for this register is hex 3B5 or hex 3D5.

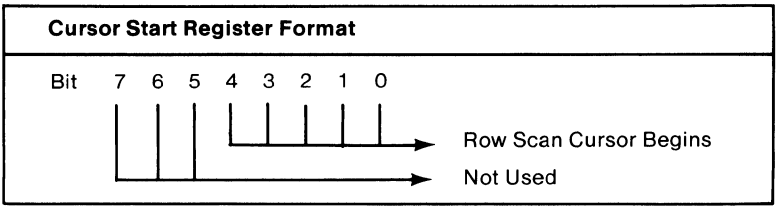


Bit 0-Bit 4 Maximum Scan Line—This register specifies the number of scan lines per character row. The number to be programmed is the maximum row scan number minus one.

Cursor Start Register

This is a write-only register pointed to when the value in the CRT Controller address register is hex 0A. The processor output port

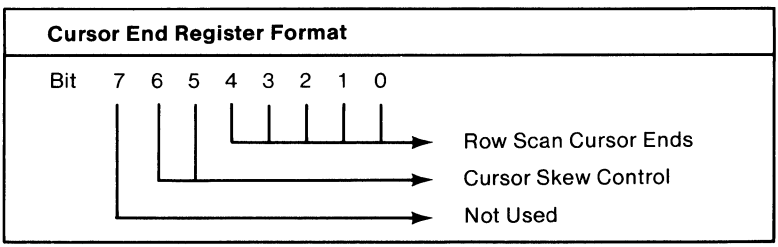
address for this register is hex 3B5 or hex 3D5.



Bit 0-Bit 4 Cursor Start—This register specifies the row scan of a character line where the cursor is to begin. The number programmed should be one less than the starting cursor row scan.

Cursor End Register

This is a write-only register pointed to when the value in the CRT Controller address register is hex 0B. The processor output port address for this register is hex 3B5 or hex 3D5.



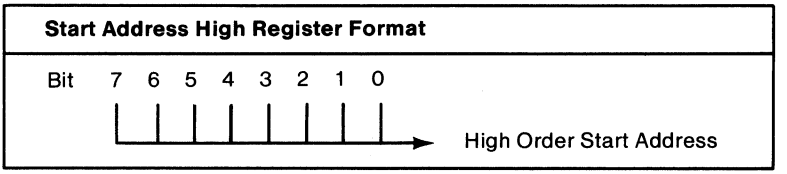
Bit 0-Bit 4 Cursor End—These bits specify the row scan where the cursor is to end.

Bit 5-Bit 6 Cursor Skew—These bits control the skew of the cursor signal.

Bits 6 5	
0 0	Zero character clock skew
0 1	One character clock skew
1 0	Two character clock skew
1 1	Three character clock skew

Start Address High Register

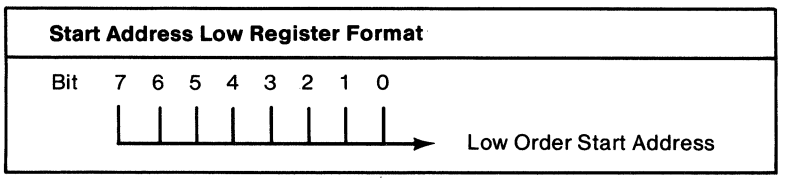
This is a read/write register pointed to when the value in the CRT Controller address register is hex 0C. The processor input/output port address for this register is hex 3B5 or hex 3D5.



Bit 0-Bit 7 Start Address High—These are the high-order eight bits of the start address. The 16-bit value, from the high-order and low-order start address registers, is the first address after the vertical retrace on each screen refresh.

Start Address Low Register

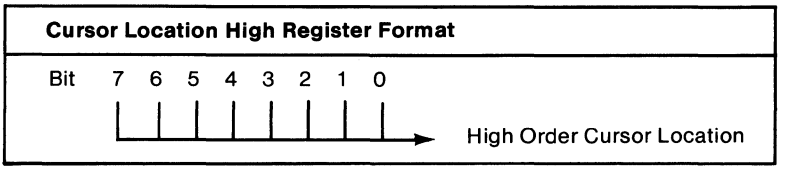
This is a read/write register pointed to when the value in the CRT Controller address register is hex 0D. The processor input/output port address for this register is hex 3B5 or hex 3D5.



Bit 0-Bit 7 Start Address Low—These are the low-order 8 bits of the start address.

Cursor Location High Register

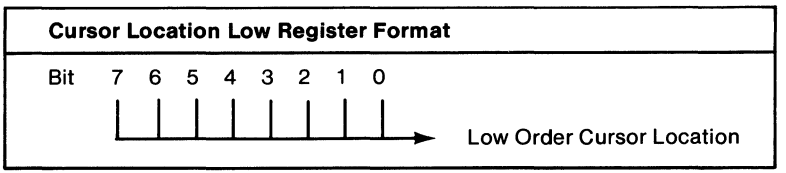
This is a read/write register pointed to when the value in the CRT Controller address register is hex 0E. The processor input/output port address for this register is hex 3B5 or hex 3D5.



Bit 0-Bit 7 Cursor Location High—These are the high-order 8 bits of the cursor location.

Cursor Location Low Register

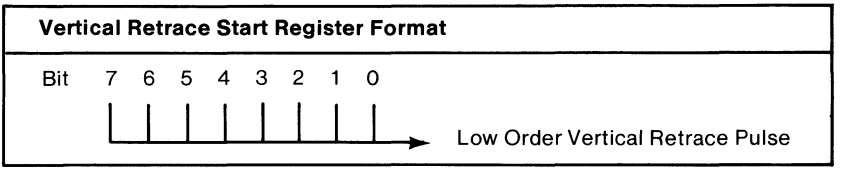
This is a read/write register pointed to when the value in the CRT Controller address register is hex 0F. The processor input/output port address for this register is hex 3B5 or Hex 3D5.



Bit 0-Bit 7 Cursor Location Low— These are the low-order 8 bits of the cursor location.

Vertical Retrace Start Register

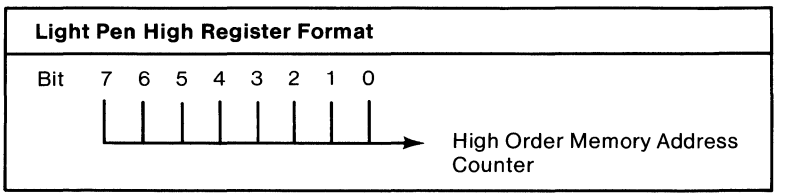
This is a write-only register pointed to when the value in the CRT Controller address register is hex 10. The processor output port address for this register is hex 3B5 or hex 3D5.



Bit 0-Bit 7 Vertical Retrace Start—This is the low-order 8 bits of the vertical retrace pulse start position programmed in horizontal scan lines. Bit 8 is in the overflow register location hex 07.

Light Pen High Register

This is a read-only register pointed to when the value in the CRT Controller address register is hex 10. The processor input port address for this register is hex 3B5 or hex 3D5.

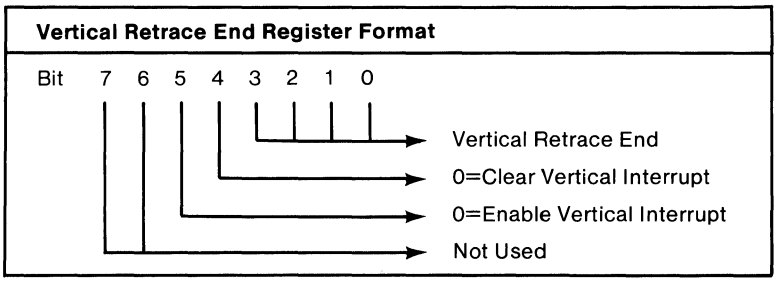


Bit 0-Bit 7 Light Pen High—This is the high order 8 bits of the memory address counter at the time the light pen was triggered.

Vertical Retrace End Register

This is a write-only register pointed to when the value in the CRT Controller address register is hex 11. The processor output port

address for this register is hex 3B5 or hex 3D5.



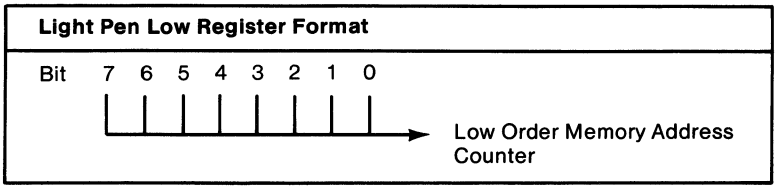
Bit 0-Bit 3 Vertical Retrace End—These bits determine the horizontal scan count value when the vertical retrace output signal becomes inactive. The register is programmed in units of horizontal scan lines. To obtain a vertical retrace signal of width W, the following algorithm is used: Value of Start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

Bit 4 Clear Vertical Interrupt—A logical 0 will clear a vertical interrupt.

Bit 5 Enable Vertical Interrupt—A logical 0 will enable vertical interrupt.

Light Pen Low Register

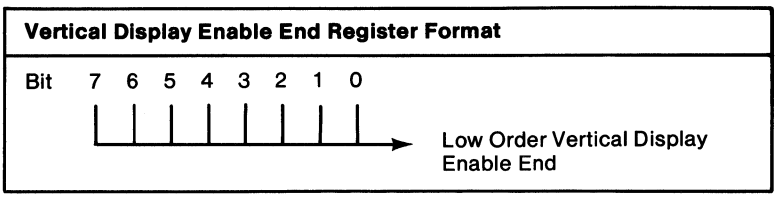
This is a read-only register pointed to when the value in the CRT Controller address register is hex 11. The processor input port address for this register is hex 3B5 or 3D5.



Bit 0-Bit 7 Light Pen Low—This is is the low-order 8 bits of the memory address counter at the time the light pen was triggered.

Vertical Display Enable End Register

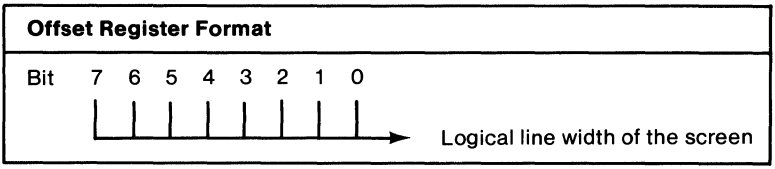
This is a write-only register pointed to when the value in the CRT Controller address register is hex 12. The processor output port address for this register is hex 3B5 or hex 3D5.



Bit 0-Bit 7 Vertical Display Enable End—These are the low-order 8 bits of the vertical display enable end position. This address specifies which scan line ends the active video area of the screen. Bit 8 is in the overflow register location hex 07.

Offset Register

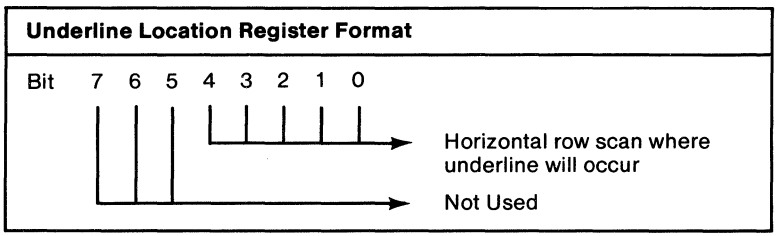
This is a write-only register pointed to when the value in the CRT Controller address register is hex 13. The processor output port address for this register is hex 3B5 or hex 3D5.



Bit 0-Bit 7 Offset—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 this amount. The Offset Register is programmed with a word address. Depending upon the method of clocking the CRT Controller, this word address is either a word or double word address.

Underline Location Register

This is a write-only register pointed to when the value in the CRT Controller address register is hex 14. The processor output port address for this register is hex 3B5 or hex 3D5.

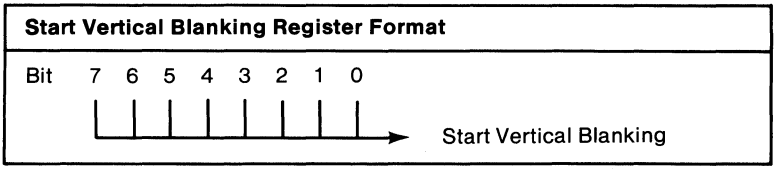


Bit 0-Bit 4 Underline Location—This register specifies the horizontal row scan on which underline will occur. The value programmed is one less than the scan line number desired.

Start Vertical Blanking Register

This is a write-only register pointed to when the value in the CRT Controller address register is hex 15. The processor output port

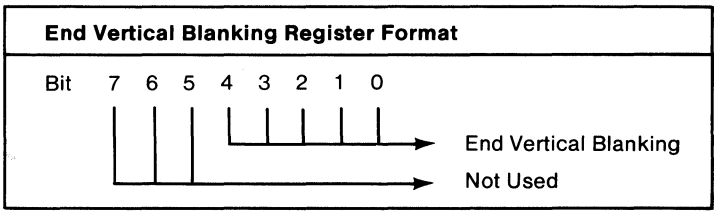
address for this register is hex 3B5 or hex 3D5.



Bit 0-Bit 7 Start Vertical Blank—These are the low 8 bits of the horizontal scan line count, at which the vertical blanking signal becomes active. Bit 8 bit is in the overflow register hex 07.

End Vertical Blanking Register

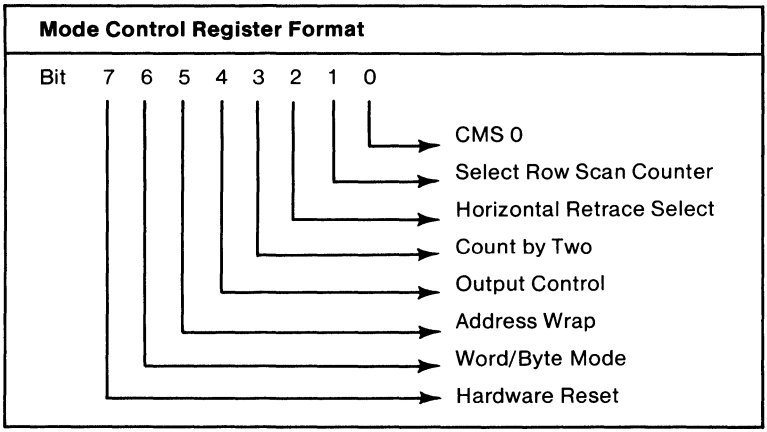
This is a write-only register pointed to when the value in the CRT Controller address register is hex 16. The processor output port address for this register is hex 3B5 or hex 3D5.



Bit 0-Bit 4 End Vertical Blank—This register specifies the horizontal scan count value when the vertical blank output signal becomes inactive. The register is programmed in units of horizontal scan lines. To obtain a vertical blank signal of width W, the following algorithm is used: Value of Start Vertical Blank Register + width of vertical blank signal in horizontal scan units = 5-bit result to be programmed into the End Vertical Blank Register.

Mode Control Register

This is a write-only register pointed to when the value in the CRT Controller address register is hex 17. The processor output port address for this register is hex 3B5 or hex 3D5.



Bit 0 Compatibility Mode Support— When this bit is a logical 0, the row scan address bit 0 is substituted for memory address bit 13 during active display time. A logical 1 enables memory address bit 13 to appear on the memory address output bit 13 signal of the CRT Controller. 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 CRTC was programmed for 100 horizontal scan lines with 2 row scan addresses per character row. Row scan address bit 0 became the most significant address bit to the display buffer. Successive scan lines of the display image were displaced in memory by 8K bytes. This bit allows compatibility with the 6845 and Color Graphics APA modes of operation.

Bit 1 Select Row Scan Counter—A logical 0 selects row scan counter bit 1 on MA 14 output pin. A logical 1 selects MA 14 counter bit on MA 14 output pin.

Bit 2 Horizontal Retrace Select—This bit 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 512 scan lines due to the 9-bit wide Vertical Total Register. If the vertical counter is clocked with the horizontal retrace divided by 2 clock, then the vertical resolution is doubled to 1024 horizontal scan lines. A logical 0 selects HRTC and a logical 1 selects HRTC divided by 2.

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

Bit 4 Output Control—A logical 0 enables the module output drivers. A logical 1 forces all outputs into high impedance state.

Bit 5 Address Wrap—This bit 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 you are not in the word address mode, MA 0 counter output appears on the MA 0 output pin. A logical 1 selects MA 15. In odd/even mode, bit MA 13 should be selected when the 64K memory is installed on the board. Bit MA 15 should be selected when greater then 64K memory is installed. This function is used to implement Color Graphics Monitor Adapter compatibility.

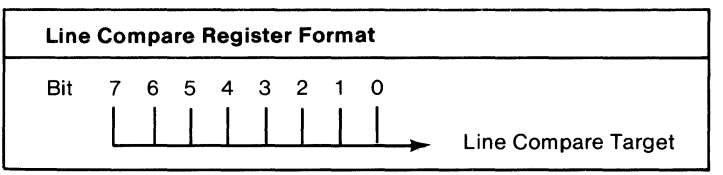
Bit 6 Word Mode or Byte Mode—When this bit is a logical 0, the Word Mode shifts all memory address counter bits down one bit, and the most significant bit of the counter appears on the least significant bit of the memory address outputs. See table below for address output details. A logical 1 selects the Byte Address mode.

	Internal Memory Address Counter Wiring to the Output Multiplexer	
CRTC Out Pin	**Byte Address Mode**	**Word Address Mode**
MA 0/RFA 0	MA 0	MA 15 or MA 13
MA 1/RFA 1	MA 1	MA 0
MA 2/RFA 2	MA 2	MA 1
MA 3/RFA 3	MA 3	MA 2
*	*	*
*	*	*
*	*	*
MA 14/RS 3	MA 14	MA 13
MA 15/RS 4	MA 15	MA 14

Bit 7 Hardware Reset—A logical 0 forces horizontal and vertical retrace to clear. A logical 1 forces horizontal and vertical retrace to be enabled.

Line Compare Register

This is a write-only register pointed to when the value in the CRT Controller address register is hex 18. The processor output port address for this register is hex 3B5 or hex 3D5.



Bit 0-Bit 7 Line Compare—This register is the low-order 8 bits of the compare target. When the vertical

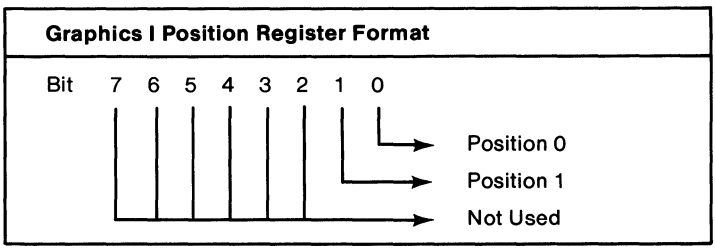
counter reaches this value, the internal start of the line counter is cleared. This allows an area of the screen to be immune to scrolling. Bit 8 of this register is in the overflow register hex 07.

Graphics Controller Registers

Name	Port	Index
Graphics 1 Position	3CC	-
Graphics 2 Position	3CA	-
Graphics 1 & 2 Address	3CE	-
Set/Reset	3CF	00
Enable Set/Reset	3CF	01
Color Compare	3CF	02
Data Rotate	3CF	03
Read Map Select	3CF	04
Mode Register	3CF	05
Miscellaneous	3CF	06
Color Don't Care	3CF	07
Bit Mask	3CF	08

Graphics 1 Position Register

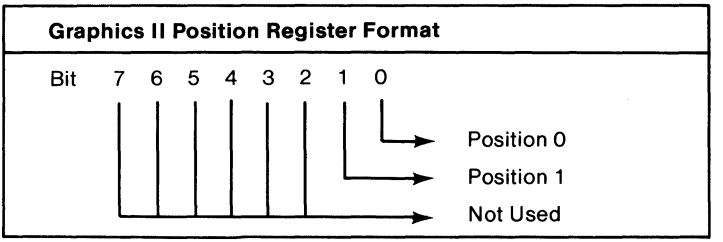
This is a write-only register. The processor output port address for this register is hex 3CC.



Bit 0-Bit 1 Position—These 2 bits are binary encoded hierarchy bits for the graphics chips. The position register controls which 2 bits of the processor data bus each chip responds to. Graphics 1 must be programmed with a position register value of 0 for this card.

Graphics 2 Position Register

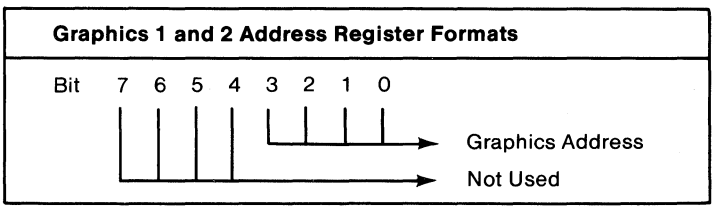
This is a write-only register. The processor output port address for this register is hex 3CA.



Bit 0-Bit 1 Position—These 2 bits are binary encoded hierarchy bits for the graphics chips. The position register controls which 2 bits of the processor data bus to which each chip responds. Graphics 2 must be programmed with a position register value of 1 for this card.

Graphics 1 and 2 Address Register

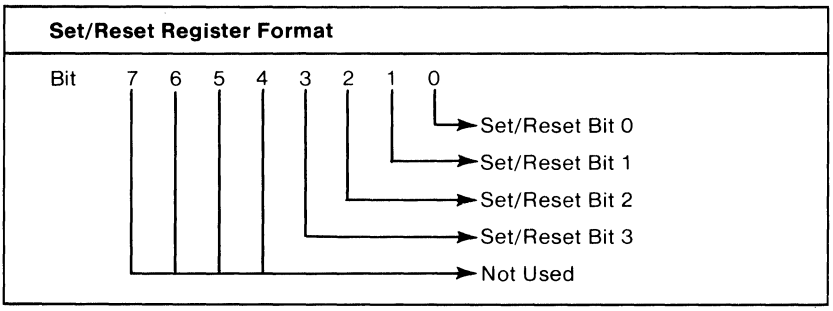
This is a write-only register and the processor output port address for this register is hex 3CE.



Bit 0-Bit 3 Graphics 1 and 2 Address Bits—This output loads the address register in both graphics chips simultaneously. This register points to the data register of the graphics chips.

Set/Reset Register

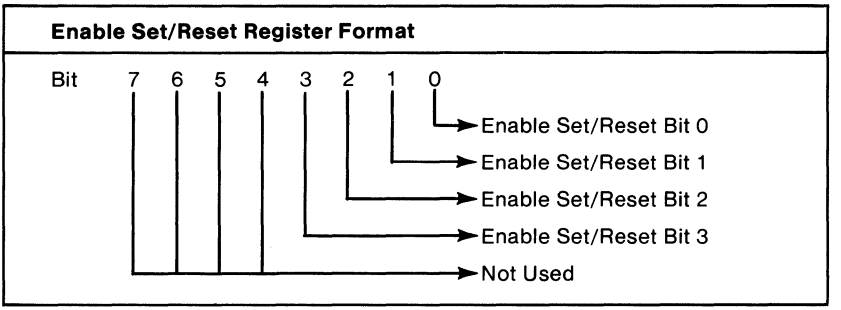
This is a write-only register pointed to by the value in the Graphics 1 and 2 address register. This value must be hex 00 before writing can take place. The processor output port address for this register is hex 3CF.



Bit 0-Bit 3 Set/Reset—These bits represent the value written to the respective memory planes when the processor does a memory write with write mode 0 selected and set/reset mode is enabled. Set/Reset can be enabled on a plane by plane basis with separate OUT commands to the Set/Reset register.

Enable Set/Reset Register

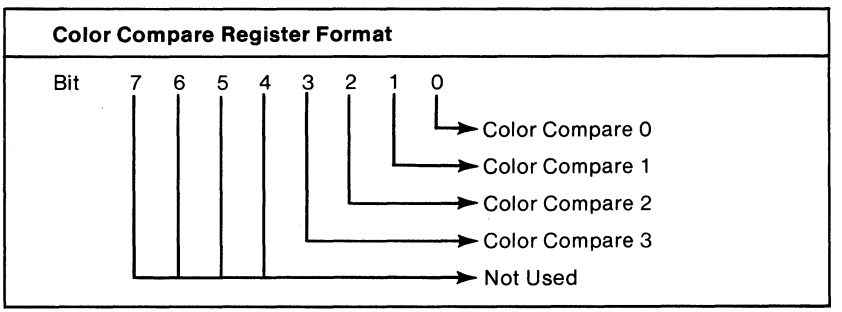
This is a write-only register and is pointed to by the value in the Graphics 1 and 2 address register. This value must be hex 01 before writing can take place. The processor output port for this register is hex 3CF.



Bit 0-Bit 3 Enable Set/Reset—These bits enable the set/reset function. The respective memory plane is written with the value of the Set/Reset register provided the write mode is 0. When write mode is 0 and Set/Reset is not enabled on a plane, that plane is written with the value of the processor data.

Color Compare Register

This is a write-only register pointed to by the value in the Graphics 1 and 2 address register. This value must be hex 02 before writing can take place. The processor output port address for this register is hex 3CF.

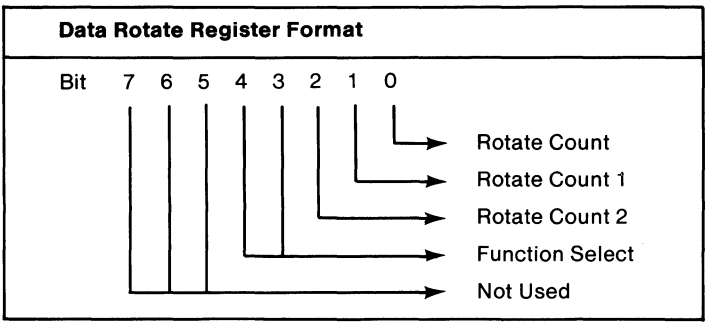


Bit 0-Bit 3 Color Compare—These bits represent a 4 bit color value to be compared. If the processor sets

read mode 1 on the graphics chips, and does a memory read, the data returned from the memory cycle will be a 1 in each bit position where the 4 bit planes equal the color compare register.

Data Rotate Register

This is a write-only register pointed to by the value in the Graphics 1 and 2 address register. This value must be hex 03 before writing can take place. The processor output port address for this register is hex 3CF.



Bit 0-Bit 2 Rotate Count—These bits represent a binary encoded value of the number of positions to rotate the processor data bus during processor memory writes. This operation is done when the write mode is 0. To write unrotated data the processor must select a count of 0.

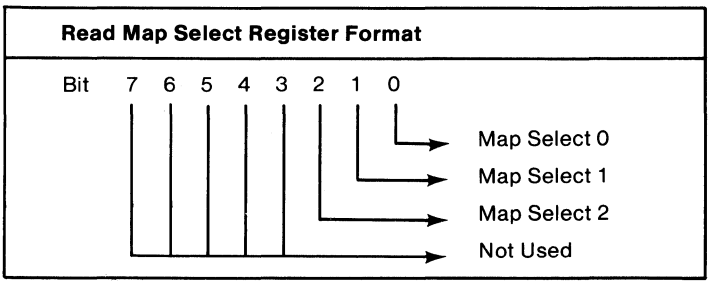
Bit 3-Bit 4 Function Select—Data written to memory can operate logically with data already in the processor latches. The bit functions are defined in the following table.

Bits 4 3	
0 0	Data unmodified.
0 1	Data AND'ed with latched data.
1 0	Data OR'ed with latched data.
1 1	Data XOR'ed with latched data.

Data may be any of the choices selected by the Write Mode Register except processor latches. If rotated data is selected, the rotate applies before the logical function.

Read Map Select Register

This is a write-only register pointed to by the value in the Graphics 1 and 2 address register. This value must be hex 04 before writing can take place. The processor output port address for this register is hex 3CF.

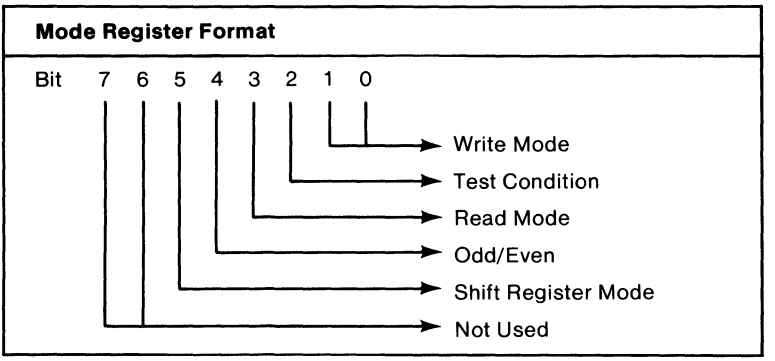


Bit 0-Bit 2 Map Select—These bits represent a binary encoded value of the memory plane number from which the processor reads data. This register has no effect on the color compare read mode described elsewhere in this section.

Mode Register

This is a write-only register pointed to by the value in the Graphics 1 and 2 address register. This value must be hex 05

before writing can take place. The processor output port address for this register is 3CF.



Bit 0-Bit 1 Write Mode

Bits 1 0	
0 0	Each memory plane is written with the processor data rotated by the number of counts in the rotate register, unless Set/Reset is enabled for the plane. Planes for which Set/Reset is enabled are written with 8 bits of the value contained in the Set/Reset register for that plane.
0 1	Each memory plane is written with the contents of the processor latches. These latches are loaded by a processor read operation.
1 0	Memory plane *n* (0 through 3) is filled with 8 bits of the value of data bit *n*.
1 1	Not Valid

The logic function specified by the function select register also applies.

Bit 2 Test Condition—A logical 1 directs graphics controller outputs to be placed in high impedance state for testing.

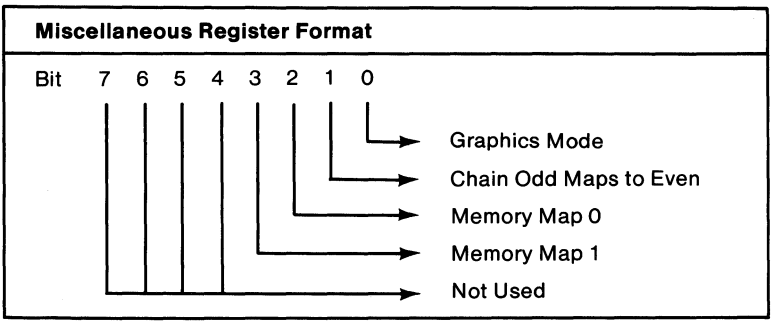
Bit 3 Read Mode—When this bit is a logical 0, the processor reads data from the memory plane selected by the read map select register. When this bit is a logical 1, the processor reads the results of the comparison of the 4 memory planes and the color compare register.

Bit 4 Odd/Even—A logical 1 selects the odd/even addressing mode, which is useful for emulation of the Color Graphics Monitor Adapter compatible modes. Normally the value here follows the value of the Memory Mode Register bit 3 of the Sequencer.

Bit 5 Shift Register—A logical 1 directs the shift registers on each graphics chip to format the serial data stream with even numbered bits on the even numbered maps and odd numbered bits on the odd maps.

Miscellaneous Register

This is a write-only register pointed to by the value in the Graphics 1 and 2 address register. This value must be hex 06 before writing can take place. The processor output port for this register is hex 3CF.



Bit 0 Graphics Mode—This bit controls alpha-mode addressing. A logical 1 selects graphics mode. When set to graphics mode, the character generator address latches are disabled.

Bit 1 Chain Odd Maps To Even Maps—When set to 1, this bit directs the processor address bit 0 to be replaced by a higher order bit and odd/even maps to be selected with odd/even values of the processor A0 bit, respectively.

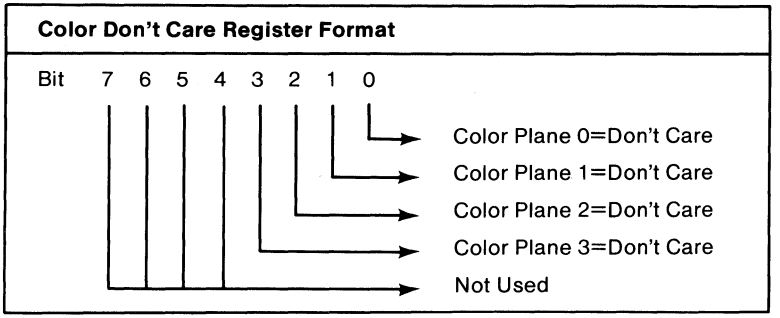
Bit 2-Bit 3 Memory Map—These bits control the mapping of the regenerative buffer into the processor address space.

Bits 3 2	
0 0	Hex A000 for 128K bytes.
0 1	Hex A000 for 64K bytes.
1 0	Hex B000 for 32K bytes
1 1	Hex B800 for 32K bytes.

If the display adapter is mapped at address hex A000 for 128K bytes, no other adapter can be installed in the system.

Color Don't Care Register

This is a write-only register and is pointed to by the value in the Graphics 1 and 2 address register. This value must be hex 07 before writing can take place. The processor output port for this register is hex 3CF.



Bit 0 Color Don't Care—Color plane 0=don't care when reading color compare when this bit is set to 1.

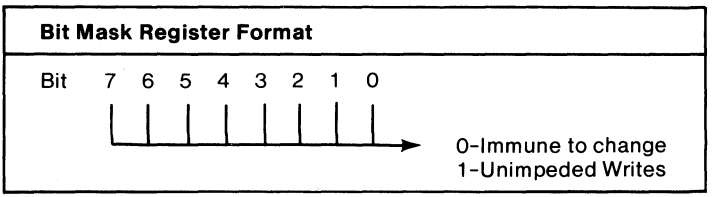
Bit 1 Color Don't Care—Color plane 1=don't care when reading color compare when this bit is set to 1.

Bit 2 Color Don't Care—Color plane 2=don't care when reading color compare when this bit is set to 1.

Bit 3 Color Don't Care—Color plane 3=don't care when reading color compare when this bit is set to 1.

Bit Mask Register

This is a write-only register and is pointed to by the value in the Graphics 1 and 2 address register. This value must be hex 08 before writing can take place. The processor output port for this register is hex 3CF.



Bit 0-Bit 7 Bit Mask—Any bit programmed to *n* causes the corresponding bit *n* in each bit plane to be immune to change provided that the location being written was the last location read by the processor. Bits programmed to a 1 allow unimpeded writes to the corresponding bits in the bit planes.

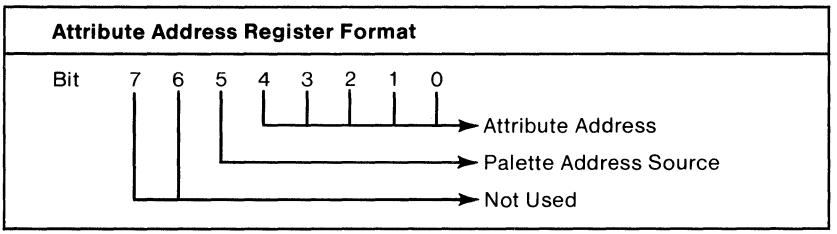
The bit mask applies to any data written by the processor (rotate, AND'ed, OR'ed, XOR'ed, DX and S/R). To preserve bits using the bit mask, data must be latched internally by reading the location. When data is written to preserve the bits, the most current data in latches is written in those positions. The bit mask applies to all bit planes simultaneously.

Attribute Controller Registers

Name	Port	Index
Address Register	3C0	-
Palette Registers	3C0	00-0F
Mode Control Register	3C0	10
Overscan Color Register	3C0	11
Color Plane Enable Register	3C0	12
Horizontal Pel Panning Register	3C0	13

Attribute Address Register

This is a write-only register. The processor output port is hex 3C0.



Bit 0-Bit 4 Attribute Address Bits—The Address Register is a pointer register located at hex 3C0. This register is loaded with a binary value that points to the attribute data register where data is to be written. The Attribute Controller does not have an address bit input to control selection of the address and data registers. An internal address flip-flop controls selection of either the address or data registers. To initialize the flip-flop, an IOR instruction is issued to the Attribute Controller at address 3BA or 3DA. This clears the flip-flop, and selects the Address Register. After the Address Register has been loaded, the next OUT instruction loads the data register.

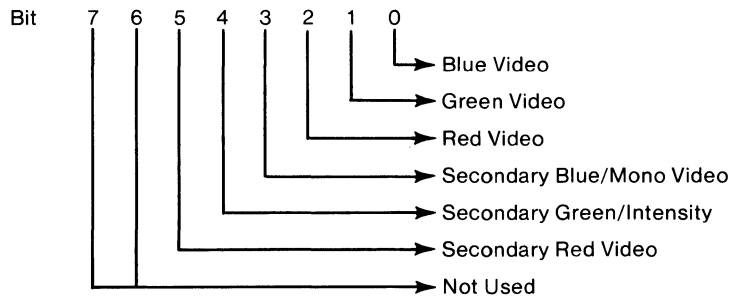
The flip-flop toggles each time an OUT is issued to the Attribute Controller.

Bit 5 Palette Address Source—When loading the color palette registers, bit 5 must be cleared to 0. To enable the memory data to access the color palette, bit 5 must be set to 1.

Palette Register Hex 00 through Hex 0F

This is a write-only register. The processor output port is hex 3C0.

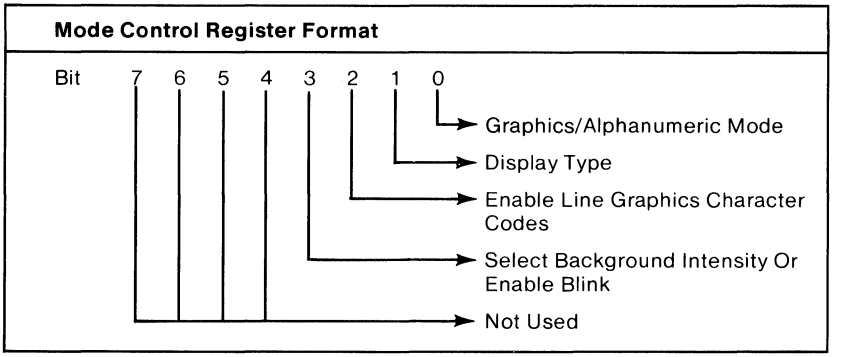
Palette Registers Hex 00 through Hex 0F Format



Bit 0-Bit 5 Palette—These 6-bit registers allow a dynamic mapping between the text attribute or graphic color input value and the display color in the CRT screen. A logical 1 selects the appropriate color. A logical 0 de-selects. The color palette register should be modified only during the vertical retrace interval to avoid glitches in the displayed image. Note that some color monitors do not have an intensity input and only a maximum of eight colors are available. Monitors with four color inputs display sixteen colors, and monitors with six color inputs display 64 colors.

Mode Control Register

This is a write-only register pointed to by the value in the Attribute address register. This value must be hex 10 before writing can take place. The processor output port address for this register is hex 3C0.



Bit 0 Graphics/Alphanumeric Mode—A logical 0 selects alphanumeric mode. A logical 1 selects graphics mode.

Bit 1 Monochrome Display/Color Display—A logical 0 selects IBM monochrome display attributes. A logical 1 selects color Display attributes.

Bit 2 Enable Line Graphics Character Codes—When this bit is set to 0, the ninth dot will be the same as the background. A logical 1 enables the special line graphics character codes for the IBM Monochrome Display adapter. 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 Display Adapter are Hex C0 through Hex DF.

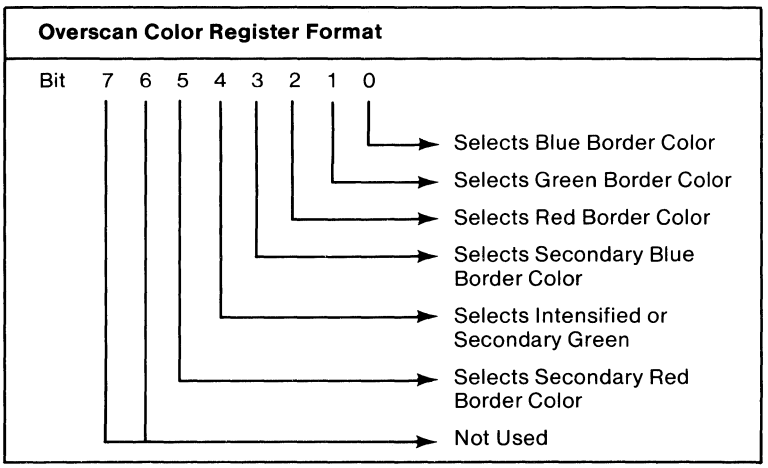
For character fonts that do not utilize the line graphics character codes in the range of Hex C0

through Hex DF, bit 2 of this register should be a logical 0. Otherwise unwanted video information will be displayed on the CRT screen.

Bit 3 Enable Blink/Select Background Intensity—A logical 0 selects the background intensity of the attribute input. This mode was available on the Monochrome and Color Graphics adapters. A logical 1 enables the blink attribute in alphanumeric modes. This bit must also be set to 1 for blinking graphics modes.

Overscan Color Register

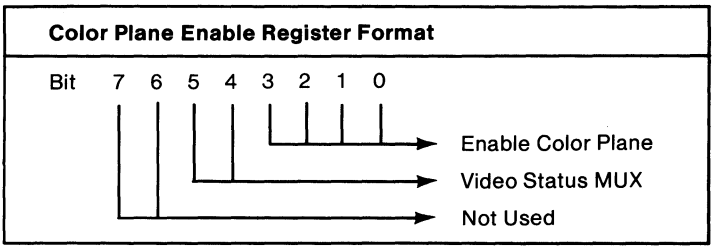
This is a write-only register pointed to by the value in the Attribute address register. This value must be hex 11 before writing can take place. The processor output port address for this register is hex 3C0.



Bit 0-Bit 5 Overscan Color—This 6-bit register determines the overscan (border) color displayed on the CRT screen. For monochrome display this register should be set to a value of 0. A logical 1 selects the appropriate color.

Color Plane Enable Register

This is a write-only register pointed to by the value in the Attribute address register. This value must be hex 12 before writing can take place. The processor output port address for this register is 3C0.



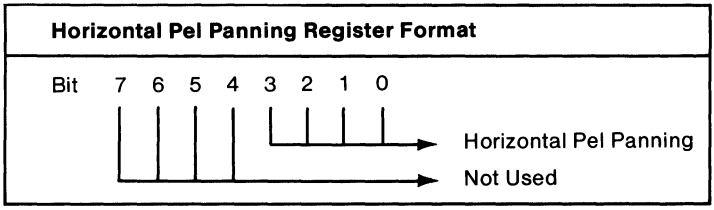
Bit 0-Bit 3 Enable Color Plane—Writing a logical 1 in any of bits 0 through 3 enables the respective display memory color plane.

Bit 4-Bit 5 Video Status MUX—Selects two of the six color outputs to be available on the status port. The following table illustrates the combinations available and the color output wiring.

COLOR PLANE ENABLE REGISTER		INPUT STATUS REGISTER ONE	
Bit 5	**Bit 4**	**Bit 5**	**Bit 4**
0	0	Red	Blue
0	1	Secondary Blue	Green
1	0	Secondary Red	Secondary Green
1	1	Not Used	Not Used

Horizontal Pel Panning Register

This is a write-only register pointed to by the value in the Attribute address register. This value must be hex 12 before writing can take place. The processor output port address for this register is hex 3C0.



Bit 0-Bit 3 Horizontal Pel Panning—This 4 bit register selects the number of picture elements (pels) to shift the video data horizontally to the left. Pel panning is available in both A/N and APA modes. In Monochrome A/N mode, the image can be shifted a maximum of 9 pels. In all other A/N and APA modes, the image can be shifted a maximum of 8 pels. The sequence for shifting the image is given below:

9 pels/character : 8, 0, 1, 2, 3, 4, 5, 6, 7 (Monochrome A/N mode only)

8 pels/character : 0, 1, 2, 3, 4, 5, 6, 7 (All other Modes)

Programming Considerations

Programming the Registers

Each of the LSI devices has an address register and a number of data registers. The address register serves as a pointer to the other registers on the LSI device. It is a write-only register that is loaded by the processor by executing an 'OUT' instruction to its I/O address with the index of the selected data register.

The data registers on each LSI device 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 external registers that are not part of an LSI device and the Graphics I and II registers are not accessed through an address register; they are written to directly.

The following tables define the values that are loaded into the registers by BIOS to support the different modes of operation supported by this adapter.

Register			Mode of Operation																	
Name	Port	Index	0	1	2	3	4	5	6	7	D	E	F	10	F‡	10‡	0*	1*	2*	3*
Miscellaneous	3C2	–	23	23	23	23	23	23	23	A6	23	23	A2	A7	A2	A7	A7	A7	A7	A7
Feature Cntrl	3?A	–	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00
Input Stat 0	3C2	–	–	–	–	–	–	–	–	–	–	–	–	–	–	–	–	–	–	–
Input Stat 1	3?2	–	–	–	–	–	–	–	–	–	–	–	–	–	–	–	–	–	–	–
? = B in monochrome modes ? = D in color modes																				
*Values for these modes when the IBM Enhanced Color Display is attached																				
‡Values for these modes when greater than 64K Graphics Memory is installed																				

External Registers

Register			Mode of Operation																	
Name	Port	Index	0	1	2	3	4	5	6	7	D	E	F	10	F‡	10‡	0*	1*	2*	3*
Seq Address	3C4	–	–	–	–	–	–	–	–	–	–	–	–	–	–	–	–	–	–	–
Reset	3C5	00	03	03	03	03	03	03	03	03	03	03	03	03	03	03	03	03	03	03
Clock Mode	3C5	01	0B	0B	01	01	0B	0B	01	00	0B	01	05	05	01	01	0B	0B	01	01
Map Mask	3C5	02	03	03	03	03	03	03	01	03	0F	0F	0F	0F	0F	0F	03	03	03	03
Char Gen Sel	3C5	03	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00
Memory Mode	3C5	04	03	03	03	03	02	02	06	03	06	06	00	00	06	06	03	03	03	03
*Values for these modes when the IBM Enhanced Color Display is attached																				
‡Values for these modes when greater than 64K Graphics Memory is installed																				

Sequencer Registers

Register			Mode of Operation																	
Name	Port	Index	0	1	2	3	4	5	6	7	D	E	F	10	F‡	10‡	0*	1*	2*	3*
Address Reg	3?4	–	–	–	–	–	–	–	–	–	–	–	–	–	–	–	–	–	–	–
Horiz Total	3?5	00	37	37	70	70	37	37	70	60	37	70	60	5B	60	5B	2D	2D	5B	5B
Hrz Disp End	3?5	01	27	27	4F	4F	27	27	4F	4F	27	4F	4F	4F	4F	4F	27	27	4F	4F
Strt Hrz Blk	3?5	02	2D	2D	5C	5C	2D	2D	59	56	2D	56	56	53	56	53	2B	2B	53	53
End Hrz Blk	3?5	03	37	37	2F	2F	37	37	2D	3A	37	2D	1A	17	3A	37	2D	2D	37	37
Strt Hrz Retr	3?5	04	31	31	5F	5F	30	30	5E	51	30	5E	50	50	50	52	28	28	51	51
End Hrz Retr	3?5	05	15	15	07	07	14	14	06	60	14	06	E0	BA	60	00	6D	6D	5B	5B
Vert Total	3?5	06	04	04	04	04	04	04	04	70	04	04	70	6C	70	6C	6C	6C	6C	6C
Overflow	3?5	07	11	11	11	11	11	11	11	1F	11	11	1F	1F	1F	1F	1F	1F	1F	1F
Preset Row SC	3?5	08	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00
Max Scan Line	3?5	09	07	07	07	07	01	01	01	0D	00	00	00	00	00	00	0D	0D	0D	0D
Cursor Start	3?5	0A	06	06	06	06	00	00	00	0B	00	00	00	00	00	00	0B	0B	0B	0B
Cursor End	3?5	0B	07	07	07	07	00	00	00	0C	00	00	00	00	00	00	0C	0C	0C	0C
Strt Addr Hi	3?5	0C	–	–	–	–	–	–	–	–	–	–	–	–	–	–	–	–	–	–
Strt Addr Lo	3?5	0D	–	–	–	–	–	–	–	–	–	–	–	–	–	–	–	–	–	–
? = B in monochrome modes ? = D in color modes																				
*Values for these modes when the IBM Enhanced Color Display is attached																				
‡Values for these modes when greater than 64K Graphics Memory is installed																				

CRT Controller Registers (1 of 2)

Register			Mode of Operation																	
Name	Port	Index	0	1	2	3	4	5	6	7	D	E	F	10	F‡	10‡	0*	1*	2*	3*
Cursor LC Hi	3?5	0E	–	–	–	–	–	–	–	–	–	–	–	–	–	–	–	–	–	–
Cursor LC Low	3?5	0F	–	–	–	–	–	–	–	–	–	–	–	–	–	–	–	–	–	–
Vrt Retr Strt	3?5	10	E1	E1	E1	E1	E1	E1	E0	5E	E1	E0	5E	5E	5E	5E	5E	5E	5E	5E
Light Pen Hi	3?5	10	–	–	–	–	–	–	–	–	–	–	–	–	–	–	–	–	–	–
Vert Retr End	3?5	11	24	24	24	24	24	24	23	2E	24	23	2E	2B	2E	2B	2B	2B	2B	2B
Light Pen Low	3?5	11	–	–	–	–	–	–	–	–	–	–	–	–	–	–	–	–	–	–
Vrt Disp End	3?5	12	C7	C7	C7	C7	C7	C7	C7	5D	C7	C7	5D	5D	5D	5D	5D	5D	5D	5D
Offset	3?5	13	14	14	28	28	14	14	28	28	14	28	14	14	28	28	14	14	28	28
Underline Loc	3?5	14	08	08	08	08	00	00	00	0D	00	00	0D	0F	0D	0F	0F	0F	0F	0F
Strt Vert Blk	3?5	15	E0	E0	E0	E0	E0	E0	DF	5E	E0	DF	5E	5F	5E	5F	5E	5E	5E	5E
End Vert Blk	3?5	16	F0	F0	F0	F0	F0	F0	EF	6E	F0	EF	6E	0A	6E	0A	0A	0A	0A	0A
Mode Control	3?5	17	A3	A3	A3	A3	A2	A2	C2	A3	E3	E3	8B	8B	E3	E3	A3	A3	A3	A3
Line Compare	3?5	18	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF
? = B in monochrome modes ? = D in color modes																				
*Values for these modes when the IBM Enhanced Color Display is attached																				
‡Values for these modes when greater than 64K Graphics Memory is installed																				

CRT Controller Registers (2 of 2)

Register			Mode of Operation																	
Name	Port	Index	0	1	2	3	4	5	6	7	D	E	F	10	F‡	10‡	0*	1*	2*	3*
Grphx I Pos	3CC	–	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00
Grphx II Pos	3CA	–	01	01	01	01	01	01	01	01	01	01	01	01	01	01	01	01	01	01
Grphx I II AD	3CE	–	–	–	–	–	–	–	–	–	–	–	–	–	–	–	–	–	–	–
Set Reset	3CF	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00
Enable S/R	3CF	01	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00
Color Compare	3CF	02	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00
Data Rotate	3CF	03	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00
Read Map Sel	3CF	04	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00
Mode Register	3CF	05	10	10	10	10	30	30	00	10	00	00	10	10	00	00	10	10	10	10
Miscellaneous	3CF	06	0E	0E	0E	0E	0F	0F	0D	0A	05	05	07	07	05	05	0E	0E	0E	0E
Color No Care	3CF	07	00	00	00	00	00	00	00	00	0F	0F	0F	0F	0F	0F	00	00	00	00
Bit Mask	3CF	08	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF
*Values for these modes when the IBM Enhanced Color Display is attached																				
‡Values for these modes when greater than 64K Graphics Memory is installed																				

Graphics SI Registers

Register			Mode of Operation																	
Name	Port	Index	0	1	2	3	4	5	6	7	D	E	F	10	F‡	10‡	0*	1*	2*	3*
Address	3?A	–	–	–	–	–	–	–	–	–	–	–	–	–	–	–	–	–	–	–
Palette	3C0	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00
Palette	3C0	01	01	01	01	01	13	13	17	08	01	01	08	01	08	01	01	01	01	01
Palette	3C0	02	02	02	02	02	15	15	17	08	02	02	00	00	00	02	02	02	02	02
Palette	3C0	03	03	03	03	03	17	17	17	08	03	03	00	00	00	03	03	03	03	03
Palette	3C0	04	04	04	04	04	02	02	17	08	04	04	18	04	18	04	04	04	04	04
Palette	3C0	05	05	05	05	05	04	04	17	08	05	05	18	07	18	05	05	05	05	05
Palette	3C0	06	06	06	06	06	06	06	17	08	06	06	00	00	00	06	14	14	14	14
Palette	3C0	07	07	07	07	07	07	07	17	08	07	07	00	00	00	07	07	07	07	07
Palette	3C0	08	10	10	10	10	10	10	17	10	10	10	00	00	00	38	38	38	38	38
Palette	3C0	09	11	11	11	11	11	11	17	18	11	11	08	01	08	39	39	39	39	39
Palette	3C0	0A	12	12	12	12	12	12	17	18	12	12	00	00	00	3A	3A	3A	3A	3A
Palette	3C0	0B	13	13	13	13	13	13	17	18	13	13	00	00	00	3B	3B	3B	3B	3B
? = B in monochrome modes ? = D in color modes																				
*Values for these modes when the IBM Enhanced Color Display is attached																				
‡Values for these modes when greater than 64K Graphics Memory is installed																				

Attribute Registers (1 of 2)

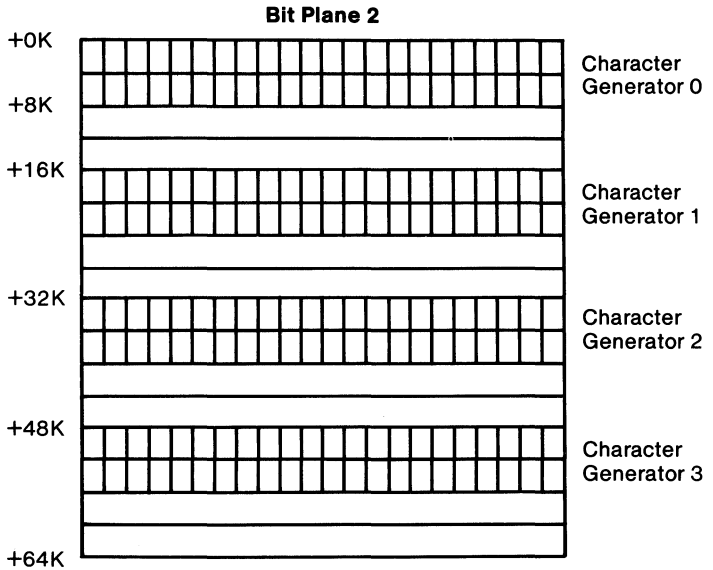
Register			Mode of Operation																
Name	Port	Index	0	1	2	3	4	5	6	7	D	E	F	10	F‡	10‡	0*	1*	2*	3*
Palette	3C0	0C	14	14	14	14	14	14	17	18	14	14	00	04	00	3C	3C	3C	3C	3C
Palette	3C0	0D	15	15	15	15	15	15	17	18	15	15	18	07	18	3D	3D	3D	3D	3D
Palette	3C0	0E	16	16	16	16	16	16	17	18	16	16	00	00	00	3E	3E	3E	3E	3E
Palette	3C0	0F	17	17	17	17	17	17	18	17	17	00	00	00	3F	3F	3F	3F	3F	3F
Mode Control	3C0	10	08	08	08	08	01	01	01	0E	01	01	0B	0B	0B	01	08	08	08	08
Overscan	3C0	11	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00
Color Plane	3C0	12	0F	0F	0F	0F	03	03	01	0F	0F	0F	05	05	05	0F	0F	0F	0F	0F
Hrz Panning	3C0	13	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00
*Values for these modes when the IBM Enhanced Color Display is attached																				
‡Values for these modes when greater than 64K Graphics Memory is installed																				

Attribute Registers (2 of 2)

RAM Loadable Character Generator

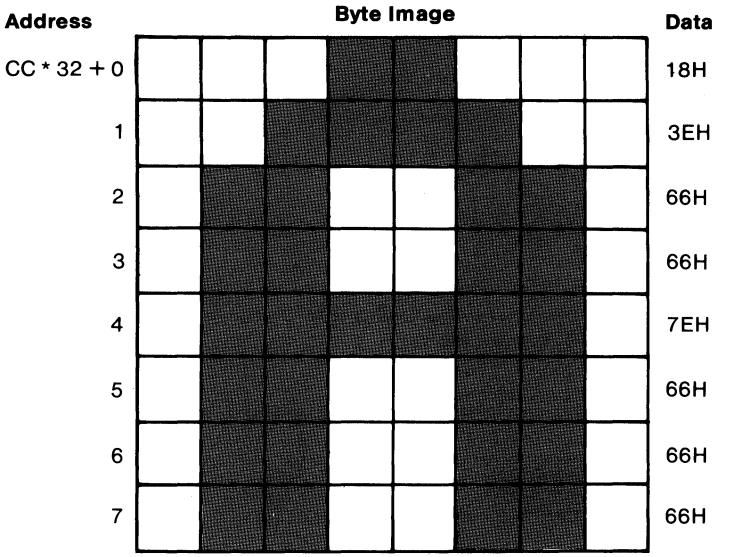
The character generator on the adapter is RAM loadable and can support characters up to 32 scan lines high. Two character generators are stored within the BIOS and one is automatically loaded into the RAM by the BIOS when an alphanumeric mode is selected. The Character Map Select Register can be programmed 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 bit plane 2. This effectively gives the user access to 512 characters instead of 256. character tables may be loaded off line.The adapter must have 128K bytes of storage to support this function. Up to four tables can be loaded can be loaded with 256K of graphics memory installed.

The structure of the character tables is described in the following figure. The character generator is in bit plane 2 and must be protected using the map mask function.



The following figure illustrates 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.



CC = Value of the character code. For example, 41H in the case of an ASCII "A".

Creating a 512 Character Set

This section describes how to create a 512 character set on the IBM Color Display. Note that only 256 characters can be printed on the printer. This is a special application which the Enhanced Graphics Adapter will support. The 9 by 14 characters will be displayed when attribute bit 3 is a logical 0, and the IBM Color/Graphics Monitor Adapter 8 by 8 characters will be displayed when the attribute bit 3 is a logical 1. This example is for demonstrative purposes only. The assembly language routine for creating 512 characters is given below. Debug 2.0 was used for this example. The starting assembly address is 100 and the character string is stored in location 200. This function requires 128K or more of graphics memory.

```
a100
mov ax,1102          ;load 8x8 character font in character
mov bl,02            ;generator number 2
int 10

mov ax,1103          ;select 512 character operation
mov bl,08            ;if attribute bit 3=1 use 8x8 font
int 10               ;if attribute bit 3=0 use 9x14 font

mov ax,1000          ;set color plane enable to 7H to disable
mov bx,0712          ;attribute bit 3 in the color palette
int 10               ;lookup table

mov ax,1301
mov bx,000F          ;write char. string with attribute bit 3=1
mov cx,003A          ;cx = character string length
mov dx,1600          ;write character on line 22 of display
mov bp,0200          ;pointer to character string location
push cs
pop es
int 10

mov ax ,1301
mov bx,0007          ;write char. string with attribute bit 3=0
mov cx,003A          ;cx = character string length
mov dx,1700          ;write character on line 23 of display
mov bp,0200          ;pointer to character string location
push cs
pop es
int 10
int 3

a200 db              "This character string is used to show 512
                     characters"
```

Creating an 80 by 43 Alphanumeric Mode

The following examples show how to create 80 column by 43 row, both alphanumeric and graphics, images on the IBM Monochrome Display. The BIOS Interface supports an 80 column by *n* row display by using the character generator load routine call. The print screen routine must be revectored to

handle the additional character rows on the screen. The assembly language required for both an alphanumeric and a graphics screen is shown below.

```
mov al,7          ;Monochrome alphanumeric mode
int 10            ;video interrupt call
mov ax,1112       ;character generator BIOS routine
mov bl,0          ;load 8 by 8 double dot character font
int 10            ;video interrupt call
mov ax,1200       ;alternate screen routine
move bl,20        ;select alternate print screen routine
int 10            ;video interrupt call
int 3

mov ax,f          ;Monochrome graphic mode
int 10            ;video interrupt call
mov ax,1123       ;character generator BIOS routine
mov bl,0          ;load 8 by 8 double dot character font
mov dl,2B         ;43 character rows
int 10            ;video interrupt call
mov ax,1200       ;alternate screen routine
mov bl,20         ;alternate print screen routine
int 10            ;video interrupt call
int 3
```

Vertical Interrupt Feature

The Enhanced Graphics Adapter can be programmed to create an interrupt each time the vertical display refresh time has ended. An interrupt handler routine must be written by the application to take advantage of this feature. The CRT Vertical interrupt is on IRQ2. The CPU can poll the Enhanced Graphics Adapter Input Status Register 0 (bit 7) to determine whether the CRTC caused the interrupt to occur.

The Vertical Retrace End Register (11H) in the CRT controller contains two bits which are used to control the interrupt circuitry. The remaining bits must be output as per the value in the mode table.

Bit 5 Enable Vertical Interrupt—A logical 0 will enable vertical interrupt.

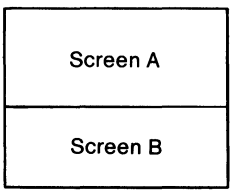
Bit 4 Clear Vertical Interrupt—A logical 0 will clear a vertical interrupt.

The sequence of events which occur in an interrupt handler are outlined below.

1. Clear IRQ latch and enable driver
2. Enable IRQ latch
3. Wait for vertical interrupt
4. Poll Interrupt Status Register 0 to determine if CRTC has caused the interrupt
5. If CRTC interrupt, then clear IRQ latch; if not, then branch to next interrupt handler.
6. Enable IRQ latch
7. Update Enhanced Graphics Adapter during vertical blanking interval
8. Wait for next vertical interrupt

Creating a Split Screen

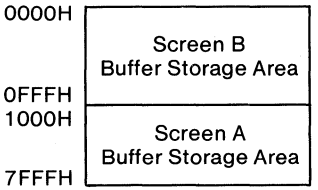
The Enhanced Graphics Adapter hardware supports an alphanumeric mode dual screen display. The top portion of the screen is designated as screen A, and the bottom portion of the screen is designated as screen B as per the following figure.



Dual Screen Definition

The following figure shows the screen mapping for a system containing a 32K byte alphanumeric storage buffer. Note that the Enhanced Graphics Adapter has a 32K byte storage buffer in alphanumeric mode. Information displayed on screen A is

defined by the start address high and low registers (0CH and 0DH) of the CRTC. Information displayed on screen B always begins at address 0000H.



Screen Mapping Within the Display Buffer Address Space

The Line Compare Register (18H) of the CRT Controller is utilized to perform the split screen function. The CRTC has an internal horizontal scan counter, and logic which compares the horizontal scan 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 zero, and each subsequent row address is 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 screen by updating the Line compare in synchronization with the vertical retrace signal. The information on screen B is immune from scrolling operations which utilize the Start Address High and Low registers to scroll through the Screen A address map.

Compatibility Issues

The CRT Controller on the IBM Enhanced Graphics Adapter is a custom design, and is different than the 6845 controller used on the IBM Monochrome Monitor Adapter and the IBM Color/Graphics Monitor Adapter. It should be noted that several CRTC register addresses differ between the adapters. The following figure illustrates the registers which do not map directly across the two controllers.

Register	6485 Function	EGA CRTC Function
02H	Start Horiz. Retrace	Start Horiz. Blanking
03H	End Horiz. Retrace	End Horiz. Blanking
04H	Vertical Total	Start Horiz. Retrace
05H	Vertical Total Adjust	End Horiz. Retrace
06H	Vertical Displayed	Vertical Total
07H	Vertical Sync Position	Overflow
08H	Interlace Mode and Skew	Preset Row Scan

Existing applications which utilize the BIOS interface will generally be compatible with the Enhanced Graphics Adapter.

Horizontal screen centering was required on the IBM Color/Graphics Monitor Adapter in order to center the screen when generating composite video. This was done through the Horizontal Sync Position Register. Since the Enhanced Graphics Adapter does not support a composite video monitor, programs which do screen centering may cause loss of the screen image if centering is attempted.

The Enhanced Graphics Adapter offers a wider variety of displayable monochrome character attributes than the IBM Monochrome Display Adapter. Some attribute values may display differently between the two Adapters. The values listed in the table below, in any combinations with the blink and intensity attributes, will display identically.

Background R G B	Foreground R G B	Function
0 0 0	0 0 0	Non-Display
0 0 0	0 0 1	Underline
0 0 0	1 1 1	White Character/Black Background
1 1 1	0 0 0	Reverse Video

Software which explicitly addresses 3D8 (Mode Select Register) or 3D9 (Color Select Register) on the Color Graphics Monitor Adapter may produce different results on the Enhanced Graphics Adapter. For example, blinking which is disabled by writing to 3D8 on the Color Graphics Adapter will not be disabled on the Enhanced Graphics Adapter.

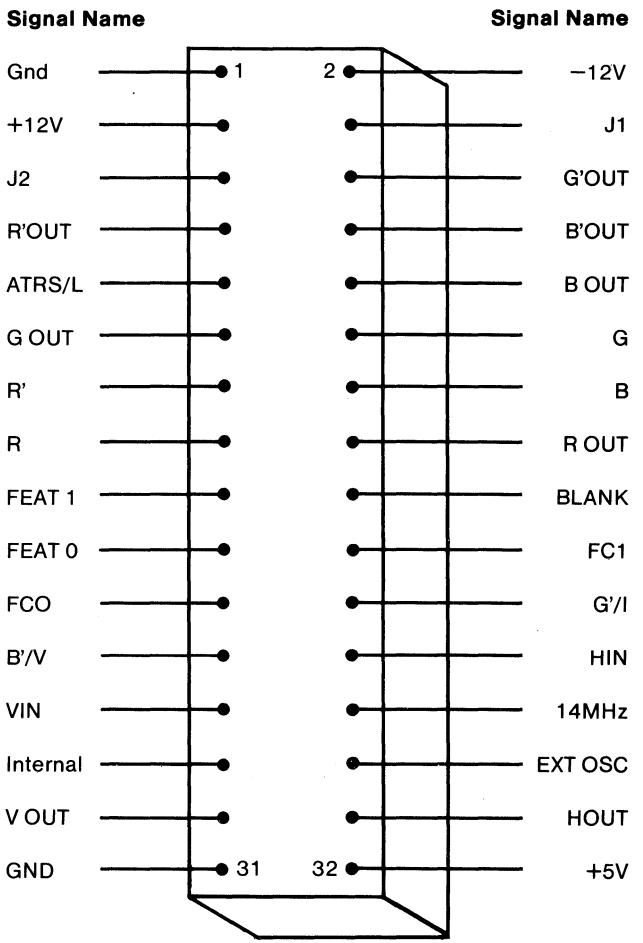
Feature Connector

The following is a description of the Enhanced Graphics Adapter feature connector. Note that signals coming from the Enhanced Graphics Adapter are labeled "inputs" and the signals coming to the Enhanced Graphics Adapter through the feature connector are labeled "outputs".

Signal	Description
J2	This pin is connected to auxiliary jack 2 on the rear panel of the adapter.
R'OUT	Secondary red output
ATRS/L	Attribute shift load. This signal controls the serialization of the video information. The shift register parallel loads at the dot clock leading edge when this signal is low.
G OUT	Primary green output
R'	Secondary red input
R	Primary red input
FC1	This signal is input from bit 1 (Feature Control Bit 1) of the Feature Control Register.
FC0	This signal is input from bit 0 (Feature Control Bit 0) of the Feature control Register.
FEAT 0	This signal is output to bit 5 (Feature Code 0) of Input Status Register 0.
B'/V	Secondary blue input/Monochrome video
VIN	Vertical retrace input

Internal	This signal is output to bit 4 (Disable Internal Video Drivers) of the Miscellaneous Output Register.
V OUT	Vertical retrace output
J1	This pin is connected to auxiliary jack 1 on the rear panel of the adapter.
G'OUT	Secondary green output
B'OUT	Secondary blue output
B OUT	Blue output
G	Green input
B	Blue input
R OUT	Red output
BLANK	This is a composite horizontal and vertical blanking signal from the CRTC.
FEAT 1	This signal is output to bit 6 (Feature Code 1) of Input Status Register 0.
G'/I	Secondary green/Intensity input
HIN	Horizontal retrace input from the CRTC
14MHZ	14 MHz signal from the system board
EXT OSC	External dot clock output
HOUT	Horizontal retrace output

The following figure shows the layout and pin numbering of the feature connector.

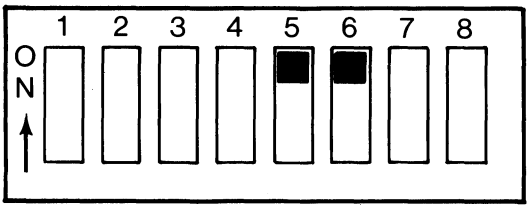


Feature Connector Diagram

Specifications

System Board Switches

The following figure shows the proper system board DIP switch settings for the IBM Enhanced Graphics Adapter when used with the Personal Computer and the Personal Computer XT. The switch block locations are illustrated in the Technical Reference Manual "System Board Component Diagram". The Personal Computer has two DIP switch blocks; the switch settings shown pertain to DIP Switch Block 1. The Personal Computer XT has one DIP switch block.

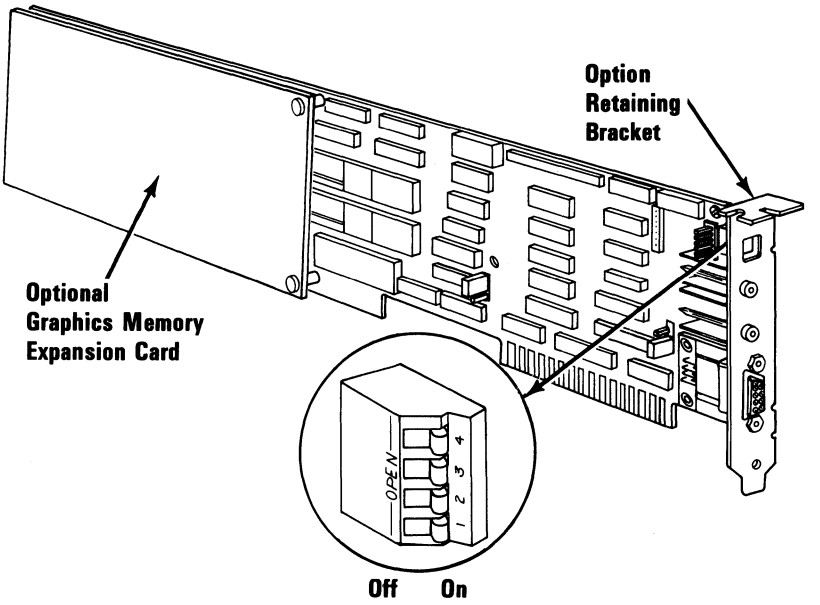


Switch Block (1)

Note: The DIP switches must be set as shown whenever the IBM Enhanced Graphics Adapter is installed, regardless of display type. This is true even when a second display adapter is installed in the system.

Configuration Switches

The following diagram shows the location and orientation of the configuration switches on the Enhanced Graphics Adapter.



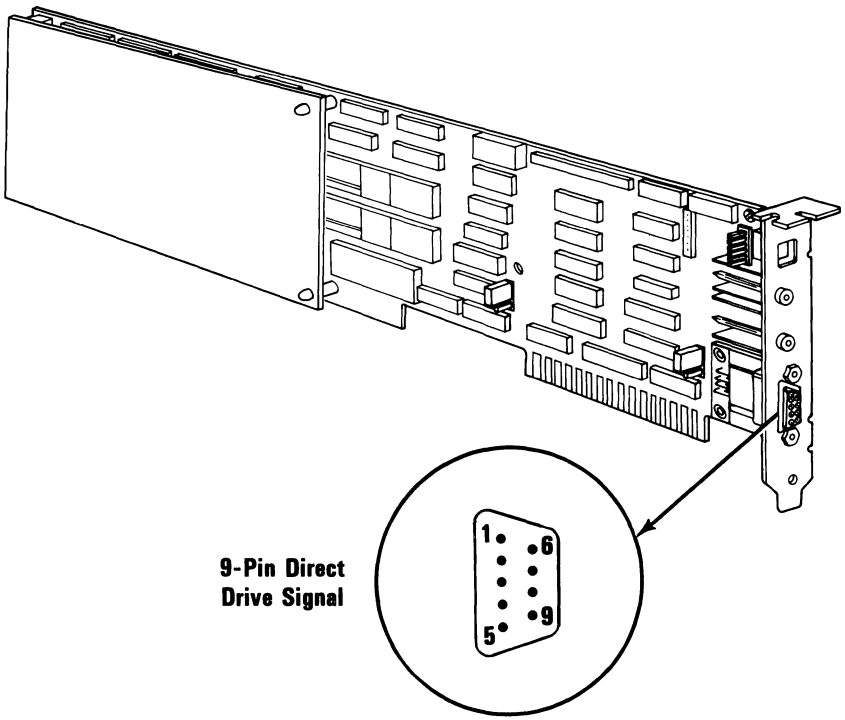
Configuration Switch Settings

The configuration switches on the Enhanced Graphics Adapter determine the type of display support the adapter provides, as follows:

Switch Settings for Enhanced Graphics Adapter as Primary Display Adapter						
				Configuration		
SW1	SW2	SW3	SW4	Enhanced Adapter	Monochrome Adapter	Color/Graphics Adapter
On	Off	Off	On	Color Display 40x25	Secondary	–
Off	Off	Off	On	Color Display 80x25	Secondary	–
On	On	On	Off	Enhanced Display Emulation Mode	Secondary	–
Off	On	On	Off	Enhanced Display Hi Res Mode	Secondary	–
On	Off	On	Off	Monochrome	–	Secondary 40x25
Off	Off	On	Off	Monochrome	–	Secondary 80x25

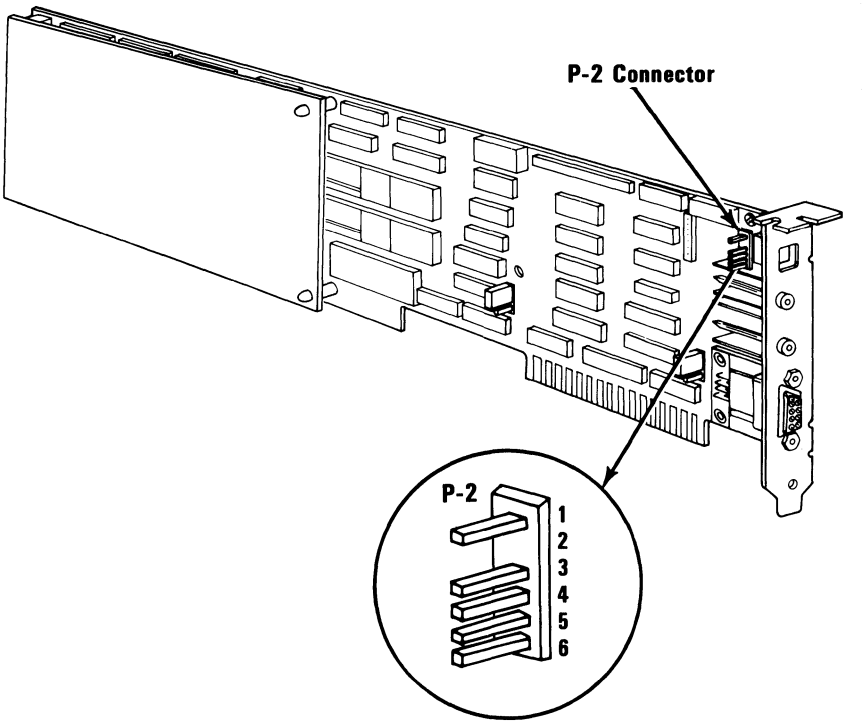
Switch Settings for Enhanced Graphics Adapter as Secondary Display Adapter						
				Configuration		
SW1	SW2	SW3	SW4	Enhanced Adapter	Monochrome Adapter	Color/Graphics Adapter
On	On	On	On	Color Display 40x25	Primary	–
Off	On	On	On	Color Display 80x25	Primary	–
On	Off	On	On	Enhanced Display Emulation Mode	Primary	–
Off	Off	On	On	Enhanced Display Hi Res Mode	Primary	–
On	On	Off	On	Monochrome	–	Primary 40x25
Off	On	Off	On	Monochrome	–	Primary 80x25

Direct Drive Connector



	Signal Name - Description	Pin	
Direct Drive Display	Ground	1	**Enhanced Graphics Adapter**
	Secondary Red	2	
	Primary Red	3	
	Primary Green	4	
	Primary Blue	5	
	Secondary Green/Intensity	6	
	Secondary Blue/Mono Video	7	
	Horizontal Retrace	8	
	Vertical Retrace	9	

Light Pen Interface



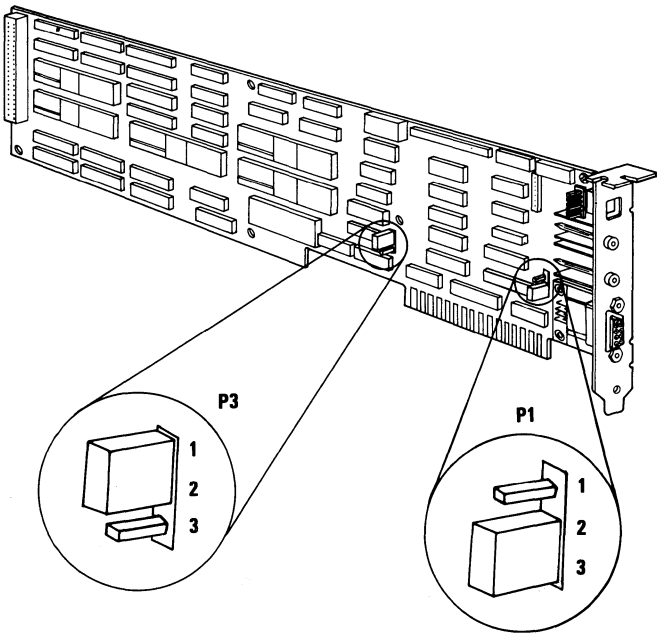
	P-2 Connector	Pin	
Light Pen Attachment	+Light Pen Input	1	Enhanced Graphics Adapter
	Not used	2	
	+Light Pen Switch	3	
	Ground	4	
	+5 Volts	5	
	12 Volts	6	

Jumper Descriptions

Located on the adapter are two jumpers designated P1 and P3. Jumper P1 changes the function of pin 2 on the direct drive interface. When placed on pins 2 and 3, jumper P1 selects ground as the function of direct drive interface, pin 2. This selection is for displays that support five color outputs, such as the IBM Color Display. When P1 is placed on pins 1 and 2, red prime output is placed on pin 2 of the direct drive interface connector. This supports the IBM Enhanced Color Display, which utilizes six color outputs on the direct drive interface.

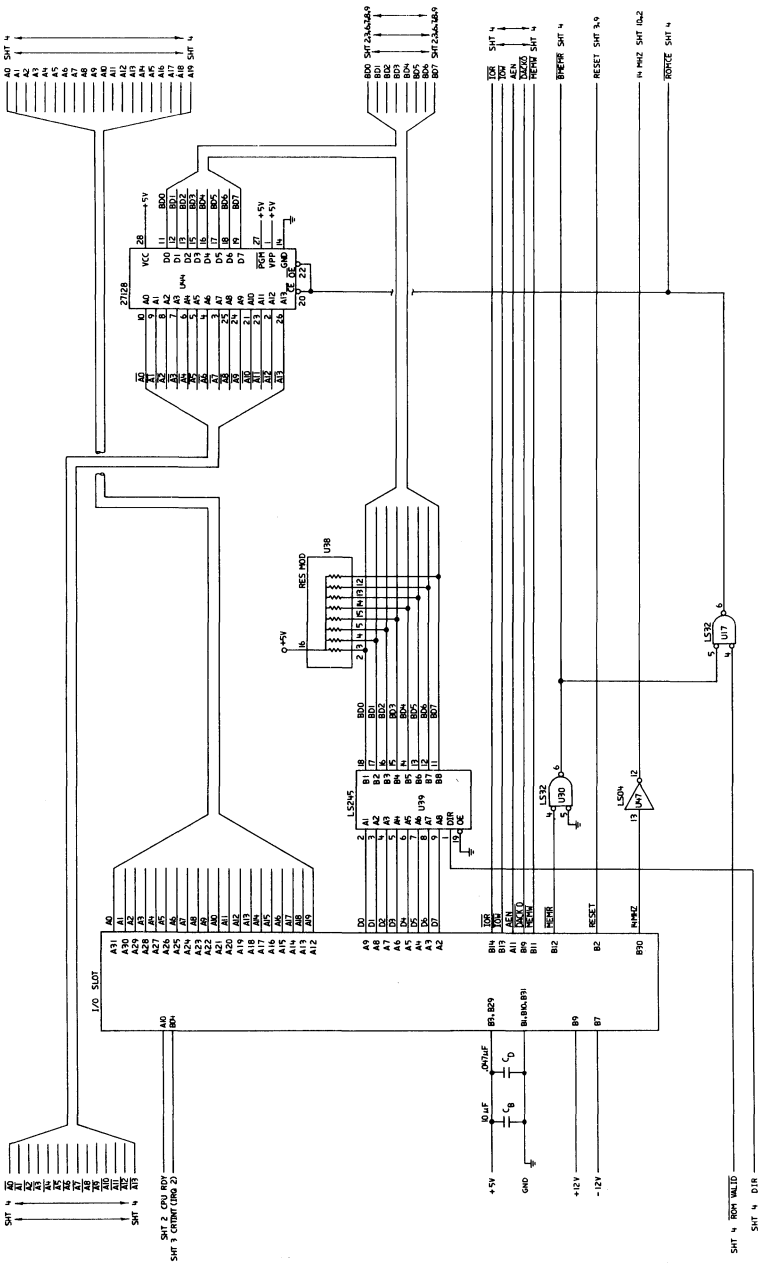
Jumper P3 changes the I/O address port of the Enhanced Graphics Adapter within the system. In its normal position, (pins 1 and 2), all Enhanced Graphics Adapter addresses are in the range 3XX. Moving jumper P3 to pins 2 and 3 changes the addresses to 2XX. Operation of the adapter in the 2XX mode is not supported in BIOS.

The following figure shows the location of the jumpers and numbering of the connectors.

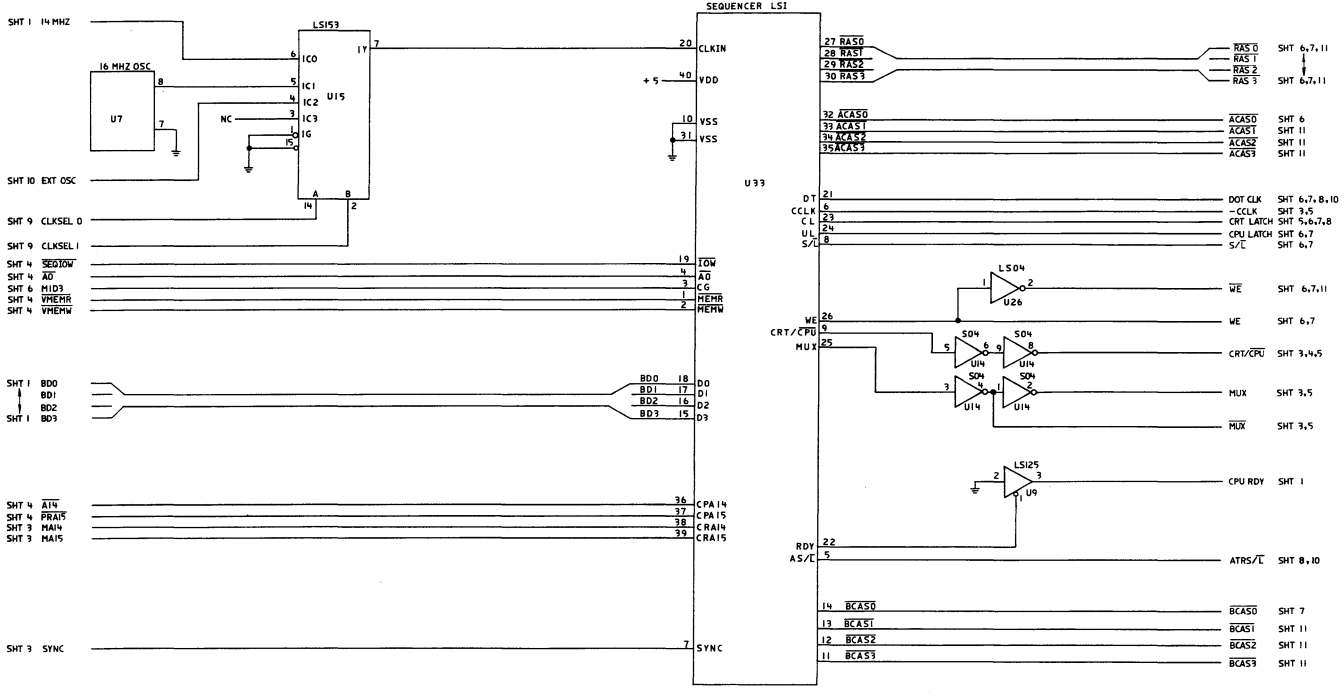


Logic Diagrams

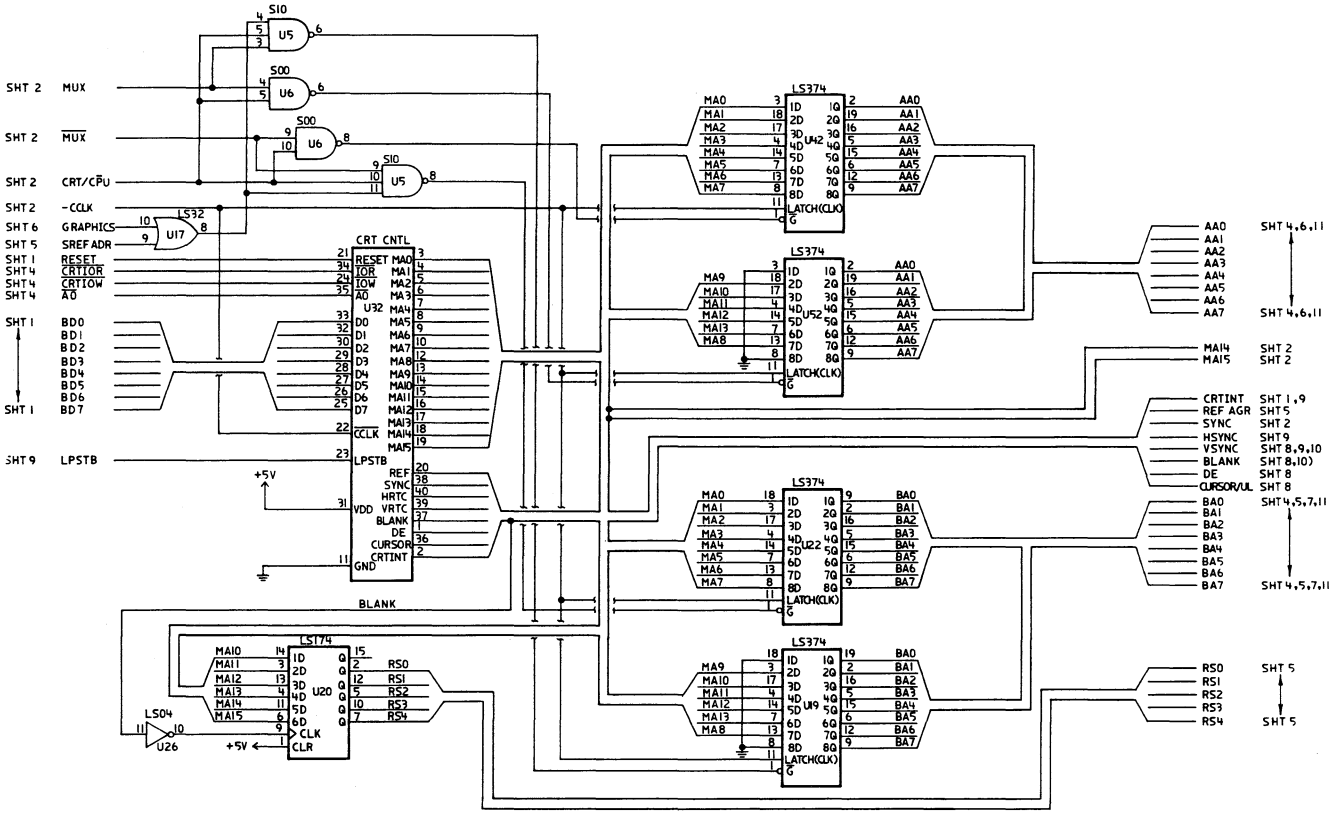
ENHANCED GRAPHICS ADAPTER



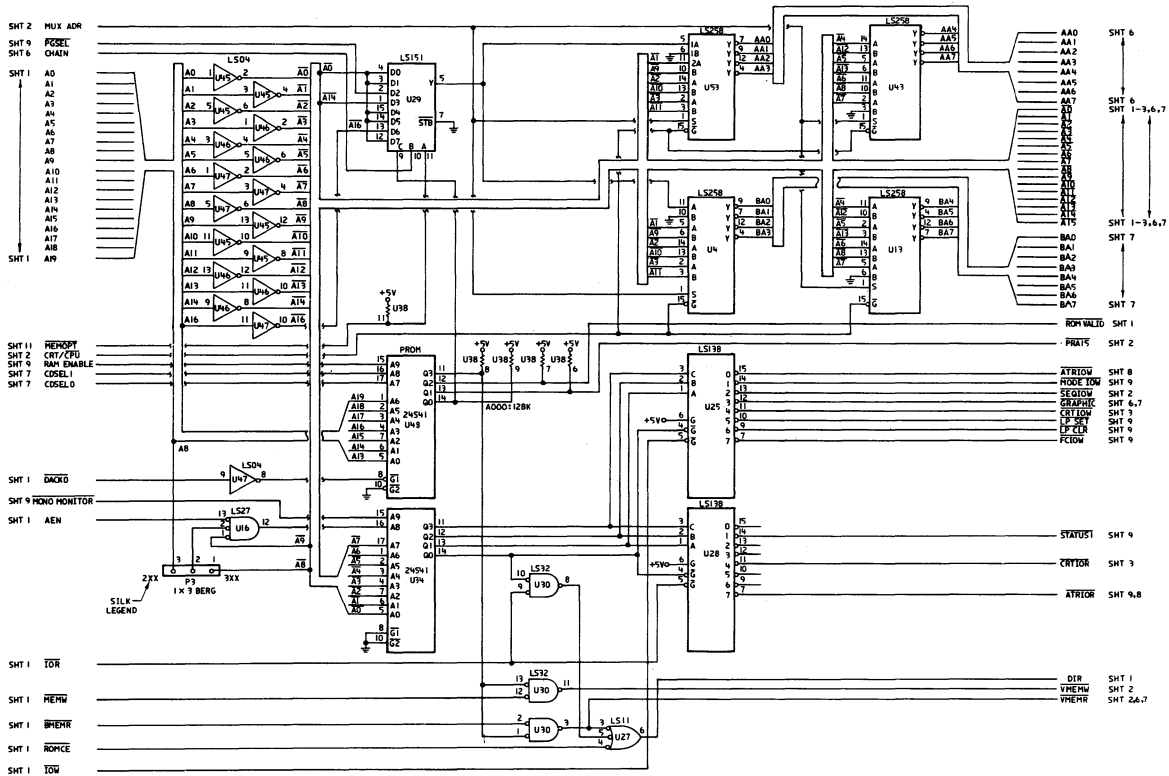
Enhanced Graphics Adapter Sheet 1 of 11



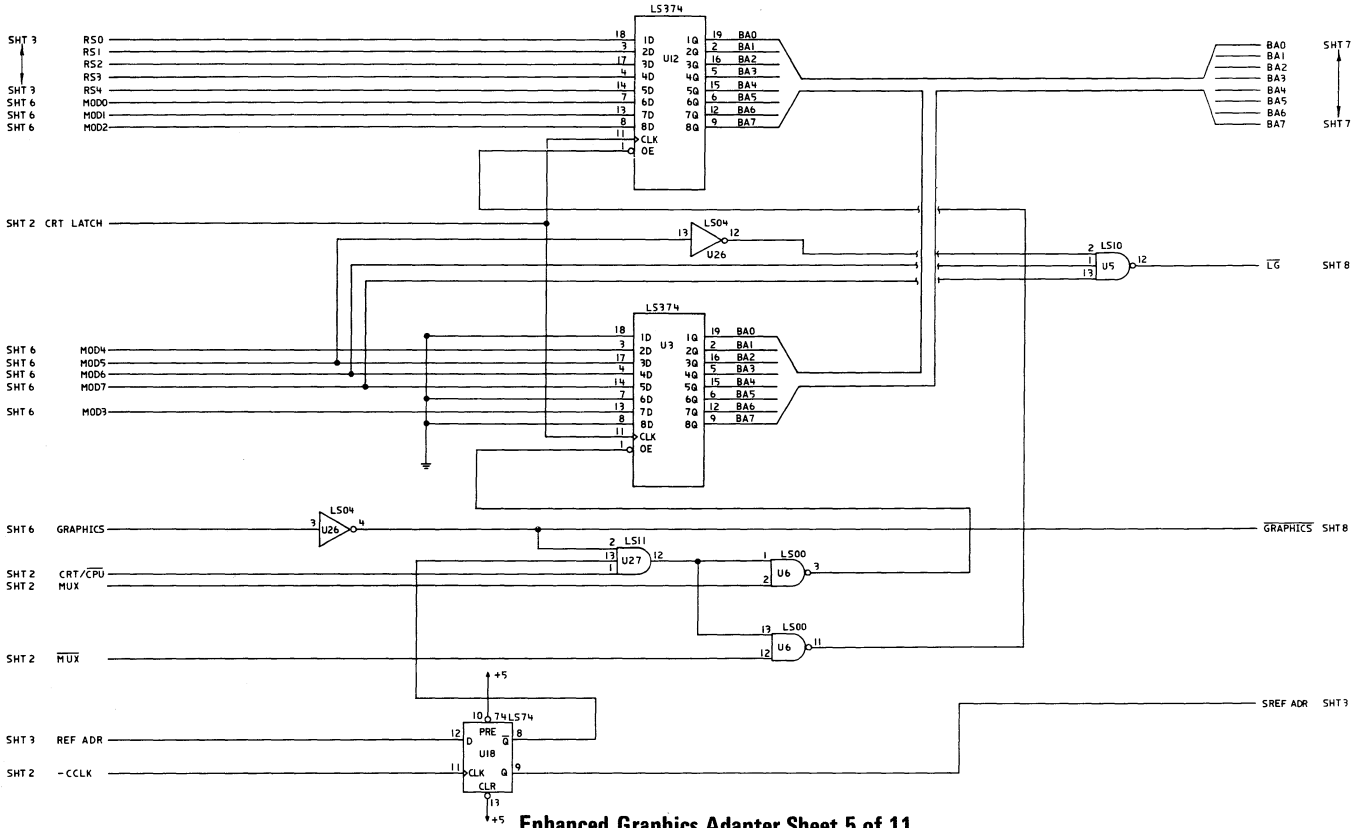
Enhanced Graphics Adapter Sheet 2 of 11



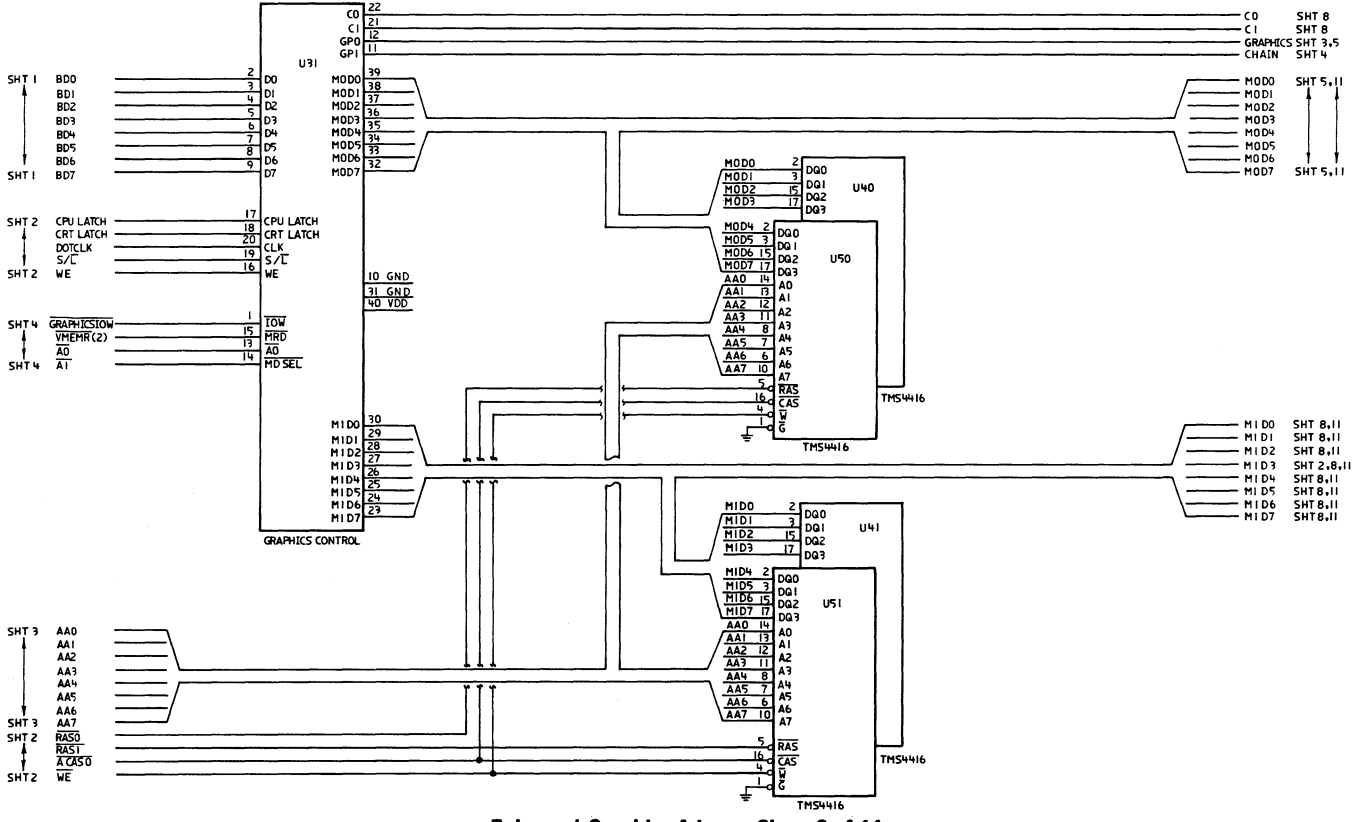
Enhanced Graphics Adapter Sheet 3 of 11



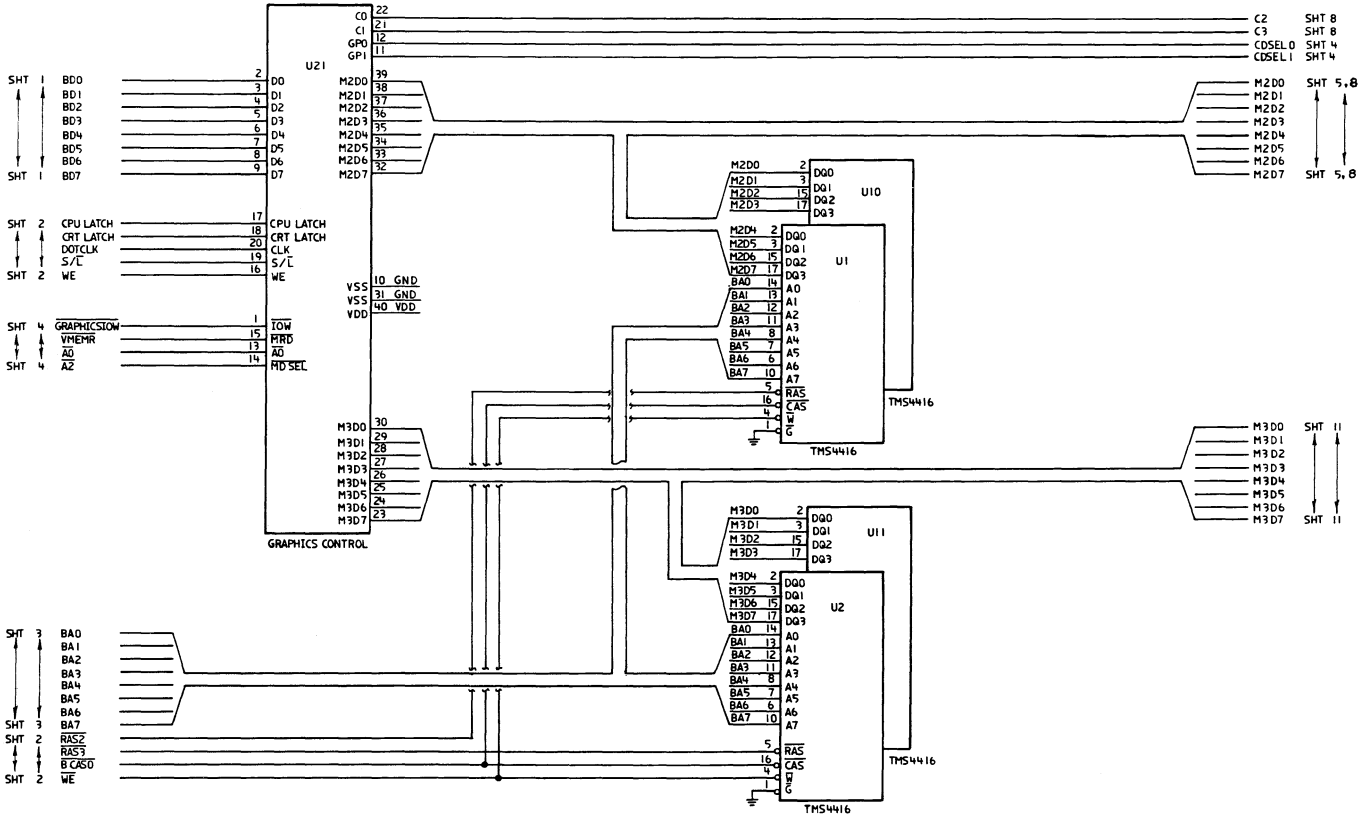
Enhanced Graphics Adapter Sheet 4 of 11



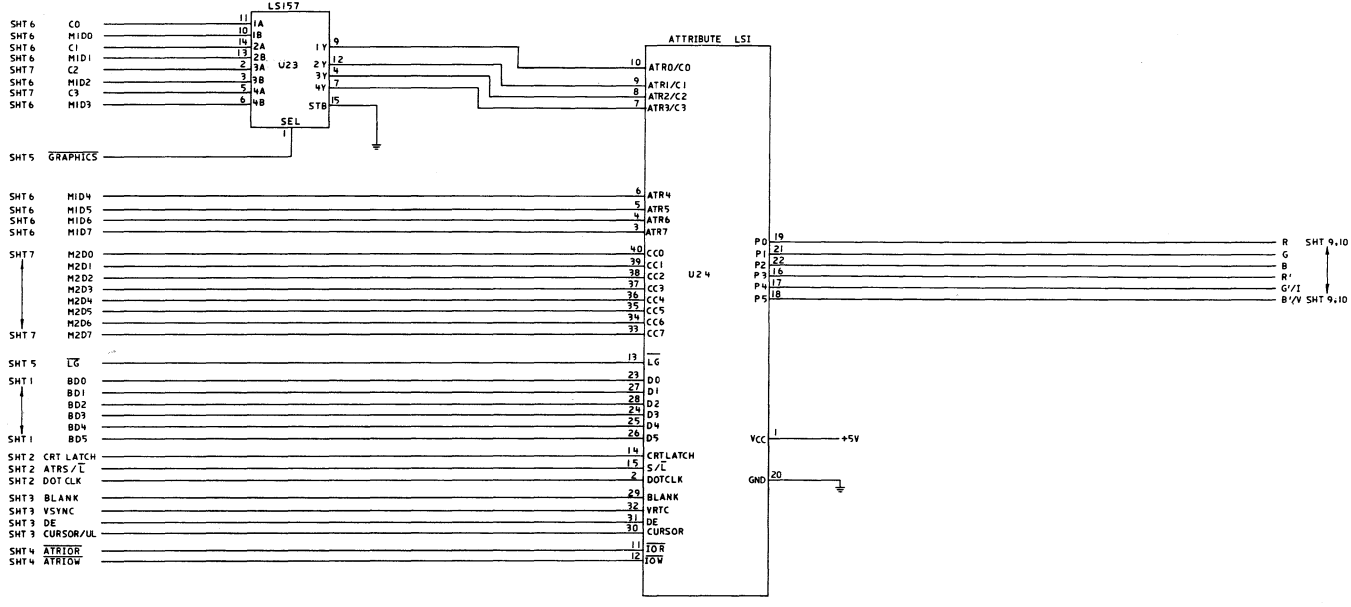
Enhanced Graphics Adapter Sheet 5 of 11



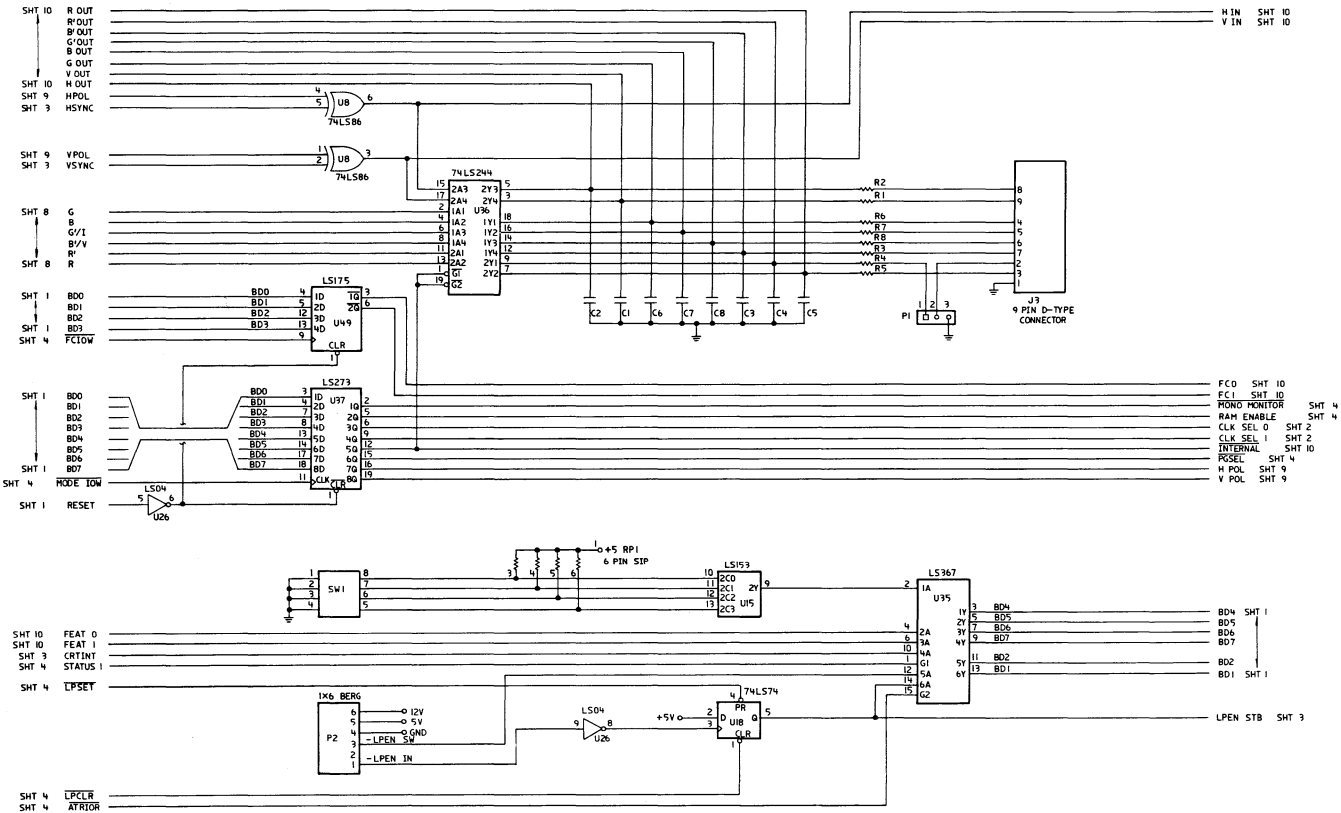
Enhanced Graphics Adapter Sheet 6 of 11



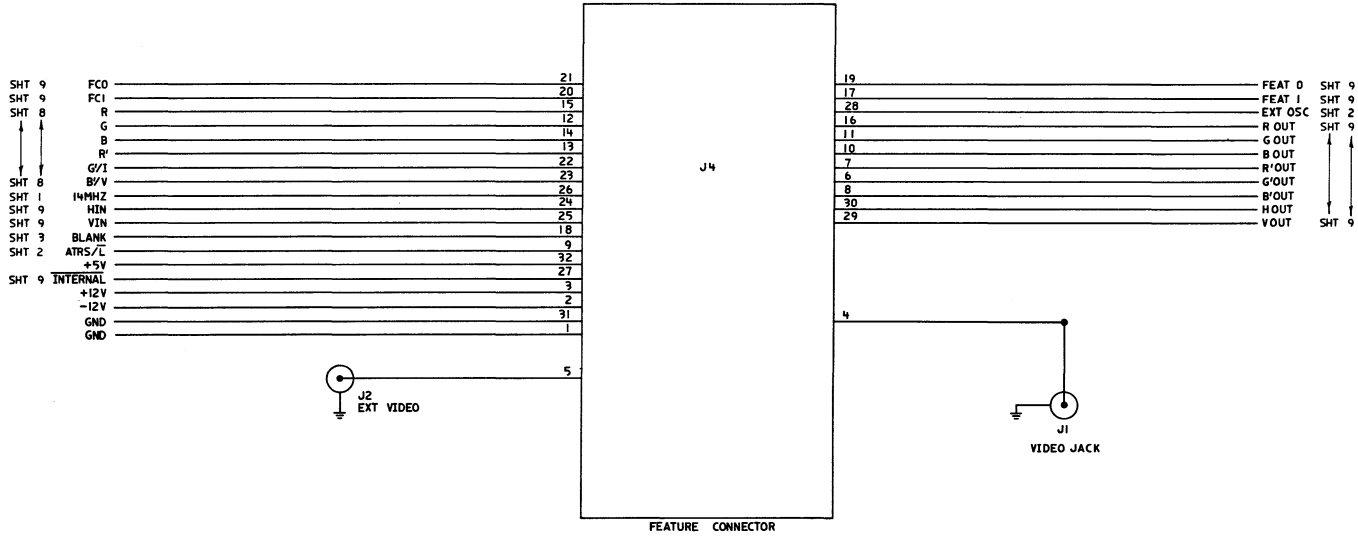
Enhanced Graphics Adapter Sheet 7 of 11



Enhanced Graphics Adapter Sheet 8 of 11



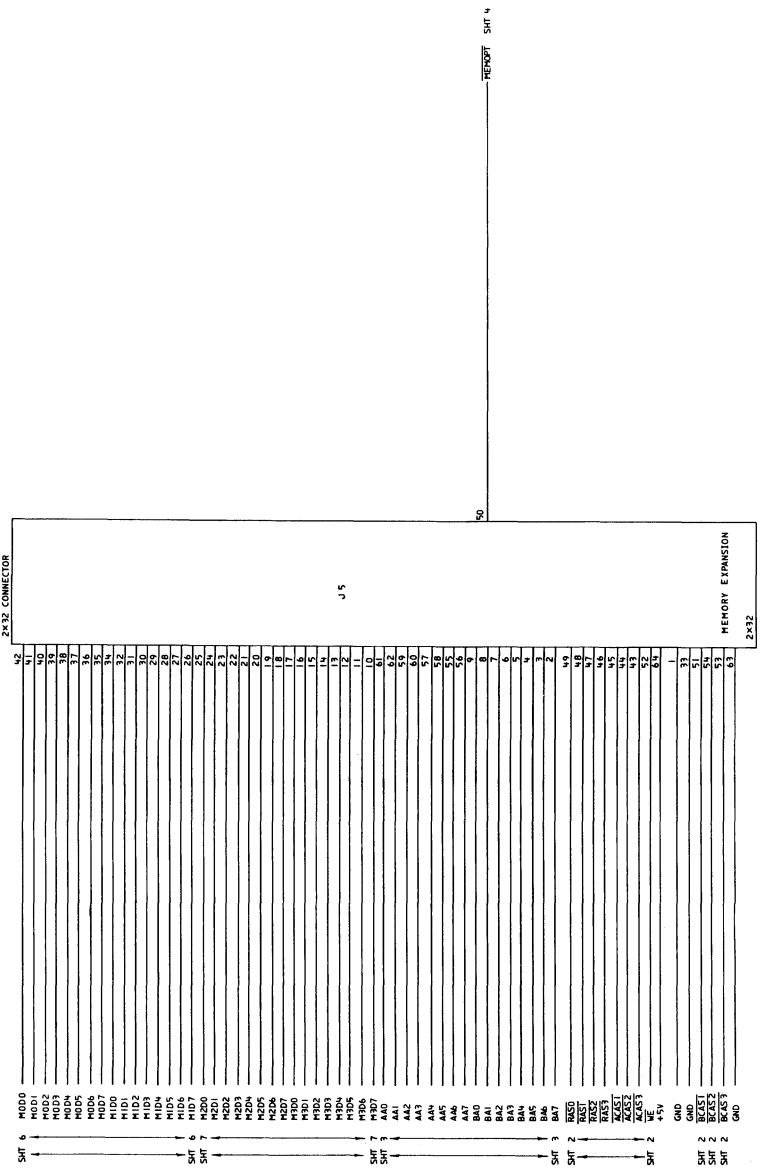
Enhanced Graphics Adapter Sheet 9 of 11



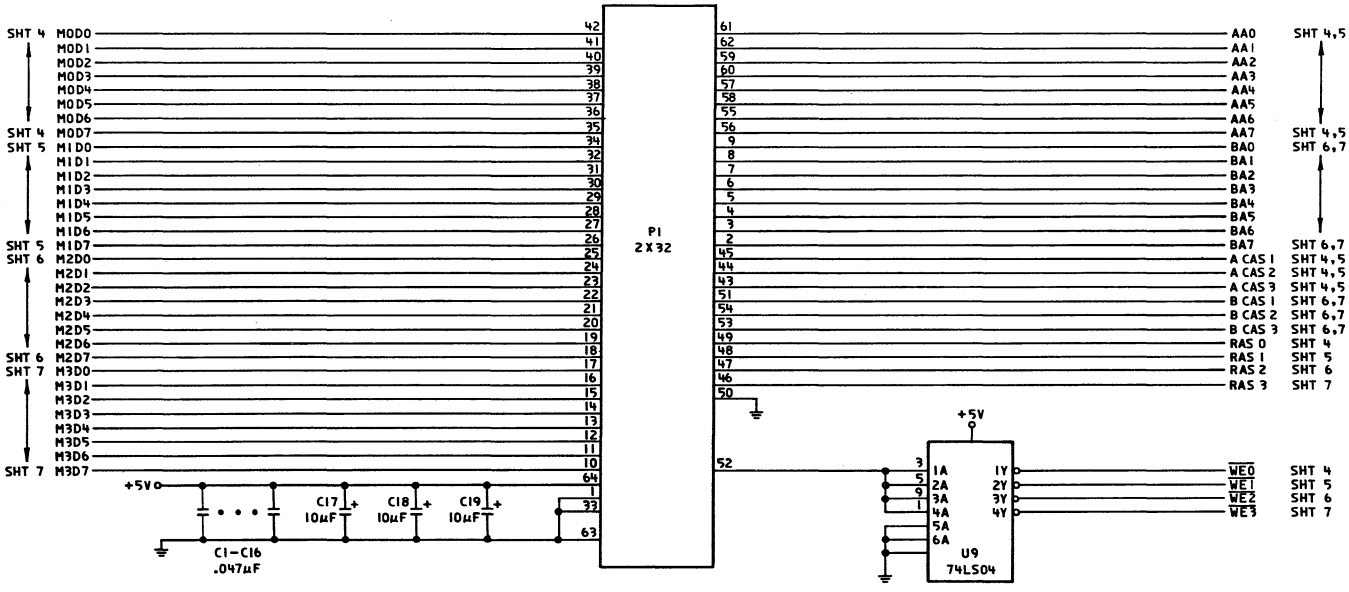
NOTE :

1 GROUNDS – ONE AT EACH END OF CONNECTOR.

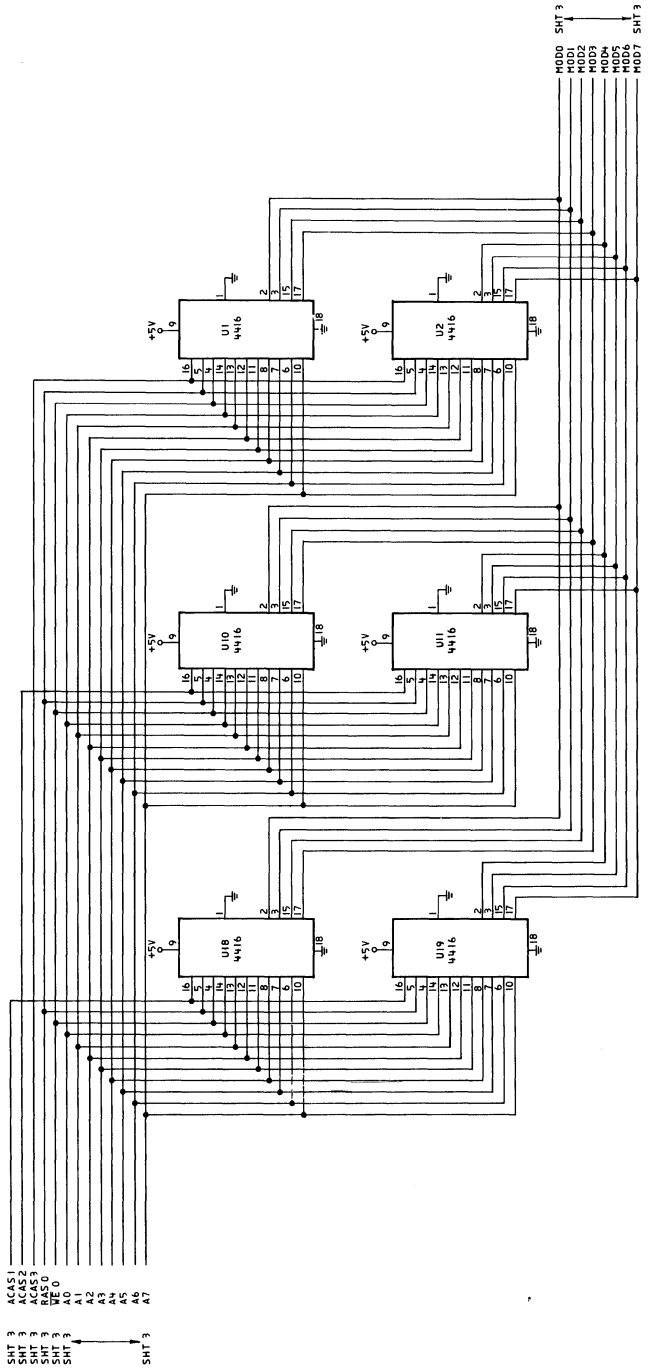
Enhanced Graphics Adapter Sheet 10 of 11



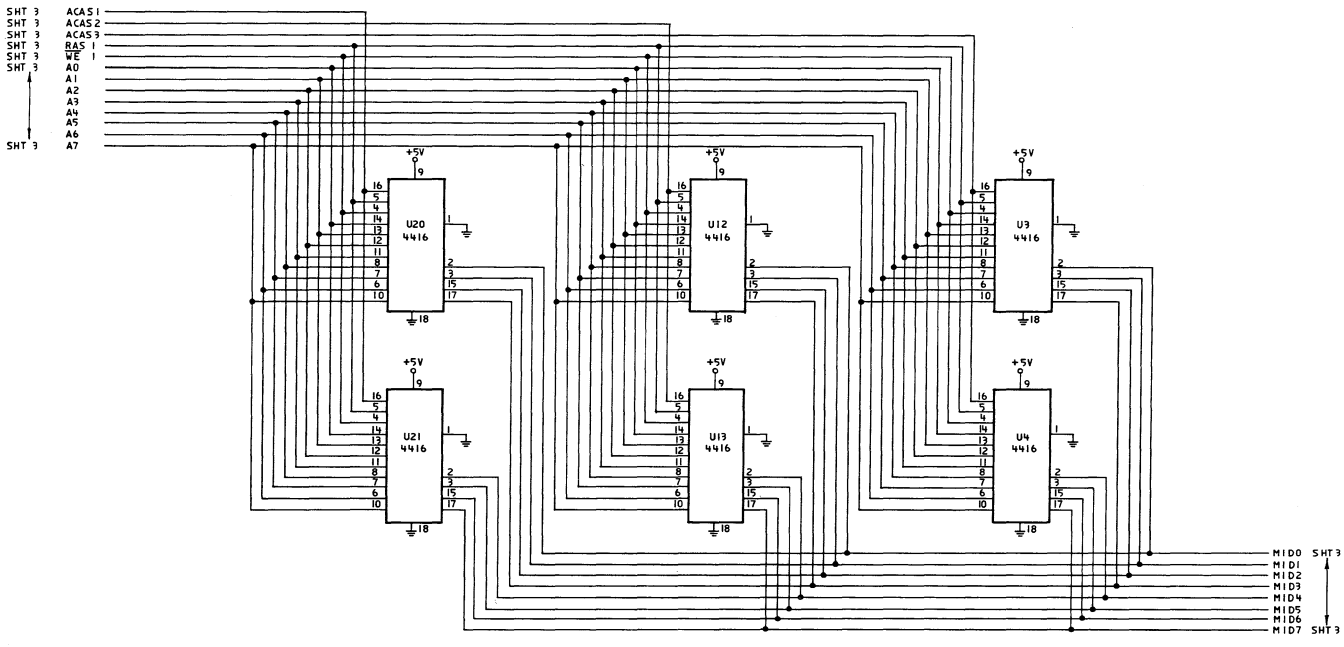
Enhanced Graphics Adapter Sheet 11 of 11



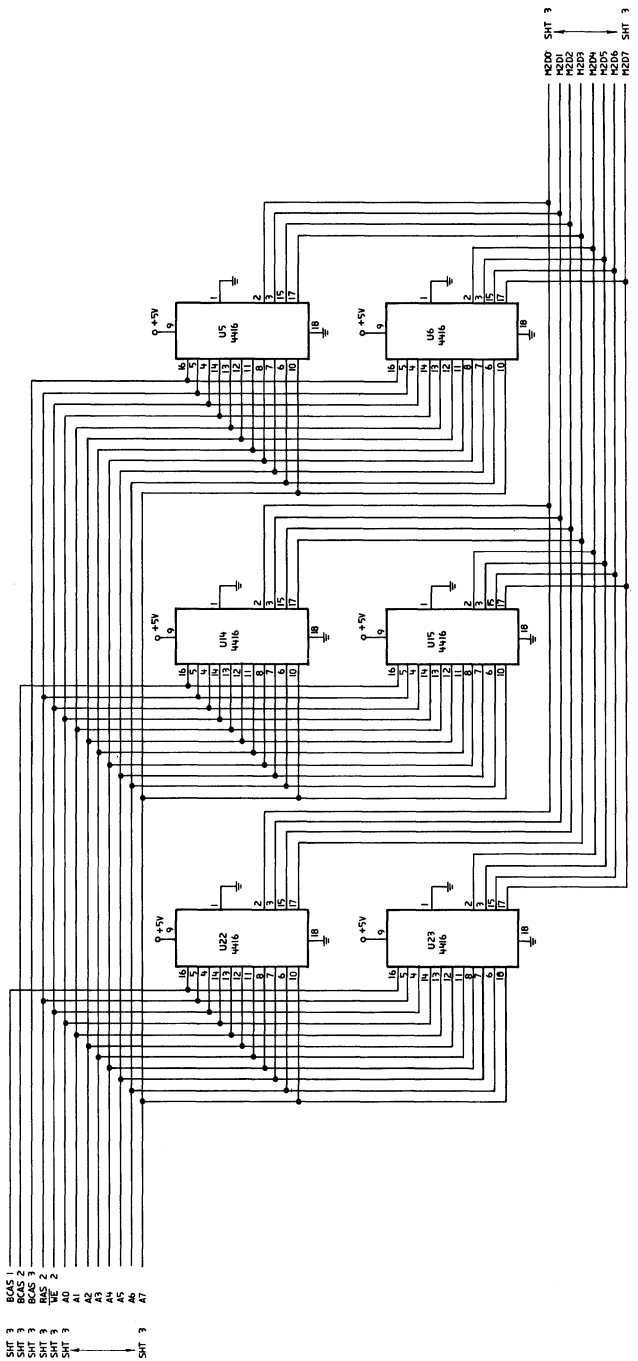
Graphics Memory Expansion Card Sheet 1 of 5



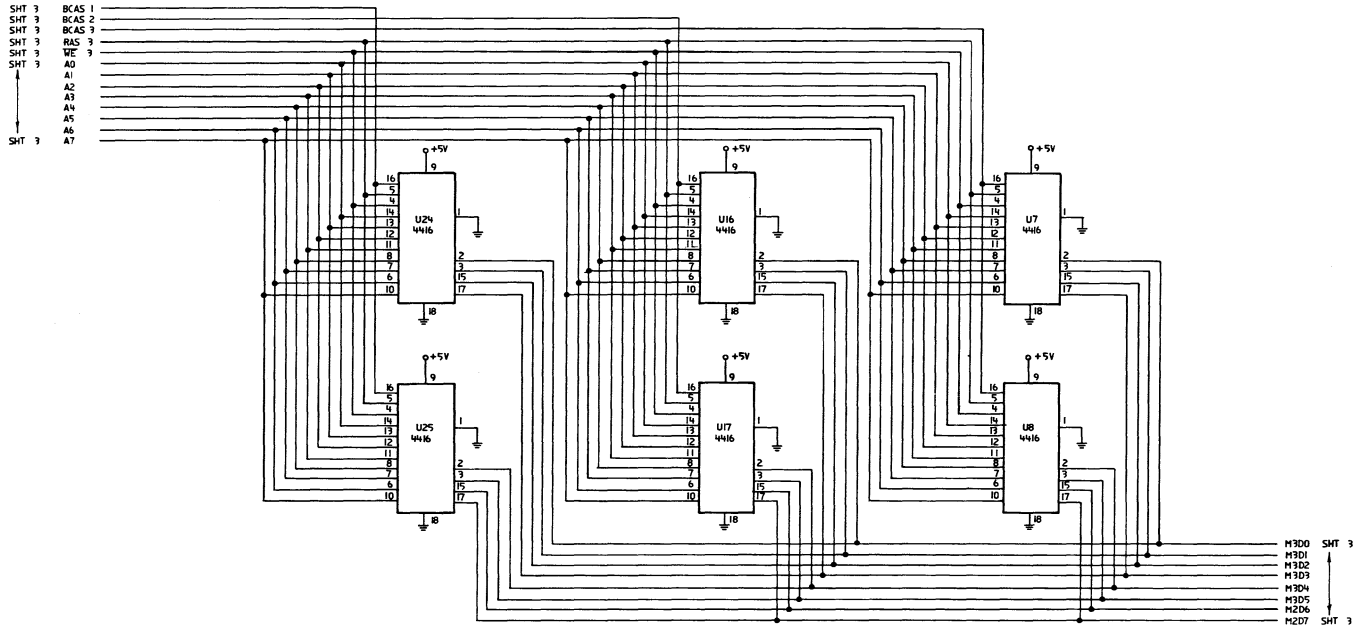
Graphics Memory Expansion Card Sheet 2 of 5



Graphics Memory Expansion Card Sheet 3 of 5



Graphics Memory Expansion Card Sheet 4 of 5



Graphics Memory Expansion Card Sheet 5 of 5

BIOS Listing

Vectors with Special Meanings

Interrupt Hex 42 - Reserved

When an IBM Enhanced Graphics Adapter is installed, the BIOS routines use interrupt 42 to revector the video pointer.

Interrupt Hex 43 - IBM Enhanced Graphics Video Parameters

When an IBM Enhanced Graphics Adapter is installed, the BIOS routines use this vector to point to a data region containing the parameters required for the initializing of the IBM Enhanced Graphics Adapter. Note that the format of the table must adhere to the BIOS conventions established in the listing. The power-on routines initialize this vector to point to the parameters contained in the IBM Enhanced Graphics Adapter ROM.

Interrupt Hex 44 - Graphics Character Table

When an IBM Enhanced Graphics Adapter is installed the BIOS routines use this vector to point to a table of dot patterns that will be used when graphics characters are to be displayed. This table will be used for the first 128 code points in video modes 4, 5, and 6. This table will be used for 256 characters in all additional graphics modes. See the appropriate BIOS interface for additional information on setting and using the graphics character table pointer.

```
1                       PAGE,120
2                       TITLE   ENHANCED GRAPHICS ADAPTER BIOS
3                       EXTRN   CGMN:NEAR, CGDDOT:NEAR, INT_1F_1:NEAR, CGMN_FDG:NEAR
4                       EXTRN   END_ADDRESS:NEAR
5
6               ;-------------------------------------------------------------
7               ;       THE  BIOS ROUTINES ARE MEANT TO BE ACCESSED THROUGH     :
8               ;       SOFTWARE INTERRUPTS ONLY.  ANY ADDRESSES PRESENT IN     :
9               ;       THE LISTINGS  ARE INCLUDED  ONLY FOR  COMPLETENESS,     :
10              ;       NOT FOR  REFERENCE.   APPLICATIONS WHICH  REFERENCE     :
11              ;       ABSOLUTE   ADDRESSES   WITHIN   THE   CODE  SEGMENT     :
12              ;       VIOLATE THE STRUCTURE AND DESIGN OF BIOS.               :
13              ;-------------------------------------------------------------
14
15              .LIST
16      C               INCLUDE         VFRONT.INC
17      C               SUBTTL  VFRONT.INC
18      C               PAGE
19      C
20      C       ;--- INT 10 ------------------------------------------------------------
21      C       ; VIDEO_IO                                                            :
22      C       ;       THESE ROUTINES PROVIDE THE CRT INTERFACE                      :
23      C       ;       THE FOLLOWING FUNCTIONS ARE PROVIDED:                         :
24      C       ;       (AH)=0  SET MODE (AL) CONTAINS MODE VALUE                     :
25      C       ;                                                                     :
26      C       ;       AL AD   TYPE    RES       NOTES  DF-DIM  DISPLAY      MAX PGS :
27      C       ;       ------------------------------------------------------------- :
28      C       ;     *  0 B8   ALPHA   640X200           40X25   COLOR - BW       8  :
29      C       ;        1 B8   ALPHA   640X200           40X25   COLOR            8  :
30      C       ;     *  2 B8   ALPHA   640X200           80X25   COLOR - BW       8  :
31      C       ;        3 B8   ALPHA   640X200           80X25   COLOR            8  :
32      C       ;        4 B8   GRPHX   320X200           40X25   COLOR            1  :
33      C       ;        5 B8   GRPHX   320X200           40X25   COLOR - BW       1  :
34      C       ;        6 B8   GRPHX   640X200           80X25   COLOR - BW       1  :
35      C       ;     *  7 B0   ALPHA   720X350           80X25   MONOCHROME       8  :
36      C       ;                                                                     :
37      C       ;        8      RESERVED                                              :
38      C       ;        9      RESERVED                                              :
39      C       ;        A      RESERVED                                              :
40      C       ;        B      RESERVED - INTERNAL USE                               :
41      C       ;        C      RESERVED - INTERNAL USE                               :
42      C       ;                                                                     :
43      C       ;        D A0   GRPHX   320X200           40X25   COLOR            8  :
44      C       ;        E A0   GRPHX   640X200           80X25   COLOR            4  :
45      C       ;        F A0   GRPHX   640X350           80X25   MONOCHROME       2  :
46      C       ;       10 A0   GRPHX   640X350           80X25   HI RES           2  :
47      C       ;                                                                     :
48      C       ;                                                                     :
49      C       ;               NOTE : HIGH BIT AL SET PREVENTS REGEN BUFFER CLEAR ON :
50      C       ;                      MODES RUNNING ON THE COMBO VIDEO ADAPTER       :
51      C       ;                                                                     :
52      C       ;               *** NOTE BW MODES OPERATE SAME AS COLOR MODES, BUT     :
53      C       ;                       COLOR BURST IS NOT ENABLED                    :
54      C       ;       (AH)=1  SET CURSOR TYPE                                       :
55      C       ;               (CH) =  BITS 4-0 = START LINE FOR CURSOR              :
56      C       ;                       ** HARDWARE WILL ALWAYS CAUSE BLINK           :
57      C       ;                       ** SETTING BIT 5 OR 6 WILL CAUSE ERRATIC      :
58      C       ;                          BLINKING OR NO CURSOR AT ALL               :
59      C       ;               (CL) =  BITS 4-0 = END LINE FOR CURSOR                :
60      C       ;       (AH)=2  SET CURSOR POSITION                                   :
61      C       ;               (DH,DL) = ROW,COLUMN  (0,0) IS UPPER LEFT             :
62      C       ;               (BH) = PAGE NUMBER                                    :
63      C       ;       (AH)=3  READ CURSOR POSITION                                  :
64      C       ;               (BH) = PAGE NUMBER                                    :
65      C       ;               ON EXIT (DH,DL) = ROW,COLUMN OF CURRENT CURSOR        :
66      C       ;                       (CH,CL) = CURSOR MODE CURRENTLY SET           :
67      C       ;       (AH)=4  READ LIGHT PEN POSITION                               :
68      C       ;               ON EXIT:                                              :
69      C       ;               (AH) = 0 -- LIGHT PEN SWITCH NOT DOWN/NOT TRIGGERED   :
70      C       ;               (AH) = 1 -- VALID LIGHT PEN VALUE IN REGISTERS        :
71      C       ;                       (DH,DL) = ROW,COLUMN OF CHARACTER LP POSN     :
72      C       ;                       (CH) = RASTER LINE (0-199)                    :
73      C       ;                       (CX) = RASTER LINE (0-NNN) NEW GRAPHICS MODES :
74      C       ;                       (BX) = PIXEL COLUMN (0-319,639)               :
75      C       ;       (AH)=5  SELECT ACTIVE DISPLAY PAGE                            :
76      C       ;               (AL) = NEW PAGE VALUE, SEE AH=0 FOR PAGE INFO         :
77      C       ;       (AH)=6  SCROLL ACTIVE PAGE UP                                 :
78      C       ;               (AL) = NUMBER OF LINES, INPUT LINES BLANKED AT BOTTOM :
79      C       ;                      OF WINDOW                                      :
80      C       ;                       AL = 0 MEANS BLANK ENTIRE WINDOW              :
81      C       ;               (CH,CL) = ROW,COLUMN OF UPPER LEFT CORNER OF SCROLL   :
82      C       ;               (DH,DL) = ROW,COLUMN OF LOWER RIGHT CORNER OF SCROLL  :
83      C       ;               (BH) = ATTRIBUTE TO BE USED ON BLANK LINE             :
84      C       ;       (AH)=7  SCROLL ACTIVE PAGE DOWN                               :
85      C       ;               (AL) = NUMBER OF LINES, INPUT LINES BLANKED AT TOP    :
86      C       ;                      OF WINDOW                                      :
87      C       ;                       AL = 0 MEANS BLANK ENTIRE WINDOW              :
88      C       ;               (CH,CL) = ROW,COLUMN OF UPPER LEFT CORNER OF SCROLL   :
89      C       ;               (DH,DL) = ROW,COLUMN OF LOWER RIGHT CORNER OF SCROLL  :
90      C       ;               (BH) = ATTRIBUTE TO BE USED ON BLANK LINE             :
91      C       ;                                                                     :
92      C       ;       CHARACTER HANDLING ROUTINES                                   :
93      C       ;                                                                     :
94      C       ;       (AH) = 8 READ ATTRIBUTE/CHARACTER AT CURRENT CURSOR POSITION  :
95      C       ;               (BH) = DISPLAY PAGE                                   :
96      C       ;               ON EXIT:                                              :
97      C       ;               (AL) = CHAR READ                                      :
98      C       ;               (AH) = ATTRIBUTE OF CHARACTER READ (ALPHA MODES ONLY) :
99      C       ;       (AH) = 9 WRITE ATTRIBUTE/CHARACTER AT CURRENT CURSOR POSITION :
100     C       ;               (BH) = DISPLAY PAGE                                   :
101     C       ;               (CX) = COUNT OF CHARACTERS TO WRITE                   :
102     C       ;               (AL) = CHAR TO WRITE                                  :
103     C       ;               (BL) = ATTRIBUTE OF CHARACTER (ALPHA)/COLOR OF CHAR   :
104     C       ;                      (GRAPHICS)                                     :
105     C       ;                       SEE NOTE ON WRITE DOT FOR BIT 7 OF BL = 1.    :
106     C       ;       (AH) = A  WRITE CHARACTER ONLY AT CURRENT CURSOR POSITION     :
107     C       ;               (BH) = DISPLAY PAGE                                   :
108     C       ;               (CX) = COUNT OF CHARACTERS TO WRITE                   :
109     C       ;               (AL) = CHAR TO WRITE                                  :
110     C       ;       FOR READ/WRITE CHARACTER INTERFACE WHILE IN GRAPHICS MODE, THE :
111     C       ;               CHARACTERS ARE FORMED FROM A CHARACTER GENERATOR IMAGE :
112     C       ;               MAINTAINED IN THE SYSTEM ROM.  ONLY THE 1ST 128 CHARS :
113     C       ;               ARE CONTAINED THERE.  TO READ/WRITE THE SECOND 128    :
114     C       ;               CHARS, THE USER MUST INITIALIZE THE POINTER AT        :
115     C       ;               INTERRUPT 1FH (LOCATION 0007CH) TO POINT TO THE 1K BYTE:
116     C       ;               TABLE CONTAINING THE CODE POINTS FOR THE SECOND       :
117     C       ;               128 CHARS (128-255).                                  :
118     C       ;                                                                     :
119     C       ;               FOR THE NEW GRAPHICS MODES 256 GRAPHICS CHARS ARE     :
120     C       ;               SUPPLIED IN THE SYSTEM ROM.                           :
121     C       ;                                                                     :
122     C       ;       FOR WRITE CHARACTER INTERFACE IN GRAPHICS MODE, THE REPLICATION :
123     C       ;               FACTOR CONTAINED IN (CX) ON ENTRY WILL PRODUCE VALID  :
124     C       ;               RESULTS ONLY FOR CHARACTERS CONTAINED ON THE SAME ROW. :
125     C       ;               CONTINUATION TO SUCCEEDING LINES WILL NOT PRODUCE      :
126     C       ;               CORRECTLY.                                            :
```

```
C ;
C ;       GRAPHICS INTERFACE                                                    :
C ;       (AH) = B  SET COLOR PALETTE                                           :
C ;                 FOR USE IN COMPATIBILITY MODES                              :
C ;                 (BH) = PALETTE COLOR ID BEING SET (0-127)                   :
C ;                 (BL) = COLOR VALUE TO BE USED WITH THAT COLOR ID            :
C ;                    NOTE: FOR THE CURRENT COLOR CARD, THIS ENTRY POINT       :
C ;                          HAS MEANING ONLY FOR 320X200 GRAPHICS.             :
C ;                          COLOR ID = 0 SELECTS THE BACKGROUND COLOR (0-15):  :
C ;                          COLOR ID = 1 SELECTS THE PALETTE TO BE USED:       :
C ;                                       0 = GREEN(1)/RED(2)/BROWN(3)          :
C ;                                       1 = CYAN(1)/MAGENTA(2)/WHITE(3)       :
C ;                          IN 40X25 OR 80X25 ALPHA MODES, THE VALUE SET       :
C ;                                  FOR PALETTE COLOR 0 INDICATES THE          :
C ;                                  BORDER COLOR TO BE USED (VALUES 0-31,      :
C ;                                  WHERE 16-31 SELECT THE HIGH INTENSITY      :
C ;                                  BACKGROUND SET).                           :
C ;       (AH) = C  WRITE DOT                                                   :
C ;                 (BH) = PAGE                                                 :
C ;                 (DX) = ROW NUMBER                                           :
C ;                 (CX) = COLUMN NUMBER                                        :
C ;                 (AL) = COLOR VALUE                                          :
C ;                          IF BIT 7 OF AL = 1, THEN THE COLOR VALUE IS        :
C ;                          EXCLUSIVE OR'D WITH THE CURRENT CONTENTS OF        :
C ;                          THE DOT                                            :
C ;       (AH) = D  READ DOT                                                    :
C ;                 (BH) = PAGE                                                 :
C ;                 (DX) = ROW NUMBER                                           :
C ;                 (CX) = COLUMN NUMBER                                        :
C ;                 (AL) RETURNS THE DOT READ                                   :
C ;                                                                             :
C ; ASCII TELETYPE ROUTINE FOR OUTPUT                                          :
C ;                                                                             :
C ;       (AH) = E  WRITE TELETYPE TO ACTIVE PAGE                               :
C ;                 (AL) = CHAR TO WRITE                                        :
C ;                 (BL) = FOREGROUND COLOR IN GRAPHICS MODE                    :
C ;                 NOTE -- SCREEN WIDTH IS CONTROLLED BY PREVIOUS MODE SET     :
C ;                                                                             :
C ;       (AH) = F  CURRENT VIDEO STATE                                         :
C ;                 RETURNS THE CURRENT VIDEO STATE                             :
C ;                 (AL) = MODE CURRENTLY SET  (SEE AH=0 FOR EXPLANATION)       :
C ;                 (AH) = NUMBER OF CHARACTER COLUMNS ON SCREEN                :
C ;                 (BH) = CURRENT ACTIVE DISPLAY PAGE                          :
C ;                                                                             :
C ;       (AH) = 10  SET PALETTE REGISTERS                                      :
C ;                                                                             :
C ;                 (AL) = 0          SET INDIVIDUAL PALETTE REGISTER           :
C ;                  BL = PALETTE REGISTER TO BE SET                            :
C ;                  BH = VALUE TO SET                                          :
C ;                                                                             :
C ;                 AL = 1          SET OVERSCAN REGISTER                       :
C ;                 BH = VALUE TO SET                                           :
C ;                                                                             :
C ;                 AL = 2          SET ALL PALETTE REGISTERS AND OVERSCAN      :
C ;                 ES:DX POINTS TO A 17 BYTE TABLE                             :
C ;                 BYTES 0 - 15 ARE THE PALETTE VALUES, RESPECTIVELY           :
C ;                 BYTE 16 IS THE OVERSCAN VALUE                               :
C ;                                                                             :
C ;                 AL = 3            TOGGLE INTENSIFY/BLINKING BIT             :
C ;                         BL - 0  ENABLE INTENSIFY                            :
C ;                         BL - 1  ENABLE BLINKING                             :
C ;                                                                             :
C ;       (AH) = 11  CHARACTER GENERATOR ROUTINE                                :
C ;                 NOTE : THIS CALL WILL INITIATE A MODE SET, COMPLETELY       :
C ;                        RESETTING THE VIDEO ENVIRONMENT BUT MAINTAINING      :
C ;                        THE REGEN BUFFER.                                    :
C ;                                                                             :
C ;                 AL = 00   USER ALPHA LOAD                                   :
C ;                        ES:BP - POINTER TO USER TABLE                        :
C ;                        CX    - COUNT TO STORE                               :
C ;                        DX    - CHARACTER OFFSET INTO TABLE                  :
C ;                        BL    - BLOCK TO LOAD                                :
C ;                        BH    - NUMBER OF BYTES PER CHARACTER                :
C ;                 AL = 01   ROM MONOCHROME SET                                :
C ;                        BL    - BLOCK TO LOAD                                :
C ;                 AL = 02   ROM 8X8 DOUBLE DOT                                :
C ;                        BL    - BLOCK TO LOAD                                :
C ;                 AL = 03   SET BLOCK SPECIFIER                               :
C ;                        BL    - CHAR GEN BLOCK SPECIFIER                     :
C ;                                D3-D2  ATTR BIT 3 ONE,  CHAR GEN 0-3         :
C ;                                D1-D0  ATTR BIT 3 ZERO, CHAR GEN 0-3         :
C ;                        NOTE : WHEN USING AL = 03 A FUNCTION CALL            :
C ;                                AX = 1000H                                   :
C ;                                BX = 0712H                                   :
C ;                                IS RECOMMENDED TO SET THE COLOR PLANES       :
C ;                                RESULTING IN 512 CHARACTERS AND EIGHT        :
C ;                                CONSISTENT COLORS.                           :
C ;                                                                             :
C ;       NOTE : THE FOLLOWING INTERFACE (AL=1X) IS SIMILAR IN FUNCTION         :
C ;              TO (AL=0X) EXCEPT THAT :                                       :
C ;                        - PAGE ZERO MUST BE ACTIVE                           :
C ;                        - POINTS (BYTES/CHAR) WILL BE RECALCULATED           :
C ;                        - ROWS WILL BE CALCULATED FROM THE FOLLOWING:        :
C ;                                INT[(200 OR 350) / POINTS] - 1               :
C ;                        - CRT_LEN WILL BE CALCULATED FROM :                  :
C ;                                (ROWS + 1) * CRT_COLS * 2                    :
C ;                        - THE CRTC WILL BE REPROGRAMMED AS FOLLOWS :         :
C ;                                R09H = POINTS - 1       MAX SCAN LINE        :
C ;                                       R09H DONE ONLY IN MODE 7              :
C ;                                R0AH = POINTS - 2       CURSOR START         :
C ;                                R0BH = 0                CURSOR END           :
C ;                                R12H =                  VERT DISP END        :
C ;                                       [(ROWS + 1) * POINTS] - 1             :
C ;                                R14H = POINTS           UNDERLINE LOC        :
C ;                                                                             :
C ;                 THE ABOVE REGISTER CALCULATIONS MUST BE CLOSE TO THE        :
C ;                 ORIGINAL TABLE VALUES OR UNDETERMINED RESULTS WILL          :
C ;                 OCCUR.                                                      :
C ;                                                                             :
C ;                 NOTE : THE FOLLOWING INTERFACE IS DESIGNED TO BE            :
C ;                        CALLED ONLY IMMEDIATELY AFTER A MODE SET HAS         :
C ;                        BEEN ISSUED. FAILURE TO ADHERE TO THIS PRACTICE      :
C ;                        MAY CAUSE UNDETERMINED RESULTS.                      :
C ;                                                                             :
C ;                 AL = 10   USER ALPHA LOAD                                   :
C ;                        ES:BP - POINTER TO USER TABLE                        :
C ;                        CX    - COUNT TO STORE                               :
C ;                        DX    - CHARACTER OFFSET INTO TABLE                  :
C ;                        BL    - BLOCK TO LOAD                                :
C ;                        BH    - NUMBER OF BYTES PER CHARACTER                :
C ;                 AL = 11   ROM MONOCHROME SET                                :
C ;                        BL    - BLOCK TO LOAD                                :
C ;                 AL = 12   ROM 8X8 DOUBLE DOT                                :
C ;                        BL    - BLOCK TO LOAD                                :
C ;                                                                             :
C ;                                                                             :
```

```
                                253      C  ;                NOTE : THE FOLLOWING INTERFACE IS DESIGNED TO BE       :
                                254      C  ;                       CALLED ONLY IMMEDIATELY AFTER A MODE SET HAS    :
                                255      C  ;                       BEEN ISSUED. FAILURE TO ADHERE TO THIS PRACTICE :
                                256      C  ;                       MAY CAUSE UNDETERMINED RESULTS.                 :
                                257      C  ;                                                                       :
                                258      C  ;                AL = 20   USER GRAPHICS CHARS  INT 01FH (8X8)      :
                                259      C  ;                       ES:BP - POINTER TO USER TABLE                   :
                                260      C  ;                AL = 21   USER GRAPHICS CHARS                      :
                                261      C  ;                       ES:BP - POINTER TO USER TABLE                   :
                                262      C  ;                       CX    - POINTS (BYTES PER CHARACTER)            :
                                263      C  ;                       BL    - ROW SPECIFIER                           :
                                264      C  ;                                                                       :
                                265      C  ;                             BL = 0  USER                              :
                                266      C  ;                                     DL - ROWS                         :
                                267      C  ;                             BL = 1  14 (0EH)                          :
                                268      C  ;                             BL = 2  25 (19H)                          :
                                269      C  ;                             BL = 3  43 (2BH)                          :
                                270      C  ;                                                                       :
                                271      C  ;                AL = 22   ROM 8 X 14 SET                           :
                                272      C  ;                       BL    - ROW SPECIFIER                           :
                                273      C  ;                AL = 23   ROM 8 X 8 DOUBLE DOT                     :
                                274      C  ;                       BL    - ROW SPECIFIER                           :
                                275      C  ;                                                                       :
                                276      C  ;                                                                       :
                                277      C  ;                AL = 30   INFORMATION                              :
                                278      C  ;                             CX    - POINTS                            :
                                279      C  ;                             DL    - ROWS                              :
                                280      C  ;                       BH    - 0     RETURN CURRENT INT 1FH PTR      :
                                281      C  ;                               ES:BP - PTR TO TABLE                    :
                                282      C  ;                       BH    - 1     RETURN CURRENT INT 44H PTR      :
                                283      C  ;                               ES:BP - PTR TO TABLE                    :
                                284      C  ;                       BH    - 2     RETURN ROM 8 X 14 PTR           :
                                285      C  ;                               ES:BP - PTR TO TABLE                    :
                                286      C  ;                       BH    - 3     RETURN ROM DOUBLE DOT PTR       :
                                287      C  ;                               ES:BP - PTR TO TABLE                    :
                                288      C  ;                       BH    - 4     RETURN ROM DOUBLE DOT PTR (TOP) :
                                289      C  ;                               ES:BP - PTR TO TABLE                    :
                                290      C  ;                       BH    - 5     RETURN ROM ALPHA ALTERNATE 9X14 :
                                291      C  ;                               ES:BP - PTR TO TABLE                    :
                                292      C  ;                                                                       :
                                293      C  ;          (AH) = 12   ALTERNATE SELECT                               :
                                294      C  ;                                                                       :
                                295      C  ;                BL = 10  RETURN EGA INFORMATION                    :
                                296      C  ;                       BH = 0 - COLOR MODE IN EFFECT <3><D><X>         :
                                297      C  ;                            1 - MONOC MODE IN EFFECT <3><B><X>         :
                                298      C  ;                       BL = MEMORY VALUE                           :
                                299      C  ;                            0 0 - 064K      0 1 - 128K             :
                                300      C  ;                            1 0 - 192K      1 1 - 256K             :
                                301      C  ;                       CH = FEATURE BITS                           :
                                302      C  ;                       CL = SWITCH SETTING                         :
                                303      C  ;                                                                       :
                                304      C  ;                BL = 20  SELECT ALTERNATE PRINT SCREEN ROUTINE     :
                                305      C  ;                                                                       :
                                306      C  ;          (AH) = 13   WRITE STRING                                   :
                                307      C  ;                       ES:BP - POINTER TO STRING TO BE WRITTEN         :
                                308      C  ;                       CX    - CHARACTER ONLY COUNT                    :
                                309      C  ;                       DX    - POSITION TO BEGIN STRING, IN CURSOR     :
                                310      C  ;                               TERMS                                   :
                                311      C  ;                       BH    - PAGE NUMBER                             :
                                312      C  ;                                                                       :
                                313      C  ;                AL = 0                                             :
                                314      C  ;                       BL    - ATTRIBUTE                               :
                                315      C  ;                       STRING - (CHAR, CHAR, CHAR, ...)                :
                                316      C  ;                       CURSOR NOT MOVED                            :
                                317      C  ;                AL = 1                                             :
                                318      C  ;                       BL    - ATTRIBUTE                               :
                                319      C  ;                       STRING - (CHAR, CHAR, CHAR, ...)                :
                                320      C  ;                       CURSOR  IS MOVED                            :
                                321      C  ;                AL = 2                                             :
                                322      C  ;                       STRING - (CHAR, ATTR, CHAR, ATTR, ...)          :
                                323      C  ;                       CURSOR NOT MOVED                            :
                                324      C  ;                AL = 3                                             :
                                325      C  ;                       STRING - (CHAR, ATTR, CHAR, ATTR, ...)          :
                                326      C  ;                       CURSOR  IS MOVED                            :
                                327      C  ;                                                                       :
                                328      C  ;                NOTE : CHAR RET, LINE FEED, BACKSPACE, AND BELL ARE    :
                                329      C  ;                       TREATED AS COMMANDS RATHER THAN PRINTABLE       :
                                330      C  ;                       CHARACTERS.                                     :
                                331      C  ;                                                                       :
                                332      C  ;-------------------------------------------------------------------------
                                333      C
                                334      C  SRLOAD  MACRO   SEGREG,VALUE
                                335      C          IFNB <VALUE>
                                336      C          IFIDN <VALUE>,<0>
                                337      C          SUB     DX,DX
                                338      C          ELSE
                                339      C          MOV     DX,VALUE
                                340      C          ENDIF
                                341      C          ENDIF
                                342      C          MOV     SEGREG,DX
                                343      C          ENDM
                                344      C
                                345      C
                                346      C  ;----- LOW MEMORY SEGMENT
                                347      C
0000                            348      C  ABS0    SEGMENT AT      0
0014                            349      C          ORG     005H*4                          ; PRINT SCREEN VECTOR
0014                            350      C  INT5_PTR        LABEL   DWORD
0040                            351      C          ORG     010H*4                          ; VIDEO I/O VECTOR
0040                            352      C  VIDEO           LABEL   DWORD
007C                            353      C          ORG     01FH*4                          ; GRAPHIC CHARS 128-255
007C                            354      C  EXT_PTR         LABEL   DWORD
                                355      C
0108                            356      C          ORG     042H*4                          ; REVECTORED 10H*4
0108                            357      C  PLANAR_VIDEO    LABEL   DWORD
                                358      C
010C                            359      C          ORG     043H*4                          ; GRAPHIC CHARS 0-255
010C                            360      C  GRX_SET         LABEL   DWORD
                                361      C
0410                            362      C          ORG     0410H
0410                            363      C  EQUIP_LOW       LABEL   BYTE
0410  ????                      364      C  EQUIP_FLAG      DW      ?
                                365      C
                                366      C  ;----- REUSE RAM FROM PLANAR BIOS
                                367      C
0449                            368      C          ORG     449H
0449  ??                        369      C  CRT_MODE        DB      ?
044A  ????                      370      C  CRT_COLS        DW      ?
044C  ????                      371      C  CRT_LEN         DW      ?
044E  ????                      372      C  CRT_START       DW      ?
0450     08 [                   373      C  CURSOR_POSN     DW      8 DUP(?)
             ????               374      C
                 ]              375      C
                                376      C
0460  ????                      377      C  CURSOR_MODE     DW      ?
0462  ??                        378      C  ACTIVE_PAGE     DB      ?
```

```
0463  ????                379     C  ADDR_6845       DW      ?
0465  ??                  380     C  CRT_MODE_SET    DB      ?
0466  ??                  381     C  CRT_PALETTE     DB      ?
                          382     C
0472                      383     C          ORG     0472H
0472  ????                384     C  RESET_FLAG      DW      ?
0484                      385     C          ORG     0484H
0484  ??                  386     C  ROWS     DB      ?                       ; ROWS ON THE SCREEN
0485  ????                387     C  POINTS   DW      ?                       ; BYTES PER CHARACTER
                          388     C
0487  ??                  389     C  INFO     DB      ?
                          390     C
                          391     C  ; INFO
                          392     C  ;       D7 - HIGH BIT OF MODE SET, CLEAR/NOT CLEAR REGEN
                          393     C  ;       D6 - MEMORY  D6 D5 = 0 0 - 064K    0 1 - 128K
                          394     C  ;       D5 - MEMORY          1 0 - 192K    1 1 - 256K
                          395     C  ;       D4 - RESERVED
                          396     C  ;       D3 - EGA ACTIVE MONITOR (0), EGA NOT ACTIVE (1)
                          397     C  ;       D2 - WAIT FOR DISPLAY ENABLE (1)
                          398     C  ;       D1 - EGA HAS A MONOCHROME ATTACHED (1)
                          399     C  ;       D0 - SET C_TYPE EMULATE ACTIVE (0)
                          400     C
0488  ??                  401     C  INFO_3   DB      ?
                          402     C
                          403     C  ; INFO_3
                          404     C  ;       D7-D4   FEATURE BITS
                          405     C  ;       D3-D0   SWITCHES
                          406     C
04A8                      407     C          ORG     04A8H
04A8                      408     C  SAVE_PTR        LABEL   DWORD
                          409     C
                          410     C  ;----- SAVE_PTR
                          411     C  ;
                          412     C  ;       SAVE_PTR IS A POINTER TO A TABLE AS DESCRIBED AS FOLLOWS :
                          413     C  ;
                          414     C  ;       DWORD_1         VIDEO PARAMETER TABLE POINTER
                          415     C  ;       DWORD_2         DYNAMIC SAVE AREA POINTER
                          416     C  ;       DWORD_3         ALPHA MODE AUXILIARY CHAR GEN POINTER
                          417     C  ;       DWORD_4         GRAPHICS MODE AUXILIARY CHAR GEN POINTER
                          418     C  ;       DWORD_5         RESERVED
                          419     C  ;       DWORD_6         RESERVED
                          420     C  ;       DWORD_7         RESERVED
                          421     C  ;
                          422     C  ;       DWORD_1         PARAMETER TABLE POINTER
                          423     C  ;                       INITIALIZED TO BIOS EGA PARAMETER TABLE.
                          424     C  ;                       THIS VALUE MUST EXIST.
                          425     C  ;
                          426     C  ;       DWORD_2         PARAMETER SAVE AREA POINTER
                          427     C  ;                       INITIALIZED TO 0000:0000, THIS VALUE IS OPTIONAL.
                          428     C  ;                       WHEN NON-ZERO, THIS POINTER WILL BE USED AS POINTER
                          429     C  ;                       TO A RAM AREA WHERE CERTAIN DYNAMIC VALUES ARE TO
                          430     C  ;                       BE SAVED. WHEN IN EGA OPERATION THIS RAM AREA WILL
                          431     C  ;                       HOLD THE 16 EGA PALETTE REGISTER VALUES PLUS
                          432     C  ;                       THE OVERSCAN VALUE IN BYTES 0-16D RESPECTIVELY.
                          433     C  ;                       AT LEAST 256 BYTES MUST BE ALLOCATED FOR THIS AREA.
                          434     C  ;
                          435     C  ;       DWORD_3         ALPHA MODE AUXILIARY POINTER
                          436     C  ;                       INITIALIZED TO 0000:0000, THIS VALUE IS OPTIONAL.
                          437     C  ;                       WHEN NON-ZERO, THIS POINTER IS USED AS A POINTER
                          438     C  ;                       TO A TABLES DESCRIBED AS FOLLOWS :
                          439     C  ;
                          440     C  ;                       BYTE    BYTES/CHARACTER
                          441     C  ;                       BYTE    BLOCK TO LOAD, SHOULD BE ZERO FOR NORMAL
                          442     C  ;                               OPERATION
                          443     C  ;                       WORD    COUNT TO STORE, SHOULD BE 256D FOR NORMAL
                          444     C  ;                               OPERATION
                          445     C  ;                       WORD    CHARACTER OFFSET, SHOULD BE ZERO FOR NORMAL
                          446     C  ;                               OPERATION
                          447     C  ;                       DWORD   POINTER TO A FONT TABLE
                          448     C  ;                       BYTE    DISPLAYABLE ROWS
                          449     C  ;                               IF 'FF' THE MAXIMUM CALCULATED VALUE WILL BE
                          450     C  ;                               USED, ELSE THIS VALUE WILL BE USED
                          451     C  ;                       BYTE    CONSECUTIVE BYTES OF MODE VALUES FOR WHICH
                          452     C  ;                               THIS FONT DESCRIPTION IS TO BE USED.
                          453     C  ;                               THE END OF THIS STREAM IS INDICATED BY A
                          454     C  ;                               BYTE CODE OF 'FF'
                          455     C  ;
                          456     C  ;                       NOTE : USE OF THIS POINTER MAY CAUSE UNEXPECTED
                          457     C  ;                              CURSOR TYPE OPERATION. FOR AN EXPLANATION
                          458     C  ;                              OF CURSOR TYPE SEE AH = 01 IN THE INTERFACE
                          459     C  ;                              SECTION.
                          460     C  ;
                          461     C  ;       DWORD_4         GRAPHICS MODE AUXILIARY POINTER
                          462     C  ;                       INITIALIZED TO 0000:0000, THIS VALUE IS OPTIONAL.
                          463     C  ;                       WHEN NON-ZERO, THIS POINTER IS USED AS A POINTER
                          464     C  ;                       TO A TABLES DESCRIBED AS FOLLOWS :
                          465     C  ;
                          466     C  ;                       BYTE    DISPLAYABLE ROWS
                          467     C  ;                       WORD    BYTES PER CHARACTER
                          468     C  ;                       DWORD   POINTER TO A FONT TABLE
                          469     C  ;                       BYTE    CONSECUTIVE BYTES OF MODE VALUES FOR WHICH
                          470     C  ;                               THIS FONT DESCRIPTION IS TO BE USED.
                          471     C  ;                               THE END OF THIS STREAM IS INDICATED BY A
                          472     C  ;                               BYTE CODE OF 'FF'
                          473     C  ;
                          474     C  ;       DWORD_5 THRU DWORD_7
                          475     C  ;                       RESERVED AND SET TO 0000:0000.
                          476     C  ;
                          477     C
0500                      478     C          ORG     0500H
0500  ??                  479     C  STATUS_BYTE     DB      ?
0501                      480     C  ABS0    ENDS
                          481     C
= 0061                    482     C  PORT_B          EQU     61H                     ; 8255 PORT B ADDR
= 0040                    483     C  TIMER           EQU     40H
                          484     C
                          485     C  ;----- EQUATES FOR CARD PORT ADDRESSES
                          486     C
= 00C4                    487     C          SEQ_ADDR        EQU     0C4H
= 00C5                    488     C          SEQ_DATA        EQU     0C5H
= 00D4                    489     C          CRTC_ADDR       EQU     0D4H
= 00B4                    490     C          CRTC_ADDR_B     EQU     0B4H
= 00D5                    491     C          CRTC_DATA       EQU     0D5H            ; OR 0B5H
= 00CC                    492     C          GRAPH_1_POS     EQU     0CCH
= 00CA                    493     C          GRAPH_2_POS     EQU     0CAH
= 00CE                    494     C          GRAPH_ADDR      EQU     0CEH
= 00CF                    495     C          GRAPH_DATA      EQU     0CFH
= 00C2                    496     C          MISC_OUTPUT     EQU     0C2H
= 00C2                    497     C          IN_STAT_0       EQU     0C2H
= 00BA                    498     C          INPUT_STATUS_B  EQU     0BAH
= 00DA                    499     C          INPUT_STATUS    EQU     0DAH
= 00DA                    500     C          ATTR_READ       EQU     0DAH
= 00C0                    501     C          ATTR_WRITE      EQU     0C0H
                          502     C
                          503     C  ;----- EQUATES FOR ADDRESS REGISTER VALUES
                          504     C
```

```
= 0000                          505     C       S_RESET         EQU     00H
= 0001                          506     C       S_CLOCK         EQU     01H
= 0002                          507     C       S_MAP           EQU     02H
= 0003                          508     C       S_CGEN          EQU     03H
= 0004                          509     C       S_MEM           EQU     04H
                                510     C
= 0000                          511     C       C_HRZ_TOT       EQU     00H
= 0001                          512     C       C_HRZ_DSP       EQU     01H
= 0002                          513     C       C_STRT_HRZ_BLK  EQU     02H
= 0003                          514     C       C_END_HRZ_BLK   EQU     03H
= 0004                          515     C       C_STRT_HRZ_SYN  EQU     04H
= 0005                          516     C       C_END_HRZ_SYN   EQU     05H
= 0006                          517     C       C_VRT_TOT       EQU     06H
= 0007                          518     C       C_OVERFLOW      EQU     07H
= 0008                          519     C       C_PRE_ROW       EQU     08H
= 0009                          520     C       C_MAX_SCAN_LN   EQU     09H
= 000A                          521     C       C_CRSR_START    EQU     0AH
= 000B                          522     C       C_CRSR_END      EQU     0BH
= 000C                          523     C       C_STRT_HGH      EQU     0CH
= 000D                          524     C       C_STRT_LOW      EQU     0DH
= 000E                          525     C       C_CRSR_LOC_HGH  EQU     0EH
= 000F                          526     C       C_CRSR_LOC_LOW  EQU     0FH
= 0010                          527     C       C_VRT_SYN_STRT  EQU     10H             ; WRITE ONLY
= 0010                          528     C       C_LGHT_PEN_HGH  EQU     10H             ; READ ONLY
= 0011                          529     C       C_VRT_SYN_END   EQU     11H             ; WRITE ONLY
= 0011                          530     C       C_LGHT_PEN_LOW  EQU     11H             ; READ ONLY
= 0012                          531     C       C_VRT_DSP_END   EQU     12H
= 0013                          532     C       C_OFFSET        EQU     13H
= 0014                          533     C       C_UNDERLN_LOC   EQU     14H
= 0015                          534     C       C_STRT_VRT_BLK  EQU     15H
= 0016                          535     C       C_END_VRT_BLK   EQU     16H
= 0017                          536     C       C_MODE_CNTL     EQU     17H
= 0018                          537     C       C_LN_COMP       EQU     18H
                                538     C
= 0000                          539     C       G_SET_RESET     EQU     00H
= 0001                          540     C       G_ENBL_SET      EQU     01H
= 0002                          541     C       G_CLR_COMP      EQU     02H
= 0003                          542     C       G_DATA_ROT      EQU     03H
= 0004                          543     C       G_READ_MAP      EQU     04H
= 0005                          544     C       G_MODE          EQU     05H
= 0006                          545     C       G_MISC          EQU     06H
= 0007                          546     C       G_COLOR         EQU     07H
= 0008                          547     C       G_BIT_MASK      EQU     08H
                                548     C
= 0010                          549     C       P_MODE          EQU     10H
= 0011                          550     C       P_OVERSC        EQU     11H
= 0012                          551     C       P_CPLANE        EQU     12H
= 0013                          552     C       P_HPEL          EQU     13H
                                553     C
                                554     C       SUBTTL
                                555
                                556         ;----- CODE SEGMENT
                                557
0000                            558         CODE    SEGMENT PUBLIC
                                559
                                560     C           INCLUDE         VPOST.INC
                                561     C           SUBTTL  VPOST.INC
                                562     C           PAGE
                                563     C
                                564     C   ;----- POST
                                565     C
                                566     C           ASSUME  CS:CODE,DS:ABSO
0000                            567     C           ORG     0H
0000  55                        568     C           DB      055H                        ; SIGNATURE
0001  AA                        569     C           DB      0AAH                        ;  BYTES
0002  20                        570     C           DB      020H                        ; LENGTH INDICATOR
                                571     C
                                572     C   ;----- NOTE : DO NOT USE THE SIGNATURE BYTES AS A PRESENCE TEST
                                573     C
                                574     C   ;       PLANAR VIDEO SWITCH SETTINGS
                                575     C
                                576     C   ;       0 0 - UNUSED
                                577     C   ;       0 1 - 40 X 25 COLOR
                                578     C   ;       1 0 - 80 X 25 COLOR
                                579     C   ;       1 1 - 80 X 25 MONOCHROME
                                580     C   ; NOTE : 0 0 MUST BE SET WHEN THIS ADAPTER IS INSTALLED.
                                581     C
                                582     C   ;       VIDEO ADAPTER SWITCH SETTINGS
                                583     C
                                584     C   ;       0 0 0 0 - MONOC PRIMARY, EGA COLOR, 40X25
                                585     C   ;       0 0 0 1 - MONOC PRIMARY, EGA COLOR, 80X25
                                586     C   ;       0 0 1 0 - MONOC PRIMARY, EGA HI RES EMULATE (SAME AS 0001)
                                587     C   ;       0 0 1 1 - MONOC PRIMARY, EGA HI RES ENHANCED
                                588     C   ;       0 1 0 0 - COLOR 40 PRIMARY, EGA MONOCHROME
                                589     C   ;       0 1 0 1 - COLOR 80 PRIMARY, EGA MONOCHROME
                                590     C
                                591     C   ;       0 1 1 0 - MONOC SECONDARY, EGA COLOR, 40X25
                                592     C   ;       0 1 1 1 - MONOC SECONDARY, EGA COLOR, 80X25
                                593     C   ;       1 0 0 0 - MONOC SECONDARY, EGA HI RES EMULATE (SAME AS 0111)
                                594     C   ;       1 0 0 1 - MONOC SECONDARY, EGA HI RES ENHANCED
                                595     C   ;       1 0 1 0 - COLOR 40 SECONDARY, EGA MONOCHROME
                                596     C   ;       1 0 1 1 - COLOR 80 SECONDARY, EGA MONOCHROME
                                597     C
                                598     C   ;       1 1 0 0 - RESERVED
                                599     C   ;       1 1 0 1 - RESERVED
                                600     C   ;       1 1 1 0 - RESERVED
                                601     C   ;       1 1 1 1 - RESERVED
                                602     C
                                603     C   ;----- SETUP ROUTINE FOR THIS MODULE
                                604     C
0003                            605     C   VIDEO_SETUP       PROC    FAR
0003  EB 28                     606     C           JMP     SHORT   L1
0005  32 34 30 30              607     C           DB      '2400'
0009  36 32 37 37 33 35        608     C           DB      '6277356 (C)COPYRIGHT IBM 1984'
      36 20 28 43 29 43
      4F 50 59 52 49 47
      48 54 20 49 42 4D
      20 31 39 38 34
0026  39 2F 31 33 2F 38        613     C           DB      '9/13/84'
      34
                                614     C
                                615     C
                                616     C   ;----- SET UP VIDEO VECTORS
                                617     C
002D                            618     C   L1:
002D  B6 03                     619     C           MOV     DH,3
002F  B2 DA                     620     C           MOV     DL,INPUT_STATUS
0031  EC                        621     C           IN      AL,DX
0032  B2 BA                     622     C           MOV     DL,INPUT_STATUS_B
0034  EC                        623     C           IN      AL,DX
0035  B2 C0                     624     C           MOV     DL,ATTR_WRITE
0037  B0 00                     625     C           MOV     AL,0
0039  EE                        626     C           OUT     DX,AL
                                627     C
                                628     C           SRLOAD  DS,0
003A  2B D2                     629     C+          SUB     DX,DX
003C  8E DA                     630     C+          MOV     DS,DX
```

```
003E  FA                          631      C          CLI
003F  C7 06 0040 R 0CD7 R         632      C          MOV     WORD PTR VIDEO,OFFSET COMBO_VIDEO
0045  8C 0E 0042 R                633      C          MOV     WORD PTR VIDEO+2, CS
0049  C7 06 0108 R F065           634      C          MOV     WORD PTR PLANAR_VIDEO,0F065H
004F  C7 06 010A R F000           635      C          MOV     WORD PTR PLANAR_VIDEO+2,0F000H
0055  C7 06 04A8 R 010C R         636      C          MOV     WORD PTR SAVE_PTR,OFFSET SAVE_TBL
005B  8C 0E 04AA R                637      C          MOV     WORD PTR SAVE_PTR+2, CS
005F  C7 06 007C R 0000 E         638      C          MOV     WORD PTR EXT_PTR, OFFSET INT_1F_1
0065  8C 0E 007E R                639      C          MOV     WORD PTR EXT_PTR+2, CS
0069  C7 06 010C R 0000 E         640      C          MOV     WORD PTR GRX_SET, OFFSET CGDDOT
006F  8C 0E 010E R                641      C          MOV     WORD PTR GRX_SET+2, CS
0073  FB                          642      C          STI
                                  643      C
                                  644      C ;----- POST FOR COMBO VIDEO CARD
                                  645      C
0074  C6 06 0487 R 04             646      C          MOV     INFO,00000100B
0079  E8 009B R                   647      C          CALL    RD_SWS
007C  88 1E 0488 R                648      C          MOV     INFO_3,BL
0080  E8 00CE R                   649      C          CALL    F_BTS
0083  08 06 0488 R                650      C          OR      INFO_3,AL
0087  8A 1E 0488 R                651      C          MOV     BL,INFO_3
008B  E8 00F3 R                   652      C          CALL    MK_ENV
008E  E9 0244 R                   653      C          JMP     POST
0091                              654      C SKIP:
0091  CB                          655      C          RET
0092                              656      C VIDEO_SETUP      ENDP
                                  657      C
                                  658      C
0092                              659      C POR_1    PROC    NEAR
0092  EE                          660      C          OUT     DX,AL
0093  50                          661      C          PUSH    AX
0094  58                          662      C          POP     AX
0095  EC                          663      C          IN      AL,DX
0096  24 10                       664      C          AND     AL,010H
0098  D0 E8                       665      C          SHR     AL,1
009A  C3                          666      C          RET
009B                              667      C POR_1    ENDP
                                  668      C
                                  669      C ;----- READ THE SWITCH SETTINGS ON THE CARD
                                  670      C
009B                              671      C RD_SWS   PROC    NEAR
                                  672      C          ASSUME  DS:ABS0
009B  B6 03                       673      C          MOV     DH,3
009D  B2 C2                       674      C          MOV     DL,MISC_OUTPUT
009F  B0 01                       675      C          MOV     AL,1
00A1  EE                          676      C          OUT     DX,AL
                                  677      C
                                  678      C ;----- COULD BE 0,4,8,C
                                  679      C
00A2  B0 0D                       680      C          MOV     AL,0DH
00A4  E8 0092 R                   681      C          CALL    POR_1
00A7  D0 E8                       682      C          SHR     AL,1
00A9  D0 E8                       683      C          SHR     AL,1
00AB  D0 E8                       684      C          SHR     AL,1
00AD  8A D8                       685      C          MOV     BL,AL
                                  686      C
00AF  B0 09                       687      C          MOV     AL,9
00B1  E8 0092 R                   688      C          CALL    POR_1
00B4  D0 E8                       689      C          SHR     AL,1
00B6  D0 E8                       690      C          SHR     AL,1
00B8  0A D8                       691      C          OR      BL,AL
                                  692      C
00BA  B0 05                       693      C          MOV     AL,5
00BC  E8 0092 R                   694      C          CALL    POR_1
00BF  D0 E8                       695      C          SHR     AL,1
00C1  0A D8                       696      C          OR      BL,AL
                                  697      C
00C3  B0 01                       698      C          MOV     AL,1
00C5  E8 0092 R                   699      C          CALL    POR_1
00C8  0A D8                       700      C          OR      BL,AL
                                  701      C
00CA  80 E3 0F                    702      C          AND     BL,0FH
00CD  C3                          703      C          RET
00CE                              704      C RD_SWS   ENDP
                                  705      C
                                  706      C ;----- OBTAIN THE FEATURE BITS FROM DAUGHTER CARD
                                  707      C
00CE                              708      C F_BTS    PROC    NEAR
00CE  B6 03                       709      C          MOV     DH,3
00D0  B2 BA                       710      C          MOV     DL,0BAH
00D2  B0 01                       711      C          MOV     AL,1
00D4  EE                          712      C          OUT     DX,AL
00D5  B2 DA                       713      C          MOV     DL,0DAH
00D7  EE                          714      C          OUT     DX,AL
00D8  B2 C2                       715      C          MOV     DL,IN_STAT_0
00DA  EC                          716      C          IN      AL,DX                          ; READ FEATURE BITS
00DB  24 60                       717      C          AND     AL,060H
00DD  D0 E8                       718      C          SHR     AL,1
00DF  8A D8                       719      C          MOV     BL,AL
00E1  B2 BA                       720      C          MOV     DL,0BAH
00E3  B0 02                       721      C          MOV     AL,2
00E5  EE                          722      C          OUT     DX,AL
00E6  B2 DA                       723      C          MOV     DL,0DAH
00E8  EE                          724      C          OUT     DX,AL
00E9  B2 C2                       725      C          MOV     DL,IN_STAT_0
00EB  EC                          726      C          IN      AL,DX                          ; READ FEATURE BITS
00EC  24 60                       727      C          AND     AL,060H
00EE  D0 E0                       728      C          SHL     AL,1
00F0  0A C3                       729      C          OR      AL,BL
00F2  C3                          730      C          RET
00F3                              731      C F_BTS    ENDP
                                  732      C
                                  733      C ;----- ESTABLISH THE VIDEO ENVIRONMENT, KEYED OFF OF THE SWITCHES
                                  734      C
00F3                              735      C MK_ENV   PROC    NEAR
                                  736      C          ASSUME  DS:ABS0
00F3  2A FF                       737      C          SUB     BH,BH
00F5  80 E3 0F                    738      C          AND     BL,0FH
00F8  D1 E3                       739      C          SAL     BX,1
00FA  52                          740      C          PUSH    DX
00FB  B6 03                       741      C          MOV     DH,3
00FD  8A E6                       742      C          MOV     AH,DH
00FF  5A                          743      C          POP     DX
0100  80 E4 01                    744      C          AND     AH,1
0103  FE C4                       745      C          INC     AH
0105  F6 D4                       746      C          NOT     AH
0107  2E: FF A7 0128 R            747      C          JMP     WORD PTR CS:[BX + OFFSET T5]
                                  748      C
010C                              749      C SAVE_TBL         LABEL   DWORD
010C  0717 R                      750      C          DW      OFFSET  VIDEO_PARMS            ; PARMS
010E  C000                        751      C          DW      0C000H                         ; PARMS
0110  0000                        752      C          DW      0                              ; PAL SAVE AREA
0112  0000                        753      C          DW      0                              ; PAL SAVE AREA
0114  0000                        754      C          DW      0                              ; ALPHA TABLES
0116  0000                        755      C          DW      0                              ; ALPHA TABLES
0118  0000                        756      C          DW      0                              ; GRAPHICS TABLES
```

```
011A  0000                    757     C               DW      0                       ; GRAPHICS TABLES
                              758     C
011C  0000                    759     C               DW      0
011E  0000                    760     C               DW      0
0120  0000                    761     C               DW      0
0122  0000                    762     C               DW      0
0124  0000                    763     C               DW      0
0126  0000                    764     C               DW      0
                              765     C
0128                          766     C  T5           LABEL   WORD
0128  0173 R                 767     C               DW      OFFSET PST_0
012A  017E R                 768     C               DW      OFFSET PST_1
012C  017E R                 769     C               DW      OFFSET PST_2
012E  0189 R                 770     C               DW      OFFSET PST_3
0130  0194 R                 771     C               DW      OFFSET PST_4
0132  01A8 R                 772     C               DW      OFFSET PST_5
0134  01BC R                 773     C               DW      OFFSET PST_6
0136  01C7 R                 774     C               DW      OFFSET PST_7
                              775     C
0138  01C7 R                 776     C               DW      OFFSET PST_8
013A  01D2 R                 777     C               DW      OFFSET PST_9
013C  01DD R                 778     C               DW      OFFSET PST_A
013E  01F1 R                 779     C               DW      OFFSET PST_B
0140  0204 R                 780     C               DW      OFFSET PST_OUT
0142  0204 R                 781     C               DW      OFFSET PST_OUT
0144  0204 R                 782     C               DW      OFFSET PST_OUT
0146  0204 R                 783     C               DW      OFFSET PST_OUT
                              784     C
0148                          785     C  ENV_X        PROC    NEAR                    ; SET 40X25 COLOR ALPHA
0148  80 26 0410 R CF        786     C               AND     EQUIP_LOW,0CFH
014D  80 0E 0410 R 10        787     C               OR      EQUIP_LOW,010H
0152  B8 0001                788     C               MOV     AX,1H
0155  CD 10                  789     C               INT     10H
0157  C3                    790     C               RET
0158                          791     C  ENV_X        ENDP
                              792     C
0158                          793     C  ENV_0        PROC    NEAR                    ; SET 80X25 COLOR ALPHA
0158  80 26 0410 R CF        794     C               AND     EQUIP_LOW,0CFH
015D  80 0E 0410 R 20        795     C               OR      EQUIP_LOW,020H
0162  B8 0003                796     C               MOV     AX,03H
0165  CD 10                  797     C               INT     10H
0167  C3                    798     C               RET
0168                          799     C  ENV_0        ENDP
                              800     C
0168                          801     C  ENV_3        PROC    NEAR                    ; SET MONOCHROME ALPHA
0168  80 0E 0410 R 30        802     C               OR      EQUIP_LOW,030H
016D  B8 0007                803     C               MOV     AX,07H
0170  CD 10                  804     C               INT     10H
0172  C3                    805     C               RET
0173                          806     C  ENV_3        ENDP
                              807     C
                              808     C
0173                          809     C  PST_0:
0173  20 26 0487 R            810     C               AND     INFO,AH
0177  E8 0148 R               811     C               CALL    ENV_X
017A  E8 0168 R               812     C               CALL    ENV_3
017D  C3                      813     C               RET
017E                          814     C  PST_1:
017E                          815     C  PST_2:
017E  20 26 0487 R            816     C               AND     INFO,AH
0182  E8 0158 R               817     C               CALL    ENV_0
0185  E8 0168 R               818     C               CALL    ENV_3
0188  C3                      819     C               RET
0189                          820     C  PST_3:
0189  20 26 0487 R            821     C               AND     INFO,AH
018D  E8 0158 R               822     C               CALL    ENV_0
0190  E8 0168 R               823     C               CALL    ENV_3
0193  C3                      824     C               RET
0194                          825     C  PST_4:
0194  B6 03                   826     C               MOV     DH,3
0196  B2 C2                   827     C               MOV     DL,MISC_OUTPUT
0198  B0 00                   828     C               MOV     AL,0
019A  EE                      829     C               OUT     DX,AL
019B  F6 D4                   830     C               NOT     AH
019D  08 26 0487 R            831     C               OR      INFO,AH
01A1  E8 0168 R               832     C               CALL    ENV_3
01A4  E8 0148 R               833     C               CALL    ENV_X
01A7  C3                      834     C               RET
01A8                          835     C  PST_5:
01A8  B6 03                   836     C               MOV     DH,3
01AA  B2 C2                   837     C               MOV     DL,MISC_OUTPUT
01AC  B0 00                   838     C               MOV     AL,0
01AE  EE                      839     C               OUT     DX,AL
01AF  F6 D4                   840     C               NOT     AH
01B1  08 26 0487 R            841     C               OR      INFO,AH
01B5  E8 0168 R               842     C               CALL    ENV_3
01B8  E8 0158 R               843     C               CALL    ENV_0
01BB  C3                      844     C               RET
01BC                          845     C  PST_6:
01BC  20 26 0487 R            846     C               AND     INFO,AH
01C0  E8 0168 R               847     C               CALL    ENV_3
01C3  E8 0148 R               848     C               CALL    ENV_X
01C6  C3                      849     C               RET
01C7                          850     C  PST_7:
01C7                          851     C  PST_8:
01C7  20 26 0487 R            852     C               AND     INFO,AH
01CB  E8 0168 R               853     C               CALL    ENV_3
01CE  E8 0158 R               854     C               CALL    ENV_0
01D1  C3                      855     C               RET
01D2                          856     C  PST_9:
01D2  20 26 0487 R            857     C               AND     INFO,AH
01D6  E8 0168 R               858     C               CALL    ENV_3
01D9  E8 0158 R               859     C               CALL    ENV_0
01DC  C3                      860     C               RET
01DD                          861     C  PST_A:
01DD  B6 03                   862     C               MOV     DH,3
01DF  B2 C2                   863     C               MOV     DL,MISC_OUTPUT
01E1  B0 00                   864     C               MOV     AL,0
01E3  EE                      865     C               OUT     DX,AL
01E4  F6 D4                   866     C               NOT     AH
01E6  08 26 0487 R            867     C               OR      INFO,AH
01EA  E8 0148 R               868     C               CALL    ENV_X
01ED  E8 0168 R               869     C               CALL    ENV_3
01F0  C3                      870     C               RET
01F1                          871     C  PST_B:
01F1  B6 03                   872     C               MOV     DH,3
01F3  B2 C2                   873     C               MOV     DL,MISC_OUTPUT
01F5  B0 00                   874     C               MOV     AL,0
01F7  EE                      875     C               OUT     DX,AL
01F8  F6 D4                   876     C               NOT     AH
01FA  08 26 0487 R            877     C               OR      INFO,AH
01FE  E8 0158 R               878     C               CALL    ENV_0
0201  E8 0168 R               879     C               CALL    ENV_3
0204                          880     C  PST_OUT:
0204  C3                      881     C               RET
0205                          882     C  MK_ENV       ENDP
```

```
                                883     C
                                884     C   ;--------------------------------------------------------------------------------
                                885     C   ; THIS ROUTINE TESTS THE CRT CARD INTERNAL DATA BUS AND IN A LIMITED    :
                                886     C   ; WAY TESTS THE CRTC VIDEO CHIP BY WRITING/READING FROM CURSOR REGISTER :
                                887     C   ; CARRY IS SET IF AN ERROR IS FOUND                                       :
                                888     C   ;                                                                         :
                                889     C   ; REGISTERS BX,SI,ES,DS ARE PRESERVED.                                   :
                                890     C   ; REGISTERS AX,CX,DX ARE MODIFYED.                                       :
                                891     C   ;--------------------------------------------------------------------------------
0205                            892     C   CD_PRESENCE_TST PROC    NEAR
0205  53                        893     C           PUSH    BX                      ; SAVE BX
0206  BB 007F                   894     C           MOV     BX,07FH                 ; INITIAL WORD PATTERN BYTE
0209  8B FB                     895     C           MOV     DI,BX
020B  50                        896     C           PUSH    AX                      ; SAVE PORT ADDRESS
020C  E8 022C R                 897     C           CALL    RD_CURSOR               ;
020F  8B F0                     898     C           MOV     SI,AX                   ; SAVE ORIGINAL VALUE
0211  58                        899     C           POP     AX                      ; RECOVER PORT ADDRESS
0212  50                        900     C           PUSH    AX                      ; SAVE PORT ADDRESS
0213  E8 0236 R                 901     C           CALL    WR_CURSOR               ; WRITE CURSOR
0216  58                        902     C           POP     AX                      ; RECOVER PORT ADDRESS
0217  50                        903     C           PUSH    AX                      ; SAVE PORT ADDRESS
0218  E8 022C R                 904     C           CALL    RD_CURSOR               ; READ IT BACK
021B  3B C7                     905     C           CMP     AX,DI                   ; SAME?
021D  58                        906     C           POP     AX
021E  75 03                     907     C           JNZ     NOT_PRESENT             ; EXIT IF NOT EQUAL
0220  EB 05 90                  908     C           JMP     TST_EX
0223                            909     C   NOT_PRESENT:
0223  33 C0                     910     C           XOR     AX,AX                   ; SET NOT PRESENT
0225  5B                        911     C           POP     BX
0226  C3                        912     C           RET
0227                            913     C   TST_EX:
0227  B8 0001                   914     C           MOV     AX,1                    ; SET PRESENT ON EXIT
022A  5B                        915     C           POP     BX                      ; RESTORE BX
022B  C3                        916     C           RET
022C                            917     C   CD_PRESENCE_TST ENDP
                                918     C
                                919     C   ;--------------------------------------------------------------------------------
                                920     C   ; MODULE NAME  RD_CURSOR                                                    :
                                921     C   ;  READ CURSOR POSITION [ADDRESS] (FROM CRTC) TO  AX                      :
                                922     C   ;                                                                         :
                                923     C   ; REGISTER AX IS MODIFIED.                                               :
                                924     C   ;--------------------------------------------------------------------------------
022C                            925     C   RD_CURSOR         PROC NEAR
022C  52                        926     C           PUSH    DX                      ; SAVE REGS USED
022D  8B D0                     927     C           MOV     DX,AX
022F  B0 0E                     928     C           MOV     AL,C_CRSR_LOC_HGH
0231  EE                        929     C           OUT     DX,AL
0232  42                        930     C           INC     DX
0233  EC                        931     C           IN      AL,DX
                                932     C
                                933     C                                           ; RETURN WITH CURSOR POS IN AX
                                                                                    ; RESTORE REGS USED
0234  5A                        934     C           POP     DX
0235  C3                        935     C           RET
0236                            936     C   RD_CURSOR         ENDP
                                937     C
                                938     C   ;--------------------------------------------------------------------------------
                                939     C   ; MODULE NAME  WR_CURSOR                                                    :
                                940     C   ;  WRITE CURSOR POSITION [ADDRESS] (TO CRTC) WITH CONTENTS OF AX          :
                                941     C   ;                                                                         :
                                942     C   ; ALL REGISTERS PRESERVED                                                 :
                                943     C   ;--------------------------------------------------------------------------------
0236                            944     C   WR_CURSOR         PROC NEAR
                                945     C
                                                                                    ; SAVE REGS USED
0236  50                        946     C           PUSH    AX
0237  52                        947     C           PUSH    DX
0238  8B D0                     948     C           MOV     DX,AX
023A  B4 0E                     949     C           MOV     AH,C_CRSR_LOC_HGH       ; CURSOR LOCATION HIGH INDEX
023C  B0 7F                     950     C           MOV     AL,07FH                 ; TEST VALUE
023E  E8 0D15 R                 951     C           CALL    OUT_DX
                                952     C
                                953     C                                           ; RETURN WITH CURSOR POS IN AX
                                                                                    ; RESTORE REGS USED
0241  5A                        954     C           POP     DX
0242  58                        955     C           POP     AX
0243  C3                        956     C           RET
0244                            957     C   WR_CURSOR         ENDP
                                958     C
0244                            959     C   POST:
                                960     C   ;-------------------------------------------------
                                961     C   ;       INITIALIZE AND START CRT CONTROLLER (6845)       :
                                962     C   ;       ON COLOR GRAPHICS AND MONOCHROME CARDS           :
                                963     C   ;       TEST VIDEO READ/WRITE STORAGE.                   :
                                964     C   ; DESCRIPTION                                            :
                                965     C   ;       RESET THE VIDEO ENABLE SIGNAL.                   :
                                966     C   ;       SELECT ALPHANUMERIC MODE, 40 * 25, B & W.        :
                                967     C   ;       READ/WRITE DATA PATTERNS TO STG. CHECK STG       :
                                968     C   ;       ADDRESSABILITY.                                  :
                                969     C   ;-------------------------------------------------
                                970     C           ASSUME  DS:ABS0,ES:ABS0
0244  E8 0CFE R                 971     C           CALL    DDS
0247  F6 06 0487 R 02           972     C           TEST    INFO,2
024C  75 12                     973     C           JNZ     COLOR_PRESENCE_TST
024E  B8 03B4                   974     C           MOV     AX,03B4H
0251  E8 0205 R                 975     C           CALL    CD_PRESENCE_TST
0254  3D 0001                   976     C           CMP     AX,1
0257  74 03                     977     C           JE      CONT1
0259  E9 0317 R                 978     C           JMP     POD14
025C                            979     C   CONT1:
025C  B4 30                     980     C           MOV     AH,30H                  ; MONOCHROME CARD INSTALLED
025E  EB 10                     981     C           JMP     SHORT   OVER
0260                            982     C   COLOR_PRESENCE_TST:
0260  B8 03D4                   983     C           MOV     AX,03D4H
0263  E8 0205 R                 984     C           CALL    CD_PRESENCE_TST
0266  3D 0001                   985     C           CMP     AX,1
0269  74 03                     986     C           JE      CONT2
026B  E9 0317 R                 987     C           JMP     POD14
026E                            988     C   CONT2:
026E  B4 20                     989     C           MOV     AH,20H                  ; COLOR GRAPHICS CARD INSTALLED
0270                            990     C   OVER:
0270  50                        991     C           PUSH    AX                      ; RESAVE VALUE
0271  BB B000                   992     C           MOV     BX,0B000H               ; BEG VIDEO RAM ADDR B/W CD
0274  BA 03B8                   993     C           MOV     DX,3B8H                 ; MODE CONTROL B/W
0277  B9 1000                   994     C           MOV     CX,4096                 ; RAM BYTE CNT FOR B/W CD
027A  B0 01                     995     C           MOV     AL,1                    ; SET MODE FOR BW CARD
027C  80 FC 30                  996     C           CMP     AH,30H                  ; B/W VIDEO CARD ATTACHED?
027F  74 08                     997     C           JE      E9                      ; YES - GO TEST VIDEO STG
0281  B7 B8                     998     C           MOV     BH,0B8H                 ; BEG VIDEO RAM ADDR COLOR CD
0283  B2 D8                     999     C           MOV     DL,0D8H                 ; MODE CONTROL COLOR
0285  B5 40                    1000     C           MOV     CH,40H                  ; RAM BYTE CNT FOR COLOR CD
0287  FE C8                    1001     C           DEC     AL                      ; SET MODE TO 0 FOR COLOR CD
0289                           1002     C   E9:                                    ; TEST_VIDEO_STG:
0289  EE                       1003     C           OUT     DX,AL                   ; DISABLE VIDEO FOR COLOR CD
                               1004     C
028A  8B 2E 0472 R             1005     C           MOV     BP,DS:RESET_FLAG        ; POD INITIALIZED BY KBD RESET
                               1006     C
028E  81 FD 1234               1007     C           CMP     BP,1234H                ; POD INITIATED BY KBD RESET?
0292  8E C3                    1008     C           MOV     ES,BX                   ; POINT ES TO VIDEO RAM STG
```

```
0294  74 07                1009    C          JE      E10                     ; YES - SKIP VIDEO RAM TEST
0296  8E DB                1010    C          MOV     DS,BX                   ; POINT DS TO VIDEO RAM STG
                           1011    C          ASSUME  DS:NOTHING,ES:NOTHING
0298  E8 02DF R            1012    C          CALL    STGTST_CNT              ; GO TEST VIDEO R/W STG
029B  75 2E                1013    C          JNE     E17                     ; R/W STG FAILURE - BEEP SPK
                           1014    C  ;--------------------------------------------------------------------
                           1015    C  ;       SETUP VIDEO DATA ON SCREEN FOR VIDEO LINE TEST.          :
                           1016    C  ; DESCRIPTION                                                    :
                           1017    C  ;       ENABLE VIDEO SIGNAL AND SET MODE.                        :
                           1018    C  ;       DISPLAY A HORIZONTAL BAR ON SCREEN.                      :
                           1019    C  ;--------------------------------------------------------------------
029D                       1020    C  E10:
029D  58                   1021    C          POP     AX                      ; GET VIDEO SENSE SWS (AH)
029E  50                   1022    C          PUSH    AX                      ; SAVE IT
029F  B8 7020              1023    C          MOV     AX,7020H                ; WRT BLANKS IN REVERSE VIDEO
02A2  2B FF                1024    C          SUB     DI,DI                   ; SETUP STARTING LOC
02A4  B9 0028              1025    C          MOV     CX,40                   ; NO. OF BLANKS TO DISPLAY
02A7  F3/ AB               1026    C          REP     STOSW                   ; WRITE VIDEO STORAGE
                           1027    C  ;--------------------------------------------------------------------
                           1028    C  ;       CRT INTERFACE LINES TEST                                 :
                           1029    C  ; DESCRIPTION                                                    :
                           1030    C  ;       SENSE ON/OFF TRANSITION OF THE VIDEO ENABLE              :
                           1031    C  ;       AND HORIZONTAL SYNC LINES.                               :
                           1032    C  ;--------------------------------------------------------------------
02A9  58                   1033    C          POP     AX                      ; GET VIDEO SENSE SW INFO
02AA  50                   1034    C          PUSH    AX                      ; SAVE IT
02AB  80 FC 30             1035    C          CMP     AH,30H                  ; B/W CARD ATTACHED?
02AE  BA 03BA              1036    C          MOV     DX,03BAH                ; SETUP ADDR OF BW STATUS PORT
02B1  74 02                1037    C          JE      E11                     ; YES - GO TEST LINES
02B3  B2 DA                1038    C          MOV     DL,0DAH                 ; COLOR CARD IS ATTACHED
02B5                       1039    C  E11:                                    ; LINE_TST:
02B5  B4 08                1040    C          MOV     AH,8
02B7                       1041    C  E12:                                    ; OFLOOP_CNT:
02B7  2B C9                1042    C          SUB     CX,CX
02B9                       1043    C  E13:
02B9  EC                   1044    C          IN      AL,DX                   ; READ CRT STATUS PORT
02BA  22 C4                1045    C          AND     AL,AH                   ; CHECK VIDEO/HORZ LINE
02BC  75 04                1046    C          JNZ     E14                     ; ITS ON - CHECK IF IT GOES OFF
02BE  E2 F9                1047    C          LOOP    E13                     ; LOOP TILL ON OR TIMEOUT
02C0  EB 09                1048    C          JMP     SHORT E17               ; GO PRINT ERROR MSG
02C2                       1049    C  E14:
02C2  2B C9                1050    C          SUB     CX,CX
02C4                       1051    C  E15:
02C4  EC                   1052    C          IN      AL,DX                   ; READ CRT STATUS PORT
02C5  22 C4                1053    C          AND     AL,AH                   ; CHECK VIDEO/HORZ LINE
02C7  74 0A                1054    C          JZ      E16                     ; ITS ON - CHECK NEXT LINE
02C9  E2 F9                1055    C          LOOP    E15                     ; LOOP IF OFF TILL IT GOES ON
02CB                       1056    C  E17:                                    ;  CRT_ERR
02CB  BA 0102              1057    C          MOV     DX,102H
02CE  E8 06C8 R            1058    C          CALL    ERR_BEEP                ; GO BEEP SPEAKER
02D1  EB 06                1059    C          JMP     SHORT E18
02D3                       1060    C  E16:                                    ; NXT_LINE
02D3  B1 03                1061    C          MOV     CL,3                    ; GET NEXT BIT TO CHECK
02D5  D2 EC                1062    C          SHR     AH,CL
02D7  75 DE                1063    C          JNZ     E12                     ; GO CHECK HORIZONTAL LINE
02D9                       1064    C  E18:                                    ;  DISPLAY_CURSOR:
02D9  58                   1065    C          POP     AX                      ; GET VIDEO SENSE SWS (AH)
02DA  EB 3B                1066    C          JMP     SHORT   POD14
                           1067    C
                           1068    C  ;--------------------------------------------------------------------
                           1069    C  ;       THIS SUBROUTINE PERFORMS A READ/WRITE STORAGE TEST ON     :
                           1070    C  ;       A 16K BLOCK OF STORAGE.                                  :
                           1071    C  ; ENTRY REQUIREMENTS:                                            :
                           1072    C  ;       ES = ADDRESS OF STORAGE SEGMENT BEING TESTED              :
                           1073    C  ;       DS = ADDRESS OF STORAGE SEGMENT BEING TESTED              :
                           1074    C  ;       WHEN ENTERING AT STGTST_CNT, CX MUST BE LOADED WITH       :
                           1075    C  ;       THE BYTE COUNT.                                          :
                           1076    C  ; EXIT PARAMETERS:                                               :
                           1077    C  ;       ZERO FLAG = 0 IF STORAGE ERROR (DATA COMPARE OR PARITY CHECK.  :
                           1078    C  ;                   AL = 0 DENOTES A PARITY CHECK. ELSE AL=XOR'ED BIT  :
                           1079    C  ;                          PATTERN OF THE EXPECTED DATA PATTERN VS THE  :
                           1080    C  ;                          ACTUAL DATA READ.                           :
                           1081    C  ;       AX,BX,CX,DX,DI, AND SI ARE ALL DESTROYED.                :
                           1082    C  ;--------------------------------------------------------------------
02DC                       1083    C  STGTST  PROC    NEAR
02DC  B9 4000              1084    C          MOV     CX,4000H                ; SETUP CNT TO TEST A 16K BLK
02DF                       1085    C  STGTST_CNT:
02DF  FC                   1086    C          CLD                             ; SET DIR FLAG TO INCREMENT
02E0  8B D9                1087    C          MOV     BX,CX                   ; SAVE CNT (4K FOR VIDEO OR 16K)
02E2  B8 AAAA              1088    C          MOV     AX,0AAAAH               ; GET DATA PATTERN TO WRITE
02E5  BA FF55              1089    C          MOV     DX,0FF55H               ; SETUP OTHER DATA PATTERNS TO USE
02E8  2B FF                1090    C          SUB     DI,DI                   ; DI = OFFSET 0 RELATIVE TO ES REG
02EA  F3/ AA               1091    C          REP     STOSB                   ; WRITE STORAGE LOCATIONS
02EC                       1092    C  C3:                                     ;  STG01
02EC  4F                   1093    C          DEC     DI                      ; POINT TO LAST BYTE JUST WRITTEN
02ED  FD                   1094    C          STD                             ; SET DIR FLAG TO GO BACKWARDS
02EE                       1095    C  C4:
02EE  8B F7                1096    C          MOV     SI,DI
02F0  8B CB                1097    C          MOV     CX,BX                   ; SETUP BYTE CNT
02F2                       1098    C  C5:                                     ; INNER TEST LOOP
02F2  AC                   1099    C          LODSB                           ; READ OLD TEST BYTE [SI]+
02F3  32 C4                1100    C          XOR     AL,AH                   ; DATA READ AS EXPECTED ?
02F5  75 1E                1101    C          JNE     C7                      ;  NO - GO TO ERROR ROUTINE
02F7  8A C2                1102    C          MOV     AL,DL                   ; GET NEXT DATA PATTERN TO WRITE
02F9  AA                   1103    C          STOSB                           ; WRITE INTO LOCATION JUST READ
02FA  E2 F6                1104    C          LOOP    C5                      ;  DECREMENT COUNT AND LOOP CX
                           1105    C
02FC  22 E4                1106    C          AND     AH,AH                   ; ENDING 0 PATTERN WRITTEN TO STG?
02FE  74 13                1107    C          JZ      C6X                     ; YES - RETURN TO CALLER WITH AL=0
0300  8A E0                1108    C          MOV     AH,AL                   ; SETUP NEW VALUE FOR COMPARE
0302  86 F2                1109    C          XCHG    DH,DL                   ; MOVE NEXT DATA PATTERN TO DL
0304  22 E4                1110    C          AND     AH,AH                   ; READING ZERO PATTERN THIS PASS ?
0306  75 04                1111    C          JNZ     C6                      ; CONTINUE TEST SEQUENCE TILL 0
0308  8A D4                1112    C          MOV     DL,AH                   ;  ELSE SET 0 FOR END READ PATTERN
030A  EB E0                1113    C          JMP     C3                      ;  AND MAKE FINAL BACKWARDS PASS
030C                       1114    C  C6:
030C  FC                   1115    C          CLD                             ; SET DIR FLAG TO GO FORWARD
030D  47                   1116    C          INC     DI                      ; SET POINTER TO BEG LOCATION
030E  74 DE                1117    C          JZ      C4                      ; READ/WRITE FORWARD IN STG
0310  4F                   1118    C          DEC     DI                      ; ADJUST POINTER
0311  EB D9                1119    C          JMP     C3                      ; READ/WRITE BACKWARD IN STG
0313                       1120    C  C6X:
0313  B0 00                1121    C          MOV     AL,000H                 ; AL=0 DATA COMPARE OK
0315                       1122    C  C7:
0315  FC                   1123    C          CLD                             ; SET DIRECTION FLAG BACK TO INC
0316  C3                   1124    C          RET
0317                       1125    C  STGTST  ENDP
                           1126    C
                           1127    C  ;--------------------------------------------------------
                           1128    C  ;                EGA CRT ATTACHMENT TEST                 :
                           1129    C  ;                                                        :
                           1130    C  ; 1. INIT CRT TO 40X25 - BW  ****SET TO MODE****        :
                           1131    C  ; 2. CHECK FOR VERTICAL AND VIDEO ENABLES, AND CHECK     :
                           1132    C  ;    TIMING OF SAME                                     :
                           1133    C  ; 3. CHECK VERTICAL INTERRUPT                            :
                           1134    C  ; 4. CHECK RED, BLUE, GREEN, AND INTENSIFY DOTS          :
```

```
                        1135     C  ; 5. INIT TO 40X25 - COLOR/MONO ****SET TO MODE****     :
                        1136     C  ;-------------------------------------------------------
                        1137     C
                        1138     C  ;----- NOMINAL TIME IS B286H FOR 60 HZ.
                        1139     C  ;----- NOMINAL TIME IS A2FEH FOR 50 HZ.
                        1140     C
= A0AC                  1141     C  MAX_VERT_COLOR  EQU     0A0ACH                  ; MAX TIME FOR VERT/VERT
                        1142     C                                                  ;  (NOMINAL + 10%)
= C460                  1143     C  MIN_VERT_COLOR  EQU     0C460H                  ; MIN TIME FOR VERT/VERT
                        1144     C                                                  ;  (NOMINAL - 10%)
= 00C8                  1145     C  CENAB_PER_FRAME EQU     200                     ; NUM OF ENABLES PER FRAME
= 8D99                  1146     C  MAX_VERT_MONO   EQU     08D99H                  ; MAX TIME FOR VERT/VERT
                        1147     C                                                  ;  (NOMINAL + 10%)
= B862                  1148     C  MIN_VERT_MONO   EQU     0B862H                  ; MIN TIME FOR VERT/VERT
                        1149     C                                                  ;  (NOMINAL - 10%)
= 015E                  1150     C  EENAB_PER_FRAME EQU     350                     ; ENHANCED ENABLES PER FRAME
= 015E                  1151     C  MENAB_PER_FRAME EQU     350                     ; NUM OF ENABLES PER FRAME
                        1152     C
= 0043                  1153     C  TIM_CTL         EQU     043H                    ; 8253 TIMER CONTROL PORT
= 0040                  1154     C  TIMER0          EQU     040H                    ; 8253 TIMER/CNTER 0 PORT
                        1155     C
0317                    1156     C  POD14   PROC    NEAR
0317  83 EC 0A          1157     C          SUB     SP,0AH                          ; RESERVE 5 WORDS ON STACK
031A  8B EC             1158     C          MOV     BP,SP                           ; INIT SCRATCH PAD POINTER
                        1159     C
                        1160     C          ASSUME  DS:ABS0,ES:ABS0
031C  E8 0CFE R         1161     C          CALL    DDS
031F  B0 30             1162     C          MOV     AL,00110000B                    ; SET TIMER 0 TO MODE 0
                        1163     C
0321  E6 43             1164     C          OUT     TIM_CTL,AL
0323  B0 00             1165     C          MOV     AL,00H
0325  E6 40             1166     C          OUT     TIMER0,AL                       ; SEND FIRST BYTE TO TIMER
0327  F6 06 0487 R 02   1167     C          TEST    INFO,2
032C  74 1F             1168     C          JZ      COLOR_EGA_V
032E  E8 0168 R         1169     C          CALL    ENV_3                           ; SET UP IN MONOCHROME
0331  C7 46 02 015E     1170     C          MOV     WORD PTR[BP][2],MENAB_PER_FRAME ; NUM.OF FRAMES FOR MONO
0336  C7 46 04 8D99     1171     C          MOV     WORD PTR[BP][4],MAX_VERT_MONO   ; MAX TIME FOR VERT/VERT
033B  C7 46 06 B862     1172     C          MOV     WORD PTR[BP][6],MIN_VERT_MONO   ; MIN TIME FOR VERT/VERT
0340  B2 B4             1173     C          MOV     DL,CRTC_ADDR_B                  ; MONO CRTC REG
0342  B4 01             1174     C          MOV     AH,C_HRZ_DSP                    ; HORIZ. TOTAL DIPLAY
0344  B0 27             1175     C          MOV     AL,27H                          ;  TO 40 COL
0346  E8 0D15 R         1176     C          CALL    OUT_DX                          ;
0349  B2 BA             1177     C          MOV     DL,INPUT_STATUS_B               ; 3BA
034B  EB 2A             1178     C          JMP     SHORT   COMMON
034D                    1179     C  COLOR_EGA_V:
034D  E8 0148 R         1180     C          CALL    ENV_X                           ; SET UP IN 40X25 COLOR
0350  E8 0E9A R         1181     C          CALL    BRST_DET                        ; ENHANCED MODE
0353  73 11             1182     C          JNC     COLOR_V                         ; NO,40X25
0355  B2 D4             1183     C          MOV     DL,CRTC_ADDR                    ; BRST MODE ONLY!
0357  B4 01             1184     C          MOV     AH,1                            ; HRZ DSP END
0359  B0 14             1185     C          MOV     AL,20                           ; MODIFY FOR TEST ONLY
035B  E8 0D15 R         1186     C          CALL    OUT_DX
035E  C7 46 02 015E     1187     C          MOV     WORD PTR[BP][2],EENAB_PER_FRAME ; NUM.OF FRAMES FOR COLOR
0363  EB 06 90          1188     C          JMP     BRST_COLOR_V
0366                    1189     C  COLOR_V:
                        1190     C
0366  C7 46 02 00C8     1191     C          MOV     WORD PTR[BP][2],CENAB_PER_FRAME ; NUM.OF FRAMES FOR COLOR
036B                    1192     C  BRST_COLOR_V:
036B  C7 46 04 A0AC     1193     C          MOV     WORD PTR[BP][4],MAX_VERT_COLOR  ; MAX TIME FOR VERT/VERT
0370  C7 46 06 C460     1194     C          MOV     WORD PTR[BP][6],MIN_VERT_COLOR  ; MIN TIME FOR VERT/VERT
0375  B2 DA             1195     C          MOV     DL,INPUT_STATUS                 ; SET ADDRESSING TO VIDEO
                        1196     C                                                  ;  ATTR STATUS
0377                    1197     C  COMMON:
0377  B8 0500           1198     C          MOV     AX,0500H                        ; SET TO VIDEO PAGE 0
037A  CD 10             1199     C          INT     10H
037C  2B C9             1200     C          SUB     CX,CX
                        1201     C
                        1202     C  ;----- LOOK FOR VERTICAL
                        1203     C
037E                    1204     C  POD14_1:
037E  EC                1205     C          IN      AL,DX                           ; GET STATUS
037F  A8 08             1206     C          TEST    AL,00001000B                    ; VERTICAL THERE YET?
0381  75 07             1207     C          JNE     POD14_2                         ; CONTINUE IF IT IS
0383  E2 F9             1208     C          LOOP    POD14_1                         ; KEEP LOOKING TILL COUNT
0385  B3 00             1209     C          MOV     BL,00                           ;  EXHAUSTED
0387  E9 0448 R         1210     C          JMP     POD14_ERR                       ; NO VERTICAL
                        1211     C
                        1212     C  ;----- GOT VERTICAL - START TIMER
                        1213     C
038A                    1214     C  POD14_2:
038A  B0 00             1215     C          MOV     AL,0
038C  E6 40             1216     C          OUT     TIMER0,AL                       ; SEND 2ND BYTE TO TIMER TO
                        1217     C                                                  ; START IT
038E  2B DB             1218     C          SUB     BX,BX                           ; INIT. ENABLE COUNTER
                        1219     C  ;----- WAIT FOR VERTICAL TO GO AWAY
0390  33 C9             1220     C          XOR     CX,CX
0392                    1221     C  POD14_25:
0392  EC                1222     C          IN      AL,DX                           ; GET STATUS
0393  A8 08             1223     C          TEST    AL,00001000B                    ; VERTICAL STILL THERE
0395  74 07             1224     C          JZ      POD14_3                         ; CONTINUE IF IT'S GONE
0397  E2 F9             1225     C          LOOP    POD14_25                        ; KEEP LOOKING TILL COUNT
0399  B3 01             1226     C          MOV     BL,01H                          ;  EXHAUSTED
039B  E9 0448 R         1227     C          JMP     POD14_ERR                       ; VERTICAL STUCK ON
                        1228     C
                        1229     C  ;----- NOW START LOOKING FOR ENABLE TRANSITIONS
                        1230     C
039E                    1231     C  POD14_3:
039E  2B C9             1232     C          SUB     CX,CX
03A0                    1233     C  POD14_4:
03A0  EC                1234     C          IN      AL,DX                           ; GET STATUS
03A1  A8 01             1235     C          TEST    AL,00000001B                    ; ENABLE ON YET?
03A3  74 15             1236     C          JE      POD14_5                         ; GO ON IF IT IS
03A5  A8 08             1237     C          TEST    AL,00001000B                    ; VERTICAL ON AGAIN?
03A7  75 23             1238     C          JNE     POD14_75                        ; CONTINUE IF IT IS
03A9  E2 F5             1239     C          LOOP    POD14_4                         ; KEEP LOOKING IF NOT
03AB  B3 02             1240     C          MOV     BL,02H
03AD  E9 0448 R         1241     C          JMP     POD14_ERR                       ; ENABLE STUCK OFF
03B0                    1242     C  POD14_4A:
03B0  B3 03             1243     C          MOV     BL,03H
03B2  E9 0448 R         1244     C          JMP     POD14_ERR                       ; VERTICAL STUCK ON
03B5                    1245     C  POD14_4B:
03B5  B3 04             1246     C          MOV     BL,04H
03B7  E9 0448 R         1247     C          JMP     POD14_ERR                       ; ENABLE STUCK ON
                        1248     C
                        1249     C  ;----- MAKE SURE VERTICAL WENT OFF WITH ENABLE GOING ON
                        1250     C
03BA                    1251     C  POD14_5:
03BA  A8 08             1252     C          TEST    AL,00001000B                    ; VERTICAL OFF?
03BC  75 F2             1253     C          JNZ     POD14_4A                        ; GO ON IF IT IS
                        1254     C  ;----- NOW WAIT FOR ENABLE TO GO OFF             ;  (ERROR IF NOT)
03BE                    1255     C  POD14_6:
03BE  EC                1256     C          IN      AL,DX                           ; GET STATUS
03BF  A8 01             1257     C          TEST    AL,00000001B                    ; ENABLE OFF YET?
03C1  E1 FB             1258     C          LOOPE   POD14_6                         ; KEEP LOOKING IF NOT
03C3  E3 F0             1259     C          JCXZ    POD14_4B                        ;  YET LOW
                        1260     C  ;----- ENABLE HAS TOGGLED, BUMP COUNTER AND TEST FOR NEXT VERTICAL
```

```
03C5                    1261    C  POD14_7:
03C5  43                1262    C          INC     BX                              ; BUMP ENABLE COUNTER
03C6  74 04             1263    C          JZ      POD14_75                        ; IF COUNTER WRAPS,
                        1264    C                                                  ;  SOMETHING IS WRONG
03C8  A8 08             1265    C          TEST    AL,00001000B                    ; DID ENABLE GO LOW
                        1266    C                                                  ;  BECAUSE OF VERTICAL
03CA  74 D2             1267    C          JZ      POD14_3                         ; IF NOT, LOOK FOR ANOTHER
                        1268    C                                                  ;  ENABLE TOGGLE
                        1269    C  ;----- HAVE HAD COMPLETE VERTICAL-VERTICAL CYCLE, NOW TEST RESULTS
03CC                    1270    C  POD14_75:
03CC  B0 00             1271    C          MOV     AL,00                           ; LATCH TIMER0
03CE  E6 43             1272    C          OUT     TIM_CTL,AL
03D0  3B 5E 02          1273    C          CMP     BX,WORD PTR[BP][2]              ; NUMBER OF ENABLES BETWEEN
                        1274    C                                                  ;  VERTICALS O.K.?
03D3  74 04             1275    C          JE      POD14_8
03D5  B3 05             1276    C          MOV     BL,05H
03D7  EB 6F             1277    C          JMP     SHORT POD14_ERR
03D9                    1278    C  POD14_8:
03D9  E4 40             1279    C          IN      AL,TIMER0                       ; GET TIMER VALUE LOW
03DB  8A E0             1280    C          MOV     AH,AL                           ; SAVE IT
03DD  90                1281    C          NOP
03DE  E4 40             1282    C          IN      AL,TIMER0                       ; GET TIMER HIGH
03E0  86 E0             1283    C          XCHG    AH,AL
03E2  90                1284    C          NOP
03E3  90                1285    C          NOP
03E4  3B 46 04          1286    C          CMP     AX,WORD PTR[BP][4]              ; MAXIMUM VERTICAL TIMING
03E7  7D 04             1287    C          JGE     POD14_9
03E9  B3 06             1288    C          MOV     BL,06H
03EB  EB 5B             1289    C          JMP     SHORT POD14_ERR
03ED                    1290    C  POD14_9:
03ED  3B 46 06          1291    C          CMP     AX,WORD PTR[BP][6]              ; MINIMUM VERTICAL TIMING
03F0  7E 04             1292    C          JLE     POD14_10
03F2  B3 07             1293    C          MOV     BL,07H
03F4  EB 52             1294    C          JMP     SHORT   POD14_ERR
                        1295    C
                        1296    C  ;----- SEE IF RED, GREEN, BLUE AND INTENSIFY DOTS WORK
                        1297    C
                        1298    C  ;----- FIRST, SET A LINE OF REVERSE VIDEO, INTENSIFIED BLANKS INTO BUFFER
03F6                    1299    C  POD14_10:
03F6  B8 09DB           1300    C          MOV     AX,09DBH                        ; WRITE CHARS, BLANKS
03F9  BB 000F           1301    C          MOV     BX,000FH                        ; PAGE 0, REVERSE VIDEO,
                        1302    C                                                  ;  HIGH INTENSITY
03FC  B9 0050           1303    C          MOV     CX,80                           ; 80 CHARACTERS
03FF  CD 10             1304    C          INT     10H
0401  EC                1305    C          IN      AL,DX
0402  52                1306    C          PUSH    DX                              ; SAVE INPUT STATUS
0403  B2 C0             1307    C          MOV     DL,ATTR_WRITE                   ; ATTRIBUTE ADDRESSS
0405  B4 0F             1308    C          MOV     AH,0FH                          ; PALETTE REG 'F'
0407  B0 3F             1309    C          MOV     AL,03FH                         ; TEST VALUE
0409  E8 0D15 R          1310    C          CALL    OUT_DX                          ; VIDEO STATUS MUX
040C  B8 000F           1311    C          MOV     AX,0FH                          ; START WITH BLUE DOTS
040F  5A                1312    C          POP     DX
0410                    1313    C  POD14_13:
0410  50                1314    C          PUSH    AX                              ; SAVE
0411  52                1315    C          PUSH    DX                              ; SAVE INPUT STATUS
0412  B2 C0             1316    C          MOV     DL,ATTR_WRITE                   ; ATTRIBUTE ADDRESSS
0414  B4 32             1317    C          MOV     AH,32H                          ; COLOR PLANE ENABLE
0416  E8 0D15 R          1318    C          CALL    OUT_DX                          ; VIDEO STATUS MUX
0419  5A                1319    C          POP     DX                              ; RECOVER INPUT STATUS
041A  58                1320    C          POP     AX
041B  2B C9             1321    C          SUB     CX,CX
                        1322    C  ;----- SEE IF DOT COMES ON
041D                    1323    C  POD14_14:
041D  EC                1324    C          IN      AL,DX                           ; GET STATUS
041E  A8 30             1325    C          TEST    AL,00110000B                    ; DOT THERE?
0420  75 09             1326    C          JNZ     POD14_15                        ; LOOK FOR DOT TO TURN OFF
0422  E2 F9             1327    C          LOOP    POD14_14                        ; CONTINUE TEST FOR DOT ON
0424  B3 10             1328    C          MOV     BL,10H
0426  0A DC             1329    C          OR      BL,AH                           ; OR IN DOT BEING TESTED
0428  EB 1E 90          1330    C          JMP     POD14_ERR                       ; DOT NOT COMING ON
                        1331    C  ;----- SEE IF DOT GOES OFF
042B                    1332    C  POD14_15:
042B  2B C9             1333    C          SUB     CX,CX
042D                    1334    C  POD14_16:
042D  EC                1335    C          IN      AL,DX                           ; GET STATUS
042E  A8 30             1336    C          TEST    AL,00110000B                    ; IS DOT STILL ON?
0430  74 08             1337    C          JE      POD14_17                        ; GO ON IF DOT OFF
0432  E2 F9             1338    C          LOOP    POD14_16                        ; ELSE, KEEP WAITING FOR
                        1339    C                                                  ;  DOT TO GO OFF
0434  B3 20             1340    C          MOV     BL,20H
0436  0A DC             1341    C          OR      BL,AH                           ; OR IN DOT BEING TESTED
0438  EB 0E             1342    C          JMP     SHORT POD14_ERR
                        1343    C
                        1344    C  ;----- ADJUST TO POINT TO NEXT DOT
                        1345    C
043A                    1346    C  POD14_17:
043A  FE C4             1347    C          INC     AH
043C  80 FC 30          1348    C          CMP     AH,030H                         ; ALL 3 DOTS DONE?
043F  74 25             1349    C          JE      POD14_18                        ; GO END
0441  80 CC 0F          1350    C          OR      AH,0FH                          ; MAKE 0F,1F,2F
0444  8A C4             1351    C          MOV     AL,AH
0446  EB C8             1352    C          JMP     POD14_13                        ; GO LOOK FOR ANOTHER DOT
0448                    1353    C  POD14_ERR:
0448  B9 0006           1354    C          MOV     CX,6
044B  BA 0103           1355    C          MOV     DX,0103H                        ; ONE LONG AND THREE SHORT
044E  E8 06C8 R         1356    C          CALL    ERR_BEEP
0451  83 C4 0A          1357    C          ADD     SP,0AH                          ; BALANCE STACK
0454  B0 36             1358    C          MOV     AL,00110110B                    ; RE-INIT TIMER 0
0456  E6 43             1359    C          OUT     TIM_CTL,AL
0458  2A C0             1360    C          SUB     AL,AL
045A  E6 40             1361    C          OUT     TIMER0,AL
045C  90                1362    C          NOP
045D  90                1363    C          NOP
045E  E6 40             1364    C          OUT     TIMER0,AL
0460  BD 0001           1365    C          MOV     BP,1
0463  E9 0091 R         1366    C          JMP     SKIP
                        1367    C          ASSUME  DS:ABS0
0466                    1368    C  POD14_18:
0466  E8 0CFE R         1369    C          CALL    DDS
0469  B8 0500           1370    C          MOV     AX,0500H                        ; SET TO VIDEO PAGE 0
046C  CD 10             1371    C          INT     10H
046E  B0 36             1372    C          MOV     AL,00110110B                    ; RE-INIT TIMER 0
0470  E6 43             1373    C          OUT     TIM_CTL,AL
0472  2A C0             1374    C          SUB     AL,AL
0474  E6 40             1375    C          OUT     TIMER0,AL
0476  90                1376    C          NOP
0477  90                1377    C          NOP
0478  E6 40             1378    C          OUT     TIMER0,AL
047A  83 C4 0A          1379    C          ADD     SP,0AH                          ; REMOVE SCRATCH PAD
047D  BD 0000           1380    C          MOV     BP,0                            ; MAKE BP NON ZERO
0480                    1381    C  POD14           ENDP
                        1382    C
                        1383    C  ;----- TEST STORAGE
                        1384    C
0480                    1385    C  MEM_TEST:
0480  1E                1386    C          PUSH    DS
```

```
0481  E8 0CFE R             1387  C          CALL    DDS
                            1388  C          ASSUME DS:ABS0
0484  F6 06 0487 R 02       1389  C          TEST    INFO,2
0489  74 12                 1390  C          JZ      D_COLOR_M
048B  80 0E 0410 R 30       1391  C          OR      EQUIP_LOW,030H
0490  B8 000F               1392  C          MOV     AX,0FH
0493  80 0E 0487 R 60       1393  C          OR      INFO,060H
0498  B8 000F               1394  C          MOV     AX,0FH
049B  EB 0D                 1395  C          JMP     SHORT D_OUT_M
049D                        1396  C  D_COLOR_M:
049D  80 26 0410 R CF       1397  C          AND     EQUIP_LOW,0CFH
04A2  80 0E 0410 R 20       1398  C          OR      EQUIP_LOW,020H
04A7  B8 000E               1399  C          MOV     AX,0EH                  ; INTERNAL COLOR MODE
04AA                        1400  C  D_OUT_M:                                ; TEST IN COLOR
04AA  CD 10                 1401  C          INT     10H
04AC  83 EC 06              1402  C          SUB     SP,6                    ; RESERVE 3 WORDS ON STACK
04AF  8B EC                 1403  C          MOV     BP,SP                   ; SET BP
04B1  B8 A000               1404  C          MOV     AX,0A000H               ; PUT BUFFER ADDRESS IN AX
                            1405  C          ASSUME  DS:NOTHING,ES:NOTHING
04B4  8E D8                 1406  C          MOV     DS,AX                   ; SET UP SEG REGS TO POINT
04B6  8E C0                 1407  C          MOV     ES,AX                   ;  TO BUFFER AREA
04B8  C7 46 02 0000         1408  C          MOV     WORD PTR[BP][2],0       ; INITIALIZE
04BD  C7 46 04 0000         1409  C          MOV     WORD PTR[BP][4],0       ; INITIALIZE
04C2  B6 03                 1410  C          MOV     DH,3
04C4  B2 C4                 1411  C          MOV     DL,SEQ_ADDR
04C6  B8 0201               1412  C          MOV     AX,0201H
04C9  E8 0D15 R             1413  C          CALL    OUT_DX
04CC  B2 CE                 1414  C          MOV     DL,GRAPH_ADDR           ; ADDRESS READ MAP SELECT
04CE  B8 0400               1415  C          MOV     AX,0400H
04D1  E8 0D15 R             1416  C          CALL    OUT_DX
04D4  52                    1417  C          PUSH    DX
04D5  B2 DA                 1418  C          MOV     DL,ATTR_READ            ; SET UP ATTRIBUTE
04D7  EC                    1419  C          IN      AL,DX
04D8  B2 C0                 1420  C          MOV     DL,ATTR_WRITE           ; ATTRIBUTE WRITE ADDRESS
04DA  B8 3200               1421  C          MOV     AX,3200H
04DD  E8 0D15 R             1422  C          CALL    OUT_DX
04E0  E8 068F R             1423  C          CALL    HOW_BIG                 ; GO FIND AMOUNT OF MEMORY
04E3  80 FC 00              1424  C          CMP     AH,0
04E6  74 03                 1425  C          JZ      AA1
04E8  E9 05CD R             1426  C          JMP     EGA_MEM_ERROR
04EB                        1427  C  AA1:
04EB  E8 05D9 R             1428  C          CALL    MEMORY_OK               ; GO TEST IT
04EE  80 FC 00              1429  C          CMP     AH,0
04F1  74 03                 1430  C          JZ      AA2
04F3  E9 05CD R             1431  C          JMP     EGA_MEM_ERROR
04F6                        1432  C  AA2:
04F6  5A                    1433  C          POP     DX
04F7  B2 C4                 1434  C          MOV     DL,SEQ_ADDR
04F9  B8 0202               1435  C          MOV     AX,0202H
04FC  E8 0D15 R             1436  C          CALL    OUT_DX
04FF  B2 CE                 1437  C          MOV     DL,GRAPH_ADDR           ; ADDRESS OF READ MAP
0501  B8 0401               1438  C          MOV     AX,0401H
0504  E8 0D15 R             1439  C          CALL    OUT_DX
0507  52                    1440  C          PUSH    DX
0508  B2 DA                 1441  C          MOV     DL,ATTR_READ            ; SET UP ATTRIBUTE
050A  EC                    1442  C          IN      AL,DX
050B  B2 C0                 1443  C          MOV     DL,ATTR_WRITE           ; ATTRIBUTE WRITE ADDRESS
050D  B8 3200               1444  C          MOV     AX,3200H
0510  E8 0D15 R             1445  C          CALL    OUT_DX
0513  C7 46 04 0000         1446  C          MOV     WORD PTR [BP][4],0      ; INITIALIZE
0518  E8 068F R             1447  C          CALL    HOW_BIG                 ; GO FIND AMOUNT OF MEMORY
051B  80 FC 00              1448  C          CMP     AH,0
051E  74 03                 1449  C          JZ      AA3
0520  E9 05CD R             1450  C          JMP     EGA_MEM_ERROR
0523                        1451  C  AA3:
0523  E8 05D9 R             1452  C          CALL    MEMORY_OK               ; GO TEST IT
0526  80 FC 00              1453  C          CMP     AH,0
0529  74 03                 1454  C          JZ      AA4
052B  E9 05CD R             1455  C          JMP     EGA_MEM_ERROR
052E                        1456  C  AA4:
052E  5A                    1457  C          POP     DX
052F  B2 C4                 1458  C          MOV     DL,SEQ_ADDR
0531  B8 0204               1459  C          MOV     AX,0204H
0534  E8 0D15 R             1460  C          CALL    OUT_DX
0537  52                    1461  C          PUSH    DX
0538  B2 CE                 1462  C          MOV     DL,GRAPH_ADDR           ; ADDRESS OF READ MAP
053A  B8 0402               1463  C          MOV     AX,0402H
053D  E8 0D15 R             1464  C          CALL    OUT_DX
0540  B2 DA                 1465  C          MOV     DL,ATTR_READ            ; SET UP ATTRIBUTE
0542  EC                    1466  C          IN      AL,DX
0543  B2 C0                 1467  C          MOV     DL,ATTR_WRITE           ; ATTRIBUTE WRITE ADDRESS
0545  B8 3200               1468  C          MOV     AX,3200H
0548  E8 0D15 R             1469  C          CALL    OUT_DX
054B  C7 46 04 0000         1470  C          MOV     WORD PTR[BP][4],0       ; INITIALIZE
0550  E8 068F R             1471  C          CALL    HOW_BIG                 ; GO FIND AMOUNT OF MEMORY
0553  80 FC 00              1472  C          CMP     AH,0
0556  74 03                 1473  C          JZ      AA5
0558  EB 73 90              1474  C          JMP     EGA_MEM_ERROR
055B                        1475  C  AA5:
055B  E8 05D9 R             1476  C          CALL    MEMORY_OK               ; GO TEST IT
055E  80 FC 00              1477  C          CMP     AH,0
0561  74 03                 1478  C          JZ      AA6
0563  EB 68 90              1479  C          JMP     EGA_MEM_ERROR
0566                        1480  C  AA6:
0566  5A                    1481  C          POP     DX
0567  B2 C4                 1482  C          MOV     DL,SEQ_ADDR
0569  B8 0208               1483  C          MOV     AX,0208H
056C  E8 0D15 R             1484  C          CALL    OUT_DX
056F  B2 CE                 1485  C          MOV     DL,GRAPH_ADDR           ; ADDRESS OF READ MAP
0571  B8 0403               1486  C          MOV     AX,0403H
0574  E8 0D15 R             1487  C          CALL    OUT_DX
0577  52                    1488  C          PUSH    DX
0578  B2 DA                 1489  C          MOV     DL,ATTR_READ            ; SET UP ATTRIBUTE
057A  EC                    1490  C          IN      AL,DX
057B  B2 C0                 1491  C          MOV     DL,ATTR_WRITE           ; ATTRIBUTE WRITE ADDRESS
057D  B8 3200               1492  C          MOV     AX,3200H
0580  E8 0D15 R             1493  C          CALL    OUT_DX
0583  C7 46 04 0000         1494  C          MOV     WORD PTR[BP][4],0       ; INITIALIZE
0588  E8 068F R             1495  C          CALL    HOW_BIG                 ; GO FIND AMOUNT OF MEMORY
058B  80 FC 00              1496  C          CMP     AH,0
058E  75 3D                 1497  C          JNZ     EGA_MEM_ERROR
0590  E8 05D9 R             1498  C          CALL    MEMORY_OK               ; GO TEST IT
0593  80 FC 00              1499  C          CMP     AH,0
0596  75 35                 1500  C          JNZ     EGA_MEM_ERROR
0598  55                    1501  C          PUSH    BP                      ; SAVE SCRATCH PAD POINTER
0599  BD 0000               1502  C          MOV     BP,0                    ; RESET BP FOR XT
059C                        1503  C  EGA_MEM_EXIT:
059C  5E                    1504  C          POP     SI                      ; RESTORE
059D  5A                    1505  C          POP     DX
059E  E8 0CFE R             1506  C          CALL    DDS                     ; SET DATA SEGMENT
                            1507  C          ASSUME  DS:ABS0
05A1  36: 8B 5C 02          1508  C          MOV     BX,WORD PTR SS:[SI][2]  ; GET EGA MEMORY SIZE
05A5  B1 06                 1509  C          MOV     CL,06H                  ; DIVIDE BY 64 TO GET
05A7  D3 EB                 1510  C          SHR     BX,CL                   ;  NUMBER OF 64KB BLOCKS
05A9  4B                    1511  C          DEC     BX
05AA  B1 05                 1512  C          MOV     CL,05H
```

```
05AC  D3 E3                1513    C           SHL     BX,CL
05AE  80 E3 60             1514    C           AND     BL,01100000B                ; ISOLATE BITS 5 AND 6
                           1515    C
05B1  80 26 0487 R 9F      1516    C           AND     INFO,10011111B
                           1517    C
05B6  08 1E 0487 R         1518    C           OR      INFO,BL
                           1519    C
05BA  80 0E 0487 R 04      1520    C           OR      INFO,00000100B              ; 04H  SET 3XX ACTIVE
05BF  8A 1E 0488 R         1521    C           MOV     BL,INFO_3
05C3  E8 00F3 R            1522    C           CALL    MK_ENV
05C6  83 C4 06             1523    C           ADD     SP,6                        ; RESTORE STACK
05C9  1F                   1524    C           POP     DS
05CA  E9 0091 R            1525    C           JMP     SKIP                        ; GO TO END
05CD                       1526    C   EGA_MEM_ERROR:
05CD  BA 0103              1527    C           MOV     DX,0103H                    ; ONE LONG AND THREE SHORT
05D0  E8 06C8 R            1528    C           CALL    ERR_BEEP
05D3  55                   1529    C           PUSH    BP                          ; SAVE SCRATCH PAD POINTER
05D4  BD 0001              1530    C           MOV     BP,1                        ; INDICATE ERROR FOR XT
05D7  EB C3                1531    C           JMP     EGA_MEM_EXIT
                           1532    C
                           1533    C   ;----- THIS ROUTINE FINDS AMOUNT OF MEMORY GOOD
                           1534    C
05D9                       1535    C   MEMORY_OK       PROC    NEAR
05D9  BB A000              1536    C           MOV     BX,0A000H                   ; SET PTR. TO BUFFER SEG
05DC  8E DB                1537    C           MOV     DS,BX                       ; SET SEG.REG.
05DE  8E C3                1538    C           MOV     ES,BX
05E0  8B 46 04             1539    C           MOV     AX,WORD PTR[BP][4]          ; SET COUNT FOR 32K WORDS
05E3  8A E8                1540    C           MOV     CH,AL                       ; SET AMOUNT OF BUFFER
05E5  2A C9                1541    C           SUB     CL,CL                       ;  TO BE TESTED
05E7  D1 E1                1542    C           SHL     CX,1                        ; MULTIPLY BY TWO
05E9  E8 05FB R            1543    C           CALL    PODSTG
05EC  80 FC 00             1544    C           CMP     AH,0                        ; TEST FOR ERROR
05EF  75 09                1545    C           JNZ     MEMORY_OK_ERR               ; IF ERROR GO PRINT IT
05F1                       1546    C   MEMORY_OK_EX:
05F1  8B 46 04             1547    C           MOV     AX,WORD PTR[BP][4]          ; AMOUNT OF MEMORY FOUND
05F4  01 46 02             1548    C           ADD     WORD PTR[BP][2],AX          ; AMOUNT OF MEMORY GOOD
05F7  B8 0000              1549    C           MOV     AX,0
05FA                       1550    C   MEMORY_OK_ERR:
05FA  C3                   1551    C           RET
05FB                       1552    C   MEMORY_OK       ENDP
                           1553    C
                           1554    C   ;--------------------------------------------------------------
                           1555    C   ; THIS ROUTINE PERFORMS A READ/WRITE TEST ON A BLOCK OF STORAGE :
                           1556    C   ; (MAX. SIZE = 32KW). IF "WARM START", FILL BLOCK WITH 0000 AND :
                           1557    C   ; RETURN.                                                       :
                           1558    C   ; ON ENTRY:                                                    :
                           1559    C   ;   ES = ADDRESS OF STORAGE TO BE TESTED                       :
                           1560    C   ;   DS = ADDRESS OF STORAGE TO BE TESTED                       :
                           1561    C   ;   CX = WORD COUNT OF STORAGE BLOCK TO BE TESTED              :
                           1562    C   ;        (MAX. = 8000H (32K WORDS))                           :
                           1563    C   ; ON EXIT:                                                     :
                           1564    C   ;   ZERO FLAG = OFF IF STORAGE ERROR                          :
                           1565    C   ; AX,BX,CX,DX,DI,SI ARE ALL DESTROYED.                        :
                           1566    C   ;--------------------------------------------------------------
05FB                       1567    C   PODSTG   PROC    NEAR
05FB  55                   1568    C           PUSH    BP
05FC  FC                   1569    C           CLD                                 ; SET DIR TO INCREMENT
05FD  2B FF                1570    C           SUB     DI,DI                       ; SET DI=0000 REL TO START
                           1571    C                                               ;  OF SEGMENT
05FF  2B C0                1572    C           SUB     AX,AX                       ; INITIAL DATA PATTERN FOR
                           1573    C                                               ;  00-FF TEST
0601  E8 0CFE R            1574    C           CALL    DDS
                           1575    C           ASSUME  DS:ABS0
0604  8B 1E 0472 R         1576    C           MOV     BX,DS:RESET_FLAG            ; WARM START?
0608  81 FB 1234           1577    C           CMP     BX,1234H
060C  8C C2                1578    C           MOV     DX,ES
060E  8E DA                1579    C           MOV     DS,DX                       ; RESTORE DS
0610  74 62                1580    C           JE      PODSTG_5                    ; GO DO FILL WITH 0000
                           1581    C                                               ;  IF WARM START
0612  81 FB 4321           1582    C           CMP     BX,4321H                    ; DCP WARM START?
0616  74 5C                1583    C           JE      PODSTG_5                    ; DO FILL IF SO
0618                       1584    C   PODSTG_1:
0618  88 05                1585    C           MOV     [DI],AL                     ; WRITE TEST DATA
061A  8A 05                1586    C           MOV     AL,[DI]                     ; GET IT BACK
061C  32 C4                1587    C           XOR     AL,AH                       ; COMPARE TO EXPECTED
061E  75 40                1588    C           JNZ     PODSTG_ERRO                 ; ERROR EXIT IF MISCOMPARE
0620  FE C4                1589    C           INC     AH                          ; FORM NEW DATA PATTERN
0622  8A C4                1590    C           MOV     AL,AH
0624  75 F2                1591    C           JNZ     PODSTG_1                    ; LOOP TILL ALL 256 DATA
                           1592    C                                               ;  PATTERNS DONE
0626  8B E9                1593    C           MOV     BP,CX                       ; SAVE WORD COUNT
0628  B8 AA55              1594    C           MOV     AX,0AA55H                   ; LOAD DATA PATTERN
062B  8B D8                1595    C           MOV     BX,AX
062D  BA 55AA              1596    C           MOV     DX,055AAH                   ; LOAD OTHER DATA PATTERN
0630  F3/ AB               1597    C           REP     STOSW                       ; FILL WORDS FROM LOW TO
                           1598    C                                               ;  HIGH WITH AAAA
0632  4F                   1599    C           DEC     DI                          ; POINT TO LAST WORD
0633  4F                   1600    C           DEC     DI                          ;  WRITTEN
0634  FD                   1601    C           STD                                 ; SET DIR FLAG TO GO DOWN
0635  8B F7                1602    C           MOV     SI,DI                       ; SET INDEX REGS. EQUAL
0637  8B CD                1603    C           MOV     CX,BP                       ; RECOVER WORD COUNT
0639                       1604    C   PODSTG_2:                                  ; GO FROM HIGH TO LOW
0639  AD                   1605    C           LODSW                               ; GET WORD FROM MEMORY
063A  33 C3                1606    C           XOR     AX,BX                       ; EQUAL WHAT S/B THERE?
063C  75 22                1607    C           JNZ     PODSTG_ERRO                 ; GO ERROR EXIT IF NOT
063E  8B C2                1608    C           MOV     AX,DX                       ; GET 55 DATA PATTERN AND
0640  AB                   1609    C           STOSW                               ;  STORE IN LOC JUST READ
0641  E2 F6                1610    C           LOOP    PODSTG_2                    ; LOOP TILL ALL BYTES DONE
0643  8B CD                1611    C           MOV     CX,BP                       ; RECOVER WORD COUNT
0645  FC                   1612    C           CLD                                 ; BACK TO INCREMENT
0646  46                   1613    C           INC     SI                          ; ADJUST PTRS
0647  46                   1614    C           INC     SI
0648  8B FE                1615    C           MOV     DI,SI
064A                       1616    C   PODSTG_3:                                  ; LOW TO HIGH DOING WORDS
064A  AD                   1617    C           LODSW                               ; GET A WORD
064B  33 C2                1618    C           XOR     AX,DX                       ; SHOULD COMPARE TO DX
064D  75 11                1619    C           JNZ     PODSTG_ERRO                 ; GO ERROR IF NOT
064F  AB                   1620    C           STOSW                               ; WRITE 0000 BACK TO LOC
                           1621    C                                               ;  JUST READ
0650  E2 F8                1622    C           LOOP    PODSTG_3                    ; LOOP TILL DONE
                           1623    C
0652  FD                   1624    C           STD                                 ; BACK TO DECREMENT
0653  4E                   1625    C           DEC     SI                          ; ADJUST POINTER DOWN TO
                           1626    C                                               ;  LAST WORD WRITTEN
0654  4E                   1627    C           DEC     SI
0655  8B CD                1628    C           MOV     CX,BP                       ; GET WORD COUNT
0657                       1629    C   PODSTG_4:
0657  AD                   1630    C           LODSW                               ; GET WORD
0658  0B C0                1631    C           OR      AX,AX                       ; = TO 0000
065A  75 04                1632    C           JNZ     PODSTG_ERRO                 ; ERROR IF NOT
065C  E2 F9                1633    C           LOOP    PODSTG_4                    ; LOOP TILL DONE
065E  EB 11                1634    C           JMP     SHORT PODSTG_ERR2
0660                       1635    C   PODSTG_ERRO:
0660  8B C8                1636    C           MOV     CX,AX                       ; SAVE BITS IN ERROR
0662  32 E4                1637    C           XOR     AH,AH
0664  0A ED                1638    C           OR      CH,CH                       ; HIGH BYTE ERROR?
```

```
0666  74 02                1639    C           JZ      PODSTG_ERR1
0668  B4 01                1640    C           MOV     AH,1                    ; SET HIGH BYTE ERROR
066A                       1641    C  PODSTG_ERR1:
066A  0A C9                1642    C           OR      CL,CL                   ; LOW BYTE ERROR?
066C  74 03                1643    C           JZ      PODSTG_ERR2
066E  80 C4 02             1644    C           ADD     AH,2
0671                       1645    C  PODSTG_ERR2:
0671  5D                   1646    C           POP     BP
0672  FC                   1647    C           CLD                             ; SET DIR FLAG BACK TO INC
0673  C3                   1648    C           RET                             ; RETURN TO CALLER
0674                       1649    C  PODSTG_5:                                ; SIMPLE FILL WITH 0000 ON
                           1650    C                                           ;  WARM-START
0674  50                   1651    C           PUSH    AX                      ; SAVE
0675  52                   1652    C           PUSH    DX                      ; SAVE VALUE
0676  B6 03                1653    C           MOV     DH,3
0678  B2 C4                1654    C           MOV     DL,SEQ_ADDR             ; SEQ_ADDR  REGISTER
067A  B8 020F              1655    C           MOV     AX,020FH
067D  E8 0D15 R            1656    C           CALL    OUT_DX                  ; DO IT
0680  5A                   1657    C           POP     DX                      ; RESTORE
0681  58                   1658    C           POP     AX                      ; RESTORE
0682  F3/ AB               1659    C           REP     STOSW
0684  E8 0CFE R            1660    C           CALL    DDS
                           1661    C           ASSUME  DS:ABS0
0687  89 1E 0472 R         1662    C           MOV     DS:RESET_FLAG,BX
068B  8E DA                1663    C           MOV     DS,DX                   ; RESTORE DS
068D  EB E2                1664    C           JMP     PODSTG_ERR2             ; AND EXIT
068F                       1665    C  PODSTG   ENDP
                           1666    C
                           1667    C  ;----- DETERMINE SIZE OF BUFFER
                           1668    C
068F                       1669    C  HOW_BIG          PROC NEAR
068F  8C DA                1670    C           MOV     DX,DS                   ; SET PNTR TO BUFFER LOC
0691  2B DB                1671    C           SUB     BX,BX                   ; BASIC COUNT OF 00K
0693                       1672    C  FILL_LOOP:
0693  8E C2                1673    C           MOV     ES,DX                   ; SET SEG. REG
0695  2B FF                1674    C           SUB     DI,DI
0697  B8 AA55              1675    C           MOV     AX,0AA55H               ; TEST PATTERN
069A  8B C8                1676    C           MOV     CX,AX
069C  26: 89 05            1677    C           MOV     ES:[DI],AX              ; SEND TO MEMORY
069F  B0 0F                1678    C           MOV     AL,0FH                  ; PUT SOMETHING IN AL
06A1  26: 8B 05            1679    C           MOV     AX,ES:[DI]              ; GET PATTERN FROM MEMORY
06A4  33 C1                1680    C           XOR     AX,CX                   ; COMPARE PATTERNS
06A6  75 14                1681    C           JNZ     HOW_BIG_END             ; GO END IF NO COMPARE
06A8  B9 2000              1682    C           MOV     CX,2000H                ; SET COUNT FOR 8K WORDS
06AB  F3/ AB               1683    C  REP      STOSW                           ; FILL 8K WORDS
06AD  81 C2 0400           1684    C           ADD     DX,0400H                ; POINT TO NEXT 16K BLOCK
06B1  83 C3 10             1685    C           ADD     BX,16                   ; BUMP COUNT BY 16KB
06B4  80 FE B0             1686    C           CMP     DH,0B0H
06B7  75 DA                1687    C           JNZ     FILL_LOOP               ; AREA YET ?(B0000H)
06B9  EB 01 90             1688    C           JMP     HOW_BIG_END
06BC                       1689    C  HOW_BIG_END:
06BC  80 FE A0             1690    C           CMP     DH,0A0H                 ; 1ST 16KB OK
06BF  74 06                1691    C           JZ      HB_ERROR_EXIT
06C1                       1692    C  RESUME:
06C1  01 5E 04             1693    C           ADD     WORD PTR[BP][4],BX      ; SAVE BUFFER FOUND
06C4  B8 0000              1694    C           MOV     AX,0
06C7                       1695    C  HB_ERROR_EXIT:
06C7  C3                   1696    C           RET
06C8                       1697    C  HOW_BIG          ENDP
                           1698    C
                           1699    C  ;--------------------------------------
                           1700    C  ;SUBROUTINES FOR POWER ON DIAGNOSTICS  :
                           1701    C  ;-------------------------------------------------------------------------
                           1702    C  ;THIS PROCEDURE WILL ISSUE ONE LONG TONE (3 SEC) AND ONE OR      :
                           1703    C  ;MORE SHORT TONES (1 SEC) TO INDICATE A FAILURE ON THE PLANAR    :
                           1704    C  ;BOARD ,A BAD RAM MODULE,OR A PROBLEM WITH THE CRT.              :
                           1705    C  ;ENTRY REQUIREMENTS:                                             :
                           1706    C  ;       DH=NUMBER OF LONG TONES TO BEEP                          :
                           1707    C  ;       DL=NUMBER OF SHORT TONES TO BEEP.                        :
                           1708    C  ;-------------------------------------------------------------------------
06C8                       1709    C  ERR_BEEP    PROC    NEAR
06C8  9C                   1710    C           PUSHF                           ; SAVE FLAGS
06C9  FA                   1711    C           CLI                             ; DISABLE SYSTEM INTS
06CA  1E                   1712    C           PUSH    DS
06CB  E8 0CFE R            1713    C           CALL    DDS
                           1714    C           ASSUME  DS:ABS0
06CE  0A F6                1715    C           OR      DH,DH                   ; ANY LONG TONES TO BEEP
06D0  74 0B                1716    C           JZ      G3                      ; NO, DO THE SHORT ONES
06D2                       1717    C  G1:                                      ; LONG BEEP
06D2  B3 06                1718    C           MOV     BL,6                    ; COUNTER FOR BEEPS
06D4  E8 0D20 R            1719    C           CALL    BEEP                    ; DO THE BEEP
06D7                       1720    C  G2:
06D7  E2 FE                1721    C           LOOP    G2                      ; DELAY BETWEEN BEEPS
06D9  FE CE                1722    C           DEC     DH                      ; ANY MORE TO DO
06DB  75 F5                1723    C           JNZ     G1                      ; DO IT
06DD                       1724    C  G3:
06DD  B3 01                1725    C           MOV     BL,1                    ; COUNTER FOR A SHORT BEEP
06DF  E8 0D20 R            1726    C           CALL    BEEP                    ; DO IT
06E2                       1727    C  G4:
06E2  E2 FE                1728    C           LOOP    G4                      ; DELAY BETWEEN BEEPS
06E4  FE CA                1729    C           DEC     DL                      ; DONE WITH SHORT BEEPS
06E6  75 F5                1730    C           JNZ     G3                      ; DO MORE
06E8                       1731    C  G5:
06E8  E2 FE                1732    C           LOOP G5                         ; DELAY BEFORE RETURN
06EA                       1733    C  G6:
06EA  E2 FE                1734    C           LOOP G6
06EC  1F                   1735    C           POP     DS                      ; RESTORE CONTENTS OF DS
06ED  9D                   1736    C           POPF                            ; RESTORE FLAGS
06EE  C3                   1737    C           RET
06EF                       1738    C  ERR_BEEP    ENDP
                           1739    C
                           1740    C           SUBTTL
                           1741
                           1742
06EF                       1743        T2      LABEL   WORD
06EF  0EB3 R               1744                DW      OFFSET  AH0             ; MODE SET
06F1  10EF R               1745                DW      OFFSET  AH1             ; SET CURSOR TYPE
06F3  1157 R               1746                DW      OFFSET  AH2             ; SET CURSOR POSITION
06F5  1186 R               1747                DW      OFFSET  AH3             ; READ CURSOR POSITION
06F7  119D R               1748                DW      OFFSET  AH4             ; READ LIGHT PEN POSITION
06F9  12A4 R               1749                DW      OFFSET  AH5             ; ACTIVE DISPLAY PAGE
06FB  150E R               1750                DW      OFFSET  AH6             ; SCROLL DOWN
06FD  15B0 R               1751                DW      OFFSET  AH7             ; SCROLL UP
06FF  17D2 R               1752                DW      OFFSET  AH8             ; READ CHAR/ATTRIBUTE
0701  1899 R               1753                DW      OFFSET  AH9             ; WRITE CHAR/ATTRIBUTE
0703  18DD R               1754                DW      OFFSET  AHA             ; WRITE CHARACTER ONLY
0705  1A75 R               1755                DW      OFFSET  AHB             ; SET COLOR PALETTE
0707  1BCB R               1756                DW      OFFSET  AHC             ; WRITE DOT
0709  1C9F R               1757                DW      OFFSET  AHD             ; READ DOT
070B  1D01 R               1758                DW      OFFSET  AHE             ; WRITE TTY
070D  1D85 R               1759                DW      OFFSET  AHF             ; CURRENT VIDEO STATE
070F  1DC5 R               1760                DW      OFFSET  AH10            ; SET PALETTE REGISTERS
0711  1F98 R               1761                DW      OFFSET  AH11            ; CHAR GENERATOR ROUTINE
0713  20BF R               1762                DW      OFFSET  AH12            ; ALTERNATE SELECT
0715  2118 R               1763                DW      OFFSET  AH13            ; WRITE STRING
= 0028                     1764        T2L     EQU     $-T2
```

```
                                1765
                                1766      C             INCLUDE         VPARMS.INC
                                1767      C             SUBTTL  VPARMS.INC
                                1768      C             PAGE
0717                            1769      C VIDEO_PARMS       LABEL   BYTE
                                1770      C
                                1771      C ; STRUCTURE OF THIS TABLE
                                1772      C ;
                                1773      C ;       COLUMNS, ROWS, PELS PER CHARACTER
                                1774      C ;       PAGE LENGTH
                                1775      C ;       SEQUENCER PARAMETERS
                                1776      C ;       MISCELLANEOUS REGISTER
                                1777      C ;       CRTC PARAMETERS
                                1778      C ;       ATTRIBUTE PARAMETERS
                                1779      C ;       GRAPHICS PARAMETERS
                                1780      C
= 0000                          1781      C BASE_1  EQU     $ - VIDEO_PARMS
0717                            1782      C BASE_1_L        LABEL   BYTE
                                1783      C
                                1784      C ;----- DEFAULT MODES
                                1785      C
                                1786      C ;--0--
0717  28 18 08                  1787      C             DB      40D,24D,08D
071A  0800                      1788      C             DW      00800H
                                1789      C
= 0005                          1790      C TFS_LEN EQU     $ - BASE_1_L
                                1791      C
071C                            1792      C SEQ_PARMS       LABEL   BYTE
071C  0B 03 00 03               1793      C             DB      00BH,003H,000H,003H
= 0004                          1794      C M1      EQU     $ - SEQ_PARMS
                                1795      C
0720  23                        1796      C             DB      023H
                                1797      C
0721                            1798      C CRT_PARMS       LABEL   BYTE
0721  37 27 2D 37 31 15         1799      C             DB      037H,027H,02DH,037H,031H,015H
0727  04 11 00 07 06 07         1800      C             DB      004H,011H,000H,007H,006H,007H
072D  00 00 00 00 E1 24         1801      C             DB      000H,000H,000H,000H,0E1H,024H
0733  C7 14 08 E0 F0 A3         1802      C             DB      0C7H,014H,008H,0E0H,0F0H,0A3H
0739  FF                        1803      C             DB      0FFH
= 0019                          1804      C M4      EQU     $-CRT_PARMS
                                1805      C
= 0023                          1806      C LN_4    EQU     $ - BASE_1_L
                                1807      C
073A                            1808      C ATTR_PARMS      LABEL   BYTE
073A  00 01 02 03 04 05         1809      C             DB      000H,001H,002H,003H,004H,005H
0740  06 07 10 11 12 13         1810      C             DB      006H,007H,010H,011H,012H,013H
0746  14 15 16 17 08 00         1811      C             DB      014H,015H,016H,017H,008H,000H
074C  0F 00                     1812      C             DB      00FH,000H
= 0014                          1813      C M5      EQU     $-ATTR_PARMS
                                1814      C
= 0037                          1815      C LN_2    EQU     $ - BASE_1_L
074E                            1816      C GRAPH_PARMS     LABEL   BYTE
074E  00 00 00 00 00 10         1817      C             DB      000H,000H,000H,000H,000H,010H
0754  0E 00 FF                  1818      C             DB      00EH,000H,0FFH
= 0009                          1819      C M6      EQU     $-GRAPH_PARMS
                                1820      C
= 0040                          1821      C M_TBL_LEN       EQU     $ - BASE_1_L
                                1822      C
                                1823      C ;--1--
0757  28 18 08                  1824      C             DB      40D,24D,08D
075A  0800                      1825      C             DW      00800H
                                1826      C
075C  0B 03 00 03               1827      C             DB      00BH,003H,000H,003H
                                1828      C
0760  23                        1829      C             DB      023H
                                1830      C
0761  37 27 2D 37 31 15         1831      C             DB      037H,027H,02DH,037H,031H,015H
0767  04 11 00 07 06 07         1832      C             DB      004H,011H,000H,007H,006H,007H
076D  00 00 00 00 E1 24         1833      C             DB      000H,000H,000H,000H,0E1H,024H
0773  C7 14 08 E0 F0 A3         1834      C             DB      0C7H,014H,008H,0E0H,0F0H,0A3H
0779  FF                        1835      C             DB      0FFH
                                1836      C
077A  00 01 02 03 04 05         1837      C             DB      000H,001H,002H,003H,004H,005H
0780  06 07 10 11 12 13         1838      C             DB      006H,007H,010H,011H,012H,013H
0786  14 15 16 17 08 00         1839      C             DB      014H,015H,016H,017H,008H,000H
078C  0F 00                     1840      C             DB      00FH,000H
                                1841      C
078E  00 00 00 00 00 10         1842      C             DB      000H,000H,000H,000H,000H,010H
0794  0E 00 FF                  1843      C             DB      00EH,000H,0FFH
                                1844      C
                                1845      C ;--2--
0797  50 18 08                  1846      C             DB      80D,24D,08D
079A  1000                      1847      C             DW      01000H
                                1848      C
079C  01 03 00 03               1849      C             DB      001H,003H,000H,003H
                                1850      C
07A0  23                        1851      C             DB      023H
                                1852      C
07A1  70 4F 5C 2F 5F 07         1853      C             DB      070H,04FH,05CH,02FH,05FH,007H
07A7  04 11 00 07 06 07         1854      C             DB      004H,011H,000H,007H,006H,007H
07AD  00 00 00 00 E1 24         1855      C             DB      000H,000H,000H,000H,0E1H,024H
07B3  C7 28 08 E0 F0 A3         1856      C             DB      0C7H,028H,008H,0E0H,0F0H,0A3H
07B9  FF                        1857      C             DB      0FFH
                                1858      C
07BA  00 01 02 03 04 05         1859      C             DB      000H,001H,002H,003H,004H,005H
07C0  06 07 10 11 12 13         1860      C             DB      006H,007H,010H,011H,012H,013H
07C6  14 15 16 17 08 00         1861      C             DB      014H,015H,016H,017H,008H,000H
07CC  0F 00                     1862      C             DB      00FH,000H
                                1863      C
07CE  00 00 00 00 00 10         1864      C             DB      000H,000H,000H,000H,000H,010H
07D4  0E 00 FF                  1865      C             DB      00EH,000H,0FFH
                                1866      C
                                1867      C ;--3--
07D7  50 18 08                  1868      C             DB      80D,24D,08D
07DA  1000                      1869      C             DW      01000H
                                1870      C
07DC  01 03 00 03               1871      C             DB      001H,003H,000H,003H
                                1872      C
07E0  23                        1873      C             DB      023H
                                1874      C
07E1  70 4F 5C 2F 5F 07         1875      C             DB      070H,04FH,05CH,02FH,05FH,007H
07E7  04 11 00 07 06 07         1876      C             DB      004H,011H,000H,007H,006H,007H
07ED  00 00 00 00 E1 24         1877      C             DB      000H,000H,000H,000H,0E1H,024H
07F3  C7 28 08 E0 F0 A3         1878      C             DB      0C7H,028H,008H,0E0H,0F0H,0A3H
07F9  FF                        1879      C             DB      0FFH
                                1880      C
07FA  00 01 02 03 04 05         1881      C             DB      000H,001H,002H,003H,004H,005H
0800  06 07 10 11 12 13         1882      C             DB      006H,007H,010H,011H,012H,013H
0806  14 15 16 17 08 00         1883      C             DB      014H,015H,016H,017H,008H,000H
080C  0F 00                     1884      C             DB      00FH,000H
                                1885      C
080E  00 00 00 00 00 10         1886      C             DB      000H,000H,000H,000H,000H,010H
0814  0E 00 FF                  1887      C             DB      00EH,000H,0FFH
                                1888      C
                                1889      C ;--4--
0817  28 18 08                  1890      C             DB      40D,24D,08D
```

```
081A  4000                        1891  C             DW      04000H
                                  1892  C
081C  0B 03 00 02                 1893  C             DB      00BH,003H,000H,002H
                                  1894  C
0820  23                         1895  C             DB      023H
                                  1896  C
0821  37 27 2D 37 30 14           1897  C             DB      037H,027H,02DH,037H,030H,014H
0827  04 11 00 01 00 00           1898  C             DB      004H,011H,000H,001H,000H,000H
082D  00 00 00 00 E1 24           1899  C             DB      000H,000H,000H,000H,0E1H,024H
0833  C7 14 00 E0 F0 A2           1900  C             DB      0C7H,014H,000H,0E0H,0F0H,0A2H
0839  FF                          1901  C             DB      0FFH
                                  1902  C
083A  00 13 15 17 02 04           1903  C             DB      000H,013H,015H,017H,002H,004H
0840  06 07 10 11 12 13           1904  C             DB      006H,007H,010H,011H,012H,013H
0846  14 15 16 17 01 00           1905  C             DB      014H,015H,016H,017H,001H,000H
084C  03 00                       1906  C             DB      003H,000H
                                  1907  C
084E  00 00 00 00 00 30           1908  C             DB      000H,000H,000H,000H,000H,030H
0854  0F 00 FF                    1909  C             DB      00FH,000H,0FFH
                                  1910  C
                                  1911  C  ;--5--
0857  28 18 08                    1912  C             DB      40D,24D,08D
085A  4000                        1913  C             DW      04000H
                                  1914  C
085C  0B 03 00 02                 1915  C             DB      00BH,003H,000H,002H
                                  1916  C
0860  23                         1917  C             DB      023H
                                  1918  C
0861  37 27 2D 37 30 14           1919  C             DB      037H,027H,02DH,037H,030H,014H
0867  04 11 00 01 00 00           1920  C             DB      004H,011H,000H,001H,000H,000H
086D  00 00 00 00 E1 24           1921  C             DB      000H,000H,000H,000H,0E1H,024H
0873  C7 14 00 E0 F0 A2           1922  C             DB      0C7H,014H,000H,0E0H,0F0H,0A2H
0879  FF                          1923  C             DB      0FFH
                                  1924  C
087A  00 13 15 17 02 04           1925  C             DB      000H,013H,015H,017H,002H,004H
0880  06 07 10 11 12 13           1926  C             DB      006H,007H,010H,011H,012H,013H
0886  14 15 16 17 01 00           1927  C             DB      014H,015H,016H,017H,001H,000H
088C  03 00                       1928  C             DB      003H,000H
                                  1929  C
088E  00 00 00 00 00 30           1930  C             DB      000H,000H,000H,000H,000H,030H
0894  0F 00 FF                    1931  C             DB      00FH,000H,0FFH
                                  1932  C
                                  1933  C  ;--6--
0897  50 18 08                    1934  C             DB      80D,24D,08D
089A  4000                        1935  C             DW      04000H
                                  1936  C
089C  01 01 00 06                 1937  C             DB      001H,001H,000H,006H
                                  1938  C
08A0  23                         1939  C             DB      023H
                                  1940  C
08A1  70 4F 59 2D 5E 06           1941  C             DB      070H,04FH,059H,02DH,05EH,006H
08A7  04 11 00 01 00 00           1942  C             DB      004H,011H,000H,001H,000H,000H
08AD  00 00 00 00 E0 23           1943  C             DB      000H,000H,000H,000H,0E0H,023H
08B3  C7 28 00 QF EF C2           1944  C             DB      0C7H,028H,000H,0DFH,0EFH,0C2H
08B9  FF                          1945  C             DB      0FFH
                                  1946  C
08BA  00 17 17 17 17 17           1947  C             DB      000H,017H,017H,017H,017H,017H
08C0  17 17 17 17 17 17           1948  C             DB      017H,017H,017H,017H,017H,017H
08C6  17 17 17 17 01 00           1949  C             DB      017H,017H,017H,017H,001H,000H
08CC  01 00                       1950  C             DB      001H,000H
                                  1951  C
08CE  00 00 00 00 00 00           1952  C             DB      000H,000H,000H,000H,000H,000H
08D4  0D 00 FF                    1953  C             DB      00DH,000H,0FFH
                                  1954  C
                                  1955  C  ;--7--
08D7  50 18 0E                    1956  C             DB      80D,24D,14D
08DA  1000                        1957  C             DW      01000H
                                  1958  C
08DC  00 03 00 03                 1959  C             DB      000H,003H,000H,003H
                                  1960  C
08E0  A6                         1961  C             DB      0A6H
                                  1962  C
08E1  60 4F 56 3A 51 60           1963  C             DB      060H,04FH,056H,03AH,051H,060H
08E7  70 1F 00 0D 0B 0C           1964  C             DB      070H,01FH,000H,00DH,00BH,00CH
08ED  00 00 00 00 5E 2E           1965  C             DB      000H,000H,000H,000H,05EH,02EH
08F3  5D 28 0D 5E 6E A3           1966  C             DB      05DH,028H,00DH,05EH,06EH,0A3H
08F9  FF                          1967  C             DB      0FFH
                                  1968  C
08FA  00 08 08 08 08 08           1969  C             DB      000H,008H,008H,008H,008H,008H
0900  08 08 10 18 18 18           1970  C             DB      008H,008H,010H,018H,018H,018H
0906  18 18 18 18 0E 00           1971  C             DB      018H,018H,018H,018H,00EH,000H
090C  0F 08                       1972  C             DB      00FH,008H
                                  1973  C
090E  00 00 00 00 00 10           1974  C             DB      000H,000H,000H,000H,000H,010H
0914  0A 00 FF                    1975  C             DB      00AH,000H,0FFH
                                  1976  C
                                  1977  C  ;--8--
0917  28 18 08                    1978  C             DB      40D,24D,08D
091A  4000                        1979  C             DW      04000H
                                  1980  C
091C  00 00 00 03                 1981  C             DB      000H,000H,000H,003H
                                  1982  C
0920  23                         1983  C             DB      023H
                                  1984  C
0921  37 27 2D 37 31 15           1985  C             DB      037H,027H,02DH,037H,031H,015H
0927  04 11 00 07 06 07           1986  C             DB      004H,011H,000H,007H,006H,007H
092D  00 00 00 00 E1 24           1987  C             DB      000H,000H,000H,000H,0E1H,024H
0933  C7 14 08 E0 F0 A3           1988  C             DB      0C7H,014H,008H,0E0H,0F0H,0A3H
0939  FF                          1989  C             DB      0FFH
                                  1990  C
093A  00 01 02 03 04 05           1991  C             DB      000H,001H,002H,003H,004H,005H
0940  06 07 10 11 12 13           1992  C             DB      006H,007H,010H,011H,012H,013H
0946  14 15 16 17 08 00           1993  C             DB      014H,015H,016H,017H,008H,000H
094C  0F 00                       1994  C             DB      00FH,000H
                                  1995  C
094E  00 00 00 00 00 10           1996  C             DB      000H,000H,000H,000H,000H,010H
0954  0E 00 FF                    1997  C             DB      00EH,000H,0FFH
                                  1998  C
                                  1999  C  ;--9--
0957  28 18 08                    2000  C             DB      40D,24D,08D
095A  4000                        2001  C             DW      04000H
                                  2002  C
095C  00 00 00 03                 2003  C             DB      000H,000H,000H,003H
                                  2004  C
0960  23                         2005  C             DB      023H
                                  2006  C
0961  37 27 2D 37 31 15           2007  C             DB      037H,027H,02DH,037H,031H,015H
0967  04 11 00 07 06 07           2008  C             DB      004H,011H,000H,007H,006H,007H
096D  00 00 00 00 E1 24           2009  C             DB      000H,000H,000H,000H,0E1H,024H
0973  C7 14 08 E0 F0 A3           2010  C             DB      0C7H,014H,008H,0E0H,0F0H,0A3H
0979  FF                          2011  C             DB      0FFH
                                  2012  C
097A  00 01 02 03 04 05           2013  C             DB      000H,001H,002H,003H,004H,005H
0980  06 07 10 11 12 13           2014  C             DB      006H,007H,010H,011H,012H,013H
0986  14 15 16 17 08 00           2015  C             DB      014H,015H,016H,017H,008H,000H
098C  0F 00                       2016  C             DB      00FH,000H
```

```
                                                2017     C
098E  00 00 00 00 00 00 10                      2018     C               DB      000H,000H,000H,000H,000H,010H
0994  0E 00 FF                                  2019     C               DB      00EH,000H,0FFH
                                                2020     C
                                                2021     C       ;--A--
0997  28 18 08                                  2022     C               DB      40D,24D,08D
099A  4000                                      2023     C               DW      04000H
                                                2024     C
099C  00 00 00 03                               2025     C               DB      000H,000H,000H,003H
                                                2026     C
09A0  23                                        2027     C               DB      023H
                                                2028     C
09A1  37 27 2D 37 31 15                         2029     C               DB      037H,027H,02DH,037H,031H,015H
09A7  04 11 00 07 06 07                         2030     C               DB      004H,011H,000H,007H,006H,007H
09AD  00 00 00 00 E1 24                         2031     C               DB      000H,000H,000H,000H,0E1H,024H
09B3  C7 14 08 E0 F0 A3                         2032     C               DB      0C7H,014H,008H,0E0H,0F0H,0A3H
09B9  FF                                        2033     C               DB      0FFH
                                                2034     C
09BA  00 01 02 03 04 05                         2035     C               DB      000H,001H,002H,003H,004H,005H
09C0  06 07 10 11 12 13                         2036     C               DB      006H,007H,010H,011H,012H,013H
09C6  14 15 16 17 08 00                         2037     C               DB      014H,015H,016H,017H,008H,000H
09CC  0F 00                                     2038     C               DB      00FH,000H
                                                2039     C
09CE  00 00 00 00 00 00 10                      2040     C               DB      000H,000H,000H,000H,000H,010H
09D4  0E 00 FF                                  2041     C               DB      00EH,000H,0FFH
                                                2042     C
                                                2043     C       ;--B--
09D7  50 18 08                                  2044     C               DB      80D,24D,08D
09DA  1000                                      2045     C               DW      01000H
                                                2046     C
09DC  01 04 00 07                               2047     C               DB      001H,004H,000H,007H
                                                2048     C
09E0  23                                        2049     C               DB      023H
                                                2050     C
09E1  70 4F 5C 2F 5F 07                         2051     C               DB      070H,04FH,05CH,02FH,05FH,007H
09E7  04 11 00 07 06 07                         2052     C               DB      004H,011H,000H,007H,006H,007H
09ED  00 00 00 00 E1 24                         2053     C               DB      000H,000H,000H,000H,0E1H,024H
09F3  C7 28 08 E0 F0 A3                         2054     C               DB      0C7H,028H,008H,0E0H,0F0H,0A3H
09F9  FF                                        2055     C               DB      0FFH
                                                2056     C
09FA  00 00 00 00 00 00                         2057     C               DB      000H,000H,000H,000H,000H,000H
0A00  00 00 00 00 00 00                         2058     C               DB      000H,000H,000H,000H,000H,000H
0A06  00 00 00 00 00 00                         2059     C               DB      000H,000H,000H,000H,000H,000H
0A0C  0F 00                                     2060     C               DB      00FH,000H
                                                2061     C
0A0E  00 00 00 00 00 00                         2062     C               DB      000H,000H,000H,000H,000H,000H
0A14  04 00 FF                                  2063     C               DB      004H,000H,0FFH
                                                2064     C       ;--C--
0A17  50 18 0E                                  2065     C               DB      80D,24D,14D
0A1A  1000                                      2066     C               DW      01000H
                                                2067     C
0A1C  00 04 00 07                               2068     C               DB      000H,004H,000H,007H
                                                2069     C
0A20  A6                                        2070     C               DB      0A6H
                                                2071     C
0A21  60 4F 56 3A 51 60                         2072     C               DB      060H,04FH,056H,03AH,051H,060H
0A27  70 1F 00 0D 0B 0C                         2073     C               DB      070H,01FH,000H,00DH,00BH,00CH
0A2D  00 00 00 00 5E 2E                         2074     C               DB      000H,000H,000H,000H,05EH,02EH
0A33  5D 28 0D 5E 6E A3                         2075     C               DB      05DH,028H,00DH,05EH,06EH,0A3H
0A39  FF                                        2076     C               DB      0FFH
                                                2077     C
0A3A  00 00 00 00 00 00                         2078     C               DB      000H,000H,000H,000H,000H,000H
0A40  00 00 00 00 00 00                         2079     C               DB      000H,000H,000H,000H,000H,000H
0A46  00 00 00 00 0E 00                         2080     C               DB      000H,000H,000H,000H,00EH,000H
0A4C  0F 08                                     2081     C               DB      00FH,008H
                                                2082     C
0A4E  00 00 00 00 00 00                         2083     C               DB      000H,000H,000H,000H,000H,000H
0A54  04 00 FF                                  2084     C               DB      004H,000H,0FFH
                                                2085     C       ;--D--
0A57  28 18 08                                  2086     C               DB      40D,24D,08D
0A5A  2000                                      2087     C               DW      02000H
                                                2088     C
0A5C  0B 0F 00 06                               2089     C               DB      00BH,00FH,000H,006H
                                                2090     C
0A60  23                                        2091     C               DB      023H
                                                2092     C
0A61  37 27 2D 37 30 14                         2093     C               DB      037H,027H,02DH,037H,030H,014H
0A67  04 11 00 00 00 00                         2094     C               DB      004H,011H,000H,000H,000H,000H
0A6D  00 00 00 00 E1 24                         2095     C               DB      000H,000H,000H,000H,0E1H,024H
0A73  C7 14 00 E0 F0 E3                         2096     C               DB      0C7H,014H,000H,0E0H,0F0H,0E3H
0A79  FF                                        2097     C               DB      0FFH
                                                2098     C
0A7A  00 01 02 03 04 05                         2099     C               DB      000H,001H,002H,003H,004H,005H
0A80  06 07 10 11 12 13                         2100     C               DB      006H,007H,010H,011H,012H,013H
0A86  14 15 16 17 01 00                         2101     C               DB      014H,015H,016H,017H,001H,000H
0A8C  0F 00                                     2102     C               DB      00FH,000H
                                                2103     C
0A8E  00 00 00 00 00 00                         2104     C               DB      000H,000H,000H,000H,000H,000H
0A94  05 0F FF                                  2105     C               DB      005H,00FH,0FFH
                                                2106     C       ;--E--
0A97  50 18 08                                  2107     C               DB      80D,24D,08D
0A9A  4000                                      2108     C               DW      04000H
                                                2109     C
0A9C  01 0F 00 06                               2110     C               DB      001H,00FH,000H,006H
                                                2111     C
0AA0  23                                        2112     C               DB      023H
                                                2113     C
0AA1  70 4F 59 2D 5E 06                         2114     C               DB      070H,04FH,059H,02DH,05EH,006H
0AA7  04 11 00 00 00 00                         2115     C               DB      004H,011H,000H,000H,000H,000H
0AAD  00 00 00 00 E0 23                         2116     C               DB      000H,000H,000H,000H,0E0H,023H
0AB3  C7 28 00 DF EF E3                         2117     C               DB      0C7H,028H,000H,0DFH,0EFH,0E3H
0AB9  FF                                        2118     C               DB      0FFH
                                                2119     C
0ABA  00 01 02 03 04 05                         2120     C               DB      000H,001H,002H,003H,004H,005H
0AC0  06 07 10 11 12 13                         2121     C               DB      006H,007H,010H,011H,012H,013H
0AC6  14 15 16 17 01 00                         2122     C               DB      014H,015H,016H,017H,001H,000H
0ACC  0F 00                                     2123     C               DB      00FH,000H
                                                2124     C
0ACE  00 00 00 00 00 00                         2125     C               DB      000H,000H,000H,000H,000H,000H
0AD4  05 0F FF                                  2126     C               DB      005H,00FH,0FFH
                                                2127     C       ;--F--
0AD7  50 18 0E                                  2128     C               DB      80D,24D,14D
0ADA  8000                                      2129     C               DW      08000H
                                                2130     C
0ADC  05 0F 00 00                               2131     C               DB      005H,00FH,000H,000H
                                                2132     C
0AE0  A2                                        2133     C               DB      0A2H
                                                2134     C
0AE1  60 4F 56 1A 50 E0                         2135     C               DB      060H,04FH,056H,01AH,050H,0E0H
0AE7  70 1F 00 00 00 00                         2136     C               DB      070H,01FH,000H,000H,000H,000H
0AED  00 00 00 00 5E 2E                         2137     C               DB      000H,000H,000H,000H,05EH,02EH
0AF3  5D 14 0D 5E 6E 8B                         2138     C               DB      05DH,014H,00DH,05EH,06EH,08BH
0AF9  FF                                        2139     C               DB      0FFH
                                                2140     C
0AFA  00 08 00 00 18 18                         2141     C               DB      000H,008H,000H,000H,018H,018H
0B00  00 00 00 08 00 00                         2142     C               DB      000H,000H,000H,008H,000H,000H
```

```
0B06  00 18 00 00 0B 00      2143     C              DB      000H,018H,000H,000H,00BH,000H
0B0C  05 00                  2144     C              DB      005H,000H
                             2145     C
0B0E  00 00 00 00 00 10      2146     C              DB      000H,000H,000H,000H,000H,010H
0B14  07 0F FF               2147     C              DB      007H,00FH,0FFH
                             2148     C  ;-10--
0B17  50 18 0E               2149     C              DB      80D,24D,14D
0B1A  8000                   2150     C              DW      08000H
                             2151     C
0B1C  05 0F 00 00            2152     C              DB      005H,00FH,000H,000H
                             2153     C
0B20  A7                     2154     C              DB      0A7H
                             2155     C
0B21  5B 4F 53 17 50 BA      2156     C              DB      05BH,04FH,053H,017H,050H,0BAH
0B27  6C 1F 00 00 00 00      2157     C              DB      06CH,01FH,000H,000H,000H,000H
0B2D  00 00 00 00 5E 2B      2158     C              DB      000H,000H,000H,000H,05EH,02BH
0B33  5D 14 0F 5F 0A 8B      2159     C              DB      05DH,014H,00FH,05FH,00AH,08BH
0B39  FF                     2160     C              DB      0FFH
                             2161     C
0B3A  00 01 00 00 04 07      2162     C              DB      000H,001H,000H,000H,004H,007H
0B40  00 00 00 01 00 00      2163     C              DB      000H,000H,000H,001H,000H,000H
0B46  04 07 00 00 01 00      2164     C              DB      004H,007H,000H,000H,001H,000H
0B4C  05 00                  2165     C              DB      005H,000H
                             2166     C
0B4E  00 00 00 00 00 10      2167     C              DB      000H,000H,000H,000H,000H,010H
0B54  07 0F FF               2168     C              DB      007H,00FH,0FFH
                             2169     C
= 0440                       2170     C  BASE_2      EQU     $ - VIDEO_PARMS
                             2171     C
                             2172     C  ;-----   > 16K MODE VALUES
                             2173     C
                             2174     C  ;--F--
0B57  50 18 0E               2175     C              DB      80D,24D,14D
0B5A  8000                   2176     C              DW      08000H
                             2177     C
0B5C  01 0F 00 06            2178     C              DB      001H,00FH,000H,006H
                             2179     C
0B60  A2                     2180     C              DB      0A2H
                             2181     C
0B61  60 4F 56 3A 50 60      2182     C              DB      060H,04FH,056H,03AH,050H,060H
0B67  70 1F 00 00 00 00      2183     C              DB      070H,01FH,000H,000H,000H,000H
0B6D  00 00 00 00 5E 2E      2184     C              DB      000H,000H,000H,000H,05EH,02EH
0B73  5D 28 0D 5E 6E E3      2185     C              DB      05DH,028H,00DH,05EH,06EH,0E3H
0B79  FF                     2186     C              DB      0FFH
                             2187     C
0B7A  00 08 00 00 18 18      2188     C              DB      000H,008H,000H,000H,018H,018H
0B80  00 00 00 08 00 00      2189     C              DB      000H,000H,000H,008H,000H,000H
0B86  00 18 00 00 0B 00      2190     C              DB      000H,018H,000H,000H,00BH,000H
0B8C  05 00                  2191     C              DB      005H,000H
                             2192     C
0B8E  00 00 00 00 00 00      2193     C              DB      000H,000H,000H,000H,000H,000H
0B94  05 0F FF               2194     C              DB      005H,00FH,0FFH
                             2195     C
                             2196     C  ;-10--
0B97  50 18 0E               2197     C              DB      80D,24D,14D
0B9A  8000                   2198     C              DW      08000H
                             2199     C
0B9C  01 0F 00 06            2200     C              DB      001H,00FH,000H,006H
                             2201     C
0BA0  A7                     2202     C              DB      0A7H
                             2203     C
0BA1  5B 4F 53 37 52 00      2204     C              DB      05BH,04FH,053H,037H,052H,000H
0BA7  6C 1F 00 00 00 00      2205     C              DB      06CH,01FH,000H,000H,000H,000H
0BAD  00 00 00 00 5E 2B      2206     C              DB      000H,000H,000H,000H,05EH,02BH
0BB3  5D 28 0F 5F 0A E3      2207     C              DB      05DH,028H,00FH,05FH,00AH,0E3H
0BB9  FF                     2208     C              DB      0FFH
                             2209     C
0BBA  00 01 02 03 04 05      2210     C              DB      000H,001H,002H,003H,004H,005H
0BC0  14 07 38 39 3A 3B      2211     C              DB      014H,007H,038H,039H,03AH,03BH
0BC6  3C 3D 3E 3F 01 00      2212     C              DB      03CH,03DH,03EH,03FH,001H,000H
0BCC  0F 00                  2213     C              DB      00FH,000H
                             2214     C
0BCE  00 00 00 00 00 00      2215     C              DB      000H,000H,000H,000H,000H,000H
0BD4  05 0F FF               2216     C              DB      005H,00FH,0FFH
                             2217     C
                             2218     C
= 04C0                       2219     C  BASE_3      EQU     $ - VIDEO_PARMS
                             2220     C
                             2221     C  ;----- HI RES ALTERNATE VALUES
                             2222     C
                             2223     C  ;--0--
0BD7  28 18 0E               2224     C              DB      40D,24D,14D
0BDA  0800                   2225     C              DW      00800H
                             2226     C
0BDC  0B 03 00 03            2227     C              DB      00BH,003H,000H,003H
                             2228     C
0BE0  A7                     2229     C              DB      0A7H
                             2230     C
0BE1  2D 27 2B 2D 28 6D      2231     C              DB      02DH,027H,02BH,02DH,028H,06DH
0BE7  6C 1F 00 0D 06 07      2232     C              DB      06CH,01FH,000H,00DH,006H,007H
0BED  00 00 00 00 5E 2B      2233     C              DB      000H,000H,000H,000H,05EH,02BH
0BF3  5D 14 0F 5E 0A A3      2234     C              DB      05DH,014H,00FH,05EH,00AH,0A3H
0BF9  FF                     2235     C              DB      0FFH
                             2236     C
0BFA  00 01 02 03 04 05      2237     C              DB      000H,001H,002H,003H,004H,005H
0C00  14 07 38 39 3A 3B      2238     C              DB      014H,007H,038H,039H,03AH,038H
0C06  3C 3D 3E 3F 08 00      2239     C              DB      03CH,03DH,03EH,03FH,008H,000H
0C0C  0F 00                  2240     C              DB      00FH,000H
                             2241     C
0C0E  00 00 00 00 00 10      2242     C              DB      000H,000H,000H,000H,000H,010H
0C14  0E 00 FF               2243     C              DB      00EH,000H,0FFH
                             2244     C
                             2245     C  ;--1--
0C17  28 18 0E               2246     C              DB      40D,24D,14D
0C1A  0800                   2247     C              DW      00800H
                             2248     C
0C1C  0B 03 00 03            2249     C              DB      00BH,003H,000H,003H
                             2250     C
0C20  A7                     2251     C              DB      0A7H
                             2252     C
0C21  2D 27 2B 2D 28 6D      2253     C              DB      02DH,027H,02BH,02DH,028H,06DH
0C27  6C 1F 00 0D 06 07      2254     C              DB      06CH,01FH,000H,00DH,006H,007H
0C2D  00 00 00 00 5E 2B      2255     C              DB      000H,000H,000H,000H,05EH,02BH
0C33  5D 14 0F 5E 0A A3      2256     C              DB      05DH,014H,00FH,05EH,00AH,0A3H
0C39  FF                     2257     C              DB      0FFH
                             2258     C
0C3A  00 01 02 03 04 05      2259     C              DB      000H,001H,002H,003H,004H,005H
0C40  14 07 38 39 3A 3B      2260     C              DB      014H,007H,038H,039H,03AH,038H
0C46  3C 3D 3E 3F 08 00      2261     C              DB      03CH,03DH,03EH,03FH,008H,000H
0C4C  0F 00                  2262     C              DB      00FH,000H
                             2263     C
0C4E  00 00 00 00 00 10      2264     C              DB      000H,000H,000H,000H,000H,010H
0C54  0E 00 FF               2265     C              DB      00EH,000H,0FFH
                             2266     C
                             2267     C  ;--2--
0C57  50 18 0E               2268     C              DB      80D,24D,14D
```

```
0C5A  1000                        2269  C          DW      01000H
                                  2270  C
0C5C  01 03 00 03                 2271  C          DB      001H,003H,000H,003H
                                  2272  C
0C60  A7                          2273  C          DB      0A7H
                                  2274  C
0C61  5B 4F 53 37 51 5B           2275  C          DB      05BH,04FH,053H,037H,051H,05BH
0C67  6C 1F 00 0D 06 07           2276  C          DB      06CH,01FH,000H,00DH,006H,007H
0C6D  00 00 00 00 5E 2B           2277  C          DB      000H,000H,000H,000H,05EH,02BH
0C73  5D 28 0F 5E 0A A3           2278  C          DB      05DH,028H,00FH,05EH,00AH,0A3H
0C79  FF                          2279  C          DB      0FFH
                                  2280  C
0C7A  00 01 02 03 04 05           2281  C          DB      000H,001H,002H,003H,004H,005H
0C80  14 07 38 39 3A 3B           2282  C          DB      014H,007H,038H,039H,03AH,03BH
0C86  3C 3D 3E 3F 08 00           2283  C          DB      03CH,03DH,03EH,03FH,008H,000H
0C8C  0F 00                       2284  C          DB      00FH,000H
                                  2285  C
0C8E  00 00 00 00 00 10           2286  C          DB      000H,000H,000H,000H,000H,010H
0C94  0E 00 FF                    2287  C          DB      00EH,000H,0FFH
                                  2288  C
                                  2289  C  ;--3--
0C97  50 18 0E                    2290  C          DB      80D,24D,14D
0C9A  1000                        2291  C          DW      01000H
                                  2292  C
0C9C  01 03 00 03                 2293  C          DB      001H,003H,000H,003H
                                  2294  C
0CA0  A7                          2295  C          DB      0A7H
                                  2296  C
0CA1  5B 4F 53 37 51 5B           2297  C          DB      05BH,04FH,053H,037H,051H,05BH
0CA7  6C 1F 00 0D 06 07           2298  C          DB      06CH,01FH,000H,00DH,006H,007H
0CAD  00 00 00 00 5E 2B           2299  C          DB      000H,000H,000H,000H,05EH,02BH
0CB3  5D 28 0F 5E 0A A3           2300  C          DB      05DH,028H,00FH,05EH,00AH,0A3H
0CB9  FF                          2301  C          DB      0FFH
                                  2302  C
0CBA  00 01 02 03 04 05           2303  C          DB      000H,001H,002H,003H,004H,005H
0CC0  14 07 38 39 3A 3B           2304  C          DB      014H,007H,038H,039H,03AH,03BH
0CC6  3C 3D 3E 3F 08 00           2305  C          DB      03CH,03DH,03EH,03FH,008H,000H
0CCC  0F 00                       2306  C          DB      00FH,000H
                                  2307  C
0CCE  00 00 00 00 00 10           2308  C          DB      000H,000H,000H,000H,000H,010H
0CD4  0E 00 FF                    2309  C          DB      00EH,000H,0FFH
                                  2310  C
                                  2311  C
                                  2312  C          SUBTTL
                                  2313
                                  2314       ;----- VECTOR INTO <AH> SPECIFIED FUNCTION
                                  2315
0CD7                              2316       COMBO_VIDEO    PROC    NEAR
0CD7  FB                          2317            STI                             ; INTERRUPTS ON
0CD8  FC                          2318            CLD                             ; SET DIRECTION FORWARD
0CD9  55                          2319            PUSH    BP                      ; SAVE THE REGISTER SET
0CDA  06                          2320            PUSH    ES
0CDB  1E                          2321            PUSH    DS
0CDC  52                          2322            PUSH    DX
0CDD  51                          2323            PUSH    CX
0CDE  53                          2324            PUSH    BX
0CDF  56                          2325            PUSH    SI
0CE0  57                          2326            PUSH    DI
                                  2327
0CE1  50                          2328            PUSH    AX                      ; SAVE AX VALUE
0CE2  8A C4                       2329            MOV     AL,AH                   ; GET INTO LOW BYTE
0CE4  32 E4                       2330            XOR     AH,AH                   ; ZERO TO HIGH BYTE
0CE6  D1 E0                       2331            SAL     AX,1                    ; * 2 FOR TABLE LOOKUP
0CE8  8B F0                       2332            MOV     SI,AX                   ; PUT INTO SI FOR BRANCH
0CEA  3D 0028                     2333            CMP     AX,T2L                  ; TEST FOR WITHIN RANGE
0CED  72 06                       2334            JB      M2                      ; BRANCH AROUND BRANCH
0CEF  58                          2335            POP     AX                      ; RECOVER REGISTER
0CF0  CD 42                       2336            INT     42H                     ; PASS UNRECOGNIZED CALL
0CF2  E9 219E R                   2337            JMP     V_RET                   ; RETURN TO CALLER
0CF5                              2338       M2:
                                  2339            ASSUME  DS:ABS0
0CF5  E8 0CFE R                   2340            CALL    DDS
0CF8  58                          2341            POP     AX                      ; RECOVER
0CF9  2E: FF A4 06EF R            2342            JMP     WORD PTR CS:[SI + OFFSET T2]    ; JMP TO AH=0 THRU AH=XX
                                  2343
                                  2344       ;----- UTILITY ROUTINES
                                  2345
                                  2346       ;----- SET DS TO THE DATA SEGMENT
                                  2347
0CFE                              2348       DDS     PROC    NEAR
0CFE  50                          2349            PUSH    AX                      ; SAVE REGISTER
0CFF  2B C0                       2350            SUB     AX,AX
0D01  8E D8                       2351            MOV     DS,AX
0D03  58                          2352            POP     AX                      ; RESTORE REGISTER
0D04  C3                          2353            RET
0D05                              2354       DDS     ENDP
                                  2355
0D05                              2356       WHAT_BASE      PROC    NEAR
                                  2357            ASSUME  DS:ABS0
0D05  1E                          2358            PUSH    DS                      ; SAVE DATA SEGMENT
0D06  E8 0CFE R                   2359            CALL    DDS                     ; GET LOW MEMORY SEGMENT
0D09  8B 16 0463 R                2360            MOV     DX,ADDR_6845            ; GET CRTC ADDRESS
0D0D  80 E2 F0                    2361            AND     DL,0F0H                 ; STRIP OFF LOW NIBBLE
0D10  80 CA 0A                    2362            OR      DL,0AH                  ; SET TO STATUS REGISTER
0D13  1F                          2363            POP     DS
0D14  C3                          2364            RET
0D15                              2365       WHAT_BASE      ENDP
                                  2366
0D15                              2367       OUT_DX  PROC    NEAR                    ; AH=INDEX,AL=DATA,DX=PORT
0D15  86 C4                       2368            XCHG    AL,AH                   ; GET INDEX VALUE
0D17  EE                          2369            OUT     DX,AL                   ; SET INDEX REG
0D18  42                          2370            INC     DX                      ; SET DX TO DATA REG
0D19  86 C4                       2371            XCHG    AL,AH                   ; GET DATA VALUE
0D1B  EE                          2372            OUT     DX,AL                   ; SET DATA REG
0D1C  4A                          2373            DEC     DX                      ; SET DX BACK TO INDEX
0D1D  C3                          2374            RET
0D1E                              2375       OUT_DX  ENDP
                                  2376
                                  2377       ;----- ROUTINE TO SOUND BEEPER
                                  2378
0D1E                              2379       BP_1    PROC    NEAR
0D1E  EE                          2380            OUT     DX,AL
0D1F  C3                          2381            RET
0D20                              2382       BP_1    ENDP
                                  2383
0D20                              2384       BEEP    PROC    NEAR
0D20  52                          2385            PUSH    DX
0D21  BA 0043                     2386            MOV     DX,TIMER+3
0D24  B0 B6                       2387            MOV     AL,10110110B            ; SEL TIM 2,LSB,MSB,BINARY
0D26  E8 0D1E R                   2388            CALL    BP_1                    ; WRITE THE TIMER MODE REG
0D29  B8 0533                     2389            MOV     AX,533H                 ; DIVISOR FOR 1000 HZ
0D2C  4A                          2390            DEC     DX
0D2D  E8 0D1E R                   2391            CALL    BP_1                    ; WRITE TIMER 2 CNT - LSB
0D30  8A C4                       2392            MOV     AL,AH
0D32  E8 0D1E R                   2393            CALL    BP_1                    ; WRITE TIMER 2 CNT - MSB
0D35  BA 0061                     2394            MOV     DX,PORT_B
```

```
0D38  EC                      2395            IN      AL,DX                   ; GET SETTING OF PORT
0D39  8A E0                   2396            MOV     AH,AL                   ; SAVE THAT SETTING
0D3B  0C 03                   2397            OR      AL,03                   ; TURN SPEAKER ON
0D3D  E8 0D1E R               2398            CALL    BP_1
0D40  2B C9                   2399            SUB     CX,CX                   ; SET CNT TO WAIT 500 MS
0D42                          2400    G7:
0D42  E2 FE                   2401            LOOP    G7                      ; DELAY BEFORE TURNING OFF
0D44  FE CB                   2402            DEC     BL                      ; DELAY CNT EXPIRED?
0D46  75 FA                   2403            JNZ     G7                      ; NO-CONTINUE BEEPING SPK
0D48  8A C4                   2404            MOV     AL,AH                   ; RECOVER VALUE OF PORT
0D4A  E8 0D1E R               2405            CALL    BP_1
0D4D  5A                      2406            POP     DX
0D4E  C3                      2407            RET                             ; RETURN TO CALLER
0D4F                          2408    BEEP    ENDP
                              2409
                              2410    ;----- FIND THE PARAMETER TABLE VECTOR IN THE SAVE TABLE
                              2411
0D4F                          2412    SET_BASE        PROC    NEAR
                              2413            ASSUME  DS:ABS0
0D4F  E8 0CFE R               2414            CALL    DDS
0D52  C4 1E 04A8 R            2415            LES     BX,SAVE_PTR             ; GET PTR TP PTR TABLE
0D56  26: C4 1F               2416            LES     BX,DWORD PTR ES:[BX]    ; GET PARAMETER PTR
0D59  C3                      2417            RET
0D5A                          2418    SET_BASE        ENDP
                              2419
                              2420    ;----- ESTABLISH ADDRESSING TO THE CORRECT MODE TABLE ENTRY
                              2421
0D5A                          2422    MAKE_BASE       PROC    NEAR
                              2423            ASSUME  DS:ABS0
0D5A  51                      2424            PUSH    CX
0D5B  52                      2425            PUSH    DX
0D5C  E8 0D4F R               2426            CALL    SET_BASE                ; GET PARM TBL PTR
0D5F  8A 26 0449 R            2427            MOV     AH,CRT_MODE
0D63  F6 06 0487 R 60         2428            TEST    INFO,060H               ; TEST FOR BASE CARD
0D68  74 18                   2429            JZ      B_M_1                   ; MIN MEMORY
                              2430
                              2431
                              2432    ;----- WE HAVE A MEMORY EXPANSION OPTION HERE
                              2433
0D6A  80 FC 0F                2434            CMP     AH,0FH
0D6D  75 07                   2435            JNE     B_M_2
0D6F  81 C3 0440              2436            ADD     BX,BASE_2 - BASE_1
0D73  EB 33 90                2437            JMP     B_M_OUT
0D76                          2438    B_M_2:
0D76  80 FC 10                2439            CMP     AH,010H
0D79  75 07                   2440            JNE     B_M_1
0D7B  81 C3 0480              2441            ADD     BX,BASE_2 + M_TBL_LEN - BASE_1
0D7F  EB 27 90                2442            JMP     B_M_OUT
0D82                          2443    B_M_1:
0D82  80 FC 03                2444            CMP     AH,03H
0D85  77 14                   2445            JA      B_M_3                   ; SKIP ENHANCED PORTION
                              2446
                              2447    ;----- CHECK THE SWITCH SETTING FOR ENHANCEMENT
                              2448
0D87  A0 0488 R               2449            MOV     AL,INFO_3
0D8A  24 0F                   2450            AND     AL,0FH
0D8C  3C 03                   2451            CMP     AL,03H                  ; SECONDARY EMULATE SETTING
0D8E  74 07                   2452            JE      BRS
0D90  3C 09                   2453            CMP     AL,09H                  ; PRIMARY EMULATE SETTING
0D92  74 03                   2454            JE      BRS
0D94  EB 05 90                2455            JMP     B_M_3
                              2456
                              2457    ;----- WE WILL PERFORM ENHANCEMENT
                              2458
0D97                          2459    BRS:
0D97  81 C3 04C0              2460            ADD     BX,BASE_3 - BASE_1      ; VECTOR TO ENHANCEMENT TBL
0D9B                          2461    B_M_3:
0D9B  8A 0E 0449 R            2462            MOV     CL,CRT_MODE
0D9F  2A ED                   2463            SUB     CH,CH
0DA1  E3 05                   2464            JCXZ    B_M_4
                              2465
                              2466    ;----- THIS LOOP WILL MOVE THE PTR TO THE INDIVIDUAL MODE ENTRY
                              2467
0DA3                          2468    B_M_5:
0DA3  83 C3 40                2469            ADD     BX,M_TBL_LEN            ; LENGTH OF ONE MODE ENTRY
0DA6  E2 FB                   2470            LOOP    B_M_5
0DA8                          2471    B_M_4:
0DA8                          2472    B_M_OUT:
0DA8  5A                      2473            POP     DX
0DA9  59                      2474            POP     CX
0DAA  C3                      2475            RET
0DAB                          2476    MAKE_BASE       ENDP
                              2477
                              2478    ;----- PROGRAM THE EGA REGISTERS FROM THE PARAMETER TABLE
                              2479
0DAB                          2480    SET_REGS        PROC    NEAR
                              2481            ASSUME  DS:ABS0,ES:NOTHING
                              2482
                              2483    ;----- PROGRAM THE SEQUENCER
                              2484
0DAB  E8 0D5A R               2485            CALL    MAKE_BASE               ; GET TABLE PTR
0DAE  83 C3 05                2486            ADD     BX,TFS_LEN              ; MODE TO SEQUENCER PARMS
0DB1  B6 03                   2487            MOV     DH,3
0DB3  B2 C4                   2488            MOV     DL,SEQ_ADDR
0DB5  B8 0001                 2489            MOV     AX,0001H                ; RESET SEQUENCER
0DB8  FA                      2490            CLI                             ; DISABLE INTERRUPTS
0DB9  E8 0D15 R               2491            CALL    OUT_DX
0DBC  26: 8A 07               2492            MOV     AL,ES:[BX]              ; GET SEQUENCER VALUE
0DBF  FE C4                   2493            INC     AH                      ; NEXT INDEX
0DC1  E8 0D15 R               2494            CALL    OUT_DX                  ; SET IT
0DC4                          2495    D1:
0DC4  FE C4                   2496            INC     AH                      ; NEXT INDEX REGISTER
0DC6  43                      2497            INC     BX                      ; NEXT TABLE ENTRY
0DC7  26: 8A 07               2498            MOV     AL,ES:[BX]
0DCA  E8 0D15 R               2499            CALL    OUT_DX
0DCD  80 FC 05                2500            CMP     AH,M1+1
0DD0  72 F2                   2501            JB      D1
                              2502
0DD2  26: 8A 07               2503            MOV     AL,ES:[BX]
0DD5  43                      2504            INC     BX
0DD6  B2 C2                   2505            MOV     DL,MISC_OUTPUT
0DD8  EE                      2506            OUT     DX,AL
0DD9  B2 C4                   2507            MOV     DL,SEQ_ADDR
0DDB  B8 0003                 2508            MOV     AX,0003H
0DDE  E8 0D15 R               2509            CALL    OUT_DX                  ; START SEQUENCER
0DE1  FB                      2510            STI                             ; ENABLE INTERRUPTS
                              2511
                              2512    ;----- PROGRAM THE CRT CONTROLLER
                              2513
0DE2  8B 16 0463 R            2514            MOV     DX,ADDR_6845            ; CRTC INDEX REGISTER
0DE6  2A E4                   2515            SUB     AH,AH                   ; COUNTER
0DE8                          2516    X1:
0DE8  26: 8A 07               2517            MOV     AL,ES:[BX]              ; GET VALUE FROM TABLE
0DEB  E8 0D15 R               2518            CALL    OUT_DX                  ; SET CRTC REGISTER
0DEE  43                      2519            INC     BX                      ; NEXT TABLE ENTRY
0DEF  FE C4                   2520            INC     AH                      ; NEXT INDEX VALUE
0DF1  80 FC 19                                CMP     AH,M4                   ; TEST REGISTER COUNT
```

```
0DF4  72 F2                2521                JB      X1                      ; DO THE REST
0DF6  26: 8B 47 F1         2522                MOV     AX,ES:[BX][-0FH]        ; GET CURSOR MODE
0DFA  86 E0                2523                XCHG    AH,AL
0DFC  A3 0460 R            2524                MOV     CURSOR_MODE,AX          ; SET LOW RAM VALUE
                           2525
                           2526        ;----- PROGRAM THE ATTRIBUTE CHIP
                           2527
0DFF  8B F3                2528                MOV     SI,BX
0E01  E8 0DD5 R            2529                CALL    WHAT_BASE
0E04  EC                   2530                IN      AL,DX
0E05  B2 C0                2531                MOV     DL,ATTR_WRITE
0E07  2A E4                2532                SUB     AH,AH                   ; INDEX COUNTER
0E09                       2533        D3:
0E09  26: 8A 07            2534                MOV     AL,ES:[BX]              ; GET DATA VALUE
0E0C  86 E0                2535                XCHG    AH,AL
0E0E  EE                   2536                OUT     DX,AL
0E0F  86 E0                2537                XCHG    AH,AL
0E11  EE                   2538                OUT     DX,AL
0E12  43                   2539                INC     BX                      ; NEXT DATA VALUE
0E13  FE C4                2540                INC     AH                      ; NEXT INDEX VALUE
0E15  80 FC 14             2541                CMP     AH,M5                   ; TEST REGISTER COUNT
0E18  72 EF                2542                JB      D3                      ; DO THE REST
                           2543
0E1A  B0 00                2544                MOV     AL,0
0E1C  EE                   2545                OUT     DX,AL
                           2546
                           2547        ;----- CHECK IF PALETTE REGISTER VALUES ARE TO BE SAVED
                           2548
0E1D  1E                   2549                PUSH    DS
0E1E  06                   2550                PUSH    ES
0E1F  C4 3E 04A8 R         2551                LES     DI,SAVE_PTR             ; GET TABLE PTR
0E23  26: C4 7D 04         2552                LES     DI,DWORD PTR ES:[DI][4] ; GET PALETTE PTR
0E27  8C C0                2553                MOV     AX,ES
0E29  0B C7                2554                OR      AX,DI
0E2B  74 09                2555                JZ      SAVE_OUT                ; IF ZERO, NO SAVE OCCURS
                           2556
                           2557        ;----- STORE AWAY THE PALETTE VALUES IN RAM SAVE AREA
                           2558
0E2D  1F                   2559                POP     DS
0E2E  1E                   2560                PUSH    DS
0E2F  B9 0010              2561                MOV     CX,16D
0E32  F3/ A4               2562        REP     MOVSB                           ; SAVE THE PALETTE REGS
0E34  46                   2563                INC     SI
0E35  A4                   2564                MOVSB                           ; SAVE THE OVERSCAN REG
0E36                       2565        SAVE_OUT:
0E36  07                   2566                POP     ES
0E37  1F                   2567                POP     DS
                           2568
                           2569        ;----- PROGRAM THE GRAPHICS CHIPS
                           2570
0E38  B2 CC                2571                MOV     DL,GRAPH_1_POS
0E3A  B0 00                2572                MOV     AL;0
0E3C  EE                   2573                OUT     DX,AL
0E3D  B2 CA                2574                MOV     DL,GRAPH_2_POS
0E3F  B0 01                2575                MOV     AL;1
0E41  EE                   2576                OUT     DX,AL
0E42  B2 CE                2577                MOV     DL;GRAPH_ADDR
0E44  2A E4                2578                SUB     AH,AH
0E46                       2579        D4:
0E46  26: 8A 07            2580                MOV     AL,ES:[BX]              ; PARAMETER BYTE
0E49  E8 0D15 R            2581                CALL    OUT_DX                  ; SET IT
0E4C  43                   2582                INC     BX                      ; NEXT BYTE
0E4D  FE C4                2583                INC     AH                      ; NEXT REGISTER
0E4F  80 FC 09             2584                CMP     AH,M6
0E52  72 F2                2585                JB      D4                      ; CONTINUE
0E54  C3                   2586                RET
0E55                       2587        SET_REGS        ENDP
                           2588
                           2589        ;----- MODE SET REGEN CLEAR ROUTINE
                           2590
0E55                       2591        BLANK   PROC    NEAR                    ; FILL REGEN WITH BLANKS
                           2592                ASSUME  DS:ABS0,ES:NOTHING
0E55  A0 0487 R            2593                MOV     AL,INFO                 ; SEE IF BLANK IS TO OCCUR
0E58  A8 80                2594                TEST    AL,080H                 ; MODE SET HIGH BIT
0E5A  75 39                2595                JNZ     OUT_1                   ; SKIP BLANK FOR REGEN
0E5C  BA B800              2596                MOV     DX,0B800H               ; COLOR MODE REGEN ADDRESS
0E5F  A0 0449 R            2597                MOV     AL,CRT_MODE             ; CURRENT MODE SET
0E62  3C 06                2598                CMP     AL,6                    ; 0-6 ARE COLOR MODES
0E64  76 0A                2599                JBE     CG0
0E66  BA B000              2600                MOV     DX,0B000H               ; MONOCHROME REGEN ADDRESS
0E69  3C 07                2601                CMP     AL,7                    ; MONOCHROME MODE
0E6B  74 03                2602                JE      CG0
0E6D  BA A000              2603                MOV     DX,0A000H               ; REMAINING MODES
0E70                       2604        CG0:
0E70  BB 0720              2605                MOV     BX,0720H                ; ALPHA BLANK VALUE
0E73  3C 04                2606                CMP     AL,4                    ; ALPHAMODES 0-3
0E75  72 06                2607                JB      WW1
0E77  3C 07                2608                CMP     AL,7                    ; ALPHA MODE
0E79  74 02                2609                JE      WW1
0E7B  2B DB                2610                SUB     BX,BX                   ; GRAPHICS BLANK VALUE
0E7D                       2611        WW1:
                           2612                SRLOAD  ES                      ; SET THE REGEN SEGMENT
0E7D  8E C2                2613    +           MOV     ES,DX
0E7F  8B 0E 044C R         2614                MOV     CX,CRT_LEN
0E83  E3 10                2615                JCXZ    OUT_1
0E85  B9 8000              2616                MOV     CX,08000H
0E88  80 FE A0             2617                CMP     DH,0A0H
0E8B  74 02                2618                JE      N_BA
0E8D  B5 40                2619                MOV     CH,040H
0E8F                       2620        N_BA:
0E8F  8B C3                2621                MOV     AX,BX                   ; BLANK VALUE
0E91  2B FF                2622                SUB     DI,DI                   ; CLEAR POINTER
0E93  F3/ AB               2623                REP     STOSW                   ; CLEAR THE PAGE
0E95                       2624        OUT_1:
0E95  C3                   2625                RET                             ; RETURN TO CALLER
0E96                       2626        BLANK   ENDP
                           2627
0E96                       2628        PH_5    PROC    NEAR
0E96  E8 1DB7 R            2629                CALL    PAL_ON
0E99  C3                   2630                RET
0E9A                       2631        PH_5    ENDP
                           2632
                           2633        ;----- SEE IF WE ARE TO SUPPORT 640 X 350 ON A 640 X 200 MODE
                           2634
0E9A                       2635        BRST_DET        PROC    NEAR
                           2636                ASSUME  DS:ABS0
0E9A  50                   2637                PUSH    AX
0E9B  1E                   2638                PUSH    DS
0E9C  E8 0CFE R            2639                CALL    DDS
0E9F  A0 0488 R            2640                MOV     AL,INFO_3
0EA2  1F                   2641                POP     DS
0EA3  24 0F                2642                AND     AL,0FH
0EA5  3C 03                2643                CMP     AL,03H                  ; EMULATE MODE
0EA7  74 07                2644                JE      B_YES
0EA9  3C 09                2645                CMP     AL,09H                  ; EMULATE MODE
0EAB  74 03                2646                JE      B_YES
```

```
0EAD  58                      2647            POP     AX
0EAE  F8                      2648            CLC
0EAF  C3                      2649            RET
0EB0                          2650    B_YES:
0EB0  58                      2651            POP     AX
0EB1  F9                      2652            STC
0EB2  C3                      2653            RET
0EB3                          2654    BRST_DET        ENDP
                              2655
                              2656    ;----- MODE SET
                              2657
0EB3                          2658    AHO:
                              2659            ASSUME  DS:ABS0
0EB3  FA                      2660            CLI
0EB4  C7 06 010C R 0000 E     2661            MOV     WORD PTR GRX_SET, OFFSET CGDDOT
0EBA  8C 0E 010E R            2662            MOV     WORD PTR GRX_SET+2, CS
0EBE  FB                      2663            STI
0EBF  80 26 0487 R F3         2664            AND     INFO,11110011B                  ; TURN OFF RETRACE BIT
                              2665                                                    ;  EGA ACTIVE BIT
0EC4  50                      2666            PUSH    AX                              ; SAVE
0EC5  F6 06 0487 R 02         2667            TEST    INFO,2
0ECA  74 2C                   2668            JZ      ST_1                            ; THERE IS NO MONOCHROME
0ECC  A1 0410 R               2669            MOV     AX,EQUIP_FLAG                   ; THERE IS A MONOCHROME
0ECF  24 30                   2670            AND     AL,030H                         ; CHECK THE EQUIPMENT FLAG
0ED1  3C 30                   2671            CMP     AL,030H                         ;  FOR MONOCHROME CALL
0ED3  74 48                   2672            JE      ST_2                            ; IT IS A MONOCHROME CALL
                              2673
                              2674    ;----- FALL THROUGH => REGULAR COLOR CARD SETUP
                              2675
0ED5  C6 06 0484 R 18         2676            MOV     ROWS,024D
0EDA  C7 06 0485 R 0008       2677            MOV     POINTS,8
0EE0  58                      2678            POP     AX                              ; RECOVER
0EE1  80 0E 0487 R 08         2679            OR      INFO,00001000B                  ; EGA NOT ACTIVE
0EE6  3C 01                   2680            CMP     AL,1
0EE8  76 09                   2681            JBE     ST_7                            ; WAIT FOR RETRACE ON
0EEA  3C 04                   2682            CMP     AL,4                            ; MODES 2,3 ONLY
0EEC  73 05                   2683            JAE     ST_7
0EEE  80 0E 0487 R 04         2684            OR      INFO,00000100B                  ; DO RETRACE
0EF3                          2685    ST_7:
0EF3  CD 42                   2686            INT     42H                             ; OTHER ADAPTER MODE CALL
0EF5  E9 219E R               2687            JMP     V_RET                           ; BACK TO CALLER
                              2688
                              2689    ;----- AT THIS POINT THERE IS NO MONOCHROME ATTACHED TO THE ADAPTER
                              2690
0EF8                          2691    ST_1:
0EF8  A1 0410 R               2692            MOV     AX,EQUIP_FLAG                   ; TEST THE EQUIPMENT FLAG
0EFB  24 30                   2693            AND     AL,030H                         ;  TO SEE IF THIS IS A
0EFD  3C 30                   2694            CMP     AL,030H                         ;  MONOCHROME SETUP CALL
0EFF  75 4O                   2695            JNE     ST_3                            ; MUST BE COLOR TO CARD
                              2696
                              2697    ;----- FALL THROUGH => REGULAR MONOCHROME CARD SETUP
                              2698
0F01  C6 06 0484 R 18         2699            MOV     ROWS,024D
0F06  C7 06 0485 R 000E       2700            MOV     POINTS,014D
0F0C  58                      2701            POP     AX                              ; RECOVER
0F0D  42                      2702            INT     42H                             ; OTHER ADAPTER MODE CALL
0F0F  C7 06 0460 R 0B0C       2703            MOV     CURSOR_MODE,0BOCH               ; FIX PLANAR VALUE
0F15  80 0E 0487 R 08         2704            OR      INFO,8                          ; THE EGA IS NOT ACTIVE
0F1A  E9 219E R               2705            JMP     V_RET                           ; BACK TO CALLER
                              2706
                              2707    ;----- MONOCHROME SETUP TO THE ADAPTER
                              2708
0F1D                          2709    ST_2:
0F1D  58                      2710            POP     AX                              ; RECOVER
0F1E  50                      2711            PUSH    AX                              ; SAVE
0F1F  B6 03                   2712            MOV     DH,3
0F21  24 80                   2713            AND     AL,080H                         ; PICK OFF THE CLEAR BIT
0F23  80 26 0487 R 7F         2714            AND     INFO,07FH                       ; MASK OFF THE OTHER BITS
0F28  08 06 0487 R            2715            OR      INFO,AL                         ; SAVE REGEN CLEAR BIT
0F2C  58                      2716            POP     AX                              ; RECOVER TRUE CALL VALUE
0F2D  24 7F                   2717            AND     AL,07FH                         ; ALREADY DEALT WITH D7
0F2F  3C OF                   2718            CMP     AL,0FH                          ; A MONOCHROME MODE
0F31  74 02                   2719            JE      ST_2A                           ; DO THIS MODE
0F33  B0 07                   2720            MOV     AL,7                            ; REGULAR MONOCHROME
0F35                          2721    ST_2A:
0F35  A2 0449 R               2722            MOV     CRT_MODE,AL                     ; SAVE MODE VALUE
0F38  B2 B4                   2723            MOV     DL,CRTC_ADDR_B                  ; IT IS 3-B-X
0F3A  89 16 0463 R            2724            MOV     ADDR_6845,DX                    ; SAVE CRTC ADDRESS
0F3E  EB 1C 90                2725            JMP     QQ1                             ; CONTINUE THE MODE SET
                              2726
                              2727    ;----- COLOR SETUP TO THE ADAPTER
                              2728
0F41                          2729    ST_3:
0F41  58                      2730            POP     AX                              ; RECOVER PARAMETER VALUE
0F42  50                      2731            PUSH    AX                              ; SAVE IT
0F43  B6 03                   2732            MOV     DH,3
0F45  24 80                   2733            AND     AL,080H                         ; ISOLATE REGEN CLEAR BIT
0F47  80 26 0487 R 7F         2734            AND     INFO,07FH                       ; PREPARE INFO BYTE
0F4C  08 06 0487 R            2735            OR      INFO,AL                         ; SET IT, OR NOT
0F50  58                      2736            POP     AX                              ; RECOVER TRUE MODE CALL
0F51  24 7F                   2737            AND     AL,07FH                         ; DONE WITH D7
0F53  A2 0449 R               2738            MOV     CRT_MODE,AL                     ; SAVE THIS MODE
0F56  B2 D4                   2739            MOV     DL,CRTC_ADDR                    ; 3-D-X
0F58  89 16 0463 R            2740            MOV     ADDR_6845,DX                    ; SAVE CRTC ADDRESS
0F5C                          2741
0F5C  C7 06 044E R 0000       2742    QQ1:    MOV     CRT_START,0                     ; SAVE START ADDRESS
0F62  C6 06 0462 R 00         2743            MOV     ACTIVE_PAGE,0                   ; RESET PAGE VALUE TO ZERO
                              2744            ASSUME  ES:NOTHING
0F67  B9 0008                 2745            MOV     CX,8                            ; 8 PAGES OF CURSOR VALUES
0F6A  BF 0450 R               2746            MOV     DI,OFFSET CURSOR_POSN           ; OFFSET
0F6D  1E                      2747            PUSH    DS                              ; ESTABLISH
0F6E  07                      2748            POP     ES                              ;  ADDRESSING
0F6F  2B CO                   2749            SUB     AX,AX                           ; O THOSE CURSOR LOCATIONS
0F71  F3/ AB                  2750            REP     STOSW                           ; CLEAR OUT SAVED VALUES
                              2751
0F73  E8 0D5A R               2752            CALL    MAKE_BASE
                              2753
0F76  26: 8A 07               2754            MOV     AL,ES:[BX]                      ; GET COLUMN COUNT
0F79  2A E4                   2755            SUB     AH,AH                           ; ZERO HIGH BYTE
0F7B  A3 044A R               2756            MOV     CRT_COLS,AX                     ; STORE COLUMN VALUE
                              2757
0F7E  26: 8A 47 01            2758            MOV     AL,ES:[BX][1]                   ; GET ROW VALUE
0F82  A2 0484 R               2759            MOV     ROWS,AL                         ; STORE ROW VALUE
                              2760
0F85  26: 8A 47 02            2761            MOV     AL,ES:[BX][2]                   ; GET THE BYTES/CHAR
0F89  2A E4                   2762            SUB     AH,AH                           ; ZERO HIGH BYTE
0F8B  A3 0485 R               2763            MOV     POINTS,AX                       ; STORE BYTES/CHAR
                              2764
0F8E  26: 8B 47 03            2765            MOV     AX,ES:[BX][3]                   ; GET PAGE SIZE
0F92  A3 044C R               2766            MOV     CRT_LEN,AX                      ; STORE PAGE LENGTH
                              2767
0F95  2B DB                   2768            SUB     BX,BX                           ; ZERO
0F97  B0 01                   2769            MOV     AL,1                            ; MONOCHROME ALPHA CHAR GEN
0F99  8A 26 0449 R            2770            MOV     AH,CRT_MODE                     ; GET CURRENT MODE
0F9D  80 FC 07                2771            CMP     AH,7                            ; IS IT MONOCHROME
0FA0  74 OC                   2772            JE      ENTRY_2                         ; 9X14 FONT
```

```
0FA2  80 FC 03                2773                CMP     AH,03H
0FA5  77 35                   2774                JA      ENTRY_1
                              2775
0FA7  E8 0E9A R               2776                CALL    BRST_DET
0FAA  72 02                   2777                JC      ENTRY_2
                              2778
0FAC  B0 02                   2779                MOV     AL,2                        ; COLOR ALPHA CHAR GEN
0FAE                          2780        ENTRY_2:
0FAE  E8 1EAE R               2781                CALL    CH_GEN                      ; LOAD ALPHA CHAR GEN
0FB1  E8 0CFE R               2782                CALL    DDS
0FB4  8A 26 0449 R            2783                MOV     AH,CRT_MODE                 ; GET CURRENT MODE
0FB8  80 FC 07                2784                CMP     AH,7                        ; IS IT MONOCHROME
0FBB  74 03                   2785                JE      FDG_IT                      ; 9X14 FONT
0FBD  EB 1D 90                2786                JMP     ENTRY_1
0FC0                          2787        FDG_IT:
0FC0  BD 0000 E               2788                MOV     BP,OFFSET CGMN_FDG          ; TABLE POINTER
0FC3  BB 0E00                 2789                MOV     BX,0E00H                    ; 14 BYTES PER CHAR
0FC6                          2790        FDG:
0FC6  0E                      2791                PUSH    CS                          ; GET THE ROM SEGMENT
0FC7  07                      2792                POP     ES                          ;  INTO ES
0FC8  26: 8B 56 00            2793                MOV     DX,ES:[BP]                  ; GET THE CHAR HEX CODE
0FCC  0B D2                   2794                OR      DX,DX                       ; ZERO = NO MORE CHARS
0FCE  74 0C                   2795                JZ      ENTRY_1                     ; NO MORE
0FD0  B9 0001                 2796                MOV     CX,1                        ; DO ONE CHAR AT A TIME
0FD3  45                      2797                INC     BP                          ; MOVE TO FIRST CODE POINT
0FD4  E8 1EF6 R               2798                CALL    DO_MAP2                     ; STORE THE CODE POINT
0FD7  83 C5 0E                2799                ADD     BP,014D                     ; ADJUST BP TO NEXT CODE
0FDA  EB EA                   2800                JMP     FDG                         ; DO ANOTHER
0FDC                          2801        ENTRY_1:
0FDC  E8 0DAB R               2802                CALL    SET_REGS
0FDF  E8 0E55 R               2803                CALL    BLANK                       ; CLEAR OUT THE BUFFER
0FE2  E8 0E96 R               2804                CALL    PH_5
                              2805
                              2806                ASSUME  DS:ABS0
0FE5  E8 0CFE R               2807                CALL    DDS
0FE8  80 3E 0449 R 0F         2808                CMP     CRT_MODE,0FH
0FED  72 06                   2809                JB      MS_1
0FEF  C7 06 010C R 0000 E     2810                MOV     WORD PTR GRX_SET , OFFSET CGMN
0FF5                          2811        MS_1:
0FF5  80 3E 0449 R 07         2812                CMP     CRT_MODE,7
0FFA  77 09                   2813                JA      SAVE_GRPH
0FFC  74 4B                   2814                JE      SAVE_ALPH
0FFE  80 3E 0449 R 03         2815                CMP     CRT_MODE,3
1003  76 44                   2816                JBE     SAVE_ALPH
1005                          2817        SAVE_GRPH:
1005  C4 1E 04A8 R            2818                LES     BX,SAVE_PTR
1009  83 C3 0C                2819                ADD     BX,0CH
100C  26: C4 1F               2820                LES     BX,DWORD PTR ES:[BX]
100F  8C C0                   2821                MOV     AX,ES
1011  0B C3                   2822                OR      AX,BX
1013  74 32                   2823                JZ      J4J                         ; JMP AHO_DONE
1015  BE 0007                 2824                MOV     SI,07H
1018                          2825        SG_1:
1018  26: 8A 00               2826                MOV     AL,ES:[BX][SI]
101B  3C FF                   2827                CMP     AL,0FFH
101D  74 7A                   2828                JE      AHO_DONE
101F  3A 06 0449 R            2829                CMP     AL,CRT_MODE
1023  74 03                   2830                JE      SG_2
1025  46                      2831                INC     SI
1026  EB F0                   2832                JMP     SG_1
1028                          2833        SG_2:
1028  FA                      2834                CLI
1029  26: 8A 07               2835                MOV     AL,BYTE PTR ES:[BX]
102C  FE C8                   2836                DEC     AL
102E  A2 0484 R               2837                MOV     ROWS,AL
1031  26: 8B 47 01            2838                MOV     AX,WORD PTR ES:[BX][1]
1035  A3 0485 R               2839                MOV     POINTS,AX
1038  26: 8B 47 03            2840                MOV     AX,WORD PTR ES:[BX][3]
103C  A3 010C R               2841                MOV     WORD PTR GRX_SET,AX
103F  26: 8B 47 05            2842                MOV     AX,WORD PTR ES:[BX][5]
1043  A3 010E R               2843                MOV     WORD PTR GRX_SET + 2,AX
1046  FB                      2844                STI
1047                          2845        J4J:
1047  EB 50                   2846                JMP     SHORT AHO_DONE
1049                          2847        SAVE_ALPH:
1049  C4 1E 04A8 R            2848                LES     BX,SAVE_PTR
104D  83 C3 08                2849                ADD     BX,08H
1050  26: C4 1F               2850                LES     BX,DWORD PTR ES:[BX]
1053  8C C0                   2851                MOV     AX,ES
1055  0B C3                   2852                OR      AX,BX
1057  74 40                   2853                JZ      AHO_DONE
1059  BE 000B                 2854                MOV     SI,0BH
105C                          2855        SA_1:
105C  26: 8A 00               2856                MOV     AL,ES:[BX][SI]
105F  3C FF                   2857                CMP     AL,0FFH
1061  74 36                   2858                JE      AHO_DONE
1063  3A 06 0449 R            2859                CMP     AL,CRT_MODE
1067  74 03                   2860                JE      SA_2
1069  46                      2861                INC     SI
106A  EB F0                   2862                JMP     SA_1
106C                          2863        SA_2:
106C  26: 8A 27               2864                MOV     AH,ES:[BX]
106F  26: 8A 47 01            2865                MOV     AL,ES:[BX][1]
1073  26: 8B 4F 02            2866                MOV     CX,ES:[BX][2]
1077  26: 8B 57 04            2867                MOV     DX,ES:[BX][4]
107B  26: 8B 6F 06            2868                MOV     BP,ES:[BX][6]
107F  26: 8E 47 08            2869                MOV     ES,ES:[BX][8]
1083  53                      2870                PUSH    BX
1084  8B D8                   2871                MOV     BX,AX
1086  B8 1110                 2872                MOV     AX,1110H
1089  CD 10                   2873                INT     10H
108B  5B                      2874                POP     BX
108C  26: 8A 47 0A            2875                MOV     AL,ES:[BX][0AH]
1090  3C FF                   2876                CMP     AL,0FFH
1092  74 05                   2877                JE      AHO_DONE
1094  FE C8                   2878                DEC     AL
1096  A2 0484 R               2879                MOV     ROWS,AL
                              2880
                              2881        ;----- SET THE LOW RAM VALUES FOR COMPATIBILITY (3D8 AND 3D9 SAVE BYTES)
                              2882
1099                          2883        AHO_DONE:
1099  E8 0CFE R               2884                CALL    DDS
109C  80 3E 0449 R 07         2885                CMP     CRT_MODE,7
10A1  77 1E                   2886                JA      DNDCS
10A3  BB 10C8 R               2887                MOV     BX,OFFSET COMPAT_MODE
10A6  A0 0449 R               2888                MOV     AL,CRT_MODE
10A9  2A E4                   2889                SUB     AH,AH
10AB  03 D8                   2890                ADD     BX,AX
10AD  2E: 8A 07               2891                MOV     AL,CS:[BX]
10B0  A2 0465 R               2892                MOV     CRT_MODE_SET,AL
10B3  B0 30                   2893                MOV     AL,030H
10B5  80 3E 0449 R 06         2894                CMP     CRT_MODE,6
10BA  75 02                   2895                JNE     DO_PAL
10BC  B0 3F                   2896                MOV     AL,03FH
10BE                          2897        DO_PAL:
10BE  A2 0466 R               2898                MOV     CRT_PALETTE,AL
```

```
10C1                            2899            DNDCS:
10C1  8B 0E 0460 R              2900                    MOV     CX,CURSOR_MODE
10C5  EB 28 90                  2901                    JMP     AH1
                                2902
10C8                            2903            COMPAT_MODE     LABEL   BYTE
10C8  2C 28 2D 29 2A 2E         2904                    DB      02CH,028H,02DH,029H,02AH,02EH
10CE  1E 29                     2905                    DB      01EH,029H
                                2906
                                2907    C               INCLUDE V1-5.INC
                                2908    C               SUBTTL V1-5.INC
                                2909    C               PAGE
                                2910    C
10D0                            2911    C       CALC_CURSOR     PROC    NEAR
                                2912    C                       ASSUME  DS:ABS0
10D0  80 FD 00                  2913    C                       CMP     CH,0                    ; CHECK FOR FULL HEIGHT
10D3  75 04                     2914    C                       JNE     CC_1                    ; NORMAL CHECK
10D5  FE C1                     2915    C                       INC     CL                      ; ADJUST END VALUE
10D7  EB 0A                     2916    C                       JMP     SHORT   CALC_OUT
10D9                            2917    C       CC_1:
10D9  FE C1                     2918    C                       INC     CL                      ; ADJUST FOR EGA REGISTERS
10DB  3A 0E 0485 R              2919    C                       CMP     CL,BYTE PTR POINTS      ; WILL IT WRAP
10DF  72 02                     2920    C                       JB      CALC_OUT                ; NO, ITS OK
10E1  2A C9                     2921    C                       SUB     CL,CL                   ; EGA METHOD FOR CURSOR END
10E3                            2922    C       CALC_OUT:
10E3  51                        2923    C                       PUSH    CX                      ; SAVE CURSOR TYPE VALUE
10E4  2A CD                     2924    C                       SUB     CL,CH                   ; END - START
10E6  80 F9 10                  2925    C                       CMP     CL,010H                 ; LOW NIBBLE EQUAL
10E9  59                        2926    C                       POP     CX                      ; RESTORE
10EA  75 02                     2927    C                       JNE     COMP_4
10EC  FE C1                     2928    C                       INC     CL                      ; ADD 1 FOR CORRECT CURSOR
10EE                            2929    C       COMP_4:
10EE  C3                        2930    C                       RET                             ; BACK TO CALLER
10EF                            2931    C       CALC_CURSOR     ENDP
                                2932    C
                                2933    C       ;-----------------------------------------------------------------
                                2934    C       ; SET_CTYPE     SET CURSOR TYPE                                 :
                                2935    C       ;       THIS ROUTINE SETS THE CURSOR VALUE                      :
                                2936    C       ; INPUT                                                         :
                                2937    C       ;       (CX) HAS CURSOR VALUE CH-START LINE, CL-STOP LINE       :
                                2938    C       ; OUTPUT                                                        :
                                2939    C       ;       NONE                                                    :
                                2940    C       ;-----------------------------------------------------------------
= 0004                          2941    C               CUT_OFF         EQU     4
10EF                            2942    C       AH1:
                                2943    C                       ASSUME  DS:ABS0
10EF  B4 0A                     2944    C                       MOV     AH,C_CRSR_START         ; CRTC REG FOR CURSOR SET
10F1  89 0E 0460 R              2945    C                       MOV     CURSOR_MODE,CX          ; SAVE IN DATA AREA
10F5  F6 06 0487 R 08           2946    C                       TEST    INFO,8                  ; EGA ACTIVE BIT
10FA  75 33                     2947    C                       JNZ     DO_SET                  ; 0=EGA, 1=OLD CARDS
                                2948    C
                                2949    C       ;----- THIS SECTION WILL EMULATE CURSOR OFF ON THE EGA
                                2950    C
10FC  8A C5                     2951    C                       MOV     AL,CH                   ; GET START VALUE
10FE  24 60                     2952    C                       AND     AL,060H                 ; TURN OFF CURSOR ?
1100  3C 20                     2953    C                       CMP     AL,020H                 ; TEST THE BITS
1102  75 05                     2954    C                       JNE     AH1_A                   ; SKIP CURSOR OFF
1104  B9 1E00                   2955    C                       MOV     CX,01E00H               ; EMULATE CURSOR OFF
1107  EB 26                     2956    C                       JMP     SHORT DO_SET
                                2957    C
                                2958    C       ;----- THIS SECTION : ADJUST THE CURSOR AND TEST FOR ENHANCED OPERATION
                                2959    C
1109                            2960    C       AH1_A:
1109  F6 06 0487 R 01           2961    C                       TEST    INFO,1                  ; CURSOR EMULATE BIT
110E  75 1F                     2962    C                       JNZ     DO_SET                  ; 0=EMULATE, 1=VALUE AS-IS
1110  80 3E 0449 R 03           2963    C                       CMP     CRT_MODE,3              ; POSSIBLE EMULATION
1115  77 15                     2964    C                       JA      AH1_S                   ; NO, SET THE CURSOR TYPE
1117  E8 0E9A R                 2965    C                       CALL    BRST_DET                ; SEE IF EMULATE MODE
111A  73 10                     2966    C                       JNC     AH1_S                   ; NOT EMULATING
111C  80 FD 04                  2967    C                       CMP     CH,CUT_OFF              ; TEST START
111F  76 03                     2968    C                       JBE     AH1_B                   ; SKIP ADJUST
1121  80 C5 05                  2969    C                       ADD     CH,5                    ; ADJUST
1124                            2970    C       AH1_B:
1124  80 F9 04                  2971    C                       CMP     CL,CUT_OFF              ; TEST END
1127  76 03                     2972    C                       JBE     AH1_S                   ; SKIP ADJUST
1129  80 C1 05                  2973    C                       ADD     CL,5
112C                            2974    C       AH1_S:
112C  E8 10D0 R                 2975    C                       CALL    CALC_CURSOR             ; ADJUST END REGISTER
112F                            2976    C       DO_SET:
112F  E8 1135 R                 2977    C                       CALL    M16                     ; OUTPUT CX REG
1132  E9 219E R                 2978    C                       JMP     V_RET                   ; RETURN TO CALLER
                                2979    C
                                2980    C       ;----- THIS ROUTINE OUTPUTS THE CX REGISTER TO THE CRTC REGS NAMED IN AH
                                2981    C
1135                            2982    C       M16:
1135  8B 16 0463 R              2983    C                       MOV     DX,ADDR_6845            ; ADDRESS REGISTER
1139  8A C5                     2984    C                       MOV     AL,CH                   ; DATA
113B  E8 0D15 R                 2985    C                       CALL    OUT_DX                  ; OUTPUT THE VALUE
113E  FE C4                     2986    C                       INC     AH                      ; NEXT REGISTER
1140  8A C1                     2987    C                       MOV     AL,CL                   ; SECOND DATA VALUE
1142  E8 0D15 R                 2988    C                       CALL    OUT_DX                  ; OUTPUT THE VALUE
1145  C3                        2989    C                       RET                             ; ALL DONE
                                2990    C
                                2991    C       ;-----------------------------------------------------------------
                                2992    C       ; POSITION                                                      :
                                2993    C       ;       THIS SERVICE ROUTINE CALCULATES THE REGEN BUFFER         :
                                2994    C       ;       ADDRESS OF A CHARACTER IN THE ALPHA MODE                 :
                                2995    C       ; INPUT                                                         :
                                2996    C       ;       AX = ROW, COLUMN POSITION                               :
                                2997    C       ; OUTPUT                                                        :
                                2998    C       ;       AX = OFFSET OF CHAR POSITION IN REGEN BUFFER             :
                                2999    C       ;-----------------------------------------------------------------
1146                            3000    C       POSITION        PROC    NEAR
1146  53                        3001    C                       PUSH    BX                      ; SAVE REGISTER
1147  8B D8                     3002    C                       MOV     BX,AX
1149  8A C4                     3003    C                       MOV     AL,AH                   ; ROWS TO AL
114B  F6 26 044A R              3004    C                       MUL     BYTE PTR CRT_COLS       ; DETERMINE BYTES TO ROW
114F  32 FF                     3005    C                       XOR     BH,BH                   ; ZERO OUT
1151  03 C3                     3006    C                       ADD     AX,BX                   ; ADD IN COLUMN VALUE
1153  D1 E0                     3007    C                       SAL     AX,1                    ; * 2 FOR ATTRIBUTE BYTES
1155  5B                        3008    C                       POP     BX                      ; RESTORE REGISTER
1156  C3                        3009    C                       RET
1157                            3010    C       POSITION        ENDP
                                3011    C
                                3012    C       ;-----------------------------------------------------------------
                                3013    C       ; SET_CPOS      SET CURSOR POSITION                             :
                                3014    C       ;       THIS ROUTINE SETS THE CURRENT CURSOR POSITION TO THE     :
                                3015    C       ;       NEW X-Y VALUES PASSED                                   :
                                3016    C       ; INPUT                                                         :
                                3017    C       ;       DX - ROW,COLUMN OF NEW CURSOR                           :
                                3018    C       ;       BH - DISPLAY PAGE OF CURSOR                             :
                                3019    C       ; OUTPUT                                                        :
                                3020    C       ;       CURSOR IS SET AT CRTC IF DISPLAY PAGE IS CURRENT        :
                                3021    C       ;       DISPLAY                                                 :
                                3022    C       ;-----------------------------------------------------------------
1157                            3023    C       AH2:
1157  E8 115D R                 3024    C                       CALL    SET_CPOS
```

```
115A  E9 219E R                 3025  C               JMP     V_RET
                                3026  C
115D                            3027  C   SET_CPOS:
115D  8A CF                     3028  C               MOV     CL,BH
115F  32 ED                     3029  C               XOR     CH,CH                           ; ESTABLISH LOOP COUNT
1161  D1 E1                     3030  C               SAL     CX,1                            ; WORD OFFSET
1163  8B F1                     3031  C               MOV     SI,CX                           ; USE INDEX REGISTER
1165  89 94 0450 R              3032  C               MOV     [SI+OFFSET CURSOR_POSN],DX      ; SAVE THE POINTER
1169  38 3E 0462 R              3033  C               CMP     ACTIVE_PAGE,BH
116D  75 05                     3034  C               JNZ     M17                             ; SET_CPOS_RETURN
116F  8B C2                     3035  C               MOV     AX,DX                           ; GET ROW/COLUMN TO AX
1171  E8 1175 R                 3036  C               CALL    M18                             ; CURSOR_SET
1174                            3037  C   M17:                                              ; SET_CPOS_RETURN
1174  C3                        3038  C               RET
                                3039  C
                                3040  C   ;----- SET CURSOR POSITION, AX HAS ROW/COLUMN FOR CURSOR
                                3041  C
1175                            3042  C   M18         PROC    NEAR
1175  E8 1146 R                 3043  C               CALL    POSITION                        ; DETERMINE LOC IN REGEN
1178  8B C8                     3044  C               MOV     CX,AX
117A  03 0E 044E R              3045  C               ADD     CX,CRT_START                    ; ADD IN THE START ADDR
                                3046  C                                                       ;  FOR THIS PAGE
117E  D1 F9                     3047  C               SAR     CX,1                            ; / 2 FOR CHAR ONLY COUNT
1180  B4 0E                     3048  C               MOV     AH,C_CRSR_LOC_HGH               ; REGISTER NUMBER FOR CURSOR
1182  E8 1135 R                 3049  C               CALL    M16                             ; SET VALUE TO CRTC
1185  C3                        3050  C               RET
1186                            3051  C   M18         ENDP
                                3052  C
                                3053  C   ;--------------------------------------------------------------
                                3054  C   ; READ_CURSOR                                              :
                                3055  C   ;       THIS ROUTINE READS THE CURRENT CURSOR VALUE FROM      :
                                3056  C   ;       MEMORY AND SENDS IT BACK TO THE CALLER                :
                                3057  C   ; INPUT                                                     :
                                3058  C   ;       BH = PAGE OF CURSOR                                   :
                                3059  C   ; OUTPUT                                                    :
                                3060  C   ;       DX = ROW, COLUMN OF THE CURRENT CURSOR POSITION       :
                                3061  C   ;       CX = CURRENT CURSOR MODE                              :
                                3062  C   ;--------------------------------------------------------------
1186                            3063  C   AH3:
1186  8A DF                     3064  C               MOV     BL,BH                           ; PAGE VALUE
1188  32 FF                     3065  C               XOR     BH,BH                           ; ZERO UPPER BYTE
118A  D1 E3                     3066  C               SAL     BX,1                            ; WORD OFFSET
118C  8B 97 0450 R              3067  C               MOV     DX,[BX + OFFSET CURSOR_POSN]    ; GET CURSOR FOR THIS PAGE
1190  8B 0E 0460 R              3068  C               MOV     CX,CURSOR_MODE                  ; GET THE CURSOR MODE
1194  5F                        3069  C               POP     DI
1195  5E                        3070  C               POP     SI
1196  5B                        3071  C               POP     BX
1197  58                        3072  C               POP     AX                              ; DISCARD CX
1198  58                        3073  C               POP     AX                              ; DISCARD DX
1199  1F                        3074  C               POP     DS
119A  07                        3075  C               POP     ES
119B  5D                        3076  C               POP     BP
119C  CF                        3077  C               IRET
                                3078  C
                                3079  C   ;----- READ LIGHT PEN POSITION
                                3080  C
119D                            3081  C   AH4:
119D  A0 0449 R                 3082  C               MOV     AL,CRT_MODE
11A0  3C 07                     3083  C               CMP     AL,07H
11A2  77 37                     3084  C               JA      READ_LPEN
                                3085  C
11A4  F6 06 0487 R 02           3086  C               TEST    INFO,2
11A9  74 07                     3087  C               JZ      EGA_IS_COLOR
                                3088  C
                                3089  C   ;----- MONOCHROME HERE (MONOC BIT 1)
                                3090  C
11AB  3C 07                     3091  C               CMP     AL,07H
11AD  74 2C                     3092  C               JE      READ_LPEN
11AF  EB 05 90                  3093  C               JMP     OLD_LP
                                3094  C
                                3095  C   ;----- EGA IS COLOR HERE (MONOC BIT 0)
                                3096  C
11B2                            3097  C   EGA_IS_COLOR:
11B2  3C 06                     3098  C               CMP     AL,06H
11B4  76 25                     3099  C               JBE     READ_LPEN
11B6                            3100  C   OLD_LP:
11B6  CD 42                     3101  C               INT     42H                             ; CALL EXISTING CODE
11B8  5F                        3102  C               POP     DI
11B9  5E                        3103  C               POP     SI
11BA  83 C4 06                  3104  C               ADD     SP,6                            ; DISCARD SAVED BX,CX,DX
11BD  1F                        3105  C               POP     DS
11BE  07                        3106  C               POP     ES
11BF  5D                        3107  C               POP     BP
11C0  CF                        3108  C               IRET
                                3109  C
                                3110  C   ;--------------------------------------------------------------
                                3111  C   ; LIGHT PEN                                                 :
                                3112  C   ;       THIS ROUTINE TESTS THE LIGHT PEN SWITCH AND THE LIGHT :
                                3113  C   ;       PEN TRIGGER. IF BOTH ARE SET, THE LOCATION OF THE LIGHT :
                                3114  C   ;       PEN IS DETERMINED. OTHERWISE, A RETURN WITH NO        :
                                3115  C   ;       INFORMATION IS MADE.                                  :
                                3116  C   ; ON EXIT                                                   :
                                3117  C   ;       (AH) = 0 IF NO LIGHT PEN INFORMATION IS AVAILABLE     :
                                3118  C   ;                BX,CX,DX ARE DESTROYED                       :
                                3119  C   ;       (AH) = 1 IF LIGHT PEN IS AVAILABLE                    :
                                3120  C   ;                (DH,DL) = ROW,COLUMN OF CURRENT LIGHT PEN    :
                                3121  C   ;                          POSITION                           :
                                3122  C   ;                (CH) = RASTER POSITION (OLD MODES)           :
                                3123  C   ;                (CX) = RASTER POSITION (NEW MODES)           :
                                3124  C   ;                (BX) = BEST GUESS AT PIXEL HORIZONTAL POSITION :
                                3125  C   ;--------------------------------------------------------------
                                3126  C               ASSUME  CS:CODE,DS:ABS0
                                3127  C   ;----- SUBTRACT_TABLE
11C1                            3128  C   V1          LABEL   BYTE
11C1  06 06 07 07 05 05         3129  C               DB      006H,006H,007H,007H,005H,005H   ; 0-5
11C7  04 05 00 00 00 00         3130  C               DB      004H,005H,000H,000H,000H,000H   ; 6-B
11CD  00 05 06 04 04 04         3131  C               DB      000H,005H,006H,004H,004H,004H   ; C-11
11D3  04 06 06 04 07 04         3132  C               DB      004H,006H,006H,004H,007H,004H   ; 12-17
11D9  07 04                     3133  C               DB      007H,004H                       ; 18-19
                                3134  C
11DB                            3135  C   READ_LPEN       PROC    NEAR
                                3136  C
                                3137  C   ;----- WAIT FOR LIGHT PEN TO BE DEPRESSED
                                3138  C
11DB  8B 16 0463 R              3139  C               MOV     DX,ADDR_6845                    ; GET BASE ADDRESS OF 6845
11DF  83 C2 06                  3140  C               ADD     DX,6                            ; POINT TO STATUS REGISTER
11E2  EC                        3141  C               IN      AL,DX                           ; GET STATUS REGISTER
11E3  A8 04                     3142  C               TEST    AL,4                            ; TEST LIGHT PEN SWITCH
11E5  B4 00                     3143  C               MOV     AH,0                            ; SET NO LIGHT PEN RETURN
11E7  74 03                     3144  C               JZ      V9                              ;  CODE
11E9  E9 1291 R                 3145  C               JMP     V6                              ; NOT SET, RETURN
                                3146  C
                                3147  C   ;----- NOW TEST FOR LIGHT PEN TRIGGER
                                3148  C
11EC                            3149  C   V9:
11EC  A8 02                     3150  C               TEST    AL,2                            ; TEST LIGHT PEN TRIGGER
```

```
11EE  75 03                   3151  C                JNZ     V7A                     ; RETURN WITHOUT RESETTING
                              3152  C                                                ;  TRIGGER
11F0  E9 129B R               3153  C                JMP     V7                      ; EXIT LIGHT PEN ROUTINE
                              3154  C
                              3155  C  ;----- TRIGGER HAS BEEN SET, READ THE VALUE IN
                              3156  C
11F3                          3157  C  V7A:
11F3  B4 10                   3158  C                MOV     AH,16                   ; LIGHT PEN REGISTERS
                              3159  C
                              3160  C  ;----- INPUT REGS POINTED TO BY AH, AND CONVERT TO ROW COLUMN IN DX
                              3161  C
11F5  8B 16 0463 R            3162  C                MOV     DX,ADDR_6845            ; ADDRESS REGISTER
11F9  8A C4                   3163  C                MOV     AL,AH                   ; REGISTER TO READ
11FB  EE                      3164  C                OUT     DX,AL                   ; SET IT UP
11FC  42                      3165  C                INC     DX                      ; DATA REGISTER
11FD  50                      3166  C                PUSH    AX
11FE  EC                      3167  C                IN      AL,DX                   ; GET THE VALUE
11FF  8A E8                   3168  C                MOV     CH,AL                   ; SAVE IN CX
1201  58                      3169  C                POP     AX
1202  4A                      3170  C                DEC     DX                      ; ADDRESS REGISTER
1203  FE C4                   3171  C                INC     AH
1205  8A C4                   3172  C                MOV     AL,AH                   ; SECOND DATA REGISTER
1207  EE                      3173  C                OUT     DX,AL
1208  42                      3174  C                INC     DX                      ; POINT TO DATA REGISTER
1209  EC                      3175  C                IN      AL,DX                   ; GET THE 2ND DATA VALUE
120A  8A E5                   3176  C                MOV     AH,CH                   ; AX HAS INPUT VALUE
                              3177  C
                              3178  C  ;----- AX HAS THE VALUE READ IN FROM THE 6845
                              3179  C
120C  8A 1E 0449 R            3180  C                MOV     BL,CRT_MODE
1210  2A FF                   3181  C                SUB     BH,BH                   ; MODE VALUE TO BX
1212  2E: 8A 9F 11C1 R        3182  C                MOV     BL,CS:V1[BX]            ; AMOUNT TO SUBTRACT
1217  2B C3                   3183  C                SUB     AX,BX                   ; TAKE IT AWAY
1219  8B 1E 044E R            3184  C                MOV     BX,CRT_START            ; SCREEN ADDRESS
121D  D1 EB                   3185  C                SHR     BX,1                    ; DIVIDE BY 2
121F  2B C3                   3186  C                SUB     AX,BX                   ; ADJUST TO ZERO START
1221  79 02                   3187  C                JNS     V2                      ; IF POSITIVE, GET MODE
1223  2B C0                   3188  C                SUB     AX,AX                   ; <0 PLAYS AS 0
                              3189  C
                              3190  C  ;----- DETERMINE MODE OF OPERATION
                              3191  C
1225                          3192  C  V2:                                           ; DETERMINE_MODE
1225  B1 03                   3193  C                MOV     CL,3                    ; SET *8 SHIFT COUNT
1227  80 3E 0449 R 04         3194  C                CMP     CRT_MODE,4              ; GRAPHICS OR ALPHA
122C  72 4D                   3195  C                JB      V4                      ; ALPHA_PEN
122E  80 3E 0449 R 07         3196  C                CMP     CRT_MODE,7
1233  74 46                   3197  C                JE      V4                      ; ALPHA_PEN
                              3198  C
1235  80 3E 0449 R 06         3199  C                CMP     CRT_MODE,06H
123A  77 28                   3200  C                JA      V8
123C  75 02                   3201  C                JNE     V8X
123E  D1 E8                   3202  C                SHR     AX,1
                              3203  C
                              3204  C  ;----- DLD GRAPHICS MODES
                              3205  C
1240                          3206  C  V8X:
1240  B2 28                   3207  C                MOV     DL,40                   ; DIVISOR FOR GRAPHICS
1242  F6 F2                   3208  C                DIV     DL                      ; ROW(AL) AND COLUMN(AH)
                              3209  C                                                ;  AL RANGE 0-99,
                              3210  C                                                ;  AH RANGE 0-39
                              3211  C  ;----- DETERMINE GRAPHIC ROW POSITION
                              3212  C
1244  8A E8                   3213  C                MOV     CH,AL                   ; SAVE ROW VALUE IN CH
1246  02 ED                   3214  C                ADD     CH,CH                   ; *2 FOR EVEN/ODD FIELD
1248  8A DC                   3215  C                MOV     BL,AH                   ; COLUMN VALUE TO BX
124A  2A FF                   3216  C                SUB     BH,BH                   ; *8 FOR MEDIUM RES
124C  80 3E 0449 R 06         3217  C                CMP     CRT_MODE,6              ; MEDIUM OR HIGH RES
1251  75 04                   3218  C                JNE     V3                      ; NOT_HIGH_RES
1253  B1 04                   3219  C                MOV     CL,4                    ; SHIFT VALUE FOR HIGH RES
1255  D0 E4                   3220  C                SAL     AH,1                    ; COLUMN VALUE *2 FOR HIGH RES
1257                          3221  G  V3:                                           ; NOT_HIGH_RES
1257  D3 E3                   3222  C                SHL     BX,CL                   ; *16 FOR HIGH RES
                              3223  C
                              3224  C  ;----- DETERMINE ALPHA CHAR POSITION
                              3225  C
1259  8A D4                   3226  C                MOV     DL,AH                   ; COLUMN VALUE FOR RETURN
125B  8A F0                   3227  C                MOV     DH,AL                   ; ROW VALUE
125D  D0 EE                   3228  C                SHR     DH,1                    ; DIVIDE BY 4
125F  D0 EE                   3229  C                SHR     DH,1                    ;  FOR VALUE IN 0-24 RANGE
1261  EB 2C 90                3230  C                JMP     V5                      ; LIGHT_PEN_RETURN_SET
1264                          3231  C  V8:
                              3232  C
                              3233  C  ;----- NEW GRAPHICS MODES
                              3234  C
1264  99                      3235  C                CWD                             ; PREPARE TO DIVIDE
1265  F7 36 044A R            3236  C                DIV     CRT_COLS                ; AX = ROW, DX = COLUMN
1269  8B DA                   3237  C                MOV     BX,DX                   ; SAVE REMAINDER
126B  D3 E3                   3238  C                SAL     BX,CL                   ; PEL COLUMN
126D  8B C8                   3239  C                MOV     CX,AX                   ; PEL ROW
126F  52                      3240  C                PUSH    DX                      ; SAVE FROM DIVIDE
1270  99                      3241  C                CWD                             ; PREPARE TO DIVIDE
1271  F7 36 0485 R            3242  C                DIV     POINTS                  ; DIVIDE BY BYTES/CHAR
1275  5A                      3243  C                POP     DX                      ; RECOVER
1276  8A F0                   3244  C                MOV     DH,AL                   ; CHARACTER ROW
1278  EB 15 90                3245  C                JMP     V5
                              3246  C
                              3247  C  ;----- ALPHA MODE ON LIGHT PEN
                              3248  C
127B                          3249  C  V4:                                           ; ALPHA_PEN
127B  F6 36 044A R            3250  C                DIV     BYTE PTR CRT_COLS       ; ROW,COLUMN VALUE
127F  8A F0                   3251  C                MOV     DH,AL                   ; ROWS TO DH
1281  8A D4                   3252  C                MOV     DL,AH                   ; COLS TO DL
1283  8A DC                   3253  C                MOV     BL,AH                   ; COLUMN VALUE
1285  32 FF                   3254  C                XOR     BH,BH                   ;  TO BX
1287  D3 E3                   3255  C                SAL     BX,CL
1289  F6 26 0485 R            3256  C                MUL     BYTE PTR POINTS
128D  8B C8                   3257  C                MOV     CX,AX
128F                          3258  C  V5:                                           ; LIGHT_PEN_RETURN_SET
128F  B4 01                   3259  C                MOV     AH,1                    ; INDICATE EVERTHING SET
1291                          3260  C  V6:                                           ; LIGHT_PEN_RETURN
1291  52                      3261  C                PUSH    DX                      ; SAVE RETURN VALUE
                              3262  C                                                ;  (IN CASE)
1292  8B 16 0463 R            3263  C                MOV     DX,ADDR_6845            ; GET BASE ADDRESS
1296  83 C2 07                3264  C                ADD     DX,7                    ; POINT TO RESET PARM
1299  EE                      3265  C                OUT     DX,AL                   ; ADDRESS, NOT DATA,
                              3266  C                                                ;  IS IMPORTANT
129A  5A                      3267  C                POP     DX                      ; RECOVER VALUE
129B                          3268  C  V7:                                           ; RETURN_NO_RESET
129B  5F                      3269  C                POP     DI
129C  5E                      3270  C                POP     SI
129D  83 C4 06                3271  C                ADD     SP,6                    ; DISCARD SAVED BX,CX,DX
12A0  1F                      3272  C                POP     DS
12A1  07                      3273  C                POP     ES
12A2  5D                      3274  C                POP     BP
12A3  CF                      3275  C                IRET
12A4                          3276  C  READ_LPEN     ENDP
```

```
                        3277    C
                        3278    C ;--------------------------------------------------------------------
                        3279    C ; ACT_DISP_PAGE           SELECT ACTIVE DISPLAY PAGE                  :
                        3280    C ;       THIS ROUTINE SETS THE ACTIVE DISPLAY PAGE, ALLOWING       :
                        3281    C ;       FOR MULTIPLE PAGES OF DISPLAYED VIDEO.                    :
                        3282    C ; INPUT                                                           :
                        3283    C ;       AL HAS THE NEW ACTIVE DISPLAY PAGE                        :
                        3284    C ; OUTPUT                                                          :
                        3285    C ;       THE CRTC IS RESET TO DISPLAY THAT PAGE                    :
                        3286    C ;--------------------------------------------------------------------
 12A4                   3287    C AH5:
 12A4  A2 0462 R        3288    C           MOV     ACTIVE_PAGE,AL                  ; SAVE ACTIVE PAGE VALUE
 12A7  8B 0E 044C R     3289    C           MOV     CX,CRT_LEN                      ; GET SAVED LENGTH OF
                        3290    C                                                   ;  REGEN BUFFER
 12AB  98               3291    C           CBW                                     ; CONVERT AL TO WORD
 12AC  50               3292    C           PUSH    AX                              ; SAVE PAGE VALUE
 12AD  F7 E1            3293    C           MUL     CX                              ; DISPLAY PAGE TIMES
                        3294    C                                                   ;  REGEN LENGTH
 12AF  A3 044E R        3295    C           MOV     CRT_START,AX                    ; SAVE START ADDRESS FOR
                        3296    C                                                   ;  LATER REQUIREMENTS
 12B2  8B C8            3297    C           MOV     CX,AX                           ; START ADDRESS TO CX
 12B4  8A 1E 0449 R     3298    C           MOV     BL,CRT_MODE
 12B8  80 FB 07         3299    C           CMP     BL,7                            ; DO NOT DIVIDE BY TWO
 12BB  77 02            3300    C           JA      ADP_1
 12BD                   3301    C ADP_2:
 12BD  D1 F9            3302    C           SAR     CX,1                            ; / 2 FOR CRTC HANDLING
 12BF                   3303    C ADP_1:
 12BF  B4 0C            3304    C           MOV     AH,C_STRT_HGH                   ; REG FOR START ADDRESS
 12C1  E8 1135 R        3305    C           CALL    M16
 12C4  5B               3306    C           POP     BX                              ; RECOVER PAGE VALUE
 12C5  D1 E3            3307    C           SAL     BX,1                            ; *2 FOR WORD OFFSET
 12C7  8B 87 0450 R     3308    C           MOV     AX,[BX + OFFSET CURSOR_POSN]    ; GET CURSOR FOR THIS PAGE
 12CB  E8 1175 R        3309    C           CALL    M18                             ; SET THE CURSOR POSITION
 12CE  E9 219E R        3310    C           JMP     V_RET
                        3311    C
                        3312    C           SUBTTL
                        3313
                        3314    C           INCLUDE VSCROLL.INC
                        3315    C           SUBTTL  VSCROLL.INC
                        3316    C           PAGE
                        3317    C
 12D1                   3318    C FLTA      PROC    NEAR                            ; CHECK FOR SCROLL COUNT
 12D1  50               3319    C           PUSH    AX
 12D2  8A E6            3320    C           MOV     AH,DH                           ; LOWER ROW
 12D4  2A E5            3321    C           SUB     AH,CH                           ; UPPER ROW
 12D6  FE C4            3322    C           INC     AH                              ; NUMBER TO SCROLL
 12D8  3A E0            3323    C           CMP     AH,AL                           ; SAME AS REQUESTED
 12DA  58               3324    C           POP     AX
 12DB  75 02            3325    C           JNE     LTA
 12DD  2A C0            3326    C           SUB     AL,AL                           ; YES, SET TO 0 FOR BLANK
 12DF                   3327    C LTA:
 12DF  C3               3328    C           RET
 12E0                   3329    C FLTA      ENDP
                        3330    C
 12E0                   3331    C CRANK     PROC    NEAR                            ; MOVE ROWS OF PELS UP
 12E0  53               3332    C           PUSH    BX
                        3333    C           ASSUME  DS:ABS0
 12E1  1E               3334    C           PUSH    DS                              ; SAVE DATA SEGMENT
 12E2  E8 0CFE R        3335    C           CALL    DDS                             ; SET DATA SEGMENT
 12E5  8B 1E 044A R     3336    C           MOV     BX,CRT_COLS
 12E9  1F               3337    C           POP     DS
 12EA                   3338    C CRANK_A:
 12EA  51               3339    C           PUSH    CX                              ; SAVE MOVE COUNT
 12EB  8A CA            3340    C           MOV     CL,DL                           ; COLUMN COUNT
 12ED  2A ED            3341    C           SUB     CH,CH                           ; CLEAR HIGH BYTE
 12EF  56               3342    C           PUSH    SI                              ; SAVE POINTERS
 12F0  57               3343    C           PUSH    DI
 12F1  F3/ A4           3344    C           REP     MOVSB                           ; MOVE THAT ROW
 12F3  5F               3345    C           POP     DI                              ; RECOVER POINTERS
 12F4  5E               3346    C           POP     SI
 12F5  03 F3            3347    C           ADD     SI,BX                           ; NEXT ROW
 12F7  03 FB            3348    C           ADD     DI,BX                           ; NEXT ROW
 12F9  59               3349    C           POP     CX                              ; RECOVER ROW COUNT
 12FA  E2 EE            3350    C           LOOP    CRANK_A                         ; DO MORE
 12FC  5B               3351    C           POP     BX
 12FD  C3               3352    C           RET                                     ; RETURN TO CALLER
 12FE                   3353    C CRANK     ENDP
                        3354    C
 12FE                   3355    C CRANK_4   PROC    NEAR                            ; MOVE ROWS OF PELS DOWN
 12FE  53               3356    C           PUSH    BX
                        3357    C           ASSUME  DS:ABS0
 12FF  1E               3358    C           PUSH    DS                              ; SAVE DATA SEGMENT
 1300  E8 0CFE R        3359    C           CALL    DDS                             ; SET DATA SEGMENT
 1303  8B 1E 044A R     3360    C           MOV     BX,CRT_COLS
 1307  1F               3361    C           POP     DS
 1308                   3362    C CRANK_B:
 1308  51               3363    C           PUSH    CX                              ; SAVE MOVE COUNT
 1309  8A CA            3364    C           MOV     CL,DL                           ; COLUMN COUNT
 130B  2A ED            3365    C           SUB     CH,CH                           ; CLEAR HIGH BYTE
 130D  56               3366    C           PUSH    SI                              ; SAVE POINTERS
 130E  57               3367    C           PUSH    DI
 130F  F3/ A4           3368    C           REP     MOVSB                           ; MOVE THAT ROW
 1311  5F               3369    C           POP     DI                              ; RECOVER POINTERS
 1312  5E               3370    C           POP     SI
 1313  2B F3            3371    C           SUB     SI,BX                           ; NEXT ROW
 1315  2B FB            3372    C           SUB     DI,BX                           ; NEXT ROW
 1317  59               3373    C           POP     CX                              ; RECOVER ROW COUNT
 1318  E2 EE            3374    C           LOOP    CRANK_B                         ; DO MORE
 131A  5B               3375    C           POP     BX
 131B  C3               3376    C           RET                                     ; RETURN TO CALLER
 131C                   3377    C CRANK_4   ENDP
                        3378    C
 131C                   3379    C PART_1    PROC    NEAR                            ; FILL ROW AFTER SCROLL
 131C  52               3380    C           PUSH    DX
 131D  B6 03            3381    C           MOV     DH,3
 131F  B2 C4            3382    C           MOV     DL,SEQ_ADDR                     ; SEQUENCER
 1321  B8 020F          3383    C           MOV     AX,020FH                        ; MAP MASK
 1324  E8 0D15 R        3384    C           CALL    OUT_DX                          ; ALL MAPS ON
 1327  5A               3385    C           POP     DX
 1328  2B C0            3386    C           SUB     AX,AX                           ; ZERO
 132A  8A CA            3387    C           MOV     CL,DL                           ; COLUMN COUNT
 132C  2A ED            3388    C           SUB     CH,CH
 132E  57               3389    C           PUSH    DI                              ; SAVE POINTER
 132F  F3/ AA           3390    C           REP     STOSB                           ; CLEAR ONE ROW OF PELS
 1331  5F               3391    C           POP     DI                              ; RECOVER POINTER
 1332  8A C6            3392    C           MOV     AL,DH                           ; GET COLOR VALUE
 1334  52               3393    C           PUSH    DX
 1335  B6 03            3394    C           MOV     DH,3
 1337  B2 C4            3395    C           MOV     DL,SEQ_ADDR                     ; SEQUENCER
 1339  B4 02            3396    C           MOV     AH,02H                          ; MAP MASK
 133B  E8 0D15 R        3397    C           CALL    OUT_DX                          ; SET THE COLOR
 133E  5A               3398    C           POP     DX
 133F  B0 FF            3399    C           MOV     AL,0FFH                         ; ALL BITS ON
 1341  8A CA            3400    C           MOV     CL,DL                           ; COLUMN COUNT
 1343  57               3401    C           PUSH    DI                              ; SAVE POINTER
 1344  F3/ AA           3402    C           REP     STOSB                           ; TURN ON THOSE BITS IN
```

```
                        3403   C                                       ;  ENABLED PLANES
1346  5F                3404   C               POP     DI              ; RECOVER POINTER
1347  C3                3405   C               RET                     ; RETURN TO CALLER
1348                    3406   C  PART_1       ENDP
                        3407   C
1348                    3408   C  PART_2       PROC    NEAR
1348  B6 03             3409   C               MOV     DH,3
134A  B2 C4             3410   C               MOV     DL,SEQ_ADDR             ; SEQUENCER
134C  B8 020F           3411   C               MOV     AX,020FH                ; MAP MASK, ALL MAPS
134F  E8 0D15 R         3412   C               CALL    OUT_DX                  ; ENABLE THE MAPS
1352  C3                3413   C               RET                             ; RETURN TO CALLER
1353                    3414   C  PART_2       ENDP
                        3415   C
1353                    3416   C  BLNK_3       PROC    NEAR                    ; BLANK FOR SCROLL UP
1353  1E                3417   C               PUSH    DS                      ; SAVE DATA SEGMENT
                        3418   C               ASSUME  DS:ABS0
1354  E8 0CFE R         3419   C               CALL    DDS                     ; GET LOW MEMORY SEGMENT
1357  8A F7             3420   C               MOV     DH,BH                   ; ATTRIBUTE FOR BLANK LINE
1359  2A FF             3421   C               SUB     BH,BH                   ; CLEAR HIGH BYTE
135B  50                3422   C               PUSH    AX                      ; SAVE
135C  52                3423   C               PUSH    DX                      ; SAVE BECAUSE OF MULTIPLY
135D  8B C3             3424   C               MOV     AX,BX                   ; ROW COUNT
135F  F7 26 0485 R      3425   C               MUL     POINTS                  ; CHARACTER HEIGHT
1363  8B D8             3426   C               MOV     BX,AX                   ; NET VALUE TO BX
1365  5A                3427   C               POP     DX                      ; RECOVER
1366  58                3428   C               POP     AX
                        3429   C
1367  1F                3430   C               POP     DS
                        3431   C               ASSUME  DS:NOTHING
1368                    3432   C  S13:
1368  E8 131C R         3433   C               CALL    PART_1                  ; BLANK OUT ROW WITH COLOR
                        3434   C               ASSUME  DS:ABS0
136B  1E                3435   C               PUSH    DS                      ; SAVE SEGMENT
136C  E8 0CFE R         3436   C               CALL    DDS                     ; LOW MEMORY SEGMENT
136F  03 3E 044A R      3437   C               ADD     DI,CRT_COLS             ; NEXT ROW
1373  1F                3438   C               POP     DS                      ; RECOVER
1374  4B                3439   C               DEC     BX                      ; NEXT
1375  75 F1             3440   C               JNZ     S13                     ; DO MORE
1377  E8 1348 R         3441   C               CALL    PART_2
137A  C3                3442   C               RET                             ; RETURN TO CALLER
137B                    3443   C  BLNK_3       ENDP
                        3444   C
137B                    3445   C  BLNK_4       PROC    NEAR                    ; BLANK FOR SCROLL DOWN
137B  1E                3446   C               PUSH    DS                      ; SAVE DATA SEGMENT
                        3447   C               ASSUME  DS:ABS0
137C  E8 0CFE R         3448   C               CALL    DDS                     ; GET LOW MEMORY SEGMENT
137F  8A F7             3449   C               MOV     DH,BH                   ; ATTRIBUTE FOR BLANK LINE
1381  2A FF             3450   C               SUB     BH,BH                   ; CLEAR HIGH BYTE
1383  50                3451   C               PUSH    AX                      ; SAVE
1384  52                3452   C               PUSH    DX                      ; SAVE BECAUSE OF MULTIPLY
1385  8B C3             3453   C               MOV     AX,BX                   ; ROW COUNT
1387  F7 26 0485 R      3454   C               MUL     POINTS                  ; CHARACTER HEIGHT
138B  8B D8             3455   C               MOV     BX,AX                   ; NET VALUE TO BX
138D  5A                3456   C               POP     DX                      ; RECOVER
138E  58                3457   C               POP     AX
                        3458   C
138F  1F                3459   C               POP     DS
                        3460   C               ASSUME  DS:NOTHING
1390                    3461   C  S13_4:
1390  E8 131C R         3462   C               CALL    PART_1                  ; BLANK OUT ROW WITH COLOR
                        3463   C               ASSUME  DS:ABS0
1393  1E                3464   C               PUSH    DS                      ; SAVE SEGMENT
1394  E8 0CFE R         3465   C               CALL    DDS                     ; LOW MEMORY SEGMENT
1397  2B 3E 044A R      3466   C               SUB     DI,CRT_COLS             ; NEXT ROW
139B  1F                3467   C               POP     DS                      ; RECOVER
139C  4B                3468   C               DEC     BX                      ; NEXT
139D  75 F1             3469   C               JNZ     S13_4                   ; DO MORE
139F  E8 1348 R         3470   C               CALL    PART_2
13A2  C3                3471   C               RET                             ; RETURN TO CALLER
13A3                    3472   C  BLNK_4       ENDP
                        3473   C
                        3474   C  ;--------------------------------------------------------
                        3475   C  ; SCROLL UP                                            :
                        3476   C  ;       THIS ROUTINE MOVES A BLOCK OF CHARACTERS UP     :
                        3477   C  ;       ON THE SCREEN                                   :
                        3478   C  ; INPUT                                                 :
                        3479   C  ;       (AH) = CURRENT CRT MODE                         :
                        3480   C  ;       (AL) = NUMBER OF ROWS TO SCROLL                 :
                        3481   C  ;       (CX) = ROW/COLUMN OF UPPER LEFT CORNER          :
                        3482   C  ;       (DX) = ROW/COLUMN OF LOWER RIGHT CORNER         :
                        3483   C  ;       (BH) = ATTRIBUTE TO BE USED ON BLANKED LINE     :
                        3484   C  ;       (DS) = DATA SEGMENT                             :
                        3485   C  ;       (ES) = REGEN BUFFER SEGMENT                     :
                        3486   C  ; OUTPUT                                                :
                        3487   C  ;       NONE -- THE REGEN BUFFER IS MODIFIED            :
                        3488   C  ;--------------------------------------------------------
                        3489   C               ASSUME  CS:CODE,DS:ABS0,ES:NOTHING
13A3                    3490   C  SCROLL_UP    PROC    NEAR
13A3  8A D8             3491   C               MOV     BL,AL                   ; SAVE LINE COUNT IN BL
13A5  E8 16EB R         3492   C               CALL    MK_ES
13A8  80 FC 04          3493   C               CMP     AH,4                    ; TEST FOR GRAPHICS MODE
13AB  72 08             3494   C               JB      N1                      ; HANDLE SEPERATELY
13AD  80 FC 07          3495   C               CMP     AH,7                    ; TEST FOR BW CARD
13B0  74 03             3496   C               JE      N1
13B2  E9 1474 R         3497   C               JMP     GRAPHICS_UP
13B5                    3498   C  N1:                                          ; UP_CONTINUE
13B5  53                3499   C               PUSH    BX                      ; SAVE FILL ATTR IN BH
13B6  8B C1             3500   C               MOV     AX,CX                   ; UPPER LEFT POSITION
13B8  E8 13F2 R         3501   C               CALL    SCROLL_POSITION         ; DO SETUP FOR SCROLL
13BB  74 31             3502   C               JZ      N7                      ; BLANK_FIELD
13BD  03 F0             3503   C               ADD     SI,AX                   ; FROM ADDRESS
13BF  8A E6             3504   C               MOV     AH,DH                   ; # ROWS IN BLOCK
13C1  2A E3             3505   C               SUB     AH,BL                   ; # ROWS TO BE MOVED
13C3                    3506   C  N2:                                          ; ROW_LOOP
13C3  E8 1432 R         3507   C               CALL    N10                     ; MOVE ONE ROW
13C6  03 F5             3508   C               ADD     SI,BP
13C8  03 FD             3509   C               ADD     DI,BP                   ; NEXT LINE IN BLOCK
13CA  FE CC             3510   C               DEC     AH                      ; COUNT OF LINES TO MOVE
13CC  75 F5             3511   C               JNZ     N2                      ; ROW_LOOP
13CE                    3512   C  N3:                                          ; CLEAR_ENTRY
13CE  58                3513   C               POP     AX                      ; RECOVER ATTRIBUTE IN AH
13CF  B0 20             3514   C               MOV     AL,' '                  ; FILL WITH BLANKS
13D1                    3515   C  N4:                                          ; CLEAR_LOOP
13D1  E8 143B R         3516   C               CALL    N11                     ; CLEAR THE ROW
13D4  03 FD             3517   C               ADD     DI,BP                   ; POINT TO NEXT LINE
13D6  FE CB             3518   C               DEC     BL                      ; LINES TO SCROLL
13D8  75 F7             3519   C               JNZ     N4                      ; CLEAR_LOOP
13DA                    3520   C  N5:                                          ; SCROLL_END
13DA  E8 0CFE R         3521   C               CALL    DDS
13DD  80 3E 0449 R 07   3522   C               CMP     CRT_MODE,7              ; IS THIS THE B/W CARD
13E2  74 07             3523   C               JE      N6                      ; SKIP THE MODE RESET
13E4  A0 0465 R         3524   C               MOV     AL,CRT_MODE_SET         ; GET THE MODE SET
13E7  BA 03D8           3525   C               MOV     DX,03D8H                ; ALWAYS SET COLOR CARD
13EA  EE                3526   C               OUT     DX,AL
13EB                    3527   C  N6:                                          ; VIDEO_RET_HERE
13EB  E9 219E R         3528   C               JMP     V_RET
```

```
13EE                            3529    C  N7:                                          ; BLANK_FIELD
13EE  8A DE                     3530    C          MOV     BL,DH                     ; GET ROW COUNT
13F0  EB DC                     3531    C          JMP     N3                        ; GO CLEAR THAT AREA
13F2                            3532    C  SCROLL_UP       ENDP
                                3533    C
                                3534    C  ;----- HANDLE COMMON SCROLL SET UP HERE
                                3535    C
13F2                            3536    C  SCROLL_POSITION PROC    NEAR
13F2  F6 06 0487 R 04           3537    C          TEST    INFO,4
13F7  74 12                     3538    C          JZ      N9
                                3539    C
                                3540    C  ;----- 80X25 COLOR CARD SCROLL
                                3541    C
13F9  52                        3542    C          PUSH    DX
13FA  B6 03                     3543    C          MOV     DH,3
13FC  B2 DA                     3544    C          MOV     DL,0DAH                   ; COLOR CARD HERE
13FE  50                        3545    C          PUSH    AX
13FF                            3546    C  N8:                                       ; WAIT_DISP_ENABLE
13FF  EC                        3547    C          IN      AL,DX
1400  A8 08                     3548    C          TEST    AL,8                      ; WAIT FOR VERT RETRACE
1402  74 FB                     3549    C          JZ      N8                        ; WAIT_DISP_ENABLE
1404  B0 25                     3550    C          MOV     AL,25H
1406  B2 D8                     3551    C          MOV     DL,0D8H                   ; DX=3D8
1408  EE                        3552    C          OUT     DX,AL                     ; TURN OFF VIDEO
1409  58                        3553    C          POP     AX                        ;  DURING VERTICAL RETRACE
140A  5A                        3554    C          POP     DX
140B                            3555    C  N9:
140B  E8 1146 R                 3556    C          CALL    POSITION                  ; CONVERT TO REGEN POINTER
140E  03 06 044E R              3557    C          ADD     AX,CRT_START              ; OFFSET OF ACTIVE PAGE
1412  8B F8                     3558    C          MOV     DI,AX                     ; TO ADDRESS FOR SCROLL
1414  8B F0                     3559    C          MOV     SI,AX                     ; FROM ADDRESS FOR SCROLL
1416  2B D1                     3560    C          SUB     DX,CX                     ; DX = #ROWS, #COLS
1418  FE C6                     3561    C          INC     DH
141A  FE C2                     3562    C          INC     DL                        ; INCREMENT FOR 0 ORIGIN
141C  32 ED                     3563    C          XOR     CH,CH                     ; ZERO HIGH BYTE OF COUNT
141E  8B 2E 044A R              3564    C          MOV     BP,CRT_COLS               ; NUM OF COLS IN DISPLAY
1422  03 ED                     3565    C          ADD     BP,BP                     ; TIMES 2 FOR ATTR BYTE
1424  8A C3                     3566    C          MOV     AL,BL                     ; GET LINE COUNT
1426  F6 26 044A R              3567    C          MUL     BYTE PTR CRT_COLS         ; OFFSET TO FROM ADDRESS
142A  03 C0                     3568    C          ADD     AX,AX                     ; *2 FOR ATTRIBUTE BYTE
142C  06                        3569    C          PUSH    ES                        ; ESTABLISH ADDRESSING
142D  1F                        3570    C          POP     DS                        ;  FOR BOTH POINTERS
142E  80 FB 00                  3571    C          CMP     BL,0                      ; 0 MEANS BLANK FIELD
1431  C3                        3572    C          RET                               ; RETURN WITH FLAGS SET
1432                            3573    C  SCROLL_POSITION ENDP
                                3574    C
                                3575    C  ;----- MOVE_ROW
                                3576    C
1432                            3577    C  N10     PROC    NEAR
1432  8A CA                     3578    C          MOV     CL,DL                     ; GET # OF COLS TO MOVE
1434  56                        3579    C          PUSH    SI
1435  57                        3580    C          PUSH    DI                        ; SAVE START ADDRESS
1436  F3/ A5                    3581    C          REP     MOVSW                     ; MOVE THAT LINE ON SCREEN
1438  5F                        3582    C          POP     DI
1439  5E                        3583    C          POP     SI                        ; RECOVER ADDRESSES
143A  C3                        3584    C          RET
143B                            3585    C  N10     ENDP
                                3586    C
                                3587    C  ;----- CLEAR_ROW
                                3588    C
143B                            3589    C  N11     PROC    NEAR
143B  8A CA                     3590    C          MOV     CL,DL                     ; GET # COLUMNS TO CLEAR
143D  57                        3591    C          PUSH    DI
143E  F3/ AB                    3592    C          REP     STOSW                     ; STORE THE FILL CHARACTER
1440  5F                        3593    C          POP     DI
1441  C3                        3594    C          RET
1442                            3595    C  N11     ENDP
                                3596    C
                                3597    C  ;--------------------------------------------------------
                                3598    C  ; SCROLL_DOWN                                          :
                                3599    C  ;       THIS ROUTINE MOVES THE CHARACTERS WITHIN A     :
                                3600    C  ;       DEFINED BLOCK DOWN ON THE SCREEN, FILLING THE  :
                                3601    C  ;       TOP LINES WITH A DEFINED CHARACTER             :
                                3602    C  ; INPUT                                                :
                                3603    C  ;       (AH) = CURRENT CRT MODE                        :
                                3604    C  ;       (AL) = NUMBER OF LINES TO SCROLL               :
                                3605    C  ;       (CX) = UPPER LEFT CORNER OF REGION             :
                                3606    C  ;       (DX) = LOWER RIGHT CORNER OF REGION            :
                                3607    C  ;       (BH) = FILL CHARACTER                          :
                                3608    C  ;       (DS) = DATA SEGMENT                            :
                                3609    C  ;       (ES) = REGEN SEGMENT                           :
                                3610    C  ; OUTPUT                                               :
                                3611    C  ;       NONE -- SCREEN IS SCROLLED                     :
                                3612    C  ;--------------------------------------------------------
1442                            3613    C  SCROLL_DOWN     PROC    NEAR
1442  FD                        3614    C          STD                               ; SCROLL DOWN
1443  8A D8                     3615    C          MOV     BL,AL                     ; LINE COUNT TO BL
1445  E8 16EB R                 3616    C          CALL    MK_ES
1448  53                        3617    C          PUSH    BX                        ; SAVE ATTRIBUTE IN BH
1449  8B C2                     3618    C          MOV     AX,DX                     ; LOWER RIGHT CORNER
144B  E8 13F2 R                 3619    C          CALL    SCROLL_POSITION           ; GET REGEN LOCATION
144E  74 20                     3620    C          JZ      N16
1450  2B F0                     3621    C          SUB     SI,AX                     ; SI IS FROM ADDRESS
1452  8A E6                     3622    C          MOV     AH,DH                     ; GET TOTAL # ROWS
1454  2A E3                     3623    C          SUB     AH,BL                     ; COUNT TO MOVE IN SCROLL
1456                            3624    C  N13:
1456  E8 1432 R                 3625    C          CALL    N10                       ; MOVE ONE ROW
1459  2B F5                     3626    C          SUB     SI,BP
145B  2B FD                     3627    C          SUB     DI,BP
145D  FE CC                     3628    C          DEC     AH
145F  75 F5                     3629    C          JNZ     N13
1461                            3630    C  N14:
1461  58                        3631    C          POP     AX                        ; RECOVER ATTRIBUTE IN AH
1462  B0 20                     3632    C          MOV     AL,' '
1464                            3633    C  N15:
1464  E8 143B R                 3634    C          CALL    N11                       ; CLEAR ONE ROW
1467  2B FD                     3635    C          SUB     DI,BP                     ; GO TO NEXT ROW
1469  FE CB                     3636    C          DEC     BL
146B  75 F7                     3637    C          JNZ     N15
146D  E9 13DA R                 3638    C          JMP     N5                        ; SCROLL_END
1470                            3639    C  N16:
1470  8A DE                     3640    C          MOV     BL,DH
1472  EB ED                     3641    C          JMP     N14
1474                            3642    C  SCROLL_DOWN     ENDP
                                3643    C
                                3644    C  ;--------------------------------------------------------------
                                3645    C  ; SCROLL UP                                                 :
                                3646    C  ;       THIS ROUTINE SCROLLS UP THE INFORMATION ON THE CRT  :
                                3647    C  ; ENTRY                                                     :
                                3648    C  ;       CH,CL = UPPER LEFT CORNER OF REGION TO SCROLL       :
                                3649    C  ;       DH,DL = LOWER RIGHT CORNER OF REGION TO SCROLL      :
                                3650    C  ;         BOTH OF THE ABOVE ARE IN CHARACTER POSITIONS       :
                                3651    C  ;       BH = FILL VALUE FOR BLANKED LINES                   :
                                3652    C  ;       AL = # LINES TO SCROLL (AL=0 MEANS BLANK THE ENTIRE  :
                                3653    C  ;            FIELD)                                         :
                                3654    C  ;       DS = DATA SEGMENT                                   :
```

```
                        3655  C  ;           ES = REGEN SEGMENT                                  :
                        3656  C  ; EXIT                                                          :
                        3657  C  ;           NOTHING, THE SCREEN IS SCROLLED                     :
                        3658  C  ;--------------------------------------------------------------------
1474                    3659  C  GRAPHICS_UP      PROC    NEAR
1474  8A D8             3660  C                   MOV     BL,AL                   ; SAVE LINE COUNT IN BL
1476  8B C1             3661  C                   MOV     AX,CX                   ; GET UPPER LEFT POSITION
                        3662  C                                                   ;  INTO AX REG
                        3663  C
                        3664  C  ;----- USE CHARACTER SUBROUTINE FOR POSITIONING
                        3665  C  ;----- ADDRESS RETURNED IS MULTIPLIED BY 2 FROM CORRECT VALUE
                        3666  C
1478  E8 16A7 R         3667  C                   CALL    GRAPH_POSN
147B  8B F8             3668  C                   MOV     DI,AX                   ; SAVE RESULT AS
                        3669  C                                                   ;  DESTINATION ADDRESS
                        3670  C
                        3671  C  ;----- DETERMINE SIZE OF WINDOW
                        3672  C
147D  2B D1             3673  C                   SUB     DX,CX
147F  81 C2 0101        3674  C                   ADD     DX,101H                 ; ADJUST VALUES
1483  D0 E6             3675  C                   SAL     DH,1                    ; MULTIPLY # ROWS BY 4
                        3676  C                                                   ;  SINCE 8 VERT DOTS/CHAR
1485  D0 E6             3677  C                   SAL     DH,1                    ;  AND EVEN/ODD ROWS
                        3678  C
                        3679  C  ;----- DETERMINE CRT MODE
                        3680  C
1487  80 3E 0449 R 06   3681  C                   CMP     CRT_MODE,6              ; TEST FOR MEDIUM RES
148C  73 04             3682  C                   JNC     R7                      ; FIND_SOURCE
                        3683  C
                        3684  C  ;----- MEDIUM RES UP
                        3685  C
148E  D0 E2             3686  C                   SAL     DL,1                    ; * 2,
1490  D1 E7             3687  C                   SAL     DI,1                    ;  SINCE 2 BYTES/CHAR
                        3688  C
                        3689  C  ;----- DETERMINE THE SOURCE ADDRESS IN THE BUFFER
                        3690  C
1492                    3691  C  R7:                                               ; FIND_SOURCE
1492  06                3692  C                   PUSH    ES                      ; GET SEGMENTS BOTH
1493  1F                3693  C                   POP     DS                      ;  POINTING TO REGEN
1494  2A ED             3694  C                   SUB     CH,CH                   ; 0 TO HIGH OF COUNT REG
1496  D0 E3             3695  C                   SAL     BL,1                    ; NUMBER OF LINES *4
1498  D0 E3             3696  C                   SAL     BL,1
149A  74 2D             3697  C                   JZ      R11                     ; IF 0, BLANK ENTIRE FIELD
149C  8A C3             3698  C                   MOV     AL,BL                   ; NUMBER OF LINES IN AL
149E  B4 50             3699  C                   MOV     AH,80                   ; 80 BYTES/ROW
14A0  F6 E4             3700  C                   MUL     AH                      ; OFFSET TO SOURCE
14A2  8B F7             3701  C                   MOV     SI,DI                   ; SET UP SOURCE
14A4  03 F0             3702  C                   ADD     SI,AX                   ;  ADD IN OFFSET TO IT
14A6  8A E6             3703  C                   MOV     AH,DH                   ; NUMBER OF ROWS IN FIELD
14A8  2A E3             3704  C                   SUB     AH,BL                   ; DETERMINE NUMBER TO MOVE
                        3705  C
                        3706  C  ;----- LOOP THROUGH, MOVING ONE ROW AT A TIME, BOTH EVEN AND ODD FIELDS
                        3707  C
14AA                    3708  C  R8:                                               ; ROW_LOOP
14AA  E8 14CD R         3709  C                   CALL    R17                     ; MOVE ONE ROW
14AD  81 EF 1FB0        3710  C                   SUB     SI,2000H-80             ; MOVE TO NEXT ROW
14B1  81 EF 1FB0        3711  C                   SUB     DI,2000H-80
14B5  FE CC             3712  C                   DEC     AH                      ; NUMBER OF ROWS TO MOVE
14B7  75 F1             3713  C                   JNZ     R8                      ; CONTINUE TILL ALL MOVED
                        3714  C
                        3715  C  ;----- FILL IN THE VACATED LINE(S)
                        3716  C
14B9                    3717  C  R9:                                               ; CLEAR_ENTRY
14B9  8A C7             3718  C                   MOV     AL,BH                   ; ATTRIBUTE TO FILL WITH
14BB                    3719  C  R10:
14BB  E8 14E6 R         3720  C                   CALL    R18                     ; CLEAR THAT ROW
14BE  81 EF 1FB0        3721  C                   SUB     DI,2000H-80             ; POINT TO NEXT LINE
14C2  FE CB             3722  C                   DEC     BL                      ; NUMBER OF LINES TO FILL
14C4  75 F5             3723  C                   JNZ     R10                     ; CLEAR_LOOP
14C6  E9 219E R         3724  C                   JMP     V_RET
14C9                    3725  C  R11:                                              ; BLANK_FIELD
14C9  8A DE             3726  C                   MOV     BL,DH                   ; SET BLANK COUNT TO
                        3727  C                                                   ;  EVERYTHING IN FIELD
14CB  EB EC             3728  C                   JMP     R9                      ; CLEAR THE FIELD
14CD                    3729  C  GRAPHICS_UP      ENDP
                        3730  C
                        3731  C  ;----- ROUTINE TO MOVE ONE ROW OF INFORMATION
                        3732  C
14CD                    3733  C  R17     PROC    NEAR
14CD  8A CA             3734  C                   MOV     CL,DL                   ; NUM OF BYTES IN THE ROW
14CF  56                3735  C                   PUSH    SI
14D0  57                3736  C                   PUSH    DI                      ; SAVE POINTERS
14D1  F3/ A4            3737  C                   REP     MOVSB                   ; MOVE THE EVEN FIELD
14D3  5F                3738  C                   POP     DI
14D4  5E                3739  C                   POP     SI
14D5  81 C6 2000        3740  C                   ADD     SI,2000H
14D9  81 C7 2000        3741  C                   ADD     DI,2000H                ; POINT TO THE ODD FIELD
14DD  56                3742  C                   PUSH    SI
14DE  57                3743  C                   PUSH    DI                      ; SAVE THE POINTERS
14DF  8A CA             3744  C                   MOV     CL,DL                   ; COUNT BACK
14E1  F3/ A4            3745  C                   REP     MOVSB                   ; MOVE THE ODD FIELD
14E3  5F                3746  C                   POP     DI
14E4  5E                3747  C                   POP     SI                      ; POINTERS BACK
14E5  C3                3748  C                   RET                             ; RETURN TO CALLER
14E6                    3749  C  R17     ENDP
                        3750  C
                        3751  C  ;----- CLEAR A SINGLE ROW
                        3752  C
14E6                    3753  C  R18     PROC    NEAR
14E6  8A CA             3754  C                   MOV     CL,DL                   ; NUMBER OF BYTES IN FIELD
14E8  57                3755  C                   PUSH    DI                      ; SAVE POINTER
14E9  F3/ AA            3756  C                   REP     STOSB                   ; STORE THE NEW VALUE
14EB  5F                3757  C                   POP     DI                      ; POINTER BACK
14EC  81 C7 2000        3758  C                   ADD     DI,2000H                ; POINT TO ODD FIELD
14F0  57                3759  C                   PUSH    DI
14F1  8A CA             3760  C                   MOV     CL,DL
14F3  F3/ AA            3761  C                   REP     STOSB                   ; FILL THE ODD FIELD
14F5  5F                3762  C                   POP     DI
14F6  C3                3763  C                   RET                             ; RETURN TO CALLER
14F7                    3764  C  R18     ENDP
                        3765  C
14F7                    3766  C  MEM_DET PROC    NEAR
                        3767  C                   ASSUME  DS:ABS0
14F7  50                3768  C                   PUSH    AX
14F8  1E                3769  C                   PUSH    DS
14F9  E8 0CFE R         3770  C                   CALL    DDS
14FC  8A 26 0487 R      3771  C                   MOV     AH,INFO
1500  80 E4 60          3772  C                   AND     AH,060H
1503  1F                3773  C                   POP     DS
1504  58                3774  C                   POP     AX
1505  74 02             3775  C                   JZ      MIN
1507  F9                3776  C                   STC
1508  C3                3777  C                   RET
1509                    3778  C  MIN:
1509  F8                3779  C                   CLC
150A  C3                3780  C                   RET
```

```
                        C  MEM_DET ENDP
                        C
                        C  ;----- SCROLL ACTIVE PAGE UP
                        C
150B                    C  SC_2:
150B  E9 13A3 R         C          JMP     SCROLL_UP
                        C
150E                    C  AH6:
                        C          ASSUME  DS:ABS0
150E  E8 12D1 R         C          CALL    FLTA
1511  8A 26 0449 R      C          MOV     AH,CRT_MODE                  ; GET CURRENT MODE
1515  80 FC 07          C          CMP     AH,07H
1518  76 F1             C          JBE     SC_2                         ; ANY OF THE OLD MODES
151A  80 FC 0D          C          CMP     AH,0DH
151D  73 17             C          JAE     GRAPHICS_UP_2                ; NEW GRAPHICS MODES
151F  E9 219E R         C          JMP     V_RET                        ; NOT A RECOGNIZED MODE
                        C
1522                    C  GR_ST_1 PROC    NEAR
1522  BA A000           C          MOV     DX,0A000H                    ; REGEN BUFFER
1525  BD 0511           C          MOV     BP,0511H                     ; GRAPHICS WRITE MODE
1528  80 FC 0F          C          CMP     AH,0FH
152B  72 08             C          JB      VV1
152D  E8 14F7 R         C          CALL    MEM_DET
1530  73 03             C          JNC     VV1
1532  BD 0501           C          MOV     BP,0501H                     ; GRAPHICS WRITE MODE
1535                    C  VV1:
1535  C3                C          RET
1536                    C  GR_ST_1 ENDP
                        C
1536                    C  GRAPHICS_UP_2   PROC    NEAR
                        C          ASSUME  DS:ABS0
1536  52                C          PUSH    DX
1537  E8 1522 R         C          CALL    GR_ST_1                      ; SET SEGMENT, WRITE MODE
                        C          SRLOAD  ES                           ; SET REGEN
153A  8E C2             C+         MOV     ES,DX
153C  5A                C          POP     DX
153D  8A D8             C          MOV     BL,AL                        ; NUMBER OF LINES
153F  8B C1             C          MOV     AX,CX                        ; UPPER LEFT CORNER
1541  53                C          PUSH    BX
1542  8A 3E 0462 R      C          MOV     BH,ACTIVE_PAGE               ; ACTIVE PAGE FOR SCROLL
1546  E8 16C6 R         C          CALL    GRX_PSN                      ; ADDRESS IN REGEN
1549  5B                C          POP     BX
154A  8B F8             C          MOV     DI,AX                        ; SET POINTER
154C  2B D1             C          SUB     DX,CX                        ; DETERMINE WINDOW
154E  81 C2 0101        C          ADD     DX,0101H                     ; ADJUST
1552  2A E4             C          SUB     AH,AH                        ; ZERO HIGH BYTE
1554  8A C3             C          MOV     AL,BL                        ; LINE COUNT
1556  52                C          PUSH    DX
1557  F7 26 0485 R      C          MUL     POINTS                       ; BYTES PER CHARACTER
155B  F7 26 044A R      C          MUL     CRT_COLS                     ; COLUMNS
155F  8B F7             C          MOV     SI,DI                        ; SET UP SOURCE INDEX
1561  03 F0             C          ADD     SI,AX                        ; ADJUST
                        C          ASSUME  DS:NOTHING
1563  06                C          PUSH    ES
1564  1F                C          POP     DS
1565  5A                C          POP     DX
1566  0A DB             C          OR      BL,BL                        ; LINE COUNT
1568  74 3F             C          JZ      AR9
156A  8A CE             C          MOV     CL,DH
156C  2A CB             C          SUB     CL,BL
156E  2A ED             C          SUB     CH,CH
                        C
                        C          ASSUME  DS:ABS0
1570  1E                C          PUSH    DS
1571  E8 0CFE R         C          CALL    DDS                          ; LOW MEMORY SEGMENT
1574  50                C          PUSH    AX
1575  52                C          PUSH    DX
1576  8B C1             C          MOV     AX,CX
1578  F7 26 0485 R      C          MUL     POINTS                       ; BYTES PER CHAR
157C  8B C8             C          MOV     CX,AX                        ; SET THE COUNT
157E  5A                C          POP     DX
157F  58                C          POP     AX
                        C          ASSUME  DS:NOTHING
1580  1F                C          POP     DS
                        C
1581  52                C          PUSH    DX
1582  8B C5             C          MOV     AX,BP
1584  B6 03             C          MOV     DH,3
1586  B2 CE             C          MOV     DL,GRAPH_ADDR                ; GRAPHICS
1588  E8 0D15 R         C          CALL    OUT_DX
158B  B2 C4             C          MOV     DL,SEQ_ADDR                  ; SEQUENCER
158D  B8 020F           C          MOV     AX,020FH                     ; ENABLE ALL MAPS
1590  E8 0D15 R         C          CALL    OUT_DX
1593  5A                C          POP     DX
1594  E8 12E0 R         C          CALL    CRANK                        ; SCROLL THE SCREEN
                        C
1597  52                C          PUSH    DX
1598  4D                C          DEC     BP
1599  8B C5             C          MOV     AX,BP
159B  B6 03             C          MOV     DH,3
159D  B2 CE             C          MOV     DL,GRAPH_ADDR
159F  E8 0D15 R         C          CALL    OUT_DX
15A2  5A                C          POP     DX
15A3                    C  AR10:
15A3  E8 1353 R         C          CALL    BLNK_3
15A6  E9 219E R         C          JMP     V_RET
15A9                    C  AR9:
15A9  8A DE             C          MOV     BL,DH                        ; BLANK ENTIRE WINDOW
15AB  EB F6             C          JMP     AR10
15AD                    C  GRAPHICS_UP_2   ENDP
                        C
                        C  ;----- SCROLL ACTIVE DISPLAY PAGE DOWN
                        C
15AD                    C  SC_3:
15AD  E9 1442 R         C          JMP     SCROLL_DOWN
                        C
15B0                    C  AH7:
                        C          ASSUME  DS:ABS0
15B0  E8 12D1 R         C          CALL    FLTA
15B3  8A 26 0449 R      C          MOV     AH,CRT_MODE
15B7  80 FC 03          C          CMP     AH,03H                       ; OLD COLOR ALPHA
15BA  76 F1             C          JBE     SC_3
15BC  80 FC 07          C          CMP     AH,07H                       ; MONOCHROME ALPHA
15BF  74 EC             C          JE      SC_3
                        C
15C1  80 FC 0D          C          CMP     AH,0DH                       ; NEW GRAPHICS MODES
15C4  73 0C             C          JAE     GRAPHICS_DN_2
15C6  80 FC 06          C          CMP     AH,06H                       ; OLD GRAPHICS MODES
15C9  77 04             C          JA      M_0
15CB  B4 07             C          MOV     AH,07H
15CD  CD 42             C          INT     42H
15CF                    C  M_0:
15CF  E9 219E R         C          JMP     V_RET
                        C
15D2                    C  GRAPHICS_DN_2   PROC    NEAR
15D2  FD                C          STD                                  ; DIRECTION TO DECREMENT
```

```
15D3  8A D8                3907  C          MOV     BL,AL                           ; LINE COUNT
15D5  52                   3908  C          PUSH    DX                              ; SAVE LOWER RIGHT
15D6  E8 1522 R            3909  C          CALL    GR_ST_1
                           3910  C          SRLOAD  ES                              ; SET REGEN SEGMENT
15D9  8E C2                3911  C+         MOV     ES,DX
15DB  5A                   3912  C          POP     DX
15DC  8B C2                3913  C          MOV     AX,DX
15DE  FE C4                3914  C          INC     AH                              ; MOV CHAR ROW UP BY ONE
15E0  53                   3915  C          PUSH    BX
15E1  8A 3E 0462 R         3916  C          MOV     BH,ACTIVE_PAGE
15E5  E8 16C6 R            3917  C          CALL    GRX_PSN                         ; ADDRESS IN REGEN
15E8  5B                   3918  C          POP     BX
15E9  2B 06 044A R         3919  C          SUB     AX,CRT_COLS                     ; ONE SCAN OVERSHOOT
15ED  8B F8                3920  C          MOV     DI,AX
15EF  2B D1                3921  C          SUB     DX,CX                           ; CALCULATE WINDOW
15F1  81 C2 0101           3922  C          ADD     DX,0101H                        ; ADJUST COUNT
15F5  2A E4                3923  C          SUB     AH,AH
15F7  8A C3                3924  C          MOV     AL,BL
15F9  52                   3925  C          PUSH    DX
15FA  F7 26 0485 R         3926  C          MUL     POINTS                          ; BYTES PER CHAR
15FE  F7 26 044A R         3927  C          MUL     CRT_COLS                        ; BYTES PER ROW
1602  8B F7                3928  C          MOV     SI,DI
1604  2B F0                3929  C          SUB     SI,AX
                           3930  C          ASSUME  DS:NOTHING
1606  06                   3931  C          PUSH    ES                              ; SET DS TO
1607  1F                   3932  C          POP     DS                              ;  THE REGEN SEGMENT
1608  5A                   3933  C          POP     DX
1609  0A DB                3934  C          OR      BL,BL                           ; SCROLL COUNT
160B  74 4D                3935  C          JZ      DXR9                            ; BLANK ENTIRE WINDOW
160D  8A CE                3936  C          MOV     CL,DH
160F  2A CB                3937  C          SUB     CL,BL
1611  2A ED                3938  C          SUB     CH,CH
                           3939  C
                           3940  C          ASSUME  DS:ABS0
1613  1E                   3941  C          PUSH    DS
1614  E8 0CFE R            3942  C          CALL    DDS
1617  50                   3943  C          PUSH    AX
1618  52                   3944  C          PUSH    DX
1619  8B C1                3945  C          MOV     AX,CX
161B  F7 26 0485 R         3946  C          MUL     POINTS                          ; BYTES PER CHAR
161F  8B C8                3947  C          MOV     CX,AX
1621  5A                   3948  C          POP     DX
1622  58                   3949  C          POP     AX
                           3950  C          ASSUME  DS:NOTHING
1623  1F                   3951  C          POP     DS
                           3952  C
1624  52                   3953  C          PUSH    DX
1625  8B C5                3954  C          MOV     AX,BP
1627  B6 03                3955  C          MOV     DH,3
1629  B2 CE                3956  C          MOV     DL,GRAPH_ADDR                   ; GRAPHICS
162B  E8 0D15 R            3957  C          CALL    OUT_DX
162E  B2 C4                3958  C          MOV     DL,SEQ_ADDR                     ; SEQUENCER
1630  B8 020F              3959  C          MOV     AX,020FH                        ; ENABLE ALL MAPS
1633  E8 0D15 R            3960  C          CALL    OUT_DX
1636  5A                   3961  C          POP     DX
1637  E8 12FE R            3962  C          CALL    CRANK_4                         ; SCROLL THE SCREEN
                           3963  C
163A  52                   3964  C          PUSH    DX
163B  4D                   3965  C          DEC     BP
163C  8B C5                3966  C          MOV     AX,BP
163E  B6 03                3967  C          MOV     DH,3
1640  B2 CE                3968  C          MOV     DL,GRAPH_ADDR
1642  E8 0D15 R            3969  C          CALL    OUT_DX
1645  5A                   3970  C          POP     DX
1646                       3971  C  DXR10:
1646  E8 137B R            3972  C          CALL    BLNK_4
1649  FC                   3973  C          CLD
164A  E9 219E R            3974  C          JMP     V_RET
164D                       3975  C  DXR9:
164D  8A DE                3976  C          MOV     BL,DH                           ; BLANK ENTIRE WINDOW
164F  EB F5                3977  C          JMP     DXR10
1651                       3978  C  GRAPHICS_DN_2   ENDP
                           3979  C
                           3980  C          SUBTTL
                           3981
                           3982  C          INCLUDE         VGRW.INC
                           3983  C          SUBTTL  VGRW.INC
                           3984  C          PAGE
                           3985  C
                           3986  C          ASSUME  DS:ABS0
1651                       3987  C  FIND_POSITION   PROC    NEAR
1651  8A CF                3988  C          MOV     CL,BH                           ; DISPLAY PAGE TO CX
1653  32 ED                3989  C          XOR     CH,CH
1655  8B F1                3990  C          MOV     SI,CX                           ; MOVE TO SI FOR INDEX
1657  D1 E6                3991  C          SAL     SI,1                            ; * 2 FOR WORD OFFSET
1659  8B 84 0450 R         3992  C          MOV     AX,[SI+ OFFSET CURSOR_POSN]     ; ROW/COLUMN OF THAT PAGE
165D  33 DB                3993  C          XOR     BX,BX                           ; SET START ADDRESS TO 0
165F  E3 06                3994  C          JCXZ    P5                              ; NO_PAGE
1661                       3995  C  P4:                                            ; PAGE_LOOP
1661  03 1E 044C R         3996  C          ADD     BX,CRT_LEN                      ; LENGTH OF BUFFER
1665  E2 FA                3997  C          LOOP    P4
1667                       3998  C  P5:                                            ; NO_PAGE
1667  E8 1146 R            3999  C          CALL    POSITION                        ; DETERMINE LOC IN REGEN
166A  03 D8                4000  C          ADD     BX,AX                           ; ADD TO START OF REGEN
166C  C3                   4001  C          RET
166D                       4002  C  FIND_POSITION   ENDP
                           4003  C
                           4004  C  ;--------------------------------------------------------
                           4005  C  ; EXPAND_MED_COLOR                                     :
                           4006  C  ;       THIS ROUTINE EXPANDS THE LOW 2 BITS IN BL TO    :
                           4007  C  ;       FILL THE ENTIRE BX REGISTER                     :
                           4008  C  ; ENTRY                                               :
                           4009  C  ;       BL = COLOR TO BE USED ( LOW 2 BITS )            :
                           4010  C  ; EXIT                                                :
                           4011  C  ;       BX = COLOR TO BE USED ( 8 REPLICATIONS OF THE   :
                           4012  C  ;       2 COLOR BITS )                                  :
                           4013  C  ;--------------------------------------------------------
166D                       4014  C  S19      PROC    NEAR
166D  80 E3 03             4015  C          AND     BL,3                            ; ISOLATE THE COLOR BITS
1670  8A C3                4016  C          MOV     AL,BL                           ; COPY TO AL
1672  51                   4017  C          PUSH    CX                              ; SAVE REGISTER
1673  B9 0003              4018  C          MOV     CX,3                            ; NUMBER OF TIMES
1676                       4019  C  S20:
1676  D0 E0                4020  C          SAL     AL,1
1678  D0 E0                4021  C          SAL     AL,1                            ; LEFT SHIFT BY 2
167A  0A D8                4022  C          OR      BL,AL                           ; ANOTHER COLOR VERSION
                           4023  C                                                  ;  INTO BL
167C  E2 F8                4024  C          LOOP    S20                             ; FILL ALL OF BL
167E  8A FB                4025  C          MOV     BH,BL                           ; FILL UPPER PORTION
1680  59                   4026  C          POP     CX                              ; REGISTER BACK
1681  C3                   4027  C          RET                                     ; ALL DONE
1682                       4028  C  S19      ENDP
                           4029  C
                           4030  C  ; EXPAND_BYTE                                        :
                           4031  C  ;       THIS ROUTINE TAKES THE BYTE IN AL AND DOUBLES    :
                           4032  C  ;       ALL OF THE BITS, TURNING THE 8 BITS INTO         :
```

```
                                4033     C  ;       16 BITS. THE RESULT IS LEFT IN AX                :
                                4034     C  ;--------------------------------------------------------
1682                            4035     C  S21     PROC    NEAR
1682 52                         4036     C          PUSH    DX                              ; SAVE REGISTERS
1683 51                         4037     C          PUSH    CX
1684 53                         4038     C          PUSH    BX
1685 2B D2                      4039     C          SUB     DX,DX                           ; RESULT REGISTER
1687 B9 0001                    4040     C          MOV     CX,1                            ; MASK REGISTER
168A                            4041     C  S22:
168A 8B D8                      4042     C          MOV     BX,AX                           ; BASE INTO TEMP
168C 23 D9                      4043     C          AND     BX,CX                           ; USE MASK TO EXTRACT BIT
168E 0B D3                      4044     C          OR      DX,BX                           ; PUT INTO RESULT REGISTER
1690 D1 E0                      4045     C          SHL     AX,1
1692 D1 E1                      4046     C          SHL     CX,1                            ; SHIFT BASE AND MASK BY 1
1694 8B D8                      4047     C          MOV     BX,AX                           ; BASE TO TEMP
1696 23 D9                      4048     C          AND     BX,CX                           ; EXTRACT THE SAME BIT
1698 0B D3                      4049     C          OR      DX,BX                           ; PUT INTO RESULT
169A D1 E1                      4050     C          SHL     CX,1                            ; SHIFT ONLY MASK NOW,
                                4051     C                                                  ;  MOVING TO NEXT BASE
169C 73 EC                      4052     C          JNC     S22                             ; USE MASK BIT COMING OUT
                                4053     C                                                  ;  TO TERMINATE
169E 8B C2                      4054     C          MOV     AX,DX                           ; RESULT TO PARM REGISTER
16A0 5B                         4055     C          POP     BX
16A1 59                         4056     C          POP     CX                              ; RECOVER REGISTERS
16A2 5A                         4057     C          POP     DX
16A3 C3                         4058     C          RET                                     ; ALL DONE
16A4                            4059     C  S21     ENDP
                                4060     C
16A4                            4061     C  S26     PROC    NEAR
16A4 A1 0450 R                  4062     C          MOV     AX,CURSOR_POSN                  ; GET CURRENT CURSOR
16A7                            4063     C  GRAPH_POSN      LABEL   NEAR
16A7 53                         4064     C          PUSH    BX                              ; SAVE REGISTER
16A8 8B D8                      4065     C          MOV     BX,AX                           ; SAVE A COPY OF CURSOR
16AA 8A C4                      4066     C          MOV     AL,AH                           ; GET ROWS TO AL
16AC F6 26 044A R               4067     C          MUL     BYTE PTR CRT_COLS               ; MULTIPLY BY BYTES/COLUMN
16B0 D1 E0                      4068     C          SHL     AX,1                            ; *4 SINCE 4 ROWS/BYTE
16B2 D1 E0                      4069     C          SHL     AX,1
16B4 2A FF                      4070     C          SUB     BH,BH                           ; ISOLATE COLUMN VALUE
16B6 03 C3                      4071     C          ADD     AX,BX                           ; DETERMINE OFFSET
16B8 5B                         4072     C          POP     BX                              ; RECOVER POINTER
16B9 C3                         4073     C          RET                                     ; ALL DONE
16BA                            4074     C  S26     ENDP
                                4075     C
                                4076     C  ;--------------------------------------------------------------------
                                4077     C  ; GR_CUR                                                           :
                                4078     C  ; ENTRY                                                            :
                                4079     C  ;       BH = DISPLAY PAGE                                          :
                                4080     C  ; EXIT                                                             :
                                4081     C  ;       AX = CURSOR POSITION FOR REQUESTED PAGE                    :
                                4082     C  ;--------------------------------------------------------------------
16BA                            4083     C  GR_CUR:
                                4084     C          ASSUME  DS:ABS0
16BA 53                         4085     C          PUSH    BX                              ; SAVE REGISTER
16BB 8A DF                      4086     C          MOV     BL,BH                           ; GET TO LOW BYTE
16BD 2A FF                      4087     C          SUB     BH,BH                           ; ZERO HIGH BYTE
16BF D1 E3                      4088     C          SAL     BX,1                            ; *2 FOR WORD COUNT
16C1 8B 87 0450 R               4089     C          MOV     AX,[BX + OFFSET CURSOR_POSN]    ; CURSOR, REQUESTED PAGE
16C5 5B                         4090     C          POP     BX                              ; RECOVER REGISTER
                                4091     C  ;--------------------------------------------------------------------
                                4092     C  ; GRX_PSN                                                          :
                                4093     C  ; ENTRY                                                            :
                                4094     C  ;       AX = CURSOR POSITION IN DESIRED PAGE                       :
                                4095     C  ;       BH = DESIRED PAGE                                          :
                                4096     C  ; EXIT                                                             :
                                4097     C  ;       AX = BYTE OFFSET INTO REGEN                                :
                                4098     C  ;--------------------------------------------------------------------
16C6                            4099     C  GRX_PSN PROC    NEAR
16C6 53                         4100     C          PUSH    BX                              ; SAVE
16C7 51                         4101     C          PUSH    CX                              ; SAVE
16C8 52                         4102     C          PUSH    DX                              ; SAVE
16C9 2A ED                      4103     C          SUB     CH,CH                           ; ZERO
16CB 8A CF                      4104     C          MOV     CL,BH                           ; PAGE NUMBER
16CD 8B D8                      4105     C          MOV     BX,AX                           ; ROW, COLUMN
16CF 8A C4                      4106     C          MOV     AL,AH                           ; ROW
16D1 F6 26 044A R               4107     C          MUL     BYTE PTR CRT_COLS               ; ROW * COLUMNS/ROW
16D5 F7 26 0485 R               4108     C          MUL     POINTS                          ; BYTES PER ROW
16D9 2A FF                      4109     C          SUB     BH,BH                           ; ZERO TO LEAVE COL VALUE
16DB 03 C3                      4110     C          ADD     AX,BX                           ; ADD IN COLUMN
16DD 8B 1E 044C R               4111     C          MOV     BX,CRT_LEN                      ; PAGE LENGTH
16E1 E3 04                      4112     C          JCXZ    GP_2                            ; NO PAGE OFFSET
16E3                            4113     C  GP_3:
16E3 03 C3                      4114     C          ADD     AX,BX                           ; ADD IN THE PAGE LENGTH
16E5 E2 FC                      4115     C          LOOP    GP_3                            ; DO FOR NUMBER OF PAGES
16E7                            4116     C  GP_2:
16E7 5A                         4117     C          POP     DX                              ; RECOVER
16E8 59                         4118     C          POP     CX                              ; RECOVER
16E9 5B                         4119     C          POP     BX                              ; RECOVER
16EA C3                         4120     C          RET
16EB                            4121     C  GRX_PSN ENDP
                                4122     C
16EB                            4123     C  MK_ES:
16EB BE B800                    4124     C          MOV     SI,0B800H
16EE 8B 3E 0410 R               4125     C          MOV     DI,EQUIP_FLAG
16F2 81 E7 0030                 4126     C          AND     DI,030H
16F6 83 FF 30                   4127     C          CMP     DI,030H
16F9 75 03                      4128     C          JNE     P6_A
16FB BE B000                    4129     C          MOV     SI,0B000H
16FE                            4130     C  P6_A:
16FE 8E C6                      4131     C          MOV     ES,SI
1700 C3                         4132     C          RET
                                4133     C
                                4134     C  ;--------------------------------------------------------
                                4135     C  ; READ_AC_CURRENT                                          :
                                4136     C  ;       THIS ROUTINE READS THE ATTRIBUTE AND CHARACTER     :
                                4137     C  ;       AT THE CURRENT CURSOR POSITION AND RETURNS THEM    :
                                4138     C  ;       TO THE CALLER                                      :
                                4139     C  ;INPUT                                                     :
                                4140     C  ;       (AH) = CURRENT CRT MODE                            :
                                4141     C  ;       (BH) = DISPLAY PAGE ( ALPHA MODES ONLY )           :
                                4142     C  ;       (DS) = DATA SEGMENT                                :
                                4143     C  ;       (ES) = REGEN SEGMENT                               :
                                4144     C  ;OUTPUT                                                    :
                                4145     C  ;       (AL) = CHAR READ                                   :
                                4146     C  ;       (AH) = ATTRIBUTE READ                              :
                                4147     C  ;--------------------------------------------------------
                                4148     C          ASSUME  CS:CODE,DS:ABS0,ES:NOTHING
1701                            4149     C  READ_AC_CURRENT PROC    NEAR
1701 E8 16EB R                  4150     C          CALL    MK_ES
1704 E8 1651 R                  4151     C          CALL    FIND_POSITION
1707 8B F3                      4152     C          MOV     SI,BX                           ; ADDRESSING IN SI
                                4153     C
1709 8B 16 0463 R               4154     C          MOV     DX,ADDR_6845                    ; GET BASE ADDRESS
170D 83 C2 06                   4155     C          ADD     DX,6                            ; POINT AT STATUS PORT
                                4156     C
1710 F6 06 0487 R 04            4157     C          TEST    INFO,4
                                4158     C
```

```
1715  06                    4159    C           PUSH    ES
1716  1F                    4160    C           POP     DS                          ; SEGMENT FOR QUICK ACCESS
                            4161    C
1717  74 0B                 4162    C           JZ      P3A
                            4163    C
                            4164    C   ;----- WAIT FOR HORIZONTAL RETRACE
                            4165    C
1719                        4166    C   P2:                                     ; WAIT FOR RETRACE LOW
1719  EC                    4167    C           IN      AL,DX                   ; GET STATUS
171A  A8 01                 4168    C           TEST    AL,1                    ; IS HORZ RETRACE LOW
171C  75 FB                 4169    C           JNZ     P2                      ; WAIT UNTIL IT IS
171E  FA                    4170    C           CLI                             ; NO MORE INTERRUPTS
171F                        4171    C   P3:                                     ; WAIT FOR RETRACE HIGH
171F  EC                    4172    C           IN      AL,DX                   ; GET STATUS
1720  A8 01                 4173    C           TEST    AL,1                    ; IS IT HIGH
1722  74 FB                 4174    C           JZ      P3                      ; WAIT UNTIL IT IS
1724                        4175    C   P3A:
1724  AD                    4176    C           LODSW                           ; GET THE CHAR/ATTR
1725  E9 219E R             4177    C           JMP     V_RET
1728                        4178    C   READ_AC_CURRENT ENDP
                            4179    C
                            4180    C   ;--------------------------------------------------------
                            4181    C   ; MED_READ_BYTE                                        :
                            4182    C   ;       THIS ROUTINE WILL TAKE 2 BYTES FROM THE REGEN   :
                            4183    C   ;       BUFFER, COMPARE AGAINST THE CURRENT FOREGROUND  :
                            4184    C   ;       COLOR, AND PLACE THE CORRESPONDING ON/OFF BIT   :
                            4185    C   ;       PATTERN INTO THE CURRENT POSITION IN THE SAVE   :
                            4186    C   ;       AREA                                           :
                            4187    C   ; ENTRY                                               :
                            4188    C   ;       SI,DS = POINTER TO REGEN AREA OF INTEREST      :
                            4189    C   ;       BX = EXPANDED FOREGROUND COLOR                 :
                            4190    C   ;       BP = POINTER TO SAVE AREA                      :
                            4191    C   ; EXIT                                                :
                            4192    C   ;       BP IS INCREMENT AFTER SAVE                     :
                            4193    C   ;--------------------------------------------------------
1728                        4194    C   S23     PROC    NEAR
1728  8A 24                 4195    C           MOV     AH,[SI]                 ; GET FIRST BYTE
172A  8A 44 01              4196    C           MOV     AL,[SI+1]               ; GET SECOND BYTE
172D  B9 C000               4197    C           MOV     CX,0C000H               ; 2 BIT MASK TO TEST
                            4198    C                                           ;  THE ENTRIES
1730  B2 00                 4199    C           MOV     DL,0                    ; RESULT REGISTER
1732                        4200    C   S24:
1732  85 C1                 4201    C           TEST    AX,CX                   ; IS THIS BACKGROUND?
1734  F8                    4202    C           CLC                             ; CLEAR CARRY IN HOPES
                            4203    C                                           ;  THAT IT IS
1735  74 01                 4204    C           JZ      S25                     ; IF 0  IT IS BACKGROUND
1737  F9                    4205    C           STC                             ; WASN'T, SO SET CARRY
1738                        4206    C   S25:
1738  D0 D2                 4207    C           RCL     DL,1                    ; MOVE THAT BIT INTO THE
173A  D1 E9                 4208    C           SHR     CX,1                    ;  RESULT
173C  D1 E9                 4209    C           SHR     CX,1                    ; MOVE THE MASK TO THE
                            4210    C                                           ;  RIGHT BY 2 BITS
173E  73 F2                 4211    C           JNC     S24                     ; DO IT AGAIN IF MASK
                            4212    C                                           ;  DIDN'T FALL OUT
1740  88 56 00              4213    C           MOV     [BP],DL                 ; STORE RESULT IN SAVE
1743  45                    4214    C           INC     BP                      ; ADJUST POINTER
1744  C3                    4215    C           RET                             ; ALL DONE
1745                        4216    C   S23     ENDP
                            4217    C
                            4218    C
1745                        4219    C   GRAPHICS_READ   PROC    NEAR
1745  E8 16EB R             4220    C           CALL    MK_ES
1748  E8 16A4 R             4221    C           CALL    S26                     ; CONVERTED TO OFFSET
174B  8B F0                 4222    C           MOV     SI,AX                   ; SAVE IN SI
174D  83 EC 08              4223    C           SUB     SP,8                    ; ALLOCATE SPACE TO SAVE
                            4224    C                                           ;  THE READ CODE POINT
1750  8B EC                 4225    C           MOV     BP,SP                   ; POINTER TO SAVE AREA
                            4226    C
                            4227    C   ;----- DETERMINE GRAPHICS MODES
                            4228    C
1752  80 3E 0449 R 06       4229    C           CMP     CRT_MODE,6
1757  06                    4230    C           PUSH    ES
1758  1F                    4231    C           POP     DS                      ; POINT TO REGEN SEGMENT
1759  72 1A                 4232    C           JC      S13P                    ; MEDIUM RESOLUTION
                            4233    C
                            4234    C   ;----- HIGH RESOLUTION READ
                            4235    C
                            4236    C   ;----- GET VALUES FROM REGEN BUFFER AND CONVERT TO CODE POINT
                            4237    C
175B  B6 04                 4238    C           MOV     DH,4                    ; NUMBER OF PASSES
175D                        4239    C   S12P:
175D  8A 04                 4240    C           MOV     AL,[SI]                 ; GET FIRST BYTE
175F  88 46 00              4241    C           MOV     [BP],AL                 ; SAVE IN STORAGE AREA
1762  45                    4242    C           INC     BP                      ; NEXT LOCATION
1763  8A 84 2000           4243    C           MOV     AL,[SI+2000H]           ; GET LOWER REGION BYTE
1767  88 46 00              4244    C           MOV     [BP],AL                 ; ADJUST AND STORE
176A  45                    4245    C           INC     BP
176B  83 C6 50              4246    C           ADD     SI,80                   ; POINTER INTO REGEN
176E  FE CE                 4247    C           DEC     DH                      ; LOOP CONTROL
1770  75 EB                 4248    C           JNZ     S12P                    ; DO IT SOME MORE
1772  EB 17 90              4249    C           JMP     S15P                    ; GO MATCH THE SAVED CODE
                            4250    C                                           ;  POINTS
                            4251    C
                            4252    C   ;----- MEDIUM RESOLUTION READ
                            4253    C
1775                        4254    C   S13P:                                   ; MED_RES_READ
1775  D1 E6                 4255    C           SAL     SI,1                    ; OFFSET*2, 2 BYTES/CHAR
1777  B6 04                 4256    C           MOV     DH,4                    ; NUMBER OF PASSES
1779                        4257    C   S14P:
1779  E8 1728 R             4258    C           CALL    S23                     ; GET PAIR BYTES
                            4259    C                                           ;  INTO SINGLE SAVE
177C  81 C6 2000           4260    C           ADD     SI,2000H                ; GO TO LOWER REGION
1780  E8 1728 R             4261    C           CALL    S23                     ; GET THIS PAIR INTO SAVE
1783  81 EE 1FB0            4262    C           SUB     SI,2000H-80             ; ADJUST POINTER BACK INTO
1787  FE CE                 4263    C           DEC     DH                      ;  UPPER
1789  75 EE                 4264    C           JNZ     S14P                    ; KEEP GOING UNTIL 8 DONE
                            4265    C
                            4266    C   ;----- SAVE AREA HAS CHARACTER IN IT, MATCH IT
                            4267    C
178B                        4268    C   S15P:                                   ; FIND_CHAR
178B  1E                    4269    C           PUSH    DS
178C  E8 0CFE R             4270    C           CALL    DDS
178F  C4 3E 010C R          4271    C           LES     DI,CRX_SET              ; ESTABLISH ADDRESSING
1793  1F                    4272    C           POP     DS
1794  83 ED 08              4273    C           SUB     BP,8                    ; ADJUST POINTER TO
                            4274    C                                           ;  BEGINNING OF SAVE AREA
1797  8B F5                 4275    C           MOV     SI,BP
1799  FC                    4276    C           CLD                             ; ENSURE DIRECTION
179A  B0 00                 4277    C           MOV     AL,0                    ; CURRENT CODE POINT BEING
179C                        4278    C   S16P:                                   ;  MATCHED
179C  16                    4279    C           PUSH    SS                      ; ADDRESSING TO STACK
179D  1F                    4280    C           POP     DS                      ; FOR THE STRING COMPARE
179E  BA 0080               4281    C           MOV     DX,128                  ; NUMBER TO TEST AGAINST
17A1                        4282    C   S17P:
17A1  56                    4283    C           PUSH    SI                      ; SAVE SAVE AREA POINTER
17A2  57                    4284    C           PUSH    DI                      ; SAVE CODE POINTER
```

```
17A3  B9 0008                 4285    C               MOV     CX,8                    ; NUMBER OF BYTES TO MATCH
17A6  F3/ A6                  4286    C               REPE    CMPSB                   ; COMPARE THE 8 BYTES
17A8  5F                      4287    C               POP     DI                      ; RECOVER THE POINTERS
17A9  5E                      4288    C               POP     SI
17AA  74 1D                   4289    C               JZ      S18P                    ; IF ZERO FLAG SET,
                              4290    C                                               ;  THEN MATCH OCCURRED
17AC  FE C0                   4291    C               INC     AL                      ; NO MATCH, MOVE TO NEXT
17AE  83 C7 08                4292    C               ADD     DI,8                    ; NEXT CODE POINT
17B1  4A                      4293    C               DEC     DX                      ; LOOP CONTROL
17B2  75 ED                   4294    C               JNZ     S17P                    ; DO ALL OF THEM
                              4295    C
                              4296    C       ;----- CHAR NOT MATCHED, MIGHT BE IN USER SUPPLIED SECOND HALF
                              4297    C
17B4  3C 00                   4298    C               CMP     AL,0                    ; AL <> 0 IF ONLY 1ST
                              4299    C                                               ;  HALF SCANNED
17B6  74 11                   4300    C               JE      S18P                    ; IF = 0, THEN ALL HAS
                              4301    C                                               ;  BEEN SCANNED
                              4302    C               ASSUME  DS:ABS0
17B8  E8 0CFE R               4303    C               CALL    DDS
17BB  C4 3E 007C R            4304    C               LES     DI,EXT_PTR              ; GET POINTER
17BF  8C C0                   4305    C               MOV     AX,ES                   ; SEE IF THE PNTR EXISTS
17C1  0B C7                   4306    C               OR      AX,DI                   ; IF ALL 0, DOESN'T EXIST
17C3  74 04                   4307    C               JZ      S18P                    ; NO SENSE LOOKING
17C5  B0 80                   4308    C               MOV     AL,128                  ; ORIGIN FOR SECOND HALF
17C7  EB D3                   4309    C               JMP     S16P                    ; GO BACK AND TRY FOR IT
                              4310    C
                              4311    C       ;----- CHARACTER IS FOUND ( AL=0 IF NOT FOUND )
                              4312    C
17C9                          4313    C       S18P:
17C9  83 C4 08                4314    C               ADD     SP,8                    ; READJUST THE STACK,
                              4315    C                                               ;  THROW AWAY SAVE
17CC  E9 219E R               4316    C               JMP     V_RET                   ; ALL DONE
17CF                          4317    C       GRAPHICS_READ   ENDP
                              4318    C
                              4319    C
                              4320    C       ;----- READ CHARACTER/ATTRIBUTE AT CURRENT CURSOR POSITION
                              4321    C
17CF                          4322    C       AH8S:
17CF  E9 1701 R               4323    C               JMP     READ_AC_CURRENT
                              4324    C
17D2                          4325    C       AH8:
                              4326    C               ASSUME  DS:ABS0
17D2  8A 26 0449 R            4327    C               MOV     AH,CRT_MODE             ; GET THE CURRENT MODE
17D6  80 FC 07                4328    C               CMP     AH,07H
17D9  74 F4                   4329    C               JE      AH8S
17DB  80 FC 03                4330    C               CMP     AH,03H
17DE  76 EF                   4331    C               JBE     AH8S
17E0  80 FC 06                4332    C               CMP     AH,06H
17E3  77 03                   4333    C               JA      Z_1
17E5  E9 1745 R               4334    C               JMP     GRAPHICS_READ
17E8                          4335    C       Z_1:
17E8  80 FC 0F                4336    C               CMP     AH,0FH
17EB  72 52                   4337    C               JB      GRX_RD2
17ED  E8 14F7 R               4338    C               CALL    MEM_DET
17F0  72 4D                   4339    C               JC      GRX_RD2
17F2  EB 0A                   4340    C               JMP     SHORT GRX_RD1
17F4  80 FC 0D                4341    C               CMP     AH,0DH                  ; RANGE TEST
17F7  73 46                   4342    C               JAE     GRX_RD2                 ; FOUR MAP READ
17F9  B0 00                   4343    C               MOV     AL,0
17FB  E9 219E R               4344    C               JMP     V_RET
                              4345    C
17FE                          4346    C       GRX_RD1 PROC    NEAR
                              4347    C               ASSUME  DS:ABS0
                              4348    C               SRLOAD  ES,0A000H               ; REGEN SEGEMNT
17FE  BA A000                 4349    C+              MOV     DX,0A000H
1801  8E C2                   4350    C+              MOV     ES,DX
1803  E8 16BA R               4351    C               CALL    GR_CUR                  ; BYTE OFFSET INTO REGEN
1806  8B F0                   4352    C               MOV     SI,AX                   ; SAVE IN SI
1808  8B 1E 0485 R            4353    C               MOV     BX,POINTS               ; BYTES PER CHARACTER
180C  2B E3                   4354    C               SUB     SP,BX                   ; ALLOCATE SPACE TO SAVE
                              4355    C                                               ;  THE READ CODE POINT
180E  8B EC                   4356    C               MOV     BP,SP                   ; POINTER TO SAVE AREA
                              4357    C
                              4358    C       ;----- GET VALUES FROM REGEN BUFFER AND CONVERT TO CODE POINT
                              4359    C
1810  53                      4360    C               PUSH    BX                      ; SAVE BYTES PER CHARACTER
1811  24 01                   4361    C               AND     AL,1                    ; ODD OR EVEN BYTE
1813  8A C8                   4362    C               MOV     CL,AL                   ; USE FOR SHIFT
1815  B0 05                   4363    C               MOV     AL,5                    ; COLOR COMP VALUE (C0-C2)
1817  D2 E0                   4364    C               SHL     AL,CL                   ; (C1-C3) IF ODD BYTE
1819  B4 07                   4365    C               MOV     AH,G_COLOR              ; COLOR COMPARE REGISTER
181B  B6 03                   4366    C               MOV     DH,3
181D  B2 CE                   4367    C               MOV     DL,GRAPH_ADDR
181F  E8 0D15 R               4368    C               CALL    OUT_DX                  ; SET GRAPHICS CHIP
1822  B8 0518                 4369    C               MOV     AX,518H                 ; READ MODE
1825  E8 0D15 R               4370    C               CALL    OUT_DX                  ; SET GRAPHICS CHIP
1828                          4371    C       S12_1:
1828  26: 8A 04               4372    C               MOV     AL,ES:[SI]              ; GET FIRST BYTE
182B  F6 D0                   4373    C               NOT     AL
182D  88 46 00                4374    C               MOV     SS:[BP],AL              ; SAVE IN STORAGE AREA
1830  45                      4375    C               INC     BP                      ; NEXT LOCATION
1831  03 36 044A R            4376    C               ADD     SI,CRT_COLS             ; POINTER INTO REGEN
1835  4B                      4377    C               DEC     BX                      ; LOOP CONTROL
1836  75 F0                   4378    C               JNZ     S12_1                   ; DO IT SOME MORE
1838  5B                      4379    C               POP     BX                      ; RECOVER BYTES PER CHAR
1839  B8 0510                 4380    C               MOV     AX,510H                 ; UNDO READ MODE
183C  EB 32 90                4381    C               JMP     GRX_RECG                ; CHAR REGONTION ROUTINE
183F                          4382    C       GRX_RD1 ENDP
                              4383    C
183F                          4384    C       GRX_RD2 PROC    NEAR
                              4385    C               ASSUME  DS:ABS0
                              4386    C               SRLOAD  ES,0A000H               ; REGEN SEGMENT
183F  BA A000                 4387    C+              MOV     DX,0A000H
1842  8E C2                   4388    C+              MOV     ES,DX
1844  E8 16BA R               4389    C               CALL    GR_CUR                  ; BYTE OFFSET INTO REGEN
1847  8B F0                   4390    C               MOV     SI,AX                   ; SAVE IN SI
1849  8B 1E 0485 R            4391    C               MOV     BX,POINTS               ; BYTES PER CHARACTER
184D  2B E3                   4392    C               SUB     SP,BX                   ; ALLOCATE SPACE TO SAVE
                              4393    C                                               ;  THE READ CODE POINT
184F  8B EC                   4394    C               MOV     BP,SP                   ; POINTER TO SAVE AREA
                              4395    C
                              4396    C       ;----- GET VALUES FROM REGEN BUFFER AND CONVERT TO CODE POINT
                              4397    C
1851  B6 03                   4398    C               MOV     DH,3
1853  B2 CE                   4399    C               MOV     DL,GRAPH_ADDR           ; GRAPHICS CHIP
1855  B8 0508                 4400    C               MOV     AX,508H                 ; COLOR COMPARE
1858  E8 0D15 R               4401    C               CALL    OUT_DX                  ; SET THE REGISTER
185B  53                      4402    C               PUSH    BX                      ; SAVE BYTES PER CHARACTER
185C                          4403    C       S12:
185C  26: 8A 04               4404    C               MOV     AL,ES:[SI]              ; GET COLOR COMPARED BYTE
185F  F6 D0                   4405    C               NOT     AL                      ; ADJUST
1861  88 46 00                4406    C               MOV     SS:[BP],AL              ; SAVE IN STORAGE AREA
1864  45                      4407    C               INC     BP                      ; NEXT LOCATION
1865  03 36 044A R            4408    C               ADD     SI,CRT_COLS             ; POINTER INTO REGEN
1869  4B                      4409    C               DEC     BX                      ; LOOP CONTROL
186A  75 F0                   4410    C               JNZ     S12                     ; DO IT SOME MORE
```

```
186C  5B                    4411     C          POP     BX                              ; RECOVER BYTES PER CHAR
186D  B8 0500               4412     C          MOV     AX,500H                         ; UNDO READ MODE
1870                        4413     C  GRX_RD2 ENDP
                            4414     C
1870                        4415     C  GRX_RECG:
                            4416     C
                            4417     C  ;----- SAVE AREA HAS CHARACTER IN IT, MATCH IT
                            4418     C
1870  E8 0D15 R             4419     C          CALL    OUT_DX                          ; SET READ MODE BACK
1873  C4 3E 010C R          4420     C          LES     DI,GRX_SET                      ; GET FONT DEFINITIONS
1877  2B EB                 4421     C          SUB     BP,BX                           ; ADJUST POINTER TO
                            4422     C                                                  ;  BEGINNING OF SAVE AREA
1879  8B F5                 4423     C          MOV     SI,BP
187B  FC                    4424     C          CLD                                     ; ENSURE DIRECTION
187C  B0 00                 4425     C          MOV     AL,0                            ; CODE POINT BEING MATCHED
187E  16                    4426     C          PUSH    SS                              ; ADDRESSING TO STACK
187F  1F                    4427     C          POP     DS                              ;  FOR THE STRING COMPARE
1880  BA 0100               4428     C          MOV     DX,256D                         ; NUMBER TO TEST AGAINST
1883                        4429     C  S17_5:
1883  56                    4430     C          PUSH    SI                              ; SAVE SAVE AREA POINTER
1884  57                    4431     C          PUSH    DI                              ; SAVE CODE POINTER
1885  8B CB                 4432     C          MOV     CX,BX                           ; NUMBER OF BYTES TO MATCH
1887  F3/ A6                4433     C          REPE    CMPSB                           ; COMPARE THE 8 BYTES
1889  5F                    4434     C          POP     DI                              ; RECOVER THE POINTERS
188A  5E                    4435     C          POP     SI
188B  74 07                 4436     C          JZ      S18_5                           ; IF ZFL SET, THEN MATCH
                            4437     C                                                  ;  OCCURRED
188D  FE C0                 4438     C          INC     AL                              ; NO MATCH, ON TO NEXT
188F  03 FB                 4439     C          ADD     DI,BX                           ; NEXT CODE POINT
1891  4A                    4440     C          DEC     DX                              ; LOOP CONTROL
1892  75 EF                 4441     C          JNZ     S17_5                           ; DO ALL OF THEM
1894                        4442     C  S18_5:                                          ; AL=CHAR, 0 IF NOT FOUND
1894  03 E3                 4443     C          ADD     SP,BX                           ; READJUST THE STACK
1896  E9 219E R             4444     C          JMP     V_RET
                            4445     C
                            4446     C  ;----- WRITE CHARACTER/ATTRIBUTE AT CURRENT CURSOR POSITION
                            4447     C
                            4448     C  ;--------------------------------------------
                            4449     C  ; WRITE_AC_CURRENT                          :
                            4450     C  ;       THIS ROUTINE WRITES THE ATTRIBUTE   :
                            4451     C  ;       AND CHARACTER AT THE CURRENT CURSOR :
                            4452     C  ;       POSITION                            :
                            4453     C  ; INPUT                                     :
                            4454     C  ;       (AH) = CURRENT CRT MODE             :
                            4455     C  ;       (BH) = DISPLAY PAGE                 :
                            4456     C  ;       (CX) = COUNT OF CHARACTERS TO WRITE :
                            4457     C  ;       (AL) = CHAR TO WRITE                :
                            4458     C  ;       (BL) = ATTRIBUTE OF CHAR TO WRITE   :
                            4459     C  ;       (DS) = DATA SEGMENT                 :
                            4460     C  ;       (ES) = REGEN SEGMENT                :
                            4461     C  ; OUTPUT                                    :
                            4462     C  ;       NONE                                :
                            4463     C  ;--------------------------------------------
1899                        4464     C  AH9:
                            4465     C          ASSUME  DS:ABS0
1899  E8 0CFE R             4466     C          CALL    DDS
189C  8A 26 0449 R          4467     C          MOV     AH,CRT_MODE
                            4468     C
18A0  80 FC 04              4469     C          CMP     AH,4                            ; IS THIS GRAPHICS
18A3  72 08                 4470     C          JC      P6
18A5  80 FC 07              4471     C          CMP     AH,7                            ; IS THIS BW CARD
18A8  74 03                 4472     C          JE      P6
18AA  EB 74 90              4473     C          JMP     GRAPHICS_WRITE
18AD                        4474     C  P6:                                             ; WRITE_AC_CONTINUE
18AD  E8 16EB R             4475     C          CALL    MK_ES
18B0  8A E3                 4476     C          MOV     AH,BL                           ; GET ATTRIBUTE TO AH
18B2  50                    4477     C          PUSH    AX                              ; SAVE ON STACK
18B3  51                    4478     C          PUSH    CX                              ; SAVE WRITE COUNT
18B4  E8 1651 R             4479     C          CALL    FIND_POSITION
18B7  8B FB                 4480     C          MOV     DI,BX                           ; ADDRESS TO DI REGISTER
18B9  59                    4481     C          POP     CX                              ; WRITE COUNT
18BA  5B                    4482     C          POP     BX                              ; CHARACTER IN BX REG
18BB  8B 16 0463 R          4483     C          MOV     DX,ADDR_6845                    ; GET BASE ADDRESS
18BF  83 C2 06              4484     C          ADD     DX,6                            ; POINT AT STATUS PORT
                            4485     C
                            4486     C  ;----- WAIT FOR HORIZONTAL RETRACE
                            4487     C
18C2                        4488     C  P7:
18C2  F6 06 0487 R 04       4489     C          TEST    INFO,4
18C7  74 0B                 4490     C          JZ      P9A
18C9                        4491     C  P8:
18C9  EC                    4492     C          IN      AL,DX                           ; GET STATUS
18CA  A8 01                 4493     C          TEST    AL,1                            ; IS IT LOW
18CC  75 FB                 4494     C          JNZ     P8                              ; WAIT UNTIL IT IS
18CE  FA                    4495     C          CLI                                     ; NO MORE INTERRUPTS
18CF                        4496     C  P9:
18CF  EC                    4497     C          IN      AL,DX                           ; GET STATUS
18D0  A8 01                 4498     C          TEST    AL,1                            ; IS IT HIGH
18D2  74 FB                 4499     C          JZ      P9                              ; WAIT UNTIL IT IS
18D4                        4500     C  P9A:
18D4  8B C3                 4501     C          MOV     AX,BX                           ; RECOVER THE CHAR/ATTR
18D6  AB                    4502     C          STOSW                                   ; PUT THE CHAR/ATTR
18D7  FB                    4503     C          STI                                     ; INTERRUPTS BACK ON
18D8  E2 E8                 4504     C          LOOP    P7                              ; AS MANY TIMES
18DA  E9 219E R             4505     C          JMP     V_RET
                            4506     C
                            4507     C  ;----- WRITE CHARACTER ONLY AT CURRENT CURSOR POSITION
                            4508     C
                            4509     C  ;--------------------------------------------
                            4510     C  ; WRITE_C_CURRENT                           :
                            4511     C  ;       THIS ROUTINE WRITES THE CHARACTER AT :
                            4512     C  ;       THE CURRENT CURSOR POSITION, ATTRIBUTE :
                            4513     C  ;       UNCHANGED                           :
                            4514     C  ; INPUT                                     :
                            4515     C  ;       (AH) = CURRENT CRT MODE             :
                            4516     C  ;       (BH) = DISPLAY PAGE                 :
                            4517     C  ;       (CX) = COUNT OF CHARACTERS TO WRITE :
                            4518     C  ;       (AL) = CHAR TO WRITE                :
                            4519     C  ;       (DS) = DATA SEGMENT                 :
                            4520     C  ;       (ES) = REGEN SEGMENT                :
                            4521     C  ; OUTPUT                                    :
                            4522     C  ;       NONE                                :
                            4523     C  ;--------------------------------------------
18DD                        4524     C  AH4:
                            4525     C          ASSUME  DS:ABS0
18DD  E8 0CFE R             4526     C          CALL    DDS
18E0  8A 26 0449 R          4527     C          MOV     AH,CRT_MODE
                            4528     C
18E4  80 FC 04              4529     C          CMP     AH,4                            ; IS THIS GRAPHICS
18E7  72 08                 4530     C          JC      P10
18E9  80 FC 07              4531     C          CMP     AH,7                            ; IS THIS BW CARD
18EC  74 03                 4532     C          JE      P10
                            4533     C
18EE  EB 30 90              4534     C          JMP     GRAPHICS_WRITE
18F1                        4535     C  P10:
18F1  E8 16EB R             4536     C          CALL    MK_ES
```

```
18F4  50                    4537  C          PUSH    AX                              ; SAVE ON STACK
18F5  51                    4538  C          PUSH    CX                              ; SAVE WRITE COUNT
18F6  E8 1651 R             4539  C          CALL    FIND_POSITION
18F9  8B FB                 4540  C          MOV     DI,BX                           ; ADDRESS TO DI
18FB  59                    4541  C          POP     CX                              ; WRITE COUNT
18FC  5B                    4542  C          POP     BX                              ; BL HAS CHAR TO WRITE
                            4543  C
                            4544  C ;----- WAIT FOR HORIZONTAL RETRACE
                            4545  C
18FD  8B 16 0463 R          4546  C          MOV     DX,ADDR_6845                    ; GET BASE ADDRESS
1901  83 C2 06              4547  C          ADD     DX,6                            ; POINT AT STATUS PORT
1904                        4548  C P11:
1904  F6 06 0487 R 04       4549  C          TEST    INFO,4
1909  74 0B                 4550  C          JZ      P13A
190B                        4551  C P12:
190B  EC                    4552  C          IN      AL,DX                           ; GET STATUS
190C  A8 01                 4553  C          TEST    AL,1                            ; IS IT LOW
190E  75 FB                 4554  C          JNZ     P12                             ; WAIT UNTIL IT IS
1910  FA                    4555  C          CLI                                     ; NO MORE INTERRUPTS
1911                        4556  C P13:
1911  EC                    4557  C          IN      AL,DX                           ; GET STATUS
1912  A8 01                 4558  C          TEST    AL,1                            ; IS IT HIGH
1914  74 FB                 4559  C          JZ      P13                             ; WAIT UNTIL IT IS
1916                        4560  C P13A:
1916  8A C3                 4561  C          MOV     AL,BL                           ; RECOVER CHAR
1918  AA                    4562  C          STOSB                                   ; PUT THE CHAR/ATTR
1919  FB                    4563  C          STI                                     ; INTERRUPTS BACK ON
191A  47                    4564  C          INC     DI                              ; BUMP POINTER PAST ATTR
191B  E2 E7                 4565  C          LOOP    P11                             ;  AS REQUESTED
191D  E9 219E R             4566  C          JMP     V_RET
                            4567  C
                            4568  C ;--------------------------------------------------------------------
                            4569  C ; GRAPHICS WRITE                                                    :
                            4570  C ;       THIS ROUTINE WRITES THE ASCII CHARACTER TO THE              :
                            4571  C ;       CURRENT POSITION ON THE SCREEN.                             :
                            4572  C ; ENTRY                                                             :
                            4573  C ;       AL = CHARACTER TO WRITE                                     :
                            4574  C ;       BL = COLOR ATTRIBUTE TO BE USED FOR FOREGROUND COLOR        :
                            4575  C ;         IF BIT 7 IS SET, THE CHAR IS XOR'D INTO THE REGEN         :
                            4576  C ;         BUFFER (0 IS USED FOR THE BACKGROUND COLOR)               :
                            4577  C ;       CX = NUMBER OF CHARS TO WRITE                               :
                            4578  C ;       DS = DATA SEGMENT                                           :
                            4579  C ;       ES = REGEN SEGMENT                                          :
                            4580  C ; EXIT                                                              :
                            4581  C ;       NOTHING IS RETURNED                                         :
                            4582  C ;                                                                   :
                            4583  C ; GRAPHICS READ                                                     :
                            4584  C ;       THIS ROUTINE READS THE ASCII CHARACTER AT THE CURRENT        :
                            4585  C ;       CURSOR POSITION ON THE SCREEN BY MATCHING THE DOTS ON        :
                            4586  C ;       THE SCREEN TO THE CHARACTER GENERATOR CODE POINTS            :
                            4587  C ; ENTRY                                                             :
                            4588  C ;       NONE  (0 IS ASSUMED AS THE BACKGROUND COLOR)                :
                            4589  C ; EXIT                                                              :
                            4590  C ;       AL = CHARACTER READ AT THAT POSITION (0 RETURNED IF          :
                            4591  C ;            NONE FOUND)                                            :
                            4592  C ;                                                                   :
                            4593  C ; FOR COMPATIBILITY ROUTINES, THE IMAGES USED TO FORM CHARS ARE     :
                            4594  C ;  CONTAINED IN ROM FOR THE 1ST 128 CHARS.  TO ACCESS CHARS         :
                            4595  C ;  IN THE SECOND HALF, THE USER MUST INITIALIZE THE VECTOR AT       :
                            4596  C ;  INTERRUPT 1FH (LOCATION 0007CH) TO POINT TO THE USER             :
                            4597  C ;  SUPPLIED TABLE OF GRAPHIC IMAGES (8X8 BOXES).                    :
                            4598  C ;  FAILURE TO DO SO WILL CAUSE IN STRANGE RESULTS                   :
                            4599  C ;--------------------------------------------------------------------
                            4600  C          ASSUME  CS:CODE,DS:ABS0,ES:NOTHING
1920                        4601  C GRAPHICS_WRITE  PROC    NEAR
1920  80 FC 07              4602  C          CMP     AH,7
1923  72 03                 4603  C          JB      S1_A
1925  E9 19D7 R             4604  C          JMP     GRX_WRT
                            4605  C S1_A:
1928  E8 16EB R             4606  C          CALL    MK_ES
192B  B4 00                 4607  C          MOV     AH,0                            ; 0 TO HIGH OF CODE POINT
192D  50                    4608  C          PUSH    AX                              ; SAVE CODE POINT VALUE
                            4609  C
                            4610  C ;----- DETERMINE POSITION IN REGEN BUFFER TO PUT CODE POINTS
                            4611  C
192E  E8 16A4 R             4612  C          CALL    S26                             ; LOC IN REGEN BUFFER
1931  8B F8                 4613  C          MOV     DI,AX                           ; REGEN POINTER IN DI
                            4614  C
                            4615  C ;----- DETERMINE REGION TO GET CODE POINTS FROM
                            4616  C
1933  58                    4617  C          POP     AX                              ; RECOVER CODE POINT
1934  3C 80                 4618  C          CMP     AL,80H                          ; IS IT IN SECOND HALF
1936  73 06                 4619  C          JAE     S1                              ; YES
                            4620  C
                            4621  C ;----- IMAGE IS IN FIRST HALF, CONTAINED IN ROM
                            4622  C
1938  C5 36 010C R          4623  C          LDS     SI,GRX_SET
193C  EB 06                 4624  C          JMP     SHORT S2                        ; DETERMINE_MODE
                            4625  C
                            4626  C ;----- IMAGE IS IN SECOND HALF, IN USER RAM
                            4627  C
193E                        4628  C S1:                                              ; EXTEND_CHAR
193E  2C 80                 4629  C          SUB     AL,80H                          ; 0 ORIGIN FOR SECOND HALF
1940  C5 36 007C R          4630  C          LDS     SI,EXT_PTR
                            4631  C
                            4632  C ;----- DETERMINE GRAPHICS MODE IN OPERATION
                            4633  C
1944                        4634  C S2:                                              ; DETERMINE_MODE
1944  D1 E0                 4635  C          SAL     AX,1                            ; MULTIPLY CODE POINT
1946  D1 E0                 4636  C          SAL     AX,1                            ;  VALUE BY 8
1948  D1 E0                 4637  C          SAL     AX,1
194A  03 F0                 4638  C          ADD     SI,AX                           ; SI HAS OFFSET OF
194C  1E                    4639  C          PUSH    DS                              ;  DESIRES CODES
194D  E8 0CFE R             4640  C          CALL    DDS
1950  80 3E 0449 R 06       4641  C          CMP     CRT_MODE,6
1955  1F                    4642  C          POP     DS
1956  72 2C                 4643  C          JC      S7                              ; TEST FOR MEDIUM RES MODE
                            4644  C
                            4645  C ;----- HIGH RESOLUTION MODE
                            4646  C
1958                        4647  C S3:                                              ; HIGH_CHAR
1958  57                    4648  C          PUSH    DI                              ; SAVE REGEN POINTER
1959  56                    4649  C          PUSH    SI                              ; SAVE CODE POINTER
195A  B6 04                 4650  C          MOV     DH,4                            ; NUMBER OF TIMES THROUGH
195C                        4651  C S4:                                              ;  LOOP
195C  AC                    4652  C          LODSB                                   ; GET BYTE FROM CODE POINT
195D  F6 C3 80              4653  C          TEST    BL,80H                          ; SHOULD WE USE THE
1960  75 16                 4654  C          JNZ     S6                              ;  FUNCTION TO PUT CHAR IN
1962  AA                    4655  C          STOSB                                   ; STORE IN REGEN BUFFER
1963  AC                    4656  C          LODSB
1964                        4657  C S5:
1964  26: 88 85 1FFF        4658  C          MOV     ES:[DI+2000H-1],AL              ; STORE IN SECOND HALF
1969  83 C7 4F              4659  C          ADD     DI,79                           ; MOVE TO NEXT ROW IN REGEN
196C  FE CE                 4660  C          DEC     DH                              ; DONE WITH LOOP
196E  75 EC                 4661  C          JNZ     S4
1970  5E                    4662  C          POP     SI
```

```
1971  5F                    4663  C          POP     DI                          ; RECOVER REGEN POINTER
1972  47                    4664  C          INC     DI                          ; POINT TO NEXT CHAR POS
1973  E2 E3                 4665  C          LOOP    S3                          ; MORE CHARS TO WRITE
1975  E9 219E R             4666  C          JMP     V_RET
1978                        4667  C  S6:
1978  26: 32 05             4668  C          XOR     AL,ES:[DI]                  ; XOR WITH CURRENT
197B  AA                    4669  C          STOSB                               ; STORE THE CODE POINT
197C  AC                    4670  C          LODSB                               ; AGAIN FOR ODD FIELD
197D  26: 32 85 1FFF        4671  C          XOR     AL,ES:[DI+2000H-1]
1982  EB E0                 4672  C          JMP     S5                          ; BACK TO MAINSTREAM
                            4673  C
                            4674  C  ;----- MEDIUM RESOLUTION WRITE
                            4675  C
1984                        4676  C  S7:                                         ; MED_RES_WRITE
1984  8A D3                 4677  C          MOV     DL,BL                       ; SAVE HIGH COLOR BIT
1986  D1 E7                 4678  C          SAL     DI,1                        ; OFFSET*2, 2 BYTES/CHAR
1988  E8 166D R             4679  C          CALL    S19                         ; EXPAND BL TO FULL WORD
198B                        4680  C  S8:                                         ;  OF COLOR
198B  57                    4681  C          PUSH    DI                          ; SAVE REGEN POINTER
198C  56                    4682  C          PUSH    SI                          ; SAVE THE CODE POINTER
198D  B6 04                 4683  C          MOV     DH,4                        ; NUMBER OF LOOPS
198F                        4684  C  S9:
198F  AC                    4685  C          LODSB                               ; GET CODE POINT
1990  E8 1682 R             4686  C          CALL    S21                         ; DOUBLE UP ALL THE BITS
1993  23 C3                 4687  C          AND     AX,BX                       ; CONVERT THEM TO FORE-
                            4688  C                                              ;  GROUND COLOR (0 BACK)
1995  F6 C2 80              4689  C          TEST    DL,80H                      ; IS THIS XOR FUNCTION
1998  74 07                 4690  C          JZ      S10                         ; NO, STORE IT IN AS IT IS
199A  26: 32 25             4691  C          XOR     AH,ES:[DI]                  ; DO FUNCTION WITH HALF
199D  26: 32 45 01          4692  C          XOR     AL,ES:[DI+1]                ;  AND WITH OTHER HALF
19A1                        4693  C  S10:
19A1  26: 88 25             4694  C          MOV     ES:[DI],AH                  ; STORE FIRST BYTE
19A4  26: 88 45 01          4695  C          MOV     ES:[DI+1],AL                ; STORE SECOND BYTE
19A8  AC                    4696  C          LODSB                               ; GET CODE POINT
19A9  E8 1682 R             4697  C          CALL    S21
19AC  23 C3                 4698  C          AND     AX,BX                       ; CONVERT TO COLOR
19AE  F6 C2 80              4699  C          TEST    DL,80H                      ; IS THIS XOR FUNCTION
19B1  74 0A                 4700  C          JZ      S11                         ; NO, JUST STORE THE VALUE
19B3  26: 32 A5 2000        4701  C          XOR     AH,ES:[DI+2000H]            ; FUNCTION WITH FIRST HALF
19B8  26: 32 85 2001        4702  C          XOR     AL,ES:[DI+2001H]            ; AND WITH SECOND HALF
19BD                        4703  C  S11:
19BD  26: 88 A5 2000        4704  C          MOV     ES:[DI+2000H],AH
19C2  26: 88 85 2001        4705  C          MOV     ES:[DI+2000H+1],AL          ; STORE IN SECOND PORTION
19C7  83 C7 50              4706  C          ADD     DI,80                       ; POINT TO NEXT LOCATION
19CA  FE CE                 4707  C          DEC     DH
19CC  75 C1                 4708  C          JNZ     S9                          ; KEEP GOING
19CE  5E                    4709  C          POP     SI                          ; RECOVER CODE PONTER
19CF  5F                    4710  C          POP     DI                          ; RECOVER REGEN POINTER
19D0  47                    4711  C          INC     DI                          ; POINT TO NEXT CHAR
19D1  47                    4712  C          INC     DI
19D2  E2 B7                 4713  C          LOOP    S8                          ; MORE TO WRITE
19D4  E9 219E R             4714  C          JMP     V_RET
19D7                        4715  C  GRAPHICS_WRITE  ENDP
                            4716  C
                            4717  C  ;--------------------------------------------------------------
                            4718  C  ; ENTRY                                                      :
                            4719  C  ;       AL = CHAR TO WRITE                                    :
                            4720  C  ;       BH = DISPLAY PAGE                                     :
                            4721  C  ;       BL = ATTRIBUTE/COLOR                                  :
                            4722  C  ;       CX = COUNT OF CHARS TO WRITE                          :
                            4723  C  ;--------------------------------------------------------------
19D7                        4724  C  GRX_WRT PROC    NEAR
                            4725  C          ASSUME  DS:ABS0, ES:NOTHING
19D7  80 FC 0F              4726  C          CMP     AH,0FH                      ; 640X350 GRAPHICS
19DA  72 0E                 4727  C          JB      NO_ADJ1
19DC  E8 14F7 R             4728  C          CALL    MEM_DET                     ; BASE CARD
19DF  72 09                 4729  C          JC      NO_ADJ1
19E1  80 E3 85              4730  C          AND     BL,10000101B                ; 85H, XOR C2 CO MASK
19E4  8A E3                 4731  C          MOV     AH,BL
19E6  D0 E4                 4732  C          SHL     AH,1                        ; EXPAND CO TO C1, C2 TO C3
19E8  0A DC                 4733  C          OR      BL,AH                       ; BUILD ?(80H) + (0,3,C,F)
19EA                        4734  C  NO_ADJ1:
19EA  2A E4                 4735  C          SUB     AH,AH                       ; ZERO
19EC  F7 26 0485 R          4736  C          MUL     POINTS                      ; OFFSET FONT TABLE BASE
19F0  50                    4737  C          PUSH    AX                          ; FONT TABLE DISPLACEMENT
19F1  E8 16BA R             4738  C          CALL    GR_CUR                      ; GET OFFSET INTO REGEN
19F4  8B F8                 4739  C          MOV     DI,AX                       ;  INTO DESTINATION
19F6  8B 2E 0485 R          4740  C          MOV     BP,POINTS                   ; BYTES PER CHAR
                            4741  C          SRLOAD  ES,0A000H                   ; REGEN SEGEMNT
19FA  BA A000               4742  C+         MOV     DX,0A000H
19FD  8E C2                 4743  C+         MOV     ES,DX
19FF  C5 36 010C R          4744  C          LDS     SI,GRX_SET                  ; ADDRESSING TO FONTS
1A03  58                    4745  C          POP     AX                          ; RECOVER OFFSET
1A04  03 F0                 4746  C          ADD     SI,AX                       ; CHARACTER IN TABLE
1A06  B6 03                 4747  C          MOV     DH,3
1A08                        4748  C  S20A:
1A08  F6 C3 80              4749  C          TEST    BL,080H                     ; TEST FOR XOR
1A0B  74 0B                 4750  C          JZ      NO_XOR                      ; NO XOR
1A0D  B2 CE                 4751  C          MOV     DL,GRAPH_ADDR
1A0F  B8 0318               4752  C          MOV     AX,0318H                    ; GRAPHICS CHIP XOR
1A12  E8 0D15 R             4753  C          CALL    OUT_DX                      ; SET REGISTER
1A15  EB 1E 90              4754  C          JMP     F_2                         ; SKIP BLANK
1A18                        4755  C  NO_XOR:                                     ; BLANK BOX FOR CHAR
1A18  57                    4756  C          PUSH    DI                          ; SAVE REGEN POINTER
1A19  B2 C4                 4757  C          MOV     DL,SEQ_ADDR
1A1B  B8 020F               4758  C          MOV     AX,020FH                    ; ENABLE ALL MAPS
1A1E  E8 0D15 R             4759  C          CALL    OUT_DX
1A21  2B C0                 4760  C          SUB     AX,AX                       ; STORE ZERO
1A23  51                    4761  C          PUSH    CX                          ; SAVE CHARACTER COUNT
1A24  8B CD                 4762  C          MOV     CX,BP                       ; GET BYTE COUNT
1A26  1E                    4763  C          PUSH    DS
1A27  E8 0CFE R             4764  C          CALL    DDS
1A2A                        4765  C  S13A:
1A2A  AA                    4766  C          STOSB                               ; ZERO REGEN BYTE
1A2B  03 3E 044A R          4767  C          ADD     DI,CRT_COLS                 ; NEXT BYTE OF BOX
1A2F  4F                    4768  C          DEC     DI                          ; ADJUST
1A30  E2 F8                 4769  C          LOOP    S13A                        ; NEXT BYTE
1A32  1F                    4770  C          POP     DS
1A33  59                    4771  C          POP     CX                          ; RECOVER CHARACTER COUNT
1A34  5F                    4772  C          POP     DI                          ; RECOVER REGEN POINTER
1A35                        4773  C  F_2:
1A35  B2 C4                 4774  C          MOV     DL,SEQ_ADDR
1A37  B4 02                 4775  C          MOV     AH,02H                      ; SET MAP MASK
1A39  8A C3                 4776  C          MOV     AL,BL                       ;  FOR COLOR
1A3B  E8 0D15 R             4777  C          CALL    OUT_DX                      ; SET THE CHIP
1A3E  57                    4778  C          PUSH    DI                          ; SAVE OFFSET IN REGEN
1A3F  53                    4779  C          PUSH    BX                          ; SAVE COLOR VALUE
1A40  51                    4780  C          PUSH    CX                          ; SAVE CHACTER COUNT
1A41  8B DD                 4781  C          MOV     BX,BP                       ; LOOP CONTROL, BYTES/CHAR
1A43  1E                    4782  C          PUSH    DS                          ; SAVE FONT SEGMENT
1A44  E8 0CFE R             4783  C          CALL    DDS                         ; SET LOW RAM SEGMENT
                            4784  C          ASSUME  DS:ABS0
1A47  8B 0E 044A R          4785  C          MOV     CX,CRT_COLS                 ; GET COLUMN COUNT
1A4B  1F                    4786  C          POP     DS                          ; RESTORE FONT SEGMENT
                            4787  C          ASSUME  DS:NOTHING
1A4C                        4788  C  S1K:                                        ; WRITE OUT THE CHARACTER
```

```
1A4C  8A 04                    4789  C           MOV     AL,DS:[SI]                      ; CODE POINT
1A4E  26: 8A 25                4790  C           MOV     AH,ES:[DI]                      ; LATCH DATA
1A51  26: 88 05                4791  C           MOV     ES:[DI],AL                      ; WRITE ONE BYTE OF FONT
1A54  46                       4792  C           INC     SI                              ; NEXT FONT POINT
1A55  03 F9                    4793  C           ADD     DI,CX                           ; ONE ROW BELOW LAST POINT
1A57  4B                       4794  C           DEC     BX                              ; BYTES PER CHAR COUNTER
1A58  75 F2                    4795  C           JNZ     S1K                             ; DO NEXT ROW OF CHARACTER
                               4796  C
1A5A  59                       4797  C           POP     CX                              ; CHARACTER COUNT
1A5B  5B                       4798  C           POP     BX                              ; COLOR VALUE
1A5C  2B F5                    4799  C           SUB     SI,BP                           ; ADJUST PTR TO FONT TABLE
1A5E  5F                       4800  C           POP     DI                              ; REGEN POINTER
1A5F  47                       4801  C           INC     DI                              ; NEXT CHAR POSN IN REGEN
1A60  E2 A6                    4802  C           LOOP    S20A                            ; WRITE ANOTHER CHARACTER
                               4803  C
1A62  B2 CE                    4804  C           MOV     DL,GRAPH_ADDR
1A64  B8 0300                  4805  C           MOV     AX,0300H                        ; NORMAL WRITE, NO ROTATE
1A67  E8 0D15 R                4806  C           CALL    OUT_DX                          ; SET THE CHIP
1A6A  B2 C4                    4807  C           MOV     DL,SEQ_ADDR
1A6C  B8 020F                  4808  C           MOV     AX,020FH                        ; ENABLE ALL MAPS
1A6F  E8 0D15 R                4809  C           CALL    OUT_DX                          ; SET THE CHIP
1A72  E9 219E R                4810  C           JMP     V_RET
1A75                           4811  C     GRX_WRT ENDP
                               4812  C
                               4813  C           SUBTTL
                               4814
                               4815        ;----- SET COLOR PALETTE
                               4816
1A75                           4817        AHB:
                               4818              ASSUME  DS:ABS0
1A75  80 3E 0463 R B4          4819              CMP     BYTE PTR ADDR_6845,0B4H
1A7A  74 09                    4820              JE      M21_B                           ; CALL VALID ONLY FOR COLOR
1A7C  F6 06 0487 R 02          4821              TEST    INFO,2                          ; SEE IF ITS THE OLD COLOR CARD
1A81  74 05                    4822              JZ      M21_A                           ; IF NOT, HANDLE IT HERE
1A83  CD 42                    4823              INT     42H                             ; OLD CODE CALL
1A85                           4824        M21_B:
1A85  E9 219E R                4825              JMP     V_RET                           ; BACK TO CALLER
1A88                           4826        M21_A:
1A88  2B C0                    4827              SUB     AX,AX
1A8A  8B E8                    4828              MOV     BP,AX
1A8C  C4 3E 04A8 R             4829              LES     DI,SAVE_PTR
1A90  83 C7 04                 4830              ADD     DI,4
1A93  26: C4 3D                4831              LES     DI,DWORD PTR ES:[DI]
1A96  8C C0                    4832              MOV     AX,ES
1A98  0B C7                    4833              OR      AX,DI
1A9A  74 01                    4834              JZ      NOT4AHB
1A9C  45                       4835              INC     BP
1A9D                           4836        NOT4AHB:
1A9D  E8 1DC0 R                4837              CALL    PAL_INIT
1AA0  0A FF                    4838              OR      BH,BH
1AA2  75 65                    4839              JNZ     M20
                               4840
                               4841        ;----- HANDLE BH = 0 HERE
                               4842        ;       ALPHA MODES  => BL = OVERSCAN COLOR
                               4843        ;       GRAPHICS     => BL = OVERSCAN AND BACKGROUND COLOR
                               4844
                               4845        ;----- MOVE INTENSITY BIT FROM D3 TO D4 FOR COMPATIBILITY
                               4846
1AA4  8A FB                    4847              MOV     BH,BL
1AA6  A0 0466 R                4848              MOV     AL,CRT_PALETTE
1AA9  24 E0                    4849              AND     AL,0E0H
1AAB  80 E3 1F                 4850              AND     BL,01FH
1AAE  0A C3                    4851              OR      AL,BL
1AB0  A2 0466 R                4852              MOV     CRT_PALETTE,AL
1AB3  8A DF                    4853              MOV     BL,BH
1AB5  80 E7 08                 4854              AND     BH,08H
1AB8  D0 E7                    4855              SHL     BH,1
1ABA  8A E8                    4856              MOV     CH,AL
1ABC  80 E5 EF                 4857              AND     CH,0EFH
1ABF  0A ED                    4858              OR      CH,CH
1AC1  80 E3 0F                 4859              AND     BL,0FH
1AC4  8A FB                    4860              MOV     BH,BL
1AC6  D0 E3                    4861              SHL     BL,1
1AC8  80 E3 10                 4862              AND     BL,010H
1ACB  80 E7 07                 4863              AND     BH,07H
1ACE  0A DF                    4864              OR      BL,BH
                               4865
1AD0  A0 0449 R                4866              MOV     AL,CRT_MODE
1AD3  3C 03                    4867              CMP     AL,3
1AD5  76 0E                    4868              JBE     M21
                               4869
                               4870        ;----- GRAPHICS MODE DONE HERE (SET PALETTE 0 AND OVERSCAN)
                               4871
1AD7  B4 00                    4872              MOV     AH,0
1AD9  8A C3                    4873              MOV     AL,BL
1ADB  E8 1D9F R                4874              CALL    PAL_SET
                               4875
1ADE  0B ED                    4876              OR      BP,BP
1AE0  74 03                    4877              JZ      M21
1AE2  26: 88 1D                4878              MOV     ES:[DI],BL
                               4879
                               4880        ;----- ALPHA MODE DONE HERE (SET OVERSCAN REGISTER)
                               4881
1AE5                           4882        M21:
1AE5  80 3E 0449 R 03          4883              CMP     CRT_MODE,3                      ; CHECK FOR AN ENHANCED MODE
1AEA  77 05                    4884              JA      SET_OVRSC                       ; NO CHANCE
1AEC  E8 0E9A R                4885              CALL    BRST_DET                        ; SEE IF WE ARE ENHANCED
1AEF  72 07                    4886              JC      SKIP_OVRSC                      ; THERE IS NO BORDER
1AF1                           4887        SET_OVRSC:
1AF1  B4 11                    4888              MOV     AH,011H                         ; OVERSCAN REGISTER
1AF3  8A C3                    4889              MOV     AL,BL
1AF5  E8 1D9F R                4890              CALL    PAL_SET                         ; SET THE BORDER
1AF8                           4891        SKIP_OVRSC:
1AF8  0B ED                    4892              OR      BP,BP
1AFA  74 04                    4893              JZ      M21Y
1AFC  26: 88 5D 10             4894              MOV     ES:[DI][16D],BL
1B00                           4895        M21Y:
1B00  8A DD                    4896              MOV     BL,CH
1B02  80 E3 20                 4897              AND     BL,020H
1B05  B1 05                    4898              MOV     CL,5
1B07  D2 EB                    4899              SHR     BL,CL
                               4900
                               4901        ;----- HANDLE BH = 1 HERE
                               4902        ;       ALPHA MODES  => NO EFFECT
                               4903        ;       GRAPHICS     => LOW BIT OF BL = 0
                               4904        ;                       PALETTE 0 = BACKGROUND
                               4905        ;                       PALETTE 1 = GREEN
                               4906        ;                       PALETTE 2 = RED
                               4907        ;                       PALETTE 3 = BROWN
                               4908        ;                    => LOW BIT OF BL = 1
                               4909        ;                       PALETTE 0 = BACKGROUND
                               4910        ;                       PALETTE 1 = CYAN
                               4911        ;                       PALETTE 2 = MAGENTA
                               4912        ;                       PALETTE 3 = WHITE
                               4913        ;
                               4914
```

```
1B09                                    4915            M20:
1B09  80 3E 0449 R 03                   4916                    CMP     CRT_MODE,3
1B0E  76 4A                             4917                    JBE     M80
                                        4918
1B10  A0 0466 R                         4919                    MOV     AL,CRT_PALETTE
1B13  24 DF                             4920                    AND     AL,0DFH
1B15  80 E3 01                          4921                    AND     BL,1
1B18  74 02                             4922                    JZ      M22
1B1A  0C 20                             4923                    OR      AL,020H
1B1C                                    4924            M22:
1B1C  A2 0466 R                         4925                    MOV     CRT_PALETTE,AL
1B1F  24 10                             4926                    AND     AL,010H
1B21  0C 02                             4927                    OR      AL,2
1B23  0A D8                             4928                    OR      BL,AL
1B25  B4 01                             4929                    MOV     AH,1
1B27  8A C3                             4930                    MOV     AL,BL
1B29  E8 1D9F R                         4931                    CALL    PAL_SET
                                        4932
1B2C  0B ED                             4933                    OR      BP,BP
1B2E  74 04                             4934                    JZ      M22Y
1B30  26: 88 5D 01                      4935                    MOV     ES:[DI][1],BL
1B34                                    4936            M22Y:
                                        4937
1B34  FE C3                             4938                    INC     BL
1B36  FE C3                             4939                    INC     BL
1B38  B4 02                             4940                    MOV     AH,2
1B3A  8A C3                             4941                    MOV     AL,BL
1B3C  E8 1D9F R                         4942                    CALL    PAL_SET
                                        4943
1B3F  0B ED                             4944                    OR      BP,BP
1B41  74 04                             4945                    JZ      M27Y
1B43  26: 88 5D 02                      4946                    MOV     ES:[DI][2],BL
1B47                                    4947            M27Y:
                                        4948
1B47  FE C3                             4949                    INC     BL
1B49  FE C3                             4950                    INC     BL
1B4B  B4 03                             4951                    MOV     AH,3
1B4D  8A C3                             4952                    MOV     AL,BL
1B4F  E8 1D9F R                         4953                    CALL    PAL_SET
                                        4954
1B52  0B ED                             4955                    OR      BP,BP
1B54  74 04                             4956                    JZ      M80
1B56  26: 88 5D 03                      4957                    MOV     ES:[DI][3],BL
                                        4958
1B5A                                    4959            M80:
1B5A  E8 1DB7 R                         4960                    CALL    PAL_ON
1B5D  E9 219E R                         4961                    JMP     V_RET
                                        4962
                                        4963    C               INCLUDE         VDOT.INC
                                        4964    C               SUBTTL  VDOT.INC
                                        4965    C               PAGE
                                        4966    C       ;----------------------------------------------
                                        4967    C       ; ENTRY                                        :
                                        4968    C       ;       DX = ROW                               :
                                        4969    C       ;       CX = COLUMN                            :
                                        4970    C       ;       BH = PAGE                              :
                                        4971    C       ; EXIT                                         :
                                        4972    C       ;       BX = OFFSET INTO REGEN                 :
                                        4973    C       ;       AL = BIT MASK FOR COLUMN BYTE          :
                                        4974    C       ;----------------------------------------------
1B60                                    4975    C       DOT_SUP_1       PROC    NEAR
                                        4976    C
                                        4977    C       ;----- OFFSET = PAGE OFFSET + ROW * BYTES/ROW + COLUMN/8
                                        4978    C
1B60  F7 26 044A R                      4979    C               MUL     WORD PTR CRT_COLS               ; ROW * BYTES/ROW
1B64  51                                4980    C               PUSH    CX                              ; SAVE COLUMN VALUE
1B65  D1 E9                             4981    C               SHR     CX,1                            ; DIVIDE BY EIGHT TO
1B67  D1 E9                             4982    C               SHR     CX,1                            ;  DETERMINE THE BYTE THAT
1B69  D1 E9                             4983    C               SHR     CX,1                            ;  THIS DOT IS IN
                                        4984    C                                                       ;  (8 BITS/BYTE)
1B6B  03 C1                             4985    C               ADD     AX,CX                           ; BYTE OFFSET INTO PAGE
1B6D  8A DF                             4986    C               MOV     BL,BH                           ; GET PAGE INTO BL
1B6F  2A FF                             4987    C               SUB     BH,BH                           ; ZERO
1B71  8B CB                             4988    C               MOV     CX,BX                           ; COUNT VALUE
1B73  8B 1E 044C R                      4989    C               MOV     BX,CRT_LEN                      ; LENGTH OF ONE PAGE
1B77  E3 04                             4990    C               JCXZ    DS_2                            ; PAGE ZERO
1B79                                    4991    C       DS_3:
1B79  03 C3                             4992    C               ADD     AX,BX                           ; BUMP TO NEXT PAGE
1B7B  E2 FC                             4993    C               LOOP    DS_3                            ; DO FOR THE REST
1B7D                                    4994    C       DS_2:
1B7D  59                                4995    C               POP     CX                              ; RECOVER COLUMN VALUE
1B7E  8B D8                             4996    C               MOV     BX,AX                           ; REGEN OFFSET
1B80  80 E1 07                          4997    C               AND     CL,07H                          ; SHIFT COUNT FOR BIT MASK
1B83  B0 80                             4998    C               MOV     AL,080H                         ; MASK BIT
1B85  D2 E8                             4999    C               SHR     AL,CL                           ; POSITION MASK BIT
1B87  C3                                5000    C               RET
1B88                                    5001    C       DOT_SUP_1       ENDP
                                        5002    C
                                        5003    C       ;--------------------------------------------------------------
                                        5004    C       ; THIS SUBROUTINE DETERMINES THE REGEN BYTE LOCATION    :
                                        5005    C       ; OF THE INDICATED ROW COLUMN VALUE IN GRAPHICS MODE.   :
                                        5006    C       ; ENTRY --                                              :
                                        5007    C       ;  DX = ROW VALUE (0-199)                               :
                                        5008    C       ;  CX = COLUMN VALUE (0-639)                            :
                                        5009    C       ; EXIT --                                               :
                                        5010    C       ;  SI = OFFSET INTO REGEN BUFFER FOR BYTE OF INTEREST   :
                                        5011    C       ;  AH = MASK TO STRIP OFF THE BITS OF INTEREST          :
                                        5012    C       ;  CL = BITS TO SHIFT TO RIGHT JUSTIFY THE MASK IN AH   :
                                        5013    C       ;  DH = # BITS IN RESULT                                :
                                        5014    C       ;--------------------------------------------------------------
1B88                                    5015    C       R3      PROC    NEAR
1B88  53                                5016    C               PUSH    BX                              ; SAVE BX DURING OPERATION
1B89  50                                5017    C               PUSH    AX                              ; WILL SAVE AL DURING OPERATION
                                        5018    C
                                        5019    C       ;----- DETERMINE 1ST BYTE IN IDICATED ROW BY MULTIPLYING ROW VALUE BY 40
                                        5020    C       ;-----  ( LOW BIT OF ROW DETERMINES EVEN/ODD, 80 BYTES/ROW
                                        5021    C
1B8A  B0 28                             5022    C               MOV     AL,40
1B8C  52                                5023    C               PUSH    DX                              ; SAVE ROW VALUE
1B8D  80 E2 FE                          5024    C               AND     DL,0FEH                         ; STRIP OFF ODD/EVEN BIT
1B90  F6 E2                             5025    C               MUL     DL                              ; AX HAS ADDRESS OF 1ST BYTE
                                        5026    C                                                       ;  OF INDICATED ROW
1B92  5A                                5027    C               POP     DX                              ; RECOVER IT
1B93  F6 C2 01                          5028    C               TEST    DL,1                            ; TEST FOR EVEN/ODD
1B96  74 03                             5029    C               JZ      R4                              ; JUMP IF EVEN ROW
1B98  05 2000                           5030    C               ADD     AX,2000H                        ; OFFSET TO LOCATION OF ODD ROWS
1B9B                                    5031    C       R4:                                             ; EVEN_ROW
1B9B  8B F0                             5032    C               MOV     SI,AX                           ; MOVE POINTER TO SI
1B9D  58                                5033    C               POP     AX                              ; RECOVER AL VALUE
1B9E  8B D1                             5034    C               MOV     DX,CX                           ; COLUMN VALUE TO DX
                                        5035    C
                                        5036    C       ;----- DETERMINE GRAPHICS MODE CURRENTLY IN EFFECT
                                        5037    C
                                        5038    C       ;--------------------------------------------------------------
                                        5039    C       ; SET UP THE REGISTERS ACCORDING TO THE MODE                    :
                                        5040    C       ;  CH = MASK FOR LOW OF COLUMN ADDRESS ( 7/3 FOR HIGH/MED RES)  :
```

```
                        5041    C  ;  CL = # OF ADDRESS BITS IN COLUMN VALUE ( 3/2 FOR H/M)          :
                        5042    C  ;  BL = MASK TO SELECT BITS FROM POINTED BYTE (80H/C0H FOR H/M)  :
                        5043    C  ;  BH = NUMBER OF VALID BITS IN POINTED BYTE ( 1/2 FOR H/M)      :
                        5044    C  '-------------------------------------------------------------------
                        5045    C
1BA0  BB 02C0           5046    C          MOV     BX,2C0H
1BA3  B9 0302           5047    C          MOV     CX,302H                 ; SET PARMS FOR MED RES
1BA6  80 3E 0449 R 06   5048    C          CMP     CRT_MODE,6
1BAB  72 06             5049    C          JC      R5                      ; HANDLE IF MED ARES
1BAD  BB 0180           5050    C          MOV     BX,180H
1BB0  B9 0703           5051    C          MOV     CX,703H                 ; SET PARMS FOR HIGH RES
                        5052    C
                        5053    C  ;----- DETERMINE BIT OFFSET IN BYTE FROM COLUMN MASK
                        5054    C
1BB3                    5055    C  R5:
1BB3  22 EA             5056    C          AND     CH,DL                   ; ADDRESS OF PEL WITHIN BYTE TO CH
                        5057    C
                        5058    C  ;----- DETERMINE BYTE OFFSET FOR THIS LOCATION IN COLUMN
                        5059    C
1BB5  D3 EA             5060    C          SHR     DX,CL                   ; SHIFT BY CORRECT AMOUNT
1BB7  03 F2             5061    C          ADD     SI,DX                   ; INCREMENT THE POINTER
1BB9  8A F7             5062    C          MOV     DH,BH                   ; GET THE # OF BITS IN RESULT TO DH
                        5063    C
                        5064    C  ;----- MULTIPLY BH (VALID BITS IN BYTE) BY CH (BIT OFFSET)
                        5065    C
1BBB  2A C9             5066    C          SUB     CL,CL                   ; ZERO INTO STORAGE LOCATION
1BBD                    5067    C  R6:
1BBD  D0 C8             5068    C          ROR     AL,1                    ; LEFT JUSTIFY THE VALUE
                        5069    C                                          ;  IN AL (FOR WRITE)
1BBF  02 CD             5070    C          ADD     CL,CH                   ; ADD IN THE BIT OFFSET VALUE
1BC1  FE CF             5071    C          DEC     BH                      ; LOOP CONTROL
1BC3  75 F8             5072    C          JNZ     R6                      ; ON EXIT, CL HAS SHIFT COUNT
                        5073    C                                          ;  TO RESTORE BITS
1BC5  8A E3             5074    C          MOV     AH,BL                   ; GET MASK TO AH
1BC7  D2 EC             5075    C          SHR     AH,CL                   ; MOVE THE MASK TO CORRECT LOCATION
1BC9  5B                5076    C          POP     BX                      ; RECOVER REG
1BCA  C3                5077    C          RET                             ; RETURN WITH EVERYTHING SET UP
1BCB                    5078    C  R3      ENDP
                        5079    C
                        5080    C  ;-------------------------------------------------------------
                        5081    C  ; READ DOT  -- WRITE DOT                                      :
                        5082    C  ;       THESE ROUTINES WILL WRITE A DOT, OR READ THE DOT AT   :
                        5083    C  ;       THE INDICATED LOCATION                               :
                        5084    C  ; ENTRY --                                                    :
                        5085    C  ;   DX = ROW (0-199)    (THE ACTUAL VALUE DEPENDS ON THE MODE)  :
                        5086    C  ;   CX = COLUMN ( 0-639) ( THE VALUES ARE NOT RANGE CHECKED )  :
                        5087    C  ;   AL = DOT VALUE TO WRITE (1,2 OR 4 BITS DEPENDING ON MODE,  :
                        5088    C  ;        REQ'D FOR WRITE DOT ONLY, RIGHT JUSTIFIED)           :
                        5089    C  ;        BIT 7 OF AL=1 INDICATES XOR THE VALUE INTO THE LOCATION :
                        5090    C  ;   DS = DATA SEGMENT                                        :
                        5091    C  ;   ES = REGEN SEGMENT                                       :
                        5092    C  ;                                                            :
                        5093    C  ; EXIT                                                       :
                        5094    C  ;       AL = DOT VALUE READ, RIGHT JUSTIFIED, READ ONLY      :
                        5095    C  ;-------------------------------------------------------------
                        5096    C
                        5097    C  ;----- WRITE DOT
                        5098    C
1BCB                    5099    C  AHC:
                        5100    C          ASSUME  DS:ABS0
1BCB  80 3E 0449 R 07   5101    C          CMP     CRT_MODE,7
1BD0  77 2A             5102    C          JA      WRITE_DOT_2
                        5103    C
1BD2                    5104    C  WRITE_DOT       PROC    NEAR
                        5105    C          ASSUME  DS:ABS0,ES:NOTHING
1BD2  52                5106    C          PUSH    DX
                        5107    C          SRLOAD  ES,0B800H
1BD3  BA B800           5108    C+         MOV     DX,0B800H
1BD6  8E C2             5109    C+         MOV     ES,DX
1BD8  5A                5110    C          POP     DX
1BD9  50                5111    C          PUSH    AX                      ; SAVE DOT VALUE
1BDA  50                5112    C          PUSH    AX                      ;  TWICE
1BDB  E8 1B88 R         5113    C          CALL    R3                      ; DETERMINE BYTE POSITION OF THE DOT
1BDE  D2 E8             5114    C          SHR     AL,CL                   ; SHIFT TO SET UP THE BITS FOR OUTPUT
1BE0  22 C4             5115    C          AND     AL,AH                   ; STRIP OFF THE OTHER BITS
1BE2  26: 8A 0C         5116    C          MOV     CL,ES:[SI]              ; GET THE CURRENT BYTE
1BE5  5B                5117    C          POP     BX                      ; RECOVER XOR FLAG
1BE6  F6 C3 80          5118    C          TEST    BL,80H                  ; IS IT ON
1BE9  75 0D             5119    C          JNZ     R2                      ; YES,  XOR THE DOT
1BEB  F6 D4             5120    C          NOT     AH                      ; SET THE MASK TO REMOVE THE
1BED  22 CC             5121    C          AND     CL,AH                   ;  INDICATED BITS
1BEF  0A C1             5122    C          OR      AL,CL                   ; OR IN THE NEW VALUE OF THOSE BITS
1BF1                    5123    C  R1:                                     ; FINISH_DOT
1BF1  26: 88 04         5124    C          MOV     ES:[SI],AL              ; RESTORE THE BYTE IN MEMORY
1BF4  58                5125    C          POP     AX
1BF5  E9 219E R         5126    C          JMP     V_RET
1BF8                    5127    C  R2:                                     ; XOR_DOT
1BF8  32 C1             5128    C          XOR     AL,CL                   ; EXCLUSIVE OR THE DOTS
1BFA  EB F5             5129    C          JMP     R1                      ; FINISH UP THE WRITING
1BFC                    5130    C  WRITE_DOT       ENDP
                        5131    C
1BFC                    5132    C  WRITE_DOT_2     PROC    NEAR
1BFC  80 3E 0449 R 0F   5133    C          CMP     CRT_MODE,0FH
1C01  72 0D             5134    C          JB      NO_ADJ2
1C03  E8 14F7 R         5135    C          CALL    MEM_DET                         ; BASE CARD
1C06  72 08             5136    C          JC      NO_ADJ2
1C08  24 85             5137    C          AND     AL,10000101B            ; 85H, XOR C2 C0 MASK
1C0A  8A E0             5138    C          MOV     AH,AL
1C0C  D0 E4             5139    C          SHL     AH,1                    ; EXPAND C0 TO C1, C2 TO C3
1C0E  0A C4             5140    C          OR      AL,AH                   ; BUILD ?(80H) + (0,3,C,F)
1C10                    5141    C  NO_ADJ2:
1C10  50                5142    C          PUSH    AX
1C11  8B C2             5143    C          MOV     AX,DX                   ; ROW VALUE
1C13  E8 1B60 R         5144    C          CALL    DOT_SUP_1               ; BX=OFFSET, AL=BIT MASK
1C16  B6 03             5145    C          MOV     DH,3
1C18  B2 CE             5146    C          MOV     DL,GRAPH_ADDR           ; GRAPHICS CHIP
1C1A  B4 08             5147    C          MOV     AH,G_BIT_MASK           ; BIT MASK REGISTER
1C1C  E8 0D15 R         5148    C          CALL    OUT_DX                  ; SET BIT MASK
1C1F  52                5149    C          PUSH    DX
                        5150    C          SRLOAD  ES,0A000H               ; REGEN SEGMENT
1C20  BA A000           5151    C+         MOV     DX,0A000H
1C23  8E C2             5152    C+         MOV     ES,DX
1C25  5A                5153    C          POP     DX
1C26  58                5154    C          POP     AX                      ; RECOVER COLOR
1C27  8A E8             5155    C          MOV     CH,AL                   ; SAVE COLOR
1C29  F6 C5 80          5156    C          TEST    CH,080H                 ; SEE IF XOR
1C2C  74 0A             5157    C          JZ      WD_A                    ; NO XOR
1C2E  B4 03             5158    C          MOV     AH,G_DATA_ROT           ; DO XOR
1C30  B0 18             5159    C          MOV     AL,018H                 ; XOR FUNCTION
1C32  E8 0D15 R         5160    C          CALL    OUT_DX                  ; SET THE REGISTER
1C35  EB 12 90          5161    C          JMP     WD_B                    ; SKIP THE BLANK
1C38                    5162    C  WD_A:                                   ; BLANK THE DOT
1C38  B2 C4             5163    C          MOV     DL,SEQ_ADDR             ; SEQUENCER
1C3A  B4 02             5164    C          MOV     AH,S_MAP                ; MAP MASK
1C3C  B0 FF             5165    C          MOV     AL,0FFH                 ; ENABLE ALL MAPS
1C3E  E8 0D15 R         5166    C          CALL    OUT_DX                  ; SET THE REGISTER
```

```
1C41  26: 8A 07                5167     C          MOV     AL,ES:[BX]                      ; LATCH DATA
1C44  2A C0                    5168     C          SUB     AL,AL                           ; ZERO
1C46  26: 88 07                5169     C          MOV     ES:[BX],AL                      ; BLANK THE DOT
1C49                           5170     C  WD_B:                                           ; SET THE COLOR MAP MASK
1C49  B2 C4                    5171     C          MOV     DL,SEQ_ADDR                     ; SEQUENCER
1C4B  B4 02                    5172     C          MOV     AH,S_MAP                        ; MAP MASK REGISTER
1C4D  8A C5                    5173     C          MOV     AL,CH                           ; COLOR VALUE
1C4F  24 0F                    5174     C          AND     AL,0FH                          ; VALUES 0-15
1C51  E8 0D15 R                5175     C          CALL    OUT_DX                          ; SET IT
1C54  26: 8A 07                5176     C          MOV     AL,ES:[BX]                      ; LATCH DATA
1C57  B0 FF                    5177     C          MOV     AL,0FFH                         ; WRITE VALUE
1C59  26: 88 07                5178     C          MOV     ES:[BX],AL                      ; SET THE DOT
                               5179     C
                               5180     C  ;----- NORMALIZE THE ENVIRONMENT
                               5181     C
1C5C  E8 0D15 R                5182     C          CALL    OUT_DX                          ; ALL MAPS ON
1C5F  B2 CE                    5183     C          MOV     DL,GRAPH_ADDR                   ; GRAPHICS CHIPS
1C61  B4 03                    5184     C          MOV     AH,G_DATA_ROT                   ; XOR REGISTER
1C63  2A C0                    5185     C          SUB     AL,AL                           ; NORMAL WRITES
1C65  E8 0D15 R                5186     C          CALL    OUT_DX                          ; SET IT
1C68  B4 08                    5187     C          MOV     AH,G_BIT_MASK                   ; BIT MASK
1C6A  B0 FF                    5188     C          MOV     AL,0FFH                         ; ALL BITS ON
1C6C  E8 0D15 R                5189     C          CALL    OUT_DX                          ; SET IT
1C6F  E9 219E R                5190     C          JMP     V_RET                           ; WRITE DOT DONE
1C72                           5191     C  WRITE_DOT_2     ENDP
                               5192     C
1C72                           5193     C  RD_S    PROC    NEAR
                               5194     C          ASSUME  DS:ABS0
1C72  50                       5195     C          PUSH    AX
1C73  52                       5196     C          PUSH    DX
                               5197     C          SRLOAD  ES,0A000H
1C74  BA A000                  5198     C+         MOV     DX,0A000H
1C77  8E C2                    5199     C+         MOV     ES,DX
1C79  5A                       5200     C          POP     DX
1C7A  58                       5201     C          POP     AX
1C7B  8B C2                    5202     C          MOV     AX,DX
1C7D  E8 1B60 R                5203     C          CALL    DOT_SUP_1
1C80  B5 07                    5204     C          MOV     CH,7
1C82  2A E9                    5205     C          SUB     CH,CL
1C84  2B D2                    5206     C          SUB     DX,DX
1C86  B0 00                    5207     C          MOV     AL,0
1C88  C3                       5208     C          RET
1C89                           5209     C  RD_S    ENDP
                               5210     C
1C89                           5211     C  RD_1S   PROC    NEAR
1C89  8A CD                    5212     C          MOV     CL,CH
1C8B  B4 04                    5213     C          MOV     AH,4
1C8D  52                       5214     C          PUSH    DX
1C8E  B6 03                    5215     C          MOV     DH,3
1C90  B2 CE                    5216     C          MOV     DL,GRAPH_ADDR
1C92  E8 0D15 R                5217     C          CALL    OUT_DX
1C95  5A                       5218     C          POP     DX
1C96  26: 8A 27                5219     C          MOV     AH,ES:[BX]
1C99  D2 EC                    5220     C          SHR     AH,CL
1C9B  80 E4 01                 5221     C          AND     AH,1
1C9E  C3                       5222     C          RET
1C9F                           5223     C  RD_1S   ENDP
                               5224     C
                               5225     C  ;----- READ DOT
                               5226     C
1C9F                           5227     C  AHD:
                               5228     C          ASSUME  DS:ABS0
1C9F  80 3E 0449 R 07          5229     C          CMP     CRT_MODE,7
1CA4  77 18                    5230     C          JA      R_1
                               5231     C
1CA6                           5232     C  READ_DOT        PROC    NEAR
                               5233     C          ASSUME  DS:ABS0,ES:NOTHING
1CA6  52                       5234     C          PUSH    DX
                               5235     C          SRLOAD  ES,0B800H
1CA7  BA B800                  5236     C+         MOV     DX,0B800H
1CAA  8E C2                    5237     C+         MOV     ES,DX
1CAC  5A                       5238     C          POP     DX
1CAD  E8 1B88 R                5239     C          CALL    R3                              ; DETERMINE BYTE POSITION OF DOT
1CB0  26: 8A 04                5240     C          MOV     AL,ES:[SI]                      ; GET THE BYTE
1CB3  22 C4                    5241     C          AND     AL,AH                           ; MASK OFF THE OTHER BITS IN THE BYTE
1CB5  D2 E0                    5242     C          SHL     AL,CL                           ; LEFT JUSTIFY THE VALUE
1CB7  8A CE                    5243     C          MOV     CL,DH                           ; GET NUMBER OF BITS IN RESULT
1CB9  D2 C0                    5244     C          ROL     AL,CL                           ; RIGHT JUSTIFY THE RESULT
1CBB  E9 219E R                5245     C          JMP     V_RET
1CBE                           5246     C  READ_DOT        ENDP
                               5247     C
1CBE                           5248     C  R_1:
1CBE  80 3E 0449 R 0F          5249     C          CMP     CRT_MODE,0FH
1CC3  72 25                    5250     C          JB      READ_DOT_2
1CC5  E8 14F7 R                5251     C          CALL    MEM_DET
1CC8  72 20                    5252     C          JC      READ_DOT_2
                               5253     C
1CCA                           5254     C  READ_DOT_1      PROC    NEAR                    ; 2 MAPS
                               5255     C          ASSUME  DS:ABS0, ES:NOTHING
1CCA  E8 1C72 R                5256     C          CALL    RD_S
1CCD  E8 1C89 R                5257     C          CALL    RD_1S
1CD0  0A D4                    5258     C          OR      DL,AH
1CD2  D0 E4                    5259     C          SHL     AH,1
1CD4  0A D4                    5260     C          OR      DL,AH
1CD6  B0 02                    5261     C          MOV     AL,2
1CD8  E8 1C89 R                5262     C          CALL    RD_1S
1CDB  D0 E4                    5263     C          SHL     AH,1
1CDD  D0 E4                    5264     C          SHL     AH,1
1CDF  0A D4                    5265     C          OR      DL,AH
1CE1  D0 E4                    5266     C          SHL     AH,1
1CE3  0A D4                    5267     C          OR      DL,AH
1CE5  8A C2                    5268     C          MOV     AL,DL
1CE7  E9 219E R                5269     C          JMP     V_RET
1CEA                           5270     C  READ_DOT_1      ENDP
                               5271     C
1CEA                           5272     C  READ_DOT_2      PROC    NEAR                    ; 4 MAPS
                               5273     C          ASSUME  DS:ABS0, ES:NOTHING
1CEA  E8 1C72 R                5274     C          CALL    RD_S
1CED                           5275     C  RD_2A:
1CED  E8 1C89 R                5276     C          CALL    RD_1S
1CF0  8A C8                    5277     C          MOV     CL,AL
1CF2  D2 E4                    5278     C          SHL     AH,CL
1CF4  0A D4                    5279     C          OR      DL,AH
1CF6  FE C0                    5280     C          INC     AL
1CF8  3C 03                    5281     C          CMP     AL,3
1CFA  76 F1                    5282     C          JBE     RD_2A
1CFC  8A C2                    5283     C          MOV     AL,DL
1CFE  E9 219E R                5284     C          JMP     V_RET
1D01                           5285     C  READ_DOT_2      ENDP
                               5286     C
                               5287     C  ;--------------------------------------------------------------------------
                               5288     C  ; WRITE_TTY     WRITE TELETYPE TO ACTIVE PAGE                               :
                               5289     C  ;       THIS INTERFACE PROVIDES A TELETYPE LIKE INTERFACE TO THE VIDEO  :
                               5290     C  ;       CARD.   THE INPUT CHARACTER  IS WRITTEN TO THE  CURRENT CURSOR  :
                               5291     C  ;       POSITION, AND THE CURSOR IS MOVED TO THE NEXT POSITION. IF THE  :
                               5292     C  ;       CURSOR LEAVES THE LAST COLUMN OF THE FIELD,  THE COLUMN IS SET  :
```

```
                        ;       TO ZERO,  AND THE ROW VALUE IS INCREMENTED.   IF THE ROW VALUE  :
                        ;       LEAVES THE FIELD,  THE CURSOR IS PLACED ON THE LAST ROW, FIRST  :
                        ;       COLUMN,  AND THE ENTIRE SCREEN IS SCROLLED UP ONE LINE.   WHEN  :
                        ;       THE SCREEN IS SCROLLED UP, THE ATTRIBUTE FOR FILLING THE NEWLY  :
                        ;       BLANKED LINE IS READ FROM THE  CURSOR POSITION ON THE PREVIOUS  :
                        ;       LINE BEFORE THE SCROLL,  IN CHARACTER MODE.  IN GRAPHICS MODE,  :
                        ;       THE 0 COLOR IS USED.                                           :
                        ; ENTRY                                                                :
                        ;       (AH) = CURRENT CRT MODE                                        :
                        ;       (AL) = CHARACTER TO BE WRITTEN                                 :
                        ;        NOTE THAT BACK SPACE, CAR RET, BELL AND LINE FEED ARE HANDLED :
                        ;        AS COMMANDS RATHER THAN AS DISPLAYABLE GRAPHICS               :
                        ;       (BL) = FOREGROUND COLOR FOR CHAR WRITE IF CURRENTLY IN A        :
                        ;        GRAPHICS MODE                                                 :
                        ; EXIT                                                                 :
                        ;       ALL REGISTERS SAVED                                            :
                        ;-------------------------------------------------------------------------
1D01                    AHE:
                                ASSUME  CS:CODE,DS:ABS0
1D01  50                        PUSH    AX                              ; SAVE REGISTERS
1D02  8A 3E 0462 R              MOV     BH,ACTIVE_PAGE                  ; GET THE ACTIVE PAGE
1D06  53                        PUSH    BX                              ; SAVE
1D07  8A DF                     MOV     BL,BH                           ; GET PAGE TO BL
1D09  32 FF                     XOR     BH,BH                           ; CLEAR HIGH BYTE
1D0B  D1 E3                     SAL     BX,1                            ; *2 FOR WORD OFFSET
1D0D  8B 97 0450 R              MOV     DX,[BX + OFFSET CURSOR_POSN]    ; CURSOR, ACTIVE PAGE
1D11  5B                        POP     BX                              ; RECOVER
                        ;
                        ;-----  DX NOW HAS THE CURRENT CURSOR POSITION
                        ;
1D12  3C 0D                     CMP     AL,0DH                          ; IS IT CARRIAGE RETURN
1D14  74 5C                     JE      U9                              ; CAR_RET
1D16  3C 0A                     CMP     AL,0AH                          ; IS TT A LINE FEED
1D18  74 5C                     JE      U10                             ; LINE_FEED
1D1A  3C 08                     CMP     AL,08H                          ; IS IT A BACKSPACE
1D1C  74 4C                     JE      U8                              ; BACK_SPACE
1D1E  3C 07                     CMP     AL,07H                          ; IS IT A BELL
1D20  74 5C                     JE      U11                             ; BELL
                        ;
                        ;-----  WRITE THE CHAR TO THE SCREEN
                        ;
1D22  B4 0A                     MOV     AH,10                           ; WRITE CHAR ONLY
1D24  B9 0001                   MOV     CX,1                            ; ONLY ONE CHAR
1D27  CD 10                     INT     10H                             ; WRITE THE CHAR
                        ;
                        ;-----  POSITION THE CURSOR FOR NEXT CHAR
                        ;
1D29  FE C2                     INC     DL
1D2B  3A 16 044A R              CMP     DL,BYTE PTR CRT_COLS            ; TEST FOR COLUMN OVERFLOW
1D2F  75 35                     JNZ     U7                              ; SET_CURSOR
1D31  2A D2                     SUB     DL,DL                           ; COLUMN FOR CURSOR
1D33  3A 36 0484 R              CMP     DH,ROWS
1D37  75 2B                     JNZ     U6                              ; SET_CURSOR_INC
                        ;
                        ;-----  SCROLL REQUIRED
                        ;
1D39                    U1:
1D39  E8 115D R                 CALL    SET_CPOS                        ; SET THE CURSOR
                        ;
                        ;-----  DETERMINE VALUE TO FILL WITH DURING SCROLL
                        ;
1D3C  A0 0449 R                 MOV     AL,CRT_MODE                     ; GET THE CURRENT MODE
1D3F  3C 04                     CMP     AL,4
1D41  72 06                     JB      U2                              ; READ-CURSOR
1D43  2A FF                     SUB     BH,BH                           ; FILL WITH BACKGROUND
1D45  3C 07                     CMP     AL,7
1D47  75 06                     JNE     U3                              ; SCROLL-UP
1D49                    U2:                                             ; READ-CURSOR
1D49  B4 08                     MOV     AH,8
1D4B  CD 10                     INT     10H                             ; READ CHAR/ATTR
1D4D  8A FC                     MOV     BH,AH                           ; STORE IN BH
1D4F                    U3:                                             ; SCROLL-UP
1D4F  B8 0601                   MOV     AX,601H                         ; SCROLL ONE LINE
1D52  2B C9                     SUB     CX,CX                           ; UPPER LEFT CORNER
1D54  8A 36 0484 R              MOV     DH,ROWS                         ; LOWER RIGHT ROW
1D58  8A 16 044A R              MOV     DL,BYTE PTR CRT_COLS            ; LOWER RIGHT COLUMN
1D5C  FE CA                     DEC     DL
1D5E                    U4:                                             ; VIDEO-CALL-RETURN
1D5E  CD 10                     INT     10H                             ; SCROLL UP THE SCREEN
1D60                    U5:                                             ; TTY-RETURN
1D60  58                        POP     AX                              ; RESTORE THE CHARACTER
1D61  E9 219E R                 JMP     V_RET                           ; RETURN TO CALLER
1D64                    U6:                                             ; SET-CURSOR-INC
1D64  FE C6                     INC     DH                              ; NEXT ROW
1D66                    U7:                                             ; SET-CURSOR
1D66  B4 02                     MOV     AH,2
1D68  EB F4                     JMP     U4                              ; ESTABLISH THE NEW CURSOR
                        ;
                        ;-----  BACK SPACE FOUND
                        ;
1D6A                    U8:
1D6A  0A D2                     OR      DL,DL                           ; ALREADY AT END OF LINE
1D6C  74 F8                     JZ      U7                              ; SET_CURSOR
1D6E  FE CA                     DEC     DL                              ; NO -- JUST MOVE IT BACK
1D70  EB F4                     JMP     U7                              ; SET_CURSOR
                        ;
                        ;-----  CARRIAGE RETURN FOUND
                        ;
1D72                    U9:
1D72  2A D2                     SUB     DL,DL                           ; MOVE TO FIRST COLUMN
1D74  EB F0                     JMP     U7                              ; SET_CURSOR
                        ;
                        ;-----  LINE FEED FOUND
                        ;
1D76                    U10:
1D76  3A 36 0484 R              CMP     DH,ROWS                         ; BOTTOM OF SCREEN
1D7A  75 E8                     JNE     U6                              ; YES, SCROLL THE SCREEN
1D7C  EB BB                     JMP     U1                              ; NO, JUST SET THE CURSOR
                        ;
                        ;-----  BELL FOUND
                        ;
1D7E                    U11:
1D7E  B3 02                     MOV     BL,2                            ; SET UP COUNT FOR BEEP
1D80  E8 0D20 R                 CALL    BEEP                            ; SOUND THE POD BELL
1D83  EB DB                     JMP     U5                              ; TTY_RETURN
                        ;
                        ;
                        ;-----  CURRENT VIDEO STATE
                        ;
1D85                    AHF:
                                ASSUME  DS:ABS0
1D85  8A 26 044A R              MOV     AH,BYTE PTR CRT_COLS            ; GET NUMBER OF COLUMNS
1D89  8A 3E 0462 R              MOV     BH,ACTIVE_PAGE
1D8D  A0 0487 R                 MOV     AL,INFO
1D90  24 80                     AND     AL,080H
1D92  0A 06 0449 R              OR      AL,CRT_MODE
```

```
1D96  5F                    5419  C           POP     DI
1D97  5E                    5420  C           POP     SI
1D98  59                    5421  C           POP     CX                      ; DISCARD BX
1D99  59                    5422  C           POP     CX
1D9A  5A                    5423  C           POP     DX
1D9B  1F                    5424  C           POP     DS
1D9C  07                    5425  C           POP     ES
1D9D  5D                    5426  C           POP     BP
1D9E  CF                    5427  C           IRET
                            5428  C
                            5429  C           SUBTTL
                            5430
                            5431
1D9F                        5432        PAL_SET PROC    NEAR
1D9F  50                    5433                PUSH    AX
1DA0  E8 0D05 R             5434                CALL    WHAT_BASE
1DA3  FA                    5435                CLI
1DA4                        5436        VR:
1DA4  EC                    5437                IN      AL,DX
1DA5  A8 08                 5438                TEST    AL,08H                  ; VERTICAL RETRACE
1DA7  74 FB                 5439                JZ      VR
1DA9  58                    5440                POP     AX
1DAA  B2 C0                 5441                MOV     DL,ATTR_WRITE
1DAC  86 C4                 5442                XCHG    AL,AH
1DAE  EE                    5443                OUT     DX,AL
1DAF  86 C4                 5444                XCHG    AL,AH
1DB1  EE                    5445                OUT     DX,AL
1DB2  B0 20                 5446                MOV     AL,020H
1DB4  EE                    5447                OUT     DX,AL
1DB5  FB                    5448                STI
1DB6  C3                    5449                RET
1DB7                        5450        PAL_SET ENDP
                            5451
1DB7                        5452        PAL_ON  PROC    NEAR
1DB7  E8 1DC0 R             5453                CALL    PAL_INIT
1DBA  B2 C0                 5454                MOV     DL,ATTR_WRITE
1DBC  B0 20                 5455                MOV     AL,020H
1DBE  EE                    5456                OUT     DX,AL
1DBF  C3                    5457                RET
1DC0                        5458        PAL_ON  ENDP
                            5459
1DC0                        5460        PAL_INIT        PROC    NEAR
1DC0  E8 0D05 R             5461                CALL    WHAT_BASE
1DC3  EC                    5462                IN      AL,DX
1DC4  C3                    5463                RET
1DC5                        5464        PAL_INIT        ENDP
                            5465
                            5466        ;----- SET PALETTE REGISTERS
                            5467
1DC5                        5468        AH10:
                            5469                ASSUME  DS:ABS0
1DC5  F6 06 0487 R 02       5470                TEST    INFO,2
1DCA  75 07                 5471                JNZ     BM_OK                   ; IN MONOCHROME MODE
                            5472
                            5473        ;----- HERE THE EGA IS IN A COLOR MODE
                            5474
1DCC  80 3E 0463 R B4       5475                CMP     BYTE PTR ADDR_6845,0B4H
1DD1  74 33                 5476                JE      BM_OUT
1DD3                        5477        BM_OK:
1DD3  8A E0                 5478                MOV     AH,AL
1DD5  0A E4                 5479                OR      AH,AH
1DD7  75 30                 5480                JNZ     BM_1
                            5481
                            5482        ;----- SET INDIVIDUAL REGISTER
                            5483
1DD9  2B ED                 5484                SUB     BP,BP
1DDB  C4 3E 04A8 R          5485                LES     DI,SAVE_PTR
1DDF  83 C7 04              5486                ADD     DI,4
1DE2  26: C4 3D             5487                LES     DI,DWORD PTR ES:[DI]
1DE5  8C C0                 5488                MOV     AX,ES
1DE7  0B C7                 5489                OR      AX,DI
1DE9  74 01                 5490                JZ      TLO_1
1DEB  45                    5491                INC     BP
1DEC                        5492        TLO_1:
                            5493
1DEC  E8 1DC0 R             5494                CALL    PAL_INIT
1DEF  8A E3                 5495                MOV     AH,BL
1DF1  8A C7                 5496                MOV     AL,BH
1DF3  E8 1D9F R             5497                CALL    PAL_SET
1DF6  E8 1DB7 R             5498                CALL    PAL_ON
1DF9  0B ED                 5499                OR      BP,BP
1DFB  74 09                 5500                JZ      BM_OUT
1DFD  8A C7                 5501                MOV     AL,BH
1DFF  2A FF                 5502                SUB     BH,BH
1E01  03 FB                 5503                ADD     DI,BX
1E03  26: 88 05             5504                MOV     ES:[DI],AL
1E06                        5505        BM_OUT:
1E06  E9 219E R             5506                JMP     V_RET
                            5507
1E09                        5508        BM_1:
1E09  FE CC                 5509                DEC     AH
1E0B  75 2D                 5510                JNZ     BM_2
                            5511
1E0D  2B ED                 5512                SUB     BP,BP
1E0F  C4 3E 04A8 R          5513                LES     DI,SAVE_PTR
1E13  83 C7 04              5514                ADD     DI,4
1E16  26: C4 3D             5515                LES     DI,DWORD PTR ES:[DI]
1E19  8C C0                 5516                MOV     AX,ES
1E1B  0B C7                 5517                OR      AX,DI
1E1D  74 01                 5518                JZ      TLO_2
1E1F  45                    5519                INC     BP
1E20                        5520        TLO_2:
                            5521
                            5522        ;----- SET OVERSCAN REGISTER
                            5523
1E20  E8 1DC0 R             5524                CALL    PAL_INIT
1E23  B4 11                 5525                MOV     AH,011H
1E25  8A C7                 5526                MOV     AL,BH
1E27  E8 1D9F R             5527                CALL    PAL_SET
1E2A  E8 1DB7 R             5528                CALL    PAL_ON
                            5529
1E2D  0B ED                 5530                OR      BP,BP
1E2F  74 D5                 5531                JZ      BM_OUT
1E31  83 C7 11              5532                ADD     DI,011H
1E34  26: 88 3D             5533                MOV     ES:[DI],BH
                            5534
1E37  E9 219E R             5535                JMP     V_RET
                            5536
1E3A                        5537        BM_2:
1E3A  FE CC                 5538                DEC     AH
1E3C  75 40                 5539                JNZ     BM_3
                            5540
                            5541        ;----- SET 16 PALETTE REGISTERS AND OVERSCAN REGISTER
                            5542
1E3E  1E                    5543                PUSH    DS
1E3F  06                    5544                PUSH    ES
```

```
                                5545
1E40  C4 3E 04A8 R              5546                    LES     DI,SAVE_PTR
1E44  83 C7 04                  5547                    ADD     DI,4
1E47  26: C4 3D                 5548                    LES     DI,DWORD PTR ES:[DI]            ; ES:DI PTR TO PAL SAVE AREA
1E4A  8C C0                     5549                    MOV     AX,ES
1E4C  0B C7                     5550                    OR      AX,DI
1E4E  74 09                     5551                    JZ      TLO_3
                                5552
1E50  1F                        5553                    POP     DS                              ; PARAMETER ES
1E51  1E                        5554                    PUSH    DS
1E52  8B F2                     5555                    MOV     SI,DX                           ; PARAMETER OFFSET
1E54  B9 0011                   5556                    MOV     CX,17D
1E57  F3/ A4                    5557            REP     MOVSB
                                5558
1E59                            5559            TLO_3:
1E59  07                        5560                    POP     ES
1E5A  1F                        5561                    POP     DS
                                5562
1E5B  8B DA                     5563                    MOV     BX,DX
1E5D  E8 1DC0 R                 5564                    CALL    PAL_INIT
1E60  2A E4                     5565                    SUB     AH,AH
                                5566            BM_2A:
1E62  26: 8A 07                 5567                    MOV     AL,ES:[BX]
1E65  E8 1D9F R                 5568                    CALL    PAL_SET
1E68  FE C4                     5569                    INC     AH
1E6A  43                        5570                    INC     BX
1E6B  80 FC 10                  5571                    CMP     AH,010H
1E6E  72 F2                     5572                    JB      BM_2A
1E70  FE C4                     5573                    INC     AH
1E72  26: 8A 07                 5574                    MOV     AL,ES:[BX]
1E75  E8 1D9F R                 5575                    CALL    PAL_SET
1E78  E8 1DB7 R                 5576                    CALL    PAL_ON
1E7B  E9 219E R                 5577                    JMP     V_RET
                                5578
1E7E                            5579            BM_3:
1E7E  FE CC                     5580                    DEC     AH
1E80  75 29                     5581                    JNZ     BM_4
                                5582
                                5583            ;----- TOGGLE INTENSIFY/BLINKING BIT
                                5584
1E82  53                        5585                    PUSH    BX
1E83  E8 0D5A R                 5586                    CALL    MAKE_BASE
1E86  83 C3 33                  5587                    ADD     BX,070H + LN_4
1E89  26: 8A 07                 5588                    MOV     AL,ES:[BX]
1E8C  5B                        5589                    POP     BX
                                5590
1E8D  0A DB                     5591                    OR      BL,BL
1E8F  75 0A                     5592                    JNZ     BM_6
                                5593
                                5594            ;----- ENABLE INTENSIFY
                                5595
1E91  80 26 0465 R DF           5596                    AND     CRT_MODE_SET,11011111B
1E96  24 F7                     5597                    AND     AL,0F7H
1E98  EB 0C 90                  5598                    JMP     BM_7
1E9B                            5599            BM_6:
1E9B  FE CB                     5600                    DEC     BL
1E9D  75 07                     5601                    JNZ     BM_7
                                5602
                                5603            ;----- ENABLE BLINK
                                5604
1E9F  80 0E 0465 R 20           5605                    OR      CRT_MODE_SET,020H
1EA4  0C 08                     5606                    OR      AL,08H
1EA6                            5607            BM_7:
1EA6  B4 10                     5608                    MOV     AH,P_MODE
1EA8  E8 1D9F R                 5609                    CALL    PAL_SET
1EAB                            5610            BM_4:
1EAB  E9 219E R                 5611                    JMP     V_RET
                                5612
                                5613    C               INCLUDE         VCHGEN.INC
                                5614    C               SUBTTL          VCHGEN.INC
                                5615    C               PAGE
                                5616    C       ;--------------------------------------------
                                5617    C       ; ENTRY                                     :
                                5618    C       ;       AL = 0  USER SPECIFIED FONT         :
                                5619    C       ;            1  8 X 14 FONT                 :
                                5620    C       ;            2  8 X 8 DOUBLE DOT            :
                                5621    C       ;       BL = BLOCK TO LOAD                  :
                                5622    C       ;--------------------------------------------
1EAE                            5623    C       CH_GEN:
1EAE  50                        5624    C               PUSH    AX                          ; SAVE THE INVOLVED REGS
1EAF  55                        5625    C               PUSH    BP
1EB0  53                        5626    C               PUSH    BX
1EB1  51                        5627    C               PUSH    CX
1EB2  52                        5628    C               PUSH    DX
1EB3  06                        5629    C               PUSH    ES
                                5630    C
                                5631    C               ASSUME  DS:ABS0
1EB4  E8 0CFE R                 5632    C               CALL    DDS                         ; SET DATA SEGMENT
1EB7  A0 0449 R                 5633    C               MOV     AL,CRT_MODE                 ; GET THE CURRENT MODE
1EBA  50                        5634    C               PUSH    AX                          ; SAVE IT
1EBB  3C 07                     5635    C               CMP     AL,7                        ; IS THIS MONOCHROME
1EBD  74 07                     5636    C               JE      H14                         ; MONOCHROME VALUES
1EBF  C6 06 0449 R 0B           5637    C               MOV     CRT_MODE,0BH                ; COLOR VALUES
1EC4  EB 05                     5638    C               JMP     SHORT H15                   ; SKIP
1EC6                            5639    C       H14:
1EC6  C6 06 0449 R 0C           5640    C               MOV     CRT_MODE,0CH                ; MONOCHROME VALUES
1ECB                            5641    C       H15:
1ECB  E8 0DAB R                 5642    C               CALL    SET_REGS
1ECE  E8 0CFE R                 5643    C               CALL    DDS                         ; RESET THE DATA SEGMENT
1ED1  58                        5644    C               POP     AX                          ; RECOVER OLD MODE VALUE
1ED2  A2 0449 R                 5645    C               MOV     CRT_MODE,AL                 ; RETURN TO LOW MEMORY
                                5646    C
1ED5  07                        5647    C               POP     ES                          ; RESTORE REGS THAT WERE
1ED6  5A                        5648    C               POP     DX                          ;  USED BY THE MODE SET
1ED7  59                        5649    C               POP     CX                          ;  ROUTINES
1ED8  5B                        5650    C               POP     BX
1ED9  5D                        5651    C               POP     BP
1EDA  58                        5652    C               POP     AX
                                5653    C
1EDB  0A C0                     5654    C               OR      AL,AL                       ; SET FLAGS
1EDD  74 17                     5655    C               JZ      DO_MAP2                     ; USER SPECIFIED FONT
1EDF  0E                        5656    C               PUSH    CS                          ; SET SEGMENT TO
1EE0  07                        5657    C               POP     ES                          ;  THIS MODULE
1EE1  2B D2                     5658    C               SUB     DX,DX                       ; ZERO OUT START OFFSET
1EE3  B9 0100                   5659    C               MOV     CX,0256D                    ; CHAR COUNT (FULL SET)
1EE6  FE C8                     5660    C               DEC     AL                          ; WHICH PARAMETER
1EE8  75 07                     5661    C               JNZ     H7                          ; MUST BE ONE
1EEA  B7 0E                     5662    C               MOV     BH,014D                     ; BYTES PER CHARACTER
1EEC  BD 0000 E                 5663    C               MOV     BP,OFFSET CGMN              ; 8 X 14 TABLE OFFSET
1EEF  EB 05                     5664    C               JMP     SHORT DO_MAP2               ; STORE IT
1EF1                            5665    C       H7:
1EF1  B7 08                     5666    C               MOV     BH,8                        ; 8 X 8 FONT
1EF3  BD 0000 E                 5667    C               MOV     BP,OFFSET CGDDOT            ; ROM 8 X 8 DOUBLE DOT
                                5668    C
                                5669    C       ;--------------------------------------------
                                5670    C       ; ALPHA CHARACTER GENERATOR LOAD            :
```

```
                                5671     C ;                                                    :
                                5672     C ; ENTRY                                              :
                                5673     C ;       ES:BP - POINTER TO TABLE                     :
                                5674     C ;       CX    - COUNT OF CHARS                       :
                                5675     C ;       DX    - CHAR COUNT OFFSET INTO MAP 2         :
                                5676     C ;       BH    - BYTES PER CHARACTER                  :
                                5677     C ;       BL    - MAP 2 BLOCK TO LOAD                  :
                                5678     C ;-------------------------------------------------
1EF6                            5679     C DO_MAP2:
1EF6  06                        5680     C         PUSH    ES                                   ; FONT TABLE SEGMENT
1EF7  1F                        5681     C         POP     DS                                   ; ADDRESSING TO TABLE
1EF8  52                        5682     C         PUSH    DX                                   ; SAVE REGISTER
                                5683     C         SRLOAD  ES,0A000H                            ; ADDRESSING TO MAP 2
1EF9  BA A000                   5684     C+        MOV     DX,0A000H
1EFC  8E C2                     5685     C+        MOV     ES,DX
1EFE  5A                        5686     C         POP     DX                                   ; RECOVER REGISTER
1EFF  51                        5687     C         PUSH    CX                                   ; MULTIPLY BY 020H SINCE
1F00  B1 05                     5688     C         MOV     CL,5                                 ;  MAXIMUM BYTES PER
1F02  D3 E2                     5689     C         SHL     DX,CL                                ;  CHARACTER IS 32D=020H
1F04  59                        5690     C         POP     CX                                   ; RECOVER
1F05  0A DB                     5691     C         OR      BL,BL                                ; WHICH 16K BLOCK TO LOAD
1F07  74 08                     5692     C         JZ      H3                                   ; BLOCK ZERO
1F09                            5693     C H4:
1F09  81 C2 4000                5694     C         ADD     DX,04000H                            ; INCREMENT TO NEXT BLOCK
1F0D  FE CB                     5695     C         DEC     BL                                   ; ANY MORE
1F0F  75 F8                     5696     C         JNZ     H4                                   ; DO ANOTHER
1F11                            5697     C H3:
1F11  8A C7                     5698     C         MOV     AL,BH                                ; BYTES PER CHARACTER
1F13  2A E4                     5699     C         SUB     AH,AH                                ; ZERO
1F15  8B FA                     5700     C         MOV     DI,DX                                ; OFFSET INTO MAP
1F17  8B F5                     5701     C         MOV     SI,BP                                ; OFFSET INTO TABLE
1F19  E3 0D                     5702     C         JCXZ    LD_OVER                              ; CHARACTER COUNT
1F1B                            5703     C LD:
1F1B  51                        5704     C         PUSH    CX                                   ; SAVE CHARACTER COUNT
1F1C  8B C8                     5705     C         MOV     CX,AX                                ; ONE ENTIRE CHARACTER
1F1E  F3/ A4                    5706     C         REP     MOVSB                                ;  AT A TIME
1F20  2B F8                     5707     C         SUB     DI,AX                                ; ADJUST OFFSET
1F22  83 C7 20                  5708     C         ADD     DI,020H                              ; NEXT CHARACTER POSITION
1F25  59                        5709     C         POP     CX                                   ; RECOVER CHARACTER COUNT
1F26  E2 F3                     5710     C         LOOP    LD                                   ; DO THE REST
1F28                            5711     C LD_OVER:
1F28  C3                        5712     C         RET
                                5713     C
1F29                            5714     C BRK_1:
                                5715     C         ASSUME  DS:ABS0
1F29  E8 0CFE R                 5716     C         CALL    DDS                                  ; SET LOW MEMORY SEGMENT
1F2C  A3 0485 R                 5717     C         MOV     POINTS,AX                            ; GET BYTES/CHARACTER
1F2F  8B 16 0463 R              5718     C         MOV     DX,ADDR_6845                         ; CRTC REGISTER
1F33  80 3E 0449 R 07           5719     C         CMP     CRT_MODE,7
1F38  75 05                     5720     C         JNE     H11A
1F3A  B4 14                     5721     C         MOV     AH,C_UNDERLN_LOC                     ; R14H
1F3C  E8 0D15 R                 5722     C         CALL    OUT_DX                               ; SET THE UNDERLINE LOC
1F3F                            5723     C H11A:
1F3F  FE C8                     5724     C         DEC     AL                                   ; POINTS - 1
1F41  B4 09                     5725     C         MOV     AH,C_MAX_SCAN_LN                     ; R09H
1F43  E8 0D15 R                 5726     C         CALL    OUT_DX                               ; SET THE CHARACTER HEIGHT
1F46  FE C8                     5727     C         DEC     AL                                   ; POINTS - 2
                                5728     C
1F48  8A E8                     5729     C         MOV     CH,AL                                ; CURSOR START
1F4A  8A C8                     5730     C         MOV     CL,AL                                ; CURSOR END
1F4C  FE C1                     5731     C         INC     CL                                   ; ADJUST END
1F4E  B4 01                     5732     C         MOV     AH,1                                 ; SET C_TYPE BIOS CALL
1F50  CD 10                     5733     C         INT     10H                                  ; SET THE CURSOR
                                5734     C
1F52  8A 1E 0449 R              5735     C         MOV     BL,CRT_MODE                          ; GET THE CURRENT MODE
1F56  B8 015E                   5736     C         MOV     AX,350D                              ; MAX SCANS ON SCREEN
1F59  80 FB 03                  5737     C         CMP     BL,3                                 ; 640X200 ALPHA MODES
1F5C  77 08                     5738     C         JA      H11                                  ; MUST BE 350
1F5E  E8 0E9A R                 5739     C         CALL    BRST_DET
1F61  72 03                     5740     C         JC      H11
1F63  B8 00C8                   5741     C         MOV     AX,200D                              ; SET FOR 200
1F66                            5742     C H11:
1F66  99                        5743     C         CWD                                          ; PREPARE TO DIVIDE
1F67  F7 36 0485 R              5744     C         DIV     POINTS                               ; MAX ROWS ON SCREEN
1F6B  48                        5745     C         DEC     AX                                   ; ADJUST
1F6C  A2 0484 R                 5746     C         MOV     ROWS,AL                              ; SAVE ROWS
1F6F  FE C0                     5747     C         INC     AL                                   ; READJUST
1F71  2A E4                     5748     C         SUB     AH,AH                                ; CLEAR
1F73  F7 26 0485 R              5749     C         MUL     POINTS                               ; ROWS*BYTES/CHAR
1F77  48                        5750     C         DEC     AX                                   ; ADJUST
1F78  8B 16 0463 R              5751     C         MOV     DX,ADDR_6845                         ; CRTC ADDRESS
1F7C  B4 12                     5752     C         MOV     AH,C_VRT_DSP_END                     ; SCANS DISPLAYED
1F7E  E8 0D15 R                 5753     C         CALL    OUT_DX                               ; SET IT
1F81  A0 0484 R                 5754     C         MOV     AL,ROWS                              ; GET CHARACTER ROWS
1F84  FE C0                     5755     C         INC     AL                                   ; ADJUST
1F86  F6 26 044A R              5756     C         MUL     BYTE PTR CRT_COLS                    ; ROWS*COLUMNS
1F8A  D1 E0                     5757     C         SHL     AX,1                                 ; *2 FOR ALPHA MODE
1F8C  05 0100                   5758     C         ADD     AX,256D                              ; SPACE BETWEEN PAGES
1F8F  A3 044C R                 5759     C         MOV     CRT_LEN,AX                           ; BYTES PER PAGE
1F92  E8 0E96 R                 5760     C         CALL    PH_5                                 ; VIDEO ON
1F95  E9 219E R                 5761     C         JMP     V_RET                                ; RETURN TO CALLER
                                5762     C
                                5763     C ;----- LOADABLE CHARACTER GENERATOR ROUTINES
                                5764     C
1F98                            5765     C AH11:
1F98  3C 10                     5766     C         CMP     AL,010H                              ; CHECK PARAMETER
1F9A  73 37                     5767     C         JAE     AH11_ALPHA1                          ; NEXT STAGE
                                5768     C
                                5769     C ;----- ALPHA MODE ACTIVITY HERE
                                5770     C
1F9C  3C 03                     5771     C         CMP     AL,03H                               ; RANGE CHECK
1F9E  73 17                     5772     C         JAE     H1                                   ; NEXT STAGE
1FA0  E8 1EAE R                 5773     C         CALL    CH_GEN                               ; SET THE CHAR GEN
1FA3  E8 0DAB R                 5774     C         CALL    SET_REGS
1FA6  E8 0E96 R                 5775     C         CALL    PH_5                                 ; VIDEO ON
                                5776     C         ASSUME  DS:ABS0
1FA9  E8 0CFE R                 5777     C         CALL    DDS                                  ; SET THE DATA SEGMENT
1FAC  8B 0E 0460 R              5778     C         MOV     CX,CURSOR_MODE                       ; GET THE MODE
1FB0  B4 01                     5779     C         MOV     AH,1                                 ; SET C_TYPE
1FB2  CD 10                     5780     C         INT     10H                                  ; EMULATE CORRECT CURSOR
1FB4  E9 219E R                 5781     C         JMP     V_RET                                ; RETURN TO CALLER
                                5782     C
                                5783     C ;----- SET THE CHARACTER GENERATOR BLOCK SELECT REGISTER
                                5784     C
1FB7                            5785     C H1:
1FB7  75 17                     5786     C         JNE     H2                                   ; NOT IN RANGE
1FB9  B6 03                     5787     C         MOV     DH,3
1FBB  B2 C4                     5788     C         MOV     DL,SEQ_ADDR                          ; SEQUENCER
                                5789     C
1FBD  B8 0001                   5790     C         MOV     AX,1                                 ; AH=S_RESET, AL=1
1FC0  E8 0D15 R                 5791     C         CALL    OUT_DX
                                5792     C
1FC3  B4 03                     5793     C         MOV     AH,S_CGEN                            ; CHAR BLOCK REGISTER
1FC5  8A C3                     5794     C         MOV     AL,BL                                ; GET THE VALUE
1FC7  E8 0D15 R                 5795     C         CALL    OUT_DX                               ; SET IT
                                5796     C
```

```
1FCA  B8 0003                 5797     C          MOV     AX,3                            ; AH=S_RESET, AL=3
1FCD  E8 0D15 R               5798     C          CALL    OUT_DX
1FD0                          5799     C  H2:
1FD0  E9 219E R               5800     C          JMP     V_RET                           ; RETURN TO CALLER
                              5801     C
1FD3                          5802     C  AH11_ALPHA1:
                              5803     C          ASSUME  DS:ABS0
1FD3  3C 20                   5804     C          CMP     AL,020H
1FD5  73 26                   5805     C          JAE     AH11_GRAPHICS
                              5806     C
                              5807     C  ;----- ALPHA MODE ACTIVITY HERE
                              5808     C
1FD7  2C 10                   5809     C          SUB     AL,010H                         ; ADJUST TO 0 - N
1FD9  3C 02                   5810     C          CMP     AL,02H                          ; RANGE CHECK
1FDB  77 F3                   5811     C          JA      H2                              ; INVALID CALL
1FDD  50                      5812     C          PUSH    AX                              ; SAVE
1FDE  53                      5813     C          PUSH    BX
1FDF  E8 1EAE R               5814     C          CALL    CH_GEN                          ; LOAD THE CHAR GEN
1FE2  E8 0DAB R               5815     C          CALL    SET_REGS
1FE5  5B                      5816     C          POP     BX
1FE6  58                      5817     C          POP     AX                              ; RESTORE
1FE7  8A E0                   5818     C          MOV     AH,AL                           ; CALLING PARAMETER
1FE9  0A E4                   5819     C          OR      AH,AH                           ; USER MODE
1FEB  8A C7                   5820     C          MOV     AL,BH
1FED  74 09                   5821     C          JZ      H13                             ; DO NOT SET BYTES/CHAR
1FEF  B0 08                   5822     C          MOV     AL,8                            ; 8 X 8 FONT
1FF1  80 FC 01                5823     C          CMP     AH,1                            ; IS THE CALL FOR MONOC
1FF4  75 02                   5824     C          JNE     H13                             ; NO, LEAVE IT AT 8
1FF6  B0 0E                   5825     C          MOV     AL,14D                          ; MONOC SET
1FF8                          5826     C  H13:
1FF8  2A E4                   5827     C          SUB     AH,AH                           ; CLEAR UPPER BYTE
1FFA  E9 1F29 R               5828     C          JMP     BRK_1                           ; CONTINUE
                              5829     C
                              5830     C  ;----- GRAPHICS MODE ACTIVITY HERE
                              5831     C
1FFD                          5832     C  AH11_GRAPHICS:
                              5833     C          ASSUME  DS:ABS0
1FFD  3C 30                   5834     C          CMP     AL,030H
1FFF  73 6A                   5835     C          JAE     AH11_INFORM
2001  2C 20                   5836     C          SUB     AL,020H
2003  75 11                   5837     C          JNZ     F10
                              5838     C
                              5839     C  ;----- COMPATIBILITY, UPPER HALF GRAPHICS CHARACTER SET
                              5840     C
                              5841     C          ASSUME  DS:ABS0
                              5842     C          SRLOAD  DS,0
2005  2B D2                   5843     C+         SUB     DX,DX
2007  8E DA                   5844     C+         MOV     DS,DX
2009  FA                      5845     C          CLI
200A  89 2E 007C R            5846     C          MOV     WORD PTR EXT_PTR , BP
200E  8C 06 007E R            5847     C          MOV     WORD PTR EXT_PTR + 2 , ES
2012  FB                      5848     C          STI
2013                          5849     C  F11:
2013  E9 219E R               5850     C          JMP     V_RET
2016                          5851     C  F10:
                              5852     C          ASSUME  DS:ABS0
2016  52                      5853     C          PUSH    DX
                              5854     C          SRLOAD  DS,0
2017  2B D2                   5855     C+         SUB     DX,DX
2019  8E DA                   5856     C+         MOV     DS,DX
201B  5A                      5857     C          POP     DX
201C  3C 03                   5858     C          CMP     AL,03H                          ; RANGE CHECK
201E  77 F3                   5859     C          JA      F11
2020  FE C8                   5860     C          DEC     AL
2022  74 14                   5861     C          JZ      F19
2024  0E                      5862     C          PUSH    CS
2025  07                      5863     C          POP     ES
2026  FE C8                   5864     C          DEC     AL
2028  75 08                   5865     C          JNZ     F13
202A  B9 000E                 5866     C          MOV     CX,14D
202D  BD 0000 E               5867     C          MOV     BP,OFFSET CGMN                  ; ROM 8 X 14 CHARACTER SET
2030  EB 06                   5868     C          JMP     SHORT F19
2032                          5869     C  F13:
2032  B9 0008                 5870     C          MOV     CX,8
2035  BD 0000 E               5871     C          MOV     BP,OFFSET CGDDOT                ; ROM 8 X 8 DOUBLE DOT
2038                          5872     C  F19:
2038  FA                      5873     C          CLI
2039  89 2E 010C R            5874     C          MOV     WORD PTR GRX_SET , BP
203D  8C 06 010E R            5875     C          MOV     WORD PTR GRX_SET + 2 , ES
2041  FB                      5876     C          STI
                              5877     C          ASSUME  DS:ABS0
2042  E8 0CFE R               5878     C          CALL    DDS
2045  89 0E 0485 R            5879     C          MOV     POINTS,CX
2049  8A C3                   5880     C          MOV     AL,BL
204B  BB 2067 R               5881     C          MOV     BX,OFFSET RT
204E  0A C0                   5882     C          OR      AL,AL
2050  75 05                   5883     C          JNZ     DR_3
2052  8A C2                   5884     C          MOV     AL,DL
2054  EB 09 90                5885     C          JMP     DR_1
2057                          5886     C  DR_3:
2057  3C 03                   5887     C          CMP     AL,3
2059  76 02                   5888     C          JBE     DR_2
205B  B0 02                   5889     C          MOV     AL,2
205D                          5890     C  DR_2:
205D  2E: D7                  5891     C          XLAT    CS:RT
205F                          5892     C  DR_1:
205F  FE C8                   5893     C          DEC     AL
2061  A2 0484 R               5894     C          MOV     ROWS,AL
2064  E9 219E R               5895     C          JMP     V_RET
                              5896     C
2067                          5897     C  RT      LABEL   BYTE
2067  00 0E 19 2B             5898     C          DB      00D,14D,25D,43D
                              5899     C
                              5900     C
                              5901     C  ;----- INFORMATION RETURN DONE HERE
                              5902     C
206B                          5903     C  AH11_INFORM:
                              5904     C          ASSUME  DS:ABS0
206B  3C 30                   5905     C          CMP     AL,030H
206D  74 03                   5906     C          JE      F6
206F                          5907     C  F5:
206F  E9 219E R               5908     C          JMP     V_RET
2072                          5909     C  F6:
2072  8B 0E 0485 R            5910     C          MOV     CX,POINTS
2076  8A 16 0484 R            5911     C          MOV     DL,ROWS
207A  80 FF 07                5912     C          CMP     BH,7
207D  77 F0                   5913     C          JA      F5
207F  80 FF 01                5914     C          CMP     BH,1
2082  77 18                   5915     C          JA      F7
                              5916     C
                              5917     C          ASSUME  DS:ABS0
2084  52                      5918     C          PUSH    DX
                              5919     C          SRLOAD  DS,0
2085  2B D2                   5920     C+         SUB     DX,DX
2087  8E DA                   5921     C+         MOV     DS,DX
2089  5A                      5922     C          POP     DX
```

```
208A  0A FF                   5923    C           OR      BH,BH
208C  75 07                   5924    C           JNZ     F9
208E  C4 2E 007C R            5925    C           LES     BP,EXT_PTR
2092  EB 1A 90                5926    C           JMP     INFORM_OUT
2095                          5927    C  F9:
2095  C4 2E 010C R            5928    C           LES     BP,GRX_SET
2099  EB 13 90                5929    C           JMP     INFORM_OUT
                              5930    C
                              5931    C  ;----- HANDLE BH = 2  THRU  BH = 5 HERE  RETURN ROM TABLE POINTERS
                              5932    C
209C                          5933    C  F7:
                              5934    C           ASSUME  DS:ABS0
209C  80 EF 02                5935    C           SUB     BH,2
209F  8A DF                   5936    C           MOV     BL,BH
20A1  2A FF                   5937    C           SUB     BH,BH
20A3  D1 E3                   5938    C           SAL     BX,1
20A5  81 C3 20B7 R            5939    C           ADD     BX,OFFSET TBL_5
20A9  2E: 8B 2F               5940    C           MOV     BP,CS:[BX]
20AC  0E                      5941    C           PUSH    CS
20AD  07                      5942    C           POP     ES
                              5943    C
20AE                          5944    C  INFORM_OUT:
20AE  5F                      5945    C           POP     DI
20AF  5E                      5946    C           POP     SI
20B0  5B                      5947    C           POP     BX
20B1  58                      5948    C           POP     AX                              ; DISCARD SAVED CX
20B2  58                      5949    C           POP     AX                              ; DISCARD SAVED DX
20B3  1F                      5950    C           POP     DS
20B4  58                      5951    C           POP     AX                              ; DISCARD SAVED ES
20B5  58                      5952    C           POP     AX                              ; DISCARD SAVED BP
20B6  CF                      5953    C           IRET
                              5954    C
                              5955    C  ;----- TABLE OF CHARACTER GENERATOR OFFSETS
                              5956    C
20B7                          5957    C  TBL_5    LABEL   WORD
20B7  0000 E                  5958    C           DW      OFFSET  CGMN
20B9  0000 E                  5959    C           DW      OFFSET  CGDDOT
20BB  0000 E                  5960    C           DW      OFFSET  INT_1F_1
20BD  0000 E                  5961    C           DW      OFFSET  CGMN_FDG
                              5962    C
                              5963    C           SUBTTL
                              5964
                              5965        ;----- ALTERNATE SELECT
                              5966
20BF                          5967        AH12:
                              5968                ASSUME  DS:ABS0
20BF  80 FB 10                5969                CMP     BL,010H                         ; RETURN ACTIVE CALL
20C2  72 51                   5970                JB      ACT_1
20C4  74 1B                   5971                JE      ACT_3
20C6  80 FB 20                5972                CMP     BL,020H                         ; ALTERNATE PRINT SCREEN
20C9  74 03                   5973                JE      ACT_2
20CB  E9 219E R               5974                JMP     V_RET                           ; INVALID CALL
20CE                          5975        ACT_2:                                          ; NEW PRINT SCREEN
                              5976                SRLOAD  DS,0
20CE  2B D2                   5977      +         SUB     DX,DX
20D0  8E DA                   5978      +         MOV     DS,DX
20D2  FA                      5979                CLI
20D3  C7 06 0014 R 21A7 R     5980                MOV     WORD PTR INT5_PTR, OFFSET PRINT_SCREEN
20D9  8C 0E 0016 R            5981                MOV     WORD PTR INT5_PTR+2, CS
20DD  FB                      5982                STI
20DE  E9 219E R               5983                JMP     V_RET
20E1                          5984        ACT_3:
20E1  8A 3E 0487 R            5985                MOV     BH,INFO                         ; LOOKING FOR MONOC BIT
20E5  80 E7 02                5986                AND     BH,2                            ; ISOLATE
20E8  D0 EF                   5987                SHR     BH,1                            ; ADJUST
                              5988
20EA  A0 0487 R               5989                MOV     AL,INFO                         ; LOOKING FOR MEMORY
20ED  24 60                   5990                AND     AL,01100000B                    ; MEMORY BITS
20EF  B1 05                   5991                MOV     CL,5                            ; SHIFT COUNT
20F1  D2 E8                   5992                SHR     AL,CL                           ; ADJUST MEM VALUE
20F3  8A D8                   5993                MOV     BL,AL                           ; RETURN REGISTER
                              5994
20F5  8A 0E 0488 R            5995                MOV     CL,INFO_3                       ; FEATURE/SWITCH
20F9  8A E9                   5996                MOV     CH,CL                           ; DUPLICATE IN CH
20FB  80 E1 0F                5997                AND     CL,0FH                          ; MASK OFF SWITCH VALUE
20FE  D0 ED                   5998                SHR     CH,1                            ; MOVE FEATURE VALUE
2100  D0 ED                   5999                SHR     CH,1
2102  D0 ED                   6000                SHR     CH,1
2104  D0 ED                   6001                SHR     CH,1
2106  80 E5 0F                6002                AND     CH,0FH                          ; MASK IT
                              6003
2109  5F                      6004                POP     DI
210A  5E                      6005                POP     SI
210B  5A                      6006                POP     DX                              ; DISCARD BX
210C  5A                      6007                POP     DX                              ; DISCARD CX
210D  5A                      6008                POP     DX
210E  1F                      6009                POP     DS
210F  07                      6010                POP     ES
2110  5D                      6011                POP     BP
2111  CF                      6012                IRET
2112                          6013        AH12_X:
2112  E9 219E R               6014                JMP     V_RET                           ; RETURN TO CALLER
2115                          6015        ACT_1:
2115                          6016        STR_OUTZ:
2115  E9 219E R               6017                JMP     V_RET                           ; RETURN TO CALLER
                              6018
                              6019        ;----- WRITE STRING
                              6020
2118                          6021        AH13:
2118  3C 04                   6022                CMP     AL,04                           ; RANGE CHECK
211A  73 F9                   6023                JAE     STR_OUTZ                        ; INVALID PARAMETER
211C  E3 F7                   6024                JCXZ    STR_OUTZ
211E  53                      6025                PUSH    BX                              ; SAVE REGISTER
211F  8A DF                   6026                MOV     BL,BH                           ; GET PAGE TO LOW BYTE
2121  2A FF                   6027                SUB     BH,BH
2123  D1 E3                   6028                SAL     BX,1                            ; *2 FOR WORD OFFSET
2125  8B B7 0450 R            6029                MOV     SI,[BX + OFFSET CURSOR_POSN]    ; GET CURSOR POSITION
2129  5B                      6030                POP     BX                              ; RESTORE
212A  56                      6031                PUSH    SI                              ; CURRENT VALUE ON STACK
                              6032
212B  50                      6033                PUSH    AX
212C  B8 0200                 6034                MOV     AX,0200H                        ; SET THE CURSOR POSITION
212F  CD 10                   6035                INT     10H
2131  58                      6036                POP     AX
2132                          6037        STR_1:
2132  51                      6038                PUSH    CX
2133  53                      6039                PUSH    BX
2134  50                      6040                PUSH    AX
2135  86 E0                   6041                XCHG    AH,AL
2137  26: 8A 46 00            6042                MOV     AL,ES:[BP]                      ; GET THE CHAR TO WRITE
213B  45                      6043                INC     BP
213C  3C 0D                   6044                CMP     AL,0DH                          ; CARRIAGE RETURN
213E  74 3D                   6045                JE      STR_CR_LF
2140  3C 0A                   6046                CMP     AL,0AH                          ; LINE FEED
2142  74 39                   6047                JE      STR_CR_LF
2144  3C 08                   6048                CMP     AL,08H                          ; BACKSPACE
```

```
2146  74 35                6049                JE      STR_CR_LF
2148  3C 07                6050                CMP     AL,07H                          ; BELL
214A  74 31                6051                JE      STR_CR_LF
214C  B9 0001              6052                MOV     CX,1                            ; COUNT OF CHARACTERS
214F  80 FC 02             6053                CMP     AH,2                            ; CHECK WHERE ATTR IS
2152  72 05                6054                JB      DO_STR                          ; NOT IN THE STRING
2154  26: 8A 5E 00          6055                MOV     BL,ES:[BP]                      ; GET THE ATTRIBUTE
2158  45                   6056                INC     BP                              ; NEXT ITEM IN STRING
2159                       6057        DO_STR:
2159  B4 09                6058                MOV     AH,09H                          ; WRITE THE CHAR/ATTR
215B  CD 10                6059                INT     10H
215D  FE C2                6060                INC     DL                              ; NEXT CURSOR POSITION
215F  3A 16 044A R         6061                CMP     DL,BYTE PTR CRT_COLS            ; COLUMN OVERFLOW
2163  72 11                6062                JB      STR_2                           ; NOT YET
2165  3A 36 0484 R         6063                CMP     DH,ROWS
2169  75 07                6064                JNE     STR_3
216B  B8 0E0A              6065                MOV     AX,0E0AH
216E  CD 10                6066                INT     10H
2170  FE CE                6067                DEC     DH
2172                       6068        STR_3:
2172  FE C6                6069                INC     DH                              ; NEXT ROW
2174  2A D2                6070                SUB     DL,DL                           ; COLUMN ZERO
2176                       6071        STR_2:
2176  B8 0200              6072                MOV     AX,0200H                        ; SET THE CURSOR
2179  CD 10                6073                INT     10H
217B  EB 0E                6074                JMP     SHORT   STR_4
217D                       6075        STR_CR_LF:
217D  B4 0E                6076                MOV     AH,0EH
217F  CD 10                6077                INT     10H
2181  8A DF                6078                MOV     BL,BH                           ; GET PAGE TO LOW BYTE
2183  2A FF                6079                SUB     BH,BH
2185  D1 E3                6080                SAL     BX,1                            ; *2 FOR WORD OFFSET
2187  8B 97 0450 R         6081                MOV     DX,[BX + OFFSET CURSOR_POSN]    ; GET CURSOR POSITION
218B                       6082        STR_4:
218B  58                   6083                POP     AX
218C  5B                   6084                POP     BX
218D  59                   6085                POP     CX
218E  E2 A2                6086                LOOP    STR_1
                           6087
2190  5A                   6088                POP     DX                              ; RECOVER CURSOR POSITION
                           6089                                                        ;  FROM PUSH SI ABOVE
2191  3C 01                6090                CMP     AL,1
2193  74 09                6091                JE      STR_OUT
2195  3C 03                6092                CMP     AL,3
2197  74 05                6093                JE      STR_OUT
2199  B8 0200              6094                MOV     AX,0200H                        ; SET CURSOR POSITION
219C  CD 10                6095                INT     10H
219E                       6096        STR_OUT:
                           6097                                                        ; ALLOW FALL THROUGH
                           6098
                           6099
219E                       6100        V_RET   PROC    NEAR                            ; VIDEO BIOS RETURN
219E  5F                   6101                POP     DI
219F  5E                   6102                POP     SI
21A0  5B                   6103                POP     BX
21A1  59                   6104                POP     CX
21A2  5A                   6105                POP     DX
21A3  1F                   6106                POP     DS
21A4  07                   6107                POP     ES
21A5  5D                   6108                POP     BP
21A6  CF                   6109                IRET
21A7                       6110        V_RET   ENDP
                           6111
21A7                       6112        COMBO_VIDEO     ENDP
                           6113
                           6114  C             INCLUDE         VPRSC.INC
                           6115  C             SUBTTL  VPRSC.INC
                           6116  C             PAGE
                           6117  C     ;--------------------------------------------------------------------------
                           6118  C     ;   INTERRUPT 5                                                         :
                           6119  C     ;       THIS LOGIC WILL BE INVOKED BY INTERRUPT 05H TO PRINT THE       :
                           6120  C     ;       SCREEN. THE CURSOR POSITION AT THE TIME THIS ROUTINE IS INVOKED :
                           6121  C     ;       WILL BE SAVED AND RESTORED UPON COMPLETION. THE ROUTINE IS      :
                           6122  C     ;       INTENDED TO RUN WITH INTERRUPTS ENABLED. IF A SUBSEQUENT       :
                           6123  C     ;       'PRINT SCREEN' KEY IS DEPRESSED DURING THE TIME THIS ROUTINE   :
                           6124  C     ;       IS PRINTING IT WILL BE IGNORED.                                :
                           6125  C     ;       ADDRESS 50:0 CONTAINS THE STATUS OF THE PRINT SCREEN:          :
                           6126  C     ;                                                                      :
                           6127  C     ;       50:0    =0      EITHER PRINT SCREEN HAS NOT BEEN CALLED        :
                           6128  C     ;                       OR UPON RETURN FROM A CALL THIS INDICATES      :
                           6129  C     ;                       A SUCCESSFUL OPERATION.                        :
                           6130  C     ;               =1      PRINT SCREEN IS IN PROGRESS                    :
                           6131  C     ;               =255    ERROR ENCOUNTERED DURING PRINTING              :
                           6132  C     ;--------------------------------------------------------------------------
                           6133  C             ASSUME  CS:CODE,DS:ABS0
21A7                       6134  C     PRINT_SCREEN    PROC    FAR
21A7  FB                   6135  C             STI                             ; MUST RUN WITH INTS ENABLED
21A8  1E                   6136  C             PUSH    DS                      ; MUST USE 50:0 FOR DATA
21A9  50                   6137  C             PUSH    AX                      ;  AREA STORAGE
21AA  53                   6138  C             PUSH    BX
21AB  51                   6139  C             PUSH    CX                      ; USE THIS LATER FOR CURSOR LIMITS
21AC  52                   6140  C             PUSH    DX                      ; WILL HOLD CURRENT CURSOR POS
21AD  E8 0CFE R            6141  C             CALL    DDS
21B0  80 3E 0500 R 01      6142  C             CMP     STATUS_BYTE,1           ; SEE IF PRINT ALREADY IN PROGRESS
21B5  74 63                6143  C             JZ      EXIT                    ; JUMP IF PRINT IN PROGRESS
21B7  C6 06 0500 R 01      6144  C             MOV     STATUS_BYTE,1           ; INDICATE PRINT NOW IN PROGRESS
21BC  B4 0F                6145  C             MOV     AH,15                   ; WILL REQUEST THE CURRENT MODE
21BE  CD 10                6146  C             INT     10H                     ;       [AL]=MODE (NOT USED)
                           6147  C                                             ;       [AH]=NUMBER COLUMNS/LINE
                           6148  C                                             ;       [BH]=VISUAL PAGE
                           6149  C     ;--------------------------------------------------------------------------
                           6150  C     ;       AT THIS POINT WE KNOW THE COLUMNS/LINE ARE IN           :
                           6151  C     ;       [AX] AND THE PAGE IF APPLICABLE IS IN [BH]. THE STACK   :
                           6152  C     ;       HAS DS,AX,BX,CX,DX PUSHED. [AL] HAS VIDEO MODE          :
                           6153  C     ;--------------------------------------------------------------------------
21C0  8A CC                6154  C             MOV     CL,AH                   ; WILL MAKE USE OF [CX] REG TO
21C2  8A 2E 0484 R         6155  C             MOV     CH,ROWS                 ; CONTROL ROW & COLUMNS
21C6  FE C5                6156  C             INC     CH                      ; ADJUST
21C8  E8 2220 R            6157  C             CALL    CRLF                    ; CAR RETURN LINE FEED ROUTINE
21CB  51                   6158  C             PUSH    CX                      ; SAVE SCREEN BOUNDS
21CC  B4 03                6159  C             MOV     AH,3                    ; WILL NOW READ THE CURSOR.
21CE  CD 10                6160  C             INT     10H                     ; AND PRESERVE THE POSITION
21D0  59                   6161  C             POP     CX                      ; RECALL SCREEN BOUNDS
21D1  52                   6162  C             PUSH    DX                      ; RECALL [BH]=VISUAL PAGE
21D2  33 D2                6163  C             XOR     DX,DX                   ; SET CURSOR POSITION TO [0,0]
                           6164  C     ;--------------------------------------------------------------------------
                           6165  C     ;       THE LOOP FROM PRI10 TO THE INSTRUCTION PRIOR TO PRI20   :
                           6166  C     ;       IS THE LOOP TO READ EACH CURSOR POSITION FROM THE        :
                           6167  C     ;       SCREEN AND PRINT.                                       :
                           6168  C     ;--------------------------------------------------------------------------
21D4                       6169  C     PRI10:
21D4  B4 02                6170  C             MOV     AH,2                    ; TO INDICATE CURSOR SET REQUEST
21D6  CD 10                6171  C             INT     10H                     ; NEW CURSOR POS ESTABLISHED
21D8  B4 08                6172  C             MOV     AH,8                    ; TO INDICATE READ CHARACTER
21DA  CD 10                6173  C             INT     10H                     ; CHARACTER NOW IN [AL]
21DC  0A C0                6174  C             OR      AL,AL                   ; SEE IF VALID CHAR
```

```
21DE  75 02                6175     C          JNZ     PRI15                   ; JUMP IF VALID CHAR
21E0  B0 20                6176     C          MOV     AL,' '                  ; MAKE A BLANK
21E2                       6177     C  PRI15:
21E2  52                   6178     C          PUSH    DX                      ; SAVE CURSOR POSITION
21E3  33 D2                6179     C          XOR     DX,DX                   ; INDICATE PRINTER 1
21E5  32 E4                6180     C          XOR     AH,AH                   ; TO INDICATE PRINT CHAR IN [AL]
21E7  CD 17                6181     C          INT     17H                     ; PRINT THE CHARACTER
21E9  5A                   6182     C          POP     DX                      ; RECALL CURSOR POSITION
21EA  F6 C4 29             6183     C          TEST    AH,029H                 ; TEST FOR PRINTER ERROR
21ED  75 21                6184     C          JNZ     ERR10                   ; JUMP IF ERROR DETECTED
21EF  FE C2                6185     C          INC     DL                      ; ADVANCE TO NEXT COLUMN
21F1  3A CA                6186     C          CMP     CL,DL                   ; SEE IF AT END OF LINE
21F3  75 DF                6187     C          JNZ     PRI10                   ; IF NOT PROCEED
21F5  32 D2                6188     C          XOR     DL,DL                   ; BACK TO COLUMN 0
21F7  8A E2                6189     C          MOV     AH,DL                   ; [AH]=0
21F9  52                   6190     C          PUSH    DX                      ; SAVE NEW CURSOR POSITION
21FA  E8 2220 R            6191     C          CALL    CRLF                    ; LINE FEED CARRIAGE RETURN
21FD  5A                   6192     C          POP     DX                      ; RECALL CURSOR POSITION
21FE  FE C6                6193     C          INC     DH                      ; ADVANCE TO NEXT LINE
2200  3A EE                6194     C          CMP     CH,DH                   ; FINISHED?
2202  75 D0                6195     C          JNZ     PRI10                   ; IF NOT CONTINUE
                           6196     C
2204  5A                   6197     C          POP     DX                      ; RECALL CURSOR POSITION
2205  B4 02                6198     C          MOV     AH,2                    ; TO INDICATE CURSOR SET REQUEST
2207  CD 10                6199     C          INT     10H                     ; CURSOR POSITION RESTORED
2209  C6 06 0500 R 00      6200     C          MOV     STATUS_BYTE,0           ; INDICATE FINISHED
220E  EB 0A                6201     C          JMP     SHORT EXIT              ; EXIT THE ROUTINE
2210                       6202     C  ERR10:
2210  5A                   6203     C          POP     DX                      ; GET CURSOR POSITION
2211  B4 02                6204     C          MOV     AH,2                    ; TO REQUEST CURSOR SET
2213  CD 10                6205     C          INT     10H                     ; CURSOR POSITION RESTORED
2215  C6 06 0500 R FF      6206     C          MOV     STATUS_BYTE,0FFH        ; INDICATE ERROR
221A                       6207     C  EXIT:
221A  5A                   6208     C          POP     DX                      ; RESTORE ALL THE REGISTERS USED
221B  59                   6209     C          POP     CX
221C  5B                   6210     C          POP     BX
221D  58                   6211     C          POP     AX
221E  1F                   6212     C          POP     DS
221F  CF                   6213     C          IRET
2220                       6214     C  PRINT_SCREEN    ENDP
                           6215     C
                           6216     C  ;------ CARRIAGE RETURN, LINE FEED SUBROUTINE
                           6217     C
2220                       6218     C  CRLF    PROC    NEAR
2220  33 D2                6219     C          XOR     DX,DX                   ; PRINTER 0
2222  32 E4                6220     C          XOR     AH,AH                   ; WILL NOW SEND INITIAL CR, LF
                           6221     C                                          ;  TO PRINTER
2224  B0 0D                6222     C          MOV     AL,0DH                  ; CR
2226  CD 17                6223     C          INT     17H                     ; SEND THE LINE FEED
2228  32 E4                6224     C          XOR     AH,AH                   ; NOW FOR THE CR
222A  B0 0A                6225     C          MOV     AL,0AH                  ; LF
222C  CD 17                6226     C          INT     17H                     ; SEND THE CARRIAGE RETURN
222E  C3                   6227     C          RET
222F                       6228     C  CRLF    ENDP
                           6229     C
                           6230     C          SUBTTL
                           6231
222F                       6232         CODE    ENDS
                           6233                 END


                           1                    PAGE,120
                           2                    SUBTTL  MONOCHROME CHARACTER GENERATOR
0000                       3         CODE    SEGMENT PUBLIC
                           4                 PUBLIC  CGMN
0000                       5         CGMN    LABEL   BYTE
                           6                                                                ; BW 8*14 PATTTERN
0000  00 00 00 00 00 00    7                 DB      000H,000H,000H,000H,000H,000H,000H,000H ; TOP_HALF 00
      00 00
0008  00 00 00 00 00 00    9                 DB      000H,000H,000H,000H,000H,000H           ; BOTTOM_HALF 00
000E  00 00 7E 81 A5 81    10                DB      000H,000H,07EH,081H,0A5H,081H,081H,0BDH ; TH_01
      81 BD
0016  99 81 7E 00 00 00    12                DB      099H,081H,07EH,000H,000H,000H           ; BT_01
001C  00 00 7E FF DB FF    13                DB      000H,000H,07EH,0FFH,0DBH,0FFH,0FFH,0C3H ; TH_02
      FF C3
0024  E7 FF 7E 00 00 00    15                DB      0E7H,0FFH,07EH,000H,000H,000H           ; BT_02
002A  00 00 00 6C FE FE    16                DB      000H,000H,000H,06CH,0FEH,0FEH,0FEH,0FEH ; TH_03
      FE FE
0032  7C 38 10 00 00 00    18                DB      07CH,038H,010H,000H,000H,000H           ; BT_03
0038  00 00 00 10 38 7C    19                DB      000H,000H,000H,010H,038H,07CH,0FEH,07CH ; TH_04
      FE 7C
0040  38 10 00 00 00 00    21                DB      038H,010H,000H,000H,000H,000H           ; BT_04
0046  00 00 18 3C 3C E7    22                DB      000H,000H,018H,03CH,03CH,0E7H,0E7H,0E7H ; TH_05
      E7 E7
004E  18 18 3C 00 00 00    24                DB      018H,018H,03CH,000H,000H,000H           ; BT_05
0054  00 00 18 3C 7E FF    25                DB      000H,000H,018H,03CH,07EH,0FFH,0FFH,07EH ; TH_06
      FF 7E
005C  18 18 3C 00 00 00    27                DB      018H,018H,03CH,000H,000H,000H           ; BT_06
0062  00 00 00 00 00 18    28                DB      000H,000H,000H,000H,000H,018H,03CH,03CH ; TH_07
      3C 3C
006A  18 00 00 00 00 00    30                DB      018H,000H,000H,000H,000H,000H           ; BT_07
0070  FF FF FF FF FF E7    31                DB      0FFH,0FFH,0FFH,0FFH,0FFH,0E7H,0C3H,0C3H ; TH_08
      C3 C3
0078  E7 FF FF FF FF FF    33                DB      0E7H,0FFH,0FFH,0FFH,0FFH,0FFH           ; BT_08
007E  00 00 00 00 3C 66    34                DB      000H,000H,000H,000H,03CH,066H,042H,042H ; TH_09
      42 42
0086  66 3C 00 00 00 00    36                DB      066H,03CH,000H,000H,000H,000H           ; BT_09
008C  FF FF FF FF C3 99    37                DB      0FFH,0FFH,0FFH,0FFH,0C3H,099H,0BDH,0BDH ; TH_0A
      BD BD
0094  99 C3 FF FF FF FF    39                DB      099H,0C3H,0FFH,0FFH,0FFH,0FFH           ; BT_0A
009A  00 00 1E 0E 1A 32    40                DB      000H,000H,01EH,00EH,01AH,032H,078H,0CCH ; TH_0B
      78 CC
00A2  CC CC 78 00 00 00    42                DB      0CCH,0CCH,078H,000H,000H,000H           ; BT_0B
00A8  00 00 3C 66 66 66    43                DB      000H,000H,03CH,066H,066H,066H,03CH,018H ; TH_0C
      3C 18
00B0  7E 18 18 00 00 00    45                DB      07EH,018H,018H,000H,000H,000H           ; BT_0C
00B6  00 00 3F 33 3F 30    46                DB      000H,000H,03FH,033H,03FH,030H,030H,030H ; TH_0D
      30 30
00BE  70 F0 E0 00 00 00    48                DB      070H,0F0H,0E0H,000H,000H,000H           ; BT_0D
00C4  00 00 7F 63 7F 63    49                DB      000H,000H,07FH,063H,07FH,063H,063H,063H ; TH_0E
      63 63
00CC  67 E7 E6 C0 00 00    51                DB      067H,0E7H,0E6H,0C0H,000H,000H           ; BT_0E
00D2  00 00 18 18 DB 3C    52                DB      000H,000H,018H,018H,0DBH,03CH,0E7H,03CH ; TH_0F
      E7 3C
00DA  DB 18 18 00 00 00    54                DB      0DBH,018H,018H,000H,000H,000H           ; BT_0F
                           55
00E0  00 00 80 C0 E0 F8    56                DB      000H,000H,080H,0C0H,0E0H,0F8H,0FEH,0F8H ; TH_10
      FE F8
00E8  E0 C0 80 00 00 00    58                DB      0E0H,0C0H,080H,000H,000H,000H           ; BT_10
00EE  00 00 02 06 0E 3E    59                DB      000H,000H,002H,006H,00EH,03EH,0FEH,03EH ; TH_11
      FE 3E
00F6  0E 06 02 00 00 00    61                DB      00EH,006H,002H,000H,000H,000H           ; BT_11
00FC  00 00 18 3C 7E 18    62                DB      000H,000H,018H,03CH,07EH,018H,018H,018H ; TH_12
      18 18
0104  7E 3C 18 00 00 00    64                DB      07EH,03CH,018H,000H,000H,000H           ; BT_12
010A  00 00 66 66 66 66    65                DB      000H,000H,066H,066H,066H,066H,066H,066H ; TH-13
```

```
                66 66                                   66
0112  00 66 66 00 00 00 00             DB      000H,066H,066H,000H,000H,000H           ; BT_13
0118  00 00 7F DB DB DB                DB      000H,000H,07FH,0DBH,0DBH,0DBH,07BH,01BH ; TH_14
      7B 1B                                     69
0120  1B 1B 1B 00 00 00                DB      01BH,01BH,01BH,000H,000H,000H           ; BT_14
0126  00 7C C6 60 38 6C                DB      000H,07CH,0C6H,060H,038H,06CH,0C6H,0C6H ; TH_15
      C6 C6                                     72
012E  C6 38 0C C6 7C 00                DB      06CH,038H,00CH,0C6H,07CH,000H           ; BT_15
0134  00 00 00 00 00 00                DB      000H,000H,000H,000H,000H,000H,000H,000H ; TH_16
      00 00                                     75
013C  FE FE FE 00 00 00                DB      0FEH,0FEH,0FEH,000H,000H,000H           ; BT_16
0142  00 00 18 3C 7E 18                DB      000H,000H,018H,03CH,07EH,018H,018H,018H ; TH_17
      18 18                                     78
014A  7E 3C 18 7E 00 00                DB      07EH,03CH,018H,07EH,000H,000H           ; BT_17
0150  00 00 18 3C 7E 18                DB      000H,000H,018H,03CH,07EH,018H,018H,018H ; TH_18
      18 18                                     81
0158  18 18 18 00 00 00                DB      018H,018H,018H,000H,000H,000H           ; BT_18
015E  00 00 18 18 18 18                DB      000H,000H,018H,018H,018H,018H,018H,018H ; TH_19
      18 18                                     84
0166  7E 3C 18 00 00 00                DB      07EH,03CH,018H,000H,000H,000H           ; BT_19
016C  00 00 00 00 18 0C                DB      000H,000H,000H,000H,018H,00CH,0FEH,00CH ; TH_1A
      FE 0C                                     87
0174  18 00 00 00 00 00                DB      018H,000H,000H,000H,000H,000H           ; BT_1A
017A  00 00 00 00 30 60                DB      000H,000H,000H,000H,030H,060H,0FEH,060H ; TH_1B
      FE 60                                     90
0182  30 00 00 00 00 00                DB      030H,000H,000H,000H,000H,000H           ; BT_1B
0188  00 00 00 00 00 C0                DB      000H,000H,000H,000H,000H,0C0H,0C0H,0C0H ; TH_1C
      C0 C0                                     93
0190  FE 00 00 00 00 00                DB      0FEH,000H,000H,000H,000H,000H           ; BT_1C
0196  00 00 00 00 28 6C                DB      000H,000H,000H,000H,028H,06CH,0FEH,06CH ; TH_1D
      FE 6C                                     96
019E  28 00 00 00 00 00                DB      028H,000H,000H,000H,000H,000H           ; BT_1D
01A4  00 00 00 10 38 38                DB      000H,000H,000H,010H,038H,038H,07CH,07CH ; TH_1E
      7C 7C                                     99
01AC  FE FE 00 00 00 00                DB      0FEH,0FEH,000H,000H,000H,000H           ; BT_1E
01B2  00 00 00 FE FE 7C                DB      000H,000H,000H,0FEH,0FEH,07CH,07CH,038H ; TH_1F
      7C 38                                     102
01BA  38 10 00 00 00 00                DB      038H,010H,000H,000H,000H,000H           ; BT_1F
                                                104
01C0  00 00 00 00 00 00                DB      000H,000H,000H,000H,000H,000H,000H,000H ; TH_20 SP
      00 00                                     106
01C8  00 00 00 00 00 00                DB      000H,000H,000H,000H,000H,000H           ; BT_20 SP
01CE  00 00 18 3C 3C 3C                DB      000H,000H,018H,03CH,03CH,03CH,018H,018H ; TH_21 !
      18 18                                     109
01D6  00 18 18 00 00 00                DB      000H,018H,018H,000H,000H,000H           ; BT_21 !
01DC  00 66 66 66 24 00                DB      000H,066H,066H,066H,024H,000H,000H,000H ; TH_22 "
      00 00                                     112
01E4  00 00 00 00 00 00                DB      000H,000H,000H,000H,000H,000H           ; BT_22 "
01EA  00 00 6C 6C FE 6C                DB      000H,000H,06CH,06CH,0FEH,06CH,06CH,06CH ; TH_23 #
      6C 6C                                     115
01F2  FE 6C 6C 00 00 00                DB      0FEH,06CH,06CH,000H,000H,000H           ; BT_23 #
01F8  18 18 7C C6 C2 C0                DB      018H,018H,07CH,0C6H,0C2H,0C0H,07CH,006H ; TH_24 $
      7C 06                                     118
0200  86 C6 7C 18 18 00                DB      086H,0C6H,07CH,018H,018H,000H           ; BT_24 $
0206  00 00 00 00 C2 C6                DB      000H,000H,000H,000H,0C2H,0C6H,00CH,018H ; TH_25 '%'
      0C 18                                     121
020E  30 66 C6 00 00 00                DB      030H,066H,0C6H,000H,000H,000H           ; BT_25 '%'
0214  00 00 38 6C 6C 38                DB      000H,000H,038H,06CH,06CH,038H,076H,0DCH ; TH_26 &
      76 DC                                     124
021C  CC CC 76 00 00 00                DB      0CCH,0CCH,076H,000H,000H,000H           ; BT_26 &
0222  00 30 30 30 60 00                DB      000H,030H,030H,030H,060H,000H,000H,000H ; TH_27 '
      00 00                                     127
022A  00 00 00 00 00 00                DB      000H,000H,000H,000H,000H,000H           ; BT 27 '
0230  00 00 0C 18 30 30                DB      000H,000H,00CH,018H,030H,030H,030H,030H ; TH_28 (
      30 30                                     130
0238  30 18 0C 00 00 00                DB      030H,018H,00CH,000H,000H,000H           ; BT_28 (
023E  00 00 30 18 0C 0C                DB      000H,000H,030H,018H,00CH,00CH,00CH,00CH ; TH_29 )
      0C 0C                                     133
0246  0C 18 30 00 00 00                DB      00CH,018H,030H,000H,000H,000H           ; BT_29 )
024C  00 00 00 00 66 3C                DB      000H,000H,000H,000H,066H,03CH,0FFH,03CH ; TH_2A *
      FF 3C                                     136
0254  66 00 00 00 00 00                DB      066H,000H,000H,000H,000H,000H           ; BT_2A *
025A  00 00 00 00 18 18                DB      000H,000H,000H,000H,018H,018H,07EH,018H ; TH_2B +
      7E 18                                     139
0262  18 00 00 00 00 00                DB      018H,000H,000H,000H,000H,000H           ; BT_2B +
0268  00 00 00 00 00 00                DB      000H,000H,000H,000H,000H,000H,000H,000H ; TH_2C ,
      00 00                                     142
0270  18 18 18 30 00 00                DB      018H,018H,018H,030H,000H,000H           ; BT_2C ,
0276  00 00 00 00 00 00                DB      000H,000H,000H,000H,000H,000H,0FEH,000H ; TH_2D -
      FE 00                                     145
027E  00 00 00 00 00 00                DB      000H,000H,000H,000H,000H,000H           ; BT_2D -
0284  00 00 00 00 00 00                DB      000H,000H,000H,000H,000H,000H,000H,000H ; TH_2E .
      00 00                                     148
028C  00 18 18 00 00 00                DB      000H,018H,018H,000H,000H,000H           ; BT_2E .
0292  00 00 02 06 0C 18                DB      000H,000H,002H,006H,00CH,018H,030H,060H ; TH_2F /
      30 60                                     151
029A  C0 80 00 00 00 00                DB      0C0H,080H,000H,000H,000H,000H           ; BT_2F /
                                                153
02A0  00 00 7C C6 CE DE                DB      000H,000H,07CH,0C6H,0CEH,0DEH,0F6H,0E6H ; TH_30 0
      F6 E6                                     155
02A8  C6 C6 7C 00 00 00                DB      0C6H,0C6H,07CH,000H,000H,000H           ; BT_30 0
02AE  00 00 18 38 78 18                DB      000H,000H,018H,038H,078H,018H,018H,018H ; TH_31 1
      18 18                                     158
02B6  18 18 7E 00 00 00                DB      018H,018H,07EH,000H,000H,000H           ; BT_31 1
02BC  00 00 7C C6 06 0C                DB      000H,000H,07CH,0C6H,006H,00CH,018H,030H ; TH_32 2
      18 30                                     161
02C4  60 C6 FE 00 00 00                DB      060H,0C6H,0FEH,000H,000H,000H           ; BT_32 2
02CA  00 00 7C C6 06 06                DB      000H,000H,07CH,0C6H,006H,006H,03CH,006H ; TH_33 3
      3C 06                                     164
02D2  06 C6 7C 00 00 00                DB      006H,0C6H,07CH,000H,000H,000H           ; BT_33 3
02D8  00 00 0C 1C 3C 6C                DB      000H,000H,00CH,01CH,03CH,06CH,0CCH,0FEH ; TH_34 4
      CC FE                                     167
02E0  0C 0C 1E 00 00 00                DB      00CH,00CH,01EH,000H,000H,000H           ; BT_34 4
02E6  00 00 FE C0 C0 C0                DB      000H,000H,0FEH,0C0H,0C0H,0C0H,0FCH,006H ; TH_35 5
      FC 06                                     170
02EE  06 C6 7C 00 00 00                DB      006H,0C6H,07CH,000H,000H,000H           ; BT_35 5
02F4  00 00 38 60 C0 C0                DB      000H,000H,038H,060H,0C0H,0C0H,0FCH,0C6H ; TH_36 6
      FC C6                                     173
02FC  C6 C6 7C 00 00 00                DB      0C6H,0C6H,07CH,000H,000H,000H           ; BT_36 6
0302  00 00 FE C6 06 0C                DB      000H,000H,0FEH,0C6H,006H,00CH,018H,030H ; TH_37 7
      18 30                                     176
030A  30 30 30 00 00 00                DB      030H,030H,030H,000H,000H,000H           ; BT_37 7
0310  00 00 7C C6 C6 C6                DB      000H,000H,07CH,0C6H,0C6H,0C6H,07CH,0C6H ; TH_38 8
      7C C6                                     179
0318  C6 C6 7C 00 00 00                DB      0C6H,0C6H,07CH,000H,000H,000H           ; BT_38 8
031E  00 00 7C C6 C6 C6                DB      000H,000H,07CH,0C6H,0C6H,0C6H,07EH,006H ; TH_39 9
      7E 06                                     182
0326  06 0C 78 00 00 00                DB      006H,00CH,078H,000H,000H,000H           ; BT_39 9
032C  00 00 00 18 18 00                DB      000H,000H,000H,018H,018H,000H,000H,000H ; TH_3A :
      00 00                                     185
0334  18 18 00 00 00 00                DB      018H,018H,000H,000H,000H,000H           ; BT_3A :
033A  00 00 00 18 18 00                DB      000H,000H,000H,018H,018H,000H,000H,000H ; TH_3B ;
      00 00                                     188
0342  18 18 30 00 00 00                DB      018H,018H,030H,000H,000H,000H           ; BT_3B ;
0348  00 00 06 0C 18 30                DB      000H,000H,006H,00CH,018H,030H,060H,030H ; TH_3C <
      60 30                                     191
```

```
0350  18 0C 06 00 00 00      192                    DB      018H,00CH,006H,000H,000H,000H           ; BT_3C <
0356  00 00 00 00 00 7E      193                    DB      000H,000H,000H,000H,000H,07EH,000H,000H ; TH_3D =
      00 00
035E  7E 00 00 00 00 00      195                    DB      07EH,000H,000H,000H,000H,000H           ; BT_3D =
0364  00 00 60 30 18 0C      196                    DB      000H,000H,060H,030H,018H,00CH,006H,00CH ; TH_3E >
      06 0C                  197
036C  18 30 60 00 00 00      198                    DB      018H,030H,060H,000H,000H,000H           ; BT_3E >
0372  00 00 7C C6 C6 0C      199                    DB      000H,000H,07CH,0C6H,0C6H,00CH,018H,018H ; TH_3F ?
      18 18                  200
037A  00 18 18 00 00 00      201                    DB      000H,018H,018H,000H,000H,000H           ; BT_3F ?
                             202
0380  00 00 7C C6 C6 DE      203                    DB      000H,000H,07CH,0C6H,0C6H,0DEH,0DEH,0DEH ; TH_40 @
      DE DE                  204
0388  DC C0 7C 00 00 00      205                    DB      0DCH,0C0H,07CH,000H,000H,000H           ; BT_40 @
038E  00 00 10 38 6C C6      206                    DB      000H,000H,010H,038H,06CH,0C6H,0C6H,0FEH ; TH_41 A
      C6 FE                  207
0396  C6 C6 C6 00 00 00      208                    DB      0C6H,0C6H,0C6H,000H,000H,000H           ; BT_41 A
039C  00 00 FC 66 66 66      209                    DB      000H,000H,0FCH,066H,066H,066H,07CH,066H ; TH_42 B
      7C 66                  210
03A4  66 66 FC 00 00 00      211                    DB      066H,066H,0FCH,000H,000H,000H           ; BT_42 B
03AA  00 00 3C 66 C2 C0      212                    DB      000H,000H,03CH,066H,0C2H,0C0H,0C0H,0C0H ; TH_43 C
      C0 C0                  213
03B2  C2 66 3C 00 00 00      214                    DB      0C2H,066H,03CH,000H,000H,000H           ; BT_43 C
03B8  00 00 F8 6C 66 66      215                    DB      000H,000H,0F8H,06CH,066H,066H,066H,066H ; TH_44 D
      66 66                  216
03C0  66 6C F8 00 00 00      217                    DB      066H,06CH,0F8H,000H,000H,000H           ; BT_44 D
03C6  00 00 FE 66 62 68      218                    DB      000H,000H,0FEH,066H,062H,068H,078H,068H ; TH_45 E
      78 68                  219
03CE  62 66 FE 00 00 00      220                    DB      062H,066H,0FEH,000H,000H,000H           ; BT_45 E
03D4  00 00 FE 66 62 68      221                    DB      000H,000H,0FEH,066H,062H,068H,078H,068H ; TH_46 F
      78 68                  222
03DC  60 60 F0 00 00 00      223                    DB      060H,060H,0F0H,000H,000H,000H           ; BT_46 F
03E2  00 00 3C 66 C2 C0      224                    DB      000H,000H,03CH,066H,0C2H,0C0H,0C0H,0DEH ; TH_47 G
      C0 DE                  225
03EA  C6 66 3A 00 00 00      226                    DB      0C6H,066H,03AH,000H,000H,000H           ; BT_47 G
03F0  00 00 C6 C6 C6 C6      227                    DB      000H,000H,0C6H,0C6H,0C6H,0C6H,0FEH,0C6H ; TH_48 H
      FE C6                  228
03F8  C6 C6 C6 00 00 00      229                    DB      0C6H,0C6H,0C6H,000H,000H,000H           ; BT_48 H
03FE  00 00 3C 18 18 18      230                    DB      000H,000H,03CH,018H,018H,018H,018H,018H ; TH_49 I
      18 18                  231
0406  18 18 3C 00 00 00      232                    DB      018H,018H,03CH,000H,000H,000H           ; BT_49 I
040C  00 00 1E 0C 0C 0C      233                    DB      000H,000H,01EH,00CH,00CH,00CH,00CH,00CH ; TH_4A J
      0C 0C                  234
0414  CC CC 78 00 00 00      235                    DB      0CCH,0CCH,078H,000H,000H,000H           ; BT_4A J
041A  00 00 E6 66 6C 6C      236                    DB      000H,000H,0E6H,066H,06CH,06CH,078H,06CH ; TH_4B K
      78 6C                  237
0422  6C 66 E6 00 00 00      238                    DB      06CH,066H,0E6H,000H,000H,000H           ; BT_4B K
0428  00 00 F0 60 60 60      239                    DB      000H,000H,0F0H,060H,060H,060H,060H,060H ; TH_4C L
      60 60                  240
0430  62 66 FE 00 00 00      241                    DB      062H,066H,0FEH,000H,000H,000H           ; BT_4C L
0436  00 00 C6 EE FE FE      242                    DB      000H,000H,0C6H,0EEH,0FEH,0FEH,0D6H,0C6H ; TH_4D M
      D6 C6                  243
043E  C6 C6 C6 00 00 00      244                    DB      0C6H,0C6H,0C6H,000H,000H,000H           ; BT_4D M
0444  00 00 C6 E6 F6 FE      245                    DB      000H,000H,0C6H,0E6H,0F6H,0FEH,0DEH,0CEH ; TH_4E N
      DE CE                  246
044C  C6 C6 C6 00 00 00      247                    DB      0C6H,0C6H,0C6H,000H,000H,000H           ; BT_4E N
0452  00 00 38 6C C6 C6      248                    DB      000H,000H,038H,06CH,0C6H,0C6H,0C6H,0C6H ; TH_4F O
      C6 C6                  249
045A  C6 6C 38 00 00 00      250                    DB      0C6H,06CH,038H,000H,000H,000H           ; BT_4F O
                             251
0460  00 00 FC 66 66 66      252                    DB      000H,000H,0FCH,066H,066H,066H,07CH,060H ; TH_50 P
      7C 60                  253
0468  60 60 F0 00 00 00      254                    DB      060H,060H,0F0H,000H,000H,000H           ; BT_50 P
046E  00 00 7C C6 C6 C6      255                    DB      000H,000H,07CH,0C6H,0C6H,0C6H,0C6H,0D6H ; TH_51 Q
      C6 D6                  256
0476  DE 7C 0C 0E 00 00      257                    DB      0DEH,07CH,00CH,00EH,000H,000H           ; BT_51 Q
047C  00 00 FC 66 66 66      258                    DB      000H,000H,0FCH,066H,066H,066H,07CH,06CH ; TH_52 R
      7C 6C                  259
0484  66 66 E6 00 00 00      260                    DB      066H,066H,0E6H,000H,000H,000H           ; BT_52 R
048A  00 00 7C C6 C6 60      261                    DB      000H,000H,07CH,0C6H,0C6H,060H,038H,00CH ; TH_53 S
      38 0C                  262
0492  C6 C6 7C 00 00 00      263                    DB      0C6H,0C6H,07CH,000H,000H,000H           ; BT_53 S
0498  00 00 7E 7E 5A 18      264                    DB      000H,000H,07EH,07EH,05AH,018H,018H,018H ; TH_54 T
      18 18                  265
04A0  18 18 3C 00 00 00      266                    DB      018H,018H,03CH,000H,000H,000H           ; BT_54 T
04A6  00 00 C6 C6 C6 C6      267                    DB      000H,000H,0C6H,0C6H,0C6H,0C6H,0C6H,0C6H ; TH_55 U
      C6 C6                  268
04AE  C6 C6 7C 00 00 00      269                    DB      0C6H,0C6H,07CH,000H,000H,000H           ; BT_55 U
04B4  00 00 C6 C6 C6 C6      270                    DB      000H,000H,0C6H,0C6H,0C6H,0C6H,0C6H,0C6H ; TH_56 V
      C6 C6                  271
04BC  6C 38 10 00 00 00      272                    DB      06CH,038H,010H,000H,000H,000H           ; BT_56 V
04C2  00 00 C6 C6 C6 C6      273                    DB      000H,000H,0C6H,0C6H,0C6H,0C6H,0D6H,0D6H ; TH_57 W
      D6 D6                  274
04CA  FE 7C 6C 00 00 00      275                    DB      0FEH,07CH,06CH,000H,000H,000H           ; BT_57 W
04D0  00 00 C6 C6 6C 38      276                    DB      000H,000H,0C6H,0C6H,06CH,038H,038H,038H ; TH_58 X
      38 38                  277
04D8  C6 C6 C6 00 00 00      278                    DB      06CH,0C6H,0C6H,000H,000H,000H           ; BT_58 X
04DE  00 00 66 66 66 66      279                    DB      000H,000H,066H,066H,066H,066H,03CH,018H ; TH_59 Y
      3C 18                  280
04E6  18 18 3C 00 00 00      281                    DB      018H,018H,03CH,000H,000H,000H           ; BT_59 Y
04EC  00 00 FE C6 8C 18      282                    DB      000H,000H,0FEH,0C6H,08CH,018H,030H,060H ; TH_5A Z
      30 60                  283
04F4  C2 C6 FE 00 00 00      284                    DB      0C2H,0C6H,0FEH,000H,000H,000H           ; BT_5A Z
04FA  00 00 3C 30 30 30      285                    DB      000H,000H,03CH,030H,030H,030H,030H,030H ; TH_5B [
      30 30                  286
0502  30 30 3C 00 00 00      287                    DB      030H,030H,03CH,000H,000H,000H           ; BT_5B [
0508  00 00 80 C0 E0 70      288                    DB      000H,000H,080H,0C0H,0E0H,070H,038H,01CH ; TH_5C
      38 1C                  289
0510  0E 06 02 00 00 00      290                    DB      00EH,006H,002H,000H,000H,000H           ; BT_5C
0516  00 00 3C 0C 0C 0C      291                    DB      000H,000H,03CH,00CH,00CH,00CH,00CH,00CH ; TH_5D ]
      0C 0C                  292
051E  0C 0C 3C 00 00 00      293                    DB      00CH,00CH,03CH,000H,000H,000H           ; BT_5D ]
0524  10 38 6C C6 00 00      294                    DB      010H,038H,06CH,0C6H,000H,000H,000H,000H ; TH_5E
      00 00                  295
052C  00 00 00 00 00 00      296                    DB      000H,000H,000H,000H,000H,000H           ; BT_5E
0532  00 00 00 00 00 00      297                    DB      000H,000H,000H,000H,000H,000H,000H,000H ; TH_5F _
      00 00                  298
053A  00 00 00 00 FF 00      299                    DB      000H,000H,000H,000H,0FFH,000H           ; BT_5F _
                             300
0540  30 30 18 00 00 00      301                    DB      030H,030H,018H,000H,000H,000H,000H,000H ; TH_60 '
      00 00                  302
0548  00 00 00 00 00 00      303                    DB      000H,000H,000H,000H,000H,000H           ; BT_60 '
054E  00 00 00 00 00 78      304                    DB      000H,000H,000H,000H,000H,078H,00CH,07CH ; TH_61 LOWER_CASE A
      0C 7C                  305
0556  CC CC 76 00 00 00      306                    DB      0CCH,0CCH,076H,000H,000H,000H           ; BT_61 LOWER_CASE A
055C  00 00 E0 60 60 78      307                    DB      000H,000H,0E0H,060H,060H,078H,06CH,066H ; TH_62 L.C. B
      6C 66                  308
0564  66 66 7C 00 00 00      309                    DB      066H,066H,07CH,000H,000H,000H           ; BT_62 L.C. B
056A  00 00 00 00 00 7C      310                    DB      000H,000H,000H,000H,000H,07CH,0C6H,0C0H ; TH_63 L.C. C
      C6 C0                  311
0572  C0 C6 7C 00 00 00      312                    DB      0C0H,0C6H,07CH,000H,000H,000H           ; BT_63 L.C. C
0578  00 00 1C 0C 0C 3C      313                    DB      000H,000H,01CH,00CH,00CH,03CH,06CH,0CCH ; TH_64 L.C. D
      6C CC                  314
0580  CC CC 76 00 00 00      315                    DB      0CCH,0CCH,076H,000H,000H,000H           ; BT_64 L.C. D
0586  00 00 00 00 00 7C      316                    DB      000H,000H,000H,000H,000H,07CH,0C6H,0FEH ; TH_65 L.C. E
      C6 FE                  317
```

```
058E  C0 C6 7C 00 00 00      318                  DB      0C0H,0C6H,07CH,000H,000H,000H           ; BT_65 L.C. E
0594  00 00 38 6C 64 60      319                  DB      000H,000H,038H,06CH,064H,060H,0F0H,060H ; TH_66 L.C. F
      F0 60
                             320
059C  60 60 F0 00 00 00      321                  DB      060H,060H,0F0H,000H,000H,000H           ; BT_66 L.C. F
05A2  00 00 00 00 00 76      322                  DB      000H,000H,000H,000H,000H,076H,0CCH,0CCH ; TH_67 L.C. G
      CC CC
                             323
05AA  CC 7C 0C CC 78 00      324                  DB      0CCH,07CH,00CH,0CCH,078H,000H           ; BT_67 L.C. G
05B0  00 00 E0 60 60 6C      325                  DB      000H,000H,0E0H,060H,060H,06CH,076H,066H ; TH_68 L.C. H
      76 66
                             326
05B8  66 66 E6 00 00 00      327                  DB      066H,066H,0E6H,000H,000H,000H           ; BT_68 L.C. H
05BE  00 00 18 18 00 38      328                  DB      000H,000H,018H,018H,000H,038H,018H,018H ; TH_69 L.C. I
      18 18
                             329
05C6  18 18 3C 00 00 00      330                  DB      018H,018H,03CH,000H,000H,000H           ; BT_69 L.C. I
05CC  00 00 06 06 00 0E      331                  DB      000H,000H,006H,006H,000H,00EH,006H,006H ; TH_6A L.C. J
      06 06
                             332
05D4  06 06 66 66 3C 00      333                  DB      006H,006H,066H,066H,03CH,000H           ; BT_6A L.C. J
05DA  00 00 E0 60 60 66      334                  DB      000H,000H,0E0H,060H,060H,066H,06CH,078H ; TH_6B L.C. K
      6C 78
                             335
05E2  6C 66 E6 00 00 00      336                  DB      06CH,066H,0E6H,000H,000H,000H           ; BT_6B L.C. K
05E8  00 00 38 18 18 18      337                  DB      000H,000H,038H,018H,018H,018H,018H,018H ; TH_6C L.C. L
      18 18
                             338
05F0  18 18 3C 00 00 00      339                  DB      018H,018H,03CH,000H,000H,000H           ; BT_6C L.C. L
05F6  00 00 00 00 00 EC      340                  DB      000H,000H,000H,000H,000H,0ECH,0FEH,0D6H ; TH_6D L.C. M
      FE D6
                             341
05FE  D6 D6 C6 00 00 00      342                  DB      0D6H,0D6H,0C6H,000H,000H,000H           ; BT_6D L.C. M
0604  00 00 00 00 00 DC      343                  DB      000H,000H,000H,000H,000H,0DCH,066H,066H ; TH_6E L.C. N
      66 66
                             344
060C  66 66 66 00 00 00      345                  DB      066H,066H,066H,000H,000H,000H           ; BT_6E L.C. N
0612  00 00 00 00 00 7C      346                  DB      000H,000H,000H,000H,000H,07CH,0C6H,0C6H ; TH_6F L.C. O
      C6 C6
                             347
061A  C6 C6 7C 00 00 00      348                  DB      0C6H,0C6H,07CH,000H,000H,000H           ; BT_6F L.C. O
                             349
0620  00 00 00 00 00 DC      350                  DB      000H,000H,000H,000H,000H,0DCH,066H,066H ; TH_70 L.C. P
      66 66
                             351
0628  66 7C 60 60 F0 00      352                  DB      066H,07CH,060H,060H,0F0H,000H           ; BT_70 L.C. P
062E  00 00 00 00 00 76      353                  DB      000H,000H,000H,000H,000H,076H,0CCH,0CCH ; TH_71 L.C. Q
      CC CC
                             354
0636  CC 7C 0C 0C 1E 00      355                  DB      0CCH,07CH,00CH,00CH,01EH,000H           ; BT_71 L.C. Q
063C  00 00 00 00 00 DC      356                  DB      000H,000H,000H,000H,000H,0DCH,076H,066H ; TH_72 L.C. R
      76 66
                             357
0644  60 60 F0 00 00 00      358                  DB      060H,060H,0F0H,000H,000H,000H           ; BT_72 L.C. R
064A  00 00 00 00 00 7C      359                  DB      000H,000H,000H,000H,000H,07CH,0C6H,070H ; TH_73 L.C. S
      C6 70
                             360
0652  1C C6 7C 00 00 00      361                  DB      01CH,0C6H,07CH,000H,000H,000H           ; BT_73 L.C. S
0658  00 00 10 30 30 FC      362                  DB      000H,000H,010H,030H,030H,0FCH,030H,030H ; TH_74 L.C. T
      30 30
                             363
0660  30 36 1C 00 00 00      364                  DB      030H,036H,01CH,000H,000H,000H           ; BT_74 L.C. T
0666  00 00 00 00 00 CC      365                  DB      000H,000H,000H,000H,000H,0CCH,0CCH,0CCH ; TH_75 L.C. U
      CC CC
                             366
066E  CC CC 76 00 00 00      367                  DB      0CCH,0CCH,076H,000H,000H,000H           ; BT_75 L.C. U
0674  00 00 00 00 00 66      368                  DB      000H,000H,000H,000H,000H,066H,066H,066H ; TH_76 L.C. V
      66 66
                             369
067C  66 3C 18 00 00 00      370                  DB      066H,03CH,018H,000H,000H,000H           ; BT_76 L.C. V
0682  00 00 00 00 00 C6      371                  DB      000H,000H,000H,000H,000H,0C6H,0C6H,0D6H ; TH_77 L.C. W
      C6 D6
                             372
068A  D6 FE 6C 00 00 00      373                  DB      0D6H,0FEH,06CH,000H,000H,000H           ; BT_77 L.C. W
0690  00 00 00 00 00 C6      374                  DB      000H,000H,000H,000H,000H,0C6H,06CH,038H ; TH_78 L.C. X
      6C 38
                             375
0698  38 6C C6 00 00 00      376                  DB      038H,06CH,0C6H,000H,000H,000H           ; BT_78 L.C. X
069E  00 00 00 00 00 C6      377                  DB      000H,000H,000H,000H,000H,0C6H,0C6H,0C6H ; TH_79 L.C. Y
      C6 C6
                             378
06A6  C6 7E 06 0C F8 00      379                  DB      0C6H,07EH,006H,00CH,0F8H,000H           ; BT_79 L.C. Y
06AC  00 00 00 00 00 FE      380                  DB      000H,000H,000H,000H,000H,0FEH,0CCH,018H ; TH_7A L.C. Z
      CC 18
                             381
06B4  30 66 FE 00 00 00      382                  DB      030H,066H,0FEH,000H,000H,000H           ; BT_7A L.C. Z
06BA  00 00 0E 18 18 18      383                  DB      000H,000H,00EH,018H,018H,018H,070H,018H ; TH_7B L BRAK
      70 18
                             384
06C2  18 18 0E 00 00 00      385                  DB      018H,018H,00EH,000H,000H,000H           ; BT_7B L BRAK
06C8  00 00 18 18 18 18      386                  DB      000H,000H,018H,018H,018H,018H,000H,018H ; TH_7C |
      00 18
                             387
06D0  18 18 18 00 00 00      388                  DB      018H,018H,018H,000H,000H,000H           ; BT_7C |
06D6  00 00 70 18 18 18      389                  DB      000H,000H,070H,018H,018H,018H,00EH,018H ; TH_7D R BRAK
      0E 18
                             390
06DE  18 18 70 00 00 00      391                  DB      018H,018H,070H,000H,000H,000H           ; BT_7D R BRAK
06E4  00 00 76 DC 00 00      392                  DB      000H,000H,076H,0DCH,000H,000H,000H,000H ; TH_7E TILDE
      00 00
                             393
06EC  00 00 00 00 00 00      394                  DB      000H,000H,000H,000H,000H,000H           ; BT_7E TILDE
06F2  00 00 00 00 10 38      395                  DB      000H,000H,000H,000H,010H,038H,06CH,0C6H ; TH_7F DELTA
      6C C6
                             396
06FA  C6 FE 00 00 00 00      397                  DB      0C6H,0FEH,000H,000H,000H,000H           ; BT_7F DELTA
                             398
0700  00 00 3C 66 C2 C0      399                  DB      000H,000H,03CH,066H,0C2H,0C0H,0C0H,0C2H ; TH_80
      C0 C2
                             400
0708  66 3C 0C 06 7C 00      401                  DB      066H,03CH,00CH,006H,07CH,000H           ; BT_80
070E  00 00 CC CC 00 CC      402                  DB      000H,000H,0CCH,0CCH,000H,0CCH,0CCH,0CCH ; TH_81
      CC CC
                             403
0716  CC CC 76 00 00 00      404                  DB      0CCH,0CCH,076H,000H,000H,000H           ; BT_81
071C  00 0C 18 30 00 7C      405                  DB      000H,00CH,018H,030H,000H,07CH,0C6H,0FEH ; TH_82
      C6 FE
                             406
0724  C0 C6 7C 00 00 00      407                  DB      0C0H,0C6H,07CH,000H,000H,000H           ; BT_82
072A  00 10 38 6C 00 78      408                  DB      000H,010H,038H,06CH,000H,078H,00CH,07CH ; TH_83
      0C 7C
                             409
0732  CC CC 76 00 00 00      410                  DB      0CCH,0CCH,076H,000H,000H,000H           ; BT_83
0738  00 00 CC CC 00 78      411                  DB      000H,000H,0CCH,0CCH,000H,078H,00CH,07CH ; TH_84
      0C 7C
                             412
0740  CC CC 76 00 00 00      413                  DB      0CCH,0CCH,076H,000H,000H,000H           ; BT_84
0746  00 60 30 18 00 78      414                  DB      000H,060H,030H,018H,000H,078H,00CH,07CH ; TH_85
      0C 7C
                             415
074E  CC CC 76 00 00 00      416                  DB      0CCH,0CCH,076H,000H,000H,000H           ; BT_85
0754  00 38 6C 38 00 78      417                  DB      000H,038H,06CH,038H,000H,078H,00CH,07CH ; TH_86
      0C 7C
                             418
075C  CC CC 76 00 00 00      419                  DB      0CCH,0CCH,076H,000H,000H,000H           ; BT_86
0762  00 00 00 00 3C 66      420                  DB      000H,000H,000H,000H,03CH,066H,060H,066H ; TH_87
      60 66
                             421
076A  3C 0C 06 3C 00 00      422                  DB      03CH,00CH,006H,03CH,000H,000H           ; BT_87
0770  00 10 38 6C 00 7C      423                  DB      000H,010H,038H,06CH,000H,07CH,0C6H,0FEH ; TH_88
      C6 FE
                             424
0778  C0 C6 7C 00 00 00      425                  DB      0C0H,0C6H,07CH,000H,000H,000H           ; BT_88
077E  00 00 CC CC 00 7C      426                  DB      000H,000H,0CCH,0CCH,000H,07CH,0C6H,0FEH ; TH_89
      C6 FE
                             427
0786  C0 C6 7C 00 00 00      428                  DB      0C0H,0C6H,07CH,000H,000H,000H           ; BT_89
078C  00 60 30 18 00 7C      429                  DB      000H,060H,030H,018H,000H,07CH,0C6H,0FEH ; TH_8A
      C6 FE
                             430
0794  C0 C6 7C 00 00 00      431                  DB      0C0H,0C6H,07CH,000H,000H,000H           ; BT_8A
079A  00 00 66 66 00 38      432                  DB      000H,000H,066H,066H,000H,038H,018H,018H ; TH_8B
      18 18
                             433
07A2  18 18 3C 00 00 00      434                  DB      018H,018H,03CH,000H,000H,000H           ; BT_8B
07A8  00 18 3C 66 00 38      435                  DB      000H,018H,03CH,066H,000H,038H,018H,018H ; TH_8C
      18 18
                             436
07B0  18 18 3C 00 00 00      437                  DB      018H,018H,03CH,000H,000H,000H           ; BT_8C
07B6  00 60 30 18 00 38      438                  DB      000H,060H,030H,018H,000H,038H,018H,018H ; TH_8D
      18 18
                             439
07BE  18 18 3C 00 00 00      440                  DB      018H,018H,03CH,000H,000H,000H           ; BT_8D
07C4  00 C6 C6 10 38 6C      441                  DB      000H,0C6H,0C6H,010H,038H,06CH,0C6H,0C6H ; TH_8E
      C6 C6
                             442
07CC  FE C6 C6 00 00 00      443                  DB      0FEH,0C6H,0C6H,000H,000H,000H           ; BT_8E
```

```
07D2  38 6C 38 00 38 6C      444                DB      038H,06CH,038H,000H,038H,06CH,0C6H,0C6H ; TH_8F
      C6 C6                  445
07DA  FE C6 C6 00 00 00      446                DB      0FEH,0C6H,0C6H,000H,000H,000H           ; BT_8F
                             447
07E0  18 30 60 00 FE 66      448                DB      018H,030H,060H,000H,0FEH,066H,060H,07CH ; TH_90
      60 7C                  449
07E8  60 66 FE 00 00 00      450                DB      060H,066H,0FEH,000H,000H,000H           ; BT_90
07EE  00 00 00 00 CC 76      451                DB      000H,000H,000H,000H,0CCH,076H,036H,07EH ; TH_91
      36 7E                  452
07F6  D8 D8 6E 00 00 00      453                DB      0D8H,0D8H,06EH,000H,000H,000H           ; BT_91
07FC  00 00 3E 6C CC CC      454                DB      000H,000H,03EH,06CH,0CCH,0CCH,0FEH,0CCH ; TH_92
      FE CC                  455
0804  CC CC CE 00 00 00      456                DB      0CCH,0CCH,0CEH,000H,000H,000H           ; BT_92
080A  00 10 38 6C 00 7C      457                DB      000H,010H,038H,06CH,000H,07CH,0C6H,0C6H ; TH_93
      C6 C6                  458
0812  C6 C6 7C 00 00 00      459                DB      0C6H,0C6H,07CH,000H,000H,000H           ; BT_93
0818  00 00 C6 C6 00 7C      460                DB      000H,000H,0C6H,0C6H,000H,07CH,0C6H,0C6H ; TH_94
      C6 C6                  461
0820  C6 C6 7C 00 00 00      462                DB      0C6H,0C6H,07CH,000H,000H,000H           ; BT_94
0826  00 60 30 18 00 7C      463                DB      000H,060H,030H,018H,000H,07CH,0C6H,0C6H ; TH_95
      C6 C6                  464
082E  C6 C6 7C 00 00 00      465                DB      0C6H,0C6H,07CH,000H,000H,000H           ; BT_95
0834  00 30 78 CC 00 CC      466                DB      000H,030H,078H,0CCH,000H,0CCH,0CCH,0CCH ; TH_96
      CC CC                  467
083C  CC CC 76 00 00 00      468                DB      0CCH,0CCH,076H,000H,000H,000H           ; BT_96
0842  00 60 30 18 00 CC      469                DB      000H,060H,030H,018H,000H,0CCH,0CCH,0CCH ; TH_97
      CC CC                  470
084A  CC CC 76 00 00 00      471                DB      0CCH,0CCH,076H,000H,000H,000H           ; BT_97
0850  00 00 C6 C6 00 C6      472                DB      000H,000H,0C6H,0C6H,000H,0C6H,0C6H,0C6H ; TH_98
      C6 C6                  473
0858  C6 7E 06 0C 78 00      474                DB      0C6H,07EH,006H,00CH,078H,000H           ; BT_98
085E  00 C6 C6 38 6C C6      475                DB      000H,0C6H,0C6H,038H,06CH,0C6H,0C6H,0C6H ; TH_99
      C6 C6                  476
0866  C6 6C 38 00 00 00      477                DB      0C6H,06CH,038H,000H,000H,000H           ; BT_99
086C  00 C6 C6 00 C6 C6      478                DB      000H,0C6H,0C6H,000H,0C6H,0C6H,0C6H,0C6H ; TH_9A
      C6 C6                  479
0874  C6 C6 7C 00 00 00      480                DB      0C6H,0C6H,07CH,000H,000H,000H           ; BT_9A
087A  00 18 18 3C 66 60      481                DB      000H,018H,018H,03CH,066H,060H,060H,066H ; TH_9B
      60 66                  482
0882  3C 18 18 00 00 00      483                DB      03CH,018H,018H,000H,000H,000H           ; BT_9B
0888  00 38 6C 64 60 F0      484                DB      000H,038H,06CH,064H,060H,0F0H,060H,060H ; TH_9C
      60 60                  485
0890  60 E6 FC 00 00 00      486                DB      060H,0E6H,0FCH,000H,000H,000H           ; BT_9C
0896  00 00 66 66 3C 18      487                DB      000H,000H,066H,066H,03CH,018H,07EH,018H ; TH_9D
      7E 18                  488
089E  7E 18 18 00 00 00      489                DB      07EH,018H,018H,000H,000H,000H           ; BT_9D
08A4  00 F8 CC CC F8 C4      490                DB      000H,0F8H,0CCH,0CCH,0F8H,0C4H,0CCH,0DEH ; TH_9E
      CC DE                  491
08AC  CC CC C6 00 00 00      492                DB      0CCH,0CCH,0C6H,000H,000H,000H           ; BT_9E
08B2  00 0E 1B 18 18 18      493                DB      000H,00EH,01BH,018H,018H,018H,07EH,018H ; TH_9F
      7E 18                  494
08BA  18 18 18 D8 70 00      495                DB      018H,018H,018H,0D8H,070H,000H           ; BT_9F
                             496
08C0  00 18 30 60 00 78      497                DB      000H,018H,030H,060H,000H,078H,00CH,07CH ; TH_A0
      0C 7C                  498
08C8  CC CC 76 00 00 00      499                DB      0CCH,0CCH,076H,000H,000H,000H           ; BT_A0
08CE  00 0C 18 30 00 38      500                DB      000H,00CH,018H,030H,000H,038H,018H,018H ; TH_A1
      18 18                  501
08D6  18 18 3C 00 00 00      502                DB      018H,018H,03CH,000H,000H,000H           ; BT_A1
08DC  00 18 30 60 00 7C      503                DB      000H,018H,030H,060H,000H,07CH,0C6H,0C6H ; TH_A2
      C6 C6                  504
08E4  C6 C6 7C 00 00 00      505                DB      0C6H,0C6H,07CH,000H,000H,000H           ; BT_A2
08EA  00 18 30 60 00 CC      506                DB      000H,018H,030H,060H,000H,0CCH,0CCH,0CCH ; TH_A3
      CC CC                  507
08F2  CC CC 76 00 00 00      508                DB      0CCH,0CCH,076H,000H,000H,000H           ; BT_A3
08F8  00 00 76 DC 00 DC      509                DB      000H,000H,076H,0DCH,000H,0DCH,066H,066H ; TH_A4
      66 66                  510
0900  66 66 66 00 00 00      511                DB      066H,066H,066H,000H,000H,000H           ; BT_A4
0906  76 DC 00 C6 E6 F6      512                DB      076H,0DCH,000H,0C6H,0E6H,0F6H,0FEH,0DEH ; TH_A5
      FE DE                  513
090E  0E C6 C6 00 00 00      514                DB      0CEH,0C6H,0C6H,000H,000H,000H           ; BT_A5
0914  00 3C 6C 6C 3E 00      515                DB      000H,03CH,06CH,06CH,03EH,000H,07EH,000H ; TH_A6
      7E 00                  516
091C  00 00 00 00 00 00      517                DB      000H,000H,000H,000H,000H,000H           ; BT_A6
0922  00 38 6C 6C 38 00      518                DB      000H,038H,06CH,06CH,038H,000H,07CH,000H ; TH_A7
      7C 00                  519
092A  00 00 00 00 00 00      520                DB      000H,000H,000H,000H,000H,000H           ; BT_A7
0930  00 00 30 30 00 30      521                DB      000H,000H,030H,030H,000H,030H,030H,060H ; TH_A8
      30 60                  522
0938  C6 C6 7C 00 00 00      523                DB      0C6H,0C6H,07CH,000H,000H,000H           ; BT_A8
093E  00 00 00 00 00 00      524                DB      000H,000H,000H,000H,000H,000H,0FEH,0C0H ; TH_A9
      FE C0                  525
0946  C0 C0 00 00 00 00      526                DB      0C0H,0C0H,000H,000H,000H,000H           ; BT_A9
094C  00 00 00 00 00 00      527                DB      000H,000H,000H,000H,000H,000H,0FEH,006H ; TH_AA
      FE 06                  528
0954  06 06 00 00 00 00      529                DB      006H,006H,000H,000H,000H,000H           ; BT_AA
095A  00 C0 C0 C6 CC D8      530                DB      000H,0C0H,0C0H,0C6H,0CCH,0D8H,030H,060H ; TH_AB
      30 60                  531
0962  DC 86 0C 18 3E 00      532                DB      0DCH,086H,00CH,018H,03EH,000H           ; BT_AB
0968  00 C0 C0 C6 CC D8      533                DB      000H,0C0H,0C0H,0C6H,0CCH,0D8H,030H,066H ; TH_AC
      30 66                  534
0970  CE 9E 3E 06 06 00      535                DB      0CEH,09EH,03EH,006H,006H,000H           ; BT_AC
0976  00 00 18 18 00 18      536                DB      000H,000H,018H,018H,000H,018H,018H,03CH ; TH_AD
      18 3C                  537
097E  3C 3C 18 00 00 00      538                DB      03CH,03CH,018H,000H,000H,000H           ; BT_AD
0984  00 00 00 00 36 6C      539                DB      000H,000H,000H,000H,036H,06CH,0D8H,06CH ; TH_AE
      D8 6C                  540
098C  36 00 00 00 00 00      541                DB      036H,000H,000H,000H,000H,000H           ; BT_AE
0992  00 00 00 00 D8 6C      542                DB      000H,000H,000H,000H,0D8H,06CH,036H,06CH ; TH_AF
      36 6C                  543
099A  D8 00 00 00 00 00      544                DB      0D8H,000H,000H,000H,000H,000H           ; BT_AF
                             545
09A0  11 44 11 44 11 44      546                DB      011H,044H,011H,044H,011H,044H,011H,044H ; TH_B0
      11 44                  547
09A8  11 44 11 44 11 44      548                DB      011H,044H,011H,044H,011H,044H           ; BT_B0
09AE  55 AA 55 AA 55 AA      549                DB      055H,0AAH,055H,0AAH,055H,0AAH,055H,0AAH ; TH_B1
      55 AA                  550
09B6  55 AA 55 AA 55 AA      551                DB      055H,0AAH,055H,0AAH,055H,0AAH           ; BT_B1
09BC  DD 77 DD 77 DD 77      552                DB      0DDH,077H,0DDH,077H,0DDH,077H,0DDH,077H ; TH_B2
      DD 77                  553
09C4  DD 77 DD 77 DD 77      554                DB      0DDH,077H,0DDH,077H,0DDH,077H           ; BT_B2
09CA  18 18 18 18 18 18      555                DB      018H,018H,018H,018H,018H,018H,018H,018H ; TH_B3
      18 18                  556
09D2  18 18 18 18 18 18      557                DB      018H,018H,018H,018H,018H,018H           ; BT_B3
09D8  18 18 18 18 18 18      558                DB      018H,018H,018H,018H,018H,018H,018H,0F8H ; TH_B4
      18 F8                  559
09E0  18 18 18 18 18 18      560                DB      018H,018H,018H,018H,018H,018H           ; BT_B4
09E6  18 18 18 18 18 F8      561                DB      018H,018H,018H,018H,018H,0F8H,018H,0F8H ; TH_B5
      18 F8                  562
09EE  18 18 18 18 18 18      563                DB      018H,018H,018H,018H,018H,018H           ; BT_B5
09F4  36 36 36 36 36 36      564                DB      036H,036H,036H,036H,036H,036H,036H,0F6H ; TH_B6
      36 F6                  565
09FC  36 36 36 36 36 36      566                DB      036H,036H,036H,036H,036H,036H           ; BT_B6
0A02  00 00 00 00 00 00      567                DB      000H,000H,000H,000H,000H,000H,000H,0FEH ; TH_B7
      00 FE                  568
0A0A  36 36 36 36 36 36      569                DB      036H,036H,036H,036H,036H,036H           ; BT_B7
```

```
0A10  00 00 00 00 00 F8      570                 DB      000H,000H,000H,000H,000H,0F8H,018H,0F8H ; TH_B8
      18 F8                  571
0A18  18 18 18 18 18 18      572                 DB      018H,018H,018H,018H,018H,018H           ; BT_B8
0A1E  36 36 36 36 36 F6      573                 DB      036H,036H,036H,036H,036H,0F6H,006H,0F6H ; TH_B9
      06 F6                  574
0A26  36 36 36 36 36 36      575                 DB      036H,036H,036H,036H,036H,036H           ; BT_B9
0A2C  36 36 36 36 36 36      576                 DB      036H,036H,036H,036H,036H,036H,036H,036H ; TH_BA
      36 36                  577
0A34  36 36 36 36 36 36      578                 DB      036H,036H,036H,036H,036H,036H           ; BT_BA
0A3A  00 00 00 00 00 FE      579                 DB      000H,000H,000H,000H,000H,0FEH,006H,0F6H ; TH_BB
      06 F6                  580
0A42  36 36 36 36 36 36      581                 DB      036H,036H,036H,036H,036H,036H           ; BT_BB
0A48  36 36 36 36 36 F6      582                 DB      036H,036H,036H,036H,036H,0F6H,006H,0FEH ; TH_BC
      06 FE                  583
0A50  00 00 00 00 00 00      584                 DB      000H,000H,000H,000H,000H,000H           ; BT_BC
0A56  36 36 36 36 36 36      585                 DB      036H,036H,036H,036H,036H,036H,036H,0FEH ; TH_BD
      36 FE                  586
0A5E  00 00 00 00 00 00      587                 DB      000H,000H,000H,000H,000H,000H           ; BT_BD
0A64  18 18 18 18 18 F8      588                 DB      018H,018H,018H,018H,018H,0F8H,018H,0F8H ; TH_BE
      18 F8                  589
0A6C  00 00 00 00 00 00      590                 DB      000H,000H,000H,000H,000H,000H           ; BT_BE
0A72  00 00 00 00 00 00      591                 DB      000H,000H,000H,000H,000H,000H,000H,0F8H ; TH_BF
      00 F8                  592
0A7A  18 18 18 18 18 18      593                 DB      018H,018H,018H,018H,018H,018H           ; BT_BF
                             594
0A80  18 18 18 18 18 18      595                 DB      018H,018H,018H,018H,018H,018H,018H,01FH ; TH_C0
      18 1F                  596
0A88  00 00 00 00 00 00      597                 DB      000H,000H,000H,000H,000H,000H           ; BT_C0
0A8E  18 18 18 18 18 18      598                 DB      018H,018H,018H,018H,018H,018H,018H,0FFH ; TH_C1
      18 FF                  599
0A96  00 00 00 00 00 00      600                 DB      000H,000H,000H,000H,000H,000H           ; BT_C1
0A9C  00 00 00 00 00 00      601                 DB      000H,000H,000H,000H,000H,000H,000H,0FFH ; TH_C2
      00 FF                  602
0AA4  18 18 18 18 18 18      603                 DB      018H,018H,018H,018H,018H,018H           ; BT_C2
0AAA  18 18 18 18 18 18      604                 DB      018H,018H,018H,018H,018H,018H,018H,01FH ; TH_C3
      18 1F                  605
0AB2  18 18 18 18 18 18      606                 DB      018H,018H,018H,018H,018H,018H           ; BT_C3
0AB8  00 00 00 00 00 00      607                 DB      000H,000H,000H,000H,000H,000H,000H,0FFH ; TH_C4
      00 FF                  608
0AC0  00 00 00 00 00 00      609                 DB      000H,000H,000H,000H,000H,000H           ; BT_C4
0AC6  18 18 18 18 18 18      610                 DB      018H,018H,018H,018H,018H,018H,018H,0FFH ; TH_C5
      18 FF                  611
0ACE  18 18 18 18 18 18      612                 DB      018H,018H,018H,018H,018H,018H           ; BT_C5
0AD4  18 18 18 18 18 1F      613                 DB      018H,018H,018H,018H,018H,01FH,018H,01FH ; TH_C6
      18 1F                  614
0ADC  18 18 18 18 18 18      615                 DB      018H,018H,018H,018H,018H,018H           ; BT_C6
0AE2  36 36 36 36 36 36      616                 DB      036H,036H,036H,036H,036H,036H,036H,037H ; TH_C7
      36 37                  617
0AEA  36 36 36 36 36 36      618                 DB      036H,036H,036H,036H,036H,036H           ; BT_C7
0AF0  36 36 36 36 36 37      619                 DB      036H,036H,036H,036H,036H,037H,030H,03FH ; TH_C8
      30 3F                  620
0AF8  00 00 00 00 00 00      621                 DB      000H,000H,000H,000H,000H,000H           ; BT_C8
0AFE  00 00 00 00 00 3F      622                 DB      000H,000H,000H,000H,000H,03FH,030H,037H ; TH_C9
      30 37                  623
0B06  36 36 36 36 36 36      624                 DB      036H,036H,036H,036H,036H,036H           ; BT_C9
0B0C  36 36 36 36 36 F7      625                 DB      036H,036H,036H,036H,036H,0F7H,000H,0FFH ; TH_CA
      00 FF                  626
0B14  00 00 00 00 00 00      627                 DB      000H,000H,000H,000H,000H,000H           ; BT_CA
0B1A  00 00 00 00 00 FF      628                 DB      000H,000H,000H,000H,000H,0FFH,000H,0F7H ; TH_CB
      00 F7                  629
0B22  36 36 36 36 36 36      630                 DB      036H,036H,036H,036H,036H,036H           ; BT_CB
0B28  36 36 36 36 36 37      631                 DB      036H,036H,036H,036H,036H,037H,030H,037H ; TH_CC
      30 37                  632
0B30  36 36 36 36 36 36      633                 DB      036H,036H,036H,036H,036H,036H           ; BT_CC
0B36  00 00 00 00 00 FF      634                 DB      000H,000H,000H,000H,000H,0FFH,000H,0FFH ; TH_CD
      00 FF                  635
0B3E  00 00 00 00 00 00      636                 DB      000H,000H,000H,000H,000H,000H           ; BT_CD
0B44  36 36 36 36 36 F7      637                 DB      036H,036H,036H,036H,036H,0F7H,000H,0F7H ; TH_CE
      00 F7                  638
0B4C  36 36 36 36 36 36      639                 DB      036H,036H,036H,036H,036H,036H           ; BT_CE
0B52  18 18 18 18 18 FF      640                 DB      018H,018H,018H,018H,018H,0FFH,000H,0FFH ; TH_CF
      00 FF                  641
0B5A  00 00 00 00 00 00      642                 DB      000H,000H,000H,000H,000H,000H           ; BT_CF
                             643
0B60  36 36 36 36 36 36      644                 DB      036H,036H,036H,036H,036H,036H,036H,0FFH ; TH_D0
      36 FF                  645
0B68  00 00 00 00 00 00      646                 DB      000H,000H,000H,000H,000H,000H           ; BT_D0
0B6E  00 00 00 00 00 FF      647                 DB      000H,000H,000H,000H,000H,0FFH,000H,0FFH ; TH_D1
      00 FF                  648
0B76  18 18 18 18 18 18      649                 DB      018H,018H,018H,018H,018H,018H           ; BT_D1
0B7C  00 00 00 00 00 00      650                 DB      000H,000H,000H,000H,000H,000H,000H,0FFH ; TH_D2
      00 FF                  651
0B84  36 36 36 36 36 36      652                 DB      036H,036H,036H,036H,036H,036H           ; BT_D2
0B8A  36 36 36 36 36 36      653                 DB      036H,036H,036H,036H,036H,036H,036H,03FH ; TH_D3
      36 3F                  654
0B92  00 00 00 00 00 00      655                 DB      000H,000H,000H,000H,000H,000H           ; BT_D3
0B98  18 18 18 18 18 1F      656                 DB      018H,018H,018H,018H,018H,01FH,018H,01FH ; TH_D4
      18 1F                  657
0BA0  00 00 00 00 00 00      658                 DB      000H,000H,000H,000H,000H,000H           ; BT_D4
0BA6  00 00 00 00 00 1F      659                 DB      000H,000H,000H,000H,000H,01FH,018H,01FH ; TH_D5
      18 1F                  660
0BAE  18 18 18 18 18 18      661                 DB      018H,018H,018H,018H,018H,018H           ; BT_D5
0BB4  00 00 00 00 00 00      662                 DB      000H,000H,000H,000H,000H,000H,000H,03FH ; TH_D6
      00 3F                  663
0BBC  36 36 36 36 36 36      664                 DB      036H,036H,036H,036H,036H,036H           ; BT_D6
0BC2  36 36 36 36 36 36      665                 DB      036H,036H,036H,036H,036H,036H,036H,0FFH ; TH_D7
      36 FF                  666
0BCA  36 36 36 36 36 36      667                 DB      036H,036H,036H,036H,036H,036H           ; BT_D7
0BD0  18 18 18 18 18 FF      668                 DB      018H,018H,018H,018H,018H,0FFH,018H,0FFH ; TH_D8
      18 FF                  669
0BD8  18 18 18 18 18 18      670                 DB      018H,018H,018H,018H,018H,018H           ; BT_D8
0BDE  18 18 18 18 18 18      671                 DB      018H,018H,018H,018H,018H,018H,018H,0F8H ; TH_D9
      18 F8                  672
0BE6  00 00 00 00 00 00      673                 DB      000H,000H,000H,000H,000H,000H           ; BT_D9
0BEC  00 00 00 00 00 00      674                 DB      000H,000H,000H,000H,000H,000H,000H,01FH ; TH_DA
      00 1F                  675
0BF4  18 18 18 18 18 18      676                 DB      018H,018H,018H,018H,018H,018H           ; BT_DA
0BFA  FF FF FF FF FF FF      677                 DB      0FFH,0FFH,0FFH,0FFH,0FFH,0FFH,0FFH,0FFH ; TH_DB
      FF FF                  678
0C02  FF FF FF FF FF FF      679                 DB      0FFH,0FFH,0FFH,0FFH,0FFH,0FFH           ; BT_DB
0C08  00 00 00 00 00 00      680                 DB      000H,000H,000H,000H,000H,000H,000H,0FFH ; TH_DC
      00 FF                  681
0C10  FF FF FF FF FF FF      682                 DB      0FFH,0FFH,0FFH,0FFH,0FFH,0FFH           ; BT_DC
0C16  F0 F0 F0 F0 F0 F0      683                 DB      0F0H,0F0H,0F0H,0F0H,0F0H,0F0H,0F0H,0F0H ; TH_DD
      F0 F0                  684
0C1E  F0 F0 F0 F0 F0 F0      685                 DB      0F0H,0F0H,0F0H,0F0H,0F0H,0F0H           ; BT_DD
0C24  0F 0F 0F 0F 0F 0F      686                 DB      00FH,00FH,00FH,00FH,00FH,00FH,00FH,00FH ; TH_DE
      0F 0F                  687
0C2C  0F 0F 0F 0F 0F 0F      688                 DB      00FH,00FH,00FH,00FH,00FH,00FH           ; BT_DE
0C32  FF FF FF FF FF FF      689                 DB      0FFH,0FFH,0FFH,0FFH,0FFH,0FFH,0FFH,000H ; TH_DF
      FF 00                  690
0C3A  00 00 00 00 00 00      691                 DB      000H,000H,000H,000H,000H,000H           ; BT_DF
                             692
0C40  00 00 00 00 00 76      693                 DB      000H,000H,000H,000H,000H,076H,0DCH,0D8H ; TH_E0
      DC D8                  694
0C48  D8 DC 76 00 00 00      695                 DB      0D8H,0DCH,076H,000H,000H,000H           ; BT_E0
```

```
0C4E  00 00 00 00 7C C6      696                DB      000H,000H,000H,000H,07CH,0C6H,0FCH,0C6H ; TH_E1
      FC C6                  697
0C56  C6 FC C0 C0 40 00      698                DB      0C6H,0FCH,0C0H,0C0H,040H,000H           ; BT_E1
0C5C  00 00 FE C6 C6 C0      699                DB      000H,000H,0FEH,0C6H,0C6H,0C0H,0C0H,0C0H ; TH_E2
      C0 C0                  700
0C64  C0 C0 C0 00 00 00      701                DB      0C0H,0C0H,0C0H,000H,000H,000H           ; BT_E2
0C6A  00 00 00 00 FE 6C      702                DB      000H,000H,000H,000H,0FEH,06CH,06CH,06CH ; TH_E3
      6C 6C                  703
0C72  6C 6C 6C 00 00 00      704                DB      06CH,06CH,06CH,000H,000H,000H           ; BT_E3
0C78  00 00 FE C6 60 30      705                DB      000H,000H,0FEH,0C6H,060H,030H,018H,030H ; TH_E4
      18 30                  706
0C80  60 C6 FE 00 00 00      707                DB      060H,0C6H,0FEH,000H,000H,000H           ; BT_E4
0C86  00 00 00 00 00 7E      708                DB      000H,000H,000H,000H,000H,07EH,0D8H,0D8H ; TH_E5
      D8 D8                  709
0C8E  D8 D8 70 00 00 00      710                DB      0D8H,0D8H,070H,000H,000H,000H           ; BT_E5
0C94  00 00 00 00 66 66      711                DB      000H,000H,000H,000H,066H,066H,066H,066H ; TH_E6
      66 66                  712
0C9C  7C 60 60 C0 00 00      713                DB      07CH,060H,060H,0C0H,000H,000H           ; BT_E6
0CA2  00 00 00 00 76 DC      714                DB      000H,000H,000H,000H,076H,0DCH,018H,018H ; TH_E7
      18 18                  715
0CAA  18 18 18 00 00 00      716                DB      018H,018H,018H,000H,000H,000H           ; BT_E7
0CB0  00 00 7E 18 3C 66      717                DB      000H,000H,07EH,018H,03CH,066H,066H,066H ; TH_E8
      66 66                  718
0CB8  3C 18 7E 00 00 00      719                DB      03CH,018H,07EH,000H,000H,000H           ; BT_E8
0CBE  00 00 38 6C C6 C6      720                DB      000H,000H,038H,06CH,0C6H,0C6H,0FEH,0C6H ; TH_E9
      FE C6                  721
0CC6  C6 6C 38 00 00 00      722                DB      0C6H,06CH,038H,000H,000H,000H           ; BT_E9
0CCC  00 00 38 6C C6 C6      723                DB      000H,000H,038H,06CH,0C6H,0C6H,0C6H,06CH ; TH_EA
      C6 6C                  724
0CD4  6C 6C EE 00 00 00      725                DB      06CH,06CH,0EEH,000H,000H,000H           ; BT_EA
0CDA  00 00 1E 30 18 0C      726                DB      000H,000H,01EH,030H,018H,00CH,03EH,066H ; TH_EB
      3E 66                  727
0CE2  66 66 3C 00 00 00      728                DB      066H,066H,03CH,000H,000H,000H           ; BT_EB
0CE8  00 00 00 00 00 7E      729                DB      000H,000H,000H,000H,000H,07EH,0DBH,0DBH ; TH_EC
      DB DB                  730
0CF0  7E 00 00 00 00 00      731                DB      07EH,000H,000H,000H,000H,000H           ; BT_EC
0CF6  00 00 03 06 7E DB      732                DB      000H,000H,003H,006H,07EH,0DBH,0DBH,0F3H ; TH_ED
      DB F3                  733
0CFE  7E 60 C0 00 00 00      734                DB      07EH,060H,0C0H,000H,000H,000H           ; BT_ED
0D04  00 00 1C 30 60 60      735                DB      000H,000H,01CH,030H,060H,060H,07CH,060H ; TH_EE
      7C 60                  736
0D0C  60 30 1C 00 00 00      737                DB      060H,030H,01CH,000H,000H,000H           ; BT_EE
0D12  00 00 00 7C C6 C6      738                DB      000H,000H,000H,07CH,0C6H,0C6H,0C6H,0C6H ; TH_EF
      C6 C6                  739
0D1A  C6 C6 C6 00 00 00      740                DB      0C6H,0C6H,0C6H,000H,000H,000H           ; BT_EF
                             741
0D20  00 00 00 FE 00 00      742                DB      000H,000H,000H,0FEH,000H,000H,0FEH,000H ; TH_F0
      FE 00                  743
0D28  00 FE 00 00 00 00      744                DB      000H,0FEH,000H,000H,000H,000H           ; BT_F0
0D2E  00 00 00 18 18 7E      745                DB      000H,000H,000H,018H,018H,07EH,018H,018H ; TH_F1
      18 18                  746
0D36  00 00 FF 00 00 00      747                DB      000H,000H,0FFH,000H,000H,000H           ; BT_F1
0D3C  00 00 30 18 0C 06      748                DB      000H,000H,030H,018H,00CH,006H,00CH,018H ; TH_F2
      0C 18                  749
0D44  30 00 7E 00 00 00      750                DB      030H,000H,07EH,000H,000H,000H           ; BT_F2
0D4A  00 00 0C 18 30 60      751                DB      000H,000H,00CH,018H,030H,060H,030H,018H ; TH_F3
      30 18                  752
0D52  0C 00 7E 00 00 00      753                DB      00CH,000H,07EH,000H,000H,000H           ; BT_F3
0D58  00 00 0E 1B 1B 18      754                DB      000H,000H,00EH,01BH,01BH,018H,018H,018H ; TH_F4
      18 18                  755
0D60  18 18 18 18 18 18      756                DB      018H,018H,018H,018H,018H,018H           ; BT_F4
0D66  18 18 18 18 18 18      757                DB      018H,018H,018H,018H,018H,018H,018H,018H ; TH_F5
      18 18                  758
0D6E  D8 D8 70 00 00 00      759                DB      0D8H,0D8H,070H,000H,000H,000H           ; BT_F5
0D74  00 00 00 18 18 00      760                DB      000H,000H,000H,018H,018H,000H,07EH,000H ; TH_F6
      7E 00                  761
0D7C  18 18 00 00 00 00      762                DB      018H,018H,000H,000H,000H,000H           ; BT_F6
0D82  00 00 00 00 76 DC      763                DB      000H,000H,000H,000H,076H,0DCH,000H,076H ; TH_F7
      00 76                  764
0D8A  DC 00 00 00 00 00      765                DB      0DCH,000H,000H,000H,000H,000H           ; BT_F7
0D90  00 38 6C 6C 38 00      766                DB      000H,038H,06CH,06CH,038H,000H,000H,000H ; TH_F8
      00 00                  767
0D98  00 00 00 00 00 00      768                DB      000H,000H,000H,000H,000H,000H           ; BT_F8
0D9E  00 00 00 00 00 00      769                DB      000H,000H,000H,000H,000H,000H,018H,018H ; TH_F9
      18 18                  770
0DA6  00 00 00 00 00 00      771                DB      000H,000H,000H,000H,000H,000H           ; BT_F9
0DAC  00 00 00 00 00 00      772                DB      000H,000H,000H,000H,000H,000H,000H,018H ; TH_FA
      00 18                  773
0DB4  00 00 00 00 00 00      774                DB      000H,000H,000H,000H,000H,000H           ; BT_FA
0DBA  00 0F 0C 0C 0C 0C      775                DB      000H,00FH,00CH,00CH,00CH,00CH,00CH,0ECH ; TH_FB
      EC                     776
0DC2  6C 3C 1C 00 00 00      777                DB      06CH,03CH,01CH,000H,000H,000H           ; BT_FB
0DC8  00 D8 6C 6C 6C 6C      778                DB      000H,0D8H,06CH,06CH,06CH,06CH,06CH,000H ; TH_FC
      6C 00                  779
0DD0  00 00 00 00 00 00      780                DB      000H,000H,000H,000H,000H,000H           ; BT_FC
0DD6  00 70 D8 30 60 C8      781                DB      000H,070H,0D8H,030H,060H,0C8H,0F8H,000H ; TH_FD
      F8 00                  782
0DDE  00 00 00 00 00 00      783                DB      000H,000H,000H,000H,000H,000H           ; BT_FD
0DE4  00 00 00 00 7C 7C      784                DB      000H,000H,000H,000H,07CH,07CH,07CH,07CH ; TH_FE
      7C 7C                  785
0DEC  7C 7C 00 00 00 00      786                DB      07CH,07CH,000H,000H,000H,000H           ; BT_FE
0DF2  00 00 00 00 00 00      787                DB      000H,000H,000H,000H,000H,000H,000H,000H ; TH_FF
      00 00                  788
0DFA  00 00 00 00 00 00      789                DB      000H,000H,000H,000H,000H,000H           ; BT_FF
0E00                         790    CODE        ENDS
                             791                END


                               1                PAGE,120
                               2                SUBTTL  MONOCHROME CHARACTER GENERATOR  -  ALPHA SUPPLEMENT
0000                           3    CODE        SEGMENT PUBLIC
                               4                PUBLIC  CGMN_FDG
0000                           5    CGMN_FDG    LABEL   BYTE
                               6
                               7    ; STRUCTURE OF THIS FILE
                               8    ;           DB      XXH     WHERE XX IS THE HEX CODE FOR THE FOLLOWING CHAR
                               9    ;           DB      [BYTES 0 - 13 OF THAT CHARACTER]
                              10    ;           ...
                              11    ;           DB      00H     INDICATES NO MORE REPLACEMENTS TO BE DONE
                              12
                              13
0000  1D                     14                DB      01DH                                    ;
0001  00 00 00 00 24 66      15                DB      000H,000H,000H,000H,024H,066H,0FFH,066H ; TH_1D
      FF 66                  16
0009  24 00 00 00 00 00      17                DB      024H,000H,000H,000H,000H,000H           ; BT_1D
000F  22                     18                DB      022H                                    ;
0010  00 63 63 63 22 00      19                DB      000H,063H,063H,063H,022H,000H,000H,000H ; TH_22 "
      00 00                  20
0018  00 00 00 00 00 00      21                DB      000H,000H,000H,000H,000H,000H           ; BT_22 "
001E  2B                     22                DB      02BH                                    ;
001F  00 00 00 18 18 18      23                DB      000H,000H,000H,018H,018H,018H,0FFH,018H ; TH_2B +
      FF 18                  24
0027  18 18 00 00 00 00      25                DB      018H,018H,000H,000H,000H,000H           ; BT_2B +
002D  2D                     26                DB      02DH                                    ;
002E  00 00 00 00 00 00      27                DB      000H,000H,000H,000H,000H,000H,0FFH,000H ; TH_2D -
      FF 00                  28
```

```
0036 00 00 00 00 00 00                  29              DB      000H,000H,000H,000H,000H,000H           ; BT_2D -
003C 4D                                 30              DB      04DH                                    ;
003D 00 00 C3 E7 FF DB                  31              DB      000H,000H,0C3H,0E7H,0FFH,0DBH,0C3H,0C3H ; TH_4D M
     C3 C3                              32
0045 C3 C3 C3 00 00 00 00               33              DB      0C3H,0C3H,0C3H,000H,000H,000H           ; BT_4D M
004B 54                                 34              DB      054H                                    ;
004C 00 00 FF DB 99 18                  35              DB      000H,000H,0FFH,0DBH,099H,018H,018H,018H ; TH_54 T
     18 18                              36
0054 18 18 3C 00 00 00 00               37              DB      018H,018H,03CH,000H,000H,000H           ; BT_54 T
005A 56                                 38              DB      056H                                    ;
005B 00 00 C3 C3 C3 C3                  39              DB      000H,000H,0C3H,0C3H,0C3H,0C3H,0C3H,0C3H ; TH_56 V
     C3 C3                              40
0063 66 3C 18 00 00 00 00               41              DB      066H,03CH,018H,000H,000H,000H           ; BT_56 V
0069 57                                 42              DB      057H                                    ;
006A 00 00 C3 C3 C3 C3                  43              DB      000H,000H,0C3H,0C3H,0C3H,0C3H,0DBH,0DBH ; TH_57 W
     DB DB                              44
0072 FF 66 66 00 00 00 00               45              DB      0FFH,066H,066H,000H,000H,000H           ; BT_57 W
0078 58'                                46              DB      058H                                    ;
0079 00 00 C3 C3 66 3C                  47              DB      000H,000H,0C3H,0C3H,066H,03CH,018H,03CH ; TH_58 X
     18 3C                              48
0081 66 C3 C3 00 00 00 00               49              DB      066H,0C3H,0C3H,000H,000H,000H           ; BT_58 X
0087 59                                 50              DB      059H                                    ;
0088 00 00 C3 C3 C3 66                  51              DB      000H,000H,0C3H,0C3H,0C3H,066H,03CH,018H ; TH_59 Y
     3C 18                              52
0090 18 18 3C 00 00 00 00               53              DB      018H,018H,03CH,000H,000H,000H           ; BT_59 Y
0096 5A                                 54              DB      05AH                                    ;
0097 00 00 FF C3 86 0C                  55              DB      000H,000H,0FFH,0C3H,086H,00CH,018H,03OH ; TH_5A Z
     18 30                              56
009F 61 C3 FF 00 00 00 00               57              DB      061H,0C3H,0FFH,000H,000H,000H           ; BT_5A Z
00A5 6D                                 58              DB      06DH                                    ;
00A6 00 00 00 00 00 E6                  59              DB      000H,000H,000H,000H,000H,0E6H,0FFH,0DBH ; TH_6D L.C. M
     FF DB                              60
00AE DB DB DB 00 00 00 00               61              DB      0DBH,0DBH,0DBH,000H,000H,000H           ; BT_6D L.C. M
00B4 76                                 62              DB      076H                                    ;
00B5 00 00 00 00 00 C3                  63              DB      000H,000H,000H,000H,000H,0C3H,0C3H,0C3H ; TH_76 L.C. V
     C3 C3                              64
00BD 66 3C 18 00 00 00 00               65              DB      066H,03CH,018H,000H,000H,000H           ; BT_76 L.C. V
00C3 77                                 66              DB      077H                                    ;
00C4 00 00 00 00 00 C3                  67              DB      000H,000H,000H,000H,000H,0C3H,0C3H,0DBH ; TH_77 L.C. W
     C3 DB                              68
00CC DB FF 66 00 00 00 00               69              DB      0DBH,0FFH,066H,000H,000H,000H           ; BT_77 L.C. W
00D2 91                                 70              DB      091H                                    ;
00D3 00 00 00 00 6E 3B                  71              DB      000H,000H,000H,000H,06EH,03BH,01BH,07EH ; TH_91
     1B 7E                              72
00DB DB DC 77 00 00 00                  73              DB      0DBH,0DCH,077H,000H,000H,000H           ; BT_91
00E1 9B                                 74              DB      09BH                                    ;
00E2 00 18 18 7E C3 C0                  75              DB      000H,018H,018H,07EH,0C3H,0C0H,0C0H,0C3H ; TH_9B
     C0 C3                              76
00EA 7E 18 18 00 00 00 00               77              DB      07EH,018H,018H,000H,000H,000H           ; BT_9B
00F0 9D                                 78              DB      09DH                                    ;
00F1 00 00 C3 66 3C 18                  79              DB      000H,000H,0C3H,066H,03CH,018H,0FFH,018H ; TH_9D
     FF 18                              80
00F9 FF 18 18 00 00 00 00               81              DB      0FFH,018H,018H,000H,000H,000H           ; BT_9D
00FF 9E                                 82              DB      09EH                                    ;
0100 00 FC 66 66 7C 62                  83              DB      000H,0FCH,066H,066H,07CH,062H,066H,06FH ; TH_9E
     66 6F                              84
0108 66 66 F3 00 00 00 00               85              DB      066H,066H,0F3H,000H,000H,000H           ; BT_9E
010E F1                                 86              DB      0F1H                                    ;
010F 00 00 18 18 18 FF                  87              DB      000H,000H,018H,018H,018H,0FFH,018H,018H ; TH_F1
     18 18                              88
0117 18 00 FF 00 00 00 00               89              DB      018H,000H,0FFH,000H,000H,000H           ; BT_F1
011D F6                                 90              DB      0F6H                                    ;
011E 00 00 18 18 00 00                  91              DB      000H,000H,018H,018H,000H,000H,0FFH,000H ; TH_F6
     FF 00                              92
0126 00 18 18 00 00 00 00               93              DB      000H,018H,018H,000H,000H,000H           ; BT_F6
012C 00                                 94              DB      000H                                    ; NO MORE
012D                                    95      CODE    ENDS
                                        96              END
```

```
                                        1               PAGE,120
                                        2               SUBTTL  DOUBLE DOT CHARACTER GENERATOR
0000                                    3       CODE    SEGMENT PUBLIC
                                        4               PUBLIC  CGDDOT,INT_1F_1
0000                                    5       CGDDOT  LABEL   BYTE
                                        6                                                               ; DOUBLE DOT
0000 00 00 00 00 00 00 00               7               DB      000H,000H,000H,000H,000H,000H,000H,000H ; D_00
     00 00                              8
0008 7E 81 A5 81 BD 99                  9               DB      07EH,081H,0A5H,081H,0BDH,099H,081H,07EH ; D_01
     81 7E                             10
0010 7E FF DB FF C3 E7                 11               DB      07EH,0FFH,0DBH,0FFH,0C3H,0E7H,0FFH,07EH ; D_02
     FF 7E                             12
0018 6C FE FE FE 7C 38                 13               DB      06CH,0FEH,0FEH,0FEH,07CH,038H,010H,000H ; D_03
     10 00                             14
0020 10 38 7C FE 7C 38                 15               DB      010H,038H,07CH,0FEH,07CH,038H,010H,000H ; D_04
     10 00                             16
0028 38 7C 38 FE FE 7C                 17               DB      038H,07CH,038H,0FEH,0FEH,07CH,038H,07CH ; D_05
     38 7C                             18
0030 10 10 38 7C FE 7C                 19               DB      010H,010H,038H,07CH,0FEH,07CH,038H,07CH ; D_06
     38 7C                             20
0038 00 00 18 3C 3C 18                 21               DB      000H,000H,018H,03CH,03CH,018H,000H,000H ; D_07
     00 00                             22
0040 FF FF E7 C3 C3 E7                 23               DB      0FFH,0FFH,0E7H,0C3H,0C3H,0E7H,0FFH,0FFH ; D_08
     FF FF                             24
0048 00 3C 66 42 42 66                 25               DB      000H,03CH,066H,042H,042H,066H,03CH,000H ; D_09
     3C 00                             26
0050 FF C3 99 BD BD 99                 27               DB      0FFH,0C3H,099H,0BDH,0BDH,099H,0C3H,0FFH ; D_0A
     C3 FF                             28
0058 0F 07 0F 7D CC CC                 29               DB      00FH,007H,00FH,07DH,0CCH,0CCH,0CCH,078H ; D_0B
     CC 78                             30
0060 3C 66 66 66 3C 18                 31               DB      03CH,066H,066H,066H,03CH,018H,07EH,018H ; D_0C
     7E 18                             32
0068 3F 33 3F 30 30 70                 33               DB      03FH,033H,03FH,030H,030H,070H,0F0H,0E0H ; D_0D
     F0 E0                             34
0070 7F 63 7F 63 63 67                 35               DB      07FH,063H,07FH,063H,063H,067H,0E6H,0C0H ; D_0E
     E6 C0                             36
0078 99 5A 3C E7 E7 3C                 37               DB      099H,05AH,03CH,0E7H,0E7H,03CH,05AH,099H ; D_0F
     5A 99                             38
                                       39
0080 80 E0 F8 FE F8 E0                 40               DB      080H,0E0H,0F8H,0FEH,0F8H,0E0H,080H,000H ; D_10
     80 00                             41
0088 02 0E 3E FE 3E 0E                 42               DB      002H,00EH,03EH,0FEH,03EH,00EH,002H,000H ; D_11
     02 00                             43
0090 18 3C 7E 18 18 7E                 44               DB      018H,03CH,07EH,018H,018H,07EH,03CH,018H ; D_12
     3C 18                             45
0098 66 66 66 66 66 00                 46               DB      066H,066H,066H,066H,066H,000H,066H,000H ; D_13
     66 00                             47
00A0 7F DB DB 7B 1B 1B                 48               DB      07FH,0DBH,0DBH,07BH,01BH,01BH,01BH,000H ; D_14
     1B 00                             49
00A8 3E 63 38 6C 6C 38                 50               DB      03EH,063H,038H,06CH,06CH,038H,0CCH,078H ; D_15
     CC 78                             51
00B0 00 00 00 00 7E 7E                 52               DB      000H,000H,000H,000H,07EH,07EH,07EH,000H ; D_16
     7E 00                             53
00B8 18 3C 7E 18 7E 3C                 54               DB      018H,03CH,07EH,018H,07EH,03CH,018H,0FFH ; D_17
     18 FF                             55
00C0 18 3C 7E 18 18 18                 56               DB      018H,03CH,07EH,018H,018H,018H,018H,000H ; D_18
```

```
                18 00                                   57
00C8  18 18 18 18 18 7E 3C              58      DB      018H,018H,018H,018H,07EH,03CH,018H,000H ; D_19
                18 00                                   59
00D0  00 18 0C FE 0C 18                 60      DB      000H,018H,00CH,0FEH,00CH,018H,000H,000H ; D_1A
                00 00                                   61
00D8  00 30 60 FE 60 30                 62      DB      000H,030H,060H,0FEH,060H,030H,000H,000H ; D_1B
                00 00                                   63
00E0  00 00 C0 C0 C0 FE                 64      DB      000H,000H,0C0H,0C0H,0C0H,0FEH,000H,000H ; D_1C
                00 00                                   65
00E8  00 24 66 FF 66 24                 66      DB      000H,024H,066H,0FFH,066H,024H,000H,000H ; D_1D
                00 00                                   67
00F0  00 18 3C 7E FF FF                 68      DB      000H,018H,03CH,07EH,0FFH,0FFH,000H,000H ; D_1E
                00 00                                   69
00F8  00 FF FF 7E 3C 18                 70      DB      000H,0FFH,0FFH,07EH,03CH,018H,000H,000H ; D_1F
                00 00                                   71
                                                        72
0100  00 00 00 00 00 00                 73      DB      000H,000H,000H,000H,000H,000H,000H,000H ; SP D_20
                00 00                                   74
0108  30 78 78 30 30 00                 75      DB      030H,078H,078H,030H,030H,000H,030H,000H ; ! D_21
                30 00                                   76
0110  6C 6C 6C 00 00 00                 77      DB      06CH,06CH,06CH,000H,000H,000H,000H,000H ; " D_22
                00 00                                   78
0118  6C 6C FE 6C FE 6C                 79      DB      06CH,06CH,0FEH,06CH,0FEH,06CH,06CH,000H ; # D_23
                6C 00                                   80
0120  30 7C C0 78 0C F8                 81      DB      030H,07CH,0C0H,078H,00CH,0F8H,030H,000H ; $ D_24
                30 00                                   82
0128  00 C6 CC 18 30 66                 83      DB      000H,0C6H,0CCH,018H,030H,066H,0C6H,000H ; PER CENT D_25
                C6 00                                   84
0130  38 6C 38 76 DC CC                 85      DB      038H,06CH,038H,076H,0DCH,0CCH,076H,000H ; & D_26
                76 00                                   86
0138  60 60 C0 00 00 00                 87      DB      060H,060H,0C0H,000H,000H,000H,000H,000H ; ' D_27
                00 00                                   88
0140  18 30 60 60 60 30                 89      DB      018H,030H,060H,060H,060H,030H,018H,000H ; ( D_28
                18 00                                   90
0148  60 30 18 18 18 30                 91      DB      060H,030H,018H,018H,018H,030H,060H,000H ; ) D_29
                60 00                                   92
0150  00 66 3C FF 3C 66                 93      DB      000H,066H,03CH,0FFH,03CH,066H,000H,000H ; * D_2A
                00 00                                   94
0158  00 30 30 FC 30 30                 95      DB      000H,030H,030H,0FCH,030H,030H,000H,000H ; + D_2B
                00 00                                   96
0160  00 00 00 00 00 30                 97      DB      000H,000H,000H,000H,000H,030H,030H,060H ; , D_2C
                30 60                                   98
0168  00 00 00 FC 00 00                 99      DB      000H,000H,000H,0FCH,000H,000H,000H,000H ; - D_2D
                00 00                                   100
0170  00 00 00 00 00 30                 101     DB      000H,000H,000H,000H,000H,030H,030H,000H ; . D_2E
                30 00                                   102
0178  06 0C 18 30 60 C0                 103     DB      006H,00CH,018H,030H,060H,0C0H,080H,000H ; / D_2F
                80 00                                   104
                                                        105
0180  7C C6 CE DE F6 E6                 106     DB      07CH,0C6H,0CEH,0DEH,0F6H,0E6H,07CH,000H ; 0 D_30
                7C 00                                   107
0188  30 70 30 30 30 30                 108     DB      030H,070H,030H,030H,030H,030H,0FCH,000H ; 1 D_31
                FC 00                                   109
0190  78 CC 0C 38 60 CC                 110     DB      078H,0CCH,00CH,038H,060H,0CCH,0FCH,000H ; 2 D_32
                FC 00                                   111
0198  78 CC 0C 38 0C CC                 112     DB      078H,0CCH,00CH,038H,00CH,0CCH,078H,000H ; 3 D_33
                78 00                                   113
01A0  1C 3C 6C CC FE 0C                 114     DB      01CH,03CH,06CH,0CCH,0FEH,00CH,01EH,000H ; 4 D_34
                1E 00                                   115
01A8  FC C0 F8 0C 0C CC                 116     DB      0FCH,0C0H,0F8H,00CH,00CH,0CCH,078H,000H ; 5 D_35
                78 00                                   117
01B0  38 60 C0 F8 CC CC                 118     DB      038H,060H,0C0H,0F8H,0CCH,0CCH,078H,000H ; 6 D_36
                78 00                                   119
01B8  FC CC 0C 18 30 30                 120     DB      0FCH,0CCH,00CH,018H,030H,030H,030H,000H ; 7 D_37
                30 00                                   121
01C0  78 CC CC 78 CC CC                 122     DB      078H,0CCH,0CCH,078H,0CCH,0CCH,078H,000H ; 8 D_38
                78 00                                   123
01C8  78 CC CC 7C 0C 18                 124     DB      078H,0CCH,0CCH,07CH,00CH,018H,070H,000H ; 9 D_39
                70 00                                   125
01D0  00 30 30 00 00 30                 126     DB      000H,030H,030H,000H,000H,030H,030H,000H ; : D_3A
                30 00                                   127
01D8  00 30 30 00 00 30                 128     DB      000H,030H,030H,000H,000H,030H,030H,060H ; ; D_3B
                30 60                                   129
01E0  18 30 60 C0 60 30                 130     DB      018H,030H,060H,0C0H,060H,030H,018H,000H ; < D_3C
                18 00                                   131
01E8  00 00 FC 00 00 FC                 132     DB      000H,000H,0FCH,000H,000H,0FCH,000H,000H ; = D_3D
                00 00                                   133
01F0  60 30 18 0C 18 30                 134     DB      060H,030H,018H,00CH,018H,030H,060H,000H ; > D_3E
                60 00                                   135
01F8  78 CC 0C 18 30 00                 136     DB      078H,0CCH,00CH,018H,030H,000H,030H,000H ; ? D_3F
                30 00                                   137
                                                        138
0200  7C C6 DE DE DE C0                 139     DB      07CH,0C6H,0DEH,0DEH,0DEH,0C0H,078H,000H ; @ D_40
                78 00                                   140
0208  30 78 CC CC FC CC                 141     DB      030H,078H,0CCH,0CCH,0FCH,0CCH,0CCH,000H ; A D_41
                CC 00                                   142
0210  FC 66 66 7C 66 66                 143     DB      0FCH,066H,066H,07CH,066H,066H,0FCH,000H ; B D_42
                FC 00                                   144
0218  3C 66 C0 C0 C0 66                 145     DB      03CH,066H,0C0H,0C0H,0C0H,066H,03CH,000H ; C D_43
                3C 00                                   146
0220  F8 6C 66 66 66 6C                 147     DB      0F8H,06CH,066H,066H,066H,06CH,0F8H,000H ; D D_44
                F8 00                                   148
0228  FE 62 68 78 68 62                 149     DB      0FEH,062H,068H,078H,068H,062H,0FEH,000H ; E D_45
                FE 00                                   150
0230  FE 62 68 78 68 60                 151     DB      0FEH,062H,068H,078H,068H,060H,0F0H,000H ; F D_46
                F0 00                                   152
0238  3C 66 C0 C0 CE 66                 153     DB      03CH,066H,0C0H,0C0H,0CEH,066H,03EH,000H ; G D_47
                3E 00                                   154
0240  CC CC CC FC CC CC                 155     DB      0CCH,0CCH,0CCH,0FCH,0CCH,0CCH,0CCH,000H ; H D_48
                CC 00                                   156
0248  78 30 30 30 30 30                 157     DB      078H,030H,030H,030H,030H,030H,078H,000H ; I D_49
                78 00                                   158
0250  1E 0C 0C 0C CC CC                 159     DB      01EH,00CH,00CH,00CH,0CCH,0CCH,078H,000H ; J D_4A
                78 00                                   160
0258  E6 66 6C 78 6C 66                 161     DB      0E6H,066H,06CH,078H,06CH,066H,0E6H,000H ; K D_4B
                E6 00                                   162
0260  F0 60 60 60 62 66                 163     DB      0F0H,060H,060H,060H,062H,066H,0FEH,000H ; L D_4C
                FE 00                                   164
0268  C6 EE FE FE D6 C6                 165     DB      0C6H,0EEH,0FEH,0FEH,0D6H,0C6H,0C6H,000H ; M D_4D
                C6 00                                   166
0270  C6 E6 F6 DE CE C6                 167     DB      0C6H,0E6H,0F6H,0DEH,0CEH,0C6H,0C6H,000H ; N D_4E
                C6 00                                   168
0278  38 6C C6 C6 C6 6C                 169     DB      038H,06CH,0C6H,0C6H,0C6H,06CH,038H,000H ; O D_4F
                38 00                                   170
                                                        171
0280  FC 66 66 7C 60 60                 172     DB      0FCH,066H,066H,07CH,060H,060H,0F0H,000H ; P D_50
                F0 00                                   173
0288  78 CC CC CC DC 78                 174     DB      078H,0CCH,0CCH,0CCH,0DCH,078H,01CH,000H ; Q D_51
                1C 00                                   175
0290  FC 66 66 7C 6C 66                 176     DB      0FCH,066H,066H,07CH,06CH,066H,0E6H,000H ; R D_52
                E6 00                                   177
0298  78 CC E0 70 1C CC                 178     DB      078H,0CCH,0E0H,070H,01CH,0CCH,078H,000H ; S D_53
                78 00                                   179
02A0  FC B4 30 30 30 30                 180     DB      0FCH,0B4H,030H,030H,030H,030H,078H,000H ; T D_54
                78 00                                   181
02A8  CC CC CC CC CC CC                 182     DB      0CCH,0CCH,0CCH,0CCH,0CCH,0CCH,0FCH,000H ; U D_55
```

```
                FC 00                                   183
02B0  CC CC CC CC CC 78                                 184             DB      0CCH,0CCH,0CCH,0CCH,0CCH,078H,030H,000H ; V D_56
      30 00                                             185
02B8  C6 C6 C6 D6 FE EE                                 186             DB      0C6H,0C6H,0C6H,0D6H,0FEH,0EEH,0C6H,000H ; W D_57
      C6 00                                             187
02C0  C6 C6 6C 38 38 6C                                 188             DB      0C6H,0C6H,06CH,038H,038H,06CH,0C6H,000H ; X D_58
      C6 00                                             189
02C8  CC CC CC 78 30 30                                 190             DB      0CCH,0CCH,0CCH,078H,030H,030H,078H,000H ; Y D_59
      78 00                                             191
02D0  FE C6 8C 18 32 66                                 192             DB      0FEH,0C6H,08CH,018H,032H,066H,0FEH,000H ; Z D_5A
      FE 00                                             193
02D8  78 60 60 60 60 60                                 194             DB      078H,060H,060H,060H,060H,060H,078H,000H ; [ D_5B
      78 00                                             195
02E0  C0 60 30 18 0C 06                                 196             DB      0C0H,060H,030H,018H,00CH,006H,002H,000H ; BACKSLASH D_5C
      02 00                                             197
02E8  78 18 18 18 18 18                                 198             DB      078H,018H,018H,018H,018H,018H,078H,000H ; ] D_5D
      78 00                                             199
02F0  10 38 6C C6 00 00                                 200             DB      010H,038H,06CH,0C6H,000H,000H,000H,000H ; CIRCUMFLEX D_5E
      00 00                                             201
02F8  00 00 00 00 00 00                                 202             DB      000H,000H,000H,000H,000H,000H,000H,0FFH ; _ D_5F
      00 FF                                             203
                                                        204
0300  30 30 18 00 00 00                                 205             DB      030H,030H,018H,000H,000H,000H,000H,000H ; ' D_60
      00 00                                             206
0308  00 00 78 0C 7C CC                                 207             DB      000H,000H,078H,00CH,07CH,0CCH,076H,000H ; LOWER CASE A D_61
      76 00                                             208
0310  E0 60 60 7C 66 66                                 209             DB      0E0H,060H,060H,07CH,066H,066H,0DCH,000H ; L.C. B D_62
      DC 00                                             210
0318  00 00 78 CC C0 CC                                 211             DB      000H,000H,078H,0CCH,0C0H,0CCH,078H,000H ; L.C. C D_63
      78 00                                             212
0320  1C 0C 0C 7C CC CC                                 213             DB      01CH,00CH,00CH,07CH,0CCH,0CCH,076H,000H ; L.C. D D_64
      76 00                                             214
0328  00 00 78 CC FC C0                                 215             DB      000H,000H,078H,0CCH,0FCH,0C0H,078H,000H ; L.C. E D_65
      78 00                                             216
0330  38 6C 60 F0 60 60                                 217             DB      038H,06CH,060H,0F0H,060H,060H,0F0H,000H ; L.C. F D_66
      F0 00                                             218
0338  00 00 76 CC CC 7C                                 219             DB      000H,000H,076H,0CCH,0CCH,07CH,00CH,0F8H ; L.C. G D_67
      0C F8                                             220
0340  E0 60 6C 76 66 66                                 221             DB      0E0H,060H,06CH,076H,066H,066H,0E6H,000H ; L.C. H D_68
      E6 00                                             222
0348  30 00 70 30 30 30                                 223             DB      030H,000H,070H,030H,030H,030H,078H,000H ; L.C. I D_69
      78 00                                             224
0350  0C 00 0C 0C 0C CC                                 225             DB      00CH,000H,00CH,00CH,00CH,0CCH,0CCH,078H ; L.C. J D_6A
      CC 78                                             226
0358  E0 60 66 6C 78 6C                                 227             DB      0E0H,060H,066H,06CH,078H,06CH,0E6H,000H ; L.C. K D_6B
      E6 00                                             228
0360  70 30 30 30 30 30                                 229             DB      070H,030H,030H,030H,030H,030H,078H,000H ; L.C. L D_6C
      78 00                                             230
0368  00 00 CC FE FE D6                                 231             DB      000H,000H,0CCH,0FEH,0FEH,0D6H,0C6H,000H ; L.C. M D_6D
      C6 00                                             232
0370  00 00 F8 CC CC CC                                 233             DB      000H,000H,0F8H,0CCH,0CCH,0CCH,0CCH,000H ; L.C. N D_6E
      CC 00                                             234
0378  00 00 78 CC CC CC                                 235             DB      000H,000H,078H,0CCH,0CCH,0CCH,078H,000H ; L.C. O D_6F
      78 00                                             236
                                                        237
0380  00 00 DC 66 66 7C                                 238             DB      000H,000H,0DCH,066H,066H,07CH,060H,0F0H ; L.C. P D_70
      60 F0                                             239
0388  00 00 76 CC CC 7C                                 240             DB      000H,000H,076H,0CCH,0CCH,07CH,00CH,01EH ; L.C. Q D_71
      0C 1E                                             241
0390  00 00 DC 76 66 60                                 242             DB      000H,000H,0DCH,076H,066H,060H,0F0H,000H ; L.C. R D_72
      F0 00                                             243
0398  00 00 7C C0 78 0C                                 244             DB      000H,000H,07CH,0C0H,078H,00CH,0F8H,000H ; L.C. S D_73
      F8 00                                             245
03A0  10 30 7C 30 30 34                                 246             DB      010H,030H,07CH,030H,030H,034H,018H,000H ; L.C. T D_74
      18 00                                             247
03A8  00 00 CC CC CC CC                                 248             DB      000H,000H,0CCH,0CCH,0CCH,0CCH,076H,000H ; L.C. U D_75
      76 00                                             249
03B0  00 00 CC CC CC 78                                 250             DB      000H,000H,0CCH,0CCH,0CCH,078H,030H,000H ; L.C. V D_76
      30 00                                             251
03B8  00 00 C6 D6 FE FE                                 252             DB      000H,000H,0C6H,0D6H,0FEH,0FEH,06CH,000H ; L.C. W D_77
      6C 00                                             253
03C0  00 00 C6 6C 38 6C                                 254             DB      000H,000H,0C6H,06CH,038H,06CH,0C6H,000H ; L.C. X D_78
      C6 00                                             255
03C8  00 00 CC CC CC 7C                                 256             DB      000H,000H,0CCH,0CCH,0CCH,07CH,00CH,0F8H ; L.C. Y D_79
      0C F8                                             257
03D0  00 00 FC 98 30 64                                 258             DB      000H,000H,0FCH,098H,030H,064H,0FCH,000H ; L.C. Z D_7A
      FC 00                                             259
03D8  1C 30 30 E0 30 30                                 260             DB      01CH,030H,030H,0E0H,030H,030H,01CH,000H ; L BRAK D_7B
      1C 00                                             261
03E0  18 18 18 00 18 18                                 262             DB      018H,018H,018H,000H,018H,018H,018H,000H ; | D_7C
      18 00                                             263
03E8  E0 30 30 1C 30 30                                 264             DB      0E0H,030H,030H,01CH,030H,030H,0E0H,000H ; R BRAK D_7D
      E0 00                                             265
03F0  76 DC 00 00 00 00                                 266             DB      076H,0DCH,000H,000H,000H,000H,000H,000H ; TILDE D_7E
      00 00                                             267
03F8  00 10 38 6C C6 C6                                 268             DB      000H,010H,038H,06CH,0C6H,0C6H,0FEH,000H ; DELTA D_7F
      FE 00                                             269
                                                        270
0400                                                    271     INT_1F_1        LABEL   BYTE
                                                        272
0400  78 CC C0 CC 78 18                                 273             DB      078H,0CCH,0C0H,0CCH,078H,018H,00CH,078H ;  D_80
      0C 78                                             274
0408  00 CC 00 CC CC CC                                 275             DB      000H,0CCH,000H,0CCH,0CCH,0CCH,07EH,000H ;  D_81
      7E 00                                             276
0410  1C 00 78 CC FC C0                                 277             DB      01CH,000H,078H,0CCH,0FCH,0C0H,078H,000H ;  D_82
      78 00                                             278
0418  7E C3 3C 06 3E 66                                 279             DB      07EH,0C3H,03CH,006H,03EH,066H,03FH,000H ;  D_83
      3F 00                                             280
0420  CC 00 78 0C 7C CC                                 281             DB      0CCH,000H,078H,00CH,07CH,0CCH,07EH,000H ;  D_84
      7E 00                                             282
0428  E0 00 78 0C 7C CC                                 283             DB      0E0H,000H,078H,00CH,07CH,0CCH,07EH,000H ;  D_85
      7E 00                                             284
0430  30 30 78 0C 7C CC                                 285             DB      030H,030H,078H,00CH,07CH,0CCH,07EH,000H ;  D_86
      7E 00                                             286
0438  00 00 78 C0 C0 78                                 287             DB      000H,000H,078H,0C0H,0C0H,078H,00CH,038H ;  D_87
      0C 38                                             288
0440  7E C3 3C 66 7E 60                                 289             DB      07EH,0C3H,03CH,066H,07EH,060H,03CH,000H ;  D_88
      3C 00                                             290
0448  CC 00 78 CC FC C0                                 291             DB      0CCH,000H,078H,0CCH,0FCH,0C0H,078H,000H ;  D_89
      78 00                                             292
0450  E0 00 78 CC FC C0                                 293             DB      0E0H,000H,078H,0CCH,0FCH,0C0H,078H,000H ;  D_8A
      78 00                                             294
0458  CC 00 70 30 30 30                                 295             DB      0CCH,000H,070H,030H,030H,030H,078H,000H ;  D_8B
      78 00                                             296
0460  7C C6 38 18 18 18                                 297             DB      07CH,0C6H,038H,018H,018H,018H,03CH,000H ;  D_8C
      3C 00                                             298
0468  E0 00 70 30 30 30                                 299             DB      0E0H,000H,070H,030H,030H,030H,078H,000H ;  D_8D
      78 00                                             300
0470  C6 38 6C C6 FE C6                                 301             DB      0C6H,038H,06CH,0C6H,0FEH,0C6H,0C6H,000H ;  D_8E
      C6 00                                             302
0478  30 30 00 78 CC FC                                 303             DB      030H,030H,000H,078H,0CCH,0FCH,0CCH,000H ;  D_8F
      CC 00                                             304
                                                        305
0480  1C 00 FC 60 78 60                                 306             DB      01CH,000H,0FCH,060H,078H,060H,0FCH,000H ;  D_90
      FC 00                                             307
0488  00 00 7F 0C 7F CC                                 308             DB      000H,000H,07FH,00CH,07FH,0CCH,07FH,000H ;  D_91
```

```
                7F 00                                   309
0490    3E 6C CC FE CC CC               DB      03EH,06CH,0CCH,0FEH,0CCH,0CCH,0CEH,000H ;  D_92
                CE 00                                   311
0498    78 CC 00 78 CC CC               DB      078H,0CCH,000H,078H,0CCH,0CCH,078H,000H ;  D_93
                78 00                                   313
04A0    00 CC CC 00 78 CC CC            DB      000H,0CCH,000H,078H,0CCH,0CCH,078H,000H ;  D_94
                78 00                                   315
04A8    00 E0 00 78 CC CC               DB      000H,0E0H,000H,078H,0CCH,0CCH,078H,000H ;  D_95
                78 00                                   317
04B0    78 CC 00 CC CC CC               DB      078H,0CCH,000H,0CCH,0CCH,0CCH,07EH,000H ;  D_96
                7E 00                                   319
04B8    00 E0 00 CC CC CC               DB      000H,0E0H,000H,0CCH,0CCH,0CCH,07EH,000H ;  D_97
                7E 00                                   321
04C0    00 CC 00 CC CC 7C               DB      000H,0CCH,000H,0CCH,0CCH,07CH,00CH,0F8H ;  D_98
                0C F8                                   323
04C8    C3 18 3C 66 66 3C               DB      0C3H,018H,03CH,066H,066H,03CH,018H,000H ;  D_99
                18 00                                   325
04D0    CC 00 CC CC CC CC               DB      0CCH,000H,0CCH,0CCH,0CCH,0CCH,078H,000H ;  D_9A
                78 00                                   327
04D8    18 18 7E C0 C0 7E               DB      018H,018H,07EH,0C0H,0C0H,07EH,018H,018H ;  D_9B
                18 18                                   329
04E0    38 6C 64 F0 60 E6               DB      038H,06CH,064H,0F0H,060H,0E6H,0FCH,000H ;  D_9C
                FC 00                                   331
04E8    CC CC 78 FC 30 FC               DB      0CCH,0CCH,078H,0FCH,030H,0FCH,030H,030H ;  D_9D
                30 30                                   333
04F0    F8 CC CC FA C6 CF               DB      0F8H,0CCH,0CCH,0FAH,0C6H,0CFH,0C6H,0C7H ;  D_9E
                C6 C7                                   335
04F8    0E 1B 18 3C 18 18               DB      00EH,01BH,018H,03CH,018H,018H,0D8H,070H ;  D_9F
                D8 70                                   337
                                                        338
0500    1C 00 78 0C 7C CC               DB      01CH,000H,078H,00CH,07CH,0CCH,07EH,000H ;  D_A0
                7E 00                                   340
0508    38 00 70 30 30 30               DB      038H,000H,070H,030H,030H,030H,078H,000H ;  D_A1
                78 00                                   342
0510    00 1C 00 78 CC CC               DB      000H,01CH,000H,078H,0CCH,0CCH,078H,000H ;  D_A2
                78 00                                   344
0518    00 1C 00 CC CC CC               DB      000H,01CH,000H,0CCH,0CCH,0CCH,07EH,000H ;  D_A3
                7E 00                                   346
0520    00 F8 00 F8 CC CC               DB      000H,0F8H,000H,0F8H,0CCH,0CCH,0CCH,000H ;  D_A4
                CC 00                                   348
0528    FC 00 CC EC FC DC               DB      0FCH,000H,0CCH,0ECH,0FCH,0DCH,0CCH,000H ;  D_A5
                CC 00                                   350
0530    3C 6C 6C 3E 00 7E               DB      03CH,06CH,06CH,03EH,000H,07EH,000H,000H ;  D_A6
                00 00                                   352
0538    38 6C 6C 38 00 7C               DB      038H,06CH,06CH,038H,000H,07CH,000H,000H ;  D_A7
                00 00                                   354
0540    30 00 30 60 C0 CC               DB      030H,000H,030H,060H,0C0H,0CCH,078H,000H ;  D_A8
                78 00                                   356
0548    00 00 00 FC C0 C0               DB      000H,000H,000H,0FCH,0C0H,0C0H,000H,000H ;  D_A9
                00 00                                   358
0550    00 00 00 FC 0C 0C               DB      000H,000H,000H,0FCH,00CH,00CH,000H,000H ;  D_AA
                00 00                                   360
0558    C3 C6 CC DE 33 66               DB      0C3H,0C6H,0CCH,0DEH,033H,066H,0CCH,00FH ;  D_AB
                CC 0F                                   362
0560    C3 C6 CC DB 37 6F               DB      0C3H,0C6H,0CCH,0DBH,037H,06FH,0CFH,003H ;  D_AC
                CF 03                                   364
0568    18 18 00 18 18 18               DB      018H,018H,000H,018H,018H,018H,018H,000H ;  D_AD
                18 00                                   366
0570    00 33 66 CC 66 33               DB      000H,033H,066H,0CCH,066H,033H,000H,000H ;  D_AE
                00 00                                   368
0578    00 CC 66 33 66 CC               DB      000H,0CCH,066H,033H,066H,0CCH,000H,000H ;  D_AF
                00 00                                   370
                                                        371
0580    22 88 22 88 22 88               DB      022H,088H,022H,088H,022H,088H,022H,088H ;  D_B0
                22 88                                   373
0588    55 AA 55 AA 55 AA               DB      055H,0AAH,055H,0AAH,055H,0AAH,055H,0AAH ;  D_B1
                55 AA                                   375
0590    DB 77 DB EE DB 77               DB      0DBH,077H,0DBH,0EEH,0DBH,077H,0DBH,0EEH ;  D_B2
                DB EE                                   377
0598    18 18 18 18 18 18               DB      018H,018H,018H,018H,018H,018H,018H,018H ;  D_B3
                18 18                                   379
05A0    18 18 18 18 F8 18               DB      018H,018H,018H,018H,0F8H,018H,018H,018H ;  D_B4
                18 18                                   381
05A8    18 18 F8 18 F8 18               DB      018H,018H,0F8H,018H,0F8H,018H,018H,018H ;  D_B5
                18 18                                   383
05B0    36 36 36 36 F6 36               DB      036H,036H,036H,036H,0F6H,036H,036H,036H ;  D_B6
                36 36                                   385
05B8    00 00 00 00 FE 36               DB      000H,000H,000H,000H,0FEH,036H,036H,036H ;  D_B7
                36 36                                   387
05C0    00 00 F8 18 F8 18               DB      000H,000H,0F8H,018H,0F8H,018H,018H,018H ;  D_B8
                18 18                                   389
05C8    36 36 F6 06 F6 36               DB      036H,036H,0F6H,006H,0F6H,036H,036H,036H ;  D_B9
                36 36                                   391
05D0    36 36 36 36 36 36               DB      036H,036H,036H,036H,036H,036H,036H,036H ;  D_BA
                36 36                                   393
05D8    00 00 FE 06 F6 36               DB      000H,000H,0FEH,006H,0F6H,036H,036H,036H ;  D_BB
                36 36                                   395
05E0    36 36 F6 06 FE 00               DB      036H,036H,0F6H,006H,0FEH,000H,000H,000H ;  D_BC
                00 00                                   397
05E8    36 36 36 36 FE 00               DB      036H,036H,036H,036H,0FEH,000H,000H,000H ;  D_BD
                00 00                                   399
05F0    18 18 F8 18 F8 00               DB      018H,018H,0F8H,018H,0F8H,000H,000H,000H ;  D_BE
                00 00                                   401
05F8    00 00 00 00 F8 18               DB      000H,000H,000H,000H,0F8H,018H,018H,018H ;  D_BF
                18 18                                   403
                                                        404
0600    18 18 18 18 1F 00               DB      018H,018H,018H,018H,01FH,000H,000H,000H ;  D_C0
                00 00                                   406
0608    18 18 18 18 FF 00               DB      018H,018H,018H,018H,0FFH,000H,000H,000H ;  D_C1
                00 00                                   408
0610    00 00 00 00 FF 18               DB      000H,000H,000H,000H,0FFH,018H,018H,018H ;  D_C2
                18 18                                   410
0618    18 18 18 18 1F 18               DB      018H,018H,018H,018H,01FH,018H,018H,018H ;  D_C3
                18 18                                   412
0620    00 00 00 00 FF 00               DB      000H,000H,000H,000H,0FFH,000H,000H,000H ;  D_C4
                00 00                                   414
0628    18 18 18 18 FF 18               DB      018H,018H,018H,018H,0FFH,018H,018H,018H ;  D_C5
                18 18                                   416
0630    18 18 1F 18 1F 18               DB      018H,018H,01FH,018H,01FH,018H,018H,018H ;  D_C6
                18 18                                   418
0638    36 36 36 36 37 36               DB      036H,036H,036H,036H,037H,036H,036H,036H ;  D_C7
                36 36                                   420
0640    36 36 37 30 3F 00               DB      036H,036H,037H,030H,03FH,000H,000H,000H ;  D_C8
                00 00                                   422
0648    00 00 3F 30 37 36               DB      000H,000H,03FH,030H,037H,036H,036H,036H ;  D_C9
                36 36                                   424
0650    36 36 F7 00 FF 00               DB      036H,036H,0F7H,000H,0FFH,000H,000H,000H ;  D_CA
                00 00                                   426
0658    00 00 FF 00 F7 36               DB      000H,000H,0FFH,000H,0F7H,036H,036H,036H ;  D_CB
                36 36                                   428
0660    36 36 37 30 37 36               DB      036H,036H,037H,030H,037H,036H,036H,036H ;  D_CC
                36 36                                   430
0668    00 00 FF 00 FF 00               DB      000H,000H,0FFH,000H,0FFH,000H,000H,000H ;  D_CD
                00 00                                   432
0670    36 36 F7 00 F7 36               DB      036H,036H,0F7H,000H,0F7H,036H,036H,036H ;  D_CE
                36 36                                   434
```

```
0678  18 18 FF 00 FF 00      435                    DB      018H,018H,0FFH,000H,0FFH,000H,000H,000H ;  D_CF
      00 00                  436
                             437
0680  36 36 36 36 FF 00      438                    DB      036H,036H,036H,036H,0FFH,000H,000H,000H ;  D_D0
      00 00                  439
0688  00 00 FF 00 FF 18      440                    DB      000H,000H,0FFH,000H,0FFH,018H,018H,018H ;  D_D1
      18 18                  441
0690  00 00 00 00 FF 36      442                    DB      000H,000H,000H,000H,0FFH,036H,036H,036H ;  D_D2
      36 36                  443
0698  36 36 36 36 3F 00      444                    DB      036H,036H,036H,036H,03FH,000H,000H,000H ;  D_D3
      00 00                  445
06A0  18 18 1F 18 1F 00      446                    DB      018H,018H,01FH,018H,01FH,000H,000H,000H ;  D_D4
      00 00                  447
06A8  00 00 1F 18 1F 18      448                    DB      000H,000H,01FH,018H,01FH,018H,018H,018H ;  D_D5
      18 18                  449
06B0  00 00 00 00 3F 36      450                    DB      000H,000H,000H,000H,03FH,036H,036H,036H ;  D_D6
      36 36                  451
06B8  36 36 36 36 FF 36      452                    DB      036H,036H,036H,036H,0FFH,036H,036H,036H ;  D_D7
      36 36                  453
06C0  18 18 FF 18 FF 18      454                    DB      018H,018H,0FFH,018H,0FFH,018H,018H,018H ;  D_D8
      18 18                  455
06C8  18 18 18 18 F8 00      456                    DB      018H,018H,018H,018H,0F8H,000H,000H,000H ;  D_D9
      00 00                  457
06D0  00 00 00 00 1F 18      458                    DB      000H,000H,000H,000H,01FH,018H,018H,018H ;  D_DA
      18 18                  459
06D8  FF FF FF FF FF FF      460                    DB      0FFH,0FFH,0FFH,0FFH,0FFH,0FFH,0FFH,0FFH ;  D_DB
      FF FF                  461
06E0  00 00 00 00 FF FF      462                    DB      000H,000H,000H,000H,0FFH,0FFH,0FFH,0FFH ;  D_DC
      FF FF                  463
06E8  F0 F0 F0 F0 F0 F0      464                    DB      0F0H,0F0H,0F0H,0F0H,0F0H,0F0H,0F0H,0F0H ;  D_DD
      F0 F0                  465
06F0  0F 0F 0F 0F 0F 0F      466                    DB      00FH,00FH,00FH,00FH,00FH,00FH,00FH,00FH ;  D_DE
      0F 0F                  467
06F8  FF FF FF FF 00 00      468                    DB      0FFH,0FFH,0FFH,0FFH,000H,000H,000H,000H ;  D_DF
      00 00                  469
                             470
0700  00 00 76 DC C8 DC      471                    DB      000H,000H,076H,0DCH,0C8H,0DCH,076H,000H ;  D_E0
      76 00                  472
0708  00 78 CC F8 CC F8      473                    DB      000H,078H,0CCH,0F8H,0CCH,0F8H,0C0H,0C0H ;  D_E1
      C0 C0                  474
0710  00 FC CC C0 C0 C0      475                    DB      000H,0FCH,0CCH,0C0H,0C0H,0C0H,0C0H,000H ;  D_E2
      C0 00                  476
0718  00 FE 6C 6C 6C 6C      477                    DB      000H,0FEH,06CH,06CH,06CH,06CH,06CH,000H ;  D_E3
      6C 00                  478
0720  FC CC 60 30 60 CC      479                    DB      0FCH,0CCH,060H,030H,060H,0CCH,0FCH,000H ;  D_E4
      FC 00                  480
0728  00 00 7E D8 D8 D8      481                    DB      000H,000H,07EH,0D8H,0D8H,0D8H,070H,000H ;  D_E5
      70 00                  482
0730  00 66 66 66 66 7C      483                    DB      000H,066H,066H,066H,066H,07CH,060H,0C0H ;  D_E6
      60 C0                  484
0738  00 76 DC 18 18 18      485                    DB      000H,076H,0DCH,018H,018H,018H,018H,000H ;  D_E7
      18 00                  486
0740  FC 30 78 CC CC 78      487                    DB      0FCH,030H,078H,0CCH,0CCH,078H,030H,0FCH ;  D_E8
      30 FC                  488
0748  38 6C C6 FE C6 6C      489                    DB      038H,06CH,0C6H,0FEH,0C6H,06CH,038H,000H ;  D_E9
      38 00                  490
0750  38 6C C6 C6 6C 6C      491                    DB      038H,06CH,0C6H,0C6H,06CH,06CH,0EEH,000H ;  D_EA
      EE 00                  492
0758  1C 30 18 7C CC CC      493                    DB      01CH,030H,018H,07CH,0CCH,0CCH,078H,000H ;  D_EB
      78 00                  494
0760  00 00 7E DB DB 7E      495                    DB      000H,000H,07EH,0DBH,0DBH,07EH,000H,000H ;  D_EC
      00 00                  496
0768  06 0C 7E DB DB 7E      497                    DB      006H,00CH,07EH,0DBH,0DBH,07EH,060H,0C0H ;  D_ED
      60 C0                  498
0770  38 60 C0 F8 C0 60      499                    DB      038H,060H,0C0H,0F8H,0C0H,060H,038H,000H ;  D_EE
      38 00                  500
0778  78 CC CC CC CC CC      501                    DB      078H,0CCH,0CCH,0CCH,0CCH,0CCH,0CCH,000H ;  D_EF
      CC 00                  502
                             503
0780  00 FC 00 FC 00 FC      504                    DB      000H,0FCH,000H,0FCH,000H,0FCH,000H,000H ;  D_F0
      00 00                  505
0788  30 30 FC 30 30 00      506                    DB      030H,030H,0FCH,030H,030H,000H,0FCH,000H ;  D_F1
      FC 00                  507
0790  60 30 18 30 60 00      508                    DB      060H,030H,018H,030H,060H,000H,0FCH,000H ;  D_F2
      FC 00                  509
0798  18 30 60 30 18 00      510                    DB      018H,030H,060H,030H,018H,000H,0FCH,000H ;  D_F3
      FC 00                  511
07A0  0E 1B 1B 18 18 18      512                    DB      00EH,01BH,01BH,018H,018H,018H,018H,018H ;  D_F4
      18 18                  513
07A8  18 18 18 18 18 D8      514                    DB      018H,018H,018H,018H,018H,0D8H,0D8H,070H ;  D_F5
      D8 70                  515
07B0  30 30 30 00 FC 00 30   516                    DB      030H,030H,000H,0FCH,000H,030H,030H,000H ;  D_F6
      30 00                  517
07B8  00 76 DC 00 76 DC      518                    DB      000H,076H,0DCH,000H,076H,0DCH,000H,000H ;  D_F7
      00 00                  519
07C0  38 6C 6C 38 00 00      520                    DB      038H,06CH,06CH,038H,000H,000H,000H,000H ;  D_F8
      00 00                  521
07C8  00 00 00 18 18 00      522                    DB      000H,000H,000H,018H,018H,000H,000H,000H ;  D_F9
      00 00                  523
07D0  00 00 00 00 18 00      524                    DB      000H,000H,000H,000H,018H,000H,000H,000H ;  D_FA
      00 00                  525
07D8  0F 0C 0C 0C EC 6C      526                    DB      00FH,00CH,00CH,00CH,0ECH,06CH,03CH,01CH ;  D_FB
      3C 1C                  527
07E0  78 6C 6C 6C 6C 00      528                    DB      078H,06CH,06CH,06CH,06CH,000H,000H,000H ;  D_FC
      00 00                  529
07E8  70 18 30 60 78 00      530                    DB      070H,018H,030H,060H,078H,000H,000H,000H ;  D_FD
      00 00                  531
07F0  00 00 3C 3C 3C 3C      532                    DB      000H,000H,03CH,03CH,03CH,03CH,000H,000H ;  D_FE
      00 00                  533
07F8  00 00 00 00 00 00      534                    DB      000H,000H,000H,000H,000H,000H,000H,000H ;  D_FF
      00 00                  535
0800                         536        CODE        ENDS
                             537                    END


                             1                      PAGE,120
                             2                      SUBTTL  END ADDRESS
0000                         3          CODE        SEGMENT PUBLIC
                             4                      PUBLIC  END_ADDRESS
0000                         5          END_ADDRESS LABEL   BYTE
0000                         6          CODE        ENDS
                             7                      END
```

Index

F

G

H

I

L

M

O

P

R

S

U

V

Personal Computer Hardware Reference Library

IBM Printer Adapter

6361507

Contents

Description

The IBM Printer Adapter is specifically designed to attach printers with a parallel port-interface, but it can be used as a general input/output port for any device or application that matches its input/output capabilities. It has 12 TTL-buffer output points, which are latched and can be written and read under program control using the microprocessor In or Out instruction. The adapter also has five steady-state input points that may be read using the microprocessor's In instructions.

In addition, one input can also be used to create a microprocessor interrupt. This interrupt can be enabled and disabled under program control. A reset from the power-on circuit is also ORed with a program output point, allowing a device to receive a 'power-on reset' when the system unit's microprocessor is reset.

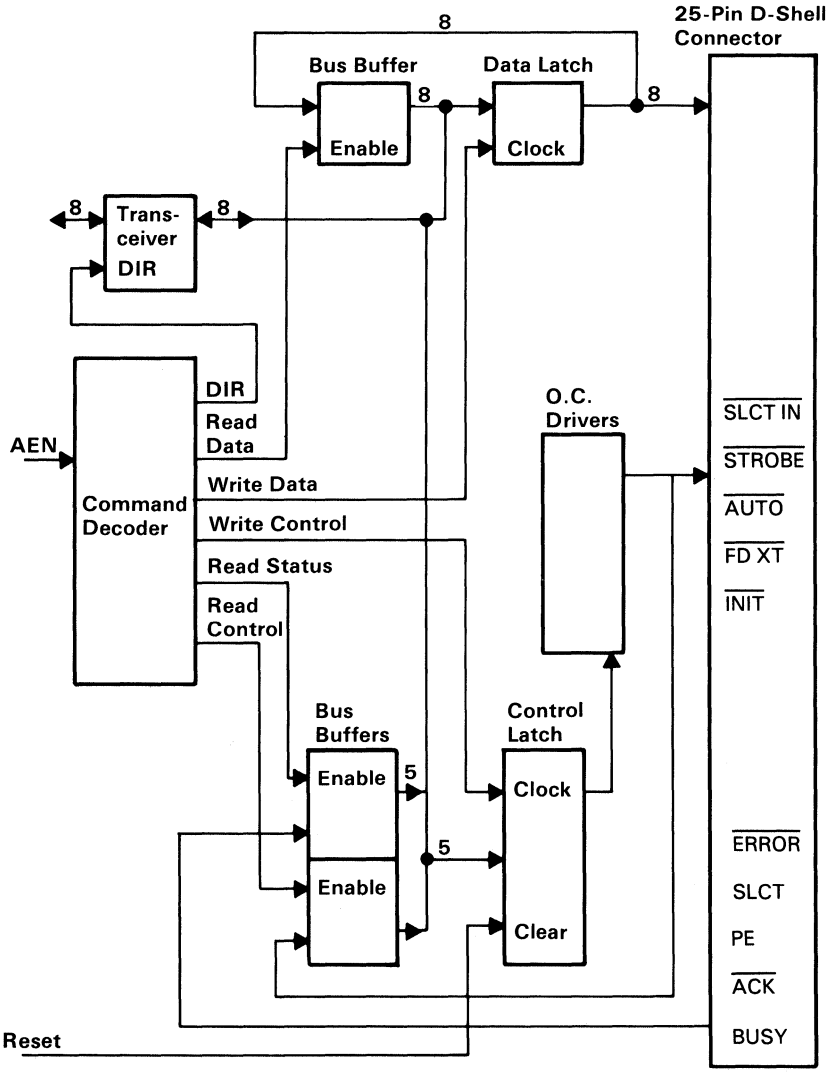
The input/output signals are made available at the back of the adapter through a right-angle, printed-circuit-board-mounted, 25-pin, D-shell connector. This connector protrudes through the rear panel of the system unit or expansion unit, where a cable may be attached.

When this adapter is used to attach a printer, data or printer commands are loaded into an 8-bit, latched, output port, and the strobe line is activated, writing data to the printer. The program then may read the input ports for printer status indicating when the next character can be written, or it may use the interrupt line to indicate "not busy" to the software.

The output ports may also be read at the card's interface for diagnostic loop functions. This allows faults to be isolated to the adapter or the attaching device.

This same function is also part of the IBM Monochrome Display and Printer Adapter.

The following is a block diagram of the Printer Adapter.



Printer Adapter Block Diagram

Programming Considerations

The Printer Adapter responds to five I/O instructions; two output and three input. The output instructions transfer data into two latches whose outputs are presented on pins of a 25-pin D-shell connector.

Two of the three input instructions allow the system unit's microprocessor to read back the contents of the two latches. The third allows the system unit's microprocessor to read the real-time status from a group of pins on the connector.

A description of each instruction follows.

Printer Adapter			
Output to address hex 378			
Bit 3 Pin 5	Bit 2 Pin 4	Bit 1 Pin 3	Bit 0 Pin 2

The instruction captures data from the data bus and is present on the respective pins. Each of these pins is capable of sourcing 2.6 mA and sinking 24 mA.

It is essential that the external device does not try to pull these lines to ground.

Printer Adapter			
Output to address hex 37A			
$\overline{\text{Bit 3}}$ Pin 17	Bit 2 Pin 16	$\overline{\text{Bit 1}}$ Pin 14	$\overline{\text{Bit 0}}$ Pin 1

This instruction causes the latch to capture the five least significant bits of the data bus. The four least significant bits present their outputs, or inverted versions of their outputs, to the

respective pins as shown in the previous figure. If bit 4 is written as a 1, the card will interrupt the system unit's microprocessor on the condition that pin 10 changes from high to low.

These pins are driven by open-collector drivers pulled to +5 Vdc through 4.7 kΩ resistors. They can each sink approximately 7 mA and maintain 0.8 volts down-level.

Printer Adapter
Input from address hex 378

This instruction presents the system unit's microprocessor with data present on the pins associated with the output to hex 3BC. This should normally reflect the exact value that was last written to hex 3BC. If an external device should be driving data on these pins at the time of an input (in violation of usage ground rules), this data will be ORed with the latch contents.

Printer Adapter
Input from address hex 379

This instruction presents the real-time status to the system unit's microprocessor from the pins, as follows.

Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
Pin 11	Pin 10	Pin 12	Pin 13	Pin 15	—	—	—

Printer Adapter
Input from address hex 37A

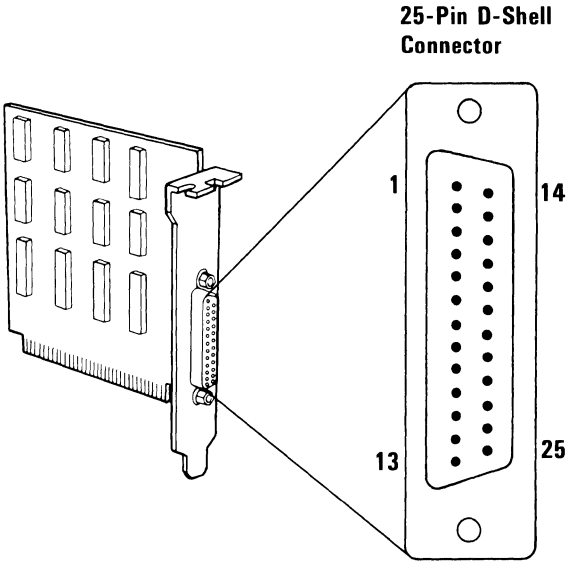
This instruction causes the data present on pins 1, 14, 16, 17, and the IRQ bit to be read by the system unit's microprocessor. In the absence of external drive applied to these pins, data read by the system unit's microprocessor will match data last written to hex 3BE in the same bit positions. Notice that data bits 0–2 are not included. If external drivers are dotted to these pins, that data will be ORed with data applied to the pins by the hex 3BE latch.

Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
			IRQ Enable	$\overline{\text{Pin 17}}$	Pin 16	$\overline{\text{Pin 14}}$	$\overline{\text{Pin 1}}$
			Por = 0	Por = 1	Por = 0	Por = 1	Por = 1

These pins assume the states shown after a reset from the system unit's microprocessor.

6 Printer Adapter

Specifications



At Standard TTL Levels

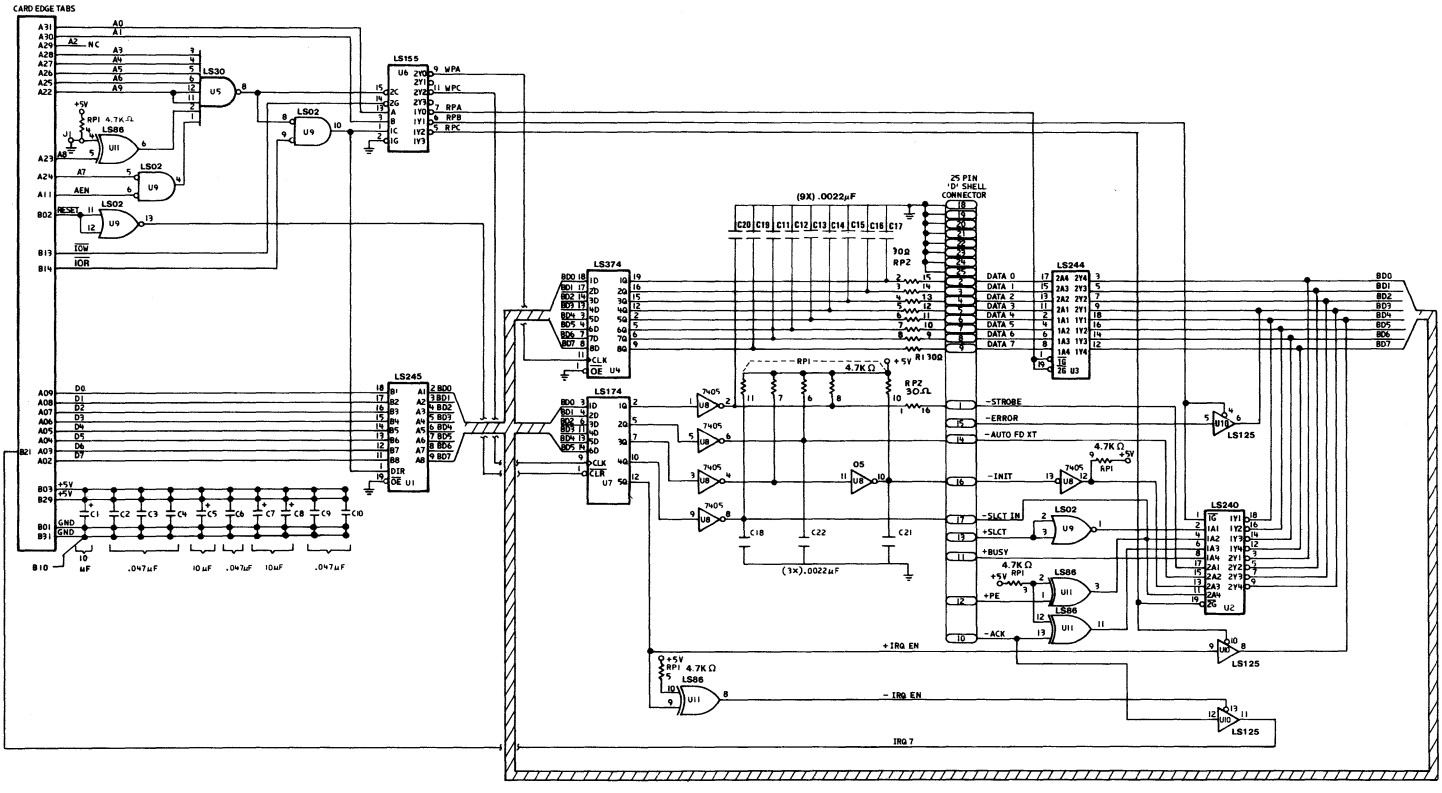
Printer	Signal Name	Adapter Pin Number	Printer Adapter
	– Strobe	1	
	+ Data Bit 0	2	
	+ Data Bit 1	3	
	+ Data Bit 2	4	
	+ Data Bit 3	5	
	+ Data Bit 4	6	
	+ Data Bit 5	7	
	+ Data Bit 6	8	
	+ Data Bit 7	9	
	– Acknowledge	10	
	+ Busy	11	
	+ P.End (out of paper)	12	
	+ Select	13	
	– Auto Feed	14	
	– Error	15	
	– Initialize Printer	16	
	– Select Input	17	
	Ground	18-25	

Connector Specifications

8 Printer Adapter

Logic Diagrams

The following page contains the logic diagram for the IBM Printer Adapter.



Printer Adapter (Sheet 1 of 1)

Personal Computer Hardware Reference Library

IBM 5-1/4" Diskette Drive Adapter

6361505

IBM 5-1/4" Diskette Drive Adapter

Contents

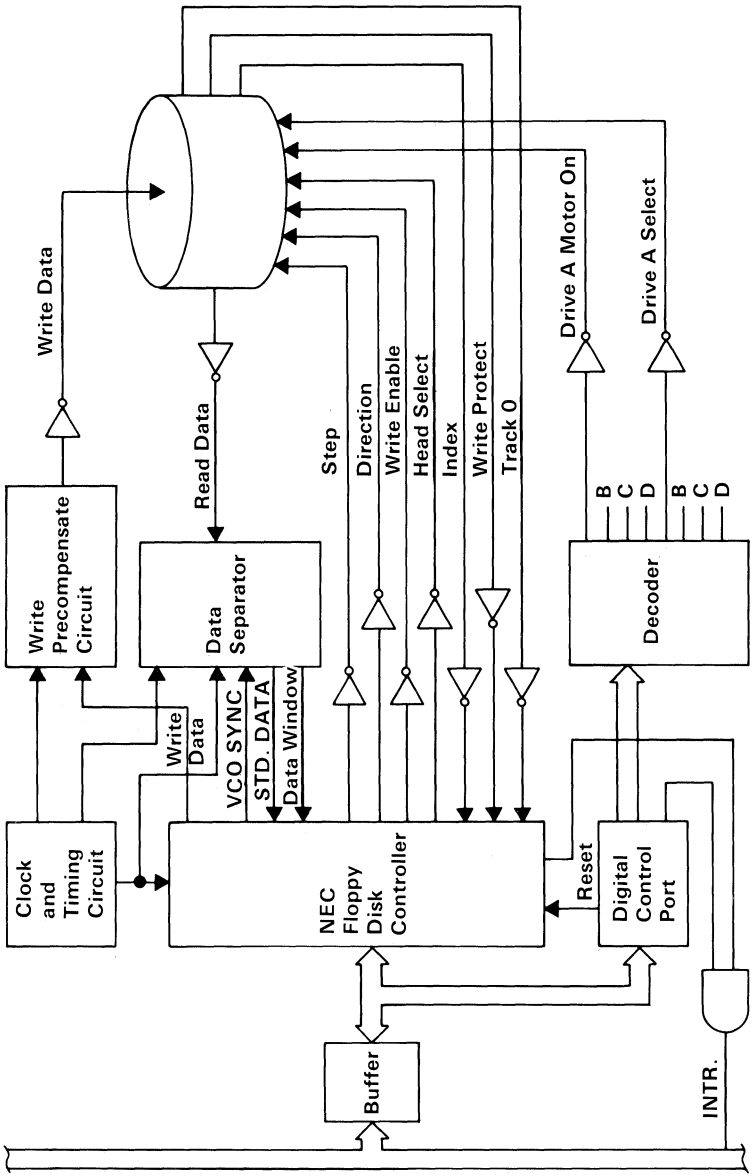
Description

The IBM 5-1/4" Diskette Drive Adapter fits into one of the expansion slots in the system unit. It is connected to one or two diskette drives through an internal, daisy-chained flat cable. The adapter has a connector at the other end that extends through the rear panel of the system unit. This connector has signals for two additional external diskette drives; thus, the 5-1/4 inch diskette drive adapter can attach four 5-1/4 inch drives — two internal and two external.

The adapter is designed for double-density, MFM-coded, diskette drives and uses write precompensation with an analog phase-lock loop for clock and data recovery. The adapter is a general-purpose device using the NEC μPD765 or equivalent controller. Therefore, the diskette drive parameters are programmable. In addition, the attachment supports the diskette drive's write-protect feature. The adapter is buffered on the I/O bus and uses the system board's direct memory access (DMA) for record data transfers. An interrupt level also is used to indicate when an operation is complete and that a status condition requires microprocessor attention.

In general, the 5-1/4 inch diskette drive adapter presents a high-level command interface to software I/O drivers.

The following is a block diagram of the IBM 5-1/4" Diskette Drive Adapter.



5-1/4 Inch Diskette Drive Adapter Block Diagram

Programming Considerations

This attachment consists of an 8-bit digital output register in parallel with a NEC μPD765 or equivalent floppy disk controller (FDC).

In the following description, drive numbers 0, 1, 2, and 3 are equivalent to drives A, B, C, and D.

Digital-Output Register

The Digital-Output register (DOR) is an output-only register used to control drive motors, drive selection, and feature enable. All bits are cleared by the I/O interface 'reset' line. The bits have the following functions:

Bits 0 and 1 These bits are decoded by the hardware to select one drive if its motor is on:

Bit 1 0	Drive
0 0	0 (A)
0 1	1 (B)
1 0	2 (C)
1 1	3 (D)

Bit 2 The FDC is held reset when this bit is clear. It must be set by the program to enable the FDC.

Bit 3 This bit allows the FDC interrupt and DMA requests to be gated onto the I/O interface. If this bit is cleared, the interrupt and DMA request I/O interface drivers are disabled.

Bits 4, 5, 6, and 7 These bits control, respectively, the motors of drives 0, 1, 2 (A, B, C), and 3 (D). If a bit is clear, the associated motor is off, and the drive cannot be selected.

Floppy Disk Controller

The floppy disk controller (FDC) contains two registers that may be accessed by the system unit's microprocessor: a status register and a data register. The 8-bit main status register contains the status information of the FDC and may be accessed at any time. The 8-bit data register (actually consisting of several registers in a stack with only one register presented to the data bus at a time) stores data, commands, parameters, and provides floppy disk drive (FDD) status information. Data bytes are read from or written to the data register in order to program or obtain results after a particular command. The main status register can only be read and is used to facilitate the transfer of data between the system unit's microprocessor and FDC.

The bits in the main status register (hex 34F) are defined as follows:

Bit Number	Name	Symbol	Description
DB0	FDD A Busy	DAB	FDD number 0 is in the Seek mode.
DB1	FDD B Busy	DBB	FDD number 1 is in the Seek mode.
DB2	FDD C Busy	DCB	FDD number 2 is in the Seek mode.
DB3	FDD D Busy	DDB	FDD number 3 is in the Seek mode.
DB4	FDC Busy	CB	A read or write command is in process.
DB5	Non-DMA Mode	NDM	The FDC is in the non-DMA mode.
DB6	Data Input/ Output	DIO	Indicates direction of data transfer between FDC and processor. If DIO = "1," then transfer is from FDC data register to the processor. If DIO = "0," then transfer is from the processor to FDC data register.
DB7	Request for Master	RQM	Indicates data register is ready to send or receive data to or from the processor. Both bits DIO and RQM should be used to perform the handshaking functions of "ready" and "direction" to the processor.

The FDC is capable of performing 15 different commands. Each command is initiated by a multi-byte transfer from the system unit's microprocessor, and the result after execution of the command may also be a multi-byte transfer back to the system

unit's microprocessor. Because of this multi-byte interchange of information between the FDC and the system unit's microprocessor, it is convenient to consider each command as consisting of three phases:

Command Phase

The FDC receives all information required to perform a particular operation from the system unit's microprocessor.

Execution Phase

The FDC performs the operation it was instructed to do.

Result Phase

After completion of the operation, status and other housekeeping information are made available to the system unit's microprocessor.

The following tables define the symbols used in the command summary. The command summary immediately follows these tables.

Symbol	Name	Description
A0	Address Line 0	A0 controls selection of main status register (A0 = 0) or data register (A0 = 1).
C	Cylinder Number	C stands for the current/selected cylinder (track) number of the medium.
D	Data	D stands for the data pattern that is going to be written into a sector.
D7-D0	Data Bus	8-bit data bus, where D7 stands for a most significant bit, and D0 stands for a least significant bit.
DTL	Data Length	When N is defined as 00, DTL stands for the data length that users are going to read from or write to the sector.
EOT	End of Track	EOT stands for the final sector number on a cylinder.
GPL	Gap Length	GPL stands for the length of gap 3 (spacing between sectors excluding VCO sync field).
H	Head Address	H stands for head number 0 or 1, as specified in ID field.
HD	Head	HD stands for a selected head number 0 or 1. (H = HD in all command words).
HLT	Head Load Time	HLT stands for the head load time in the FDD (4 to 512 ms in 4-ms increments).
HUT	Head Unload Time	HUT stands for the head unload time after a read or write operation has occurred (0 to 480 ms in 32-ms increments).
MF	FM or MFM Mode	If MF is low, FM mode is selected; if it is high, MFM mode is selected only if MFM is implemented.
MT	Multi-Track	If MT is high, a multi-track operation is to be performed. (A cylinder under both HD0 and HD1 will be read or written.)
N	Number	N stands for the number of data bytes written in a sector.

Symbol Descriptions (Part 1 of 2)

Symbol	Name	Description
NCN	New Cylinder Number	NCN stands for a new cylinder number, which is going to be reached as a result of the seek operation. (Desired position of the head.)
ND	Non-DMA Mode	ND stands for operation in the non-DMA mode.
PCN	Present Cylinder Number	PCN stands for cylinder number at the completion of sense-interrupt-status command indicating the position of the head at present time.
R	Record	R stands for the sector number, which will be read or written.
R/W	Read/Write	R/W stands for either read (R) or write (W) signal.
SC	Sector	SC indicates the number of sectors per cylinder.
SK	Skip	SK stands for skip deleted-data address mark.
SRT	Step Rate Time	SRT stands for the stepping rate for the FDD (2 to 32 ms in 2-ms increments).
ST 0 ST 1 ST 2 ST 3	Status 0 Status 1 Status 2 Status 3	ST0-3 stand for one of four registers that store the status information after a command has been executed. This information is available during the result phase after command execution. These registers should not be confused with the main status register (selected by A0 = 0). ST 0-3 may be read only after a command has been executed and contain information relevant to that particular command.
STP	Scan Test	During a scan operation, if STP = 1, the data in contiguous sectors is compared byte-by-byte with data sent from the processor (or DMA), and if STP = 2, then alternate sectors are read and compared.
US0, US1	Unit Select	US stands for a selected drive number encoded the same as bits 0 and 1 of the digital output register (DOR).

Symbol Descriptions (Part 2 of 2)

Command Summary

In the following table, 0 indicates "logical 0" for that bit, 1 means "logical 1," and X means "don't care."

Phase	R/W	Data Bus D7	D6	D5	D4	D3	D2	D1	D0	Remarks
		Read Data								
Command	W	MT	MF	SK	0	0	1	1	0	Command Codes
	W	X	X	X	X	X	HD	US1	US0	
	W				C					Sector ID information
	W				H					prior to command
	W				R					execution.
	W				N					
	W				EOT					
	W				GPL					
	W				DTL					
Execution										Data transfer between the FDD and main system.
Result	R				ST 0					Status information
	R				ST 1					after command
	R				ST 2					execution.
	R				C					Sector ID information
	R				H					after command
	R				R					execution.
	R				N					
		Read Deleted Data								
Command	W	MT	MF	SK	0	1	1	0	0	Command Codes
	W	X	X	X	X	X	HD	US1	US0	
	W				C					Sector ID information
	W				H					prior to command
	W				R					execution.
	W				N					
	W				EOT					
	W				GPL					
	W				DTL					
Execution										Data transfer between the FDD and main system.
Result	R				ST 0					Status information
	R				ST 1					after command
	R				ST 2					execution.
	R				C					Sector ID information
	R				H					after command
	R				R					execution.
	R				N					

Phase	R/W	Data Bus D7	D6	D5	D4	D3	D2	D1	D0	Remarks
		Write Data								
Command	W	MT	MF	0	0	0	1	0	1	Command Codes
	W	X	X	X	X	X	HD	US1	US0	
	W				C					Sector ID information
	W				H					prior to command
	W				R					execution.
	W				N					
	W				EOT					
	W				GPL					
	W				DTL					
Execution										Data transfer between the main system and FDD.
Result	R				ST 0					Status information
	R				ST 1					after command
	R				ST 2					execution.
	R				C					Sector ID information
	R				H					after command
	R				R					execution.
	R				N					
		Write Deleted Data								
Command	W	MT	MF	0	0	1	0	0	1	Command Codes
	W	X	X	X	X	X	HD	US1	US0	
	W				C					Sector ID information
	W				H					prior to command
	W				R					execution.
	W				N					
	W				EOT					
	W				GPL					
	W				DTL					
Execution										Data transfer between the FDD and main system.
Result	R				ST 0					Status ID information
	R				ST 1					after command
	R				ST 2					execution.
	R				C					Sector ID information
	R				H					after command
	R				R					execution.
	R				N					

Phase	R/W	Data Bus D7	D6	D5	D4	D3	D2	D1	D0	Remarks
		Read a Track								
Command	W	0	MF	SK	0	0	0	1	0	Command Codes
	W	X	X	X	X	X	HD	US1	US0	
	W	C								Sector ID information
	W	H								prior to command
	W	R								execution.
	W	N								
	W	EOT								
	W	GPL								
	W	DTL								
Execution										Data transfer between the FDD and main system. FDC reads all of cylinder's contents from index hole to EOT.
Result	R	ST 0								Status information
	R	ST 1								after command
	R	ST 2								execution.
	R	C								Sector ID information
	R	H								after command
	R	R								execution.
	R	N								
		Read ID								
Command	W	0	MF	0	0	1	0	1	0	Command Codes
	W	X	X	X	X	X	HD	US1	US0	
Execution										The first correct ID information on the cylinder is stored in data register.
Result	R	ST 0								Status information
	R	ST 1								after command
	R	ST 2								execution.
	R	C								Sector ID information
	R	H								during execution
	R	R								phase.
	R	N								

Phase	R/W	Data Bus D7	D6	D5	D4	D3	D2	D1	D0	Remarks
		Format a Track								
Command	W	0	MF	0	0	1	1	0	0	Command Codes
	W	X	X	X	X	X	HD	US1	US0	
	W				N					Bytes/Sector
	W				SC					Sector/Track
	W				GPL					Gap 3
	W				D					filler byte.
Execution										FDC formats an entire cylinder.
Result	R				ST 0					Status information
	R				ST 1					after command
	R				ST 2					execution.
	R				C					In this case, the ID
	R				H					information has no
	R				R					meaning.
	R				N					
		Scan Equal								
Command	W	MT	MF	SK	1	0	0	0	1	Command Codes
	W	X	X	X	X	X	HD	US1	US0	
	W				C					Sector ID information
	W				H					prior to command
	W				R					execution.
	W				N					
	W				EOT					
	W				GPL					
	W				STP					
Execution										Data compared between the FDD and the main system.
Result	R				ST 0					Status information
	R				ST 1					after Command
	R				ST 2					execution.
	R				C					Sector ID information
	R				H					after command
	R				R					execution.
	R				N					

Phase	R/W	Data Bus D7	D6	D5	D4	D3	D2	D1	D0	Remarks
		Scan Low or Equal								
Command	W	MT	MF	SK	1	1	0	0	1	Command Codes
	W	X	X	X	X	X	HD	US1	US0	
	W				C					Sector ID information
	W				H					prior to command
	W				R					execution.
	W				N					
	W				EOT					
	W				GPL					
	W				STP					
Execution										Data compared between the FDD and main system.
Result	R				ST 0					Status information
	R				ST 1					after command
	R				ST 2					execution.
	R				C					Sector ID information
	R				H					after command
	R				R					execution.
	R				N					
		Scan High or Equal								
Command	W	MT	MF	SK	1	1	1	0	1	Command Codes
	W	X	X	X	X	X	HD	US1	US0	
	W				C					Sector ID information
	W				H					prior to command
	W				R					execution.
	W				N					
	W				EOT					
	W				GPL					
	W				STP					
Execution										Data compared between the FDD and main system.
Result	R				ST 0					Status information
	R				ST 1					after command
	R				ST 2					execution.
	R				C					Sector ID information
	R				H					after command
	R				R					execution.
	R				N					

Phase	R/W	Data Bus D7	D6	D5	D4	D3	D2	D1	D0	Remarks
		Recalibrate								
Command	W	0	0	0	0	0	1	1	1	Command Codes
	W	X	X	X	X	X	0	US1	US0	
Execution										Head retracted to
No Result										track 0
Phase										
		Sense Interrupt Status								
Command	W	0	0	0	0	1	0	0	0	Command Codes
Result	R					ST 0				Status information at
	R					PCN				the end of seek operation about the FDC
		Specify								
Command	W	0	0	0	0	0	0	1	1	Command Codes
	W	—SRT—						—HUT—		
	W	—HLT—							ND	
No Result										
Phase										
		Sense Drive Status								
Command	W	0	0	0	0	0	1	0	0	Command Codes
	W	X	X	X	X	X	HD	US1	US0	
Result	R					ST 3				Status information about FDD.
		Seek								
Command	W	0	0	0	0	1	1	1	1	Command Codes
	W	X	X	X	X	X	HD	US1	US0	
	W					NCN				
Execution										Head is positioned over proper cylinder on diskette.
No Result										
Phase										
		Invalid								
Command	W	Invalid Codes								Invalid command codes (NoOp — FDC goes into standby state).
Result	R					ST 0				ST 0 = 80.

Bit			Description
No.	Name	Symbol	
D7 D6	Interrupt Code	IC	D7 = 0 and D6 = 0 Normal termination of command (NT). Command was completed and properly executed. D7 = 0 and D6 = 1 Abnormal termination of command (AT). Execution of command was started, but was not successfully completed. D7 = 1 and D6 = 0 Invalid command issue (IC). Command that was issued was never started. D7 = 1 and D6 = 1 Abnormal termination because, during command execution, the ready signal from FDD changed state.
D5	Seek End	SE	When the FDC completes the seek command, this flag is set to 1 (high).
D4	Equipment Check	EC	If a fault signal is received from the FDD, or if the track 0 signal fails to occur after 77 step pulses (recalibrate command), then this flag is set.
D3	Not Ready	NR	When the FDD is in the not-ready state and a read or write command is issued, this flag is set. If a read or write command is issued to side 1 of a single-sided drive, then this flag is set.
D2	Head Address	HD	This flag is used to indicate the state of the head at interrupt.
D1 D0	Unit Select 1 Unit Select 0	US 1 US 0	These flags are used to indicate a drive unit number at interrupt.

Command Status Register 0

Bit			Description
No.	Name	Symbol	
D7	End of Cylinder	EN	When the FDC tries to access a sector beyond the final sector of a cylinder, this flag is set.
D6	—	—	Not used. This bit is always 0 (low).
D5	Data Error	DE	When the FDC detects a CRC error in either the ID field or the data field, this flag is set.
D4	Over Run	OR	If the FDC is not serviced by the main system during data transfers within a certain time interval, this flag is set.
D3	—	—	Not used. This bit is always 0 (low).
D2	No Data	ND	During execution of a read data, write deleted data, or scan command, if the FDC cannot find the sector specified in the ID register, this flag is set. During execution of the read ID command, if the FDC cannot read the ID field without an error, then this flag is set. During the execution of the read a cylinder command, if the starting sector cannot be found, then this flag is set.
D1	Not Writable	NW	During execution of a write data, write deleted data, or format-a-cylinder command, if the FDC detects a write-protect signal from the FDD, then this flag is set.
D0	Missing Address Mark	MA	If the FDC cannot detect the ID address mark, this flag is set. Also, at the same time, the MD (missing address mark in the data field) of status register 2 is set.

Command Status Register 1

Bit			Description
No.	Name	Symbol	
D7	—	—	Not used. This bit is always 0 (low).
D6	Control Mark	CM	During execution of the read data or scan command, if the FDC encounters a sector that contains a deleted data address mark, this flag is set.
D5	Data Error in Data Field	DD	If the FDC detects a CRC error in the data, then this flag is set.
D4	Wrong Cylinder	WC	This bit is related to the ND bit, and when the contents of C on the medium are different from that stored in the ID register, this flag is set.
D3	Scan Equal Hit	SH	During execution of the scan command, if the condition of "equal" is satisfied, this flag is set.
D2	Scan Not Satisfied	SN	During execution of the scan command, if the FDC cannot find a sector on the cylinder that meets the condition, then this flag is set.
D1	Bad Cylinder	BC	This bit is related to the ND bit, and when the contents of C on the medium are different from that stored in the ID register, and the contents of C is FF, then this flag is set.
D0	Missing Address Mark in Data Field	MD	When data is read from the medium, if the FDC cannot find a data address mark or deleted data address mark, then this flag is set.

Command Status Register 2

Bit			Description
No.	Name	Symbol	
D7	Fault	FT	This bit is the status of the fault signal from the FDD.
D6	Write Protected	WP	This bit is the status of the write-protected signal from the FDD.
D5	Ready	RY	This bit is the status of the ready signal from the FDD.
D4	Track 0	T0	This bit is the status of the track 0 signal from the FDD.
D3	Two Side	TS	This bit is the status of the two-side signal from the FDD.
D2	Head Address	HD	This bit is the status of the side-select signal from the FDD.
D1	Unit Select 1	US 1	This bit is the status of the unit-select-1 signal from the FDD.
D0	Unit Select 0	US 0	This bit is the status of the unit-select-0 signal from the FDD.

Command Status Register 3

Programming Summary

FDC Data Register I/O Address Hex 3F5

FDC Main Status Register I/O Address Hex 3F4

Digital Output Register I/O Address Hex 3F2

Bit 0	Drive Select	00: DR #A 10: DR #C 01: DR #B 11: DR #D
1		
2	Not FDC Reset	
3	Enable INT & DMA Requests	
4	Drive A Motor Enable	
5	Drive B Motor Enable	
6	Drive C Motor Enable	
7	Drive D Motor Enable	

All bits cleared with channel reset.

DPC Registers

FDC Constants (in hex)

N:	02	GPL Format:	05
SC:	08	GPL R/W:	2A
HUT:	F	HLT:	01
SRT:	C	(6 ms track-to-track)	

Drive Constants

Head Load	35 ms
Head Settle	15 ms
Motor Start	250 ms

Comments

- Head loads with drive select, wait HD load time before R/W.
- Following access, wait HD settle time before R/W.
- Drive motors should be off when not in use. Only A or B and C or D may run simultaneously. Wait motor start time before R/W.
- Motor must be on for drive to be selected.
- Data errors can occur while using a home television as the system display. Placing the TV too close to the diskette area can cause this to occur. To correct the problem, move the TV away from, or to the opposite side of the system unit.

Interface

System I/O Channel Interface

All signals are TTL-compatible:

Most Positive Up Level	+ 5.5 Vdc
Least Positive Up Level	+ 2.7 Vdc
Most Positive Down Level	+ 0.5 Vdc
Least Positive Down Level	− 0.5 Vdc

The following lines are used by this adapter.

+D0–7 (Bidirectional, Load: 1 74LS, Driver: 74LS 3-state): These eight lines form a bus through which all commands, status, and data are transferred. Bit 0 is the low-order bit.

+A0–9 (Adapter input, Load: 1 74LS): These 10 lines form an address bus by which a register is selected to receive or supply the byte transferred through lines D0–7. Bit 0 is the low-order bit.

+AEN (Adapter input, load: 1 74LS): The content of lines A0–9 is ignored if this line is active.

-IOW (Adapter input, Load: 1 74LS): The content of lines D0–7 is stored in the register addressed by lines A0–9 or DACK2 at the trailing edge of this signal.

-IOR (Adapter input, Load: 1 74LS): The content of the register addressed by lines A0–9 or DACK2 is gated onto lines D0–7 when this line is active.

-DACK2 (Adapter input, load: 2 74LS): This line being active degates output DRQ2, selects the FDC data register as the source or destination of bus D0–7, and indirectly gates T/C to IRQ6.

+T/C (Adapter input, load: 4 74LS): This line along with DACK2 being active indicates that the byte of data for which the DMA count was initialized is now being transferred.

+RESET (Adapter input, load: 1 74LS): An up level ends any operation in process and clears the digital output register (DOR).

+DRQ2 (Adapter output, driver: 74LS 3-state): This line is made active when the attachment is ready to transfer a byte of data to or from main storage. The line is made inactive by DACK2 becoming active or an I/O read of the FDC data register.

+IRQ6 (Adapter output, driver: 74LS 3-state): This line is made active when the FDC has completed an operation. It results in an interrupt to a routine that should examine the FDC result bytes to reset the line and determine the ending condition.

Drive A and B Interface

All signals are TTL-compatible:

Most Positive Up Level	+ 5.5 Vdc
Least Positive Up Level	+ 2.4 Vdc
Most Positive Down Level	+ 0.4 Vdc
Least Positive Down Level	− 0.5 Vdc

All adapter outputs are driven by open-collector gates. The drives must provide termination networks to Vcc (except 'motor enable', which has a 2,000-ohm resistor to Vcc).

Each adapter input is terminated with a 150-ohm resistor to Vcc.

Adapter Outputs

-Drive Select A and B

(Driver: 7438): These two lines are used by drives A and B to degate all drivers to the adapter and receivers from the attachment (except 'motor enable') when the line associated with a drive is inactive.

-Motor Enable A and B

(Driver: 7438): The drive associated with each of these lines must control its spindle motor such that it starts when the line becomes active and stops when the line becomes inactive.

-Step

(Driver: 7438): The selected drive moves the read/write head one cylinder in or out per the direction line for each pulse present on this line.

-Direction

(Driver: 7438): For each recognized pulse of the 'step' line, the read/write head moves one cylinder toward the spindle if this line is active, and away from the spindle if inactive.

-Head Select

(Driver: 7438): Head 1 (upper head) will be selected when this line is active (low).

-Write Data

(Driver: 7438): For each inactive-to-active transition of this line while 'write enable' is active, the selected drive causes a flux change to be stored on the diskette.

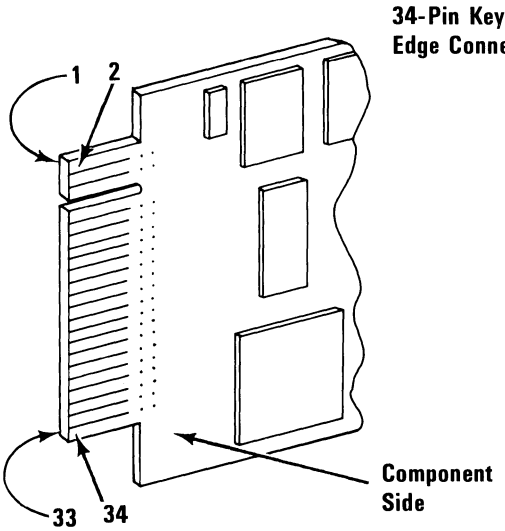
-Write Enable

(Driver: 7348): The drive disables write current in the head unless this line is active.

Adapter Inputs

-Index	The selected drive must supply one pulse per diskette revolution on this line.
-Write Protect	The selected drive must make this line active if a write-protected diskette is in the drive.
-Track 0	The selected drive must make this line active if the read/write head is over track 0.
-Read Data	The selected drive supplies a pulse on this line for each flux change encountered on the diskette.

Specifications



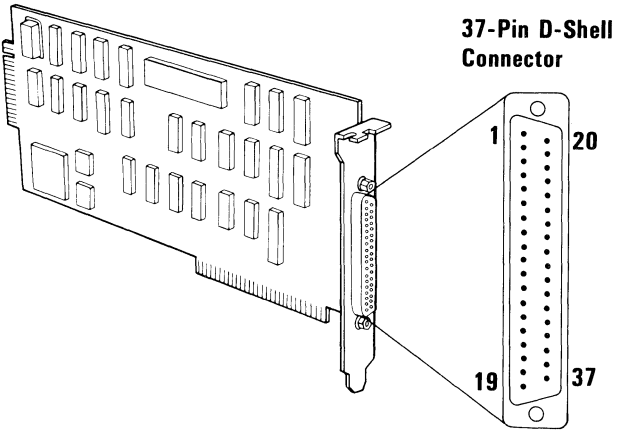
Note: Lands 1-33 (odd numbers) are on the back of the board. Lands 2-34 (even numbers) are on the front, or component side.

	At Standard TTL Levels	Land Number	
	Ground-Odd Numbers	1-33	
	Unused	2,4,6	
	Index	8	→
←	Motor Enable A	10	
←	Drive Select B	12	
←	Drive Select A	14	
←	Motor Enable B	16	
←	Direction (Stepper Motor)	18	
←	Step Pulse	20	
←	Write Data	22	
←	Write Enable	24	
←	Track 0	26	→
←	Write Protect	28	→
←	Read Data	30	→
←	Select Head 1	32	
	Unused	34	

Diskette Drives

Drive Adapter

Connector Specifications (Part 1 of 2)

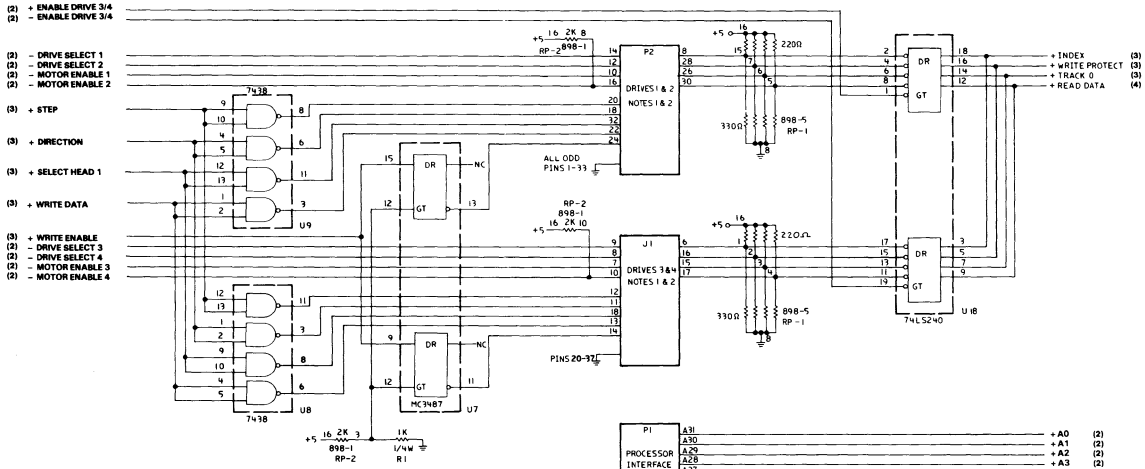


	At Standard TTL Levels	Pin Number	
	Unused	1-5	
	Index	6	→
←	Motor Enable C	7	
←	Drive Select D	8	
←	Drive Select C	9	
←	Motor Enable D	10	
←	Direction (Stepper Motor)	11	
←	Step Pulse	12	
←	Write Data	13	
←	Write Enable	14	
←	Track 0	15	→
	Write Protect	16	→
	Read Data	17	→
←	Select Head 1	18	
	Ground	20-37	

External Drives

Drive Adapter

Connector Specifications (Part 2 of 2)



NOTES:

1 SIGNALS ON DRIVE PINS 10 THRU 16 ARE SWAPPED BY THE DRIVE CABLE BETWEEN DRIVES 1 & 2 (AND 3 & 4) AS FOLLOWS:

10	16
11	15
12	14
13	13
14	12
15	11
16	10

2 ALL DRIVES ARE JUMPERED FOR MULTIPLEX OPERATION. HEAD LOAD WITH DRIVE SELECT AND DRIVE SELECT VIA INPUT PIN 12. TERMINATING R-PACS ARE LEFT IN DRIVES 1 & 3 ONLY

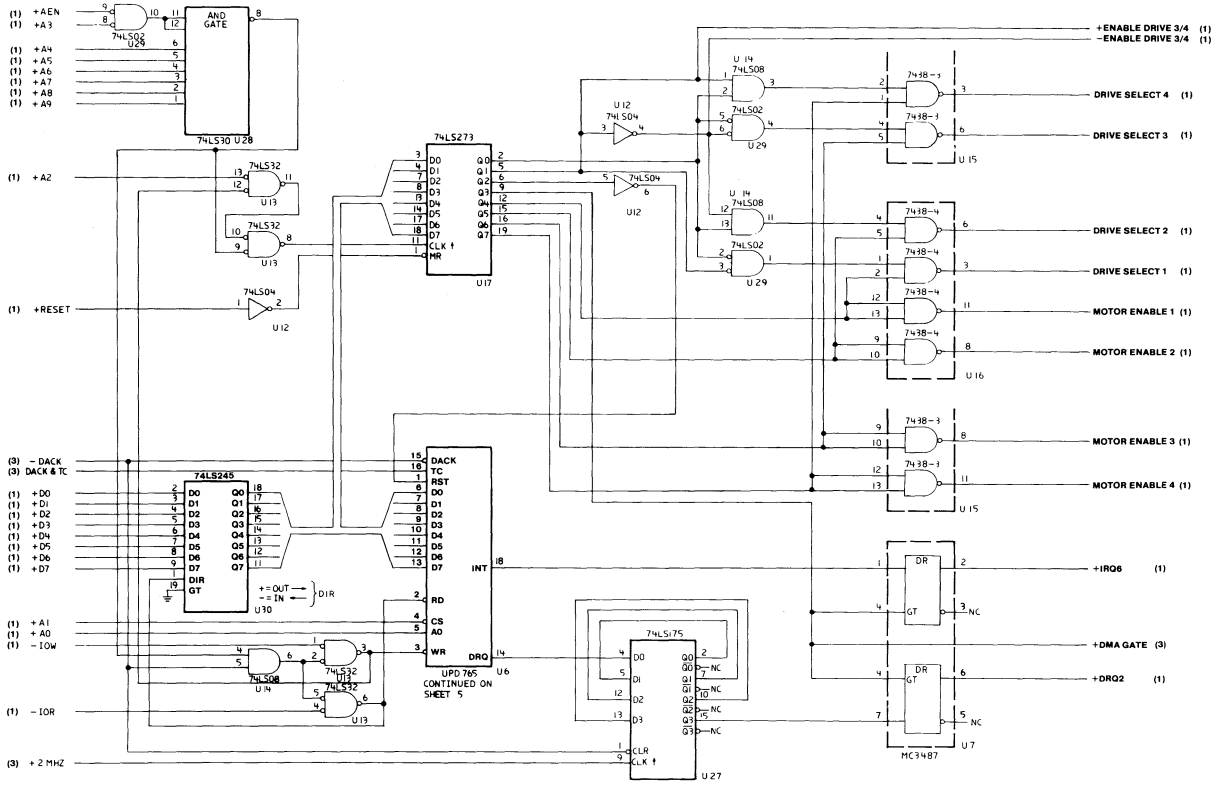
3 .047 μFD SHOULD BE ADJACENT TO MODULES MC3487, 7438, 74S74, 16 MHZ OSC. RP-1, MC4044, MC4024, 74LS161 & 74LS191 8.2μFD CAPS SHOULD BE NEAR ASSOCIATED PI PINS

4 ALL SIGNAL LINES HIGHER THAN OR EQUAL TO 1MHZ SHOULD BE KEPT TO THE SHORTEST POSSIBLE LENGTH. THIS IS A PRIMARY DESIGN GOAL

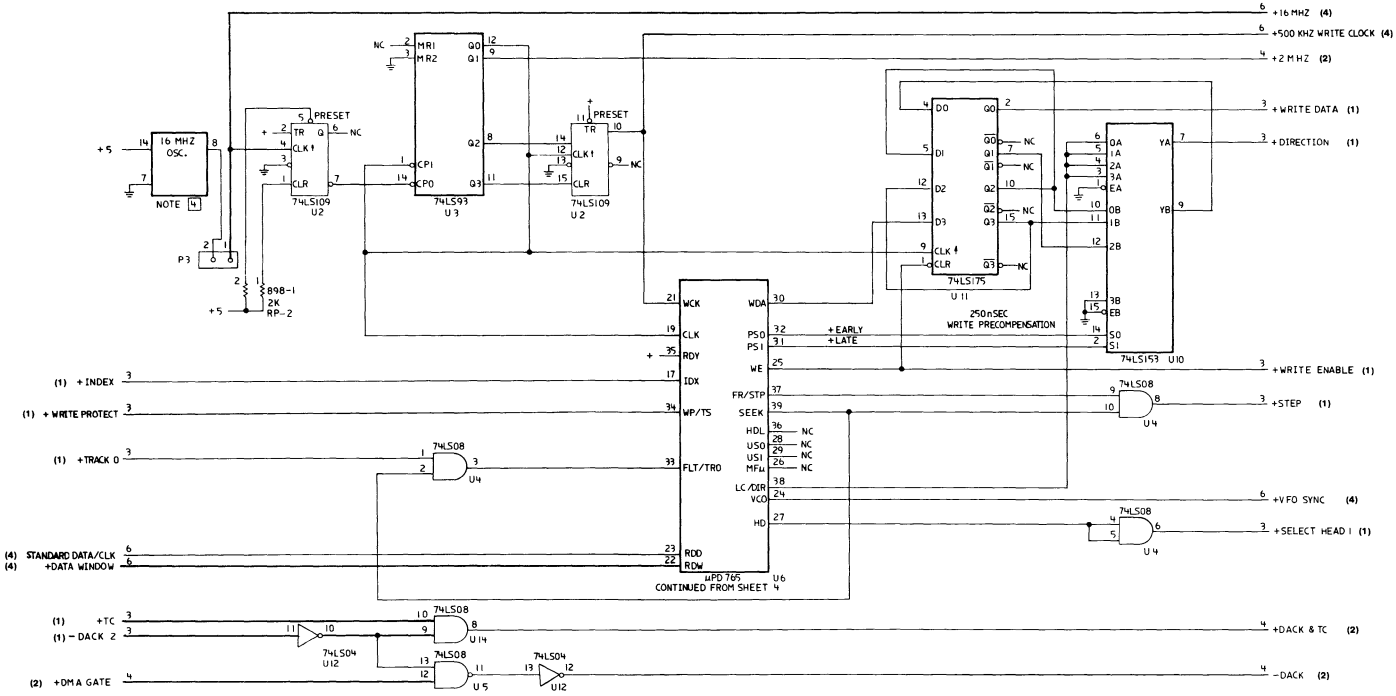
5 MAKE NO CONNECTION TO UNUSED PINS ON THE VCO, CHARGE PUMP & DATA SEPARATOR MODULES

6 ALL VOLTAGE AND GROUND CONNECTIONS TO THE VCO, CHARGE PUMP AND ASSOCIATED DISCRETE COMPONENTS SHOULD BE SEPARATE FROM OTHER CIRCUITS AND THEN JOINED TO THE OTHER CIRCUITS AT ONE POINT

5-1/4 Inch Diskette Drive Adapter (Sheet 1 of 4)

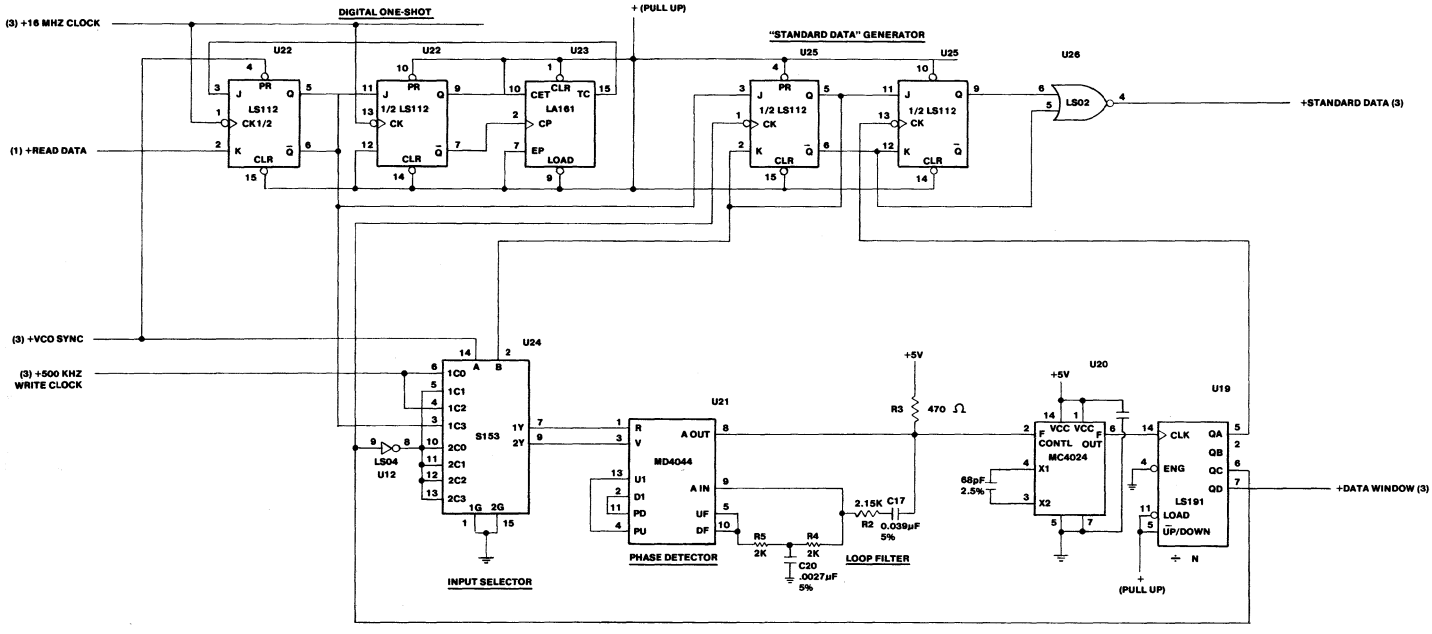


5-1/4 Inch Diskette Drive Adapter (Sheet 2 of 4)



NOTE: U4 (74LS08) PINS 12 AND 13 ARE CONNECTED ONLY ON CARDS BUILT USING RAW CARD P/N 5001293

5-1/4 Inch Diskette Drive Adapter (Sheet 3 of 4)



5-1/4 Inch Diskette Drive Adapter (Sheet 4 of 4)

Personal Computer Hardware Reference Library

IBM Fixed Disk Adapter

6361503

IBM Fixed Disk Adapter

Contents

Description

The Fixed Disk Adapter attaches to one or two fixed disk drive units through an internal, daisy-chained, flat cable (data/control cable). Each system supports a maximum of one Fixed Disk Adapter and two fixed disk drives.

The adapter is buffered on the I/O bus and uses the system board's direct memory access (DMA) for record data transfers. An interrupt level also is used to indicate operation completion and status conditions that require microprocessor attention.

The Fixed Disk Adapter provides automatic 11-bit burst error detection and correction in the form of 32-bit error checking and correction (ECC).

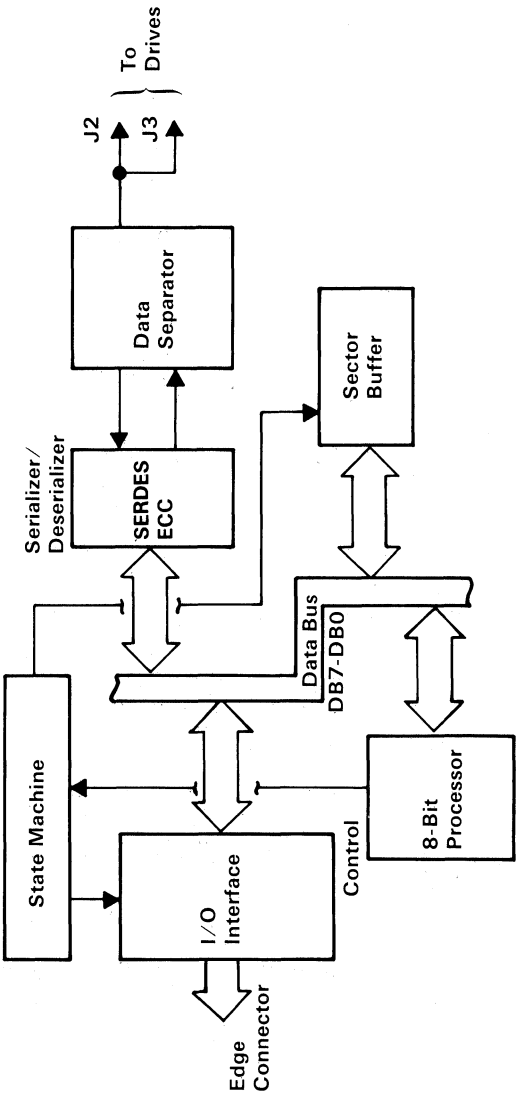
The device level control for the Fixed Disk Adapter is contained on a ROM module on the adapter. A listing of this device level control can be found in "BIOS Listing" of this section.

> **Warning:** The last cylinder on the fixed disk drive is reserved for diagnostic use. The diagnostic write test will destroy any data on this cylinder.

Fixed Disk Controller

The disk controller has two registers that may be accessed by the system unit's microprocessor: a status register and a data register. The 8-bit status register contains the status information of the disk controller, and can be accessed at any time. The 8-bit data register (actually consisting of several registers in a stack with only one register presented to the data bus) stores data, commands, and parameters, and provides the disk controller's status 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 status register is a read-only register that is used to help the transfer of data between the system unit's microprocessor and the disk controller. The controller-select pulse is generated by writing to port address hex 322.

The following is a block diagram of the IBM Fixed Disk Adapter.



Fixed Disk Adapter Block Diagram

Programming Considerations

Status Register

At the end of all commands from the system board, the disk controller sends a completion status byte to the system board. This byte informs the system unit's microprocessor if an error occurred during the execution of the command. The following shows the format of this byte.

Bit	7	6	5	4	3	2	1	0
	0	0	d	0	0	0	e	0

Bits 0, 1, 2, 3, 4, 6, 7 These bits are set to zero.

Bit 1 When set, this bit shows an error has occurred during command execution.

Bit 5 This bit shows the logical unit number of the drive.

If the interrupts are enabled, the controller sends an interrupt when it is ready to transfer the status byte. Busy from the disk controller is unasserted when the byte is transferred to complete the command.

Sense Bytes

If the status register receives an error (bit 1 set), the disk controller requests four bytes of sense data. The format for the four bytes is as follows:

Bits	7	6	5	4	3	2	1	0
Byte 0	Address Valid	0	Error Type		Error Code			
Byte 1	0	0	d	Head Number				
Byte 2	Cylinder High			Sector Number				
Byte 3	Cylinder Low							

Remarks
d = drive

Byte 0 **Bits 0, 1, 2, 3** Error code.

Byte 0 **Bits 4, 5** Error type.

Byte 0 **Bit 6** Set to 0 (spare)

Byte 0 **Bit 7** The address-valid bit. Set only when the previous command required a disk address, in which case it is returned as a 1; otherwise, it is 0.

Disk Controller Error Tables

The following disk controller error tables list the error types and error codes found in byte 0:

	Error Type		Error Code				
Bits	5	4	3	2	1	0	Description
	0	0	0	0	0	0	The controller did not detect any error during the execution of the previous operation.
	0	0	0	0	0	1	The controller did not detect an index signal from the drive.
	0	0	0	0	1	0	The controller did not get a seek-complete signal from the drive after a seek operation (for all non-buffered step seeks).
	0	0	0	0	1	1	The controller detected a write fault from the drive during the last operation.
	0	0	0	1	0	0	After the controller selected the drive, the drive did not respond with a ready signal.
	0	0	0	1	0	1	Not used.
	0	0	0	1	1	0	After stepping the maximum number of cylinders, the controller did not receive the track 00 signal from the drive.
	0	0	0	1	1	1	Not used.
	0	0	1	0	0	0	The drive is still seeking. This status is reported by the Test Drive Ready command for an overlap seek condition when the drive has not completed the seek. No time-out is measured by the controller for the seek to complete.

	Error Type		Error Code				
Bits	5	4	3	2	1	0	Description
	0	**1**	**0**	**0**	**0**	**0**	ID Read Error: The controller detected an ECC error in the target ID field on the disk.
	0	1	0	0	0	1	Data Error: The controller detected an uncorrectable ECC error in the target sector during a read operation.
	0	1	0	0	1	0	Address Mark: The controller did not detect the target address mark (AM) on the disk.
	0	1	0	0	1	1	Not used.
	0	1	0	1	0	0	Sector Not Found: The controller found the correct cylinder and head, but not the target sector.
	0	1	0	1	0	1	Seek Error: The cylinder or head address (either or both) did not compare with the expected target address as a result of a seek.
	0	1	0	1	1	0	Not used.
	0	1	0	1	1	1	Not used.
	0	1	1	0	0	0	Correctable Data Error: The controller detected a correctable ECC error in the target field.
	0	1	1	0	0	1	Bad Track: The controller detected a bad track flag during the last operation. No retries are attempted on this error.

	Error Type		Error Code				
Bits	5	4	3	2	1	0	Description
	1	0	0	0	0	0	Invalid Command: The controller has received an invalid command from the system unit.
	1	0	0	0	0	1	Illegal Disk Address. The controller detected an address that is beyond the maximum range.

	Error Type		Error Code				
Bits	5	4	3	2	1	0	Description
	1	1	0	0	0	0	RAM Error: The controller detected a data error during the RAM sector-buffer diagnostic test.
	1	1	0	0	0	1	Program Memory Checksum Error: During this internal diagnostic test, the controller detected a program-memory checksum error.
	1	1	0	0	1	0	ECC Polynominal Error: During the controller's internal diagnostic tests, the hardware ECC generator failed its test.

Data Register

The system unit's microprocessor specifies the operation by sending the 6-byte device control block (DCB) to the controller. The figure below shows the composition of the DCB, and defines the bytes that make up the DCB.

Bit	7	6	5	4	3	2	1	0
Byte 0	Command Class			Opcode				
Byte 1	0	0	d	Head Number				
Byte 2	Cylinder High			Sector Number				
Byte 3	Cylinder Low							
Byte 4	Interleave or Block Count							
Byte 5	Control Field							

Byte 0 Bits 7, 6, and 5 identify the class of the command. Bits 4 through 0 contain the Opcode command.

Byte 1 Bit 5 identifies the drive number. Bits 4 through 0 contain the disk head number to be selected. Bits 6 and 7 are not used.

Byte 2 Bits 6 and 7 contain the two most significant bits of the cylinder number. Bits 0 through 5 contain the sector number.

Byte 3 Bits 0 through 7 are the eight least-significant bits of the cylinder number.

Byte 4 Bits 0 through 7 specify the interleave or block count.

Byte 5 Bits 0 through 7 contain the control field.

Control Byte

Byte 5 is the control field of the DCB and allows the user to select options for several types of disk drives. The format of this byte is as follows:

Bits	7	6	5	4	3	2	1	0
	r	a	0	0	0	s	s	s

Remarks

r = retries
s = step option
a = retry option on data ECC error

Bit 7 Disables the four retries by the controller on all disk-access commands. Set this bit only during the evaluation of the performance of a disk drive.

Bit 6 If set to 0 during read commands, a reread is attempted when an ECC error occurs. If no error occurs during reread, the command will finish without an error status. If this bit is set to 1, no reread is attempted.

Bits 5, 4, 3 Set to 0.

Bits 2, 1, 0 These bits define the type of drive and select the step option. See the following figure.

Bits 2, 1, 0	
0 0 0	This drive is not specified and defaults to 3 milliseconds per step
0 0 1	N/A
0 1 0	N/A
0 1 1	N/A
1 0 0	200 microseconds per step.
1 0 1	70 microseconds per step (specified by BIOS).
1 1 0	3 milliseconds per step.
1 1 1	3 milliseconds per step.

Command Summary

Test Drive Ready (Class 0, Opcode 00)

Bit	7	6	5	4	3	2	1	0
Byte 0	0	0	0	0	0	0	0	0
Byte 1	0	0	d	x	x	x	x	x

d = drive (0 or 1)
x = don't care
Bytes 2, 3, 4, 5 = don't care

Recalibrate (Class 0, Opcode 01)

Bit	7	6	5	4	3	2	1	0
Byte 0	0	0	0	0	0	0	0	1
Byte 1	0	0	d	x	x	x	x	x
Byte 5	r	0	0	0	0	s	s	s

d = drive (0 or 1)
x = don't care
r = retries
s = Step Option
Bytes 2, 3, 4 = don't care
ch = cylinder high

Reserved (Class 0, Opcode 02)

This Opcode is not used.

Request Sense Status (Class 0, Opcode 03)

Bit	7	6	5	4	3	2	1	0
Byte 0	0	0	0	0	0	0	1	1
Byte 1	0	0	d	x	x	x	x	x

d = drive (0 or 1)
x = don't care
Bytes 2, 3, 4, 5 = don't care

Format Drive (Class 0, Opcode 04)

Bit	7	6	5	4	3	2	1	0
Byte 0	0	0	0	0	0	1	0	0
Byte 1	0	0	d	Head Number				
Byte 2	ch		0	0	0	0	0	0
Byte 3	Cylinder Low							
Byte 4	0	0	0	Interleave				
Byte 5	r	0	0	0	0	s	s	s

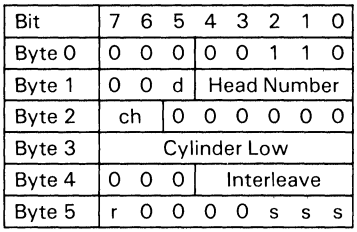
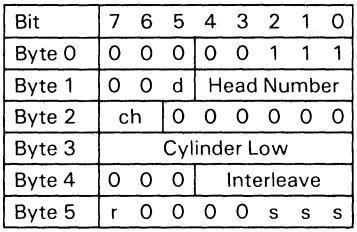
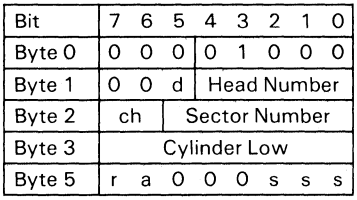
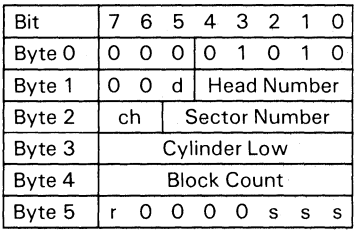
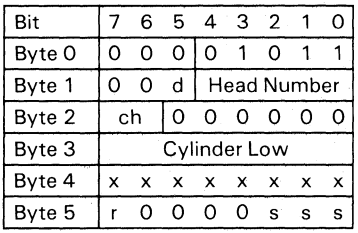
d = drive (0 or 1)
r = retries
s = step option
ch = cylinder high

Interleave 1 to 16 for 512-byte sectors.

Ready Verify (Class 0, Opcode 05)

Bit	7	6	5	4	3	2	1	0
Byte 0	0	0	0	0	0	1	0	1
Byte 1	0	0	d	Head Number				
Byte 2	ch		Sector Number					
Byte 3	Cylinder Low							
Byte 4	Block Count							
Byte 5	r	a	0	0	0	s	s	s

d = drive (0 or 1)
r = retries
s = step option
a = retry option on data ECC
ch = cylinder high

Command	Data Control Block	Remarks
Format Track (Class 0, Opcode 06)		d = drive (0 or 1) r = retries s = step option ch = cylinder high Interleave 1 to 16 for 512-byte sectors.
Format Bad Track (Class 0, Opcode 07)		d = drive (0 or 1) r = retries s = step option ch = cylinder high Interleave 1 to 16 for 512-byte sectors.
Read (Class 0, Opcode 08)		d = drive (0 or 1) r = retries a = retry option on data ECC error s = step option ch = cylinder high
Reserved (Class 0, Opcode 09)		This Opcode is not used.
Write (Class 0, Opcode 0A)		d = drive (0 or 1) r = retries s = step option ch = cylinder high
Seek (Class 0, Opcode 0B)		d = drive (0 or 1) r = retries s = step option x = don't care ch = cylinder high

Command	Data Control Block	Remarks
Initialize Drive Characteristics* (Class 0, Opcode 0C)	Bit 7 6 5 4 3 2 1 0 Byte 0 0 0 0 0 1 1 0 0	Bytes 1, 2, 3, 4, 5, = don't care
Read ECC Burst Error Length (Class 0, Opcode 0D)	Bit 7 6 5 4 3 2 1 0 Byte 0 0 0 0 0 1 1 0 1	Bytes 1, 2, 3, 4, 5, = don't care
Read Data from Sector Buffer (Class 0, Opcode 0E)	Bit 7 6 5 4 3 2 1 0 Byte 0 0 0 0 0 1 1 1 0	Bytes 1, 2, 3, 4, 5, = don't care
Write Data to Sector Buffer (Class 0, Opcode 0F)	Bit 7 6 5 4 3 2 1 0 Byte 0 0 0 0 0 1 1 1 1	Bytes 1, 2, 3, 4, 5, = don't care
RAM Diagnostic (Class 7, Opcode 00)	Bit 7 6 5 4 3 2 1 0 Byte 0 1 1 1 0 0 0 0 0	Bytes 1, 2, 3, 4, 5, = don't care
Reserved (Class 7, Opcode 01)		This Opcode is not used.
Reserved (Class 7, Opcode 02)		This Opcode is not used.

*Initialize Drive Characteristics: The DBC must be followed by eight additional bytes.

Maximum number of cylinders	(2 bytes)
Maximum number of heads	(1 byte)
Start reduced write current cylinder	(2 bytes)
Start write precompensation cylinder	(2 bytes)
Maximum ECC data burst length	(1 byte)

Drive Diagnostic (Class 7, Opcode 03)

Bit	7	6	5	4	3	2	1	0
Byte 0	1	1	1	0	0	0	1	1
Byte 1	0	0	d	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	x	x	x	x	x	x	x	x
Byte 5	r	0	0	0	0	s	s	s

d = drive (0 or 1)
s = step option
r = retries
x = don't care

Controller Internal Diagnostics (Class 7, Opcode 04)

Bit	7	6	5	4	3	2	1	0
Byte 0	1	1	1	0	0	1	0	0

Bytes 1, 2, 3, 4, 5, = don't care

Read Long* (Class 7, Opcode 05)

Bit	7	6	5	4	3	2	1	0
Byte 0	1	1	1	0	0	1	0	1
Byte 1	0	0	d	Head Number				
Byte 2	ch			Sector Number				
Byte 3	Cylinder Low							
Byte 4	Block Count							
Byte 5	r	0	0	0	0	s	s	s

d = drive (0 or 1)
s = step option
r = retries
ch = cylinder high

Write Long (Class 7, Opcode 06)**

Bit	7	6	5	4	3	2	1	0
Byte 0	1	1	1	0	0	1	1	0
Byte 1	0	0	d	Head Number				
Byte 2	ch			Sector Number				
Byte 3	Cylinder Low							
Byte 4	Block Count							
Byte 5	r	0	0	0	0	s	s	s

d = drive (0 or 1)
s = step option
r = retries
ch = cylinder high

*Returns 512 bytes plus 4 bytes of ECC data per sector.

**Requires 512 bytes plus 4 bytes of ECC data per sector.

Programming Summary

The two least-significant bits of the address bus are sent to the system board's I/O port decoder, which has two sections. One section is enabled by the I/O read signal (-IOR) and the other by the I/O write signal (-IOW). The result is a total of four read/write ports assigned to the disk controller board.

The address enable signal (AEN) is asserted by the system board when DMA is controlling data transfer. When AEN is asserted, the I/O port decoder is disabled.

The following figure is a table of the read/write ports.

R/W	Port Address	Function
Read Write	320 320	Read data (from controller to system unit). Write data (from system unit to controller).
Read Write	321 321	Read controller hardware status. Controller reset.
Read Write	322 322	Reserved. Generate controller-select pulse.
Read Write	323 323	Not used. Write pattern to DMA and interrupt mask register.

Interface

The following lines are used by the disk controller:

A0-A19 Positive true 20-bit address. The least-significant 10 bits contain the I/O address within the range of hex 320 to hex 323 when an I/O read or write is executed by the system unit. The full 20 bits are decoded to address the read-only memory (ROM) between the addresses of hex C8000 and C9FFF.

DO-D7 Positive 8-bit data bus over which data and status information is passed between the system board and the controller.

-IOR Negative true signal that is asserted when the system board reads status or data from the controller under either programmed I/O or DMA control.

-IOW Negative true signal that is asserted when the system board sends a command or data to the controller under either programmed I/O or DMA control.

AEN Positive true signal that is asserted when the DMA in the system board is generating the I/O Read (-IOR) or I/O Write (-IOW) signals and has control of the address and data buses.

RESET Positive true signal that forces the disk controller to its initial power-up condition.

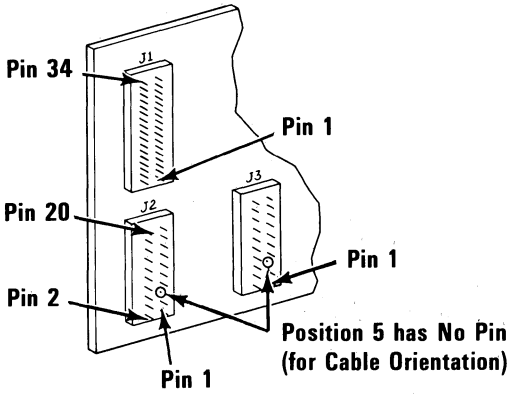
IRQ 5 Positive true interrupt-request signal that is asserted by the controller when enabled to interrupt the system board on the return ending status byte from the controller.

DRQ 3 Positive true DMA-request signal that is asserted by the controller when data is available for transfer to or from the controller under DMA control. This signal remains active until the system board's DMA channel activates the DMA-acknowledge signal (-DACK 3) in response.

-DACK 3 This signal is true when negative, and is generated by the system board DMA channel in response to a DMA request (DRQ 3).

Specifications

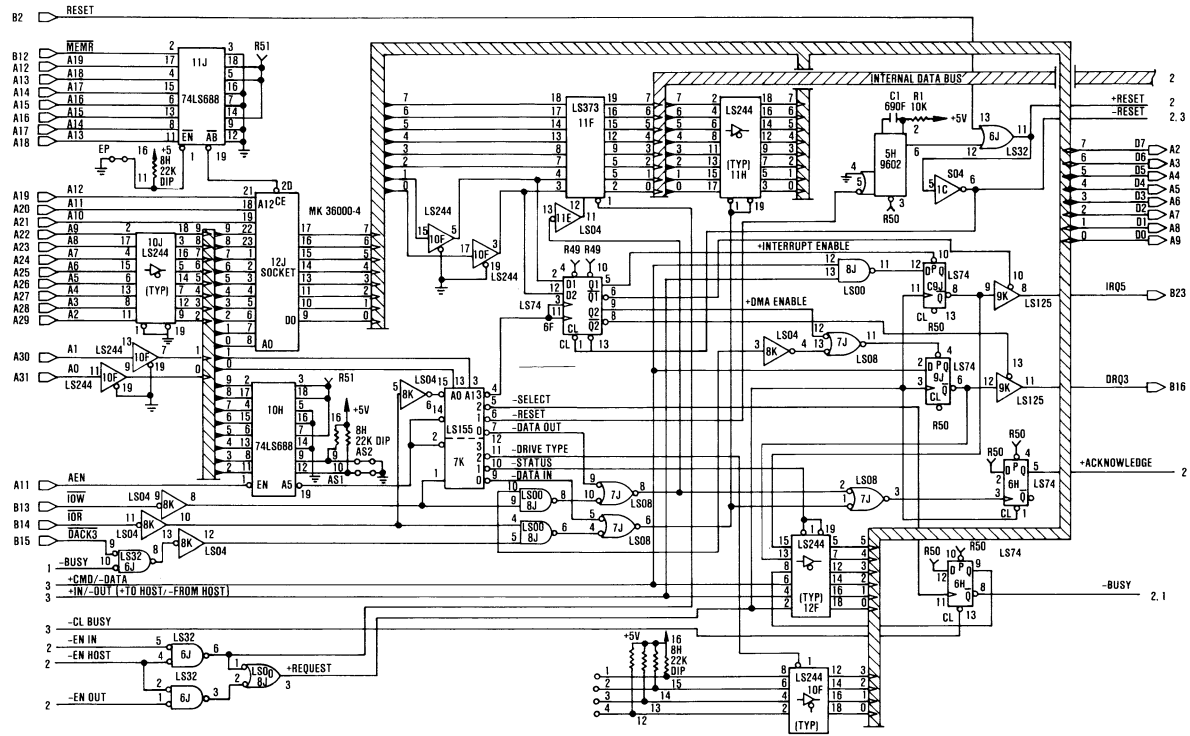
The Fixed Disk Adapter connector and interface specifications follow.



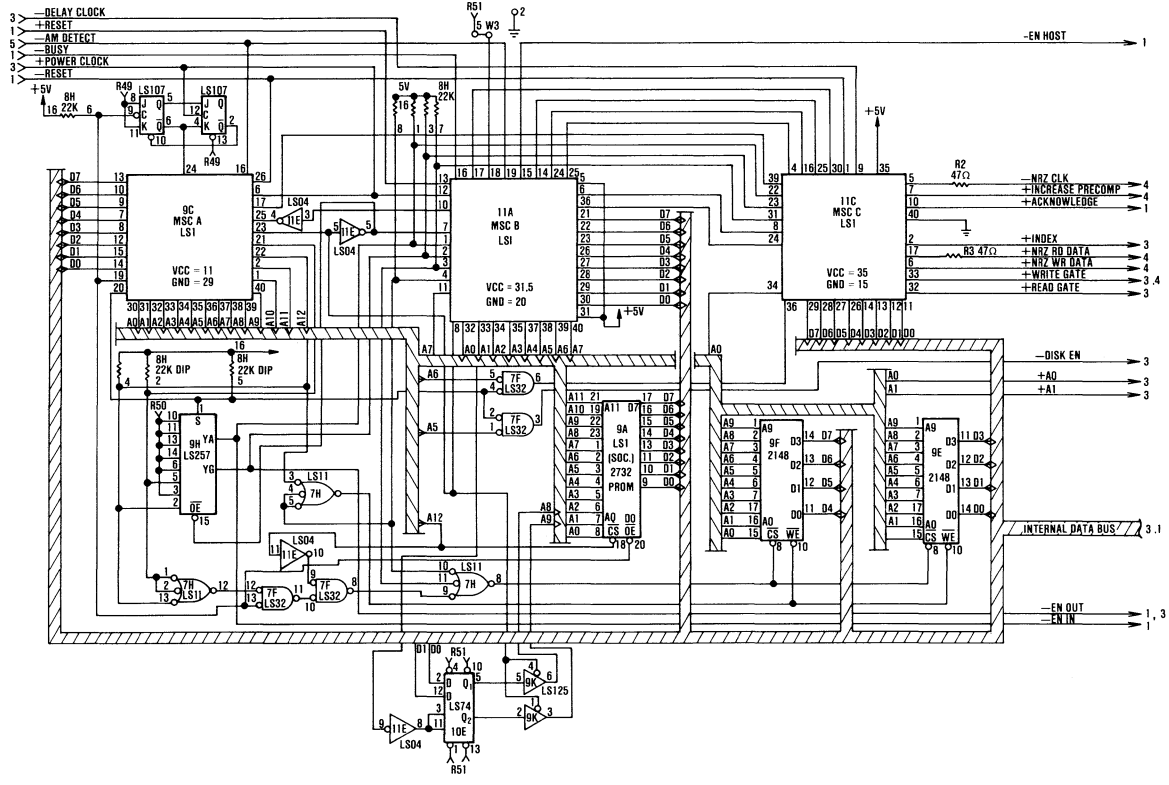
	Signal	Pin Number	
Disk Drive Connector J1	Ground-Odd Numbers	1-33	Disk Adapter Connector J1
	Reserved	4,16,30,32	
	← – Reduced Write Current	2	
	← – Write Gate	6	
	← – Seek Complete	8	
	→ – Track 00	10	
	→ – Write Fault	12	
	← – Head Select 2^0	14	
	← – Head Select 2^1	18	
	→ – Index	20	
	→ – Ready	22	
	← – Step	24	
	← – Drive Select 1	26	
	← – Drive Select 2	28	
	← – Direction In	34	

	Signal	Pin Number	
Disk Drive Connector J2 or J3	Ground	2,4,6,8,12,16,20	Disk Adapter Connector J2 or J3
	Drive Select	1 →	
	Reserved	3,7	
	Spare	9,10,5 (No Pin)	
	Ground	11	
	MFM Wire Data	13	
	← – MTM Write Data	14	
	Ground	15	
	MFM Read Data	17 →	
	→ – MFM Read Data	18	
	Ground	19	

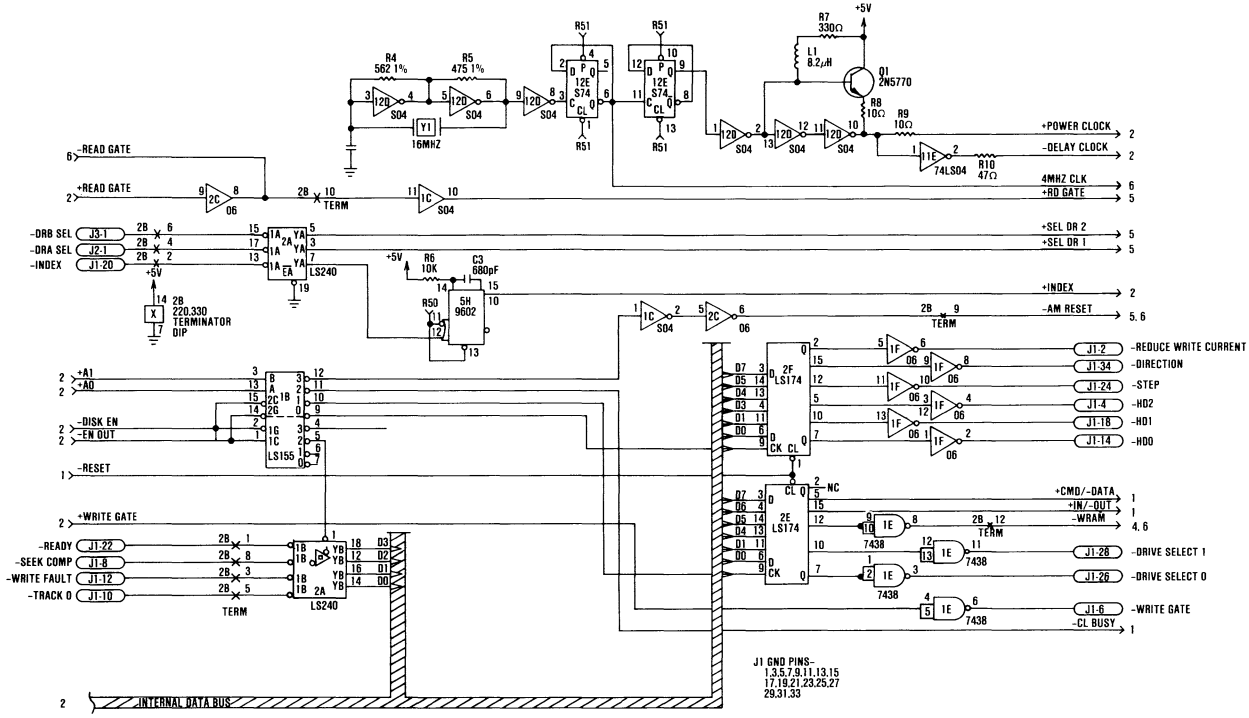
Fixed Disk Adapter Interface Specifications



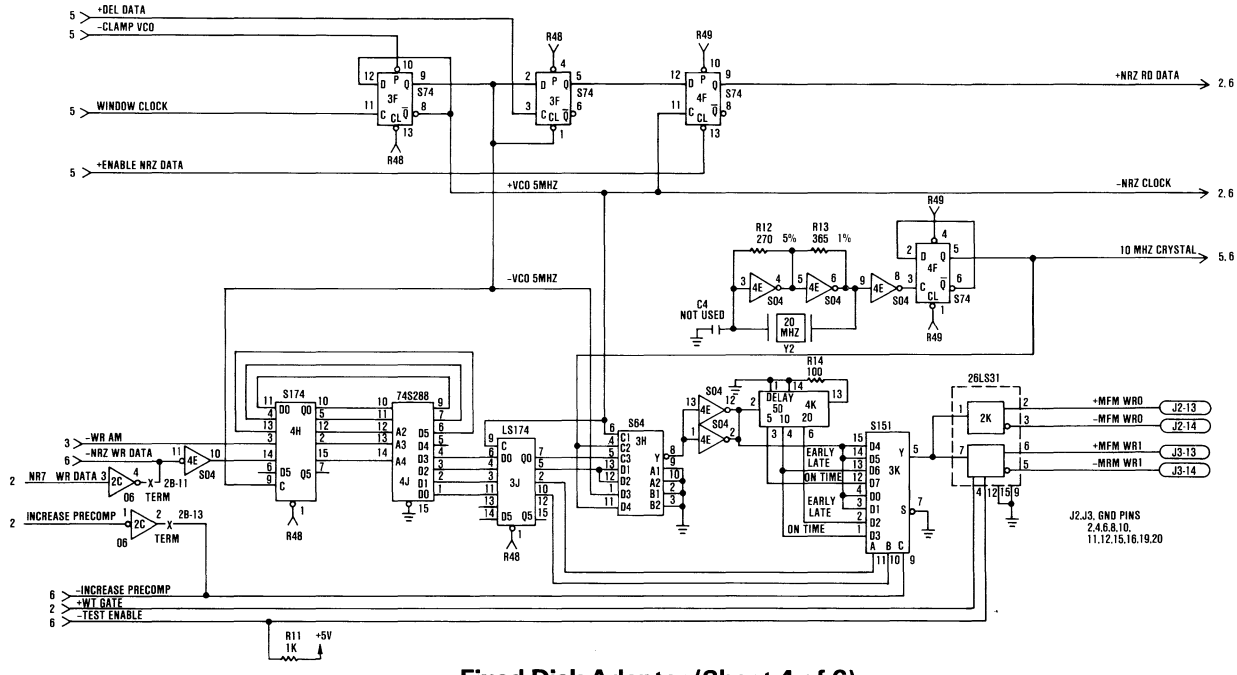
Fixed Disk Adapter (Sheet 1 of 6)



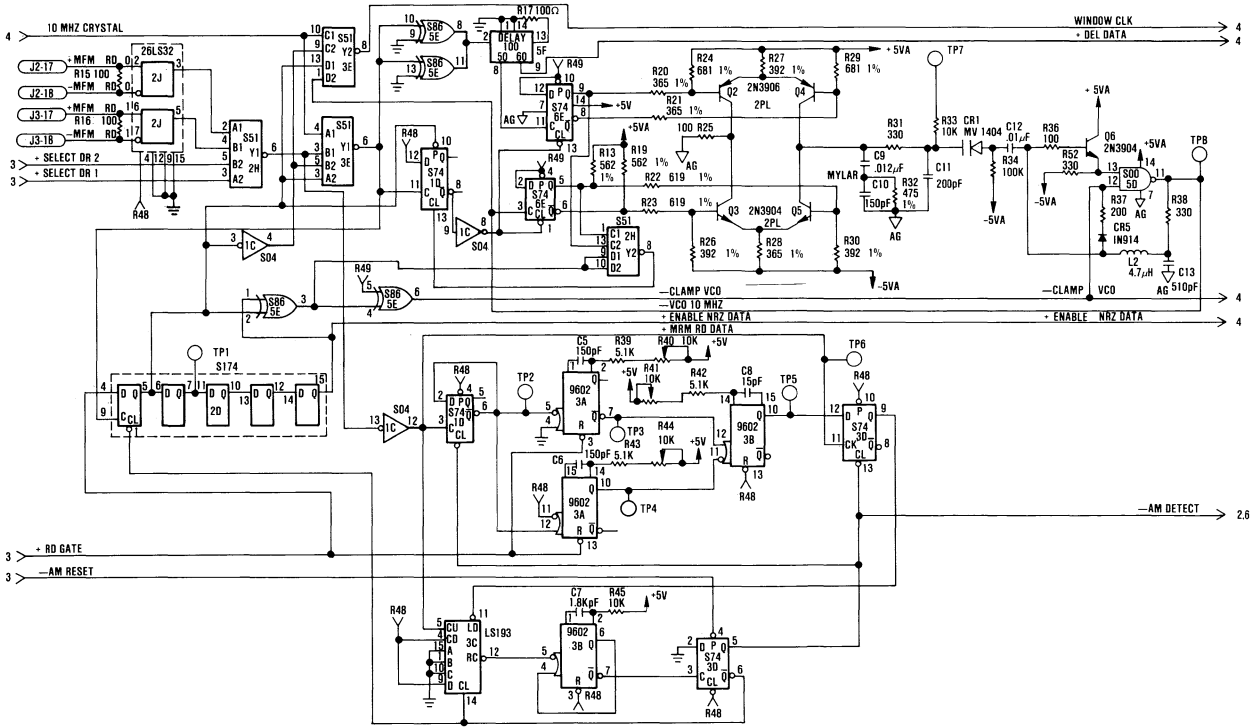
Fixed Disk Adapter (Sheet 2 of 6)



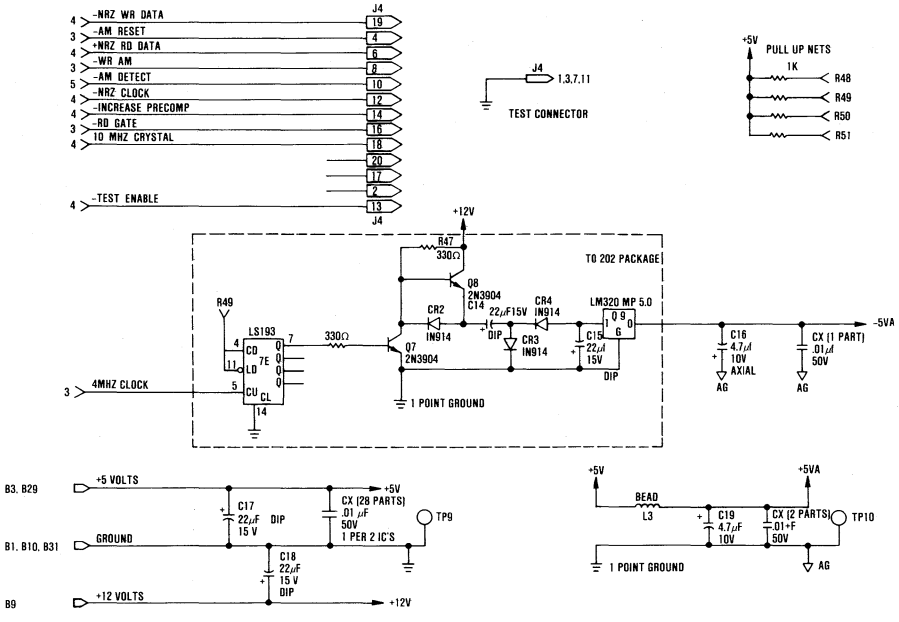
Fixed Disk Adapter (Sheet 3 of 6)



Fixed Disk Adapter (Sheet 4 of 6)



Fixed Disk Adapter (Sheet 5 of 6)



NOTES:
UNLESS OTHERWISE SPECIFIED:
1. ALL RESISTORS 1/4 W, 5%, CARBON FILTER.
2. ALL CAPS +10V OR GREATER +10%.
3. NO MORE THAN 15 LOADS PER PULLUP NET.

Fixed Disk Adapter (Sheet 6 of 6)

BIOS Listing

The BIOS Listing for the IBM Fixed Disk Adapter follows.

```
LOC OBJ                 LINE    SOURCE

                          1     $TITLE(FIXED DISK BIOS FOR IBM DISK CONTROLLER)
                          2
                          3     ;-- INT 13 ------------------------------------------------------
                          4     ;                                                               :
                          5     ; FIXED DISK I/O INTERFACE                                      :
                          6     ;                                                               :
                          7     ;       THIS INTERFACE PROVIDES ACCESS TO 5 1/4" FIXED DISKS    :
                          8     ;       THROUGH THE IBM FIXED DISK CONTROLLER.                  :
                          9     ;                                                               :
                         10     ;---------------------------------------------------------------
                         11
                         12     ;---------------------------------------------------------------
                         13     ;       THE  BIOS ROUTINES ARE MEANT TO BE ACCESSED THROUGH     :
                         14     ;       SOFTWARE INTERRUPTS ONLY.  ANY ADDRESSES PRESENT IN     :
                         15     ;       THE LISTINGS  ARE INCLUDED  ONLY FOR  COMPLETENESS,     :
                         16     ;       NOT FOR  REFERENCE.   APPLICATIONS WHICH  REFERENCE     :
                         17     ;       ABSOLUTE   ADDRESSES   WITHIN   THE   CODE  SEGMENT     :
                         18     ;       VIOLATE THE STRUCTURE AND DESIGN OF BIOS.               :
                         19     ;---------------------------------------------------------------
                         20     ;
                         21     ; INPUT    (AH = HEX VALUE)
                         22     ;
                         23     ;       (AH)=00 RESET DISK (DL = 80H,81H) / DISKETTE
                         24     ;       (AH)=01 READ THE STATUS OF THE LAST DISK OPERATION INTO (AL)
                         25     ;               NOTE: DL < 80H - DISKETTE
                         26     ;                     DL > 80H - DISK
                         27     ;       (AH)=02 READ THE DESIRED SECTORS INTO MEMORY
                         28     ;       (AH)=03 WRITE THE DESIRED SECTORS FROM MEMORY
                         29     ;       (AH)=04 VERIFY THE DESIRED SECTORS
                         30     ;       (AH)=05 FORMAT THE DESIRED TRACK
                         31     ;       (AH)=06 FORMAT THE DESIRED TRACK AND SET BAD SECTOR FLAGS
                         32     ;       (AH)=07 FORMAT THE DRIVE STARTING AT THE DESIRED TRACK
                         33     ;       (AH)=08 RETURN THE CURRENT DRIVE PARAMETERS
                         34     ;
                         35     ;       (AH)=09 INITIALIZE DRIVE PAIR CHARACTERISTICS
                         36     ;               INTERRUPT 41 POINTS TO DATA BLOCK
                         37     ;       (AH)=0A READ LONG
                         38     ;       (AH)=0B WRITE LONG
                         39     ;       NOTE: READ AND WRITE LONG ENCOMPASS 512 + 4 BYTES ECC
                         40     ;       (AH)=0C SEEK
                         41     ;       (AH)=0D ALTERNATE DISK RESET (SEE DL)
                         42     ;       (AH)=0E READ SECTOR BUFFER
                         43     ;       (AH)=0F WRITE SECTOR BUFFER,
                         44     ;               (RECOMMENDED PRACTICE BEFORE FORMATTING)
                         45     ;       (AH)=10 TEST DRIVE READY
                         46     ;       (AH)=11 RECALIBRATE
                         47     ;       (AH)=12 CONTROLLER RAM DIAGNOSTIC
                         48     ;       (AH)=13 DRIVE DIAGNOSTIC
                         49     ;       (AH)=14 CONTROLLER INTERNAL DIAGNOSTIC
                         50     ;
                         51     ;               REGISTERS USED FOR FIXED DISK OPERATIONS
                         52     ;
                         53     ;               (DL)    -  DRIVE NUMBER     (80H-87H FOR DISK, VALUE CHECKED)
                         54     ;               (DH)    -  HEAD NUMBER      (0-7 ALLOWED, NOT VALUE CHECKED)
                         55     ;               (CH)    -  CYLINDER NUMBER  (0-1023, NOT VALUE CHECKED)(SEE CL)
                         56     ;               (CL)    -  SECTOR NUMBER    (1-17, NOT VALUE CHECKED)
                         57     ;
                         58     ;                          NOTE: HIGH 2 BITS OF CYLINDER NUMBER ARE PLACED
                         59     ;                                IN THE HIGH 2 BITS OF THE CL REGISTER
                         60     ;                                (10 BITS TOTAL)
                         61     ;               (AL)    -  NUMBER OF SECTORS (MAXIMUM POSSIBLE RANGE 1-80H,
                         62     ;                                             FOR READ/WRITE LONG 1-79H)
                         63     ;                          (INTERLEAVE VALUE FOR FORMAT 1-16D)
                         64     ;               (ES:BX) -  ADDRESS OF BUFFER FOR READS AND WRITES,
                         65     ;                          (NOT REQUIRED FOR VERIFY)
                         66     ;
                         67     ; OUTPUT
                         68     ;       AH = STATUS OF CURRENT OPERATION
                         69     ;            STATUS BITS ARE DEFINED IN THE EQUATES BELOW
                         70     ;       CY = 0  SUCCESSFUL OPERATION (AH=0 ON RETURN)
                         71     ;       CY = 1  FAILED OPERATION (AH HAS ERROR REASON)
                         72     ;
                         73     ;       NOTE:   ERROR 11H  INDICATES THAT THE DATA READ HAD A RECOVERABLE
                         74     ;               ERROR WHICH WAS CORRECTED BY THE ECC ALGORITHM.  THE DATA
                         75     ;               IS PROBABLY GOOD,   HOWEVER THE BIOS ROUTINE INDICATES AN
                         76     ;               ERROR TO ALLOW THE CONTROLLING PROGRAM A CHANCE TO DECIDE
                         77     ;               FOR  ITSELF.  THE  ERROR  MAY  NOT  RECUR  IF THE DATA IS
```

```
LOC OBJ                 LINE    SOURCE

                         78     ;               REWRITTEN. (AL) CONTAINS THE BURST LENGTH.
                         79     ;
                         80     ;       IF DRIVE PARAMETERS WERE REQUESTED,
                         81     ;
                         82     ;       DL = NUMBER OF CONSECUTIVE ACKNOWLEDGING DRIVES ATTACHED (0-2)
                         83     ;               (CONTROLLER CARD ZERO TALLY ONLY)
                         84     ;       DH = MAXIMUM USEABLE VALUE FOR HEAD NUMBER
                         85     ;       CH = MAXIMUM USEABLE VALUE FOR CYLINDER NUMBER
                         86     ;       CL = MAXIMUM USEABLE VALUE FOR SECTOR NUMBER
                         87     ;            AND CYLINDER NUMBER HIGH BITS
                         88     ;
                         89     ;       REGISTERS WILL BE PRESERVED EXCEPT WHEN THEY ARE USED TO RETURN
                         90     ;       INFORMATION.
                         91     ;
                         92     ;       NOTE: IF AN ERROR IS REPORTED BY THE DISK CODE, THE APPROPRIATE
                         93     ;               ACTION IS TO RESET THE DISK, THEN RETRY THE OPERATION.
                         94     ;
                         95     ;--------------------------------------------------------------------------
                         96
  00FF                   97     SENSE_FAIL      EQU     0FFH            ; SENSE OPERATION FAILED
  00BB                   98     UNDEF_ERR       EQU     0BBH            ; UNDEFINED ERROR OCCURRED
  0080                   99     TIME_OUT        EQU     80H             ; ATTACHMENT FAILED TO RESPOND
  0040                  100     BAD_SEEK        EQU     40H             ; SEEK OPERATION FAILED
  0020                  101     BAD_CNTLR       EQU     20H             ; CONTROLLER HAS FAILED
  0011                  102     DATA_CORRECTED  EQU     11H             ; ECC CORRECTED DATA ERROR
  0010                  103     BAD_ECC         EQU     10H             ; BAD ECC ON DISK READ
  000B                  104     BAD_TRACK       EQU     0BH             ; BAD TRACK FLAG DETECTED
  0009                  105     DMA_BOUNDARY    EQU     09H             ; ATTEMPT TO DMA ACROSS 64K BOUNDARY
  0007                  106     INIT_FAIL       EQU     07H             ; DRIVE PARAMETER ACTIVITY FAILED
  0005                  107     BAD_RESET       EQU     05H             ; RESET FAILED
  0004                  108     RECORD_NOT_FND  EQU     04H             ; REQUESTED SECTOR NOT FOUND
  0002                  109     BAD_ADDR_MARK   EQU     02H             ; ADDRESS MARK NOT FOUND
  0001                  110     BAD_CMD         EQU     01H             ; BAD COMMAND PASSED TO DISK I/O
                        111
                        112     ;---------------------------------------
                        113     ;       INTERRUPT AND STATUS AREAS      :
                        114     ;---------------------------------------
                        115
  ----                  116     DUMMY   SEGMENT AT 0
  0034                  117             ORG     0DH*4                   ; FIXED DISK INTERRUPT VECTOR
  0034                  118     HDISK_INT       LABEL   DWORD
  004C                  119             ORG     13H*4                   ; DISK INTERRUPT VECTOR
  004C                  120     ORG_VECTOR      LABEL   DWORD
  0064                  121             ORG     19H*4                   ; BOOTSTRAP INTERRUPT VECTOR
  0064                  122     BOOT_VEC        LABEL   DWORD
  0078                  123             ORG     1EH*4                   ; DISKETTE PARAMETERS
  0078                  124     DISKETTE_PARM   LABEL   DWORD
  0100                  125             ORG     040H*4                  ; NEW DISKETTE INTERRUPT VECTOR
  0100                  126     DISK_VECTOR     LABEL   DWORD
  0104                  127             ORG     041H*4                  ; FIXED DISK PARAMETER VECTOR
  0104                  128     HF_TBL_VEC      LABEL   DWORD
  7C00                  129             ORG     7C00H                   ; BOOTSTRAP LOADER VECTOR
  7C00                  130     BOOT_LOCN       LABEL   FAR
  ----                  131     DUMMY   ENDS
                        132
  ----                  133     DATA    SEGMENT AT 40H
  0042                  134             ORG     42H
  0042                  135     CMD_BLOCK       LABEL   BYTE
  0042 (7 ??)           136     HD_ERROR        DB      7 DUP(?)        ; OVERLAYS DISKETTE STATUS
  006C                  137             ORG     06CH
  006C ????             138     TIMER_LOW       DW      ?               ; TIMER LOW WORD
  0072                  139             ORG     72H
  0072 ????             140     RESET_FLAG      DW      ?               ; 1234H IF KEYBOARD RESET UNDERWAY
  0074                  141             ORG     74H
  0074 ??               142     DISK_STATUS     DB      ?               ; FIXED DISK STATUS BYTE
  0075 ??               143     HF_NUM          DB      ?               ; COUNT OF FIXED DISK DRIVES
  0076 ??               144     CONTROL_BYTE    DB      ?               ; CONTROL BYTE DRIVE OPTIONS
  0077 ??               145     PORT_OFF        DB      ?               ; PORT OFFSET
  ----                  146     DATA    ENDS
                        147
  ----                  148     CODE    SEGMENT
                        149
                        150     ;--------------------------------------------------------
                        151     ; HARDWARE SPECIFIC VALUES                              :
                        152     ;                                                       :
                        153     ;  -  CONTROLLER I/O PORT                              :
                        154     ;     > WHEN READ FROM:                                 :
```

```
LOC OBJ                 LINE    SOURCE

                        155     ;       HF_PORT+0 - READ DATA (FROM CONTROLLER TO CPU)  :
                        156     ;       HF_PORT+1 - READ CONTROLLER HARDWARE STATUS     :
                        157     ;                   (CONTROLLER TO CPU)                 :
                        158     ;       HF_PORT+2 - READ CONFIGURATION SWITCHES         :
                        159     ;       HF_PORT+3 - NOT USED                            :
                        160     ;     > WHEN WRITTEN TO:                                :
                        161     ;       HF_PORT+0 - WRITE DATA (FROM CPU TO CONTROLLER) :
                        162     ;       HF_PORT+1 - CONTROLLER RESET                    :
                        163     ;       HF_PORT+2 - GENERATE CONTROLLER SELECT PULSE     :
                        164     ;       HF_PORT+3 - WRITE PATTERN TO DMA AND INTERRUPT  :
                        165     ;                   MASK REGISTER                       :
                        166     ;                                                       :
                        167     ;-------------------------------------------------------
                        168
  0320                  169     HF_PORT         EQU     0320H           ; DISK PORT
  0008                  170     R1_BUSY         EQU     00001000B       ; DISK PORT 1 BUSY BIT
  0004                  171     R1_BUS          EQU     00000100B       ;             COMMAND/DATA BIT
  0002                  172     R1_IOMODE       EQU     00000010B       ;             MODE BIT
  0001                  173     R1_REQ          EQU     00000001B       ;             REQUEST BIT
                        174
  0047                  175     DMA_READ        EQU     01000111B       ; CHANNEL 3 (047H)
  004B                  176     DMA_WRITE       EQU     01001011B       ; CHANNEL 3 (04BH)
  0000                  177     DMA             EQU     0               ; DMA ADDRESS
  0082                  178     DMA_HIGH        EQU     082H            ; PORT FOR HIGH 4 BITS OF DMA
                        179
  0000                  180     TST_RDY_CMD     EQU     00000000B       ; CNTLR READY (00H)
  0001                  181     RECAL_CMD       EQU     00000001B       ;       RECAL (01H)
  0003                  182     SENSE_CMD       EQU     00000011B       ;       SENSE (03H)
  0004                  183     FMTDRV_CMD      EQU     00000100B       ;       DRIVE (04H)
  0005                  184     CHK_TRK_CMD     EQU     00000101B       ;       T CHK (05H)
  0006                  185     FMTTRK_CMD      EQU     00000110B       ;       TRACK (06H)
  0007                  186     FMTBAD_CMD      EQU     00000111B       ;       BAD   (07H)
  0008                  187     READ_CMD        EQU     00001000B       ;       READ  (08H)
  000A                  188     WRITE_CMD       EQU     00001010B       ;       WRITE (0AH)
  000B                  189     SEEK_CMD        EQU     00001011B       ;       SEEK  (0BH)
  000C                  190     INIT_DRV_CMD    EQU     00001100B       ;       INIT  (0CH)
  000D                  191     RD_ECC_CMD      EQU     00001101B       ;       BURST (0DH)
  000E                  192     RD_BUFF_CMD     EQU     00001110B       ;       BUFFR (0EH)
  000F                  193     WR_BUFF_CMD     EQU     00001111B       ;       BUFFR (0FH)
  00E0                  194     RAM_DIAG_CMD    EQU     11100000B       ;       RAM   (E0H)
  00E3                  195     CHK_DRV_CMD     EQU     11100011B       ;       DRV   (E3H)
  00E4                  196     CNTLR_DIAG_CMD  EQU     11100100B       ;       CNTLR (E4H)
  00E5                  197     RD_LONG_CMD     EQU     11100101B       ;       RLONG (E5H)
  00E6                  198     WR_LONG_CMD     EQU     11100110B       ;       WLONG (E6H)
                        199
  0020                  200     INT_CTL_PORT    EQU     20H             ; 8259 CONTROL PORT
  0020                  201     EOI             EQU     20H             ; END OF INTERRUPT COMMAND
                        202
  0008                  203     MAX_FILE        EQU     8
  0002                  204     S_MAX_FILE      EQU     2
                        205
                        206             ASSUME  CS:CODE
0000                    207             ORG     0H
0000 55                 208             DB      055H                    ; GENERIC BIOS HEADER
0001 AA                 209             DB      0AAH
0002 10                 210             DB      16D
                        211
                        212     ;--------------------------------------------------------------
                        213     ; FIXED DISK I/O SETUP                                       :
                        214     ;                                                            :
                        215     ;  -  ESTABLISH TRANSFER VECTORS FOR THE FIXED DISK          :
                        216     ;  -  PERFORM POWER ON DIAGNOSTICS                           :
                        217     ;     SHOULD AN ERROR OCCUR A "1701" MESSAGE IS DISPLAYED    :
                        218     ;                                                            :
                        219     ;--------------------------------------------------------------
                        220
0003                    221     DISK_SETUP      PROC    FAR
0003 EB1E               222             JMP     SHORT   L3
0005 353030303035       223             DB      '5000059 (C)COPYRIGHT  IBM 1982'        ; COPYRIGHT NOTICE
     20284329434F50
     59524947485420
     2049424D203139
     3832
0023                    224     L3:
                        225             ASSUME  DS:DUMMY
0023 2BC0               226             SUB     AX,AX                                   ; ZERO
0025 8ED8               227             MOV     DS,AX
```

```
LOC OBJ                 LINE    SOURCE

0027 FA                 228             CLI
0028 A14C00             229             MOV     AX,WORD PTR ORG_VECTOR                  ; GET DISKETTE VECTOR
002B A30001             230             MOV     WORD PTR DISK_VECTOR,AX                 ;  INTO INT 40H
002E A14E00             231             MOV     AX,WORD PTR ORG_VECTOR+2
0031 A30201             232             MOV     WORD PTR DISK_VECTOR+2,AX
0034 C7064C005602       233             MOV     WORD PTR ORG_VECTOR, OFFSET DISK_IO     ; HDISK HANDLER
003A 8C0E4E00           234             MOV     WORD PTR ORG_VECTOR+2,CS
003E B86007             235             MOV     AX, OFFSET HD_INT                       ; HDISK INTERRUPT
0041 A33400             236             MOV     WORD PTR HDISK_INT,AX
0044 8C0E3600           237             MOV     WORD PTR HDISK_INT+2,CS
0048 C70664008601       238             MOV     WORD PTR BOOT_VEC,OFFSET BOOT_STRAP     ; BOOTSTRAP
004E 8C0E6600           239             MOV     WORD PTR BOOT_VEC+2,CS
0052 C7060401E703       240             MOV     WORD PTR HF_TBL_VEC,OFFSET FD_TBL       ; PARAMETER TBL
0058 8C0E0601           241             MOV     WORD PTR HF_TBL_VEC+2,CS
005C FB                 242             STI
                        243
                        244             ASSUME  DS:DATA
005D B84000             245             MOV     AX,DATA                 ; ESTABLISH SEGMENT
0060 8ED8               246             MOV     DS,AX
0062 C606740000         247             MOV     DISK_STATUS,0           ; RESET THE STATUS INDICATOR
0067 C606750000         248             MOV     HF_NUM,0                ; ZERO COUNT OF DRIVES
006C C606430000         249             MOV     CMD_BLOCK+1,0           ; DRIVE ZERO, SET VALUE IN BLOCK
0071 C606770000         250             MOV     PORT_OFF,0              ; ZERO CARD OFFSET
                        251
0076 B92500             252             MOV     CX,25H                  ; RETRY COUNT
0079                    253     L4:
0079 E8F200             254             CALL    HD_RESET_1              ; RESET CONTROLLER
007C 7305               255             JNC     L7
007E E2F9               256             LOOP    L4                      ; TRY RESET AGAIN
0080 E9BF00             257             JMP     ERROR_EX
0083                    258     L7:
0083 B90100             259             MOV     CX,1
0086 BA8000             260             MOV     DX,80H
                        261
0089 B80012             262             MOV     AX,1200H                ; CONTROLLER DIAGNOSTICS
008C CD13               263             INT     13H
008E 7303               264             JNC     P7
0090 E9AF00             265             JMP     ERROR_EX
0093                    266     P7:
0093 B80014             267             MOV     AX,1400H                ; CONTROLLER DIAGNOSTICS
0096 CD13               268             INT     13H
0098 7303               269             JNC     P9
009A E9A500             270             JMP     ERROR_EX
009D                    271     P9:
009D C7066C000000       272             MOV     TIMER_LOW,0             ; ZERO TIMER
00A3 A17200             273             MOV     AX,RESET_FLAG
00A6 3D3412             274             CMP     AX,1234H                ; KEYBOARD RESET
00A9 7506               275             JNE     P8
00AB C7066C009A01       276             MOV     TIMER_LOW,410D          ; SKIP WAIT ON RESET
00B1                    277     P8:
00B1 E421               278             IN      AL,021H                 ; TIMER
00B3 24FE               279             AND     AL,0FEH                 ; ENABLE TIMER
00B5 E621               280             OUT     021H,AL                 ; START TIMER
00B7                    281     P4:
00B7 E8B400             282             CALL    HD_RESET_1              ; RESET CONTROLLER
00BA 7207               283             JC      P10
00BC B80010             284             MOV     AX,1000H                ; READY
00BF CD13               285             INT     13H
00C1 730B               286             JNC     P2
00C3                    287     P10:
00C3 A16C00             288             MOV     AX,TIMER_LOW
00C6 3DBE01             289             CMP     AX,446D                 ; 25 SECONDS
00C9 72EC               290             JB      P4
00CB EB7590             291             JMP     ERROR_EX
00CE                    292     P2:
00CE B90100             293             MOV     CX,1
00D1 BA8000             294             MOV     DX,80H
                        295
00D4 B80011             296             MOV     AX,1100H                ; RECALIBRATE
00D7 CD13               297             INT     13H
00D9 7267               298             JC      ERROR_EX
                        299
00DB B80009             300             MOV     AX,0900H                ; SET DRIVE PARAMETERS
00DE CD13               301             INT     13H
00E0 7260               302             JC      ERROR_EX
                        303
00E2 B800C8             304             MOV     AX,0C800H               ; DMA TO BUFFER
```

```
LOC OBJ                  LINE    SOURCE

00E5 8EC0                305             MOV     ES,AX                   ; SET SEGMENT
00E7 2BDB                306             SUB     BX,BX
00E9 B8000F              307             MOV     AX,0F00H                ; WRITE SECTOR BUFFER
00EC CD13                308             INT     13H
00EE 7252                309             JC      ERROR_EX
                         310
00F0 FE067500            311             INC     HF_NUM                  ; DRIVE ZERO RESPONDED
                         312
00F4 BA1302              313             MOV     DX,213H                 ; EXPANSION BOX
00F7 B000                314             MOV     AL,0
00F9 EE                  315             OUT     DX,AL                   ; TURN BOX OFF
00FA BA2103              316             MOV     DX,321H                 ; TEST IF CONTROLLER
00FD EC                  317             IN      AL,DX                   ; ... IS IN THE SYSTEM UNIT
00FE 240F                318             AND     AL,0FH
0100 3C0F                319             CMP     AL,0FH
0102 7406                320             JE      BOX_ON
0104 C7066C00A401        321             MOV     TIMER_LOW,420D          ; CONTROLLER IS IN SYSTEM UNIT
010A                     322     BOX_ON:
010A BA1302              323             MOV     DX,213H                 ; EXPANSION BOX
010D B0FF                324             MOV     AL,0FFH
010F EE                  325             OUT     DX,AL                   ; TURN BOX ON
                         326
0110 B90100              327             MOV     CX,1                    ; ATTEMPT NEXT DRIVES
0113 BA8100              328             MOV     DX,081H
0116                     329     P3:
0116 2BC0                330             SUB     AX,AX                   ; RESET
0118 CD13                331             INT     13H
011A 7240                332             JC      POD_DONE
011C B80011              333             MOV     AX,01100H               ; RECAL
011F CD13                334             INT     13H
0121 730B                335             JNC     P5
0123 A16C00              336             MOV     AX,TIMER_LOW
0126 3DBE01              337             CMP     AX,446D                 ; 25 SECONDS
0129 72EB                338             JB      P3
012B EB2F90              339             JMP     POD_DONE
012E                     340     P5:
012E B80009              341             MOV     AX,0900H                ; INITIALIZE CHARACTERISTICS
0131 CD13                342             INT     13H
0133 7227                343             JC      POD_DONE
0135 FE067500            344             INC     HF_NUM                  ; TALLY ANOTHER DRIVE
0139 81FA8100            345             CMP     DX,(80H + S_MAX_FILE - 1)
013D 731D                346             JAE     POD_DONE
013F 42                  347             INC     DX
0140 EBD4                348             JMP     P3
                         349
                         350     ;----- POD ERROR
                         351
0142                     352     ERROR_EX:
0142 BD0F00              353             MOV     BP,0FH                  ; POD ERROR FLAG
0145 2BC0                354             SUB     AX,AX
0147 8BF0                355             MOV     SI,AX
0149 B9060090            356             MOV     CX,F17L                 ; MESSAGE CHARACTER COUNT
014D B700                357             MOV     BH,0                    ; PAGE ZERO
014F                     358     OUT_CH:
014F 2E8A846801          359             MOV     AL,CS:F17[SI]           ; GET BYTE
0154 B40E                360             MOV     AH,14D                  ; VIDEO OUT
0156 CD10                361             INT     10H                     ; DISPLAY CHARACTER
0158 46                  362             INC     SI                      ; NEXT CHAR
0159 E2F4                363             LOOP    OUT_CH                  ; DO MORE
015B F9                  364             STC
015C                     365     POD_DONE:
015C FA                  366             CLI
015D E421                367             IN      AL,021H                 ; BE SURE TIMER IS DISABLED
015F 0C01                368             OR      AL,01H
0161 E621                369             OUT     021H,AL
0163 FB                  370             STI
0164 E8A500              371             CALL    DSBL
0167 CB                  372             RET
                         373
0168 31373031            374     F17     DB      '1701',0DH,0AH
016C 0D
016D 0A
  0006                   375     F17L    EQU     $-F17
                         376
016E                     377     HD_RESET_1      PROC    NEAR
016E 51                  378             PUSH    CX                      ; SAVE REGISTER
016F 52                  379             PUSH    DX
```

```
LOC OBJ                 LINE    SOURCE

0170 F8                 380             CLC                             ; CLEAR CARRY
0171 B90001             381             MOV     CX,0100H                ; RETRY COUNT
0174                    382     L6:
0174 E80706             383             CALL    PORT_1
0177 EE                 384             OUT     DX,AL                   ; RESET CARD
0178 E80306             385             CALL    PORT_1
017B EC                 386             IN      AL,DX                   ; CHECK STATUS
017C 2402               387             AND     AL,2                    ; ERROR BIT
017E 7403               388             JZ      R3
0180 E2F2               389             LOOP    L6
0182 F9                 390             STC
0183                    391     R3:
0183 5A                 392             POP     DX                      ; RESTORE REGISTER
0184 59                 393             POP     CX
0185 C3                 394             RET
                        395     HD_RESET_1      ENDP
                        396
                        397     DISK_SETUP      ENDP
                        398
                        399     ;----- INT 19 ----------------------------------------------------
                        400     ;                                                                 :
                        401     ; INTERRUPT 19 BOOT STRAP LOADER                                  :
                        402     ;                                                                 :
                        403     ;  -  THE FIXED DISK BIOS REPLACES THE INTERRUPT 19               :
                        404     ;     BOOT STRAP VECTOR WITH A POINTER TO THIS BOOT ROUTINE       :
                        405     ;  -  RESET THE DEFAULT DISK AND DISKETTE PARAMETER VECTORS       :
                        406     ;  -  THE BOOT BLOCK TO BE READ IN WILL BE ATTEMPTED FROM         :
                        407     ;     CYLINDER 0 SECTOR 1 OF THE DEVICE.                          :
                        408     ;  -  THE BOOTSTRAP SEQUENCE IS:                                 :
                        409     ;     > ATTEMPT TO LOAD FROM THE DISKETTE INTO THE BOOT          :
                        410     ;       LOCATION (0000:7C00) AND TRANSFER CONTROL THERE          :
                        411     ;     > IF THE DISKETTE FAILS THE FIXED DISK IS TRIED FOR A      :
                        412     ;       VALID BOOTSTRAP BLOCK. A VALID BOOT BLOCK ON THE         :
                        413     ;       FIXED DISK CONSISTS OF THE BYTES  055H 0AAH  AS THE      :
                        414     ;       LAST TWO BYTES OF THE BLOCK                              :
                        415     ;     > IF THE ABOVE FAILS CONTROL IS PASSED TO RESIDENT BASIC    :
                        416     ;                                                                 :
                        417     ;-----------------------------------------------------------------
                        418
0186                    419     BOOT_STRAP:
                        420             ASSUME  DS:DUMMY,ES:DUMMY
0186 2BC0               421             SUB     AX,AX
0188 8ED8               422             MOV     DS,AX                   ; ESTABLISH SEGMENT
                        423
                        424     ;----- RESET PARAMETER VECTORS
                        425
018A FA                 426             CLI
018B C7060401E703       427             MOV     WORD PTR HF_TBL_VEC, OFFSET FD_TBL
0191 8C0E0601           428             MOV     WORD PTR HF_TBL_VEC+2, CS
0195 C70678000102       429             MOV     WORD PTR DISKETTE_PARM, OFFSET DISKETTE_TBL
019B 8C0E7A00           430             MOV     WORD PTR DISKETTE_PARM+2, CS
019F FB                 431             STI
                        432
                        433     ;----- ATTEMPT BOOTSTRAP FROM DISKETTE
                        434
01A0 B90300             435             MOV     CX,3                    ; SET RETRY COUNT
01A3                    436     H1:                                     ; IPL_SYSTEM
01A3 51                 437             PUSH    CX                      ; SAVE RETRY COUNT
01A4 2BD2               438             SUB     DX,DX                   ; DRIVE ZERO
01A6 2BC0               439             SUB     AX,AX                   ; RESET THE DISKETTE
01A8 CD13               440             INT     13H                     ; FILE IO CALL
01AA 720F               441             JC      H2                      ; IF ERROR, TRY AGAIN
01AC B80102             442             MOV     AX,0201H                ; READ IN THE SINGLE SECTOR
                        443
01AF 2BD2               444             SUB     DX,DX
01B1 8EC2               445             MOV     ES,DX                   ; ESTABLISH SEGMENT
01B3 BB007C             446             MOV     BX,OFFSET BOOT_LOCN
                        447
01B6 B90100             448             MOV     CX,1                    ; SECTOR 1, TRACK 0
01B9 CD13               449             INT     13H                     ; FILE IO CALL
01BB 59                 450     H2:     POP     CX                      ; RECOVER RETRY COUNT
01BC 730A               451             JNC     H4                      ; CF SET BY UNSUCCESSFUL READ
01BE 80FC80             452             CMP     AH,80H                  ; IF TIME OUT, NO RETRY
01C1 740A               453             JZ      H5                      ; TRY FIXED DISK
01C3 E2DE               454             LOOP    H1                      ; DO IT FOR RETRY TIMES
01C5 EB0690             455             JMP     H5                      ; UNABLE TO IPL FROM THE DISKETTE
01C8                    456     H4:                                     ; IPL WAS SUCCESSFUL
```

```
LOC OBJ                 LINE    SOURCE

01C8 EA007C0000         457             JMP     BOOT_LOCN
                        458
                        459     ;----- ATTEMPT BOOTSTRAP FROM FIXED DISK
                        460
01CD                    461     H5:
01CD 2BC0               462             SUB     AX,AX                   ; RESET DISKETTE
01CF 2BD2               463             SUB     DX,DX
01D1 CD13               464             INT     13H
01D3 B90300             465             MOV     CX,3                    ; SET RETRY COUNT
01D6                    466     H6:                                     ; IPL_SYSTEM
01D6 51                 467             PUSH    CX                      ; SAVE RETRY COUNT
01D7 BA8000             468             MOV     DX,0080H                ; FIXED DISK ZERO
01DA 2BC0               469             SUB     AX,AX                   ; RESET THE FIXED DISK
01DC CD13               470             INT     13H                     ; FILE IO CALL
01DE 7212               471             JC      H7                      ; IF ERROR, TRY AGAIN
01E0 B80102             472             MOV     AX,0201H                ; READ IN THE SINGLE SECTOR
01E3 2BDB               473             SUB     BX,BX
01E5 8EC3               474             MOV     ES,BX
01E7 BB007C             475             MOV     BX,OFFSET BOOT_LOCN     ; TO THE BOOT LOCATION
01EA BA8000             476             MOV     DX,80H                  ; DRIVE NUMBER
01ED B90100             477             MOV     CX,1                    ; SECTOR 1, TRACK 0
01F0 CD13               478             INT     13H                     ; FILE IO CALL
01F2 59                 479     H7:     POP     CX                      ; RECOVER RETRY COUNT
01F3 7208               480             JC      H8
01F5 A1FE7D             481             MOV     AX,WORD PTR BOOT_LOCN+510D
01F8 3D55AA             482             CMP     AX,0AA55H               ; TEST FOR GENERIC BOOT BLOCK
01FB 74CB               483             JZ      H4
01FD                    484     H8:
01FD E2D7               485             LOOP    H6                      ; DO IT FOR RETRY TIMES
                        486
                        487     ;------ UNABLE TO IPL FROM THE DISKETTE OR FIXED DISK
                        488
01FF CD18               489             INT     18H                     ; RESIDENT BASIC
                        490
0201                    491     DISKETTE_TBL:
                        492
0201 CF                 493             DB      11001111B               ; SRT=C, HD UNLOAD=OF - 1ST SPEC BYTE
0202 02                 494             DB      2                       ; HD LOAD=1, MODE=DMA - 2ND SPEC BYTE
0203 25                 495             DB      25H                     ; WAIT AFTER OPN TIL MOTOR OFF
0204 02                 496             DB      2                       ; 512 BYTES PER SECTOR
0205 08                 497             DB      8                       ; EOT (LAST SECTOR ON TRACK)
0206 2A                 498             DB      02AH                    ; GAP LENGTH
0207 FF                 499             DB      0FFH                    ; DTL
0208 50                 500             DB      050H                    ; GAP LENGTH FOR FORMAT
0209 F6                 501             DB      0F6H                    ; FILL BYTE FOR FORMAT
020A 19                 502             DB      25                      ; HEAD SETTLE TIME (MILLISECONDS)
020B 04                 503             DB      4                       ; MOTOR START TIME (1/8 SECOND)
                        504
                        505     ;----- MAKE SURE THAT ALL HOUSEKEEPING IS DONE BEFORE EXIT
                        506
020C                    507     DSBL    PROC    NEAR
                        508             ASSUME  DS:DATA
020C 1E                 509             PUSH    DS                      ; SAVE SEGMENT
020D B84000             510             MOV     AX,DATA
0210 8ED8               511             MOV     DS,AX
                        512
0212 8A267700           513             MOV     AH,PORT_OFF
0216 50                 514             PUSH    AX                      ; SAVE OFFSET
                        515
0217 C606770000         516             MOV     PORT_OFF,0H
021C E86905             517             CALL    PORT_3
021F 2AC0               518             SUB     AL,AL
0221 EE                 519             OUT     DX,AL                   ; RESET INT/DMA MASK
0222 C606770004         520             MOV     PORT_OFF,4H
0227 E85E05             521             CALL    PORT_3
022A 2AC0               522             SUB     AL,AL
022C EE                 523             OUT     DX,AL                   ; RESET INT/DMA MASK
022D C606770008         524             MOV     PORT_OFF,8H
0232 E85305             525             CALL    PORT_3
0235 2AC0               526             SUB     AL,AL
0237 EE                 527             OUT     DX,AL                   ; RESET INT/DMA MASK
0238 C60677000C         528             MOV     PORT_OFF,0CH
023D E84805             529             CALL    PORT_3
0240 2AC0               530             SUB     AL,AL
0242 EE                 531             OUT     DX,AL                   ; RESET INT/DMA MASK
0243 B007               532             MOV     AL,07H
0245 E60A               533             OUT     DMA+10,AL               ; SET DMA MODE TO DISABLE
```

```
LOC OBJ                 LINE    SOURCE

0247 FA                 534             CLI                             ; DISABLE INTERRUPTS
0248 E421               535             IN      AL,021H
024A 0C20               536             OR      AL,020H
024C E621               537             OUT     021H,AL                 ; DISABLE INTERRUPT 5
024E FB                 538             STI                             ; ENABLE INTERRUPTS
024F 58                 539             POP     AX                      ; RESTORE OFFSET
0250 88267700           540             MOV     PORT_OFF,AH
0254 1F                 541             POP     DS                      ; RESTORE SEGMENT
0255 C3                 542             RET
                        543     DSBL    ENDP
                        544
                        545     ;---------------------------------------
                        546     ;       FIXED DISK BIOS ENTRY POINT     :
                        547     ;---------------------------------------
                        548
0256                    549     DISK_IO PROC    FAR
                        550             ASSUME  DS:NOTHING,ES:NOTHING
0256 80FA80             551             CMP     DL,80H                  ; TEST FOR FIXED DISK DRIVE
0259 7305               552             JAE     HARD_DISK               ; YES, HANDLE HERE
025B CD40               553             INT     40H                     ; DISKETTE HANDLER
025D                    554     RET_2:
025D CA0200             555             RET     2                       ; BACK TO CALLER
0260                    556     HARD_DISK:
                        557             ASSUME  DS:DATA
0260 FB                 558             STI                             ; ENABLE INTERRUPTS
0261 0AE4               559             OR      AH,AH
0263 7509               560             JNZ     A3
0265 CD40               561             INT     40H                     ; RESET NEC WHEN AH=0
0267 2AE4               562             SUB     AH,AH
0269 80FA81             563             CMP     DL,(80H + S_MAX_FILE - 1)
026C 77EF               564             JA      RET_2
026E                    565     A3:
026E 80FC08             566             CMP     AH,08                   ; GET PARAMETERS IS A SPECIAL CASE
0271 7503               567             JNZ     A2
0273 E91A01             568             JMP     GET_PARM_N
0276                    569     A2:
0276 53                 570             PUSH    BX                      ; SAVE REGISTERS DURING OPERATION
0277 51                 571             PUSH    CX
0278 52                 572             PUSH    DX
0279 1E                 573             PUSH    DS
027A 06                 574             PUSH    ES
027B 56                 575             PUSH    SI
027C 57                 576             PUSH    DI
                        577
027D E86A00             578             CALL    DISK_IO_CONT            ; PERFORM THE OPERATION
                        579
0280 50                 580             PUSH    AX
0281 E888FF             581             CALL    DSBL                    ; BE SURE DISABLES OCCURRED
0284 B84000             582             MOV     AX,DATA
0287 8ED8               583             MOV     DS,AX                   ; ESTABLISH SEGMENT
0289 58                 584             POP     AX
028A 8A267400           585             MOV     AH,DISK_STATUS          ; GET STATUS FROM OPERATION
028E 80FC01             586             CMP     AH,1                    ; SET THE CARRY FLAG TO INDICATE
0291 F5                 587             CMC                             ;  SUCCESS OR FAILURE
0292 5F                 588             POP     DI                      ; RESTORE REGISTERS
0293 5E                 589             POP     SI
0294 07                 590             POP     ES
0295 1F                 591             POP     DS
0296 5A                 592             POP     DX
0297 59                 593             POP     CX
0298 5B                 594             POP     BX
0299 CA0200             595             RET     2                       ; THROW AWAY SAVED FLAGS
                        596     DISK_IO ENDP
                        597
029C                    598     M1      LABEL   WORD                    ; FUNCTION TRANSFER TABLE
029C 3803               599             DW      DISK_RESET              ; 000H
029E 4D03               600             DW      RETURN_STATUS           ; 001H
02A0 5603               601             DW      DISK_READ               ; 002H
02A2 6003               602             DW      DISK_WRITE              ; 003H
02A4 6A03               603             DW      DISK_VERF               ; 004H
02A6 7203               604             DW      FMT_TRK                 ; 005H
02A8 7903               605             DW      FMT_BAD                 ; 006H
02AA 8003               606             DW      FMT_DRV                 ; 007H
02AC 3003               607             DW      BAD_COMMAND             ; 008H
02AE 2704               608             DW      INIT_DRV                ; 009H
02B0 CF04               609             DW      RD_LONG                 ; 00AH
02B2 DD04               610             DW      WR_LONG                 ; 00BH
```

```
LOC OBJ                 LINE    SOURCE

02B4 F204               611             DW      DISK_SEEK               ; 00CH
02B6 3803               612             DW      DISK_RESET              ; 00DH
02B8 F904               613             DW      RD_BUFF                 ; 00EH
02BA 0705               614             DW      WR_BUFF                 ; 00FH
02BC 1505               615             DW      TST_RDY                 ; 010H
02BE 1C05               616             DW      HDISK_RECAL             ; 011H
02C0 2305               617             DW      RAM_DIAG                ; 012H
02C2 2A05               618             DW      CHK_DRV                 ; 013H
02C4 3105               619             DW      CNTLR_DIAG              ; 014H
  002A                  620     M1L     EQU     $-M1
                        621
02C6                    622     SETUP_A PROC    NEAR
                        623
02C6 C606740000         624             MOV     DISK_STATUS,0           ; RESET THE STATUS INDICATOR
02CB 51                 625             PUSH    CX                      ; SAVE CX
                        626
                        627     ;----- CALCULATE THE PORT OFFSET
                        628
02CC 8AEA               629             MOV     CH,DL                   ; SAVE DL
02CE 80CA01             630             OR      DL,1
02D1 FECA               631             DEC     DL
02D3 D0E2               632             SHL     DL,1                    ; GENERATE OFFSET
02D5 88167700           633             MOV     PORT_OFF,DL             ; STORE OFFSET
02D9 8AD5               634             MOV     DL,CH                   ; RESTORE DL
02DB 80E201             635             AND     DL,1
                        636
02DE B105               637             MOV     CL,5                    ; SHIFT COUNT
02E0 D2E2               638             SHL     DL,CL                   ; DRIVE NUMBER (0,1)
02E2 0AD6               639             OR      DL,DH                   ; HEAD NUMBER
02E4 88164300           640             MOV     CMD_BLOCK+1,DL
02E8 59                 641             POP     CX
02E9 C3                 642             RET
                        643     SETUP_A ENDP
                        644
02EA                    645     DISK_IO_CONT    PROC    NEAR
02EA 50                 646             PUSH    AX
02EB B84000             647             MOV     AX,DATA
02EE 8ED8               648             MOV     DS,AX                   ; ESTABLISH SEGMENT
02F0 58                 649             POP     AX
02F1 80FC01             650             CMP     AH,01H                  ; RETURN STATUS
02F4 7503               651             JNZ     A4
02F6 EB5590             652             JMP     RETURN_STATUS
02F9                    653     A4:
02F9 80EA80             654             SUB     DL,80H                  ; CONVERT DRIVE NUMBER TO 0 BASED RANGE
02FC 80FA08             655             CMP     DL,MAX_FILE             ; LEGAL DRIVE TEST
02FF 732F               656             JAE     BAD_COMMAND
                        657
0301 E8C2FF             658             CALL    SETUP_A
                        659
                        660     ;----- SET UP COMMAND BLOCK
                        661
0304 FEC9               662             DEC     CL                      ; SECTORS 0-16 FOR CONTROLLER
0306 C606420000         663             MOV     CMD_BLOCK+0,0
030B 880E4400           664             MOV     CMD_BLOCK+2,CL          ; SECTOR AND HIGH 2 BITS CYLINDER
030F 882E4500           665             MOV     CMD_BLOCK+3,CH          ; CYLINDER
0313 A24600             666             MOV     CMD_BLOCK+4,AL          ; INTERLEAVE / BLOCK COUNT
0316 A07600             667             MOV     AL,CONTROL_BYTE         ; CONTROL BYTE (STEP OPTION)
0319 A24700             668             MOV     CMD_BLOCK+5,AL
031C 50                 669             PUSH    AX                      ; SAVE AX
031D 8AC4               670             MOV     AL,AH                   ; GET INTO LOW BYTE
031F 32E4               671             XOR     AH,AH                   ; ZERO HIGH BYTE
0321 D1E0               672             SAL     AX,1                    ; *2 FOR TABLE LOOKUP
0323 8BF0               673             MOV     SI,AX                   ; PUT INTO SI FOR BRANCH
0325 3D2A00             674             CMP     AX,M1L                  ; TEST WITHIN RANGE
0328 58                 675             POP     AX                      ; RESTORE AX
0329 7305               676             JNB     BAD_COMMAND
032B 2EFFA49C02         677             JMP     WORD PTR CS:[SI + OFFSET M1]
0330                    678     BAD_COMMAND:
0330 C606740001         679             MOV     DISK_STATUS,BAD_CMD     ; COMMAND ERROR
0335 B000               680             MOV     AL,0
0337 C3                 681             RET
                        682     DISK_IO_CONT    ENDP
                        683
                        684     ;--------------------------------------------
                        685     ;       RESET THE DISK SYSTEM  (AH = 000H)      :
                        686     ;--------------------------------------------
                        687
```

```
LOC OBJ                 LINE    SOURCE

0338                    688     DISK_RESET      PROC    NEAR
0338 E84304             689             CALL    PORT_1                  ; RESET PORT
033B EE                 690             OUT     DX,AL                   ; ISSUE RESET
033C E83F04             691             CALL    PORT_1                  ; CONTROLLER HARDWARE STATUS
033F EC                 692             IN      AL,DX                   ; GET STATUS
0340 2402               693             AND     AL,2                    ; ERROR BIT
0342 7406               694             JZ      DR1
0344 C606740005         695             MOV     DISK_STATUS,BAD_RESET
0349 C3                 696             RET
034A                    697     DR1:
034A E9DA00             698             JMP     INIT_DRV                ; SET THE DRIVE PARAMETERS
                        699     DISK_RESET      ENDP
                        700
                        701     ;--------------------------------------------
                        702     ;       DISK STATUS ROUTINE   (AH = 001H)       :
                        703     ;--------------------------------------------
                        704
034D                    705     RETURN_STATUS   PROC    NEAR
034D A07400             706             MOV     AL,DISK_STATUS          ; OBTAIN PREVIOUS STATUS
0350 C606740000         707             MOV     DISK_STATUS,0           ; RESET STATUS
0355 C3                 708             RET
                        709     RETURN_STATUS   ENDP
                        710
                        711     ;--------------------------------------------
                        712     ;       DISK READ ROUTINE   (AH = 002H)         :
                        713     ;--------------------------------------------
                        714
0356                    715     DISK_READ       PROC    NEAR
0356 B047               716             MOV     AL,DMA_READ             ; MODE BYTE FOR DMA READ
0358 C606420008         717             MOV     CMD_BLOCK+0,READ_CMD
035D E9E501             718             JMP     DMA_OPN
                        719     DISK_READ       ENDP
                        720
                        721     ;--------------------------------------------
                        722     ;       DISK WRITE ROUTINE   (AH = 003H)        :
                        723     ;--------------------------------------------
                        724
0360                    725     DISK_WRITE      PROC    NEAR
0360 B04B               726             MOV     AL,DMA_WRITE            ; MODE BYTE FOR DMA WRITE
0362 C60642000A         727             MOV     CMD_BLOCK+0,WRITE_CMD
0367 E9DB01             728             JMP     DMA_OPN
                        729     DISK_WRITE      ENDP
                        730
                        731     ;--------------------------------------------
                        732     ;       DISK VERIFY   (AH = 004H)               :
                        733     ;--------------------------------------------
                        734
036A                    735     DISK_VERF       PROC    NEAR
036A C606420005         736             MOV     CMD_BLOCK+0,CHK_TRK_CMD
036F E9C401             737             JMP     NDMA_OPN
                        738     DISK_VERF       ENDP
                        739
                        740     ;--------------------------------------------
                        741     ;       FORMATTING   (AH = 005H 006H 007H)      :
                        742     ;--------------------------------------------
                        743
0372                    744     FMT_TRK PROC    NEAR                    ; FORMAT TRACK  (AH = 005H)
0372 C606420006         745             MOV     CMD_BLOCK,FMTTRK_CMD
0377 EB0C               746             JMP     SHORT   FMT_CONT
                        747     FMT_TRK ENDP
                        748
0379                    749     FMT_BAD PROC    NEAR                    ; FORMAT BAD TRACK  (AH = 006H)
0379 C606420007         750             MOV     CMD_BLOCK,FMTBAD_CMD
037E EB05               751             JMP     SHORT   FMT_CONT
                        752     FMT_BAD ENDP
                        753
0380                    754     FMT_DRV PROC    NEAR                    ; FORMAT DRIVE  (AH = 007H)
0380 C606420004         755             MOV     CMD_BLOCK,FMTDRV_CMD
                        756     FMT_DRV ENDP
                        757
0385                    758     FMT_CONT:
0385 A04400             759             MOV     AL,CMD_BLOCK+2          ; ZERO OUT SECTOR FIELD
0388 24C0               760             AND     AL,11000000B
038A A24400             761             MOV     CMD_BLOCK+2,AL
038D E9A601             762             JMP     NDMA_OPN
                        763
```

```
LOC OBJ                 LINE    SOURCE

                        764     ;--------------------------------------------
                        765     ;       GET PARAMETERS   (AH = 8)            :
                        766     ;--------------------------------------------
                        767
0390                    768     GET_PARM_N      LABEL   NEAR
0390                    769     GET_PARM        PROC    FAR             ; GET DRIVE PARAMETERS
0390 1E                 770             PUSH    DS                      ; SAVE REGISTERS
0391 06                 771             PUSH    ES
0392 53                 772             PUSH    BX
                        773
                        774             ASSUME  DS:DUMMY
0393 2BC0               775             SUB     AX,AX                   ; ESTABLISH ADDRESSING
0395 8ED8               776             MOV     DS,AX
0397 C41E0401           777             LES     BX,HF_TBL_VEC
                        778             ASSUME  DS:DATA
039B B84000             779             MOV     AX,DATA
039E 8ED8               780             MOV     DS,AX                   ; ESTABLISH SEGMENT
                        781
03A0 80EA80             782             SUB     DL,80H
03A3 80FA08             783             CMP     DL,MAX_FILE             ; TEST WITHIN RANGE
03A6 732F               784             JAE     G4
                        785
03A8 E81BFF             786             CALL    SETUP_A
                        787
03AB E8DF03             788             CALL    SW2_OFFS
03AE 7227               789             JC      G4
03B0 03D8               790             ADD     BX,AX
                        791
03B2 268B07             792             MOV     AX,ES:[BX]              ; MAX NUMBER OF CYLINDERS
03B5 2D0200             793             SUB     AX,2                    ; ADJUST FOR 0-N
                        794                                             ; AND RESERVE LAST TRACK
03B8 8AE8               795             MOV     CH,AL
03BA 250003             796             AND     AX,0300H                ; HIGH TWO BITS OF CYL
03BD D1E8               797             SHR     AX,1
03BF D1E8               798             SHR     AX,1
03C1 0C11               799             OR      AL,011H                 ; SECTORS
03C3 8AC8               800             MOV     CL,AL
                        801
03C5 268A7702           802             MOV     DH,ES:[BX][2]           ; HEADS
03C9 FECE               803             DEC     DH                      ; 0-N RANGE
03CB 8A167500           804             MOV     DL,HF_NUM               ; DRIVE COUNT
03CF 2BC0               805             SUB     AX,AX
03D1                    806     G5:
03D1 5B                 807             POP     BX                      ; RESTORE REGISTERS
03D2 07                 808             POP     ES
03D3 1F                 809             POP     DS
03D4 CA0200             810             RET     2
03D7                    811     G4:
03D7 C606740007         812             MOV     DISK_STATUS,INIT_FAIL   ; OPERATION FAILED
03DC B407               813             MOV     AH,INIT_FAIL
03DE 2AC0               814             SUB     AL,AL
03E0 2BD2               815             SUB     DX,DX
03E2 2BC9               816             SUB     CX,CX
03E4 F9                 817             STC                             ; SET ERROR FLAG
03E5 EBEA               818             JMP     G5
                        819     GET_PARM        ENDP
                        820
                        821     ;-------------------------------------------------------
                        822     ; INITIALIZE DRIVE CHARACTERISTICS                     :
                        823     ;                                                      :
                        824     ; FIXED DISK PARAMETER TABLE                           :
                        825     ;                                                      :
                        826     ;  -  THE TABLE IS COMPOSED OF A BLOCK DEFINED AS:    :
                        827     ;                                                      :
                        828     ;       (1 WORD) - MAXIMUM NUMBER OF CYLINDERS         :
                        829     ;       (1 BYTE) - MAXIMUM NUMBER OF HEADS             :
                        830     ;       (1 WORD) - STARTING REDUCED WRITE CURRENT CYL  :
                        831     ;       (1 WORD) - STARTING WRITE PRECOMPENSATION CYL  :
                        832     ;       (1 BYTE) - MAXIMUM ECC DATA BURST LENGTH       :
                        833     ;       (1 BYTE) - CONTROL BYTE (DRIVE STEP OPTION)    :
                        834     ;                  BIT    7 DISABLE DISK-ACCESS RETRIES :
                        835     ;                  BIT    6 DISABLE ECC RETRIES         :
                        836     ;                  BITS 5-3 ZERO                        :
                        837     ;                  BITS 2-0 DRIVE OPTION                :
                        838     ;       (1 BYTE) - STANDARD TIME OUT VALUE (SEE BELOW) :
                        839     ;       (1 BYTE) - TIME OUT VALUE FOR FORMAT DRIVE     :
                        840     ;       (1 BYTE) - TIME OUT VALUE FOR CHECK DRIVE      :
                        841     ;       (4 BYTES)                                      :
```

```
LOC OBJ                 LINE    SOURCE

                        842     ;               - RESERVED FOR FUTURE USE               :
                        843     ;                                                       :
                        844     ;       - TO DYNAMICALLY DEFINE A SET OF PARAMETERS     :
                        845     ;         BUILD A TABLE OF VALUES AND PLACE THE         :
                        846     ;         CORRESPONDING VECTOR INTO INTERRUPT 41.       :
                        847     ;                                                       :
                        848     ;       NOTE:                                           :
                        849     ;               THE DEFAULT TABLE IS VECTORED IN FOR    :
                        850     ;               AN INTERRUPT 19H (BOOTSTRAP)            :
                        851     ;                                                       :
                        852     ;                                                       :
                        853     ; ON THE CARD SWITCH SETTINGS                           :
                        854     ;                                                       :
                        855     ;                 DRIVE 0    DRIVE 1                    :
                        856     ;                ------------------------               :
                        857     ;         ON     :          /          :                :
                        858     ;                : -1-  -2- / -3-  -4- :                :
                        859     ;         OFF    :          /          :                :
                        860     ;                ------------------------               :
                        861     ;                                                       :
                        862     ;                                                       :
                        863     ;       TRANSLATION TABLE                               :
                        864     ;                                                       :
                        865     ;       1/3  :  2/4  :  TABLE ENTRY                     :
                        866     ;       ---------------------------                     :
                        867     ;        ON  :   ON   :      0                          :
                        868     ;        ON  :  OFF   :      1                          :
                        869     ;       OFF  :   ON   :      2                          :
                        870     ;       OFF  :  OFF   :      3                          :
                        871     ;                                                       :
                        872     ;-------------------------------------------------------
                        873
03E7                    874     FD_TBL:
                        875
                        876     ;----- DRIVE TYPE 00
                        877
03E7 3201               878             DW      0306D
03E9 02                 879             DB      02D
03EA 3201               880             DW      0306D
03EC 0000               881             DW      0000D
03EE 0B                 882             DB      0BH
03EF 00                 883             DB      00H
03F0 0C                 884             DB      0CH                     ; STANDARD
03F1 B4                 885             DB      0B4H                    ; FORMAT DRIVE
03F2 28                 886             DB      028H                    ; CHECK DRIVE
03F3 00000000           887             DB      0,0,0,0
                        888
                        889     ;----- DRIVE TYPE 01
                        890
03F7 7701               891             DW      0375D
03F9 08                 892             DB      08D
03FA 7701               893             DW      0375D
03FC 0000               894             DW      0000D
03FE 0B                 895             DB      0BH
03FF 05                 896             DB      05H
0400 0C                 897             DB      0CH                     ; STANDARD
0401 B4                 898             DB      0B4H                    ; FORMAT DRIVE
0402 28                 899             DB      028H                    ; CHECK DRIVE
0403 00000000           900             DB      0,0,0,0
                        901
                        902     ;----- DRIVE TYPE 02
                        903
0407 3201               904             DW      0306D
0409 06                 905             DB      06D
040A 8000               906             DW      0128D
040C 0001               907             DW      0256D
040E 0B                 908             DB      0BH
040F 05                 909             DB      05H
0410 0C                 910             DB      0CH                     ; STANDARD
0411 B4                 911             DB      0B4H                    ; FORMAT DRIVE
0412 28                 912             DB      028H                    ; CHECK DRIVE
0413 00000000           913             DB      0,0,0,0
                        914
                        915     ;----- DRIVE TYPE 03
                        916
0417 3201               917             DW      0306D
0419 04                 918             DB      04D
```

```
LOC OBJ                 LINE    SOURCE

041A 3201               919             DW      0306D
041C 0000               920             DW      0000D
041E 0B                 921             DB      0BH
041F 05                 922             DB      05H
0420 0C                 923             DB      0CH                     ; STANDARD
0421 B4                 924             DB      0B4H                    ; FORMAT DRIVE
0422 28                 925             DB      028H                    ; CHECK DRIVE
0423 00000000           926             DB      0,0,0,0
                        927
0427                    928     INIT_DRV        PROC    NEAR
                        929
                        930     ;----- DO DRIVE ZERO
                        931
0427 C60642000C         932             MOV     CMD_BLOCK+0,INIT_DRV_CMD
042C C606430000         933             MOV     CMD_BLOCK+1,0
0431 E81000             934             CALL    INIT_DRV_R
0434 720D               935             JC      INIT_DRV_OUT
                        936
                        937     ;----- DO DRIVE ONE
                        938
0436 C60642000C         939             MOV     CMD_BLOCK+0,INIT_DRV_CMD
043B C606430020         940             MOV     CMD_BLOCK+1,00100000B
0440 E80100             941             CALL    INIT_DRV_R
0443                    942     INIT_DRV_OUT:
0443 C3                 943             RET
                        944     INIT_DRV        ENDP
                        945
0444                    946     INIT_DRV_R      PROC    NEAR
                        947             ASSUME  ES:CODE
0444 2AC0               948             SUB     AL,AL
0446 E81901             949             CALL    COMMAND                 ; ISSUE THE COMMAND
0449 7301               950             JNC     B1
044B C3                 951             RET
044C                    952     B1:
044C 1E                 953             PUSH    DS                      ; SAVE SEGMENT
                        954             ASSUME  DS:DUMMY
044D 2BC0               955             SUB     AX,AX
044F 8ED8               956             MOV     DS,AX                   ; ESTABLISH SEGMENT
0451 C41E0401           957             LES     BX,HF_TBL_VEC
0455 1F                 958             POP     DS                      ; RESTORE SEGMENT
                        959             ASSUME  DS:DATA
0456 E83403             960             CALL    SW2_OFFS
0459 7257               961             JC      B3
045B 03D8               962             ADD     BX,AX
                        963
                        964     ;----- SEND DRIVE PARAMETERS MOST SIGNIFICANT BYTE FIRST
                        965
045D BF0100             966             MOV     DI,1
0460 E85F00             967             CALL    INIT_DRV_S
0463 724D               968             JC      B3
                        969
0465 BF0000             970             MOV     DI,0
0468 E85700             971             CALL    INIT_DRV_S
046B 7245               972             JC      B3
                        973
046D BF0200             974             MOV     DI,2
0470 E84F00             975             CALL    INIT_DRV_S
0473 723D               976             JC      B3
                        977
0475 BF0400             978             MOV     DI,4
0478 E84700             979             CALL    INIT_DRV_S
047B 7235               980             JC      B3
                        981
047D BF0300             982             MOV     DI,3
0480 E83F00             983             CALL    INIT_DRV_S
0483 722D               984             JC      B3
                        985
0485 BF0600             986             MOV     DI,6
0488 E83700             987             CALL    INIT_DRV_S
048B 7225               988             JC      B3
                        989
048D BF0500             990             MOV     DI,5
0490 E82F00             991             CALL    INIT_DRV_S
0493 721D               992             JC      B3
                        993
0495 BF0700             994             MOV     DI,7
0498 E82700             995             CALL    INIT_DRV_S
```

```
LOC OBJ                 LINE    SOURCE

049B 7215               996             JC      B3
                        997
049D BF0800             998             MOV     DI,8                    ; DRIVE STEP OPTION
04A0 268A01             999             MOV     AL,ES:[BX + DI]
04A3 A27600            1000             MOV     CONTROL_BYTE,AL
                       1001
04A6 2BC9              1002             SUB     CX,CX
04A8                   1003     B5:
04A8 E8D302            1004             CALL    PORT_1
04AB EC                1005             IN      AL,DX
04AC A802              1006             TEST    AL,R1_IOMODE            ; STATUS INPUT MODE
04AE 7509              1007             JNZ     B6
04B0 E2F6              1008             LOOP    B5
04B2                   1009     B3:
04B2 C606740007        1010             MOV     DISK_STATUS,INIT_FAIL   ; OPERATION FAILED
04B7 F9                1011             STC
04B8 C3                1012             RET
                       1013
04B9                   1014     B6:
04B9 E8B502            1015             CALL    PORT_0
04BC EC                1016             IN      AL,DX
04BD 2402              1017             AND     AL,2                    ; MASK ERROR BIT
04BF 75F1              1018             JNZ     B3
04C1 C3                1019             RET
                       1020             ASSUME  ES:NOTHING
                       1021     INIT_DRV_R      ENDP
                       1022
                       1023     ;----- SEND THE BYTE OUT TO THE CONTROLLER
                       1024
04C2                   1025     INIT_DRV_S      PROC    NEAR
04C2 E8C501            1026             CALL    HD_WAIT_REQ
04C5 7207              1027             JC      D1
04C7 E8A702            1028             CALL    PORT_0
04CA 268A01            1029             MOV     AL,ES:[BX + DI]
04CD EE                1030             OUT     DX,AL
04CE                   1031     D1:
04CE C3                1032             RET
                       1033     INIT_DRV_S      ENDP
                       1034
                       1035     ;---------------------------------------
                       1036     ;       READ LONG  (AH = 0AH)           :
                       1037     ;---------------------------------------
                       1038
04CF                   1039     RD_LONG         PROC    NEAR
04CF E81900            1040             CALL    CHK_LONG
04D2 726B              1041             JC      G8
04D4 C6064200E5        1042             MOV     CMD_BLOCK+0,RD_LONG_CMD
04D9 B047              1043             MOV     AL,DMA_READ
04DB EB68              1044             JMP     SHORT   DMA_OPN
                       1045     RD_LONG         ENDP
                       1046
                       1047     ;---------------------------------------
                       1048     ;       WRITE LONG  (AH = 0BH)          :
                       1049     ;---------------------------------------
                       1050
04DD                   1051     WR_LONG         PROC    NEAR
04DD E80B00            1052             CALL    CHK_LONG
04E0 725D              1053             JC      G8
04E2 C6064200E6        1054             MOV     CMD_BLOCK+0,WR_LONG_CMD
04E7 B04B              1055             MOV     AL,DMA_WRITE
04E9 EB5A              1056             JMP     SHORT   DMA_OPN
                       1057     WR_LONG         ENDP
                       1058
04EB                   1059     CHK_LONG        PROC    NEAR
04EB A04600            1060             MOV     AL,CMD_BLOCK+4
04EE 3C80              1061             CMP     AL,080H
04F0 F5                1062             CMC
04F1 C3                1063             RET
                       1064     CHK_LONG        ENDP
                       1065
                       1066     ;---------------------------------------
                       1067     ;       SEEK   (AH = 0CH)               :
                       1068     ;---------------------------------------
                       1069
04F2                   1070     DISK_SEEK       PROC    NEAR
04F2 C60642000B        1071             MOV     CMD_BLOCK,SEEK_CMD
04F7 EB3D              1072             JMP     SHORT   NDMA_OPN
```

```
LOC OBJ                 LINE    SOURCE

                        1073    DISK_SEEK       ENDP
                        1074
                        1075    ;--------------------------------------------
                        1076    ;       READ SECTOR BUFFER   (AH = 0EH)      :
                        1077    ;--------------------------------------------
                        1078
04F9                    1079    RD_BUFF PROC    NEAR
04F9 C60642000E         1080            MOV     CMD_BLOCK+0,RD_BUFF_CMD
04FE C606460001         1081            MOV     CMD_BLOCK+4,1           ; ONLY ONE BLOCK
0503 B047               1082            MOV     AL,DMA_READ
0505 EB3E               1083            JMP     SHORT   DMA_OPN
                        1084    RD_BUFF ENDP
                        1085
                        1086    ;--------------------------------------------
                        1087    ;       WRITE SECTOR BUFFER   (AH = 0FH)     :
                        1088    ;--------------------------------------------
                        1089
0507                    1090    WR_BUFF PROC    NEAR
0507 C60642000F         1091            MOV     CMD_BLOCK+0,WR_BUFF_CMD
050C C606460001         1092            MOV     CMD_BLOCK+4,1           ; ONLY ONE BLOCK
0511 B04B               1093            MOV     AL,DMA_WRITE
0513 EB30               1094            JMP     SHORT   DMA_OPN
                        1095    WR_BUFF ENDP
                        1096
                        1097    ;--------------------------------------------
                        1098    ;       TEST DISK READY   (AH = 010H)        :
                        1099    ;--------------------------------------------
                        1100
0515                    1101    TST_RDY PROC    NEAR
0515 C606420000         1102            MOV     CMD_BLOCK+0,TST_RDY_CMD
051A EB1A               1103            JMP     SHORT   NDMA_OPN
                        1104    TST_RDY ENDP
                        1105
                        1106    ;--------------------------------------------
                        1107    ;       RECALIBRATE   (AH = 011H)            :
                        1108    ;--------------------------------------------
                        1109
051C                    1110    HDISK_RECAL     PROC    NEAR
051C C606420001         1111            MOV     CMD_BLOCK,RECAL_CMD
0521 EB13               1112            JMP     SHORT   NDMA_OPN
                        1113    HDISK_RECAL     ENDP
                        1114
                        1115    ;-------------------------------------------------------
                        1116    ;       CONTROLLER RAM DIAGNOSTICS   (AH = 012H)     :
                        1117    ;-------------------------------------------------------
                        1118
0523                    1119    RAM_DIAG        PROC    NEAR
0523 C6064200E0         1120            MOV     CMD_BLOCK+0,RAM_DIAG_CMD
0528 EB0C               1121            JMP     SHORT   NDMA_OPN
                        1122    RAM_DIAG        ENDP
                        1123
                        1124    ;--------------------------------------------
                        1125    ;       DRIVE DIAGNOSTICS   (AH = 013H)      :
                        1126    ;--------------------------------------------
                        1127
052A                    1128    CHK_DRV PROC    NEAR
052A C6064200E3         1129            MOV     CMD_BLOCK+0,CHK_DRV_CMD
052F EB05               1130            JMP     SHORT   NDMA_OPN
                        1131    CHK_DRV ENDP
                        1132
                        1133    ;-----------------------------------------------------------
                        1134    ;      CONTROLLER INTERNAL DIAGNOSTICS   (AH = 014H)      :
                        1135    ;-----------------------------------------------------------
                        1136
0531                    1137    CNTLR_DIAG      PROC    NEAR
0531 C6064200E4         1138            MOV     CMD_BLOCK+0,CNTLR_DIAG_CMD
                        1139    CNTLR_DIAG      ENDP
                        1140
                        1141    ;--------------------------------------------------
                        1142    ;                   SUPPORT ROUTINES               :
                        1143    ;--------------------------------------------------
                        1144
0536                    1145    NDMA_OPN:
0536 B002               1146            MOV     AL,02H
0538 E82700             1147            CALL    COMMAND                 ; ISSUE THE COMMAND
053B 7221               1148            JC      G11
053D EB16               1149            JMP     SHORT   G3
```

```
LOC OBJ                 LINE    SOURCE

053F                    1150    G8:
053F C606740009         1151            MOV     DISK_STATUS,DMA_BOUNDARY
0544 C3                 1152            RET
0545                    1153    DMA_OPN:
0545 E85701             1154            CALL    DMA_SETUP               ; SET UP FOR DMA OPERATION
0548 72F5               1155            JC      G8
054A B003               1156            MOV     AL,03H
054C E81300             1157            CALL    COMMAND                 ; ISSUE THE COMMAND
054F 720D               1158            JC      G11
0551 B003               1159            MOV     AL,03H
0553 E60A               1160            OUT     DMA+10,AL               ; INITIALIZE THE DISK CHANNEL
0555                    1161    G3:
0555 E421               1162            IN      AL,021H
0557 24DF               1163            AND     AL,0DFH
0559 E621               1164            OUT     021H,AL
055B E8AA01             1165            CALL    WAIT_INT
055E                    1166    G11:
055E E83B00             1167            CALL    ERROR_CHK
0561 C3                 1168            RET
                        1169
                        1170    ;-------------------------------------------------------
                        1171    ; COMMAND                                              :
                        1172    ;       THIS ROUTINE OUTPUTS THE COMMAND BLOCK          :
                        1173    ; INPUT                                                :
                        1174    ;       AL = CONTROLLER DMA/INTERRUPT REGISTER MASK     :
                        1175    ;                                                      :
                        1176    ;-------------------------------------------------------
                        1177
0562                    1178    COMMAND PROC    NEAR
0562 BE4200             1179            MOV     SI,OFFSET CMD_BLOCK
0565 E81B02             1180            CALL    PORT_2
0568 EE                 1181            OUT     DX,AL                   ; CONTROLLER SELECT PULSE
0569 E81C02             1182            CALL    PORT_3
056C EE                 1183            OUT     DX,AL
056D 2BC9               1184            SUB     CX,CX                   ; WAIT COUNT
056F E80C02             1185            CALL    PORT_1
0572                    1186    WAIT_BUSY:
0572 EC                 1187            IN      AL,DX                   ; GET STATUS
0573 240F               1188            AND     AL,0FH
0575 3C0D               1189            CMP     AL,R1_BUSY OR R1_BUS OR R1_REQ
0577 7409               1190            JE      C1
0579 E2F7               1191            LOOP    WAIT_BUSY
057B C606740080         1192            MOV     DISK_STATUS,TIME_OUT
0580 F9                 1193            STC
0581 C3                 1194            RET                             ; ERROR RETURN
0582                    1195    C1:
0582 FC                 1196            CLD
0583 B90600             1197            MOV     CX,6                    ; BYTE COUNT
0586                    1198    CM3:
0586 E8E801             1199            CALL    PORT_0
0589 AC                 1200            LODSB                           ; GET THE NEXT COMMAND BYTE
058A EE                 1201            OUT     DX,AL                   ; OUT IT GOES
058B E2F9               1202            LOOP    CM3                     ; DO MORE
                        1203
058D E8EE01             1204            CALL    PORT_1                  ; STATUS
0590 EC                 1205            IN      AL,DX
0591 A801               1206            TEST    AL,R1_REQ
0593 7406               1207            JZ      CM7
0595 C606740020         1208            MOV     DISK_STATUS,BAD_CNTLR
059A F9                 1209            STC
059B                    1210    CM7:
059B C3                 1211            RET
                        1212    COMMAND ENDP
                        1213
                        1214    ;-------------------------------------------------
                        1215    ;               SENSE STATUS BYTES              :
                        1216    ;                                               :
                        1217    ; BYTE 0                                        :
                        1218    ;       BIT     7   ADDRESS VALID, WHEN SET     :
                        1219    ;       BIT     6   SPARE, SET TO ZERO          :
                        1220    ;       BITS 5-4    ERROR TYPE                  :
                        1221    ;       BITS 3-0    ERROR CODE                  :
                        1222    ;                                               :
                        1223    ; BYTE 1                                        :
                        1224    ;       BITS 7-6    ZERO                        :
                        1225    ;       BIT     5   DRIVE (0-1)                 :
                        1226    ;       BITS 4-0    HEAD NUMBER                 :
```

```
LOC OBJ                 LINE    SOURCE

                        1227    ;                                               :
                        1228    ; BYTE 2                                        :
                        1229    ;       BITS 7-5   CYLINDER HIGH                :
                        1230    ;       BITS 4-0   SECTOR NUMBER                :
                        1231    ;                                               :
                        1232    ; BYTE 3                                        :
                        1233    ;       BITS 7-0   CYLINDER LOW                 :
                        1234    ;                                               :
                        1235    ;-----------------------------------------------
                        1236
059C                    1237    ERROR_CHK       PROC    NEAR
                        1238            ASSUME  ES:DATA
059C A07400             1239            MOV     AL,DISK_STATUS          ; CHECK IF THERE WAS AN ERROR
059F 0AC0               1240            OR      AL,AL
05A1 7501               1241            JNZ     G21
05A3 C3                 1242            RET
                        1243
                        1244    ;----- PERFORM SENSE STATUS
                        1245
05A4                    1246    G21:
05A4 B84000             1247            MOV     AX,DATA
05A7 8EC0               1248            MOV     ES,AX                   ; ESTABLISH SEGMENT
05A9 2BC0               1249            SUB     AX,AX
05AB 8BF8               1250            MOV     DI,AX
05AD C606420003         1251            MOV     CMD_BLOCK+0,SENSE_CMD
05B2 2AC0               1252            SUB     AL,AL
05B4 E8ABFF             1253            CALL    COMMAND                 ; ISSUE SENSE STATUS COMMAND
05B7 7223               1254            JC      SENSE_ABORT             ; CANNOT RECOVER
05B9 B90400             1255            MOV     CX,4
05BC                    1256    G22:
05BC E8CB00             1257            CALL    HD_WAIT_REQ
05BF 7220               1258            JC      G24
05C1 E8AD01             1259            CALL    PORT_0
05C4 EC                 1260            IN      AL,DX
05C5 26884542           1261            MOV     ES:HD_ERROR[DI],AL      ; STORE AWAY SENSE BYTES
05C9 47                 1262            INC     DI
05CA E8B101             1263            CALL    PORT_1
05CD E2ED               1264            LOOP    G22
05CF E8B800             1265            CALL    HD_WAIT_REQ
05D2 720D               1266            JC      G24
05D4 E89A01             1267            CALL    PORT_0
05D7 EC                 1268            IN      AL,DX
05D8 A802               1269            TEST    AL,2
05DA 740F               1270            JZ      STAT_ERR
05DC                    1271    SENSE_ABORT:
05DC C6067400FF         1272            MOV     DISK_STATUS,SENSE_FAIL
05E1                    1273    G24:
05E1 F9                 1274            STC
05E2 C3                 1275            RET
                        1276    ERROR_CHK       ENDP
                        1277
05E3 1A06               1278    T_0     DW      TYPE_0
05E5 2706               1279    T_1     DW      TYPE_1
05E7 6A06               1280    T_2     DW      TYPE_2
05E9 7706               1281    T_3     DW      TYPE_3
                        1282
05EB                    1283    STAT_ERR:
05EB 268A1E4200         1284            MOV     BL,ES:HD_ERROR          ; GET ERROR BYTE
05F0 8AC3               1285            MOV     AL,BL
05F2 240F               1286            AND     AL,0FH
05F4 80E330             1287            AND     BL,00110000B            ; ISOLATE TYPE
05F7 2AFF               1288            SUB     BH,BH
05F9 B103               1289            MOV     CL,3
05FB D3EB               1290            SHR     BX,CL                   ; ADJUST
05FD 2EFFA7E305         1291            JMP     WORD PTR CS:[BX + OFFSET T_0]
                        1292            ASSUME ES:NOTHING
                        1293
0602                    1294    TYPE0_TABLE     LABEL   BYTE
0602 00204020800020     1295            DB      0,BAD_CNTLR,BAD_SEEK,BAD_CNTLR,TIME_OUT,0,BAD_CNTLR
0609 0040               1296            DB      0,BAD_SEEK
  0009                  1297    TYPE0_LEN       EQU     $-TYPE0_TABLE
060B                    1298    TYPE1_TABLE     LABEL   BYTE
060B 1010020004         1299            DB      BAD_ECC,BAD_ECC,BAD_ADDR_MARK,0,RECORD_NOT_FND
0610 400000110B         1300            DB      BAD_SEEK,0,0,DATA_CORRECTED,BAD_TRACK
  000A                  1301    TYPE1_LEN       EQU     $-TYPE1_TABLE
0615                    1302    TYPE2_TABLE     LABEL   BYTE
0615 0102               1303            DB      BAD_CMD,BAD_ADDR_MARK
```

```
LOC OBJ                 LINE    SOURCE

  0002                  1304    TYPE2_LEN       EQU     $-TYPE2_TABLE
0617                    1305    TYPE3_TABLE     LABEL   BYTE
0617 202010             1306            DB      BAD_CNTLR,BAD_CNTLR,BAD_ECC
  0003                  1307    TYPE3_LEN       EQU     $-TYPE3_TABLE
                        1308
                        1309    ;----- TYPE 0 ERROR
                        1310
061A                    1311    TYPE_0:
061A BB0206             1312            MOV     BX,OFFSET TYPE0_TABLE
061D 3C09               1313            CMP     AL,TYPE0_LEN            ; CHECK IF ERROR IS DEFINED
061F 7363               1314            JAE     UNDEF_ERR_L
0621 2ED7               1315            XLAT    CS:TYPE0_TABLE          ; TABLE LOOKUP
0623 A27400             1316            MOV     DISK_STATUS,AL          ; SET ERROR CODE
0626 C3                 1317            RET
                        1318
                        1319    ;------ TYPE 1 ERROR
                        1320
0627                    1321    TYPE_1:
0627 BB0B06             1322            MOV     BX,OFFSET TYPE1_TABLE
062A 8BC8               1323            MOV     CX,AX
062C 3C0A               1324            CMP     AL,TYPE1_LEN            ; CHECK IF ERROR IS DEFINED
062E 7354               1325            JAE     UNDEF_ERR_L
0630 2ED7               1326            XLAT    CS:TYPE1_TABLE          ; TABLE LOOKUP
0632 A27400             1327            MOV     DISK_STATUS,AL          ; SET ERROR CODE
0635 80E108             1328            AND     CL,08H                  ; CORRECTED ECC
0638 80F908             1329            CMP     CL,08H
063B 752A               1330            JNZ     G30
                        1331
                        1332    ;----- OBTAIN ECC ERROR BURST LENGTH
                        1333
063D C606420000         1334            MOV     CMD_BLOCK+0,RD_ECC_CMD
0642 2AC0               1335            SUB     AL,AL
0644 E81BFF             1336            CALL    COMMAND
0647 721E               1337            JC      G30
0649 E83E00             1338            CALL    HD_WAIT_REQ
064C 7219               1339            JC      G30
064E E82001             1340            CALL    PORT_0
0651 EC                 1341            IN      AL,DX
0652 8AC8               1342            MOV     CL,AL
0654 E83300             1343            CALL    HD_WAIT_REQ
0657 720E               1344            JC      G30
0659 E81501             1345            CALL    PORT_0
065C EC                 1346            IN      AL,DX
065D A801               1347            TEST    AL,01H
065F 7406               1348            JZ      G30
0661 C606740020         1349            MOV     DISK_STATUS,BAD_CNTLR
0666 F9                 1350            STC
0667                    1351    G30:
0667 8AC1               1352            MOV     AL,CL
0669 C3                 1353            RET
                        1354
                        1355    ;----- TYPE 2 ERROR
                        1356
066A                    1357    TYPE_2:
066A BB1506             1358            MOV     BX,OFFSET TYPE2_TABLE
066D 3C02               1359            CMP     AL,TYPE2_LEN            ; CHECK IF ERROR IS DEFINED
066F 7313               1360            JAE     UNDEF_ERR_L
0671 2ED7               1361            XLAT    CS:TYPE1_TABLE          ; TABLE LOOKUP
0673 A27400             1362            MOV     DISK_STATUS,AL          ; SET ERROR CODE
0676 C3                 1363            RET
                        1364
                        1365    ;----- TYPE 3 ERROR
                        1366
0677                    1367    TYPE_3:
0677 BB1706             1368            MOV     BX,OFFSET TYPE3_TABLE
067A 3C03               1369            CMP     AL,TYPE3_LEN
067C 7306               1370            JAE     UNDEF_ERR_L
067E 2ED7               1371            XLAT    CS:TYPE3_TABLE
0680 A27400             1372            MOV     DISK_STATUS,AL
0683 C3                 1373            RET
                        1374
0684                    1375    UNDEF_ERR_L:
0684 C6067400BB         1376            MOV     DISK_STATUS,UNDEF_ERR
0689 C3                 1377            RET
                        1378
068A                    1379    HD_WAIT_REQ     PROC    NEAR
068A 51                 1380            PUSH    CX
```

```
LOC OBJ                 LINE    SOURCE

068B 2BC9               1381            SUB     CX,CX
068D E8EE00             1382            CALL    PORT_1
0690                    1383    L1:
0690 EC                 1384            IN      AL,DX
0691 A801               1385            TEST    AL,R1_REQ
0693 7508               1386            JNZ     L2
0695 E2F9               1387            LOOP    L1
0697 C606740080         1388            MOV     DISK_STATUS,TIME_OUT
069C F9                 1389            STC
069D                    1390    L2:
069D 59                 1391            POP     CX
069E C3                 1392            RET
                        1393    HD_WAIT_REQ     ENDP
                        1394
                        1395    ;--------------------------------------------------------
                        1396    ; DMA_SETUP                                            :
                        1397    ;       THIS ROUTINE SETS UP FOR DMA OPERATIONS.       :
                        1398    ; INPUT                                                :
                        1399    ;       (AL) = MODE BYTE FOR THE DMA                   :
                        1400    ;       (ES:BX) = ADDRESS TO READ/WRITE THE DATA       :
                        1401    ; OUTPUT                                               :
                        1402    ;       (AX) DESTROYED                                 :
                        1403    ;--------------------------------------------------------
069F                    1404    DMA_SETUP       PROC    NEAR
069F 50                 1405            PUSH    AX
06A0 A04600             1406            MOV     AL,CMD_BLOCK+4
06A3 3C81               1407            CMP     AL,81H                  ; BLOCK COUNT OUT OF RANGE
06A5 58                 1408            POP     AX
06A6 7202               1409            JB      J1
06A8 F9                 1410            STC
06A9 C3                 1411            RET
06AA                    1412    J1:
06AA 51                 1413            PUSH    CX                      ; SAVE THE REGISTER
06AB FA                 1414            CLI                             ; NO MORE INTERRUPTS
06AC E60C               1415            OUT     DMA+12,AL               ; SET THE FIRST/LAST F/F
06AE 50                 1416            PUSH    AX
06AF 58                 1417            POP     AX
06B0 E60B               1418            OUT     DMA+11,AL               ; OUTPUT THE MODE BYTE
06B2 8CC0               1419            MOV     AX,ES                   ; GET THE ES VALUE
06B4 B104               1420            MOV     CL,4                    ; SHIFT COUNT
06B6 D3C0               1421            ROL     AX,CL                   ; ROTATE LEFT
06B8 8AE8               1422            MOV     CH,AL                   ; GET HIGHEST NYBBLE OF ES TO CH
06BA 24F0               1423            AND     AL,0F0H                 ; ZERO THE LOW NYBBLE FROM SEGMENT
06BC 03C3               1424            ADD     AX,BX                   ; TEST FOR CARRY FROM ADDITION
06BE 7302               1425            JNC     J33
06C0 FEC5               1426            INC     CH                      ; CARRY MEANS HIGH 4 BITS MUST BE INC
06C2                    1427    J33:
06C2 50                 1428            PUSH    AX                      ; SAVE START ADDRESS
06C3 E606               1429            OUT     DMA+6,AL                ; OUTPUT LOW ADDRESS
06C5 8AC4               1430            MOV     AL,AH
06C7 E606               1431            OUT     DMA+6,AL                ; OUTPUT HIGH ADDRESS
06C9 8AC5               1432            MOV     AL,CH                   ; GET HIGH 4 BITS
06CB 240F               1433            AND     AL,0FH
06CD E682               1434            OUT     DMA_HIGH,AL             ; OUTPUT THE HIGH 4 BITS TO PAGE REG
                        1435
                        1436    ;------ DETERMINE COUNT
                        1437
06CF A04600             1438            MOV     AL,CMD_BLOCK+4          ; RECOVER BLOCK COUNT
06D2 D0E0               1439            SHL     AL,1                    ; MULTIPLY BY 512 BYTES PER SECTOR
06D4 FEC8               1440            DEC     AL                      ;   AND DECREMENT VALUE BY ONE
06D6 8AE0               1441            MOV     AH,AL
06D8 B0FF               1442            MOV     AL,0FFH
                        1443
                        1444    ;----- HANDLE READ AND WRITE LONG (516D BYTE BLOCKS)
                        1445
06DA 50                 1446            PUSH    AX                      ; SAVE REGISTER
06DB A04200             1447            MOV     AL,CMD_BLOCK+0          ; GET COMMAND
06DE 3CE5               1448            CMP     AL,RD_LONG_CMD
06E0 7407               1449            JE      ADD4
06E2 3CE6               1450            CMP     AL,WR_LONG_CMD
06E4 7403               1451            JE      ADD4
06E6 58                 1452            POP     AX                      ; RESTORE REGISTER
06E7 EB11               1453            JMP     SHORT   J20
06E9                    1454    ADD4:
06E9 58                 1455            POP     AX                      ; RESTORE REGISTER
06EA B80402             1456            MOV     AX,516D                 ; ONE BLOCK (512) PLUS 4 BYTES ECC
06ED 53                 1457            PUSH    BX
```

```
LOC OBJ                 LINE    SOURCE

06EE 2AFF               1458            SUB     BH,BH
06F0 8A1E4600           1459            MOV     BL,CMD_BLOCK+4
06F4 52                 1460            PUSH    DX
06F5 F7E3               1461            MUL     BX                      ; BLOCK COUNT TIMES 516
06F7 5A                 1462            POP     DX
06F8 5B                 1463            POP     BX
06F9 48                 1464            DEC     AX                      ; ADJUST
06FA                    1465    J20:
                        1466
06FA 50                 1467            PUSH    AX                      ; SAVE COUNT VALUE
06FB E607               1468            OUT     DMA+7,AL                ; LOW BYTE OF COUNT
06FD 8AC4               1469            MOV     AL,AH
06FF E607               1470            OUT     DMA+7,AL                ; HIGH BYTE OF COUNT
0701 FB                 1471            STI                             ; INTERRUPTS BACK ON
0702 59                 1472            POP     CX                      ; RECOVER COUNT VALUE
0703 58                 1473            POP     AX                      ; RECOVER ADDRESS VALUE
0704 03C1               1474            ADD     AX,CX                   ; ADD, TEST FOR 64K OVERFLOW
0706 59                 1475            POP     CX                      ; RECOVER REGISTER
0707 C3                 1476            RET                     ; RETURN TO CALLER, CFL SET BY ABOVE IF ERROR
                        1477    DMA_SETUP       ENDP
                        1478
                        1479    ;-----------------------------------------------
                        1480    ; WAIT_INT                                    :
                        1481    ;       THIS ROUTINE WAITS FOR THE FIXED DISK   :
                        1482    ;       CONTROLLER TO SIGNAL THAT AN INTERRUPT  :
                        1483    ;       HAS OCCURRED.                           :
                        1484    ;-----------------------------------------------
0708                    1485    WAIT_INT        PROC    NEAR
0708 FB                 1486            STI                             ; TURN ON INTERRUPTS
0709 53                 1487            PUSH    BX                      ; PRESERVE REGISTERS
070A 51                 1488            PUSH    CX
070B 06                 1489            PUSH    ES
070C 56                 1490            PUSH    SI
070D 1E                 1491            PUSH    DS
                        1492            ASSUME  DS:DUMMY
070E 2BC0               1493            SUB     AX,AX
0710 8ED8               1494            MOV     DS,AX                   ; ESTABLISH SEGMENT
0712 C4360401           1495            LES     SI,HF_TBL_VEC
                        1496            ASSUME  DS:DATA
0716 1F                 1497            POP     DS
                        1498
                        1499    ;----- SET TIMEOUT VALUES
                        1500
0717 2AFF               1501            SUB     BH,BH
0719 268A5C09           1502            MOV     BL,BYTE PTR ES:[SI][9]          ; STANDARD TIME OUT
071D 8A264200           1503            MOV     AH,CMD_BLOCK
0721 80FC04             1504            CMP     AH,FMTDRV_CMD
0724 7506               1505            JNZ     W5
0726 268A5C0A           1506            MOV     BL,BYTE PTR ES:[SI][0AH]        ; FORMAT DRIVE
072A EB09               1507            JMP     SHORT   W4
072C 80FCE3             1508    W5:     CMP     AH,CHK_DRV_CMD
072F 7504               1509            JNZ     W4
0731 268A5C0B           1510            MOV     BL,BYTE PTR ES:[SI][0BH]        ; CHECK DRIVE
0735                    1511    W4:
0735 2BC9               1512            SUB     CX,CX
                        1513
                        1514    ;----- WAIT FOR INTERRUPT
                        1515
0737                    1516    W1:
0737 E84400             1517            CALL    PORT_1
073A EC                 1518            IN      AL,DX
073B 2420               1519            AND     AL,020H
073D 3C20               1520            CMP     AL,020H                 ; DID INTERRUPT OCCUR
073F 740A               1521            JZ      W2
0741 E2F4               1522            LOOP    W1                      ; INNER LOOP
0743 4B                 1523            DEC     BX
0744 75F1               1524            JNZ     W1                      ; OUTER LOOP
0746 C606740080         1525            MOV     DISK_STATUS,TIME_OUT
074B                    1526    W2:
074B E82300             1527            CALL    PORT_0
074E EC                 1528            IN      AL,DX
074F 2402               1529            AND     AL,2                    ; ERROR BIT
0751 08067400           1530            OR      DISK_STATUS,AL          ; SAVE
0755 E83000             1531            CALL    PORT_3                  ; INTERRUPT MASK REGISTER
0758 32C0               1532            XOR     AL,AL                   ; ZERO
075A EE                 1533            OUT     DX,AL                   ; RESET MASK
075B 5E                 1534            POP     SI                      ; RESTORE REGISTERS
```

```
LOC OBJ                 LINE    SOURCE

075C 07                 1535            POP     ES
075D 59                 1536            POP     CX
075E 5B                 1537            POP     BX
075F C3                 1538            RET
                        1539    WAIT_INT        ENDP
                        1540
0760                    1541    HD_INT  PROC    NEAR
0760 50                 1542            PUSH    AX
0761 B020               1543            MOV     AL,EOI                  ; END OF INTERRUPT
0763 E620               1544            OUT     INT_CTL_PORT,AL
0765 B007               1545            MOV     AL,07H                  ; SET DMA MODE TO DISABLE
0767 E60A               1546            OUT     DMA+10,AL
0769 E421               1547            IN      AL,021H
076B 0C20               1548            OR      AL,020H
076D E621               1549            OUT     021H,AL
076F 58                 1550            POP     AX
0770 CF                 1551            IRET
                        1552    HD_INT  ENDP
                        1553
                        1554    ;---------------------------------------
                        1555    ; PORTS                                 :
                        1556    ;       GENERATE PROPER PORT VALUE      :
                        1557    ;       BASED ON THE PORT OFFSET        :
                        1558    ;---------------------------------------
                        1559
0771                    1560    PORT_0  PROC    NEAR
0771 BA2003             1561            MOV     DX,HF_PORT              ; BASE VALUE
0774 50                 1562            PUSH    AX
0775 2AE4               1563            SUB     AH,AH
0777 A07700             1564            MOV     AL,PORT_OFF             ; ADD IN THE OFFSET
077A 03D0               1565            ADD     DX,AX
077C 58                 1566            POP     AX
077D C3                 1567            RET
                        1568    PORT_0  ENDP
                        1569
077E                    1570    PORT_1  PROC    NEAR
077E E8F0FF             1571            CALL    PORT_0
0781 42                 1572            INC     DX                      ; INCREMENT TO PORT ONE
0782 C3                 1573            RET
                        1574    PORT_1  ENDP
                        1575
0783                    1576    PORT_2  PROC    NEAR
0783 E8F8FF             1577            CALL    PORT_1
0786 42                 1578            INC     DX                      ; INCREMENT TO PORT TWO
0787 C3                 1579            RET
                        1580    PORT_2  ENDP
                        1581
0788                    1582    PORT_3  PROC    NEAR
0788 E8F8FF             1583            CALL    PORT_2
078B 42                 1584            INC     DX                      ; INCREMENT TO PORT THREE
078C C3                 1585            RET
                        1586    PORT_3  ENDP
                        1587
                        1588    ;-------------------------------------------------
                        1589    ; SW2_OFFS                                      :
                        1590    ;       DETERMINE PARAMETER TABLE OFFSET        :
                        1591    ;       USING CONTROLLER PORT TWO AND           :
                        1592    ;       DRIVE NUMBER SPECIFIER (0-1)            :
                        1593    ;-------------------------------------------------
                        1594
078D                    1595    SW2_OFFS        PROC    NEAR
078D E8F3FF             1596            CALL    PORT_2
0790 EC                 1597            IN      AL,DX                   ; READ PORT 2
0791 50                 1598            PUSH    AX
0792 E8E9FF             1599            CALL    PORT_1
0795 EC                 1600            IN      AL,DX
0796 2402               1601            AND     AL,2                    ; CHECK FOR ERROR
0798 58                 1602            POP     AX
0799 7516               1603            JNZ     SW2_OFFS_ERR
079B 8A264300           1604            MOV     AH,CMD_BLOCK+1
079F 80E420             1605            AND     AH,00100000B            ; DRIVE 0 OR 1
07A2 7504               1606            JNZ     SW2_AND
07A4 D0E8               1607            SHR     AL,1                    ; ADJUST
07A6 D0E8               1608            SHR     AL,1
07A8                    1609    SW2_AND:
07A8 2403               1610            AND     AL,011B                 ; ISOLATE
07AA B104               1611            MOV     CL,4
```

```
LOC OBJ                  LINE    SOURCE

07AC D2E0                1612            SHL     AL,CL                   ; ADJUST
07AE 2AE4                1613            SUB     AH,AH
07B0 C3                  1614            RET
07B1                     1615    SW2_OFFS_ERR:
07B1 F9                  1616            STC
07B2 C3                  1617            RET
                         1618    SW2_OFFS        ENDP
                         1619
07B3 30382F31362F38      1620            DB      '08/16/82'              ; RELEASE MARKER
     32
                         1621
07BB                     1622    END_ADDRESS     LABEL   BYTE
----                     1623    CODE    ENDS
                         1624            END
```

Notes:

Personal Computer Hardware Reference Library

Fixed Disk and Diskette Drive Adapter

Contents

Notes:

Description

The IBM Personal Computer AT Fixed Disk and Diskette Drive Adapter connects to the system board using one of the system expansion slots. The adapter controls the 5-1/4 inch diskette drives and fixed disk drives. Connectors on the adapter supply all the signals necessary to operate up to two fixed drives and one diskette drive or one fixed drive and two diskette drives. The adapter will allow concurrent data operations on one diskette and one fixed disk drive.

The adapter operates when connected to a system board expansion slot. This channel is described in the "System Board" section of the IBM Personal Computer AT *Technical Reference* Manual.

Fixed Disk Function

The fixed disk function features 512-byte sectors; high-speed, programmed input/output (PIO) data transfers; error correction code (ECC) correction of up to five bits on data fields; multiple sector operations across track and cylinder boundaries; and on-board diagnostic tests. The adapter will support two fixed disks with up to 16 read/write heads and 1024 cylinders.

Task File

A task file, which contains eight registers, controls fixed-disk operations. The following figure shows the addresses and functions of these registers.

I/O Address		Read	Write
Primary	Secondary		
1F0	170	Data Register	Data Register
1F1	171	Error Register	Write Precomp
1F2	172	Sector Count	Sector Count
1F3	173	Sector Number	Sector Number
1F4	174	Cylinder Low	Cylinder Low
1F5	175	Cylinder High	Cylinder High
1F6	176	Drive/Head	Drive/Head
1F7	177	Status Register	Command Register

Task File

Task File Registers

Data Register

The data register provides access to the sector buffer for read and write operations in the PIO mode. This register must not be accessed unless a Read or Write command is being executed. The register provides a 16-bit path into the sector buffer for normal Read and Write commands. When a R/W Long is issued, the 4 ECC bytes are transferred by byte with at least 2 microseconds between transfers. 'Data Request' (DRQ) must be active before the transferring of the ECC bytes.

Error Register

The error register is a read-only register that contains specific information related to the previous command. The data is valid only when the error bit in the status register is set, unless the adapter is in diagnostic mode. Diagnostic mode is the state immediately after power is switched on or after a Diagnose command. In these cases, the register must be checked regardless of the status register indicator. The following are bit values for the diagnostic mode.

Diagnostic Mode

01 No errors

02 Controller error

03 Sector buffer error

04 ECC device error

05 Control processor error

The following are bit definitions for the operational mode.

Operational Mode

Bit 0 Data Address Mark (DAM) Not Found—This bit indicates that DAM could not be found within 16 bytes of the ID field.

Bit 1 TR 000 Error—This bit will be set if, during a Restore command, the track 000 line from the fixed disk is not true within 1023 step pulses to the drive.

Bit 2 Aborted Command—A command is aborted based on the drive status (Write Fault, Not Seek Complete, Drive Not Ready, or an invalid command). The status and error registers may be decoded to determine the cause.

Bit 3 Not used.

Bit 4 ID Not Found—The ID field with the specified cylinder, head, and sector number could not be found. If retries are enabled, the controller attempts to read the ID 16 times before indicating the error. If retries are disabled, the track is scanned a maximum of two times before setting this error bit.

Bit 5 Not used

Bit 6 Data ECC Error—This bit indicates that an uncorrectable ECC error occurred in the target's data field during a read command.

Bit 7 Bad Block Detect—This bit indicates that the bad block mark was detected in the target's ID field. No Read or Write commands will be executed in any data fields marked bad.

Write Precompensation Register

The value in this register is the starting cylinder number divided by 4. The 'reduced write current' signal to the drive is activated and the adapter's write precompensation logic is turned on when this number is entered into the register.

Sector Count Register

The sector count register defines the number of sectors to be transferred during a Verify, Read, Write, or Format command. During a multi-sector operation, the sector count is decremented and the sector number is incremented. When the disk is being formatted, the number of sectors per track must be loaded into the register prior to each Format command. The adapter supports multi-sector transfers across track and cylinder boundaries. The drive characteristics must be set up by the Set Parameters command before initiating a multi-sector transfer. The sector count register must be loaded with the number of sectors to be transferred for any data-related command.

> **Note:** A 0 in the sector count register specifies a 256-sector transfer.

Sector Number Register

The target's logical sector number for Read, Write, and Verify commands is loaded into this register. The starting sector number is loaded into this register for multi-sector operations.

Cylinder Number Registers

The target number for Read, Write, Seek, and Verify commands is loaded into these registers as shown in the following figure. The cylinder-number registers address up to 1024 cylinders.

	Cylinder High	Cylinder Low
Register Bits	76543210	76543210
Cylinder Bits	———98	76543210

Cylinder Number Registers

Drive/Head Register

Bit 7 Set to 1

Bit 6 Set to 0

Bit 5 Set to 1

Bit 4 Drive Select—This bit selects the drive. A 0 indicates the first fixed disk drive, and a 1 indicates the second.

Bit 3–Bit 0 Head Select Bits—Bits 3 through 0 specify the desired read/write head. Bit 0 is the least-significant (0101 selects head 5). The adapter supports up to 16 read/write heads. For access to heads 8 through 15, bit 3 of the fixed disk register (address hex 3F6) must be set to 1.

Note: This register must be loaded with the maximum number of heads for each drive before a Set Parameters command is issued.

Status Register

The controller sets up the status register with the command status after execution. The program must look at this register to determine the result of any operation. If the busy bit is set, no other bits are valid. A read of the status register clears interrupt

request 14. If '-write fault' or 'error' is active, or if '-seek complete' or '-ready' is inactive, a multi-sector operation is aborted.

The following defines the bits of the status register:

Bit 7 Busy—This bit indicates the controller's status. A 1 indicates the controller is executing a command. If this bit is set, no other status register bit is valid, and the other registers reflect the status register's contents; therefore, the busy bit must be examined before any fixed disk register is read.

Bit 6 Drive Ready—A 1 on this bit together with a 1 on seek complete bit (bit 4) indicates that the fixed disk drive is ready to read, write, or seek. A 0 indicates that read, write, and seek are inhibited.

Bit 5 Write Fault—A 1 on this bit indicates improper operation of the drive; read, write, or seek is inhibited.

Bit 4 Seek Complete—A 1 on this bit indicates that the read/write heads have completed a seek operation.

Bit 3 Data Request—This bit indicates that the sector buffer requires servicing during a Read or Write command. If either bit 7 (busy) or this bit is active, a command is being executed. Upon receipt of any command, this bit is reset.

Bit 2 Corrected Data—A 1 on this bit indicates that the data read from the disk was successfully corrected by the ECC algorithm. Soft errors will not end multi-sector operations.

Bit 1 Index—This bit is set to 1 each revolution of the disk.

Bit 0 Error—A 1 on this bit indicates that the previous command ended in an error, and that one or more bits are set in the error register. The next command from the controller resets the error bit. This bit, when set, halts multi-sector operations.

Command Register

The command register accepts eight commands to perform fixed disk operations. Commands are executed by loading the task file and writing in the command register while the controller status is not busy. If '-write fault' is active or if '-drive ready' or '-seek complete' are inactive, the controller will not execute any command. Any code not defined in the following figure causes an Aborted Command error. Interrupt request 14 is reset when any command is written. The following are acceptable commands to the command register.

Command	Bits 7	6	5	4	3	2	1	0
Restore	0	0	0	1	R3	R2	R1	R0
Seek	0	1	1	1	R3	R2	R1	R0
Read Sector	0	0	1	0	0	0	L	T
Write Sector	0	0	1	1	0	0	L	T
Format Track	0	1	0	1	0	0	0	0
Read Verify	0	1	0	0	0	0	0	T
Diagnose	1	0	0	1	0	0	0	0
Set Parameters	1	0	0	1	0	0	0	1

Valid Command-Register Commands

Note: Stepping rate values and bit definitions for L and T are shown in the following figures.

The following figure shows the stepping rate as defined by R3 through R0.

R3	R2	R1	R0	Stepping Rate
0	0	0	0	35 us
0	0	0	1	0.5 ms
0	0	1	0	1.0 ms
0	0	1	1	1.5 ms
0	1	0	0	2.0 ms
0	1	0	1	2.5 ms
0	1	1	0	3.0 ms
0	1	1	1	3.5 ms
1	0	0	0	4.0 ms
1	0	0	1	4.5 ms
1	0	1	0	5.0 ms
1	0	1	1	5.5 ms
1	1	0	0	6.0 ms
1	1	0	1	6.5 ms
1	1	1	0	7.0 ms
1	1	1	1	7.5 ms

Stepping Rate

Note: After a Diagnose or Reset Command, the stepping rate is set to 7.5 milliseconds.

The following figure shows the bit definitions for bits L and T.

Bit	Definition	0	1
L	Data Mode	Data Only	Data Plus 4 Byte ECC
T	Retry Mode	Retries Enabled	Retries Disabled

L and T Bit Definitions

Note: When retries are disabled, ECC and ID field retries are limited to less than two complete revolutions.

Following are descriptions of the valid command-register commands.

Restore: The controller issues step pulses to the drive until the Track 000 indicator from the drive is active. If Track 000 is not active within 1023 steps the error bit in the status register is set and a Track 000 error is posted in the error register. The implied seek step rate can be set up using the stepping rate figure on the

previous page. The restore step rate is established by the seek complete signal from the drive (each step pulse is issued after seek complete is asserted by the drive from the previous step).

Seek: The Seek command moves the R/W heads to the cylinder specified in the task files. The adapter supports overlapped seeking on two drives or setup of the buffered seek stepping rate for the implied seek during a Read/Write command. An interrupt is generated at the completion of the command.

Read Sector: A number of sectors (1–256) may be read from the fixed disk with or without the ECC field appended in the Programmed I/O (PIO) mode. If the heads are not over the target track, the controller issues step pulses to the drive and checks for the proper ID field before reading any data. The stepping rate used during the implied seek is the value specified during the previous Seek or Restore command. Data errors, up to 5 bits in length, are automatically corrected on Read Short commands. If an uncorrectable error occurs, the data transfer still takes place; however, a multi-sector read ends after the system reads the sector in error. Interrupts occur as each sector is ready to be read by the system. No interrupt is generated at the end of the command, after the last sector is read by the system.

Write Sector: A number of sectors (1–256) may be written to the fixed disk with or without the ECC field appended in the PIO mode. The Write Sector command also supports implied seeks. Interrupts for the Write command occur before each sector is transferred to the buffer (except the first) and at the end of the command. The first sector may be written to the buffer immediately after the command has been sent, and '-data request' is active.

Format Track: The track specified by the task file is formatted with ID and data fields according to the interleave table transferred to the buffer. The interleave table is composed of two bytes per sector as follows: 00, Physical Sector 1, 00, Physical Sector 2, ... 00, Physical Sector 17. The table for 2-to-1 interleave is: 00, 01, 00, 0A, 00, 02, 00, 0B, 00, 03, 00, 0C, 00, 04, 00, 0D, 00, 05, 00, 0E, 00, 06, 00, 0F, 00, 07, 00, 10, 00, 08, 00, 11, 00, 09. The data transfer must be 512 bytes even though the table may be only 34 bytes. The sector count register must be loaded with the number of sectors per track before each Format

Track command. An interrupt is generated at the completion of the command; the Format Track command supports no error reporting. A bad block may be specified by replacing a 00 table entry with an 80.

When switching between drives, a restore command must be executed prior to attempting a format.

Preform the following when formatting a drive with more than 8 read/write heads:

1. Restore

2. Format all cylinders, heads 0 - 7 only

3. Restore

4. Format all cylinders, heads 8 and above.

Read Verify: This command is similar to to a Read command except that no data is sent to the host. This allows the system to verify the integrity of the fixed disk drive. A single interrupt is generated upon completion of the command or in the event of an error.

Diagnose: This command causes the adapter to execute its self-test code and return the results to the error register. An interrupt is generated at the completion of this command.

Set Parameters: This command sets up the drive parameters (maximum number of heads and sectors per track). The drive/head register specifies the drive affected. The sector count and drive/head registers must be set up before this command is issued. The adapter uses the values specified for track and cylinder crossing during multi-sector operations. An interrupt is generated at the completion of this command. This command must be issued before any multi-sector operations are attempted. The adapter supports two fixed disk drives with different characteristics, as defined by this command.

Miscellaneous Information

The following is miscellaneous information about the fixed disk drive function.

- The adapter performs normal read/write operations on a data field only after a successful match of that sector's ID with the targeted ID.
- ID fields are checked for errors when read from the disk.
- The adapter supports only ECC on data fields and only CRC on ID fields. The CRC polynomial is X16 + X12 + X5 + 1; the ECC polynomial is X32 + X28 + X26 + X19 + X17 + X10 + X6 + X2 + 1. All shift registers are preset to hex F before calculating the checksums, which begin with the respective address marks.

Diskette Function

The 5-1/4 inch diskette drive function is an integral part of the Fixed Disk and Diskette Drive Adapter. One or two diskette drives are attached to the adapter through an internal, daisy-chained, flat cable. The attachment will support 160K.-, 320K.-, and 1.2M.-byte diskette drives.

The address assignments for diskette functions are shown in the following figure.

I/O Address			
Primary	Secondary	Read	Write
3F2	372	-	Digital Output Register
3F4	374	Main Status Register	Main Status Register
3F5	375	Diskette Data Register	Diskette Data Register
3F6	376	-	Fixed Disk Register
3F7	377	Digital Input Register	Diskette Control Register

Diskette Function

The adapter is designed for a double-density, MFM-coded, diskette drive and uses write precompensation with an analog circuit for clock and data recovery. The diskette-drive parameters are programmable, and the diskette drive's write-protect feature is supported. The adapter is buffered on the I/O bus and uses the system board's direct memory access (DMA) for record data transfers. An interrupt level also is used to indicate when an operation is complete and that a status condition requires microprocessor attention.

Digital Output Register (hex 3F2)

The digital output register (DOR) is an output-only register used to control drive motors, drive selection, and feature enable. The bit definitions follow:

Bit 7 Reserved

Bit 6 Reserved

Bit 5 Drive B Motor Enable

Bit 4 Drive A Motor Enable

Bit 3 Enable Diskette Interrupts and DMA

Bit 2 Diskette Function Reset

Bit 1 Set to a logical 0

Bit 0 Drive Select—A 0 on this bit indicates that drive A is selected.

Note: A channel reset clears all bits.

Digital Input Register (hex 3F7)

The digital input register is an 8-bit, read-only register used for diagnostic purposes. The following are bit definitions for this register:

Bit 7	Diskette Change
Bit 6	Write Gate
Bit 5	Head Select 3/Reduced Write Current
Bit 4	Head Select 2
Bit 3	Head Select 1
Bit 2	Head Select 0
Bit 1	Drive Select 1
Bit 0	Drive Select 0

Note: Bits 0 through 6 apply to the currently selected fixed disk drive. These bits are valid for 50 microseconds after a write to the Drive Head Register.

Data Rates

The diskette function will support three data rates: 250,000, 300,000 and 500,000 bits per second.

Diskette Controller

The diskette controller has two registers to which the system unit's microprocessor has access: a status register and a data register. The status register may only be read and is used to facilitate the transfer of data between the processor and diskette controller. The 8-bit status register has the status information about the diskette and may be accessed at any time. The 8-bit data register (hex 3F5), which actually consists of several registers in a stack with only one register presented to the data bus at a time, stores data, commands, and parameters, and provides diskette-drive status information. Data bytes are read from or written to the data register in order to program or obtain results after a particular command.

The bits in the status register (hex 34F) are defined as follows:

Bit 7 Request for Master (RQM)— The data register is ready to send or receive data to or from the processor.

Bit 6 Data Input/Output (DIO)—The direction of data transfer between the diskette controller and the processor. If this bit is a 1, transfer is from the diskette controller's data register to the processor; if it is a 0, the opposite is true.

Bit 5 Non-DMA Mode (NDM)—The diskette controller is in the non-DMA mode.

Bit 4 Diskette Controller Busy (CB)— A Read or Write command is being executed.

Bit 3 Reserved

Bit 2 Reserved

Bit 1 Diskette Drive B Busy (DBB)— Diskette drive B is in the seek mode.

Bit 0 Diskette Drive A Busy (DAB)— Diskette drive A is in the seek mode.

Diskette Control Register (hex 3F7)

This register is assigned two addresses, hex 3F7 (primary) and hex 377 (secondary). This is a four bit write only register. The bits are defined as follows:

Bits 7 - 2 Reserved

Bits 2 - 0 Diskette Data Rate— These bits select the diskette data rate as shown in the following figure:

Bit 0	Bit 1	Diskette Data Rate
0	0	500,000 bps
0	1	300,000 bps
1	0	250.000 bps
1	1	Unused

Diskette Data Rate

Fixed Disk Register (hex 3F6)

This register is assigned two addresses, 3F6 (primary) and 376 (secondary). This is a four bit write only register. The bits are defined as follows:

Bits 7 - 4 Reserved

Bit 3 A logical 0 enables reduced write current. A logical 1 enables head select 3.

Bit 2 A logical 1 enables reset fixed disk function.

Bit 1 A logical 0 enables fixed disk interrupts.

Bit 0 Reserved

Note: Bit 3 defines the function of the fixed disk control interface connector (pin 2).

Diskette Controller Commands

The diskette controller can perform 16 different commands. Each command is initiated by a multibyte transfer from the processor, and the result after execution of the command may also be a multibyte transfer back to the processor. Because of this multibyte interchange of information between the diskette controller and the processor, each command can be considered to consist of three phases:

Command Phase: The processor issues a sequence of Write commands to the diskette controller that direct the controller to perform a specific operation.

Execution Phase: The diskette controller performs the specified operation.

Result Phase: After completion of the operation, status and other housekeeping information is made available to the processor through a sequence of Read commands to the processor.

The following is a list of commands that may be issued to the diskette controller:

- Read Data
- Read Deleted Data
- Write Data
- Write Deleted Data
- Read a Track
- Read ID
- Format a Track
- Scan Equal
- Scan Low or Equal
- Scan High or Equal

- Recalibrate
- Sense Interrupt Status
- Specify
- Sense Drive Status
- Seek
- Invalid.

Symbol Descriptions

The following are descriptions of the symbols used in the "Command Definitions" later in this section.

A0 Address Line 0—A logical 0 selects the main status register, and a 1 selects the data register.

C Cylinder Number—Contains the current or selected cylinder number in binary notation.

D Data—Contains the data pattern to be written to a sector.

D7-D0 Data Bus—An 8-bit data bus in which D7 is the most-significant bit and D0 is the least- significant.

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 VCO 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 256 milliseconds in 2- millisecond increments for the 1.2M-byte drive and 4 to 512 milliseconds in 4 millisecond increments for the 320K-byte drive).

HUT Head Unload Time—The head unload time after a read or write operation (0 to 240 milliseconds in 16-millisecond increments for the 1.2M-byte drive and 0 to 480 milliseconds in 32- millisecond increments for the 320K-byte drive.

MF FM or MFM Mode—A 0 selects FM mode and a 1 selects MFM (MFM is selected only if it is implemented.)

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—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.

R/W Read/Write—This stands for either a 'read' or 'write' signal.

SC Sector—The number of sectors per cylinder.

SK Skip—This stands for skip deleted-data address mark.

SRT This 4 bit byte indicates the stepping rate for the diskette drive as follows:

1.2M-Byte Diskette Drive

1111 1 millisecond
1110 2 milliseconds
1101 3 milliseconds

320K-Byte Diskette Drive

1111 2 milliseconds
1110 4 milliseconds
1101 6 milliseconds

ST 0—ST 3 Status 0–Status 3—One of the four registers that stores 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 processor during a scan operation. If STP is 2, then alternate sections are read and compared.

US0–US1 Unit Select—The selected driver number encoded the same as bits 0 and 1 of the digital output register (DOR).

Controller Commands

The following are commands that may be issued to the controller.

> **Note:** An X is used to indicate a don't-care condition. Commands not shown in binary format are shown as bytes.

Read Data

Command Phase: The following bytes are issued by the processor in the command phase:

D7	D6	D5	D4	D3	D2	D1	D0
MT	MF	SK	0	0	1	1	0
X	X	X	X	X	HD	US1	US0
			C				
			H				
			R				
			N				
			EOT				
			GPL				
			DTL				

Result Phase: The following bytes are issued by the controller in the result phase:

ST0
ST1
ST2
C
H
R
N

Read Deleted Data

Command Phase: The following bytes are issued by the processor in the command phase:

D7	D6	D5	D4	D3	D2	D1	D0
MT	MF	SK	0	1	1	0	0
X	X	X	X	X	HD	US1	US0
			C				
			H				
			R				
			N				
			EOT				
			GPL				
			DTL				

Result Phase: The following bytes are issued by the controller in the result phase:

ST0
ST1
C
H
R
N

Write Data

Command Phase: The following bytes are issued by the processor in the command phase:

D7	D6	D5	D4	D3	D2	D1	D0
MT	MF	0	0	0	1	0	1
X	X	X	X	X	HD	US1	US0
				C			
				H			
				R			
				N			
				EOT			
				GPL			
				DTL			

Result Phase: The following bytes are issued by the controller in the result phase:

ST0
ST1
ST2
C
H
R
N

Write Deleted Data

Command Phase: The following bytes are issued by the processor in the command phase:

D7	D6	D5	D4	D3	D2	D1	D0
MT	MF	0	0	1	0	0	1
X	X	X	X	X	HD	US1	US0
				C			
				H			
				R			
				N			
				EOT			
				GPL			
				DTL			

Result Phase: The following bytes are issued by the controller in the result phase:

ST0
ST1
ST2
C
H
R
N

Read a Track

Command Phase: The following bytes are issued by the processor in the command phase:

D7	D6	D5	D4	D3	D2	D1	D0
0	MF	SK	0	0	0	1	0
X	X	X	X	X	HD	US1	US0
			C				
			H				
			R				
			N				
			EOT				
			GPL				
			DTL				

Result Phase: The following bytes are issued by the controller in the result phase:

ST0
ST1
ST2
C
H
R
N

Read ID

Command Phase: The following bytes are issued by the processor in the command phase:

D7	D6	D5	D4	D3	D2	D1	D0
0	MF	0	0	1	0	1	0
X	X	X	X	X	HD	US1	US0

Result Phase: The following bytes are issued by the processor in the command phase:

ST0
ST1
ST2
C
H
R
N

Format a Track

Command Phase: The following bytes are issued by the processor in the command phase:

D7	D6	D5	D4	D3	D2	D1	D0
0	MF	0	0	1	1	0	0
X	X	X	X	X	HD	US1	US0
			N				
			SC				
			GPL				
			D				

Result Phase: The following bytes are issued by the controller in the result phase:

ST0
ST1
ST2
C
H
R
N

Scan Equal

Command Phase: The following bytes are issued by the processor in the command phase:

D7	D6	D5	D4	D3	D2	D1	D0
MT	MF	SK	1	0	0	0	1
X	X	X	X	X	HD	US1	US0

C
H
R
N
EOT
GPL
STP

Result Phase: The following bytes are issued by the controller in the result phase:

ST0
ST1
ST2
C
H
R
N

Scan Low or Equal

Command Phase: The following bytes are issued by the processor in the command phase:

D7	D6	D5	D4	D3	D2	D1	D0
MT	MF	SK	1	1	0	0	1
X	X	X	X	X	HD	US1	US0
			C				
			H				
			R				
			N				
			EOT				
			GPL				
			STP				

Result Phase: The following bytes are issued by the controller in the result phase:

ST0
ST1
ST2
C
H
R
N

Scan High or Equal

Command Phase: The following bytes are issued by the processor in the command phase:

D7	D6	D5	D4	D3	D2	D1	D0
MT	MF	SK	1	1	1	0	1
X	X	X	X	X	HD	US1	US0
				C			
				H			
				R			
				N			
				EOT			
				GPL			
				STP			

Result Phase: The following bytes are issued by the controller in the result phase:

ST0
ST1
ST2
C
H
R
N

Recalibrate

Command Phase: The following bytes are issued by the processor in the command phase:

D7	D6	D5	D4	D3	D2	D1	D0
0	0	0	0	0	1	1	1
X	X	X	X	X	0	US1	US0

Result Phase: This command has no result phase.

Sense Interrupt Status

Command Phase: The following bytes are issued by the processor in the command phase:

D7	D6	D5	D4	D3	D2	D1	D0
0	0	0	0	1	0	0	0

Result Phase: The following bytes are issued by the controller in the result phase:

STO
PCN

Specify

Command Phase: The following bytes are issued by the processor in the command phase:

D7	D6	D5	D4	D3	D2	D1	D0
0	0	0	0	0	0	1	1
(	SRT	)		(	HUT	)	
(	HLT	)		(	ND	)	

Result Phase: This command has no result phase.

Sense Driver Status

Command Phase: The following bytes are issued by the processor in the command phase:

D7	D6	D5	D4	D3	D2	D1	D0
0	0	0	0	0	0	1	0
X	X	X	X	X	HD	US1	US0

Result Phase: The following bytes are issued by the controller in the result phase:

ST3

Seek

Command Phase: The following bytes are issued by the processor in the command phase:

D7	D6	D5	D4	D3	D2	D1	D0
0	0	0	0	1	1	1	1
X	X	X	X	X	HD	US1	US0
			NCN				

Result Phase: This command has no result phase.

Invalid

Command Phase: The following bytes are issued by the processor in the command phase:

D7	D6	D5	D4	D3	D2	D1	D0
		Invalid Codes					
X	X	X	X	X	HD	US1	US0

Result Phase: The following byte is issued by the controller in the result phase:

ST0

Command Status Registers

The following is information about the command status registers ST0 through ST3.

Command Status Register 0 (ST0)

The following are bit definitions for command status register 0.

Bit 7–Bit 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** Abnormal termination because, during the execution of a command, the 'ready' signal from the diskette drive changed state.

Bit 5 Seek End (SE)—Set to 1 when the controller completes the Seek command.

Bit 4 Equipment Check (EC)—Set if a 'fault' signal is received from the diskette drive, or if the 'track–0' signal fails to occur after 77 step pulses (Recalibrate Command).

Bit 3 Not Ready (NR)—This flag is set when the diskette drive is in the not-ready state and a Read or Write command is issued. It is also set if a Read or Write command is issued to side 1 of a single-sided diskette drive.

Bit 2 Head Address (HD)—Indicates the state of the head at interrupt.

Bit 1–Bit 0 Unit select 0 and 1 (US 0 and 1)—Indicate a drive's unit number at interrupt. The following figure shows the binary values to select each drive:

Bit 1	Bit 0	Drive Selected
0	0	A
0	1	B
1	0	Unused
1	1	Unused

Unit Selection

Command Status Register 1 (ST1)

The following are bit definitions for command status register 1.

Bit 7 End of Cylinder (EC)—Set when the controller tries to gain access to a sector beyond the final sector of a cylinder.

Bit 6 Not Used—Always 0.

Bit 5 Data Error (DE)—Set when the controller detects a CRC error in either the ID field or the data field.

Bit 4 Overrun (OR)—Set if the controller is not serviced by the main system within a certain time limit during data transfers.

Bit 3 Not Used—This bit is always set to 0.

Bit 2 No Data (ND)—Set if the controller cannot find the sector specified in the ID register during the execution of a Read Data, Write Deleted Data, or Scan Command. This flag is also set 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 Cylinder command.

Bit 1 Not Writable (NW)—Set if the controller detects a 'write-protect' signal from the diskette drive during execution of a Write Data, Write Deleted Data, or Format Cylinder command.

Bit 0 Missing Address Mark (MA)—Set if the controller cannot detect the ID address mark. At the same time, the MD of status register 2 is set.

Command Status Register 2 (ST2)

Bit 7 Not Used—Always 0.

Bit 6 Control Mark (CM)—This flag is set if the controller encounters a sector that has a deleted data-address mark during execution of a Read Data or Scan command.

Bit 5 Data Error in Data Field (DD)—Set if the controller detects an error in the data.

Bit 4 Wrong Cylinder (WC)—This flag is related to ND (no data) and when the contents of C on the medium are different from that stored in the ID register, this flag is set.

Bit 3 Scan Equal Hit (SH)—Set if the contiguous sector data equals the processor data during the execution of a Scan command.

Bit 2 Scan Not Satisfied (SN)—Set if the controller cannot find a sector on the cylinder that meets the condition during a Scan command.

Bit 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 FF, this flag is set.

Bit 0 Missing Address Mark in Data Field (MD)— Set if the controller cannot find a data address mark or a deleted data address mark when data is read from the medium.

Command Status Register 3 (ST3)

The following are bit definitions for command status register 3.

Bit 7 Fault (FT)—Status of the 'fault' signal from the diskette drive.

Bit 6 Write Protect (WP)—Status of the 'write-protect' signal from the diskette drive.

Bit 5 Ready (RY)—Status of the 'ready' signal from the diskette drive.

Bit 4 Track 0 (T0)—Status of the 'track 0' signal from the diskette drive.

Bit 3 Two Side (TS)—Status of the 'two side' signal from the diskette drive.

Bit 2 Head Address (HD)—Status of the 'side-select' signal from the diskette drive.

Bit 1 Unit Select 1 (US 1)—Status of the 'unit-select-1' signal from the diskette drive.

Bit 0 Unit Select 0 (US 0)—Status of the 'unit select 0' signal from the diskette drive.

Interfaces

The system interface is through the I/O channel. The address, DMA, and interrupt assignments are shown in the following figures.

I/O Address			
Primary	Secondary	Read	Write
3F2	372		Digital Output Register
3F4	374	Main Status Register	Main Status Register
3F5	375	Diskette Data Register	Diskette Data Register
3F6	376		Fixed Disk Register
3F7	377	Digital Input Register	Diskette Control Register

Diskette Function

Note: DMA request is level 2 and interrupt request is level 6.

I/O Address			
Primary	Secondary	Read	Write
1F0	170	Data Register	Data Register
1F1	171	Error Register	Write Precomp
1F2	172	Sector Count	Sector Count
1F3	173	Sector Number	Sector Number
1F4	174	Cylinder Low	Cylinder Low
1F5	175	Cylinder High	Cylinder High
1F6	176	Drive/Head Register	Drive/Head Register
1F7	177	Status Register	Command Register

Fixed Disk Function

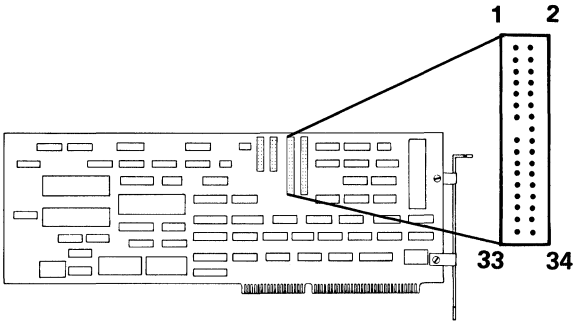
Note: Interrupt request is level 14.

The following operations are supported by this adapter:

- 16 bit programmed I/O (PIO), data transfers to the fixed disk. All other transfers are 8 bits wide.
- The I/O addresses, recognized by the adapter for either the fixed disk or the diskette function, are independently selected by jumpers.

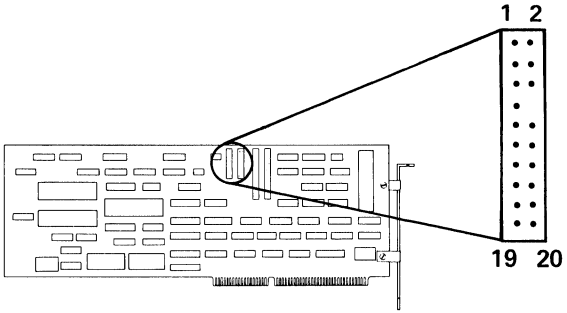
Interface Lines

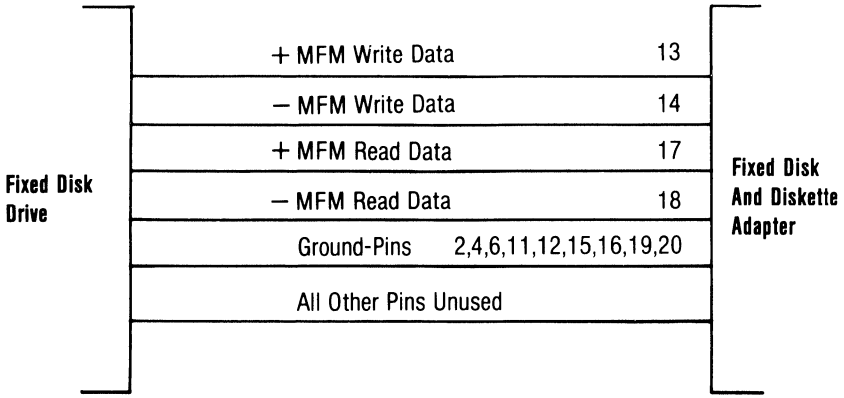
The interface to the fixed disk drive consists of the Control cable and the Data cable. The following figures show signals and pin assignments for these cables.



Fixed Disk Drive			Fixed Disk And Diskette Adapter
	Ground – Odd Numbers	1-33	
←	– Reduced Write Current/– Head Select 3	2	
←	– Head Select 2	4	
←	– Write Gate	6	
→	– Seek Complete	8	
→	– Track 000	10	
→	– Write Fault	12	
←	– Head Select 0	14	
	Reserved	16	
←	– Head Select 1	18	
→	– Index	20	
→	– Ready	22	
←	– Step	24	
←	– Drive Select 1	26	
←	– Drive Select 2	28	
	Reserved	30	
	Reserved	32	
←	– Direction In	34	

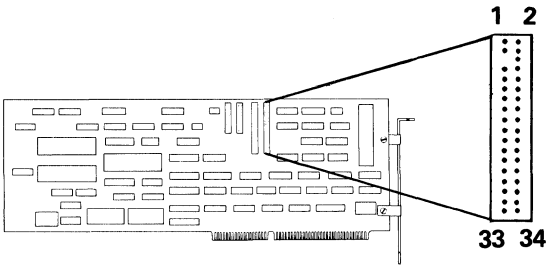
Note: Connection is through a 2-by-17 Berg connector. Pin 15 is reserved to polarize the connector.





Note: Connection is through a 2-by-10 Berg connector. Pin 8 is reserved to polarize the connector.

The interface to the diskette drives is a single cable that carries both data and control signals. The signals and pin assignments are as follows.



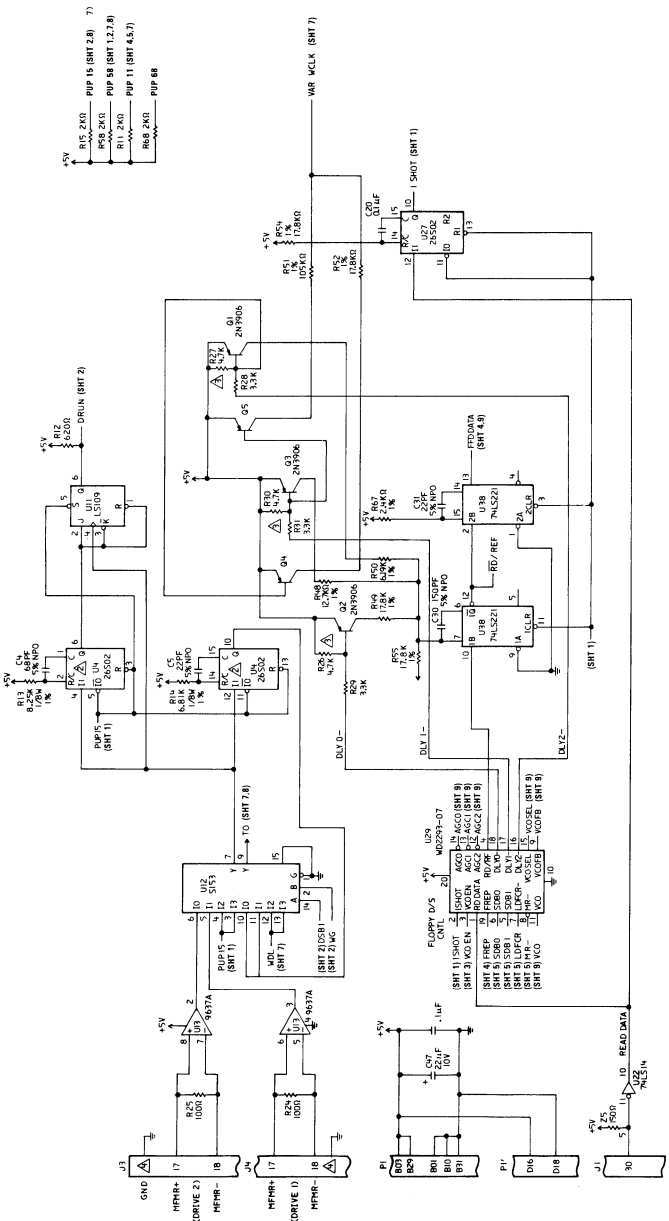
	Ground – Odd Numbers	1-33
←	Reduced Write	2
	Reserved	4
←	Drive Select 3	6
→	Index	8
←	Drive Select 0	10
←	Drive Select 1	12
←	Drive Select 2	14
←	Motor On	16
←	Direction Select	18
←	Step	20
←	Write Data	22
←	Write Gate	24
→	Track 00	26
→	Write Protect	28
→	Read Data	30
←	Side 1 Select	32
→	Diskette Change	34

Diskette Drive

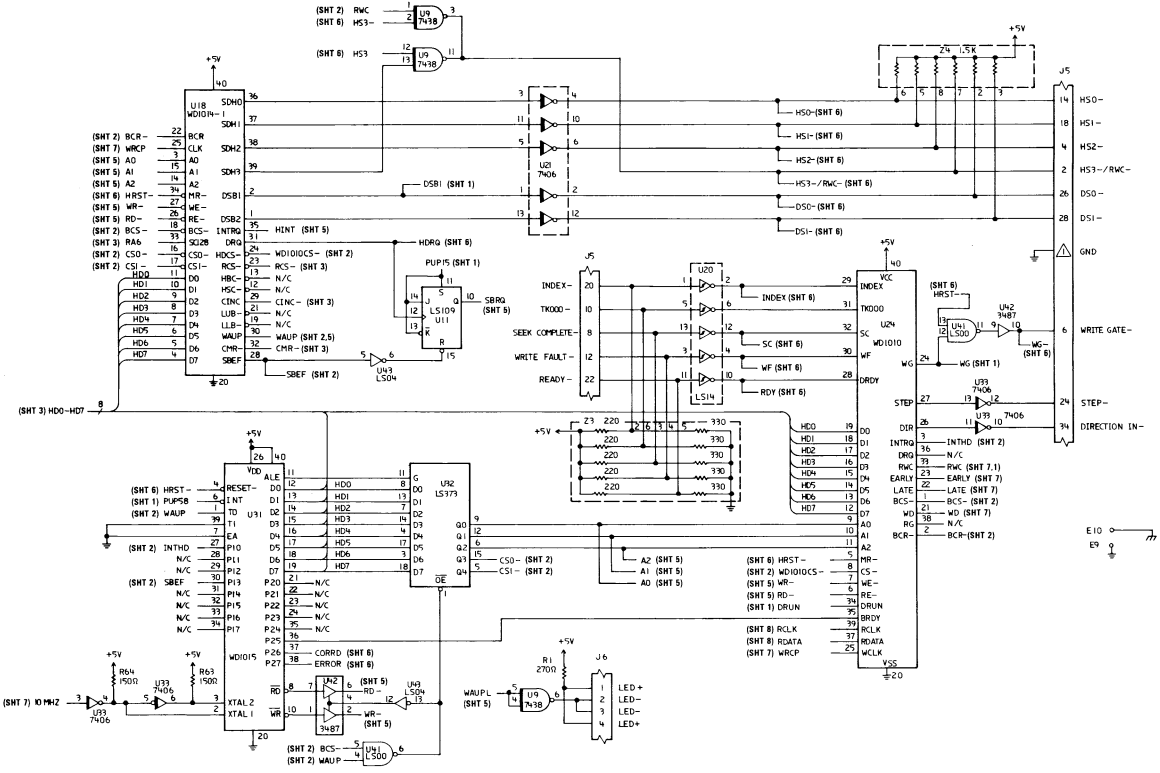
Fixed Disk And Diskette Adapter

Note: Connection is through a 2-by-17 Berg connector. Pin 5 is reserved to polarize the connector.

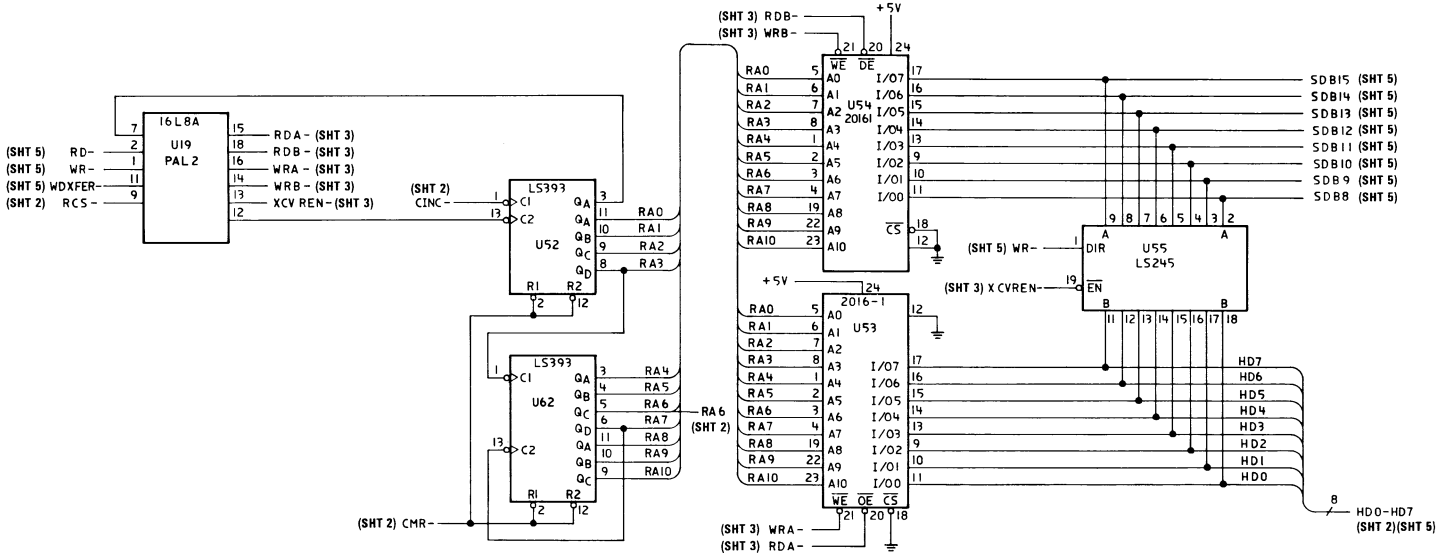
Logic Diagrams



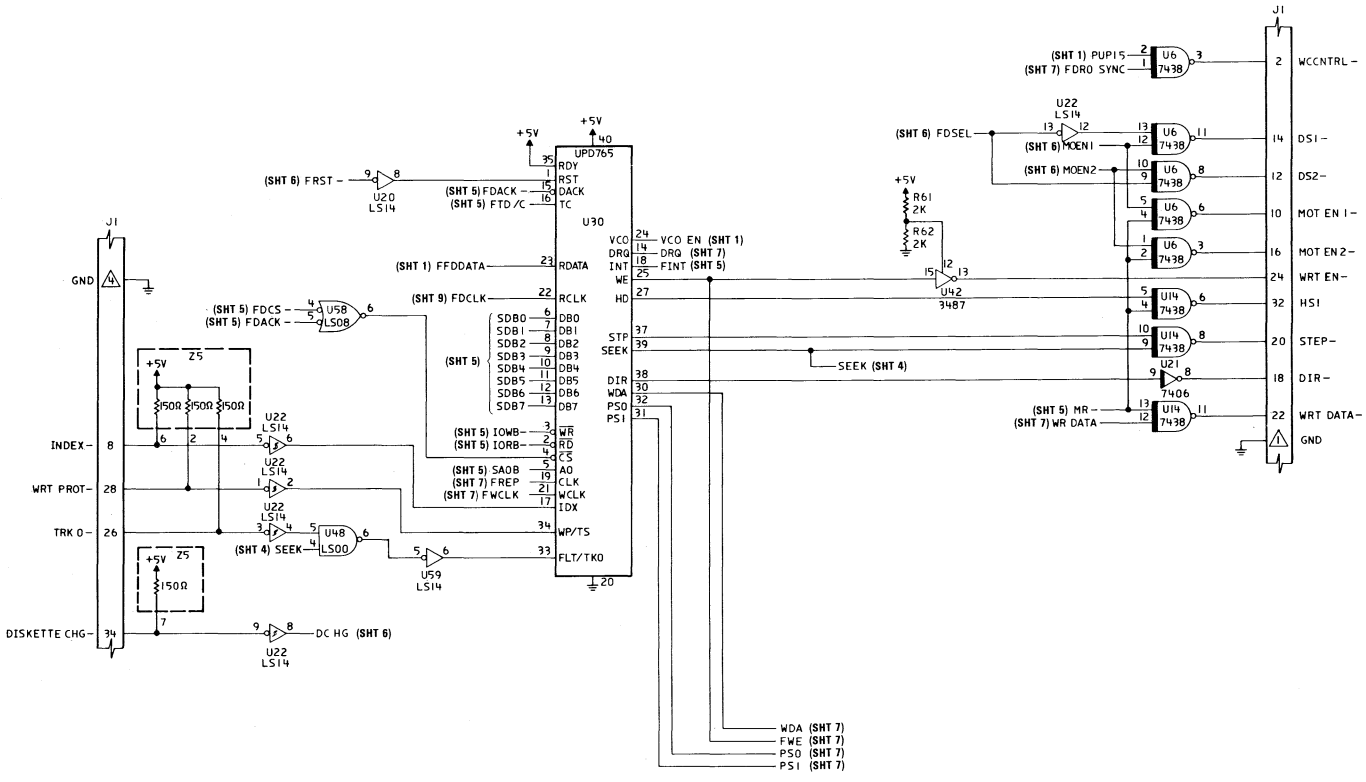
Fixed Disk and Diskette Drive Adapter (Sheet 1 of 9)



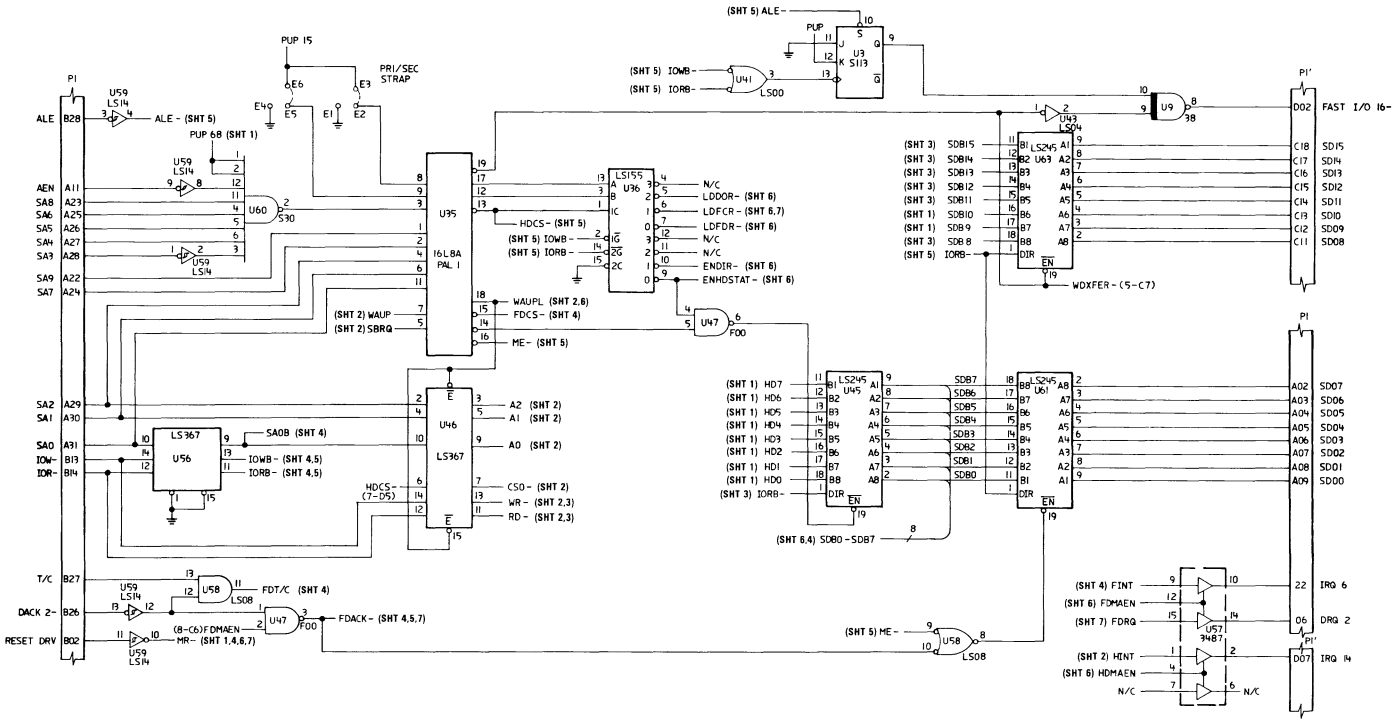
Fixed Disk and Diskette Drive Adapter (Sheet 2 of 9)



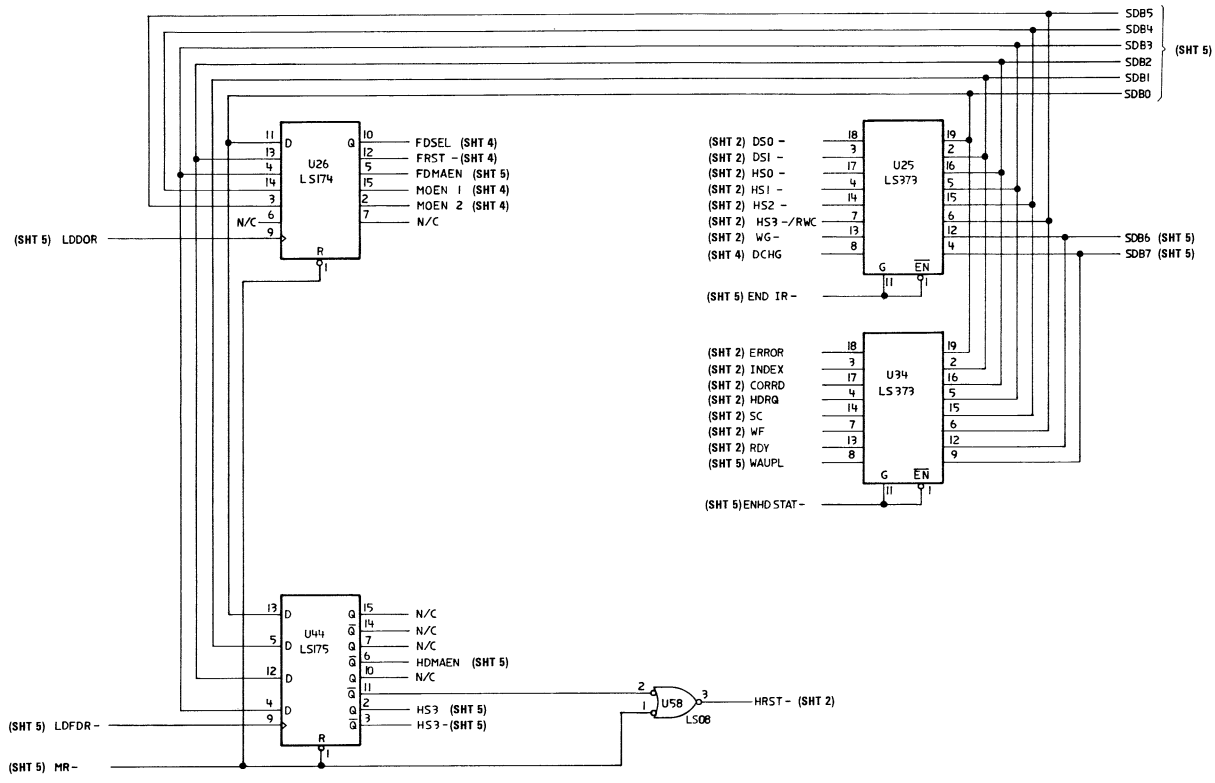
Fixed Disk and Diskette Drive Adapter (Sheet 3 of 9)



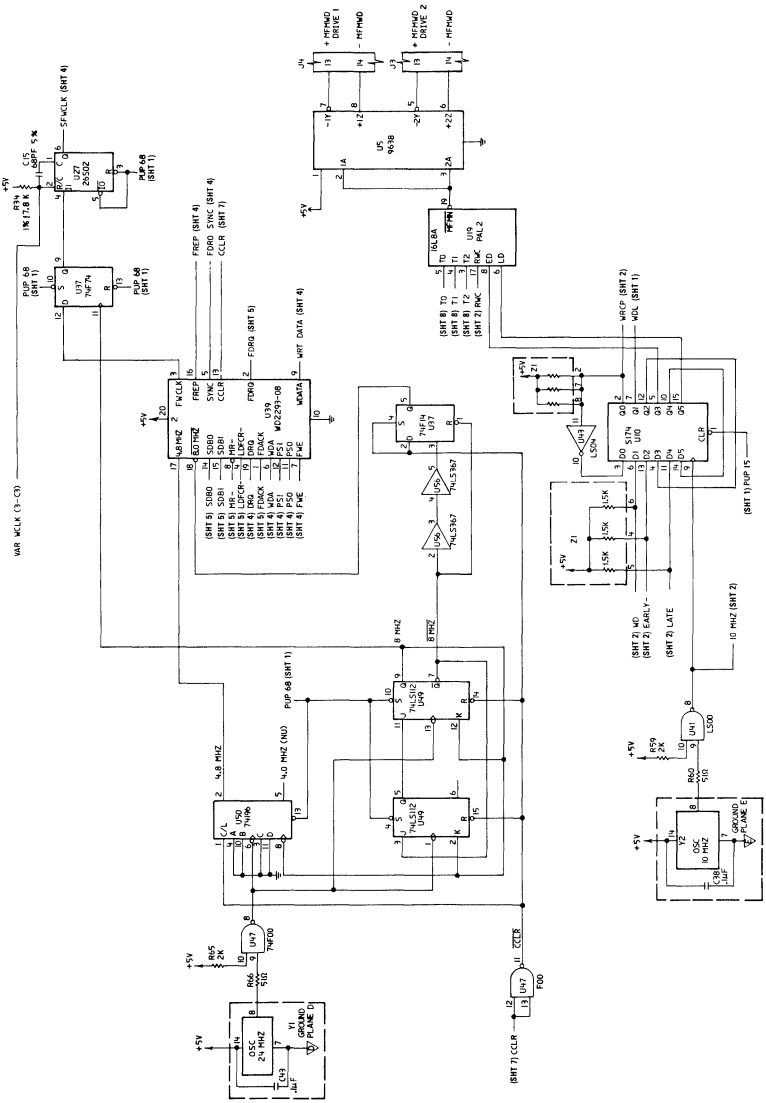
Fixed Disk and Diskette Drive Adapter (Sheet 4 of 9)



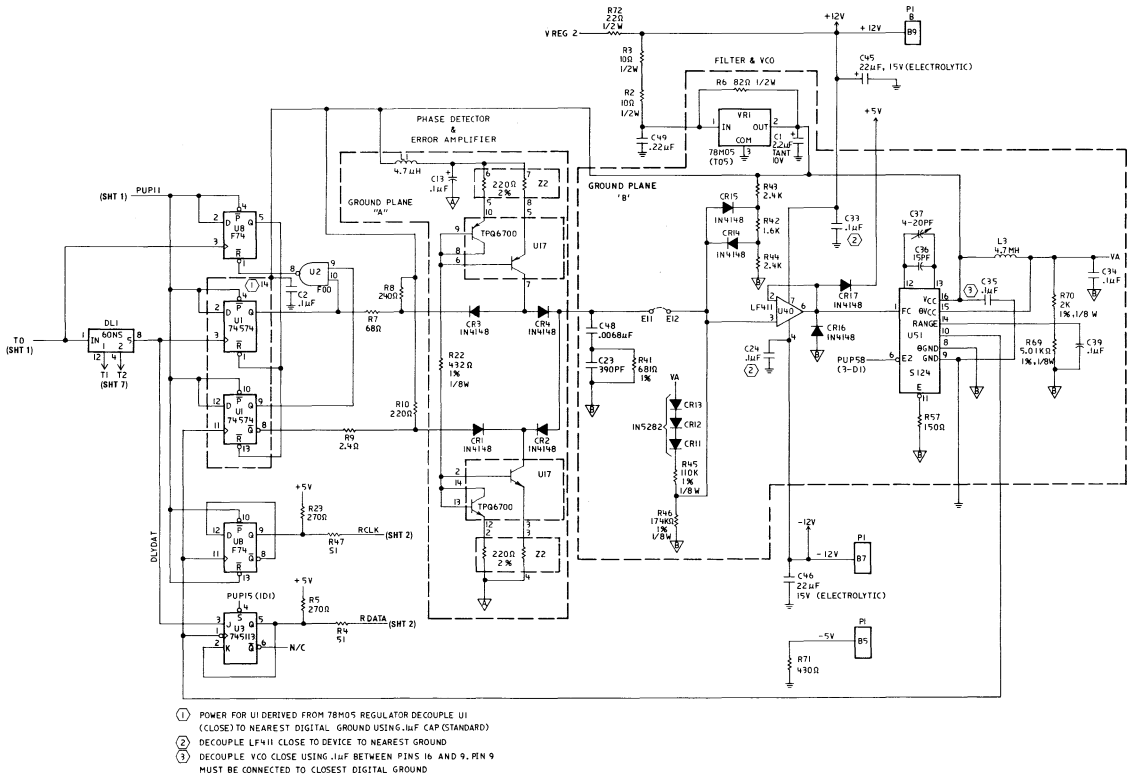
Fixed Disk and Diskette Drive Adapter (Sheet 5 of 9)



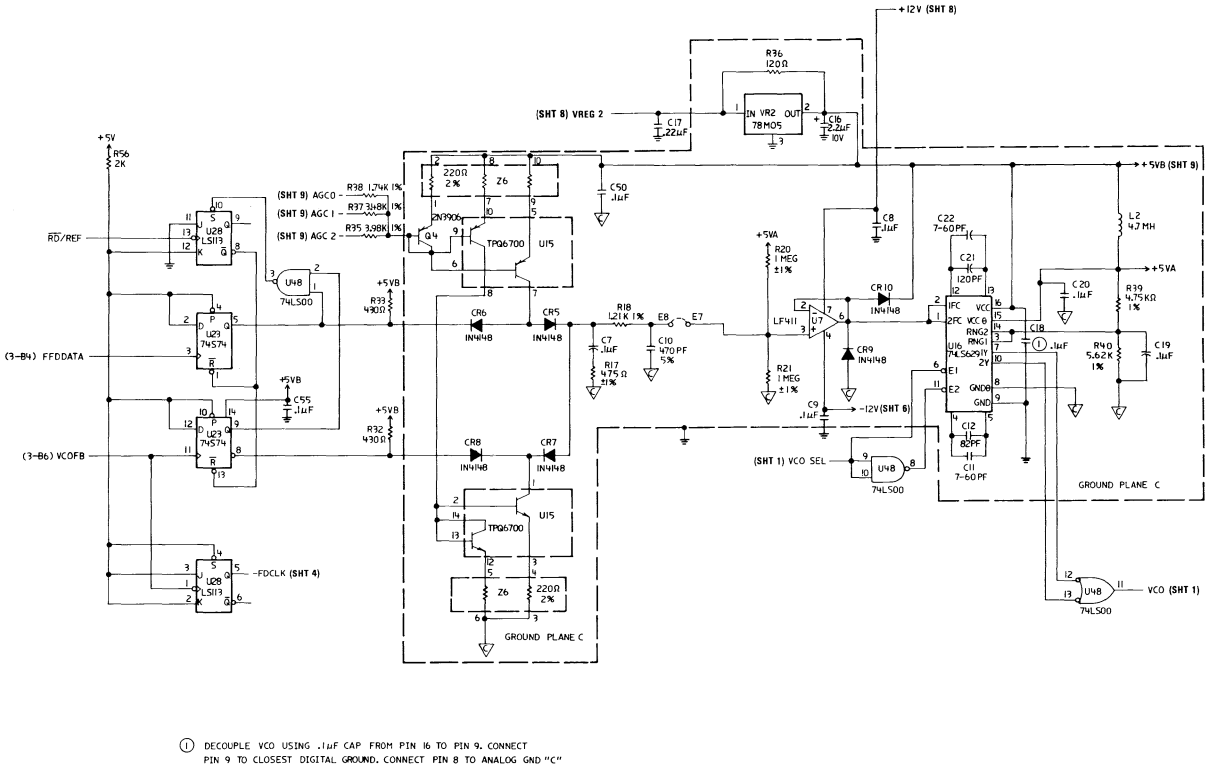
Fixed Disk and Diskette Drive Adapter (Sheet 6 of 9)



Fixed Disk and Diskette Drive Adapter (Sheet 7 of 9)



Fixed Disk and Diskette Drive Adapter (Sheet 8 of 9)



① DECOUPLE VCO USING .1μF CAP FROM PIN 16 TO PIN 9. CONNECT PIN 9 TO CLOSEST DIGITAL GROUND. CONNECT PIN 8 TO ANALOG GND "C"

Fixed Disk and Diskette Drive Adapter (Sheet 9 of 9)

Notes:

Personal Computer
Hardware Reference
Library

IBM Personal Computer 20MB Fixed Disk Drive Adapter

6139790

March 17, 1986

Notes:

Contents

Notes:

Description

The 20MB Fixed Disk Drive Adapter attaches to one or two fixed disk drive units through an internal, daisy-chained, flat cable (data/control cable).

The adapter is buffered on the I/O bus and uses the system board's direct memory access (DMA) for fixed-disk-drive data transfers. When the adapter is enabled, an interrupt request occurs on the IRQ-5 line to the 8259A Interrupt Controller. The 8259A then causes an interrupt hex 0D.

The Fixed Disk Drive Adapter provides automatic 11-bit burst error detection and correction in the form of 32-bit error checking and correction (ECC).

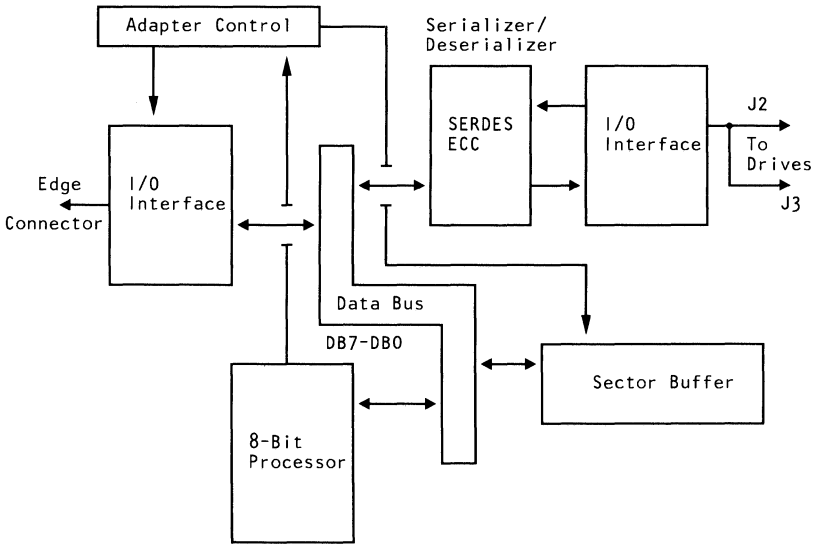
The device level control for the Fixed Disk Adapter is contained on a ROM module on the adapter. A listing of this device level control can be found in "BIOS Listing" of this section.

Warning: The last cylinder on the fixed disk drive is reserved for diagnostic use. The diagnostic write test will destroy any data on this cylinder.

Fixed Disk Controller

The disk controller has three registers that may be accessed by the system unit's microprocessor: a status register, a data register, and a read-option-jumpers register. The 8-bit status register contains the status information of the disk controller, and can be accessed at any time. This register is read-only and is used to help the transfer of data between the system unit's microprocessor and the disk controller. The 8-bit data register (actually 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 status 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 controller-select pulse is generated by writing to port address hex 322.

The following is a block diagram of the IBM 20MB Fixed Disk Drive Adapter.



Programming Considerations

Types of Drives

The fixed disk drive adapter will accommodate any two of four different types of drives. The figure below shows the configuration of the different type drives.

Type	Cylinders	Heads	Start of Write Pre-Comp	Landing Zone
1	306	4	0	306
2	615	4	300	615
13	306	8	128	336
16	612	4	0	663

Fixed Disk Types

The figure below shows the switch settings for the above mentioned drive types. Switches 1 and 2 set the parameters of Drive 0, and switches 3 and 4 set Drive 1.

	Drive 0		Drive 1	
	Switch		Switch	
	1	2	3	4
Type 1	On	On	On	On
Type 2	Off	On	Off	On
Type 13	Off	Off	Off	Off
Type 16	On	Off	On	Off

Status Register

At the end of all commands from the system board, the disk controller sends a completion status byte to the system board. This byte informs the system unit's microprocessor if an error occurred during the execution of the command. The following shows the format of this byte.

Bit	7	6	5	4	3	2	1	0
	0	0	d	0	0	0	e	0

Bit 5 This bit shows the logical unit number of the drive.

Bit 1 When set, this bit shows an error has occurred during command execution.

Bits 7, 6, 4, 3, 2, 0 These bits are set to zero.

If the interrupts are enabled, the controller sends an interrupt when it is ready to transfer the status byte. Busy from the disk controller is unasserted when the byte is transferred to complete the command.

Sense Bytes

If the status register receives an error (bit 1 set), the disk controller requests four bytes of sense data. The format for the four bytes is as follows:

Bits	7	6	5	4	3	2	1	0
Byte 0	Address Valid	0	Error Type			Error Code		
Byte 1	0	0	d	Head Number				
Byte 2	Cylinder High			Sector Number				
Byte 3	Cylinder Low							

Remarks: d = drive

Disk Controller Error Tables

The following disk controller error tables list the error types and error codes found in byte 0.

The address-valid bit (bit 7) is only set when the previous command required a disk address. Bit 6 is set to 0 (spare).

	Error Type		Error Code				
Bits	5	4	3	2	1	0	Description
	0	0	0	0	0	0	The controller did not detect any error during the execution of the previous operation.
	0	0	0	0	0	1	The controller did not detect an index signal from the drive.
	0	0	0	0	1	0	The controller did not get a seek-complete signal from the drive after a seek operation (for all non-buffered step seeks).
	0	0	0	0	1	1	The controller detected a write fault from the drive during the last operation.
	0	0	0	1	0	0	After the controller selected the drive, the drive did not respond with a ready signal.
	0	0	0	1	0	1	Not Used.
	0	0	0	1	1	0	After stepping the maximum number of cylinders, the controller did not receive the track 00 signal from the drive.
	0	0	0	1	1	1	Not Used.
	0	0	1	0	0	0	The drive is still seeking. This status is reported by the test Drive Ready command for an overlap seek condition when the drive had not completed the seek. No time-out is measured by the controller for the seek to complete.

	Error Type		Error Code				
Bits	5	4	3	2	1	0	Description
	0	1	0	0	0	0	ID Read Error: The controller detected an ECC error in the target ID field on the disk.
	0	1	0	0	0	1	Data Error: The controller detected an uncorrectable ECC error in the target sector during a read operation.
	0	1	0	0	1	0	Address Mark: The controller did not detect the target address mark (AM) on the disk.
	0	1	0	0	1	1	Not Used.
	0	1	0	1	0	0	Sector Not Found: The controller found the correct cylinder and head, but not the target sector.
	0	1	0	1	0	1	Seek Error: The cylinder or head address (either or both) did not compare with the expected target address as a result of a seek.
	0	1	0	1	1	0	Not Used.
	0	1	0	1	1	1	Not Used.
	0	1	1	0	0	0	Correctable Data Error: The controller detected a correctable ECC error in the target field.
	0	1	1	0	0	1	Bad Track: The controller detected a bad track flag during the last operation. No retries are attempted on this error.

	Error Type		Error Code				
Bits	5	4	3	2	1	0	Description
	1	0	0	0	0	0	Invalid Command: The controller had received an invalid command from the system unit.
	1	0	0	0	0	1	Illegal Disk Address: The controller detected an address that is beyond the maximum range.

	Error Type		Error Code				
Bits	5	4	3	2	1	0	Description
	1	1	0	0	0	0	RAM Error: the controller detected a data error during the RAM sector-buffer diagnostic test.
	1	1	0	0	0	1	Program Memory Checksum Error: During this internal diagnostic test, the controller detected a program-memory checksum error.
	1	1	0	0	1	0	ECC Polynomial Error: During the controller's internal diagnostic tests, the hardware ECC generator failed its test.

Data Register

The system unit's microprocessor specifies the operation by sending the 6-byte device control block (DCB) to the controller. The figure below shows the format of the DCB, and defines the bytes that make up the DCB.

Bits	7	6	5	4	3	2	1	0
Byte 5	Control Field							
Byte 4	Interleave or Block Count							
Byte 3	Cylinder Low							
Byte 2	Cylinder High				Sector Number			
Byte 1	0	0	d	Head Number				
Byte 0	Command Class			Opcode				

Byte 5 Bits 7 through 0 contain the control field.

Byte 4 Bits 7 through 0 specify the interleave or block count.

Byte 3 Bits 7 through 0 are the eight least-significant bits of the cylinder number.

Byte 2 Bits 7 and 6 are the two most significant bits of the cylinder number. Bits 0 through 5 define the sector number.

Byte 1 Bit 5 identifies the drive number. Bits 4 through 0 contain the disk head number to be selected. Bits 6 and 7 are not used.

Byte 0 Bits 7, 6, and 5 identify the class of the command. Bits 4 through 0 contain the Opcode (see command byte on page 10

Control Byte

Byte 5 is the control field of the DCB and allows the user to select options for several types of disk drives. The format of this byte is as follows:

Bit	7	6	5	4	3	2	1	0
	r	a	0	0	0	s	s	s

Bit 7 Disables the four retries by the controller on all disk-access commands. Set this bit only during the evaluation of the performance of a disk drive.

Bit 6 If set to 0 during read commands, a reread is attempted when an ECC error occurs. If no error occurs during reread, the command will finish without an error status. If this bit is set to 1, no reread is attempted.

Bits 5, 4, 3 Set to 0.

Bits 2, 1, 0 These bits define the type of drive and select the step option. See the following figure.

Bits 2, 1, 0	
0 0 0	This drive is not specified and defaults to 3 milliseconds per step.
0 0 1	N/A
0 1 0	N/A
0 1 1	N/A
1 0 0	200 microseconds per step.
1 0 1	70 microseconds per step (specified by BIOS).
1 1 0	3 milliseconds per step.
1 1 1	3 milliseconds per step.

Command Byte

Test Drive Ready (Class 0, Opcode 00)

Bit	7	6	5	4	3	2	1	0
Byte 0	0	0	0	0	0	0	0	0
Byte 1	0	0	d	x	x	x	x	x

d = drive (0 or 1)
x = don't care
Bytes 2, 3, 4, 5, = don't care.

Recalibrate (Class 0, Opcode 00)

Bit	7	6	5	4	3	2	1	0
Byte 0	0	0	0	0	0	0	0	1
Byte 1	0	0	d	x	x	x	x	x
Byte 5	r	0	0	0	0	s	s	s

d = drive (0 or 1)
x = don't care
r = retries
s = Step Option
Bytes 2, 3, 4, = don't care
ch = cylinder high

Reserved (Class 0, Opcode 02)

This Opcode is not used.

Request Sense Status (Class 0, Opcode 03)

Bit	7	6	5	4	3	2	1	0
Byte 0	0	0	0	0	0	0	1	1
Byte 1	0	0	d	x	x	x	x	x

d = drive (0 or 1)
x = don't care
Bytes 2, 3, 4, 5, = don't care.

Format Drive (Class 0, Opcode 04)

Bit	7	6	5	4	3	2	1	0
Byte 0	0	0	0	0	0	1	0	0
Byte 1	0	0	d	Head No.				
Byte 2	ch	0	0	0	0	0	0	
Byte 3	Cylinder Low							
Byte 4	0	0	0	Interleave				
Byte 5	r	0	0	0	0	s	s	s

d = drive (0 or 1)
r = retries
s = Step Option
ch = cylinder high
Interleave 1 to 16
for 512-byte sectors.

Ready Verify (Class 0, Opcode 05)

Bit	7	6	5	4	3	2	1	0
Byte 0	0	0	0	0	0	1	0	1
Byte 1	0	0	d	Head No.				
Byte 2	ch	Sector No.						
Byte 3	Cylinder Low							
Byte 4	Block Count							
Byte 5	r	a	0	0	0	s	s	s

d = drive (0 or 1)
r = retries
s = Step Option
a = retry option on data ECC
ch = cylinder high
for 512-byte sectors.

Command	Data Control Block	Remarks
Format Track (Class 0, Opcode 06)	Bit 7 6 5 4 3 2 1 0 Byte 0 0 0 0 0 0 1 1 0 Byte 1 0 0 d Head No. Byte 2 ch 0 0 0 0 0 0 Byte 3 Cylinder Low Byte 4 0 0 0 Interleave Byte 5 r 0 0 0 0 s s s	d = drive (0 or 1) r = retries s = step option ch = cylinder high Interleave 1 to 16 for 512-byte sectors.
Format Bad Track (Class 0, Opcode 07)	Bit 7 6 5 4 3 2 1 0 Byte 0 0 0 0 0 0 1 1 1 Byte 1 0 0 d Head No. Byte 2 ch 0 0 0 0 0 0 Byte 3 Cylinder Low Byte 4 0 0 0 Interleave Byte 5 r 0 0 0 0 s s s	d = drive (0 or 1) x = don't care s = Step Option ch = cylinder high Interleave 1 to 16 for 512-byte sectors.
Read (Class 0, Opcode 08)	Bit 7 6 5 4 3 2 1 0 Byte 0 0 0 0 0 1 0 0 0 Byte 1 0 0 d Head No. Byte 2 ch Sector No. Byte 3 Cylinder Low Byte 5 r a 0 0 0 s s s	d = drive (0 or 1) r = retries a = retry option on data ECC error s = step option ch = cylinder high
Reserved (Class 0, Opcode 09)		This Opcode is not used.
Write (Class 0, Opcode 0A)	Bit 7 6 5 4 3 2 1 0 Byte 0 0 0 0 0 1 0 1 0 Byte 1 0 0 d Head No. Byte 2 ch Sector No. Byte 3 Cylinder Low Byte 4 Block Count Byte 5 r 0 0 0 0 s s s	d = drive (0 or 1) r = retries s = step option ch = cylinder high

Command	Data Control Block	Remarks
Seek (Class 0, Opcode 0B)	Bit 7 6 5 4 3 2 1 0 Byte 0 0 0 0 0 1 0 1 1 Byte 1 0 0 d Head No. Byte 2 ch 0 0 0 0 0 0 Byte 3 Cylinder Low Byte 4 x x x x x x x x Byte 5 r 0 0 0 0 s s s	d = drive (0 or 1) r = retries s = Step Option x = don't care
Initialize Drive Characteristics* (Class 0, Opcode 0C)	Bit 7 6 5 4 3 2 1 0 Byte 0 0 0 0 0 1 1 0 0	Bytes 1, 2, 3, 4, 5, = don't care.
Read ECC Burst Length (Class 0, Opcode 0D)	Bit 7 6 5 4 3 2 1 0 Byte 0 0 0 0 0 1 1 0 1	Bytes 1, 2, 3, 4, 5, = don't care.
Read Data from Sector Buffer (Class 0, Opcode 0E)	Bit 7 6 5 4 3 2 1 0 Byte 0 0 0 0 0 1 1 1 0	Bytes 1, 2, 3, 4, 5, = don't care.
Write Data to Sector Buffer (Class 0, Opcode 0F)	Bit 7 6 5 4 3 2 1 0 Byte 0 0 0 0 0 1 1 1 1	Bytes 1, 2, 3, 4, 5, = don't care.
RAM Diagnostic (Class 7, Opcode 00)	Bit 7 6 5 4 3 2 1 0 Byte 0 1 1 1 0 0 0 0 0	Bytes 1, 2, 3, 4, 5, = don't care.
Reserved (Class 7, Opcode 01)		This Opcode is not used.
Reserved (Class 7, Opcode 02)		This Opcode is not used.

*Initialize Drive Characteristics: The DBC must be followed by eight additional bytes.

Maximum number of cylinders	(2 bytes)
Maximum number of heads	(1 byte)
Start reduced write current cylinder	(2 bytes)
Start write precompensation cylinder	(2 bytes)
Maximum ECC data burst length	(1 byte)

Drive Diagnostic (Class 7, Opcode 03)

Bit	7	6	5	4	3	2	1	0
Byte 0	1	1	1	0	0	0	1	1
Byte 1	0	0	d	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	x	x	x	x	x	x	x	x
Byte 5	r	0	0	0	0	s	s	s

d = drive (0 or 1)
r = retries
s = step option
x = don't care

Controller Internal Diagnostics (Class 7, Opcode 04)

Bit	7	6	5	4	3	2	1	0
Byte 0	1	1	1	0	0	1	0	0

Bytes 1, 2, 3, 4, 5, = don't care.

Read Long * Track (Class 7, Opcode 05)

Bit	7	6	5	4	3	2	1	0
Byte 0	1	1	1	0	0	1	0	1
Byte 1	0	0	d	Head No.				
Byte 2	ch	Sector No.						
Byte 3	Cylinder Low							
Byte 4	Block Count							
Byte 5	r	0	0	0	0	s	s	s

d = drive (0 or 1)
r = retries
s = step option
ch = cylinder high

Write Long ** (Class 7, Opcode 06)

Bit	7	6	5	4	3	2	1	0
Byte 0	1	1	1	0	0	1	1	0
Byte 1	0	0	d	Head No.				
Byte 2	ch	Sector No.						
Byte 3	Cylinder Low							
Byte 4	Block Count							
Byte 5	r	0	0	0	0	s	s	s

d = drive (0 or 1)
s = step option
s = step option
ch = cylinder high
s = step option

* Returns 512 bytes plus 4 bytes of ECC data per sector.
** Requires 512 bytes plus 4 bytes of ECC data per sector.

Programming Summary

The two least-significant bits of the address bus are sent to the system board's I/O port decoder, which has two sections. One section is enabled by the I/O read signal (-IOR) and the other by the I/O write signal (-IOW). The result is a total of four read/write ports assigned to the disk controller board.

The address enable signal (AEN) is asserted by the system board when DMA is controlling data transfer. When AEN is active, the I/O port decoder is disabled.

The following figure is a table of the read/write ports.

R/W	Port Address	Function
Read Write	320 320	Read data (from controller to system unit) Write data (from system unit to controller)
Read Write	321 321	Read controller hardware status. Controller reset.
Read Write	322 322	Read option jumpers Generate controller-select-pulse
Read Write	323 323	Not used. Write pattern to DMA and interrupt mask register.

Interface

The following lines are used by the disk controller:

A0–A19 Positive true 20-bit address. The least-significant 10 bits contain the I/O address within the range of hex 320 to hex 323 when an I/O read or write is executed by the system unit. The full 20 bits are decoded to address the read-only memory (ROM) between the addresses of hex C8000 and hex C9FFF.

DO–D7 Positive 8-bit data bus over which data and status information is passed between the system board and the controller.

-IOR This signal is active when the system board reads status or data from the controller under either programmed I/O or DMA control.

-IOW This signal is active when the system board sends a command or data to the controller under either programmed I/O or DMA control.

AEN This signal is active when the DMA in the system board is generating the I/O Read (-IOR) or I/O Write (-IOW) signals and has control of the address and data buses.

RESET This signal forces the disk controller to its initial power-up condition.

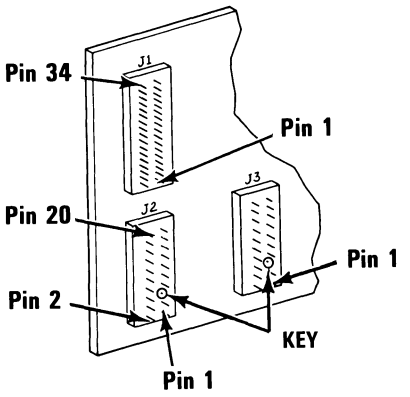
IRQ 5 This signal is active by the controller when enabled to interrupt the system board on the return ending status byte from the controller.

DRQ 3 This signal is activated by the controller when data is available for transfer to or from the controller under DMA control. This signal remains active until the system board's DMA channel activates the DMA-acknowledge signal (-DACK 3) in response.

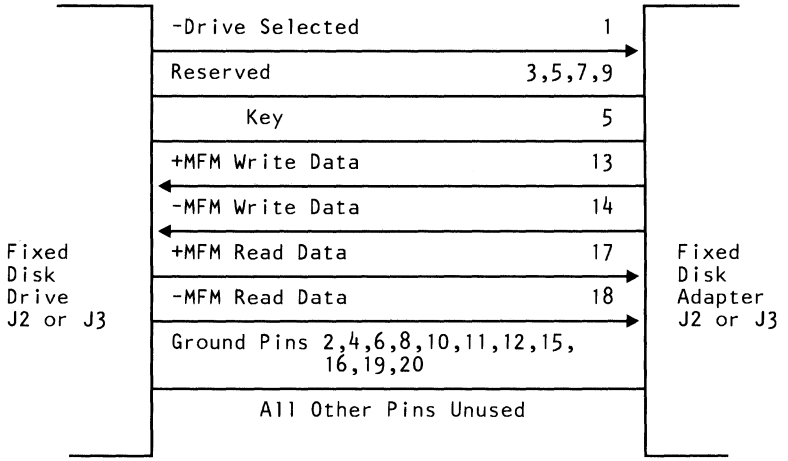
-DACK 3 This signal is active when negative, and is generated by the system board DMA channel in response to a DMA request (DRQ 3).

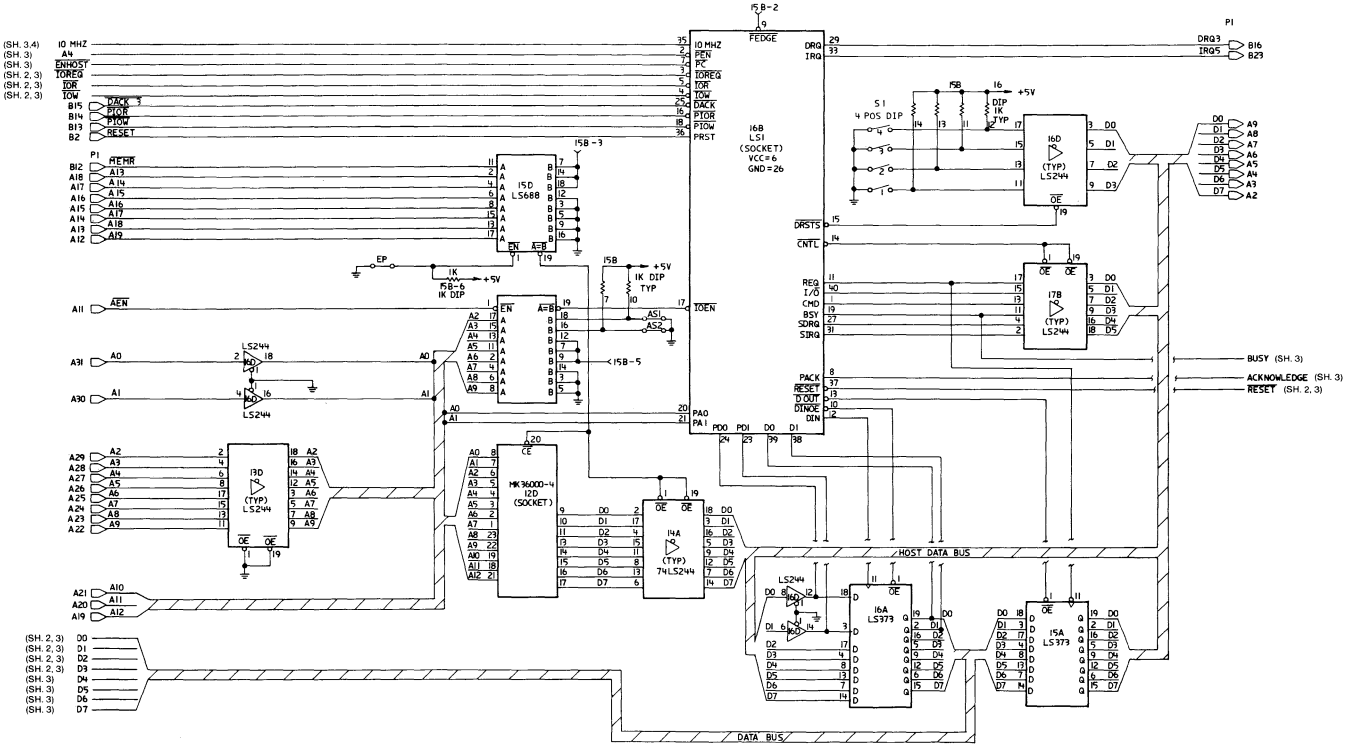
Connectors

The 20MB Fixed Disk Drive Adapter connector and interface specifications follow.

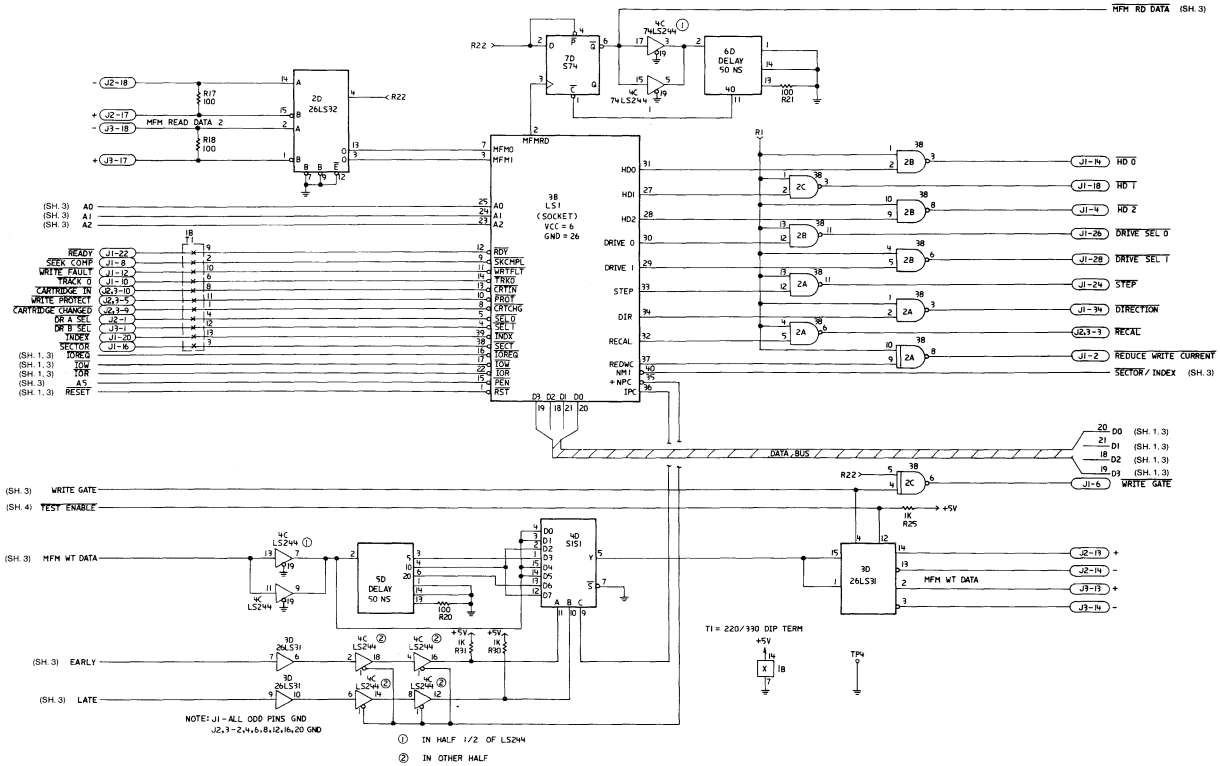


Disk Drive J1	At Standard TTL Levels	Land Number	Disk Adapter J1
	Ground–Odd Numbers	1–33	
	-Reserved	2, 16, 30, 32	
	-Head Select 2	4	
←	-Write Gate	6	
←	-Seek Complete	8	
→	-Track 000	10	
→	-Write Fault	12	
→	-Head Select 0	14	
←	-Head Select 1	18	
←	-Index	20	
→	-Ready	22	
→	-Step	24	
←	-Drive Select 1	26	
←	-Drive Select 2	28	
←	-Drive Select 3	30	
←	-Drive Select 4	32	
←	-Direction In	34	
←			

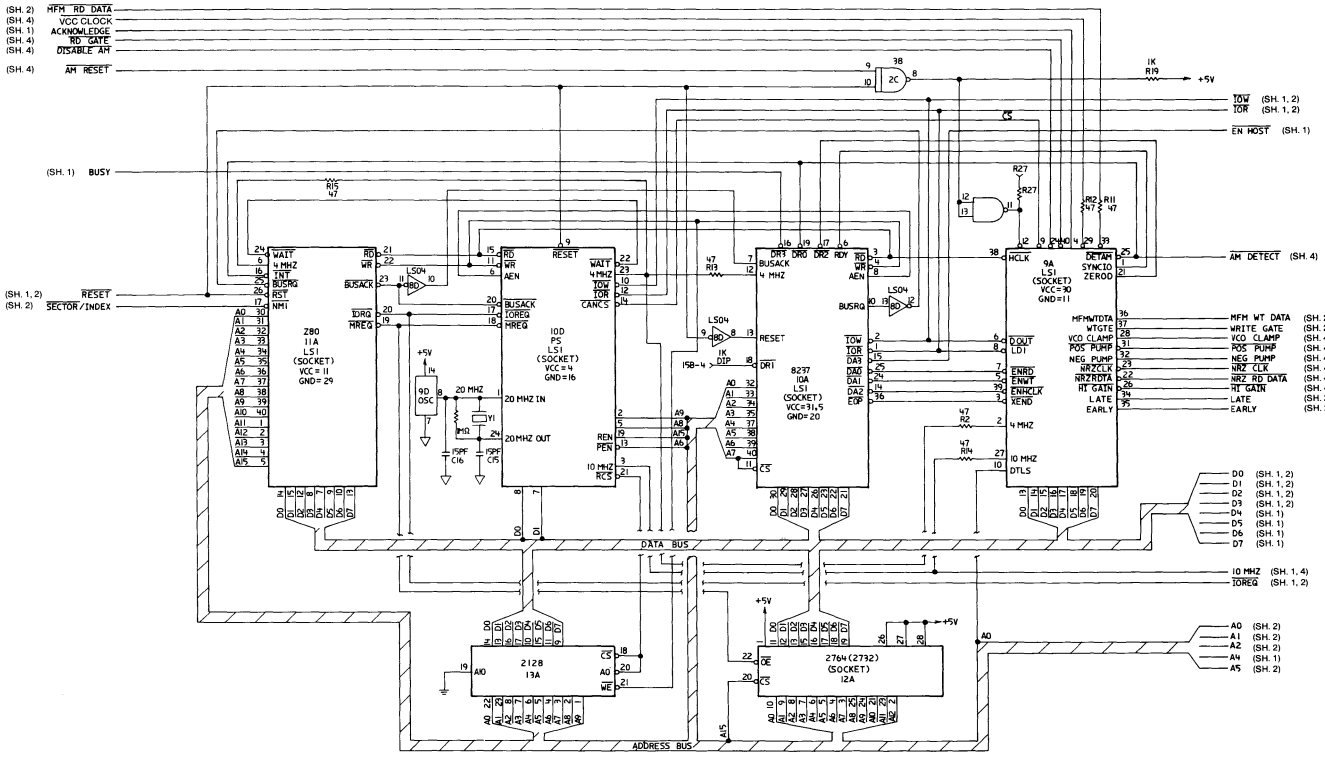

Fixed Disk Drive J2 or J3
-Drive Selected 1
Reserved 3,5,7,9
Key 5
+MFM Write Data 13
-MFM Write Data 14
+MFM Read Data 17
-MFM Read Data 18
Ground Pins 2,4,6,8,10,11,12,15, 16,19,20
All Other Pins Unused
Fixed Disk Adapter J2 or J3



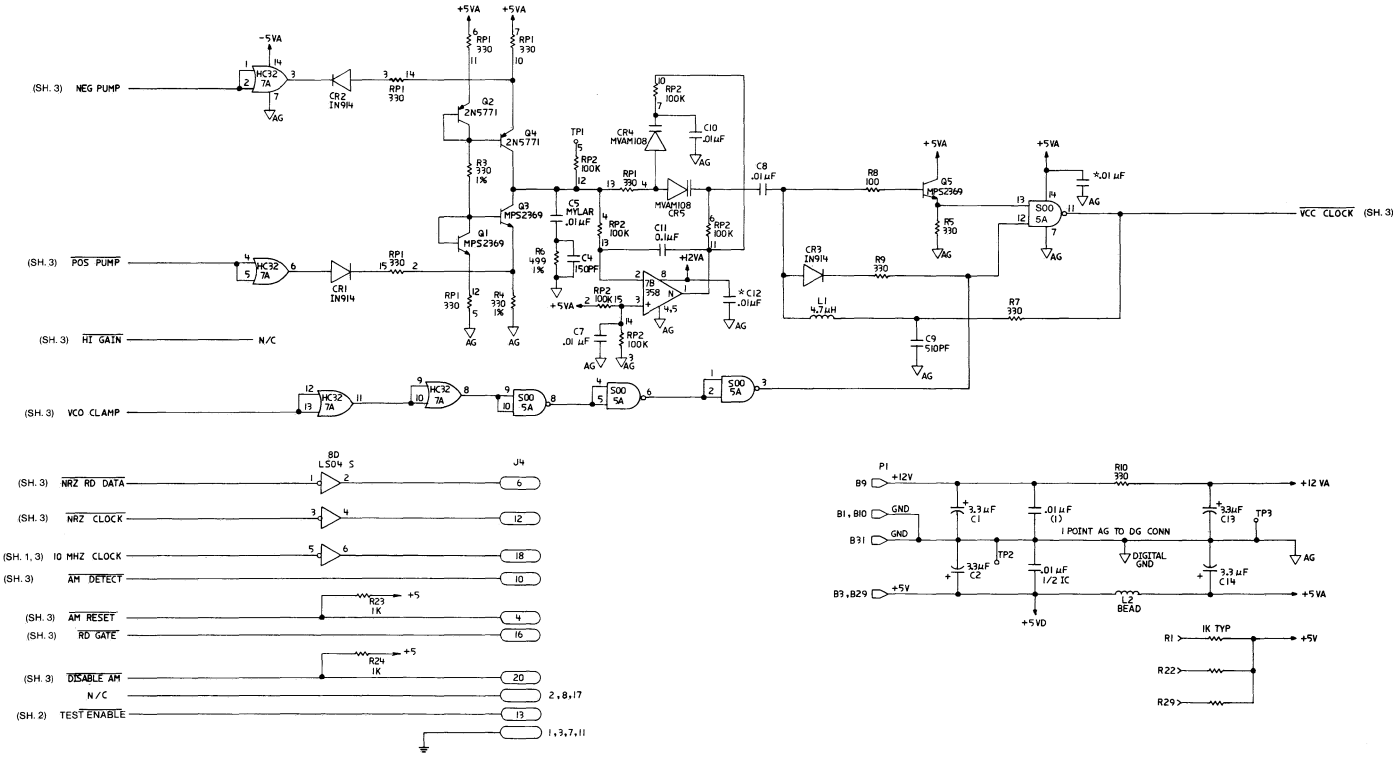
20MB Fixed Disk Drive Adapter (Sheet 1 of 4)



20MB Fixed Disk Drive Adapter (Sheet 2 of 4)



20MB Fixed Disk Drive Adapter (Sheet 3 of 4)



20MB Fixed Disk Drive Adapter (Sheet 4 of 4)

BIOS Listing

The BIOS Listing for the IBM 20MB Fixed Disk Drive Adapter follows.

```
                        PAGE    118,121
                        TITLE DISK2 ---- 10/28/85  FIXED DISK BIOS

                        ;--  INT  13H  ------------------------------------------------
                        ;                                                             :
                        ; FIXED DISK I/O INTERFACE                                    :
                        ;                                                             :
                        ;       THIS INTERFACE PROVIDES ACCESS TO FIXED DISKS         :
                        ;       THROUGH THE IBM FIXED DISK CONTROLLER.                :
                        ;                                                             :
                        ;-------------------------------------------------------------

                        ;-------------------------------------------------------------
                        ;       THE  BIOS ROUTINES ARE MEANT TO BE ACCESSED THROUGH   :
                        ;       SOFTWARE INTERRUPTS ONLY.  ANY ADDRESSES PRESENT IN   :
                        ;       THE LISTINGS  ARE INCLUDED  ONLY FOR  COMPLETENESS,   :
                        ;       NOT FOR  REFERENCE.   APPLICATIONS WHICH  REFERENCE   :
                        ;       ABSOLUTE   ADDRESSES   WITHIN   THE   CODE  SEGMENT   :
                        ;       VIOLATE THE STRUCTURE AND DESIGN OF BIOS.              :
                        ;-------------------------------------------------------------
                        ;
                        ; INPUT (AH = HEX VALUE)
                        ;
                        ;       (AH)= 00H  RESET DISK (DL = 80H,81H) / DISKETTE
                        ;       (AH)= 01H  READ THE STATUS OF THE LAST DISK OPERATION INTO (AL)
                        ;                  NOTE: DL < 80H - DISKETTE
                        ;                        DL > 80H - DISK
                        ;       (AH)= 02H  READ THE DESIRED SECTORS INTO MEMORY
                        ;       (AH)= 03H  WRITE THE DESIRED SECTORS FROM MEMORY
                        ;       (AH)= 04H  VERIFY THE DESIRED SECTORS
                        ;       (AH)= 05H  FORMAT THE DESIRED TRACK
                        ;       (AH)= 06H  FORMAT THE DESIRED TRACK AND SET BAD SECTOR FLAGS
                        ;       (AH)= 07H  FORMAT THE DRIVE STARTING AT THE DESIRED TRACK
                        ;       (AH)= 08H  RETURN THE CURRENT DRIVE PARAMETERS
                        ;
                        ;       (AH)= 09H  INITIALIZE DRIVE PAIR CHARACTERISTICS
                        ;                  INTERRUPT 41H POINTS TO DATA BLOCK
                        ;       (AH)= 0AH  READ LONG
                        ;       (AH)= 0BH  WRITE LONG
                        ;                  NOTE: READ AND WRITE LONG ENCOMPASS
                        ;                  512 BYTES + 4 BYTES OF ECC
                        ;       (AH)= 0CH  SEEK
                        ;       (AH)= 0DH  ALTERNATE DISK RESET (SEE DL)
                        ;       (AH)= 0EH  READ SECTOR BUFFER
                        ;       (AH)= 0FH  WRITE SECTOR BUFFER,
                        ;                  (RECOMMENDED PRACTICE BEFORE FORMATTING)
                        ;       (AH)= 10H  TEST DRIVE READY
                        ;       (AH)= 11H  RECALIBRATE
                        ;       (AH)= 12H  CONTROLLER RAM DIAGNOSTIC
                        ;       (AH)= 13H  DRIVE DIAGNOSTIC
                        ;       (AH)= 14H  CONTROLLER INTERNAL DIAGNOSTIC
                        ;
                        ;                  REGISTERS USED FOR FIXED DISK OPERATIONS
                        ;
                        ;                  (DL)     -  DRIVE NUMBER      (80H-87H FOR DISK, VALUE CHECKED)
                        ;                  (DH)     -  HEAD NUMBER       (0-7D ALLOWED, NOT VALUE CHECKED)
                        ;                  (CH)     -  CYLINDER NUMBER   (0-1023D, NOT VALUE CHECKED)(SEE CL)
                        ;                  (CL)     -  SECTOR NUMBER     (1-17D, NOT VALUE CHECKED)
                        ;
                        ;                             NOTE: HIGH 2 BITS OF CYLINDER NUMBER ARE PLACED
                        ;                                   IN THE HIGH 2 BITS OF THE CL REGISTER
                        ;                                   (10 BITS TOTAL)
                        ;                  (AL)     -  NUMBER OF SECTORS (MAXIMUM POSSIBLE RANGE 1-80H,
                        ;                                                 FOR READ/WRITE LONG 1-79H)
                        ;                             (INTERLEAVE VALUE FOR FORMAT 1-16D)
                        ;                  (ES:BX)  -  ADDRESS OF BUFFER FOR READS AND WRITES,
                        ;                             (NOT REQUIRED FOR VERIFY)
                        ;
                        ; OUTPUT
                        ;       AH = STATUS OF CURRENT OPERATION
                        ;            STATUS BITS ARE DEFINED IN THE EQUATES BELOW
                        ;       CY = 0  SUCCESSFUL OPERATION (AH= 00H ON RETURN)
                        ;       CY = 1  FAILED OPERATION (AH HAS ERROR REASON)
                        ;
                        ;       NOTE:   ERROR 11H  INDICATES THAT THE DATA READ HAD A RECOVERABLE
                        ;               ERROR WHICH WAS CORRECTED BY THE ECC ALGORITHM.  THE DATA
                        ;               IS PROBABLY GOOD,  HOWEVER THE BIOS ROUTINE INDICATES AN
                        ;               ERROR TO ALLOW THE CONTROLLING PROGRAM A CHANCE TO DECIDE
                        ;               FOR  ITSELF.  THE  ERROR  MAY  NOT  RECUR  IF THE DATA IS
                        ;               REWRITTEN. (AL) CONTAINS THE BURST LENGTH.
                        ;
                        ;       IF DRIVE PARAMETERS WERE REQUESTED,
                        ;
                        ;               DL = NUMBER OF CONSECUTIVE ACKNOWLEDGING DRIVES
                        ;                    ATTACHED (0-2)  (CONTROLLER CARD ZERO TALLY ONLY)
                        ;               DH = MAXIMUM USEABLE VALUE FOR HEAD NUMBER
                        ;               CH = MAXIMUM USEABLE VALUE FOR CYLINDER NUMBER
                        ;               CL = MAXIMUM USEABLE VALUE FOR SECTOR NUMBER
                        ;                    AND CYLINDER NUMBER HIGH BITS
                        ;
                        ;       IF AN ERROR OCCURS ON READ DRIVE PARAMETERS,
                        ;
                        ;               AH = ERROR CODE (INIT_FAIL)
                        ;               AL = CX = DX = 0
                        ;
                        ;       REGISTERS WILL BE PRESERVED EXCEPT WHEN THEY ARE USED TO RETURN
                        ;       INFORMATION.
                        ;
                        ;       NOTE: IF AN ERROR IS REPORTED BY THE DISK CODE, THE APPROPRIATE
                        ;             ACTION IS TO RESET THE DISK, THEN RETRY THE OPERATION.
                        ;
                        ;--------------------------------------------------------------------------
```

```
                        PAGE
                        ;--------------------------------------------------------
                        ;       ERROR RETURN STATUS  (AH)= ??H  WHEN  CY= 1      :
                        ;--------------------------------------------------------

= 00FF                  SENSE_FAIL      EQU     0FFH            ; SENSE OPERATION FAILED
= 00CC                  WRITE_FAULT     EQU     0CCH            ; WRITE FAULT ON SELECTED DRIVE
= 00BB                  UNDEF_ERR       EQU     0BBH            ; UNDEFINED ERROR OCCURRED
= 0080                  TIME_OUT        EQU     080H            ; ATTACHMENT FAILED TO RESPOND
= 0040                  BAD_SEEK        EQU     040H            ; SEEK OPERATION FAILED
= 0020                  BAD_CNTLR       EQU     020H            ; CONTROLLER HAS FAILED
= 0011                  DATA_CORRECTED  EQU     011H            ; ECC CORRECTED DATA ERROR
= 0010                  BAD_ECC         EQU     010H            ; BAD ECC ON DISK READ
= 000B                  BAD_TRACK       EQU     00BH            ; BAD TRACK FLAG DETECTED
= 0009                  DMA_BOUNDARY    EQU     009H            ; ATTEMPT TO DMA ACROSS 64K BOUNDARY
= 0007                  INIT_FAIL       EQU     007H            ; DRIVE PARAMETER ACTIVITY FAILED
= 0005                  BAD_RESET       EQU     005H            ; RESET FAILED
= 0004                  RECORD_NOT_FND  EQU     004H            ; REQUESTED SECTOR NOT FOUND
= 0002                  BAD_ADDR_MARK   EQU     002H            ; ADDRESS MARK NOT FOUND
= 0001                  BAD_CMD         EQU     001H            ; BAD COMMAND PASSED TO DISK I/O

                        ;--------------------------------------
                        ;       INTERRUPT AND STATUS AREAS     :
                        ;--------------------------------------

0000                    ABS0    SEGMENT AT 0H
0034                            ORG     00DH*4          ; FIXED DISK INTERRUPT VECTOR
0034                    HDISK_INT       LABEL   DWORD
004C                            ORG     013H*4          ; DISK INTERRUPT VECTOR
004C                    ORG_VECTOR      LABEL   DWORD
0064                            ORG     019H*4          ; BOOTSTRAP INTERRUPT VECTOR
0064                    BOOT_VEC        LABEL   DWORD
0078                            ORG     01EH*4          ; DISKETTE PARAMETERS
0078                    DISKETTE_PARM   LABEL   DWORD
0100                            ORG     040H*4          ; NEW DISKETTE INTERRUPT VECTOR
0100                    DISK_VECTOR     LABEL   DWORD
0104                            ORG     041H*4          ; FIXED DISK PARAMETER VECTOR
0104                    HF_TBL_VEC      LABEL   DWORD
7C00                            ORG     7C00H           ; BOOTSTRAP LOADER VECTOR
7C00                    BOOT_LOCN       LABEL   FAR
7C00                    ABS0    ENDS

0000                    DATA    SEGMENT AT 40H
006C                            ORG     06CH
006C ????               TIMER_LOW       DW      ?               ; TIMER LOW WORD
0072                            ORG     072H
0072 ????               RESET_FLAG      DW      ?               ; 1234H IF KEYBOARD RESET UNDERWAY
0074                            ORG     074H
0074 ??                 DISK_STATUS     DB      ?               ; FIXED DISK STATUS BYTE
0075 ??                 HF_NUM          DB      ?               ; COUNT OF FIXED DISK DRIVES
0076 ??                 CONTROL_BYTE    DB      ?               ; CONTROL BYTE DRIVE OPTIONS
0077 ??                 PORT_OFF        DB      ?               ; PORT OFFSET
0078                    DATA    ENDS

0000                    CODE    SEGMENT

                        ;--------------------------------------------------------
                        ; HARDWARE SPECIFIC VALUES                              :
                        ;                                                       :
                        ;  - CONTROLLER I/O PORT                                :
                        ;      > WHEN READ FROM:                                :
                        ;        HF_PORT+0 - READ DATA (FROM CONTROLLER TO CPU)  :
                        ;        HF_PORT+1 - READ CONTROLLER HARDWARE STATUS     :
                        ;                    (CONTROLLER TO CPU)                 :
                        ;        HF_PORT+2 - READ CONFIGURATION SWITCHES         :
                        ;        HF_PORT+3 - NOT USED                            :
                        ;      > WHEN WRITTEN TO:                               :
                        ;        HF_PORT+0 - WRITE DATA (FROM CPU TO CONTROLLER) :
                        ;        HF_PORT+1 - CONTROLLER RESET                    :
                        ;        HF_PORT+2 - GENERATE CONTROLLER SELECT PULSE    :
                        ;        HF_PORT+3 - WRITE PATTERN TO DMA AND INTERRUPT  :
                        ;                    MASK REGISTER                       :
                        ;                                                       :
                        ;--------------------------------------------------------

=                       CMD_BLOCK       EQU     BYTE PTR [BP]-8 ; CMD_BLOCK HEAD
= 0320                  HF_PORT         EQU     0320H           ; DISK PORT
= 0020                  INTA00          EQU     020H            ; 8259 PORT
= 0021                  INTA01          EQU     021H            ; 8259 PORT
= 0020                  EOI             EQU     020H            ; END OF INTERRUPT COMMAND
= 0008                  RI_BUSY         EQU     00001000B       ; DISK PORT 1 BUSY BIT
= 0004                  RI_BUS          EQU     00000100B       ;             COMMAND/DATA BIT
= 0002                  RI_IOMODE       EQU     00000010B       ;             MODE BIT
= 0001                  RI_REQ          EQU     00000001B       ;             REQUEST BIT

= 0047                  DMA_READ        EQU     01000111B       ; CHANNEL 3 (047H)
= 004B                  DMA_WRITE       EQU     01001011B       ; CHANNEL 3 (04BH)
= 0000                  DMA             EQU     000H            ; DMA ADDRESS
= 0082                  DMA_HIGH        EQU     082H            ; PORT FOR HIGH 4 BITS OF DMA

= 0000                  TST_RDY_CMD     EQU     00000000B       ; CNTLR READY (00H)
= 0001                  RECAL_CMD       EQU     00000001B       ;       RECAL (01H)
= 0003                  SENSE_CMD       EQU     00000011B       ;       SENSE (03H)
= 0004                  FMTDRV_CMD      EQU     00000100B       ;       DRIVE (04H)
= 0005                  CHK_TRK_CMD     EQU     00000101B       ;       T CHK (05H)
= 0006                  FMTTRK_CMD      EQU     00000110B       ;       TRACK (06H)
= 0007                  FMTBAD_CMD      EQU     00000111B       ;       BAD   (07H)
= 0008                  READ_CMD        EQU     00001000B       ;       READ  (08H)
= 000A                  WRITE_CMD       EQU     00001010B       ;       WRITE (0AH)
= 000B                  SEEK_CMD        EQU     00001011B       ;       SEEK  (0BH)
= 000C                  IND_RDY_CMD     EQU     00001100B       ;       INIT  (0CH)
= 000D                  RD_ECC_CMD      EQU     00001101B       ;       BURST (0DH)
= 000E                  RD_BUFF_CMD     EQU     00001110B       ;       BUFFR (0EH)
= 000F                  WR_BUFF_CMD     EQU     00001111B       ;       BUFFR (0FH)
= 00E0                  RAM_DIAG_CMD    EQU     11100000B       ;       RAM   (E0H)
= 00E3                  CHK_DRV_CMD     EQU     11100011B       ;       DRV   (E3H)
= 00E4                  CNTLR_DIAG_CMD  EQU     11100100B       ;       CNTLR (E4H)
= 00E5                  RD_LONG_CMD     EQU     11100101B       ;       RLONG (E5H)
= 00E6                  WR_LONG_CMD     EQU     11100110B       ;       WLONG (E6H)

= 0008                  MAX_FILE        EQU     8
= 0002                  S_MAX_FILE      EQU     2
```

```
                        PAGE
                                ASSUME  CS:CODE,DS:ABS0
0000                            ORG     0H
0000 55                         DB      055H                    ; GENERIC BIOS HEADER
0001 AA                         DB      0AAH
0002 08                         DB      08D                     ; 4K MODULE

                        ;--------------------------------------------------------------
                        ; FIXED DISK I/O SETUP                                        :
                        ;                                                            :
                        ;  - ESTABLISH TRANSFER VECTORS FOR THE FIXED DISK           :
                        ;  - PERFORM POWER ON DIAGNOSTICS                            :
                        ;    SHOULD AN ERROR OCCUR A "1701" MESSAGE IS DISPLAYED     :
                        ;                                                            :
                        ;--------------------------------------------------------------

0003                    DISK_SETUP      PROC    FAR
0003 EB 35                      JMP     SHORT   L3
0005 35 39 58 37 32 39          DB      '59X7291 (C) COPYRIGHT IBM  CORP.'      ; COPYRIGHT NOTICE
     31 20 28 43 29 20
     43 4F 50 59 52 49
     47 48 54 20 49 42
     4D 20 20 43 4F 52
     50 2E
0025 2C 31 39 38 32 20          DB      ',1982 ,1985.'
     2C 31 39 38 35 2E
0031 20 31 30 2F 32 38          DB      ' 10/28/85'                             ; RELEASE MARKER
     2F 38 35
003A                    L3:
003A 2B C0                      SUB     AX,AX                                   ; ADDRESS LOW RAM
003C 8E D8                      MOV     DS,AX
003E FA                         CLI
003F A1 004C R                  MOV     AX,WORD PTR ORG_VECTOR                  ; LOAD DISKETTE IP
0042 A3 0100 R                  MOV     WORD PTR DISK_VECTOR,AX                 ; STORE AT INT 40H
0045 A1 004E R                  MOV     AX,WORD PTR ORG_VECTOR+2                ; LOAD DISKETTE CS
0048 A3 0102 R                  MOV     WORD PTR DISK_VECTOR+2,AX               ; STORE AT INT 40H
004B C7 06 004C R 0251 R        MOV     WORD PTR ORG_VECTOR, OFFSET DISK_IO     ; FIXED DISK HANDLER
0051 8C 0E 004E R               MOV     WORD PTR ORG_VECTOR+2,CS                ;  AT INT 13H
0055 B8 0755 R                  MOV     AX, OFFSET HD_INT                       ; FIXED DISK INTERRUPT
0058 A3 0034 R                  MOV     WORD PTR HDISK_INT,AX                   ;  HANDLER AT INT 0DH
005B 8C 0E 0036 R               MOV     WORD PTR HDISK_INT+2,CS
005F C7 06 0064 R 0192 R        MOV     WORD PTR BOOT_VEC,OFFSET BOOT_STRAP     ; BOOTSTRAP ROUTINE AT
0065 8C 0E 0066 R               MOV     WORD PTR BOOT_VEC+2,CS                  ;  INT 19H
0069 C7 06 0104 R 03FF R        MOV     WORD PTR HF_TBL_VEC,OFFSET FD_TBL       ; PARAMETER TABLE AT
006F 8C 0E 0106 R               MOV     WORD PTR HF_TBL_VEC+2,CS                ;  INT 41H
0073 FB                         STI

                                ASSUME  DS:DATA
0074 B8 ---- R                  MOV     AX,DATA                 ; ESTABLISH SEGMENT
0077 8E D8                      MOV     DS,AX
0079 C6 06 0074 R 00            MOV     DISK_STATUS,0           ; RESET THE STATUS INDICATOR
007E C6 06 0075 R 00            MOV     HF_NUM,0                ; ZERO COUNT OF DRIVES
0083 C6 06 0077 R 00            MOV     PORT_OFF,0              ; ZERO CARD OFFSET
0088 B9 0025                    MOV     CX,25H                  ; RETRY COUNT
008B                    L4:
008B E8 0177 R                  CALL    HD_RESET_1              ; RESET CONTROLLER
008E 73 05                      JNC     L7
0090 E2 F9                      LOOP    L4                      ; TRY RESET AGAIN
0092 E9 0154 R                  JMP     ERROR_EX
0095                    L7:
0095 B9 0001                    MOV     CX,1
0098 BA 0080                    MOV     DX,80H
009B B8 1200                    MOV     AX,1200H                ; CONTROLLER DIAGNOSTICS
009E CD 13                      INT     13H                     ;  CHECK THE INTERNAL RAM
00A0 73 03                      JNC     P7                      ;  BUFFERS
00A2 E9 0154 R                  JMP     ERROR_EX
00A5                    P7:
00A5 B8 1400                    MOV     AX,1400H                ; CONTROLLER DIAGNOSTICS
00A8 CD 13                      INT     13H                     ;  INTERNAL CHECKSUM AND
00AA 73 03                      JNC     P9                      ;  ECC CIRCUITRY TEST.
00AC E9 0154 R                  JMP     ERROR_EX
00AF                    P9:
00AF C7 06 006C R 0000          MOV     TIMER_LOW,0             ; ZERO TIMER
00B5 81 3E 0072 R 1234          CMP     RESET_FLAG,1234H        ; KEYBOARD RESET
00BB 75 06                      JNE     P8
00BD C7 06 006C R 019A          MOV     TIMER_LOW,410D          ; SKIP WAIT ON RESET
00C3                    P8:
00C3 FA                         CLI                             ; DISABLE INTERRUPTS
00C4 E4 21                      IN      AL,INTA01               ; TIMER
00C6 24 FE                      AND     AL,0FEH                 ; ENABLE TIMER
00C8 E6 21                      OUT     INTA01,AL               ; START TIMER
00CA FB                         STI                             ; INTERRUPTS ON
00CB                    P4:
00CB E8 0177 R                  CALL    HD_RESET_1              ; RESET CONTROLLER
00CE 72 07                      JC      P10
00D0 B8 1000                    MOV     AX,1000H                ; TEST TO SEE IF THE DRIVE
00D3 CD 13                      INT     13H                     ;  IS READY
00D5 73 0A                      JNC     P2
00D7                    P10:
00D7 A1 006C R                  MOV     AX,TIMER_LOW
00DA 3D 01BE                    CMP     AX,446D                 ; 25 SECONDS
00DD 72 EC                      JB      P4
00DF EB 73                      JMP     SHORT ERROR_EX
00E1                    P2:
00E1 B8 1100                    MOV     AX,1100H                ; RECALIBRATE THE DRIVE 0
00E4 CD 13                      INT     13H
00E6 72 6C                      JC      ERROR_EX

00E8 B8 0900                    MOV     AX,0900H                ; SET DRIVE PARAMETERS
00EB CD 13                      INT     13H                     ;  FOR DRIVE 0
00ED 72 65                      JC      ERROR_EX

00EF B8 C800                    MOV     AX,0C800H               ; DMA TO BUFFER
00F2 8E C0                      MOV     ES,AX                   ; SET SEGMENT
00F4 2B DB                      SUB     BX,BX
00F6 B8 0F00                    MOV     AX,0F00H                ; WRITE SECTOR BUFFER
00F9 CD 13                      INT     13H
00FB 72 57                      JC      ERROR_EX

00FD FE 06 0075 R               INC     HF_NUM                  ; DRIVE ZERO RESPONDED
0101 BA 0213                    MOV     DX,213H                 ; EXPANSION BOX
0104 B0 00                      MOV     AL,0
0106 EE                         OUT     DX,AL                   ; TURN BOX OFF
0107 BA 0321                    MOV     DX,321H                 ; TEST IF CONTROLLER
```

```
010A EC                         IN      AL,DX                   ; ... IS IN THE SYSTEM UNIT
010B 24 0F                      AND     AL,0FH
010D 3C 0F                      CMP     AL,0FH
010F 74 06                      JE      BOX_ON
0111 C7 06 006C R 01A4          MOV     TIMER_LOW,420D          ; CONTROLLER IS IN SYSTEM UNIT
0117                    BOX_ON:
0117 BA 0213                    MOV     DX,213H                 ; EXPANSION BOX
011A B0 FF                      MOV     AL,0FFH
011C EE                         OUT     DX,AL                   ; TURN BOX ON
011D B9 0001                    MOV     CX,1                    ; ATTEMPT NEXT DRIVES
0120 BA 0081                    MOV     DX,081H
0123                    P3:
0123 2B C0                      SUB     AX,AX                   ; RESET THE CONTROLLER
0125 CD 13                      INT     13H
0127 72 42                      JC      POD_DONE
0129 B8 1100                    MOV     AX,01100H               ; RECALIBRATE THE DRIVE 1
012C CD 13                      INT     13H
012E 73 0A                      JNC     P5
0130 A1 006C R                  MOV     AX,TIMER_LOW
0133 3D 01BE                    CMP     AX,446D                 ; 25 SECONDS
0136 72 EB                      JB      P3
0138 EB 31                      JMP     SHORT POD_DONE
013A                    P5:
013A B8 0900                    MOV     AX,0900H                ; INITIALIZE DRIVE CHARACTERISTICS
013D CD 13                      INT     13H                     ;  FOR DRIVE 1
013F 72 2A                      JC      POD_DONE
0141 FE 06 0075 R               INC     HF_NUM                  ; TALLY ANOTHER DRIVE
0145 81 FA 0081                 CMP     DX,(80H + S_MAX_FILE - 1)
0149 73 20                      JAE     POD_DONE
014B 42                         INC     DX
014C EB D5                      JMP     P3

014E 31 37 30 31 0D 0A  F17     DB      '1701',0DH,0AH          ; POST MESSAGE
= 0006                  F17L    EQU     $-F17

                        ;-----  POD ERROR

0154                    ERROR_EX:
0154 BD 000F                    MOV     BP,0FH                  ; POD ERROR FLAG
0157 2B F6                      SUB     SI,SI
0159 B9 0006                    MOV     CX,F17L                 ; MESSAGE CHARACTER COUNT
015C B7 00                      MOV     BH,0                    ; PAGE ZERO
015E                    OUT_CH:
015E 2E: 8A 84 014E R           MOV     AL,CS:F17[SI]           ; GET BYTE
0163 B4 0E                      MOV     AH,14D                  ; VIDEO OUT
0165 CD 10                      INT     10H                     ; DISPLAY CHARACTER
0167 46                         INC     SI                      ; NEXT CHAR
0168 E2 F4                      LOOP    OUT_CH                  ; DO MORE
016A F9                         STC
016B                    POD_DONE:
016B FA                         CLI                             ; NO INTERRUPTS
016C E4 21                      IN      AL,INTA01               ; READ THE INTERRUPT MASK
016E 0C 01                      OR      AL,01H                  ; DISABLE THE TIMER
0170 E6 21                      OUT     INTA01,AL
0172 FB                         STI                             ; ENABLE INTERRUPTS
0173 E8 0232 R                  CALL    DSBL                    ; DISABLE THE CARD MASKS
0176 CB                         RET

0177                    HD_RESET_1      PROC    NEAR
0177 51                         PUSH    CX                      ; SAVE REGISTER
0178 52                         PUSH    DX
0179 B9 0100                    MOV     CX,0100H                ; RETRY COUNT
017C                    L6:
017C E8 076D R                  CALL    PORT_0
017F 42                         INC     DX                      ; ADDRESS PORT_1
0180 EE                         OUT     DX,AL                   ; RESET CARD
0181 EB 00                      JMP     $+2                     ; I/O DELAY AT LEAST +5us
0183 EB 00                      JMP     $+2                     ; ALLOW TIME TO CLEAR THE
0185 EB 00                      JMP     $+2                     ;  HARDWARE STATUS REGISTER
0187 EC                         IN      AL,DX                   ; READ THE HARDWARE STATUS
0188 24 3F                      AND     AL,00111111B            ; MASK OFF UPPER 2 BITS AND CLEAR CY
018A 74 03                      JZ      R3                      ; EXIT IF REGISTER IS CLEARED WITH CY=0
018C E2 EE                      LOOP    L6                      ; TRY AGAIN
018E F9                         STC                             ; SET ERROR CONDITION CY=1
018F                    R3:
018F 5A                         POP     DX                      ; RESTORE REGISTER
0190 59                         POP     CX
0191 C3                         RET
0192                    HD_RESET_1      ENDP
0192                    DISK_SETUP      ENDP
```

```
                        PAGE
                        ;---  INT  19 H ----------------------------------------------
                        ;                                                            :
                        ; INTERRUPT 19 BOOT STRAP LOADER                            :
                        ;                                                            :
                        ;  -  THE FIXED DISK BIOS REPLACES THE INTERRUPT 19H BOOT    :
                        ;       STRAP VECTOR WITH A POINTER TO THIS BOOT ROUTINE AND  :
                        ;       RESETS THE DEFAULT DISK AND DISKETTE PARAMETER VECTORS:
                        ;                                                            :
                        ;  -  THE BOOT BLOCK TO BE READ IN WILL BE ATTEMPTED FROM    :
                        ;       CYLINDER 0 SECTOR 1 OF THE DEVICE.                   :
                        ;                                                            :
                        ;  -  THE BOOTSTRAP SEQUENCE IS:                            :
                        ;       ATTEMPT TO LOAD FROM THE DISKETTE INTO THE BOOT      :
                        ;       LOCATION (0000:7C00H) WHERE CONTROL IS TRANSFERRED.  :
                        ;       IF THE DISKETTE FAILS THE FIXED DISK IS TRIED FOR A  :
                        ;       VALID BOOTSTRAP BLOCK. A VALID BOOT BLOCK ON THE     :
                        ;       FIXED DISK CONSISTS OF THE BYTES  055H 0AAH  AS THE  :
                        ;       LAST TWO BYTES OF THE BLOCK.                         :
                        ;       IF THE ABOVE FAILS CONTROL IS PASSED TO RESIDENT BASIC:
                        ;                                                            :
                        ;------------------------------------------------------------

0192                    BOOT_STRAP:
                                ASSUME  DS:ABS0,ES:ABS0
0192 2B C0                      SUB     AX,AX
0194 8E D8                      MOV     DS,AX                   ; ESTABLISH SEGMENT
0196 B4 C0                      MOV     AH,0C0H
0198 CD 15                      INT     15H                     ; READ CONFIGURATION PARAMETERS
                                                                ; IF XT OR PC, INTERRUPTS ARE DISABLED
                        ;-----  RESET PARAMETER VECTORS         ;  AT THIS POINT.

019A FA                         CLI
019B C7 06 0104 R 03FF R        MOV     WORD PTR HF_TBL_VEC,OFFSET FD_TBL
01A1 8C 0E 0106 R               MOV     WORD PTR HF_TBL_VEC+2,CS
01A5 73 0A                      JNC     H0                      ; JMP IF INT 15 FUNCTION IMPLEMENTED

01A7 C7 06 0078 R 0227 R        MOV     WORD PTR DISKETTE_PARM,OFFSET DISKETTE_TBL
01AD 8C 0E 007A R               MOV     WORD PTR DISKETTE_PARM+2,CS
01B1                    H0:
01B1 FB                         STI

                        ;-----  ATTEMPT BOOTSTRAP FROM DISKETTE

01B2 2B D2                      SUB     DX,DX                   ; DRIVE ZERO

                        ;-----  ESTABLISH ES:BX POINTER

01B4 8E C2                      MOV     ES,DX                   ; ESTABLISH SEGMENT
01B6 BB 7C00 R                  MOV     BX,OFFSET BOOT_LOCN     ; SET BX TO 7C00H

                        ;-----  CLEAR  BOOT_LOCN

01B9 FC                         CLD                             ; DIRECTION FORWARD
01BA 33 C0                      XOR     AX,AX
01BC B9 0100                    MOV     CX,256                  ; CLEAR 256 WORDS
01BF 8B FB                      MOV     DI,BX                   ; POINT TO BOOT LOCATION BUFFER
01C1 F3/ AB                     REP     STOSW                   ; ZERO THE BOOT LOCATION BUFFER

01C3 B9 0004                    MOV     CX,4                    ; SET RETRY COUNT
01C6                    H1:                                     ; IPL_SYSTEM
01C6 51                         PUSH    CX                      ; SAVE RETRY COUNT
01C7 2B C0                      SUB     AX,AX                   ; RESET THE DISKETTE
01C9 CD 13                      INT     13H                     ; FILE IO CALL
01CB 72 08                      JC      H2                      ; IF ERROR, TRY AGAIN

01CD B8 0201                    MOV     AX,0201H                ; READ IN THE SINGLE SECTOR
01D0 B9 0001                    MOV     CX,1                    ; SECTOR 1, TRACK 0
01D3 CD 13                      INT     13H                     ; FILE IO CALL
01D5 59                 H2:     POP     CX                      ; RECOVER RETRY COUNT
01D6 73 09                      JNC     H3                      ; CARRY FLAG SET BY UNSUCCESSFUL READ

01D8 80 FC 80                   CMP     AH,80H                  ; IF TIME OUT, NO RETRY
01DB 74 22                      JZ      H6                      ; TRY FIXED DISK

01DD E2 E7                      LOOP    H1                      ; DO IT FOR RETRY TIMES
01DF EB 1E                      JMP     SHORT H6                ; UNABLE TO IPL FROM THE DISKETTE

01E1 80 3E 7C00 R 06    H3:     CMP     BYTE PTR BOOT_LOCN,06H  ; CHECK FOR FIRST INSTRUCTION INVALID
01E6 72 3D                      JB      H10                     ; IF BOOT NOT VALID, GO TO BASIC

                        ;-----  INSURE DATA PATTERN FIRST 8 WORDS NOT ALL EQUAL

01E8 BF 7C00 R                  MOV     DI,OFFSET BOOT_LOCN     ; CHECK DATA PATTERN
01EB B9 0008                    MOV     CX,8                    ; CHECK THE NEXT 8 WORDS
01EE A1 7C00 R                  MOV     AX,WORD PTR BOOT_LOCN   ; LOAD THE FIRST WORD

01F1 83 C7 02           H4:     ADD     DI,2                    ; POINT TO NEXT WORD
01F4 3B 05                      CMP     AX,[DI]                 ; CHECK DATA PATTERN FOR A FILL PATTERN
01F6 E1 F9                      LOOPZ   H4                      ;
01F8 74 2B                      JZ      H10                     ; BOOT NOT VALID, GO TO BASIC
01FA                    H5:
01FA EA 7C00 ---- R             JMP     BOOT_LOCN

                        ;-----  ATTEMPT BOOTSTRAP FROM FIXED DISK

01FF                    H6:
01FF 2B C0                      SUB     AX,AX                   ; RESET DISKETTE
0201 CD 13                      INT     13H
0203 B9 0003                    MOV     CX,3                    ; SET RETRY COUNT
0206 BA 0080                    MOV     DX,0080H                ; FIXED DISK ZERO
0209                    H7:                                     ; IPL_SYSTEM
0209 51                         PUSH    CX                      ; SAVE RETRY COUNT
020A 2B C0                      SUB     AX,AX                   ; RESET THE FIXED DISK
020C CD 13                      INT     13H                     ; FILE IO CALL
020E 72 08                      JC      H8                      ; IF ERROR, TRY AGAIN

                        ;-----  ES AND BX ALREADY ESTABLISHED

0210 B8 0201                    MOV     AX,0201H                ; READ IN THE SINGLE SECTOR
0213 B9 0001                    MOV     CX,1                    ; SECTOR 1, TRACK 0
0216 CD 13                      INT     13H                     ; FILE IO CALL
0218 59                 H8:     POP     CX                      ; RECOVER RETRY COUNT
0219 72 08                      JC      H9
```

```
021B A1 7DFE R                  MOV     AX,WORD PTR BOOT_LOCN+510D
021E 3D AA55                    CMP     AX,0AA55H               ; TEST FOR GENERIC BOOT BLOCK
0221 74 D7                      JZ      H5                      ; GO TO BOOT LOCATION
0223                    H9:
0223 E2 E4                      LOOP    H7                      ; DO IT FOR RETRY TIMES

                        ;-----  UNABLE TO IPL FROM THE DISKETTE OR FIXED DISK

0225                    H10:
0225 CD 18                      INT     18H                     ; RESIDENT BASIC

0227                    DISKETTE_TBL:

0227 CF                         DB      11001111B               ; SRT=D, HD UNLOAD=0F - 1ST SPEC BYTE
0228 02                         DB      2                       ; HD LOAD=1, MODE=DMA - 2ND SPEC BYTE
0229 25                         DB      25H                     ; MOTOR TIMEOUT AFTER OPERATION
022A 02                         DB      2                       ; 512 BYTES PER SECTOR
022B 08                         DB      8                       ; EOT (LAST SECTOR ON TRACK)
022C 2A                         DB      02AH                    ; GAP LENGTH
022D FF                         DB      0FFH                    ; DTL
022E 50                         DB      050H                    ; GAP LENGTH FOR FORMAT
022F F6                         DB      0F6H                    ; FILL BYTE FOR FORMAT
0230 19                         DB      25                      ; HEAD SETTLE TIME (MILLISECONDS)
0231 04                         DB      4                       ; MOTOR START TIME (1/8 SECOND)

                        ;-----  MAKE SURE THAT ALL HOUSEKEEPING IS DONE BEFORE EXIT

0232                    DSBL    PROC    NEAR
0232 2A C0                      SUB     AL,AL                   ; RESET INT/DMA MASK
0234 BA 0323                    MOV     DX,HF_PORT+3            ; LOAD FOR PORT_ADDRESS 3
0237 FA                         CLI                             ; DISABLE INTERRUPTS
0238 EE                         OUT     DX,AL                   ; RESET INT/DMA MASK CARD 0
0239 83 C2 04                   ADD     DX,4
023C EE                         OUT     DX,AL                   ; RESET INT/DMA MASK CARD 1
023D 83 C2 04                   ADD     DX,4
0240 EE                         OUT     DX,AL                   ; RESET INT/DMA MASK CARD 2
0241 83 C2 04                   ADD     DX,4
0244 EE                         OUT     DX,AL                   ; RESET INT/DMA MASK CARD 3

0245 B0 07                      MOV     AL,07H
0247 E6 0A                      OUT     DMA+10,AL               ; SET DMA MODE TO DISABLE
0249 E4 21                      IN      AL,INTA01
024B 0C 20                      OR      AL,020H
024D E6 21                      OUT     INTA01,AL               ; DISABLE IREQ 5
024F FB                         STI                             ; ENABLE INTERRUPTS
0250 C3                         RET
0251                    DSBL    ENDP

                        ;---  DISK_IO  ----------------------------------------
                        ;                                                     :
                        ;                 FIXED DISK BIOS ENTRY POINT         :
                        ;                                                     :
                        ;-------------------------------------------------------

0251                    DISK_IO PROC    FAR
                                ASSUME  DS:DATA,ES:NOTHING
0251 80 FA 80                   CMP     DL,080H                 ; TEST FOR FIXED DISK DRIVE
0254 73 05                      JAE     HARD_DISK               ; YES, HANDLE HERE
0256 CD 40                      INT     40H                     ; DISKETTE HANDLER
0258                    RET_2:
0258 CA 0002                    RET     2                       ; BACK TO CALLER

025B                    HARD_DISK:
025B FB                         STI                             ; ENABLE INTERRUPTS
025C 0A E4                      OR      AH,AH
025E 75 09                      JNZ     A3
0260 CD 40                      INT     40H                     ; RESET NEC WHEN AH=0
0262 2A E4                      SUB     AH,AH
0264 80 FA 81                   CMP     DL,(80H+S_MAX_FILE-1)   ; DL IN LIMITS?
0267 77 EF                      JA      RET_2
0269                    A3:
0269 80 FC 08                   CMP     AH,8                    ; GET PARAMETERS IS A SPECIAL CASE
026C 75 03                      JNZ     A2
026E E9 0380 R                  JMP     GET_PARM_N
0271                    A2:
0271 55                         PUSH    BP                      ; SAVE THE BASE POINTER
0272 8B EC                      MOV     BP,SP                   ; LOAD THE CMD_BLOCK POINTER
0274 83 EC 08                   SUB     SP,8                    ; ALLOCATE SPACE FOR THE COMMAND BLOCK
                                                                ;  ON THE STACK.
0277 53                         PUSH    BX                      ; SAVE REGISTERS DURING OPERATION
0278 51                         PUSH    CX
0279 52                         PUSH    DX
027A 1E                         PUSH    DS
027B 06                         PUSH    ES
027C 56                         PUSH    SI
027D 57                         PUSH    DI
027E BE ---- R                  MOV     SI,DATA
0281 8E DE                      MOV     DS,SI                   ; ESTABLISH DATA SEGMENT

0283 E8 02D0 R                  CALL    DISK_IO_CONT            ; PERFORM THE OPERATION

0286 50                         PUSH    AX
0287 E8 0232 R                  CALL    DSBL                    ; BE SURE DISABLES OCCURRED
028A B8 ---- R                  MOV     AX,DATA
028D 8E D8                      MOV     DS,AX                   ; ESTABLISH SEGMENT
028F 58                         POP     AX                      ; RESTORE THE REGISTERS
0290 8A 26 0074 R               MOV     AH,DISK_STATUS          ; GET STATUS FROM OPERATION
0294 5F                         POP     DI
0295 5E                         POP     SI
0296 07                         POP     ES
0297 1F                         POP     DS
0298 5A                         POP     DX
0299 59                         POP     CX
029A 5B                         POP     BX

029B 83 C4 08                   ADD     SP,8                    ; ADJUST FOR THE COMMAND BLOCK.
029E 5D                         POP     BP                      ; RESTORE BASE POINTER
029F 80 FC 01                   CMP     AH,1                    ; SET THE CARRY FLAG TO INDICATE
02A2 F5                         CMC                             ;  SUCCESS OR FAILURE
02A3 CA 0002                    RET     2                       ; THROW AWAY SAVED FLAGS
02A6                    DISK_IO ENDP
```

```
                        PAGE
02A6                    M1      LABEL   WORD                    ; FUNCTION TRANSFER TABLE
02A6 032E R                     DW      DISK_RESET              ; 000H
02A8 0347 R                     DW      RETURN_STATUS           ; 001H
02AA 0350 R                     DW      DISK_READ               ; 002H
02AC 0359 R                     DW      DISK_WRITE              ; 003H
02AE 0362 R                     DW      DISK_VERF               ; 004H
02B0 0369 R                     DW      FMT_TRK                 ; 005H
02B2 036F R                     DW      FMT_BAD                 ; 006H
02B4 0375 R                     DW      FMT_DRV                 ; 007H
02B6 0326 R                     DW      BAD_COMMAND             ; 008H
02B8 043F R                     DW      INIT_DRV                ; 009H
02BA 04F4 R                     DW      RD_LONG                 ; 00AH
02BC 0501 R                     DW      WR_LONG                 ; 00BH
02BE 0515 R                     DW      DISK_SEEK               ; 00CH
02C0 032E R                     DW      DISK_RESET              ; 00DH
02C2 051B R                     DW      RD_BUFF                 ; 00EH
02C4 0527 R                     DW      WR_BUFF                 ; 00FH
02C6 0533 R                     DW      TST_RDY                 ; 010H
02C8 0539 R                     DW      HDISK_RECAL             ; 011H
02CA 053F R                     DW      RAM_DIAG                ; 012H
02CC 0545 R                     DW      CHK_DRV                 ; 013H
02CE 054B R                     DW      CNTLR_DIAG              ; 014H
= 002A                  MIL     EQU     $-M1
                        DISK_IO_CONT    PROC    NEAR
02D0 80 FC 01                   CMP     AH,01H                  ; RETURN STATUS
02D3 74 72                      JE      RETURN_STATUS
02D5 80 EA 80                   SUB     DL,080H                 ; CONVERT DRIVE NUMBER TO 0 BASED RANGE
02D8 80 FA 08                   CMP     DL,MAX_FILE             ; LEGAL DRIVE TEST
02DB 73 49                      JAE     BAD_COMMAND
02DD C6 06 0074 R 00            MOV     DISK_STATUS,0           ; RESET THE STATUS INDICATOR
                        ;-----  SET UP COMMAND BLOCK
02E2 FE C9                      DEC     CL                      ; SECTORS 0-16 FOR CONTROLLER
02E4 C6 46 F8 00                MOV     CMD_BLOCK+0,0           ; SET TO ZERO THE OP CODE
02E8 88 4E FA                   MOV     CMD_BLOCK+2,CL          ; SECTOR AND HIGH 2 BITS CYLINDER
02EB 88 6E FB                   MOV     CMD_BLOCK+3,CH          ; CYLINDER LOW
02EE 88 46 FC                   MOV     CMD_BLOCK+4,AL          ; INTERLEAVE / BLOCK COUNT
02F1 A0 0076 R                  MOV     AL,CONTROL_BYTE         ; CONTROL BYTE (STEP OPTION)
02F4 88 46 FD                   MOV     CMD_BLOCK+5,AL          ; SET THE CONTROL FIELD
                        ;-----  CALCULATE THE PORT OFFSET
02F7 8A EA                      MOV     CH,DL                   ; SAVE DL
02F9 80 CA 01                   OR      DL,1
02FC FE CA                      DEC     DL
02FE D0 E2                      SHL     DL,1                    ; GENERATE OFFSET
0300 88 16 0077 R               MOV     PORT_OFF,DL             ; STORE OFFSET
0304 8A D5                      MOV     DL,CH                   ; RESTORE DL
0306 80 E2 01                   AND     DL,1                    ; MAKE DRIVE 0 OR 1
0309 B1 05                      MOV     CL,5                    ; SHIFT COUNT
030B D2 E2                      SHL     DL,CL                   ; DRIVE NUMBER (0,1)
030D 0A D6                      OR      DL,DH                   ; HEAD NUMBER
030F 88 56 F9                   MOV     CMD_BLOCK+1,DL          ; SET THE DRIVE AND HEAD
0312 8B C8                      MOV     CX,AX                   ; CALCULATE JUMP ADDRESS
0314 8A CD                      MOV     CL,CH                   ; GET INTO LOW BYTE
0316 32 ED                      XOR     CH,CH                   ; ZERO HIGH BYTE
0318 D1 E1                      SAL     CX,1                    ; *2 FOR TABLE LOOKUP
031A 8B F1                      MOV     SI,CX                   ; PUT INTO SI FOR BRANCH
031C 83 F9 2A                   CMP     CX,M1L                  ; TEST WITHIN RANGE
031F 73 05                      JNB     BAD_COMMAND
0321 2E: FF A4 02A6 R           JMP     WORD PTR CS:[SI+OFFSET M1]      ; GO DO THE COMMAND
0326                    BAD_COMMAND:
0326 C6 06 0074 R 01            MOV     DISK_STATUS,BAD_CMD     ; SET BAD COMMAND ERROR
032B B0 00                      MOV     AL,0
032D C3                         RET                             ; EXIT
032E                    DISK_IO_CONT    ENDP

                        ;--------------------------------------------------------
                        ;       RESET THE DISK SYSTEM                   (AH = 000H) :
                        ;--------------------------------------------------------

032E                    DISK_RESET      PROC    NEAR
032E E8 076D R                  CALL    PORT_0                  ; RESET PORT
0331 42                         INC     DX                      ; PORT 1 ADDRESS
0332 EE                         OUT     DX,AL                   ; RESET CARD
0333 EB 00                      JMP     $+2                     ; I/O DELAY AT LEAST +5us
0335 EB 00                      JMP     $+2                     ; ALLOW TIME TO CLEAR THE
0337 EB 00                      JMP     $+2                     ;  HARDWARE STATUS REGISTER
0339 EC                         IN      AL,DX                   ; READ THE HARDWARE STATUS
033A 24 3F                      AND     AL,00111111B            ; MASK OFF UPPER 2 BITS AND CLEAR CY
033C 74 06                      JZ      DR1                     ; EXIT IF REGISTER IS CLEARED WITH CY=0
033E C6 06 0074 R 05            MOV     DISK_STATUS,BAD_RESET   ; SET THE ERROR CONDITION
0343 C3                         RET                             ; EXIT
0344                    DR1:
0344 E9 043F R                  JMP     INIT_DRV                ; SET THE DRIVE PARAMETERS
0347                    DISK_RESET      ENDP

                        ;--------------------------------------------------------
                        ;       DISK STATUS ROUTINE                     (AH = 001H) :
                        ;--------------------------------------------------------

0347                    RETURN_STATUS   PROC    NEAR
0347 A0 0074 R                  MOV     AL,DISK_STATUS          ; OBTAIN PREVIOUS STATUS
034A C6 06 0074 R 00            MOV     DISK_STATUS,0           ; RESET STATUS
034F C3                         RET
0350                    RETURN_STATUS   ENDP

                        ;--------------------------------------------------------
                        ;       DISK READ ROUTINE                       (AH = 002H) :
                        ;--------------------------------------------------------

0350                    DISK_READ       PROC    NEAR
0350 B0 47                      MOV     AL,DMA_READ             ; MODE BYTE FOR DMA READ
0352 C6 46 F8 08                MOV     CMD_BLOCK+0,READ_CMD
0356 E9 055E R                  JMP     DMA_OPN
0359                    DISK_READ       ENDP
```

```
                        ;--------------------------------------------------------
                        ;       DISK WRITE ROUTINE                      (AH = 003H) :
                        ;--------------------------------------------------------

0359                    DISK_WRITE      PROC    NEAR
0359 B0 4B                      MOV     AL,DMA_WRITE            ; MODE BYTE FOR DMA WRITE
035B C6 46 F8 0A                MOV     CMD_BLOCK+0,WRITE_CMD
035F E9 055E R                  JMP     DMA_OPN
0362                    DISK_WRITE      ENDP

                        ;--------------------------------------------------------
                        ;       DISK VERIFY                             (AH = 004H) :
                        ;--------------------------------------------------------

0362                    DISK_VERF       PROC    NEAR
0362 C6 46 F8 05                MOV     CMD_BLOCK+0,CHK_TRK_CMD
0366 E9 054F R                  JMP     NDMA_OPN
0369                    DISK_VERF       ENDP

                        ;--------------------------------------------------------
                        ;       FORMATTING                      (AH = 005H 006H 007H) :
                        ;--------------------------------------------------------

0369                    FMT_TRK PROC    NEAR                    ; FORMAT TRACK  (AH = 005H)
0369 C6 46 F8 06                MOV     CMD_BLOCK+0,FMTTRK_CMD
036D EB 0A                      JMP     SHORT   FMT_CONT
036F                    FMT_TRK ENDP

036F                    FMT_BAD PROC    NEAR                    ; FORMAT BAD TRACK  (AH = 006H)
036F C6 46 F8 07                MOV     CMD_BLOCK+0,FMTBAD_CMD
0373 EB 04                      JMP     SHORT   FMT_CONT
0375                    FMT_BAD ENDP

0375                    FMT_DRV PROC    NEAR                    ; FORMAT DRIVE  (AH = 007H)
0375 C6 46 F8 04                MOV     CMD_BLOCK+0,FMTDRV_CMD
0379                    FMT_DRV ENDP

0379                    FMT_CONT:
0379 80 66 FA C0                AND     CMD_BLOCK+2,11000000B   ; ZERO OUT SECTOR FIELD
037D E9 054F R                  JMP     NDMA_OPN

                        ;--------------------------------------------------------
                        ;       GET PARAMETERS                                  AH = 8) :
                        ;--------------------------------------------------------

0380                    GET_PARM_N      LABEL   NEAR
0380                    GET_PARM        PROC    FAR             ; GET DRIVE PARAMETERS
0380 1E                         PUSH    DS                      ; SAVE REGISTERS
0381 06                         PUSH    ES
0382 53                         PUSH    BX

                                ASSUME  DS:ABS0
0383 2B C0                      SUB     AX,AX                   ; ESTABLISH ADDRESSING
0385 8E D8                      MOV     DS,AX
0387 C4 1E 0104 R               LES     BX,HF_TBL_VEC

                                ASSUME  DS:DATA
038B B8 ---- R                  MOV     AX,DATA
038E 8E D8                      MOV     DS,AX                   ; ESTABLISH SEGMENT
0390 80 EA 80                   SUB     DL,80H
0393 80 FA 08                   CMP     DL,MAX_FILE             ; TEST WITHIN RANGE
0396 73 57                      JAE     G4
0398 C6 06 0074 R 00            MOV     DISK_STATUS,0           ; RESET THE STATUS INDICATOR
039D 8A EA                      MOV     CH,DL                   ; SAVE THE DRIVE
039F 80 CA 01                   OR      DL,1
03A2 FE CA                      DEC     DL
03A4 D0 E2                      SHL     DL,1                    ; GENERATE OFFSET
03A6 88 16 0077 R               MOV     PORT_OFF,DL             ; STORE OFFSET
03AA 8A D5                      MOV     DL,CH                   ; RESTORE DL
03AC 80 E2 01                   AND     DL,00000001B            ; DRIVE 0 OR DRIVE 1
03AF 8A E2                      MOV     AH,DL
03B1 E8 076D R                  CALL    PORT_0
03B4 42                         INC     DX                      ; PORT_2 ADDRESS
03B5 42                         INC     DX
03B6 EC                         IN      AL,DX                   ; READ SWITCH SETTINGS
03B7 80 FC 00                   CMP     AH,0                    ; DRIVE 0 OR 1
03BA 75 04                      JNZ     G0
03BC D0 E8                      SHR     AL,1                    ; RIGHT JUSTIFY THE SWITCH BITS
03BE D0 E8                      SHR     AL,1
03C0                    G0:
03C0 24 03                      AND     AL,00000011B            ; ISOLATE THE TABLE BITS
03C2 B1 04                      MOV     CL,4                    ; TABLE LENGTH IS 16 BYTES
03C4 D2 E0                      SHL     AL,CL                   ; ADJUST
03C6 2A E4                      SUB     AH,AH
03C8 03 D8                      ADD     BX,AX
03CA 26: 8B 07                  MOV     AX,ES:[BX]              ; MAX NUMBER OF CYLINDERS
03CD 2D 0002                    SUB     AX,2                    ; ADJUST FOR 0-N
                                                                ; AND RESERVE LAST TRACK
03D0 8A E8                      MOV     CH,AL
03D2 25 0300                    AND     AX,0300H                ; HIGH TWO BITS OF CYLINDER
03D5 D1 E8                      SHR     AX,1
03D7 D1 E8                      SHR     AX,1
03D9 0C 11                      OR      AL,011H                 ; SECTORS
03DB 8A C8                      MOV     CL,AL
03DD 26: 8A 77 02               MOV     DH,ES:[BX][2]           ; HEADS
03E1 FE CE                      DEC     DH                      ; 0-N RANGE
03E3 8A 16 0075 R               MOV     DL,HF_NUM               ; DRIVE COUNT
03E7 2B C0                      SUB     AX,AX
03E9                    G5:
03E9 5B                         POP     BX                      ; RESTORE REGISTERS
03EA 07                         POP     ES
03EB 1F                         POP     DS
03EC CA 0002                    RET     2                       ; EXIT
03EF                    G4:
03EF C6 06 0074 R 07            MOV     DISK_STATUS,INIT_FAIL   ; OPERATION FAILED
03F4 B4 07                      MOV     AH,INIT_FAIL
03F6 2A C0                      SUB     AL,AL
03F8 2B D2                      SUB     DX,DX
03FA 2B C9                      SUB     CX,CX
03FC F9                         STC                             ; SET ERROR FLAG
03FD EB EA                      JMP     G5                      ; EXIT
03FF                    GET_PARM        ENDP
```

```
                        PAGE
                        ;--------------------------------------------------------
                        ; INITIALIZE DRIVE CHARACTERISTICS                      :
                        ;                                                      :
                        ; FIXED DISK PARAMETER TABLE                           :
                        ;                                                      :
                        ;  -  THE TABLE IS COMPOSED OF A BLOCK DEFINED AS:    :
                        ;                                                      :
                        ;       (1 WORD) - MAXIMUM NUMBER OF CYLINDERS         :
                        ;       (1 BYTE) - MAXIMUM NUMBER OF HEADS             :
                        ;       (1 WORD) - STARTING REDUCED WRITE CURRENT CYL  :
                        ;       (1 WORD) - STARTING WRITE PRECOMPENSATION CYL  :
                        ;       (1 BYTE) - MAXIMUM ECC DATA BURST LENGTH       :
                        ;       (1 BYTE) - CONTROL BYTE (DRIVE STEP OPTION)    :
                        ;                  BIT    7 DISABLE DISK-ACCESS RETRIES :
                        ;                  BIT    6 DISABLE ECC RETRIES         :
                        ;                  BITS 5-3 ZERO                        :
                        ;                  BITS 2-0 DRIVE OPTION                :
                        ;       (1 BYTE) - STANDARD TIME OUT VALUE (SEE BELOW) :
                        ;       (1 BYTE) - TIME OUT VALUE FOR FORMAT DRIVE     :
                        ;       (1 BYTE) - TIME OUT VALUE FOR CHECK DRIVE      :
                        ;       (1 WORD) - LANDING ZONE                        :
                        ;       (1 BYTE) - SECTORS/TRACK                       :
                        ;       (1 BYTE) - RESERVED FOR FUTURE USE             :
                        ;                                                      :
                        ;        - TO DYNAMICALLY DEFINE A SET OF PARAMETERS   :
                        ;          BUILD A TABLE OF VALUES AND PLACE THE       :
                        ;          CORRESPONDING VECTOR INTO INTERRUPT 41.      :
                        ;                                                      :
                        ;       NOTE:   THE DEFAULT TABLE IS VECTORED IN FOR   :
                        ;               AN INTERRUPT 19H (BOOTSTRAP)           :
                        ;                                                      :
                        ; ON THE CARD SWITCH SETTINGS                          :
                        ;                                                      :
                        ;                  DRIVE 0    DRIVE 1                  :
                        ;                 -------------------------            :
                        ;         ON      :          /          :             :
                        ;                 : -1-  -2- / -3-  -4- :             :
                        ;         OFF     :          /          :             :
                        ;                 -------------------------            :
                        ;                                                      :
                        ;       TRANSLATION TABLE                              :
                        ;                                                      :
                        ;       DRIVE 0  :  DRIVE 1  :  TABLE ENTRY            :
                        ;         1/2    :    3/4     :                       :
                        ;       ------------------------------------           :
                        ;        ON ON   :   ON ON    :       0                 :
                        ;        ON OFF  :   ON OFF   :       1                 :
                        ;        OFF ON  :   OFF ON   :       2                 :
                        ;        OFF OFF :   OFF OFF  :       3                 :
                        ;                                                      :
                        ;--------------------------------------------------------

03FF                    FD_TBL:

                        ;-----  DRIVE TABLE 0

03FF 0132                       DW      0306D                   ; MAX CYLINDERS
0401 04                         DB      04D                     ; MAX HEADS
0402 0132                       DW      0306D                   ; START REDUCED WRITE CURRENT CYL
0404 0000                       DW      0                       ; START WRITE PRECOMPENSATION CYL
0406 0B                         DB      0BH                     ; MAX ECC BURST DATA LENGTH
0407 05                         DB      00000101B               ; CONTROL BYTE
0408 10                         DB      010H                    ; STANDARD TIME OUT
0409 C0                         DB      0C0H                    ; TIME OUT FOR FORMAT DRIVE
040A 28                         DB      028H                    ; TIME FOR CHECK DRIVE
040B 0132                       DW      0306D                   ; LANDING ZONE
040D 11                         DB      017D                    ; SECTORS/TRACK
040E 00                         DB      0                       ; RESERVED

                        ;-----  DRIVE TABLE 1

040F 0264                       DW      0612D                   ; MAX CYLINDERS
0411 04                         DB      04D                     ; MAX HEADS
0412 0264                       DW      0612D                   ; START REDUCED WRITE CURRENT CYL
0414 0000                       DW      0                       ; START WRITE PRECOMPENSATION CYL
0416 0B                         DB      0BH                     ; MAX ECC BURST DATA LENGTH
0417 05                         DB      00000101B               ; CONTROL BYTE
0418 28                         DB      028H                    ; STANDARD TIME OUT
0419 E0                         DB      0E0H                    ; TIME OUT FOR FORMAT DRIVE
041A 42                         DB      042H                    ; TIME FOR CHECK DRIVE
041B 0297                       DW      0663D                   ; LANDING ZONE
041D 11                         DB      017D                    ; SECTORS/TRACK
041E 00                         DB      0                       ; RESERVED

                        ;-----  DRIVE TABLE 2

041F 0267                       DW      0615D                   ; MAX CYLINDERS
0421 04                         DB      04D                     ; MAX HEADS
0422 0267                       DW      0615D                   ; START REDUCED WRITE CURRENT CYL
0424 012C                       DW      0300D                   ; START WRITE PRECOMPENSATION CYL
0426 0B                         DB      0BH                     ; MAX ECC BURST DATA LENGTH
0427 05                         DB      00000101B               ; CONTROL BYTE
0428 28                         DB      028H                    ; STANDARD TIME OUT
0429 E0                         DB      0E0H                    ; TIME OUT FOR FORMAT DRIVE
042A 42                         DB      042H                    ; TIME FOR CHECK DRIVE
042B 0267                       DW      0615D                   ; LANDING ZONE
042D 11                         DB      017D                    ; SECTORS/TRACK
042E 00                         DB      0                       ; RESERVED

                        ;-----  DRIVE TABLE 3

042F 0132                       DW      0306D                   ; MAX CYLINDERS
0431 08                         DB      08D                     ; MAX HEADS
0432 0132                       DW      0306D                   ; START REDUCED WRITE CURRENT CYL
0434 0080                       DW      0128D                   ; START WRITE PRECOMPENSATION CYL
0436 0B                         DB      0BH                     ; MAX ECC BURST DATA LENGTH
0437 05                         DB      00000101B               ; CONTROL BYTE
0438 28                         DB      028H                    ; STANDARD TIME OUT
0439 E0                         DB      0E0H                    ; TIME OUT FOR FORMAT DRIVE
043A 42                         DB      042H                    ; TIME FOR CHECK DRIVE
043B 0150                       DW      0336D                   ; LANDING ZONE
043D 11                         DB      017D                    ; SECTORS/TRACK
043E 00                         DB      0                       ; RESERVED
```

```
                        ;--------------------------------------------------------
                        ;       INITIALIZE DRIVE                        (AH = 09H) :
                        ;--------------------------------------------------------

043F                    INIT_DRV        PROC    NEAR

                        ;-----  DO DRIVE ZERO

043F C6 46 F8 0C                MOV     CMD_BLOCK+0,INIT_DRV_CMD
0443 C6 46 F9 00                MOV     CMD_BLOCK+1,0           ; SET FOR DRIVE 0
0447 E8 0458 R                  CALL    INIT_DRV_R              ; SEND THE PARAMETERS
044A 72 0B                      JC      INIT_DRV_OUT            ; ERROR?

                        ;-----  DO DRIVE ONE

044C C6 46 F8 0C                MOV     CMD_BLOCK+0,INIT_DRV_CMD
0450 C6 46 F9 20                MOV     CMD_BLOCK+1,00100000B   ; SET TO DRIVE 1
0454 E8 0458 R                  CALL    INIT_DRV_R              ; SEND THE PARAMETERS
0457                    INIT_DRV_OUT:
0457 C3                         RET                             ; EXIT
0458                    INIT_DRV        ENDP

0458                    INIT_DRV_R      PROC    NEAR
0458 2A C0                      SUB     AL,AL
045A E8 057C R                  CALL    COMMAND                 ; ISSUE THE COMMAND
045D 73 01                      JNC     B1                      ;  DX = PORT 0 AFTER CALL
045F C3                         RET
0460                    B1:
0460 8C D9                      MOV     CX,DS                   ; SAVE SEGMENT

                                ASSUME  DS:ABS0
0462 2B C0                      SUB     AX,AX
0464 8E D8                      MOV     DS,AX                   ; ESTABLISH SEGMENT
0466 C4 1E 0104 R               LES     BX,HF_TBL_VEC           ; LOAD THE TABLE VECTOR
046A 8E D9                      MOV     DS,CX                   ; RESTORE SEGMENT

                                ASSUME  DS:DATA
                        ;-------------------------------------------
                        ;       DETERMINE PARAMETER TABLE OFFSET    :
                        ;       USING CONTROLLER PORT TWO AND       :
                        ;       DRIVE NUMBER SPECIFIER (0-1)        :
                        ;-------------------------------------------
046C 42                         INC     DX
046D 42                         INC     DX                      ; ADDRESS PORT 2
046E EC                         IN      AL,DX                   ; READ THE SWITCH SETTINGS
046F 8A 66 F9                   MOV     AH,CMD_BLOCK+1
0472 80 E4 20                   AND     AH,00100000B            ; DRIVE 0 OR 1
0475 75 04                      JNZ     B2
0477 D0 E8                      SHR     AL,1                    ; ADJUST
0479 D0 E8                      SHR     AL,1
047B                    B2:
047B 24 03                      AND     AL,011B                 ; ISOLATE
047D B1 04                      MOV     CL,4
047F D2 E0                      SHL     AL,CL                   ; ADJUST
0481 2A E4                      SUB     AH,AH
0483 03 D8                      ADD     BX,AX
0485 B4 09                      MOV     AH,00001001B            ; SET MASK FOR DATA MODE CPU TO CARD

                        ;-----  SEND DRIVE PARAMETERS MOST SIGNIFICANT BYTE FIRST

0487 BF 0001                    MOV     DI,1                    ; SEND MSB OF MAX CYLINDER
048A E8 04E9 R                  CALL    INIT_DRV_S
048D 72 4C                      JC      B3

048F BF 0000                    MOV     DI,0                    ; SEND LSB OF MAX CYLINDER
0492 E8 04E9 R                  CALL    INIT_DRV_S
0495 72 44                      JC      B3

0497 BF 0002                    MOV     DI,2                    ; SEND THE MAXIMUM HEADS
049A E8 04E9 R                  CALL    INIT_DRV_S
049D 72 3C                      JC      B3

049F BF 0004                    MOV     DI,4                    ; SEND MSB OF REDUCE WRITE CURRENT
04A2 E8 04E9 R                  CALL    INIT_DRV_S              ;  CYLINDER
04A5 72 34                      JC      B3

04A7 BF 0003                    MOV     DI,3                    ; SEND LSB OF REDUCE WRITE CURRENT
04AA E8 04E9 R                  CALL    INIT_DRV_S              ;  CYLINDER
04AD 72 2C                      JC      B3

04AF BF 0006                    MOV     DI,6                    ; SEND MSB OF WRITE PRECOMP CYLINDER
04B2 E8 04E9 R                  CALL    INIT_DRV_S
04B5 72 24                      JC      B3

04B7 BF 0005                    MOV     DI,5                    ; SEND LSB OF WRITE PRECOMP CYLINDER
04BA E8 04E9 R                  CALL    INIT_DRV_S
04BD 72 1C                      JC      B3

04BF BF 0007                    MOV     DI,7                    ; SEND ECC BURST LENGTH
04C2 E8 04E9 R                  CALL    INIT_DRV_S
04C5 72 14                      JC      B3

04C7 BF 0008                    MOV     DI,8                    ; LOAD THE CONTROL BYTE AND PLACE IN
04CA 26: 8A 01                  MOV     AL,ES:[BX+DI]           ;  MEMORY AT 40:76H
04CD A2 0076 R                  MOV     CONTROL_BYTE,AL

04D0 2B C9                      SUB     CX,CX
04D2 B4 0F                      MOV     AH,00001111B            ; SET THE MASK FOR STATUS MODE
04D4                    B5:
04D4 E8 068D R                  CALL    HD_WAIT                 ; GO WAIT FOR THE STATE TO HAPPEN
04D7 73 09                      JNC     B6                      ; JMP TO READ THE STATUS BYTE
04D9 E2 F9                      LOOP    B5                      ; TRY AGAIN
04DB                    B3:
04DB C6 06 0074 R 07            MOV     DISK_STATUS,INIT_FAIL   ; OPERATION FAILED
04E0 F9                         STC                             ; SET THE ERROR CONDITION
04E1 C3                         RET
04E2                    B6:
04E2 4A                         DEC     DX                      ; ADDRESS PORT 0
04E3 EC                         IN      AL,DX                   ; READ STATUS BYTE OF THE OPERATION
04E4 24 02                      AND     AL,2                    ; MASK ERROR BIT
04E6 75 F3                      JNZ     B3                      ; ERROR BIT SET?
04E8 C3                         RET
04E9                    INIT_DRV_R      ENDP
```

```
                        ;-----  SEND THE BYTE OUT TO THE CONTROLLER

04E9                    INIT_DRV_S      PROC    NEAR
04E9 E8 068D R                  CALL    HD_WAIT                 ; GO WAIT FOR REQUEST
04EC 72 05                      JC      D1                      ;  AFTER CALL DX = PORT 1
04EE 4A                         DEC     DX                      ; ADDRESS PORT 0
04EF 26: 8A 01                  MOV     AL,ES:[BX+DI]
04F2 EE                         OUT     DX,AL                   ; WRITE THE DATA TO THE CARD
04F3                    D1:
04F3 C3                         RET
04F4                    INIT_DRV_S      ENDP

                        ;--------------------------------------------------------
                        ;       READ LONG                               (AH = 0AH) :
                        ;--------------------------------------------------------

04F4                    RD_LONG         PROC    NEAR
04F4 E8 050E R                  CALL    CHK_LONG                ; CHECK LIMITS
04F7 72 5F                      JC      G8
04F9 C6 46 F8 E5                MOV     CMD_BLOCK+0,RD_LONG_CMD
04FD B0 47                      MOV     AL,DMA_READ
04FF EB 5D                      JMP     SHORT   DMA_OPN
0501                    RD_LONG         ENDP

                        ;--------------------------------------------------------
                        ;       WRITE LONG                              (AH = 0BH) :
                        ;--------------------------------------------------------

0501                    WR_LONG         PROC    NEAR
0501 E8 050E R                  CALL    CHK_LONG                ; CHECK LIMITS
0504 72 52                      JC      G8
0506 C6 46 F8 E6                MOV     CMD_BLOCK+0,WR_LONG_CMD
050A B0 4B                      MOV     AL,DMA_WRITE
050C EB 50                      JMP     SHORT   DMA_OPN
050E                    WR_LONG         ENDP

050E                    CHK_LONG        PROC    NEAR
050E 8A 46 FC                   MOV     AL,CMD_BLOCK+4          ; LOAD THE NUMBER OF SECTORS
0511 3C 80                      CMP     AL,080H                 ; COMPARE WITH LIMITS
0513 F5                         CMC                             ; SET THE CONDITION
0514 C3                         RET
0515                    CHK_LONG        ENDP

                        ;--------------------------------------------------------
                        ;       SEEK                                    (AH = 0CH) :
                        ;--------------------------------------------------------

0515                    DISK_SEEK       PROC    NEAR
0515 C6 46 F8 0B                MOV     CMD_BLOCK+0,SEEK_CMD
0519 EB 34                      JMP     SHORT   NDMA_OPN
051B                    DISK_SEEK       ENDP

                        ;--------------------------------------------------------
                        ;       READ SECTOR BUFFER                      (AH = 0EH) :
                        ;--------------------------------------------------------

051B                    RD_BUFF PROC    NEAR
051B C6 46 F8 0E                MOV     CMD_BLOCK+0,RD_BUFF_CMD
051F C6 46 FC 01                MOV     CMD_BLOCK+4,1           ; ONLY ONE BLOCK
0523 B0 47                      MOV     AL,DMA_READ
0525 EB 37                      JMP     SHORT   DMA_OPN
0527                    RD_BUFF ENDP

                        ;--------------------------------------------------------
                        ;       WRITE SECTOR BUFFER                     (AH = 0FH) :
                        ;--------------------------------------------------------

0527                    WR_BUFF PROC    NEAR
0527 C6 46 F8 0F                MOV     CMD_BLOCK+0,WR_BUFF_CMD
052B C6 46 FC 01                MOV     CMD_BLOCK+4,1           ; ONLY ONE BLOCK
052F B0 4B                      MOV     AL,DMA_WRITE
0531 EB 2B                      JMP     SHORT   DMA_OPN
0533                    WR_BUFF ENDP

                        ;--------------------------------------------------------
                        ;       TEST DISK READY                         (AH = 010H) :
                        ;--------------------------------------------------------

0533                    TST_RDY PROC    NEAR
0533 C6 46 F8 00                MOV     CMD_BLOCK+0,TST_RDY_CMD
0537 EB 16                      JMP     SHORT   NDMA_OPN
0539                    TST_RDY ENDP

                        ;--------------------------------------------------------
                        ;       RECALIBRATE                             (AH = 011H) :
                        ;--------------------------------------------------------

0539                    HDISK_RECAL     PROC    NEAR
0539 C6 46 F8 01                MOV     CMD_BLOCK+0,RECAL_CMD
053D EB 10                      JMP     SHORT   NDMA_OPN
053F                    HDISK_RECAL     ENDP
```

```
                        PAGE
                        ;--------------------------------------------------------
                        ;       CONTROLLER RAM DIAGNOSTICS              (AH = 012H) :
                        ;--------------------------------------------------------

053F                    RAM_DIAG        PROC    NEAR
053F C6 46 F8 E0                MOV     CMD_BLOCK+0,RAM_DIAG_CMD
0543 EB 0A                      JMP     SHORT   NDMA_OPN
0545                    RAM_DIAG        ENDP

                        ;--------------------------------------------------------
                        ;       DRIVE DIAGNOSTICS                       (AH = 013H) :
                        ;--------------------------------------------------------

0545                    CHK_DRV PROC    NEAR
0545 C6 46 F8 E3                MOV     CMD_BLOCK+0,CHK_DRV_CMD
0549 EB 04                      JMP     SHORT   NDMA_OPN
054B                    CHK_DRV ENDP

                        ;--------------------------------------------------------
                        ;       CONTROLLER INTERNAL DIAGNOSTICS         (AH = 014H) :
                        ;--------------------------------------------------------

054B                    CNTLR_DIAG      PROC    NEAR
054B C6 46 F8 E4                MOV     CMD_BLOCK+0,CNTLR_DIAG_CMD
054F                    CNTLR_DIAG      ENDP

                        ;--------------------------------------------------------
                        ;                       SUPPORT ROUTINES                :
                        ;--------------------------------------------------------

054F                    NDMA_OPN:
054F B0 02                      MOV     AL,02H
0551 E8 057C R                  CALL    COMMAND                 ; ISSUE THE COMMAND
0554 72 22                      JC      G11
0556 EB 16                      JMP     SHORT   G3
0558                    G8:
0558 C6 06 0074 R 09            MOV     DISK_STATUS,DMA_BOUNDARY
055D C3                         RET
055E                    DMA_OPN:
055E E8 06A5 R                  CALL    DMA_SETUP               ; SET UP FOR DMA OPERATION
0561 72 F5                      JC      G8
0563 B0 03                      MOV     AL,03H
0565 E8 057C R                  CALL    COMMAND                 ; ISSUE THE COMMAND
0568 72 0E                      JC      G11
056A B0 03                      MOV     AL,03H
056C E6 0A                      OUT     DMA+10,AL               ; INITIALIZE THE DISK CHANNEL
056E                    G3:
056E FA                         CLI                             ; NO INTERRUPTS
056F E4 21                      IN      AL,INTA01               ; READ THE MASK
0571 24 DF                      AND     AL,0DFH                 ; ENABLE IRQ-5
0573 E6 21                      OUT     INTA01,AL               ; WRITE THE MASK OUT
0575 E8 0700 R                  CALL    WAIT_INT                ; PROCEDURE DOES STI
0578                    G11:
0578 E8 05AD R                  CALL    ERROR_CHK               ; SEE IF THERE IS AN ERROR
057B C3                         RET                             ; EXIT

                        ;--------------------------------------------------------
                        ; COMMAND                                               :
                        ;       THIS ROUTINE OUTPUTS THE COMMAND BLOCK          :
                        ; INPUT                                                 :
                        ;       AL = CONTROLLER DMA/INTERRUPT REGISTER MASK      :
                        ;                                                       :
                        ;--------------------------------------------------------

057C                    COMMAND PROC    NEAR
057C E8 076D R                  CALL    PORT_0                  ; GET THE BASE ADDRESS
057F 42                         INC     DX
0580 42                         INC     DX                      ; ADDRESS PORT_2
0581 EE                         OUT     DX,AL                   ; ISSUE CONTROLLER SELECT PULSE
0582 42                         INC     DX                      ; ADDRESS PORT 3
0583 2B C9                      SUB     CX,CX                   ; WAIT COUNT
0585 EE                         OUT     DX,AL                   ; WRITE DMA MASK REGISTER
0586 4A                         DEC     DX
0587 4A                         DEC     DX                      ; ADDRESS PORT 1
0588                    WAIT_BUSY:
0588 EC                         IN      AL,DX                   ; READ THE HARDWARE STATUS
0589 24 0F                      AND     AL,0FH
058B 3C 0D                      CMP     AL,R1_BUSY OR R1_BUS OR R1_REQ  ; CHECK FOR BUSY,COMMAND
058D 74 09                      JE      C1                              ;  AND REQUEST BITS
058F E2 F7                      LOOP    WAIT_BUSY               ; KEEP TRYING
0591 C6 06 0074 R 80            MOV     DISK_STATUS,TIME_OUT
0596 F9                         STC                             ; SET THE ERROR CONDITION
0597 C3                         RET                             ; ERROR RETURN
0598                    C1:
0598 B9 0006                    MOV     CX,6                    ; SET FOR 6 BYTES OF COMMAND
059B 4A                         DEC     DX                      ; ADDRESS PORT 0
059C 8B F5                      MOV     SI,BP                   ; SAVE THE BASE POINTER
059E 83 ED 08                   SUB     BP,8                    ; SET FIRST BYTE OF COMMAND BLOCK
05A1 FA                         CLI                             ; NO INTERRUPTS IN COMMAND SEQUENCE
05A2                    CM3:
05A2 8A 46 00                   MOV     AL,[BP]                 ; GET A COMMAND BYTE
05A5 EE                         OUT     DX,AL                   ; ALLOW AT LEAST 2us BETWEEN EACH BYTE
05A6 45                         INC     BP                      ;  ON SENDING THE COMMAND SEQUENCE.
05A7 E2 F9                      LOOP    CM3                     ; DO MORE
05A9 8B EE                      MOV     BP,SI                   ; RESTORE THE BASE POINTER
05AB FB                         STI                             ; INTERRUPTS BACK ON
05AC C3                         RET
05AD                    COMMAND ENDP
```

```
                        PAGE
                        ;--------------------------------------------------------
                        ;                                                       :
                        ;                       SENSE STATUS BYTES              :
                        ;       BYTE 0                                          :
                        ;            BIT    7   ADDRESS VALID, WHEN SET         :
                        ;            BIT    6   SPARE, SET TO ZERO              :
                        ;            BITS 5-4   ERROR TYPE                      :
                        ;            BITS 3-0   ERROR CODE                      :
                        ;                                                       :
                        ;       BYTE 1                                          :
                        ;            BITS 7-6   ZERO                            :
                        ;            BIT    5   DRIVE (0-1)                     :
                        ;            BITS 4-0   HEAD NUMBER                     :
                        ;                                                       :
                        ;       BYTE 2                                          :
                        ;            BITS 7-5   CYLINDER HIGH                   :
                        ;            BITS 4-0   SECTOR NUMBER                   :
                        ;                                                       :
                        ;       BYTE 3                                          :
                        ;            BITS 7-0   CYLINDER LOW                    :
                        ;                                                       :
                        ;--------------------------------------------------------

05AD                    ERROR_CHK       PROC    NEAR
05AD A0 0074 R                  MOV     AL,DISK_STATUS          ; CHECK IF THERE WAS AN ERROR
05B0 0A C0                      OR      AL,AL                   ; ANYTHING IN AL?
05B2 75 01                      JNZ     G21
05B4 C3                         RET

                        ;---------------------------------
                        ;     PERFORM SENSE STATUS      :
                        ;---------------------------------

                                                                ; SENSE STATUS CAN BE ISSUED MULTIPLE
                                                                ;  TIMES
05B5                    G21:
05B5 C6 46 F8 03                MOV     CMD_BLOCK+0,SENSE_CMD
05B9 2A C0                      SUB     AL,AL                   ; WRITE ZERO IN INT/DMA MASK
05BB E8 057C R                  CALL    COMMAND                 ; ISSUE SENSE STATUS COMMAND
05BE 72 26                      JC      G24                     ; CANNOT RECOVER-EXIT WITH COMMAND
                                                                ;  ERROR
05C0 2B FF                      SUB     DI,DI                   ; SET INDEX POINTER TO ZERO
05C2 B9 0004                    MOV     CX,4                    ; READ FOUR BYTES
05C5 B4 0B                      MOV     AH,00001011B            ; SET MASK FOR DATA MODE CARD TO CPU
05C7                    G22:
05C7 E8 068D R                  CALL    HD_WAIT                 ; GO WAIT FOR DATA INPUT STATE
05CA 72 1A                      JC      G24
05CC 4A                         DEC     DX                      ; ADDRESS PORT 0
05CD EC                         IN      AL,DX                   ; READ THE DATA BYTE
05CE 88 43 F8                   MOV     [DI+CMD_BLOCK],AL       ; STORE AWAY SENSE BYTES
05D1 47                         INC     DI                      ; NEXT DATA LOCATION
05D2 E2 F3                      LOOP    G22                     ; LOOP TILL ALL FOUR READ.
05D4 B4 0F                      MOV     AH,00001111B            ; SET THE MASK FOR STATUS MODE
05D6 E8 068D R                  CALL    HD_WAIT                 ; GO WAIT FOR STATUS STATE
05D9 72 0B                      JC      G24
05DB 4A                         DEC     DX                      ; ADDRESS PORT 0
05DC EC                         IN      AL,DX                   ; READ THE STATUS BYTE
05DD A8 02                      TEST    AL,2                    ; SENSE OPERATION FAIL?
05DF 74 0F                      JZ      STAT_ERR                ; GO GET THE ERROR.

05E1 C6 06 0074 R FF            MOV     DISK_STATUS,SENSE_FAIL  ; SET SENSE OPERATION FAIL
05E6                    G24:
05E6 F9                         STC
05E7 C3                         RET
05E8                    ERROR_CHK       ENDP

05E8 061E R             T_0     DW      TYPE_0                  ; ERROR TYPE JUMP TABLE
05EA 062B R                     DW      TYPE_1
05EC 066D R                     DW      TYPE_2
05EE 067A R                     DW      TYPE_3

05F0                    STAT_ERR:
05F0 8A 5E F8                   MOV     BL,CMD_BLOCK+0          ; GET ERROR BYTE
05F3 8A C3                      MOV     AL,BL
05F5 24 0F                      AND     AL,0FH
05F7 80 E3 30                   AND     BL,00110000B            ; ISOLATE THE TYPE OF ERROR
05FA 2A FF                      SUB     BH,BH
05FC B1 03                      MOV     CL,3
05FE D3 EB                      SHR     BX,CL                   ; ADJUST
0600 2E: FF A7 05E8 R           JMP     WORD PTR CS:[BX + OFFSET T_0]

0605                    TYPE0_TABLE     LABEL   BYTE
0605 00 20 40 CC 80 00          DB      0,BAD_CNTLR,BAD_SEEK,WRITE_FAULT,TIME_OUT,0,BAD_CNTLR
       20
060C 00 40                      DB      0,BAD_SEEK
= 0009                  TYPE0_LEN       EQU     $-TYPE0_TABLE

060E                    TYPE1_TABLE     LABEL   BYTE
060E 04 10 02 00 04             DB      RECORD_NOT_FND,BAD_ECC,BAD_ADDR_MARK,0,RECORD_NOT_FND
0613 40 00 00 11 0B             DB      BAD_SEEK,0,0,DATA_CORRECTED,BAD_TRACK
= 000A                  TYPE1_LEN       EQU     $-TYPE1_TABLE

0618                    TYPE2_TABLE     LABEL   BYTE
0618 01 02 01                   DB      BAD_CMD,BAD_ADDR_MARK,BAD_CMD
= 0003                  TYPE2_LEN       EQU     $-TYPE2_TABLE

061B                    TYPE3_TABLE     LABEL   BYTE
061B 20 20 10                   DB      BAD_CNTLR,BAD_CNTLR,BAD_ECC
= 0003                  TYPE3_LEN       EQU     $-TYPE3_TABLE
```

```
                        PAGE
                        ;-----  TYPE 0 ERROR
                        TYPE_0:
                                MOV     BX,OFFSET TYPE0_TABLE
                                CMP     AL,TYPE0_LEN            ; CHECK IF ERROR IS DEFINED
                                JAE     UNDEF_ERR_L
                                XLAT    CS:TYPE0_TABLE          ; TABLE LOOKUP
                                MOV     DISK_STATUS,AL          ; SET ERROR CODE
                                RET

                        ;------ TYPE 1 ERROR

                        TYPE_1:
                                MOV     BX,OFFSET TYPE1_TABLE
                                MOV     CX,AX
                                CMP     AL,TYPE1_LEN            ; CHECK IF ERROR IS DEFINED
                                JAE     UNDEF_ERR_L
                                XLAT    CS:TYPE1_TABLE          ; TABLE LOOKUP
                                MOV     DISK_STATUS,AL          ; SET ERROR CODE
                                AND     CL,08H                  ; CORRECTED ECC
                                CMP     CL,08H
                                JNZ     G30

                        ;-----  OBTAIN ECC ERROR BURST LENGTH

                                MOV     CMD_BLOCK+0,RD_ECC_CMD
                                SUB     AL,AL
                                CALL    COMMAND                 ; ISSUE THE COMMAND
                                JC      G30
                                MOV     AH,00001011B            ; SET MASK FOR DATA INPUT CARD TO CPU
                                CALL    HD_WAIT                 ; GO WAIT FOR THE INPUT STATE
                                JC      G30
                                DEC     DX                      ; ADDRESS PORT 0
                                IN      AL,DX                   ; READ THE LENGTH OF THE ERROR
                                MOV     CL,AL                   ;  CORRECTED AND SAVE IN CL
                                MOV     AH,00001111B            ; SET MASK FOR STATUS STATE
                                CALL    HD_WAIT                 ; GO WAIT FOR STATUS STATE
                                JC      G30
                                DEC     DX                      ; ADDRESS PORT 0
                                IN      AL,DX                   ; READ THE STATUS BYTE
                                TEST    AL,2                    ; ERROR BIT SET?
                                JZ      G30
                                MOV     DISK_STATUS,BAD_CNTLR
                                STC
                        G30:
                                MOV     AL,CL
                                RET

                        ;-----  TYPE 2 ERROR

                        TYPE_2:
                                MOV     BX,OFFSET TYPE2_TABLE
                                CMP     AL,TYPE2_LEN            ; CHECK IF ERROR IS DEFINED
                                JAE     UNDEF_ERR_L
                                XLAT    CS:TYPE2_TABLE          ; TABLE LOOKUP
                                MOV     DISK_STATUS,AL          ; SET ERROR CODE
                                RET

                        ;-----  TYPE 3 ERROR

                        TYPE_3:
                                MOV     BX,OFFSET TYPE3_TABLE
                                CMP     AL,TYPE3_LEN            ; CHECK IF ERROR IS DEFINED
                                JAE     UNDEF_ERR_L
                                XLAT    CS:TYPE3_TABLE          ; TABLE LOOKUP
                                MOV     DISK_STATUS,AL          ; SET ERROR CODE
                                RET

                        UNDEF_ERR_L:
                                MOV     DISK_STATUS,UNDEF_ERR
                                RET

                        ;--------------------------------------------------------
                        ; ON ENTRY AH CONTAINS THE CONTROLLER BUS STATUS DECODE :
                        ; MASK USED TO CHECK THE HARDWARE STATUS.               :
                        ;--------------------------------------------------------
                        HD_WAIT         PROC    NEAR
                                PUSH    CX                      ; SAVE CX
                                SUB     CX,CX                   ; SET THE LOOP COUNT
                        L1:
                                CALL    PORT_0
                                INC     DX                      ; PORT_1 ADDRESS
                                IN      AL,DX                   ; READ THE HARDWARE STATUS
                                AND     AL,00001111B            ; CLEAR UPPER NIBBLE OF HARDWARE STATUS
                                CMP     AL,AH                   ; CHECK THE STATE WITH THE MASK
                                JZ      L2                      ; JMP IF O.K WITH CARRY CLEARED
                                LOOP    L1                      ; TRY AGAIN
                                MOV     DISK_STATUS,TIME_OUT
                                STC                             ; SET ERROR CONDITION
                        L2:
                                POP     CX                      ; RESTORE CX
                                RET
                        HD_WAIT         ENDP
```

```
                        PAGE
                        ;--------------------------------------------------------
                        ; DMA_SETUP                                             :
                        ;       THIS ROUTINE SETS UP FOR DMA OPERATIONS.        :
                        ; INPUT                                                 :
                        ;       (AL) = MODE BYTE FOR THE DMA                    :
                        ;       (ES:BX) = ADDRESS TO READ/WRITE THE DATA        :
                        ; OUTPUT                                                :
                        ;       (AX) DESTROYED                                  :
                        ;--------------------------------------------------------

06A5                    DMA_SETUP       PROC    NEAR
06A5 80 7E FC 81                CMP     CMD_BLOCK+4,81H         ; BLOCK COUNT OUT OF RANGE
06A9 72 02                      JB      J1

06AB F9                         STC                             ; SET THE ERROR CONDITION
06AC C3                         RET

06AD                    J1:
06AD FA                         CLI                             ; NO MORE INTERRUPTS
06AE E6 0C                      OUT     DMA+12,AL               ; SET THE FIRST/LAST F/F
06B0 B1 04                      MOV     CL,4                    ; SHIFT COUNT
06B2 E6 0B                      OUT     DMA+11,AL               ; OUTPUT THE MODE BYTE
06B4 8C C0                      MOV     AX,ES                   ; GET THE ES VALUE
06B6 D3 C0                      ROL     AX,CL                   ; ROTATE LEFT
06B8 8A E8                      MOV     CH,AL                   ; GET HIGHEST NIBBLE OF ES TO CH
06BA 24 F0                      AND     AL,0F0H                 ; ZERO THE LOW NIBBLE FROM SEGMENT
06BC 03 C3                      ADD     AX,BX                   ; TEST FOR CARRY FROM ADDITION
06BE 80 D5 00                   ADC     CH,0                    ; CARRY MEANS HIGH 4 BITS MUST BE INC

06C1 8B F0                      MOV     SI,AX                   ; SAVE START ADDRESS
06C3 E6 06                      OUT     DMA+6,AL                ; OUTPUT LOW ADDRESS
06C5 8A C4                      MOV     AL,AH
06C7 E6 06                      OUT     DMA+6,AL                ; OUTPUT HIGH ADDRESS
06C9 8A C5                      MOV     AL,CH                   ; GET HIGH 4 BITS
06CB 24 0F                      AND     AL,0FH
06CD E6 82                      OUT     DMA_HIGH,AL             ; OUTPUT THE HIGH 4 BITS TO PAGE REG

                        ;------ DETERMINE COUNT

06CF 8A 66 FC                   MOV     AH,CMD_BLOCK+4          ; RECOVER BLOCK COUNT
06D2 D0 E4                      SHL     AH,1                    ; MULTIPLY BY 512 BYTES PER SECTOR
06D4 32 C0                      XOR     AL,AL                   ; CLEAR LOW BYTE
06D6 48                         DEC     AX                      ;   AND DECREMENT VALUE BY ONE

                        ;-----  HANDLE READ AND WRITE LONG (516D BYTE BLOCKS)

06D7 80 7E F8 E5                CMP     CMD_BLOCK+0,RD_LONG_CMD
06DB 74 06                      JE      ADD4

06DD 80 7E F8 E6                CMP     CMD_BLOCK+0,WR_LONG_CMD
06E1 75 0F                      JNE     J20
06E3                    ADD4:
06E3 B8 0204                    MOV     AX,516D                 ; ONE BLOCK (512) PLUS 4 BYTES ECC
06E6 53                         PUSH    BX
06E7 2A FF                      SUB     BH,BH
06E9 8A 5E FC                   MOV     BL,CMD_BLOCK+4
06EC 52                         PUSH    DX
06ED F7 E3                      MUL     BX                      ; BLOCK COUNT TIMES 516
06EF 5A                         POP     DX
06F0 5B                         POP     BX
06F1 48                         DEC     AX                      ; ADJUST
06F2                    J20:
06F2 8B C8                      MOV     CX,AX                   ; SAVE COUNT VALUE
06F4 E6 07                      OUT     DMA+7,AL                ; LOW BYTE OF COUNT
06F6 8A C4                      MOV     AL,AH
06F8 E6 07                      OUT     DMA+7,AL                ; HIGH BYTE OF COUNT
06FA FB                         STI                             ; INTERRUPTS BACK ON
06FB 8B C6                      MOV     AX,SI                   ; RECOVER ADDRESS VALUE
06FD 03 C1                      ADD     AX,CX                   ; ADD, TEST FOR 64K OVERFLOW
06FF C3                         RET                             ; RETURN TO CALLER,
                                                                ;   CY SET BY ABOVE IF ERROR
0700                    DMA_SETUP       ENDP
```

```
                        PAGE
                        ;--------------------------------------------
                        ; WAIT_INT                                   :
                        ;       THIS ROUTINE WAITS FOR THE FIXED DISK  :
                        ;       CONTROLLER TO SIGNAL THAT AN INTERRUPT  :
                        ;       HAS OCCURRED.                          :
                        ;--------------------------------------------

0700                    WAIT_INT        PROC    NEAR
                                ASSUME  DS:ABS0
0700 FB                         STI                             ; TURN ON INTERRUPTS
0701 8C DB                      MOV     BX,DS                   ; SAVE DS
0703 2B C0                      SUB     AX,AX
0705 8E D8                      MOV     DS,AX                   ; ESTABLISH SEGMENT
0707 C4 36 0104 R               LES     SI,HF_TBL_VEC           ; LOAD THE TABLE VECTOR
                                ASSUME  DS:DATA,ES:NOTHING
070B 8E DB                      MOV     DS,BX                   ; RESTORE DS

                        ;-----  SET TIMEOUT VALUES

070D 2A FF                      SUB     BH,BH
070F 26: 8A 5C 09               MOV     BL,BYTE PTR ES:[SI][9]  ; LOAD THE STANDARD TIME OUT
0713 8A 66 F8                   MOV     AH,CMD_BLOCK+0
0716 80 FC 04                   CMP     AH,FMTDRV_CMD
0719 75 06                      JNZ     W5

071B 26: 8A 5C 0A               MOV     BL,BYTE PTR ES:[SI][0AH]        ; LOAD THE FORMAT DRIVE
071F EB 09                      JMP     SHORT   W4                      ;  TIME OUT VALUE
0721 80 FC E3           W5:     CMP     AH,CHK_DRV_CMD
0724 75 04                      JNZ     W4

0726 26: 8A 5C 0B               MOV     BL,BYTE PTR ES:[SI][0BH]        ; LOAD THE CHECK DRIVE
072A                    W4:                                             ;  TIME OUT VALUE
072A F8                         CLC                             ; CLEAR CY
072B B8 9000                    MOV     AX,9000H                ; DEVICE WAIT INTERRUPT
072E CD 15                      INT     15H
0730 FB                         STI                             ; ENABLE INTERRUPTS FOR PC AND
                                                                ;  XT MACHINES.
0731 2B C9                      SUB     CX,CX                   ; SET THE LOOP COUNT

                        ;-----  WAIT FOR INTERRUPT

0733                    W1:
0733 E8 076D R                  CALL    PORT_0
0736 42                         INC     DX                      ; PORT_1 ADDRESS
0737 EC                         IN      AL,DX                   ; READ THE HARDWARE STATUS
0738 A8 20                      TEST    AL,020H                 ; DID INTERRUPT OCCUR
073A 75 0A                      JNZ     W2                      ; JUMP IF YES

073C E2 F5                      LOOP    W1                      ; INNER LOOP
073E 4B                         DEC     BX
073F 75 F2                      JNZ     W1                      ; OUTER LOOP

0741 C6 06 0074 R 80            MOV     DISK_STATUS,TIME_OUT
0746                    W2:
0746 4A                         DEC     DX                      ; ADDRESS PORT 0
0747 EC                         IN      AL,DX                   ; READ THE STATUS BYTE
0748 24 02                      AND     AL,2                    ; ISOLATE THE ERROR BIT
074A 08 06 0074 R               OR      DISK_STATUS,AL          ; SAVE IN THE STATUS
074E 83 C2 03                   ADD     DX,3                    ; PORT_3 ADDRESS
0751 32 C0                      XOR     AL,AL                   ; ZERO
0753 EE                         OUT     DX,AL                   ; RESET INTERRUPT MASK
0754 C3                         RET

0755                    WAIT_INT        ENDP

                        ;---  HD_INT  ------------------------------------------
                        ;                                                       :
                        ;     FIXED DISK INTERRUPT 0DH ROUTINE IRQ-5            :
                        ;                                                       :
                        ;-------------------------------------------------------

0755                    HD_INT  PROC    NEAR
0755 50                         PUSH    AX                      ; SAVE WORK REGISTER
0756 B0 07                      MOV     AL,07H                  ; SET DMA MODE TO DISABLE
0758 E6 0A                      OUT     DMA+10,AL
075A FA                         CLI                             ; NO INTERRUPTS
075B E4 21                      IN      AL,INTA01               ; LOAD THE INTERRUPT ENABLE MASK
075D 0C 20                      OR      AL,020H                 ; TURN OFF FIXED DISK IRQ-5
075F E6 21                      OUT     INTA01,AL               ; REPLACE THE MASK
0761 B0 20                      MOV     AL,EOI                  ; LOAD THE END OF INTERRUPT MASK
0763 E6 20                      OUT     INTA00,AL               ; CLEAR THE ACTIVE INTERRUPT LEVEL
0765 FB                         STI                             ; INTERRUPTS BACK ON
0766 B8 9100                    MOV     AX,9100H                ; DEVICE POST
0769 CD 15                      INT     15H                     ;  INTERRUPT
076B 58                         POP     AX                      ; RESTORE AX
076C CF                         IRET
076D                    HD_INT  ENDP

                        ;--------------------------------------
                        ; PORTS                                :
                        ;       GENERATE PROPER PORT VALUE     :
                        ;       BASED ON THE PORT OFFSET       :
                        ;--------------------------------------

076D                    PORT_0  PROC    NEAR
076D BA 0320                    MOV     DX,HF_PORT              ; BASE VALUE
0770 02 16 0077 R               ADD     DL,PORT_OFF             ; ADD IN OFFSET VALUE (00,04,08,0C)
0774 C3                         RET
0775                    PORT_0  ENDP

0775                    END_ADDRESS     LABEL   BYTE
0775                    CODE    ENDS
                                END
```

Notes:

Index

Personal Computer Hardware Reference Library

IBM Asynchronous Communications Adapter

6361501

IBM Asynchronous Communications Adapter

Contents

Description

The Asynchronous Communications Adapter's system control signals and voltage requirements are provided through a 2- by 31-position card-edge connector. Two jumper modules are provided on the adapter. One jumper module selects either RS-232C or current-loop operation. The other jumper module selects one of two addresses for the adapter, so two adapters may be used in one system. An additional jumper is required on connector J13 if the adapter is to be installed in expansion slot 8 of an IBM Personal Computer XT or IBM Portable Personal Computer (see "Selecting the Interface Format and Adapter Address" in this section).

The adapter is fully programmable and supports asynchronous communications only. It will add and remove start bits, stop bits, and parity bits. A programmable baud-rate generator allows operation from 50 baud to 9600 baud. Five-, six-, seven-, or eight-bit characters with 1, 1-1/2, or 2 stop bits are supported. A fully prioritized interrupt system controls transmit, receive, error, line status, and data set interrupts. Diagnostic capabilities provide loopback functions of transmit/receive and input/output signals.

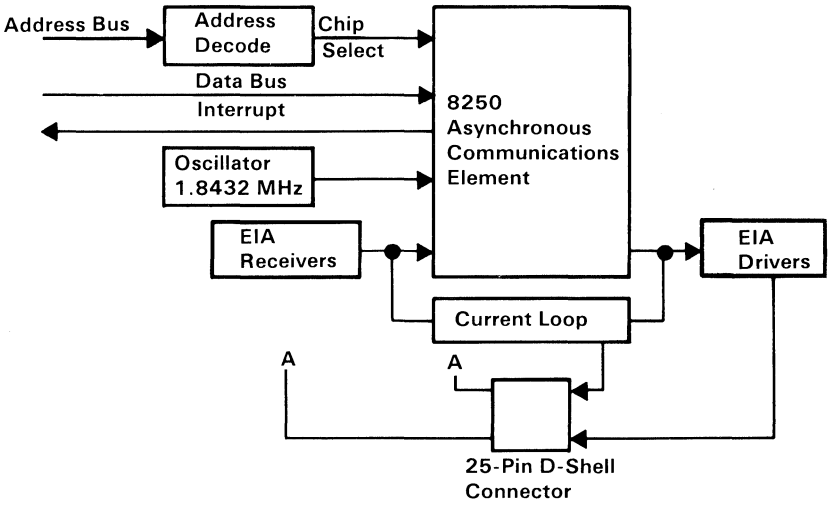
The major component of the adapter is an INS8250 LSI chip or functional equivalent. Features in addition to those listed above are:

- Full double buffering eliminating the need for precise synchronization
- Independent receiver clock input
- 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)
  Carrier detect (CD)

All communication protocol is a function of the system microcode and must be loaded before the adapter is operational. All pacing of the interface and control signal status must be handled by the system software. The following figure is a block diagram of the IBM Asynchronous Communications Adapter.



Asynchronous Communications Adapter Block Diagram

Programming Considerations

Modes Of Operation

The different modes of operation are selected by programming the 8250 Asynchronous Communications Element. This is done by selecting the I/O address (hex 3F8 to 3FF primary, and hex 2F8 to 2FF secondary) and writing data out to the adapter. Address bits A0, A1, and A2, select the different registers that define the modes of operation. Also, bit 7—the divisor latch access bit (DLAB)—of the line-control register is used to select certain registers.

I/O Decode (in Hex)		Register Selected	DLAB State
Primary Adapter	Alternate Adapter		
3F8	2F8	TX Buffer	DLAB = 0 (Write)
3F8	2F8	RX Buffer	DLAB = 0 (Read)
3F8	2F8	Divisor Latch LSB	DLAB = 1
3F9	2F9	Divisor Latch MSB	DLAB = 1
3F9	2F9	Interrupt Enable Register	
3FA	2FA	Interrupt Identification Registers	
3FB	2FB	Line Control Register	
3FC	2FC	Modem Control Register	
3FD	2FD	Line Status Register	
3FE	2FE	Modem Status Register	

I/O Decodes

Hex Addresses 3F8 to 3FF AND 2F8 TO 2FF											
A9	A8	A7	A6	A5	A4	A3	A2	A1	A0	DLAB	Register
1	1/0	1	1	1	1	1	x	x	x		
							0	0	0	0	Receive Buffer (read). Transmit Holding Reg. (write)
							0	0	1	0	Interrupt Enable
							0	1	0	x	Interrupt Identification
							0	1	1	x	Line Control
							1	0	0	x	Modem Control
							1	0	1	x	Line Status
							1	1	0	x	Modem Status
							1	1	1	x	None
							0	0	0	1	Divisor Latch (LSB)
							0	0	1	1	Divisor Latch (MSB)

Note: Bit 8 will be logical 1 for the adapter designated as primary or a logical 0 for the adapter designated as alternate (as defined by the address jumper module on the adapter).

A2, A1 and A0 bits are "don't cares" and are used to select the different register of the communications chip.

Address Bits

INS8250

The INS8250 has a number of accessible registers. The system programmer may access or control any of the INS8250 registers through the system unit's microprocessor. These registers are used to control INS8250 operations and to transmit and receive data. The following figure provides a listing and description of the accessible registers.

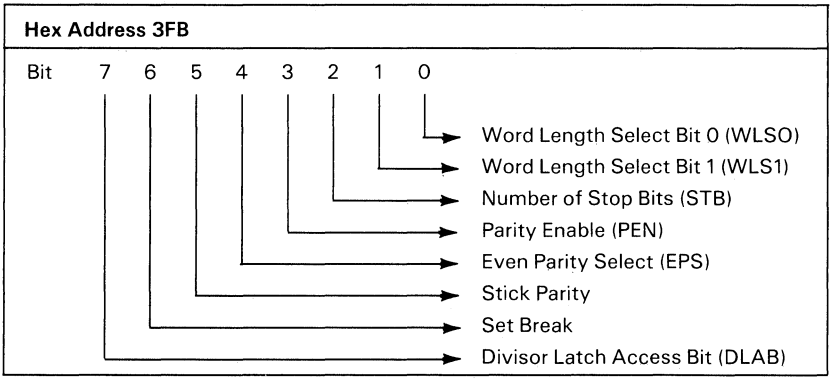
Register/Signal	Reset Control	Reset State
Interrupt Enable Register	Master Reset	All bits Low (0-3 Forced and 4-7 Permanent).
Interrupt Identification Register	Master Reset	Bit 0 is High, Bits 1 and 2 Low Bits 3-7 are Permanently Low
Line Control Register	Master Reset	All Bits Low
Modem Control Register	Master Reset	All Bits Low
Line Status Register	Master Reset	Except Bits 5 and 6 are High
Modem Status Register	Master Reset	Bits 0-3 Low Bits 4-7 - Input Signal
SOUT	Master Reset	High
INTRPT (RCVR Errors)	Read LSR/MR	Low
INTRPT (RCVR Data Ready)	Read RBR/MR	Low
INTRPT (RCVR Data Ready)	Read IIR/ Write THR/MR	Low
INTRPT (Modem Status Changes)	Read MSR/MR	Low
OUT 2	Master Reset	High
RTS	Master Reset	High
DTR	Master Reset	High
OUT 1	Master Reset	High

Asynchronous Communications Reset Functions

Line-Control Register

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

The contents of the line-control register are as follows:



Bits 0 and 1: These two bits specify the number of bits in each transmitted or received serial character. The encoding of bits 0 and 1 is as follows:

Bit 1	Bit 0	Word Length
0	0	5 Bits
0	1	6 Bits
1	0	7 Bits
1	1	8 Bits

Bit 2: This bit specifies the number of stop bits in each transmitted or received serial character. If bit 2 is a logical 0, one stop bit is generated or checked in the transmitted or received data, respectively. If bit 2 is logical 1 when a 5-bit word length is selected through bits 0 and 1, 1-1/2 stop bits are generated or checked. If bit 2 is logical 1 when either a 6-, 7-, or 8-bit word length is selected, two stop bits are generated or checked.

Bit 3: This bit is the parity enable bit. When bit 3 is a logical 1, a parity bit is generated (transmit data) or checked (receive data) between the last data word bit and stop bit of the serial data. (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 4: This bit is the even parity select bit. When bit 3 is a logical 1 and bit 4 is a logical 0, an odd number of logical 1's is transmitted or checked in the data word bits and parity bit. When bit 3 is a logical 1 and bit 4 is a logical 1, an even number of bits is transmitted or checked.

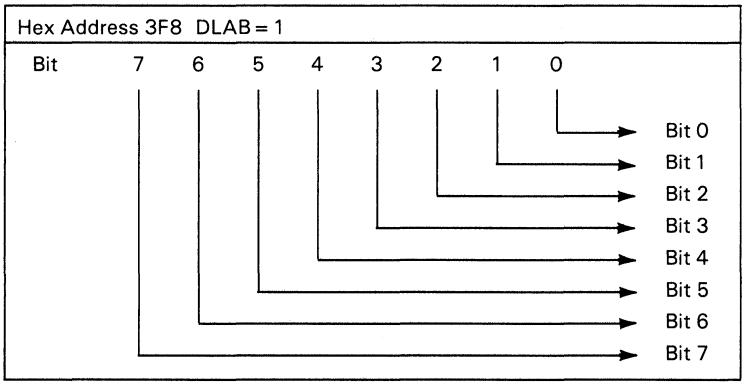
Bit 5: This bit is the stick parity bit. When bit 3 is a logical 1 and bit 5 is a logical 1, the parity bit is transmitted and then detected by the receiver as a logical 0 if bit 4 is a logical 1, or as a logical 1 if bit 4 is a logical 0.

Bit 6: This bit is the set break control bit. When bit 6 is a logical 1, the serial output (SOUT) is forced to the spacing (logical 0) state and remains there regardless of other transmitter activity. The set break is disabled by setting bit 6 to a logical 0. This feature enables the system unit's microprocessor to alert a terminal in a computer communications system.

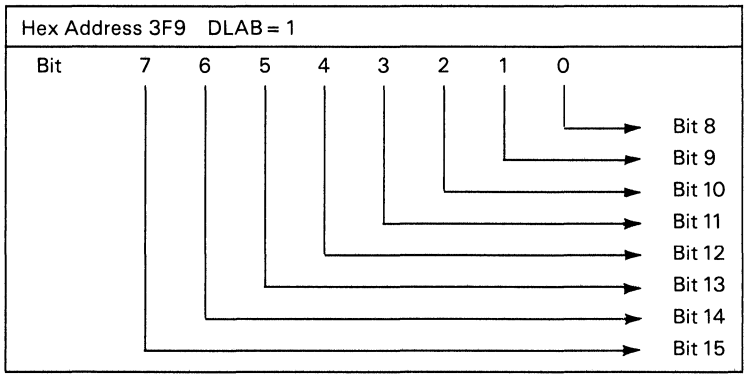
Bit 7: This bit is the divisor latch access bit (DLAB). It must be set high (logical 1) to access the divisor latches of the baud-rate generator during a read or write operation. It must be set low (logical 0) to access the receiver buffer, the transmitter holding register, or the interrupt enable register.

Programmable Baud-Rate Generator

The INS8250 contains a programmable baud-rate generator that is capable of taking the clock input (1.8432 MHz) and dividing it by any divisor from 1 to (2^{16}-1). The output frequency of the baud generator is 16 x the baud rate (divisor # =(frequency input)/(baud rate x 16)). Two 8-bit latches store the divisor in a 16-bit binary format. These divisor latches must be loaded during initialization in order to ensure desired operation of the baud-rate generator. Upon loading either of the divisor latches, a 16-bit baud counter is immediately loaded. This prevents long counts on initial load.



Divisor Latch Least Significant Bit (DLL)



Divisor Latch Most Significant Bit (DLM)

The following figure illustrates the use of the baud-rate generator with a frequency of 1.8432 MHz. For baud rates of 9600 and below, the error obtained is minimal.

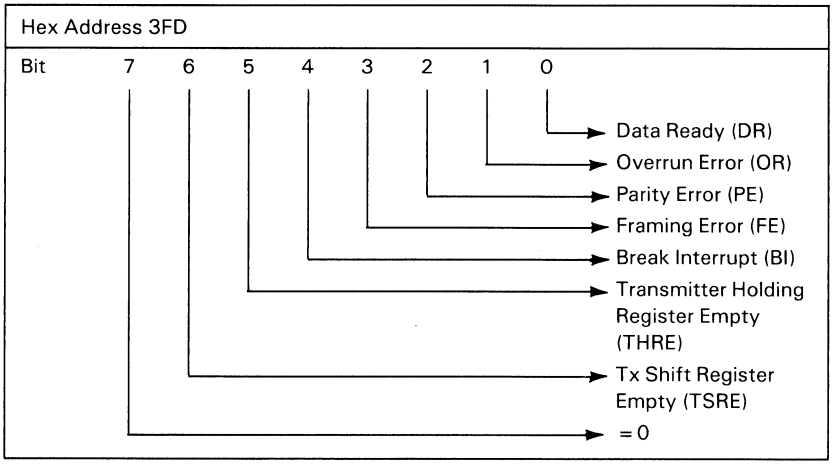
Note: The maximum operating frequency of the baud-rate generator is 3.1 MHz. In no case should the data speed be greater than 9600 baud.

Desired Baud Rate	Divisor Used to Generate 16x Clock (Decimal)	(Hex)	Percent Error Difference Between Desired and Actual
50	2304	900	—
75	1536	600	—
110	1047	417	0.026
134.5	857	359	0.058
150	768	300	—
300	384	180	—
600	192	0C0	—
1200	96	060	—
1800	64	040	—
2000	58	03A	0.69
2400	48	030	—
3600	32	020	—
4800	24	018	—
7200	16	010	—
9600	12	00C	—

Baud Rate at 1.843 MHz

Line Status Register (LSR)

This 8-bit register provides status information to the system unit's microprocessor concerning the data transfer. The contents of the line status register are indicated and described in the following figure.



Line Status Register (LSR)

Bit 0: This bit is the receiver data ready (DR) indicator. Bit 0 is set to logical 1 whenever a complete incoming character has been received and transferred into the receiver buffer register. Bit 0 may be reset to a logical 0 either by the system unit's microprocessor reading the data in the receiver buffer register or by writing logical 0 into it from the system unit's microprocessor.

Bit 1: This bit is the overrun error (OE) indicator. Bit 1 indicates that data in the receiver buffer register was not read by the system unit's microprocessor before the next character was transferred into the receiver buffer register, thereby destroying the previous character. The OE indicator is reset whenever the system unit's microprocessor reads the contents of the line status register.

Bit 2: This bit is the parity error (PE) indicator. Bit 2 indicates that the received data character does not have the correct even or odd parity, as selected by the even-parity select bit. The PE bit is

set to logical 1 upon detection of a parity error and is reset to logical 0 whenever the system unit's microprocessor reads the contents of the line status register.

Bit 3: This bit is the framing error (FE) indicator. Bit 3 indicates that the received character did not have a valid stop bit. Bit 3 is set to logical 1 whenever the stop bit following the last data bit or parity is detected as a zero bit (spacing level).

Bit 4: This bit is the break interrupt (BI) indicator. Bit 4 is set to logical 1 whenever the received data input is held in the spacing (logical 0) 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 5: This bit is the transmitter-holding-register-empty (THRE) indicator. Bit 5 indicates that the INS8250 is ready to accept a new character for transmission. In addition, this bit causes the INS8250 to issue an interrupt to the system unit's microprocessor when the transmit-holding-register-empty interrupt enable is set high. The THRE bit is set to logical 1 when a character is transferred from the transmitter holding register into the transmitter shift register. The bit is reset to logical 0 concurrently with the loading of the transmitter holding register by the system unit's microprocessor.

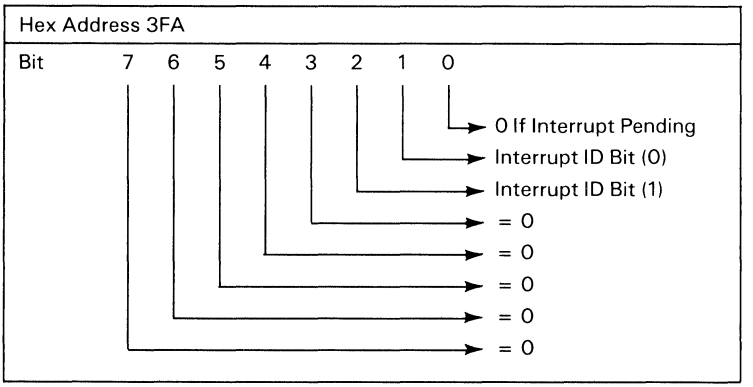
Bit 6: This bit is the transmitter-shift-register-empty (TSRE) indicator. Bit 6 is set to logical 1 whenever the transmitter shift register is idle. It is reset to logical 0 upon a data transfer from the transmitter holding register to the transmitter shift register. Bit 6 is a read-only bit.

Bit 7: This bit is permanently set to logical 0.

Interrupt Identification Register (IIR)

The INS8250 has an on-chip interrupt capability that allows for complete flexibility in interfacing to all the popular microprocessors presently available. In order to provide minimum software overhead during data character transfers, the INS8250 prioritizes interrupts into four levels: receiver line status (priority 1), received data ready (priority 2), transmitter holding register empty (priority 3), and modem status (priority 4).

Information indicating that a prioritized interrupt is pending, and the type of prioritized interrupt, is stored in the interrupt identification register. Refer to the "Interrupt Control Functions" table. The interrupt identification register (IIR), when addressed during chip-select time, freezes the highest priority interrupt pending, and no other interrupts are acknowledged until that particular interrupt is serviced by the system unit's microprocessor. The contents of the IIR are indicated and described in the following figure.



Interrupt Identification Register (IIR)

Bit 0: This bit can be used in either a hard-wired prioritized or polled environment to indicate whether an interrupt is pending, and the IIR contents may be used as a pointer to the appropriate interrupt service routine. When bit 0 is logical 1, no interrupt is pending, and polling (if used) is continued.

Bits 1 and 2: These two bits of the IIR are used to identify the highest priority interrupt pending, as indicated in the "Interrupt Control Functions" table.

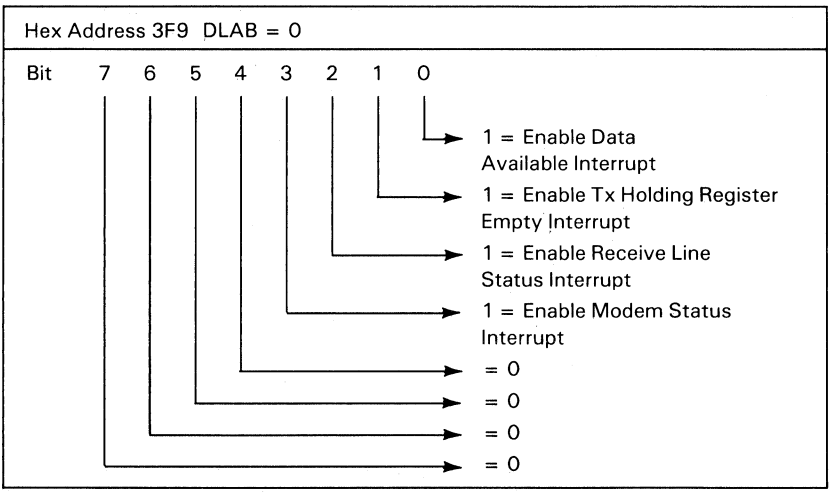
Bits 3 through 7: These five bits of the IIR are always logical 0.

Interrupt ID Register			Interrupt Set and Reset Functions			
Bit 2	Bit 1	Bit 0	Priority Level	Interrupt Type	Interrupt Source	Interrupt Reset Control
0	0	1	—	None	None	—
1	1	0	Highest	Receiver Line Status	Overrun Error or Parity Error or Framing Error or Break Interrupt	Reading the Line Status Register
1	0	0	Second	Received Data Available	Receiver Data Available	Reading the Receiver Buffer Register
0	1	0	Third	Transmitter Holding Register Empty	Transmitter Holding Register Empty	Reading the IIR Register (if source of interrupt) or Writing into the Transmitter Holding Register
0	0	0	Fourth	Modem Status	Clear to Send or Data Set Ready or Ring Indicator or Received Line Signal Direct	Reading the Modem Status Register

Interrupt Control Functions

Interrupt Enable Register

This 8-bit register enables the four types of interrupt of the INS8250 to separately activate the chip interrupt (INTRPT) output signal. It is possible to totally disable the interrupt system by resetting bits 0 through 3 of the interrupt enable register. Similarly, by setting the appropriate bits of this register to logical 1, selected interrupts can be enabled. Disabling the interrupt system inhibits the interrupt identification register and the active (high) INTRPT output from the chip. All other system functions operate in their normal manner, including the setting of the line status and modem status registers. The contents of the interrupt enable register are indicated and described in the following figure.



Interrupt Enable Register (IER)

Bit 0: This bit enables the received-data-available interrupt when set to logical 1.

Bit 1: This bit enables the transmitter-holding-register-empty interrupt when set to logical 1.

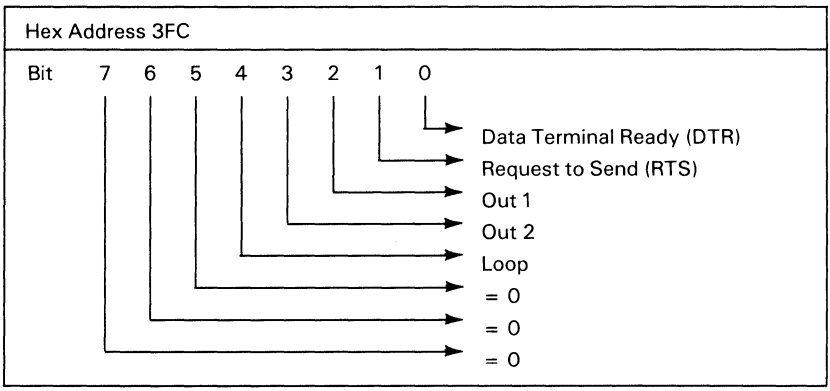
Bit 2: This bit enables the receiver-line-status interrupt when set to logical 1.

Bit 3: This bit enables the modem-status interrupt when set to logical 1.

Bits 4 through 7: These four bits are always logical 0.

Modem Control Register

This 8-bit register controls the interface with the modem or data set (or a peripheral device emulating a modem). The contents of the modem control register are indicated and described as follows:



Modem Control Register (MCR)

Bit 0: This bit controls the data terminal ready (-DTR) output. When bit 0 is set to a high level, the -DTR output is forced to an active low. When bit 0 is reset to low level, the -DTR output is forced high.

> **Note:** The -DTR output of the INS8250 may be applied to an EIA inverting line driver (such as the DS1488) to obtain the proper polarity input at the succeeding modem or data set.

Bit 1: This bit controls the request to send (-RTS) output. Bit 1 affects the -RTS output in a manner identical to that described above for bit 0.

Bit 2: This bit controls the output 1 (-OUT 1) signal, which is an auxiliary user-designated output. Bit 2 affects the -OUT 1 output in a manner identical to that described above for bit 0.

Bit 3: This bit controls the output 2 (-OUT 2) signal, which is an auxiliary user-designated output. Bit 3 affects the -OUT 2 output in a manner identical to that described above for bit 0.

Bit 4: This bit provides a loopback feature for diagnostic testing of the INS8250. When bit 4 is set to logical 1, the following occurs: the transmitter serial output (SOUT) is set to the marking (logical 1) state; the receiver serial input (SIN) is disconnected; the output of the transmitter shift register is "looped back" into the receiver shift register input; the four modem control inputs (-CTS, -DSR, -RLSD, and -RI) are disconnected; and the four modem control outputs (-DTR, -RTS, -OUT 1, and -OUT 2) are internally connected to the four modem control inputs. In the diagnostic mode, data that is transmitted is immediately received. This feature allows the system unit's microprocessor to verify the transmit-data and receive-data paths of the INS8250.

In the diagnostic mode, the receiver and transmitter interrupts are fully operational. The modem control interrupts also are operational, but the interrupts' sources are now the lower four bits of the modem control register instead of the four modem control inputs. The interrupts are still controlled by the interrupt enable register.

The INS8250 interrupt system can be tested by writing into the lower four bits of the modem status register. Setting any of these bits to a logical 1 generates the appropriate interrupt (if enabled). The resetting of these interrupts is the same as in normal INS8250 operation. To return to normal operation, the registers must be reprogrammed for normal operation, then bit 4 of the modem control register must be reset to logical 0.

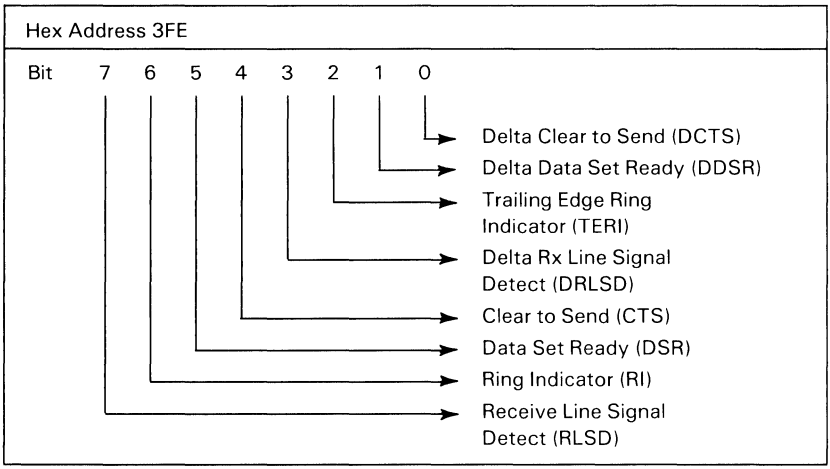
Bits 5 through 7: These bits are permanently set to logical 0.

Modem Status Register

This 8-bit register provides the current state of the control lines from the modem (or peripheral device) to the system unit's microprocessor. In addition to this current-state information, four bits of the modem status register provide change information. These bits are set to logical 1 whenever a control

input from the modem changes state. They are reset to logical 0 whenever the system unit's microprocessor reads the modem status register.

The content of the modem status register is indicated and described in the following figure.



Modem Status Register (MSR)

Bit 0: This bit is the delta clear to send (DCTS) indicator. Bit 0 indicates that the -CTS input to the chip has changed state since the last time it was read by the system unit's microprocessor.

Bit 1: This bit is the delta data set ready (DDSR) indicator. Bit 1 indicates that the -DSR input to the chip has changed state since the last time it was read by the system unit's microprocessor.

Bit 2: This bit is the trailing edge of the ring indicator (TERI) detector. Bit 2 indicates that the -RI input to the chip has changed from an on (logical 1) to an off (logical 0) condition.

Bit 3: This bit is the delta received line signal detector (DRLSD) indicator. Bit 3 indicates that the -RLSD input to the chip has changed state.

> **Note:** Whenever bit 0, 1, 2, or 3 is set to logical 1, a modem status interrupt is generated.

Bit 4: This bit is the complement of the clear to send (-CTS) input. If bit 4 (LOOP) of the modem control register (MCR) is set to logical 1, the bit is equivalent to RTS in the MCR.

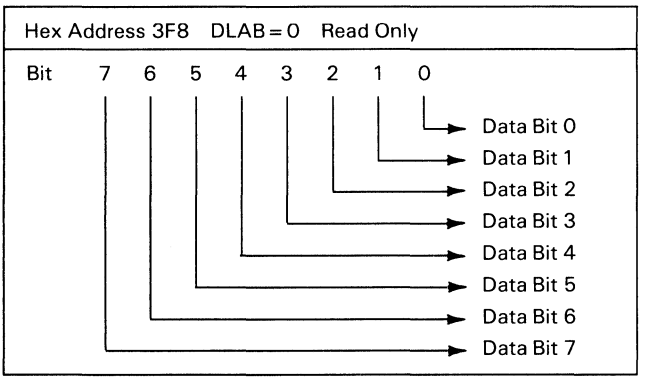
Bit 5: This bit is the complement of the data set ready (-DSR) input. If bit 4 of the MCR is set to logical 1, the bit is equivalent to DTR in the MCR.

Bit 6: This bit is the complement of the ring indicator (-RI) input. If bit 4 of the MCR is set to logical 1, the bit is equivalent to OUT 1 in the MCR.

Bit 7: This bit is the complement of the received line signal detect (-RLSD) input. If bit 4 of the MCR is set to logical 1, the bit is equivalent to OUT 2 of the MCR.

Receiver Buffer Register

The receiver buffer register contains the received character, which is defined in the following figure.

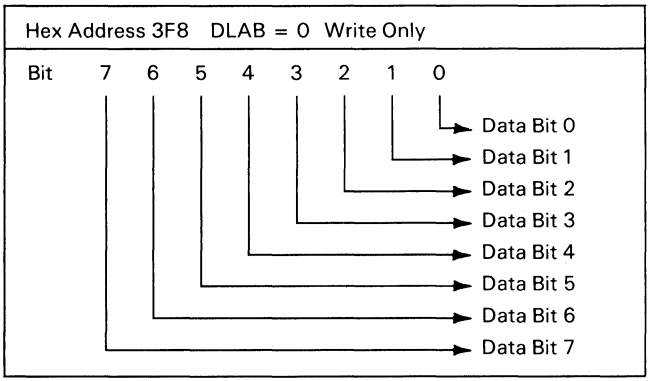


Receiver Buffer Register (RBR)

Bit 0 is the least-significant bit and is the first bit serially received.

Transmitter Holding Register

The transmitter holding register contains the character to be serially transmitted and is defined as follows:

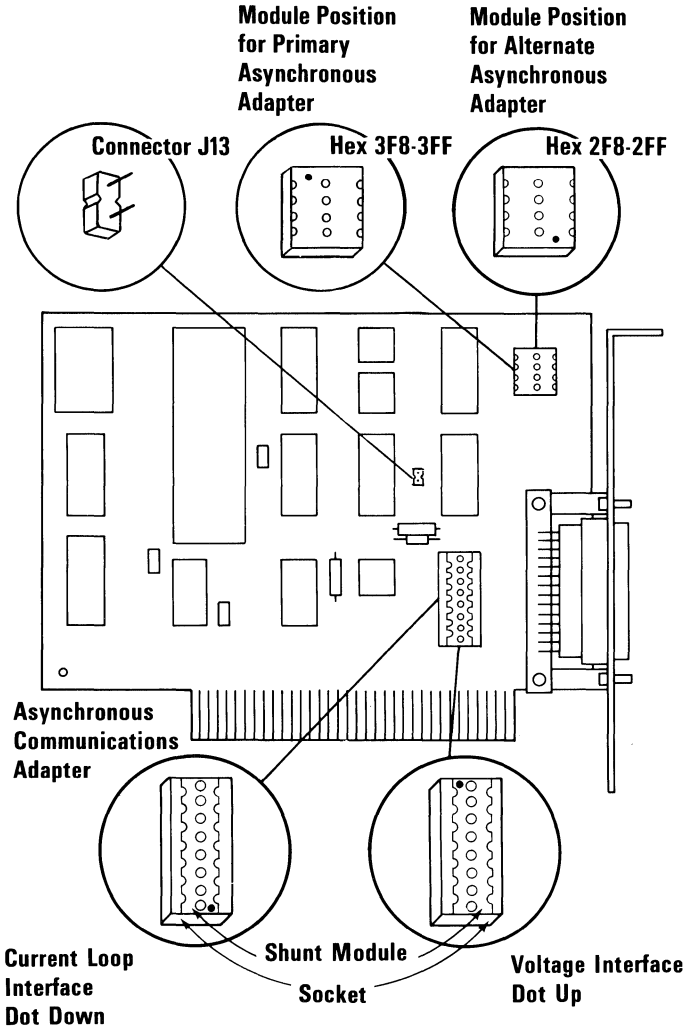


Transmitter Holding Register (THR)

Bit 0 is the least-significant bit and is the first bit serially transmitted.

Selecting the Interface Format and Adapter Address

The voltage or current-loop interface and adapter address are selected by plugging in programmed shunt modules with the locator dots up or down. See the following figure for the configurations.

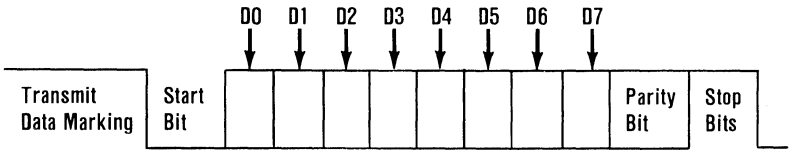


If the adapter is to be installed in expansion slot 8 of an IBM Personal Computer XT or IBM Portable Personal Computer, a jumper is required on connector J13.

Interrupts

One interrupt line is provided to the system. This interrupt is IRQ4 for a primary adapter, or IRQ3 for an alternate adapter, and is positive active. To allow the communications adapter to send interrupts to the system, bit 3 of the modem control register must be set to 1 (high). At this point, any interrupts allowed by the interrupt enable register will cause an interrupt.

The data format will be as follows:



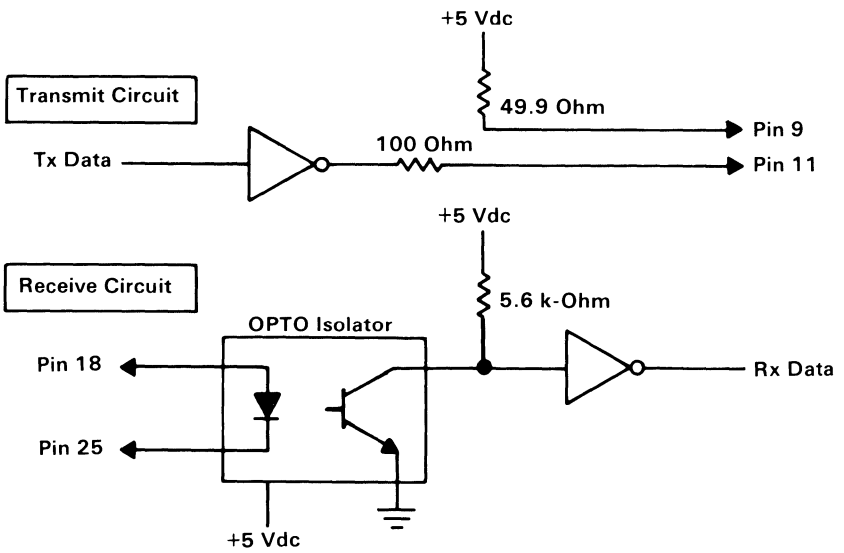
Data bit 0 is the first bit to be transmitted or received. The adapter automatically inserts the start bit, the correct parity bit if programmed to do so, and the stop bit (1, 1-1/2, or 2 depending on the command in the line-control register).

Interface

The communications adapter provides an EIA RS-232C-like interface. One 25-pin, D-shell, male connector is provided to attach various peripheral devices. In addition, a current-loop interface is also located in this same connector. A jumper block is provided to manually select either the voltage interface or the current-loop interface.

The current-loop interface is provided to attach certain printers provided by IBM that use this particular type of interface. IBM recommends that the current loop not be used beyond a distance of 15.3 meters (50 feet) as measured by the length of cable between the two interconnected points.

Pin 18	+ receive current loop data
Pin 25	- receive current loop return
Pin 11	- transmit current loop data
Pin 9	+ transmit current loop return



Current Loop Interface

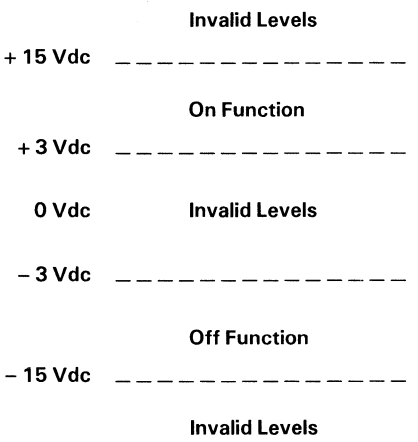
The voltage interface is a serial interface. It supports certain data and control signals, as follows:

Pin 2	Transmitted Data
Pin 3	Received Data
Pin 4	Request to Send
Pin 5	Clear to Send
Pin 6	Data Set Ready
Pin 7	Signal Ground
Pin 8	Carrier Detect
Pin 20	Data Terminal Ready
Pin 22	Ring Indicator

The adapter converts these signals to or from TTL levels from or to EIA voltage levels. These signals are sampled or generated by the communications control chip. These signals can then be sensed by the system software to determine the state of the interface or peripheral device.

Voltage Interchange Information

Interchange Voltage	Binary State	Signal Condition	Interface Control Function
Positive Voltage =	Binary (0)	= Spacing	= On
Negative Voltage =	Binary (1)	= Marking	= Off



The signal will be considered in the *marking* condition when the voltage on the interchange circuit, measured at the interface

point, is more negative than -3 Vdc with respect to signal ground. The signal will be considered in the *spacing* condition when the voltage is more positive than +3 Vdc with respect to signal ground. The region between +3 Vdc and -3 Vdc is defined as the transition region, and considered an invalid level. The voltage that is more negative than -15 Vdc or more positive than +15 Vdc will also be considered an invalid level.

During the transmission of data, the marking condition will be used to denote the binary state 1, and the spacing condition will be used to denote the binary state 0.

For interface control circuits, the function is on when the voltage is more positive than +3 Vdc with respect to signal ground and is off when the voltage is more negative than -3 Vdc with respect to signal ground.

INS8250 Functional Pin Description

The following describes the function of all INS8250 input/output pins. Some of these descriptions refer to internal circuits.

> **Note:** In the following descriptions, a low represents a logical 0 (0 Vdc nominal) and a high represents a logical 1 (+2.4. Vdc nominal).

Input Signals

Chip Select (CS0, CS1, -CS2), Pins 12-14: When CS0 and CS1 are high and -CS2 is low, the chip is selected. Chip selection is complete when the decoded chip select signal is latched with an active (low) address strobe (-ADS) input. This enables communications between the INS8250 and the system unit's microprocessor.

Data Input Strobe (DISTR, -DISTR), Pins 22 and 21: When DISTR is high or -DISTR is low while the chip is selected, it allows the system unit's microprocessor to read status information or data from a selected register of the INS8250.

> **Note:** Only an active DISTR or -DISTR input is required to transfer data from the INS8250 during a read operation. Therefore, tie either the DISTR input permanently low or the -DISTR line input permanently high, if not used.

Data Output Strobe (DOSTR, -DOSTR), Pins 19 and 18: When DOSTR is high or -DOSTR is low while the chip is selected, it allows the system unit's microprocessor to write data or control words into a selected register of the INS8250.

> **Note:** Only an active DOSTR or -DOSTR input is required to transfer data to the INS8250 during a write operation. Therefore, tie either the DOSTR input permanently low or the -DOSTR input permanently high, if not used.

Address Strobe (-ADS), Pin 25: When low, provides latching for the register select (A0, A1, A2) and chip select (CS0, CS1, -CS2) signals.

> **Note:** An active -ADS input is required when the register select (A0, A1, A2) signals are not stable for the duration of a read or write operation. If not required, tie the ADS input permanently low.

Register Select (A0, A1, A2), Pins 26-28: These three inputs are used during a read or write operation to select an INS8250 register to read or write to as indicated in the following table. Note that the state of the divisor latch access bit (DLAB), which is the most significant bit of the line-control register, affects the selection of certain INS8250 registers. The DLAB must be set high by the system software to access the baud-rate generator divisor latches.

DLAB	A2	A1	A0	Register
0	0	0	0	Receiver Buffer (Read), Transmitter Holding Register (Write)
0	0	0	1	Interrupt Enable
X	0	1	0	Interrupt Identification (Read Only)
X	0	1	1	Line Control
X	1	0	0	Modem Control
X	1	0	1	Line Status
X	1	1	0	Modem Control Status
X	1	1	1	None
1	0	0	0	Divisor Latch (Least Significant Bit)
1	0	0	1	Divisor Latch (Most Significant Bit)

Master Reset (MR), Pin 35: When high, clears all registers (except the receiver buffer, transmitter holding, and divisor latches), and the control logic of the INS8250. Also, the state of various output signals (SOUT, INTRPT, -OUT 1, -OUT 2, -RTS, -DTR) are affected by an active MR input. Refer to the "Asynchronous Communications Reset Functions" table.

Receiver Clock (RCLK), Pin 9: This input is the 16 x baud-rate clock for the receiver section of the chip.

Serial Input (SIN), Pin 10: Serial data input from the communications link (peripheral device, modem, or data set).

Clear to Send (-CTS), Pin 36: The -CTS signal is a modem control function input whose condition can be tested by the system unit's microprocessor by reading bit 4 (CTS) of the modem status register. Bit 0 (DCTS) of the modem status register indicates whether the CTS input has changed state since the previous reading of the modem status register.

> **Note:** Whenever the CTS bit of the modem status register changes state, an interrupt is generated if the modem status interrupt is enabled.

Data Set Ready (-DSR), Pin 37: When low, indicates that the modem or data set is ready to establish the communications link and transfer data with the INS8250. The -DSR signal is a modem-control function input whose condition can be tested by

the system unit's microprocessor by reading bit 5 (DSR) of the modem status register. Bit 1 (DDSR) of the modem status register indicates whether the -DSR input has changed since the previous reading of the modem status register.

> **Note:** Whenever the DSR bit of the modem status register changes state, an interrupt is generated if the modem status interrupt is enabled.

Received Line Signal Detect (-RLSD), Pin 38: When low, indicates that the data carrier had been detected by the modem or data set. The -RLSD signal is a modem-control function input whose condition can be tested by the system unit's microprocessor by reading bit 7 (RLSD) of the modem status register. Bit 3 (DRLSD) of the modem status register indicates whether the RLSD input has changed state since the previous reading of the modem status register.

> **Note:** Whenever the RLSD bit of the modem status register changes state, an interrupt is generated if the modem status interrupt is enabled.

Ring Indicator (-RI), Pin 39: When low, indicates that a telephone ringing signal has been received by the modem or data set. The -RI signal is a modem-control function input whose condition can be tested by the system unit's microprocessor by reading bit 6 (RI) of the modem status register. Bit 2 (TERI) of the modem status register indicates whether the -RI input has changed from a low to high state since the previous reading of the modem status register.

> **Note:** Whenever the RI bit of the modem status register changes from a high to a low state, an interrupt is generated if the modem status interrupt is enabled.

VCC, Pin 40: +5 Vdc supply.

VSS, Pin 20: Ground (0 Vdc) reference.

Output Signals

Data Terminal Ready (-DTR), Pin 33: When low, informs the modem or data set that the INS8250 is ready to communicate. The -DTR output signal can be set to an active low by programming bit 0 (DTR) of the modem control register to a high level. The -DTR signal is set high upon a master reset operation.

Request to Send (-RTS), Pin 32: When low, informs the modem or data set that the INS8250 is ready to transmit data. The -RTS output signal can be set to an active low by programming bit 1 (RTS) of the modem control register to a high level. The -RTS signal is set high upon a master reset operation.

Output 1 (-OUT 1), Pin 34: User-designated output that can be set to an active low by programming bit 2 (-OUT 1) of the modem control register to a high level. The -OUT 1 signal is set high upon a master reset operation.

Output 2 (-OUT 2), Pin 31: User-designated output that can be set to an active low by programming bit 3 (-OUT 2) of the modem control register to a high level. The -OUT 2 signal is set high upon a master reset operation.

Chip Select Out (CSOUT), Pin 24: When high, indicates that the chip has been selected by active CS0, CS1, and -CS2 inputs. No data transfer can be initiated until the CSOUT signal is a logical 1.

Driver Disable (DDIS), Pin 23: Goes low whenever the system unit's microprocessor is reading data from the INS8250. A high-level DDIS output can be used to disable an external transceiver (if used between the system unit's microprocessor and the INS8250 on the D7–D0 data bus) at all times, except when the system unit's microprocessor is reading data.

Baud Out (-BAUDOUT), Pin 15: 16 x clock signal for the transmitter section of the INS8250. The clock rate is equal to the main reference oscillator frequency divided by the specified divisor in the baud-rate generator divisor latches. The -BAUDOUT may also be used for the receiver section by typing this output to the RCLK input of the chip.

Interrupt (INTRPT), Pin 30: Goes high whenever any one of the following interrupt types has an active high condition and is enabled through the interrupt enable register: receiver error flag, received data available, transmitter holding register empty, or modem status. The INTRPT signal is reset low upon the appropriate interrupt service or a master reset operation.

Serial Output (SOUT), Pin 11: Composite serial data output to the communications link (peripheral device, modem, or data set). The SOUT signal is set to the marking (logical 1) state upon a master reset operation.

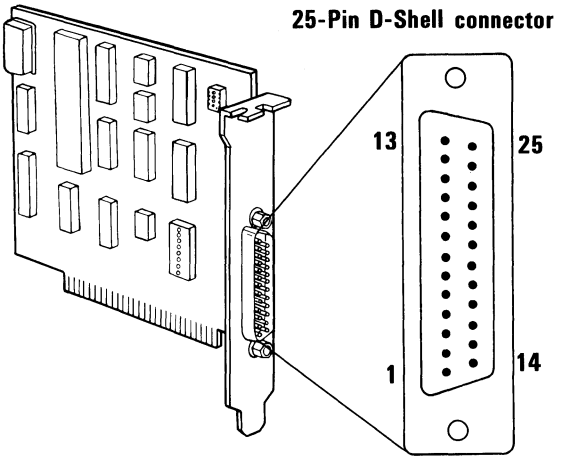
Input/Output Signals

Data Bus (D7–D0), Pins 1–8: This bus comprises eight tri-state input/output lines. The bus provides bidirectional communications between the INS8250 and the system unit's microprocessor. Data, control words, and status information are transferred through the D7-D0 data bus.

External Clock Input/Output (XTAL1, XTAL2), Pins 16 and 17: These two pins connect the main timing reference (crystal or signal clock) to the INS8250.

Specifications

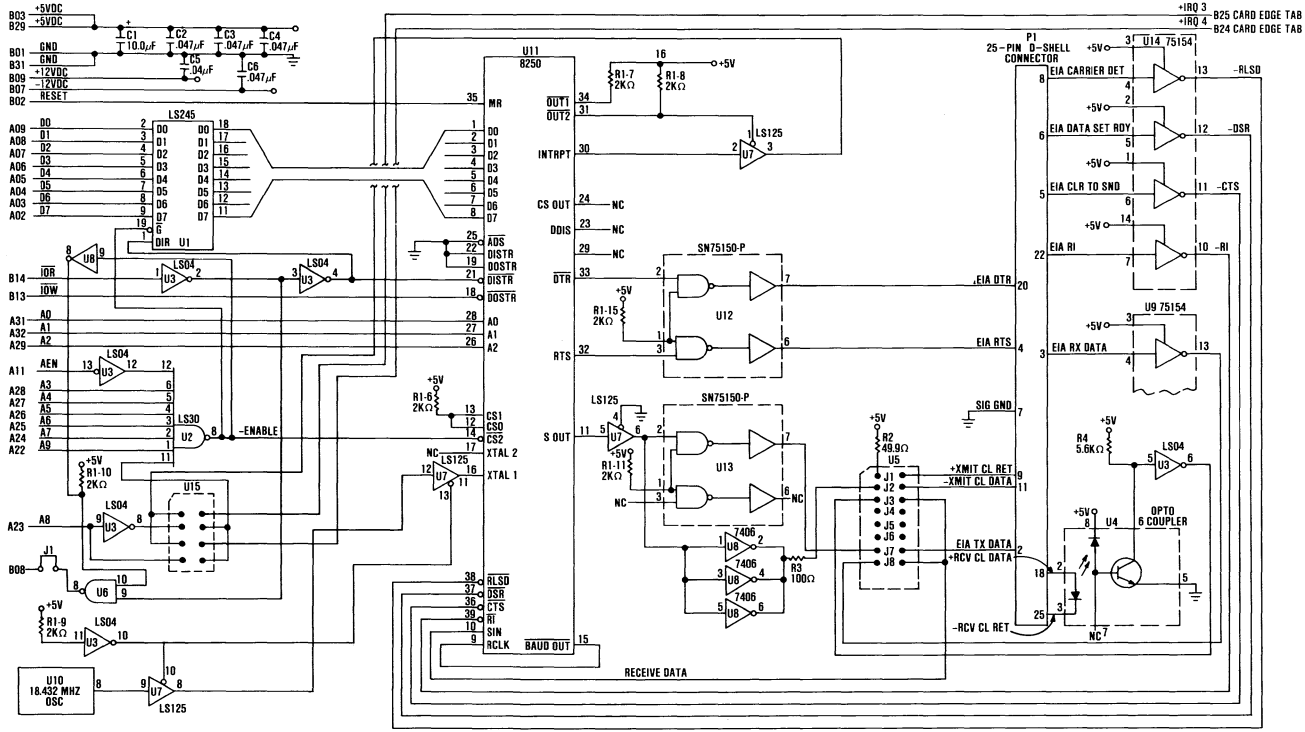
The following page shows the connecter pin assignments and specifications for the Asynchronous Communications Adapter.



	Description	Pin	
	NC	1	
←	Transmitted Data	2	
→	Received Data	3	
←	Request to Send	4	
→	Clear to Send	5	
→	Data Set Ready	6	
	Signal Ground	7	
→	Received Line Signal Detector	8	
←	+ Transmit Current Loop Data	9	
	NC	10	
←	– Transmit Current Loop Data	11	
	NC	12	Asynchronous Communications Adapter (RS-232C)
External Device	NC	13	
	NC	14	
	NC	15	
	NC	16	
	NC	17	
→	+ Receive Current Loop Data	18	
	NC	19	
←	Data Terminal Ready	20	
	NC	21	
→	Ring Indicator	22	
	NC	23	
	NC	24	
→	– Receive Current Loop Return	25	

Note: To avoid inducing voltage surges on interchange circuits, signals from interchange circuits shall not be used to drive inductive devices, such as relay coils.

Connector Specifications



Asynchronous Communications Adapter (Sheet 1 of 1)

Index

A

B

C

D

E

F

I

L

M

O

P

R

S

T

V

Personal Computer Hardware Reference Library

Serial/Parallel Adapter

Contents

Notes:

Description

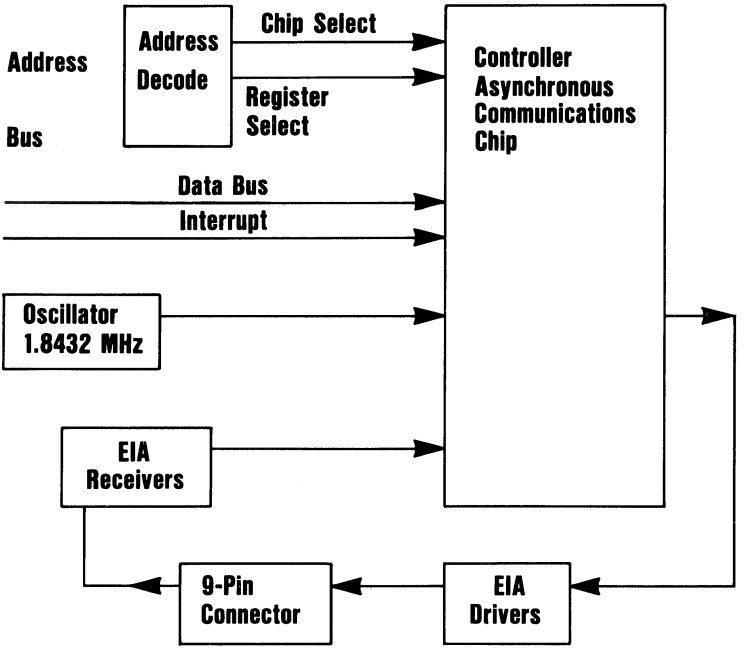
The IBM Personal Computer AT Serial/Parallel Adapter provides a parallel port and a serial port. It plugs into a system-board expansion slot. All system-control signals and voltage requirements are provided through a 2- by 31-position card edge connector.

Serial Portion of the Adapter

The serial portion of the adapter is fully programmable and supports asynchronous communications. It will add and remove start, stop, and parity bits. A programmable baud-rate generator allows operation from 50 baud to 9600 baud. Five-, six-, seven- and eight-bit characters with 1, 1.5, or 2 stop bits are supported. A prioritized interrupt system controls transmit, receive, error, and line status as well as data-set interrupts.

The rear of the adapter has a 9-pin D-shell connector that is classified as an RS-232C port. When the optional IBM Communications Cable (9-Pin), which has a 9-pin D-shell connector on one end and a 25-pin D-shell connector on the other end, is connected to the adapter, the 25-pin end of the cable has all the signals of a standard EIA RS-232C interface.

The following figure is a block diagram of the serial portion of the adapter.



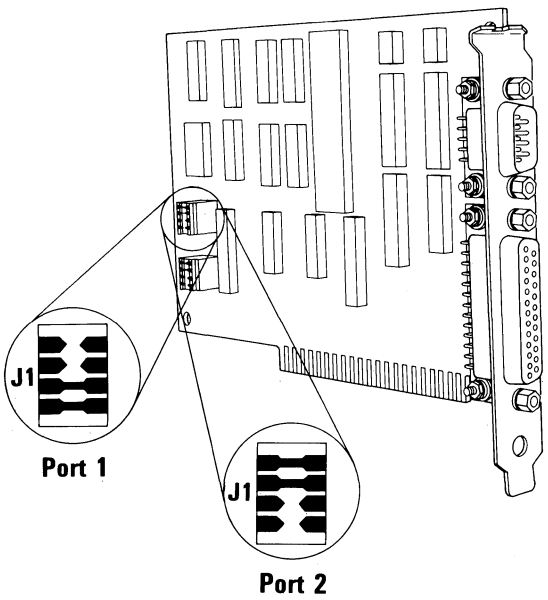
Serial Portion Block Diagram

The serial portion of the adapter has a controller that provides the following functions:

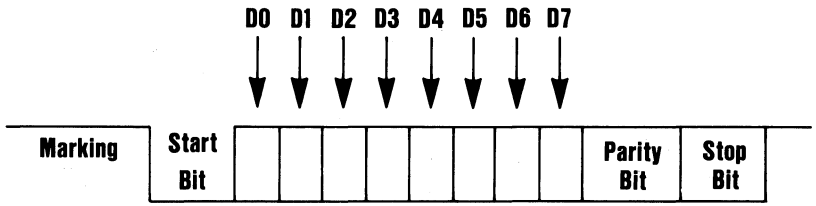
- Adds or deletes standard, asynchronous-communications bits to or from a serial data stream.
- Provides full, double buffering, which eliminates the need for precise synchronization.
- Provides a programmable baud-rate generator.
- Provides modem controls (CTS, RTS, DSR, DTR, RI, and CD).

Communications Application

The serial output port may be addressed as either communications port 1 or communications port 2 as defined by jumper J1 (see the following figure). In this section hex addresses begin with an X which can be either a 3 for communications port 1 (interrupt level 4) or a 2 for communications port 2 (interrupt level 3).



The data format will be as follows:



Bit 0 is the first data 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 bit (1, 1.5, or 2, depending on the command in the line-control register).

Controller Specifications

The following describes the function of controller input/output signals.

Input Signals

-Clear to Send: (-CTS), Pin 36—The '-CTS' signal is a modem-control function input, the condition of which can be tested by the processor by reading bit 4 (-CTS) of the modem status register. Bit 0 (DCTS) of the modem status register indicates if the '-CTS' input has changed state since the previous reading.

> **Note:** Whenever the CTS bit of the modem status register changes state, an interrupt is generated if the modem-status interrupt is enabled.

-Data Set Ready: (-DSR), Pin 37—When low, indicates the modem or data set is ready to establish the communications link and transfer data with the controller. The '-DSR' signal is a modem-control function input, the condition of which can be tested by the processor reading bit 5 (-DSR) of the modem status register. Bit 1 (DDSR) of the modem status register indicates if the '-DSR' input has changed since the previous reading.

> **Note:** Whenever the DSR bit of the modem status register changes state, an interrupt is generated if the modem-status interrupt is enabled.

-Data Carrier Detect: (-DCD), Pin 38—When low, indicates the modem or data set detected a data carrier. The '-DCD' signal is a modem-control function input, the condition of which can be tested by the processor reading bit 7 (-DCD) of the modem status register. Bit 3 (DDCD) of the modem status register indicates if the '-DCD' input has changed state since the previous reading.

> **Note:** Whenever the DCD bit of the modem status register changes state, an interrupt is generated if the modem status interrupt is enabled.

-Ring Indicator: (-RI), Pin 39—When low, indicates the modem or data set detected a telephone ringing signal. The '-RI' signal is a modem-control function input, the condition of which can be tested by the processor reading bit 6 (-RI) of the modem status register. Bit 2 (TERI) of the modem status register indicates if the '-RI' input has changed from an active to an inactive state since the previous reading.

> **Note:** Whenever the RI bit of the modem status register changes from an inactive to an active state, an interrupt is generated if the modem-status interrupt is enabled.

VCC Pin 40—+5 Vdc supply

VSS Pin 20—Ground (0 Vdc) reference

Output Signals

-Data Terminal Ready: (-DTR), Pin 33—When active, informs the modem or data set that the controller is ready to communicate. The '-DTR' output signal can be set to an active level by programming bit 0 (-DTR) of the modem control register to an active level. The '-DTR' signal is set inactive upon a master reset operation.

-Request to Send: (-RTS), Pin 32—When active, informs the modem or data set that the controller is ready to send data. The '-RTS' output signal can be set to an active level by programming bit 1 (-RTS) of the modem control register to an active level. The '-RTS' signal is set inactive upon a master reset operation.

-Output 1: (-OUT 1), Pin 34—User-designated output that can be set to an active level by programming bit 2 (-OUT 1) of the modem control register to an inactive level. The '-OUT 1' signal is set inactive upon a master reset operation. Pin 34 is connected to an active source.

-Output 2: (-OUT 2), Pin 31—User-designated output that can be set to an active level by programming bit 3 (-OUT 2) of the modem control register to an inactive level. The '-OUT 2' signal is set inactive upon a master reset operation. Pin 31 controls interrupts to the system.

Controller-Accessible Registers

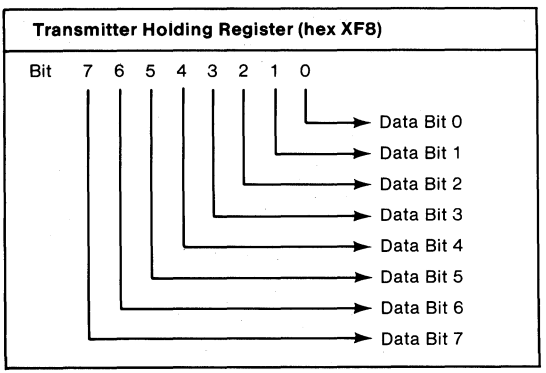
The controller has a number of accessible registers. The system programmer may gain access to or control any of the controller registers through the microprocessor. These registers are used to control the controller's operations and to transmit and receive data. The X in the register address determines the the port selected; 3 is for port 1 and 2 is for port 2.

Specific registers are selected according to the following figure:

I/O Address	Register Selected	DLAB State
XF8	TX Buffer	0 (Write)
XF8	RX Buffer	0 (Read)
XF8	Divisor Latch LSB	1
XF9	Divisor Latch MSB	1
XF9	Interrupt Enable Register	0
XFA	Interrupt Identification Register	
XFB	Line Control Register	
XFC	Modem Control Register	
XFD	Line Status Register	
XFE	Modem Status Register	
XFF	Reserved	

Controller-Accessible Registers

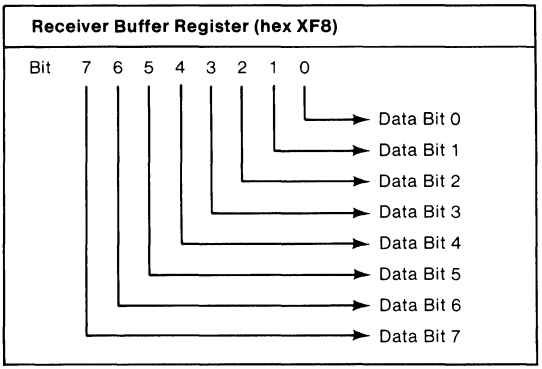
Transmitter Holding Register (hex XF8): The transmitter holding register (THR) contains the character to be sent.



Transmitter Holding Register

Bit 0 is the least-significant bit and the first bit sent serially.

Receiver Buffer Register (hex XF8): The receiver buffer register (RBR) contains the received character.

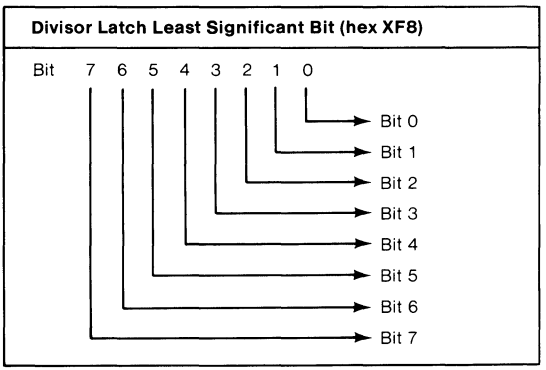


Receiver Buffer Register

Bit 0 is the least-significant bit and the first bit received serially.

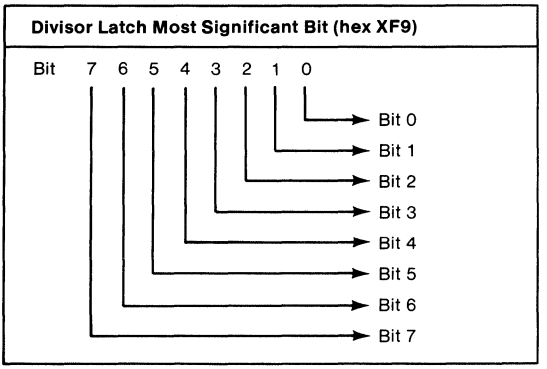
Programmable Baud-Rate Generator: The controller has a programmable baud-rate generator that can divide the clock input (1.8432 MHz) by any divisor from 1 to 655,535 or 2^{16}-1. The output frequency of the baud-rate generator is the baud rate multiplied by 16. Two 8-bit latches store the divisor in 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 prevents long counts on the first load.

Divisor Latch LSB (hex XF8)



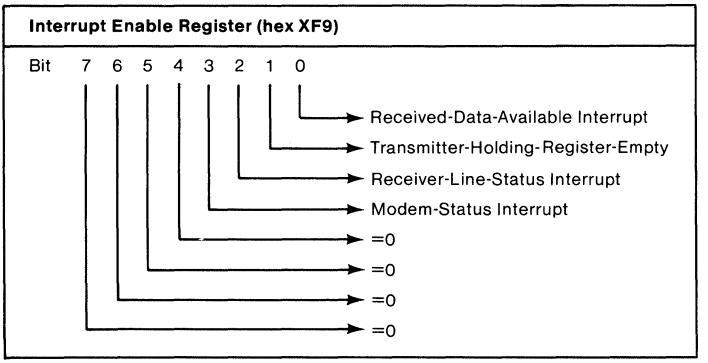
Divisor Latch Least Significant Bit

Divisor Latch MSB (hex XF9)



Divisor Latch Most Significant Bit

Interrupt Enable Register (hex XF9): This 8-bit register allows the four types of controller interrupts to separately activate the 'chip-interrupt' (INTRPT) output signal. The interrupt system can be totally disabled by resetting bits 0 through 3 of the interrupt enable register (IER). Similarly, by setting the appropriate bits of this register to logical 1, selected interrupts can be enabled. Disabling the interrupt system inhibits the 'IER' and the active 'INTRPT' output from the chip. All other system functions operate normally, including the setting of the line-status and modem-status registers.



Interrupt Enable Register

Bit 0 When set to logical 1, enables the received-data-available interrupt.

Bit 1 When set to logical 1, enables the transmitter-holding-register-empty interrupt.

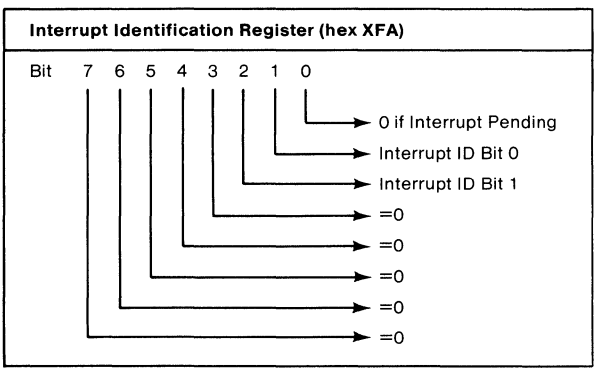
Bit 2 When set to logical 1, enables the receiver-line-status interrupt.

Bit 3 When set to logical 1, enables the modem-status interrupt.

Bits 4–7 These four bits are always logical 0.

Interrupt Identification Register (hex XFA): The controller has an on-chip interrupt capability that makes communications possible with all of the currently popular microprocessors. In order to minimize programming overhead during data character transfers, the controller prioritizes interrupts into four levels: receiver-line-status (priority 1), received-data-available (priority 2), transmitter-holding-register-empty (priority 3), and modem status (priority 4).

Information about a pending prioritized interrupt is stored in the interrupt identification register (IIR). (See the figure "Interrupt Control Functions," later.) The IIR, when addressed during chip-select time, stops the pending interrupt with the highest priority, no other interrupts are acknowledged until the processor services that interrupt.



Interrupt Identification Register

Bit 0 This bit can be used in either hard-wired, prioritized, or polled conditions to indicate if an interrupt is pending. When bit 0 is logical 0, an interrupt is pending, and the IIR contents may be used as a

pointer to the appropriate interrupt service routine. When bit 0 is logical 1, no interrupt is pending, and polling (if used) continues.

Bits 1–2 These two bits identify the pending interrupt that has the highest priority, as shown in the following figure:

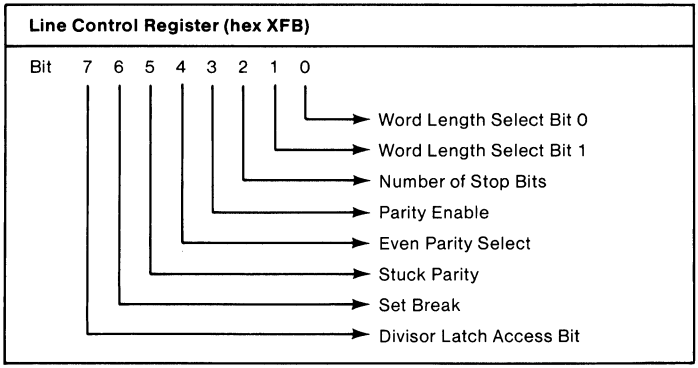
Interrupt ID Register			Interrupt Set And Reset Functions			
Bit 2	**Bit 1**	**Bit 0**	**Priority Level**	**Interrupt Type**	**Interrupt Source**	**Interrupt Reset Control**
0	0	0	-	None	None	-
1	1	0	Highest	Receiver Line Status	Overrun Error or Parity Error or Framing Error or Break Interrupt	Reading the Line Status Register
1	0	0	Second	Received Data Available	Receiver Data Available	Reading the Receiver Buffer Register
0	1	0	Third	Transmitter Holding Register Empty	Transmitter Holding Register Empty	Reading the IIR (if source of interrupt) or writing into the THR
0	0	0	Fourth	Modem Status	Clear to Send or Data Set Ready or Ring Indicator or Received Line Signal Detect	Reading the Modem Status Register

Interrupt Priority

Bits 3–7 These five bits are always logical 0.

Line-Control Register (hex XFB): The system programmer specifies the format of the asynchronous data communications exchange through the line control register. In addition to

controlling the format, the programmer may retrieve the contents of the line control register for inspection. This feature simplifies system programming and eliminates the need to store line characteristics separately in system memory.



Line Control Register

Bits 0, 1 These two bits specify the number of bits in each serial character that is sent or received. The encoding of bits 0 and 1 is as follows:

Bit 0	Bit 1	Word Length (Bits)
0	0	5
0	1	6
1	0	7
1	1	8

Word Length

Bit 2 This bit specifies the number of stop bits in each serial character that is sent or received. If bit 2 is a logical 0, one stop bit is generated or checked in the data sent or received. If bit 2 is logical 1 when a 5-bit word length is selected through bits 0 and 1, 1-1/2 stop bits are generated or checked. If bit 2 is logical 1 when either a 6-, 7-, or 8-bit word length is selected, two stop bits are generated or checked.

Bit 3 This bit is the parity-enable bit. When bit 3 is logical 1, a parity bit is generated (transmit data) or checked (receive data) between the last data word

and stop bit of the serial data. (The parity bit is used to produce an even or odd number of 1's when the data-word bits and parity bit are summed.)

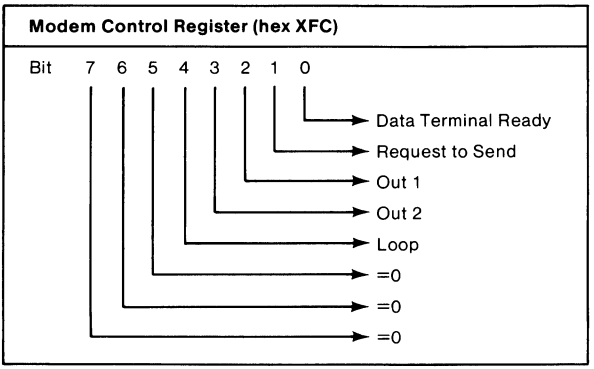
Bit 4 This bit is the even-parity-select bit. When bit 3 is a logical 1 and bit 4 is a logical 0, an odd number of logical 1's are sent or checked in the data word bits and parity bit. When both bit 3 and bit 4 are a logical 1, an even number of bits are sent or checked.

Bit 5 This bit is the stuck-parity bit. When bit 3 is a logical 1 and bit 5 is a logical 1, the parity bit is sent and then detected by the receiver as a logical 0, if bit 4 is a logical 1, or as a logical 1 if bit 4 is a logical 0.

Bit 6 This bit is the set-break control bit. When bit 6 is set to a logical 1, the serial output (SOUT) is forced to the spacing (logical 0) state and remains there regardless of other transmitter activity. The set-break is disabled by setting bit 6 to logical 0. This feature enables the microprocessor to select a specific terminal in a computer communications system.

Bit 7 This bit is the divisor-latch access bit (DLAB). It must be set high (logical 1) to gain access to the divisor latches of the baud-rate generator during a read or write operation. It must be set low (logical 0) to gain access to the receiver buffer, the transmitter holding register, or the interrupt enable register.

Modem Control Register (hex XFC): This 8-bit register controls the data exchange with the modem or data set (an external device acting as a modem).



Modem Control Register

Bit 0 This bit controls the '-data terminal ready' (-DTR) output. When bit 0 is set to logical 1, the -DTR output is forced active. When bit 0 is reset to logical 0, the '-DTR' output is forced inactive.

Bit 1 This bit controls the '-request-to-send' (-RTS) output. Bit 1 affects the '-RTS' output in the same way bit 0 affects the '-DTR' output.

Bit 2 This bit controls the '-Output 1' (-OUT 1) signal, which is a spare the programmer can use. Bit 2 affects the '-OUT 1' output in the same way bit 0 affects the '-DTR' output.

Bit 3 This bit controls the '-Output 2' (-OUT 2) signal, which is a spare the programmer can use. Bit 3 affects the '-OUT 2' output in the same way bit 0 affects the '-DTR' output.

Bit 4 This bit provides a loopback feature for diagnostic testing of the controller. When bit 4 is set to logical 1, the following occur: the 'transmitter serial output' (SOUT) is set to the active state; the 'receiver serial input' (SIN) is disconnected; the output of the transmitter shift register is "looped back" to the receiver shift register input; the four modem-control inputs ('-CTS', '-DSR', '-RLSD', and '-RI') are

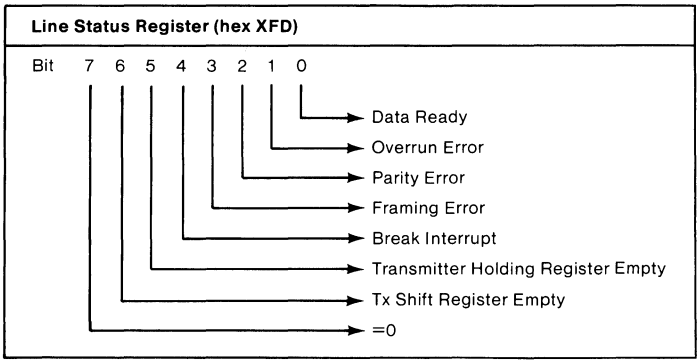
disconnected; and the four modem-control outputs ('-DTR', '-RTS', '-OUT 1' and '-OUT 2') are internally connected to the four modem control inputs. In the diagnostic mode, data sent is immediately received. This feature allows the processor to verify the transmit- and receive-data paths of the controller.

In the diagnostic mode, the receiver and transmitter interrupts are fully operational, as are the modem-control interrupts. But the interrupts' sources are now the lower four bits of the modem control register (MCR) instead of the four modem-control inputs. The interrupts are still controlled by the interrupt enable register.

The controller's interrupt system can be tested by writing to the lower six bits of the line status register and the lower four bits of the modem status register. Setting any of these bits to logical 1 generates the appropriate interrupt (if enabled). Resetting these interrupts is the same as for normal controller operation. To return to normal operation, the registers must be reprogrammed for normal operation, and then bit 4 of the MCR must be reset to logical 0.

Bits 5–7 These bits are permanently set to logical 0.

Line Status Register (hex XFD): This 8-bit register provides the processor with status information about the data transfer.



Line Status Register

Bit 0 This bit is the receiver data ready (DR) indicator. It is set to logical 1 whenever a complete incoming character has been received and transferred into the receiver buffer register. Bit 0 may be reset to logical 0 by reading the data in the receiver buffer register.

Bit 1 This bit is the overrun error (OE) indicator. It indicates that data in the receiver's buffer register was not read by the processor before the next character was transferred into the register, thereby destroying the previous character. The OE indicator is reset whenever the processor reads the contents of the line status register.

Bit 2 This bit is the parity error (PE) indicator and indicates the received data character does not have the correct even or odd parity, as selected by the even-parity-select bit. The PE bit is set to logical 1 upon detection of a parity error, and is reset to logical 0 whenever the processor reads the contents of the line status register.

Bit 3 This bit is the framing error (FE) indicator. It indicates the received character did not have a valid stop bit. Bit 3 is set to logical 1 whenever the stop bit following the last data bit or parity bit is detected as a zero bit (spacing level).

Bit 4 This bit is the break interrupt (BI) indicator. It is set to logical 1 whenever the received data input is held in the spacing state (logical 0) for longer than a full word transmission time (that is, the total time of start bit + data bits + parity bits + stop bits).

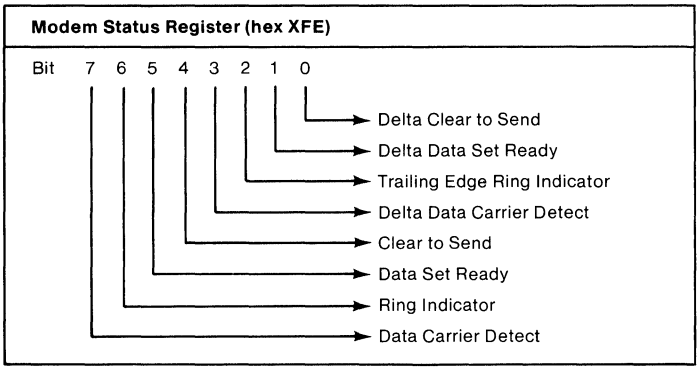
> **Note:** Bits 1 through 4 are error conditions that produce a receiver line-status interrupt whenever any of the corresponding conditions are detected.

Bit 5 This bit is the transmitter holding register empty (THRE) indicator. It indicates the controller is ready to accept a new character for transmission. In addition, this bit causes the controller to issue an interrupt to the processor when the THRE interrupt enable is set active. The THRE bit is set to logical 1 when a character is transferred from the transmitter holding register into the transmitter shift register. It is reset to logical 0 when the processor loads the transmitter holding register.

Bit 6 This bit is the transmitter empty (TEMT) indicator. It is set to logical 1 whenever the transmitter holding register (THR) and the transmitter shift register (TSR) are both empty. It is reset to logical 0 if THR or TSR contain a data character.

Bit 7 This bit is permanently set to logical 0.

Modem Status Register (hex XFE): The 8-bit MSR provides the current state of the control lines from the modem (or external device) to the processor. In addition, four bits of the MSR provide change information. These four bits are set to logical 1 whenever a control input from the modem changes state. They are reset to logical 0 whenever the processor reads this register.



Modem Status Register

Bit 0 This bit is the delta clear-to-send (DCTS) indicator. It indicates the '-CTS' input to the chip has changed state since the last time it was read by the processor.

Bit 1 This bit is the delta data-set-ready (DDSR) indicator. It indicates the '-DSR' input to the chip has changed state since the last time it was read by the processor.

Bit 2 This bit is the trailing-edge ring-indicator (TERI) detector. It indicates the '-RI' input to the chip has changed from an active condition to an inactive condition.

Bit 3 This bit is the delta data-carrier-detect (DDCD) indicator. It indicates the '-DCD' input to the chip has changed state.

> **Note:** Whenever bit 0, 1, 2, or 3 is set to a logical 1, a modem status interrupt is generated.

Bit 4 This bit is the opposite of the '-clear-to-send' (-CTS) input. If bit 4 of the MCR loop is set to a logical 1, this bit is equivalent to RTS of the MCR.

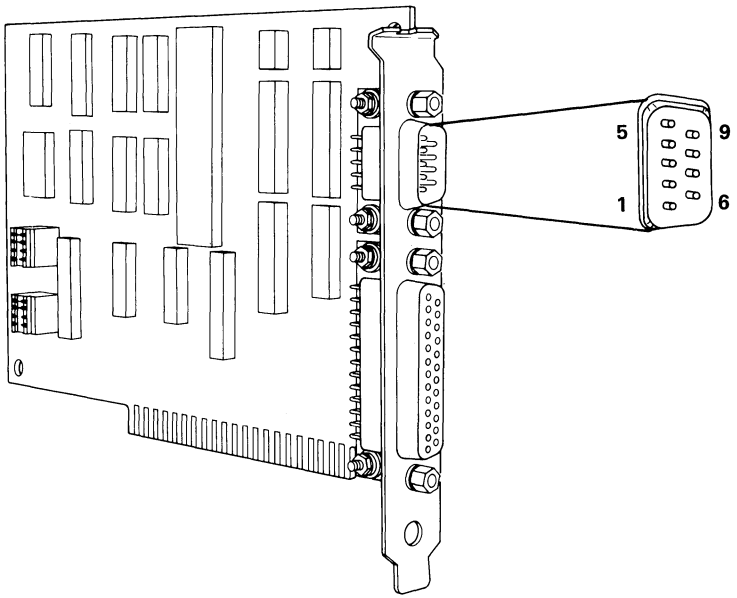
Bit 5 This bit is the opposite of the '-data-set-ready' (-DSR) input. If bit 4 of the MCR is set to a logical 1, this bit is equivalent to DTR of the MCR.

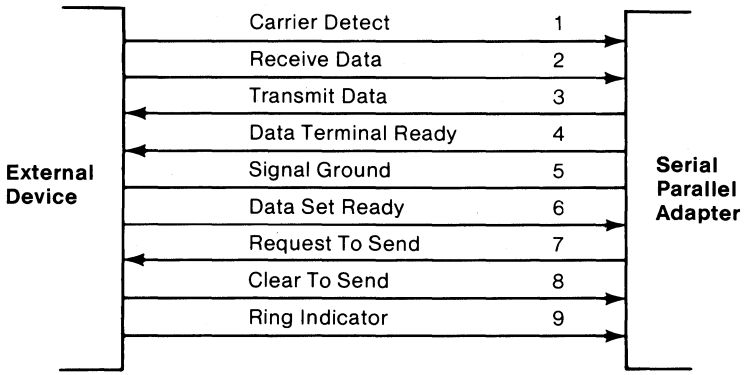
Bit 6 This bit is the opposite of the '-ring-indicator' (-RI) input. If bit 4 of the MCR is set to a logical 1, this bit is equivalent to OUT 1 of the MCR.

Bit 7 This bit is the opposite of the '-data-carrier-detect' (-DCD) input. If bit 4 of the MCR is set to a logical 1, this bit is equivalent to OUT 2 of the MCR.

Pin Assignment for Serial Port

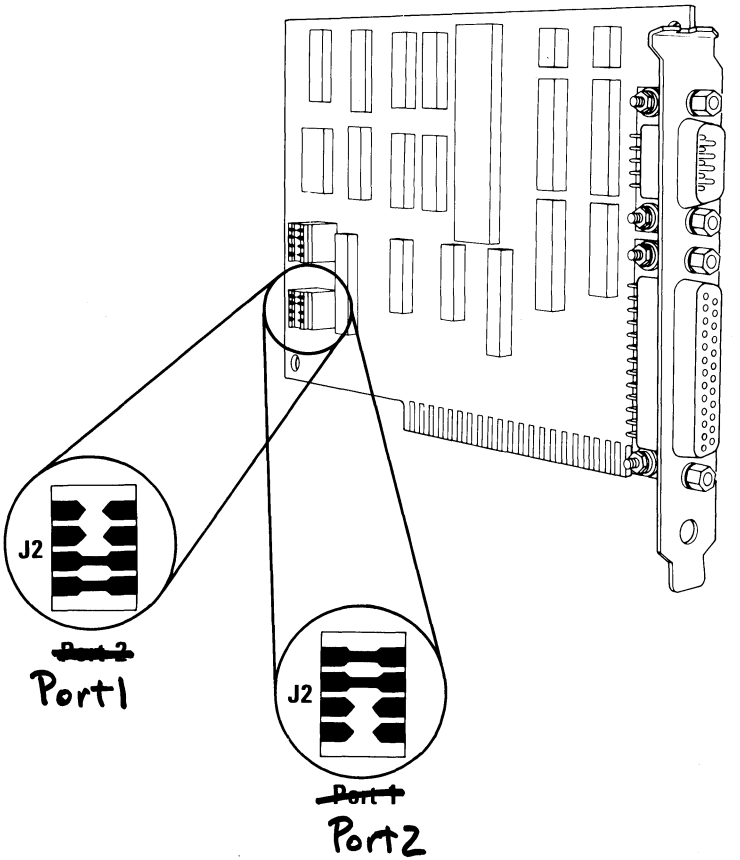
The following figure shows the pin assignments for the serial port in a communications environment.



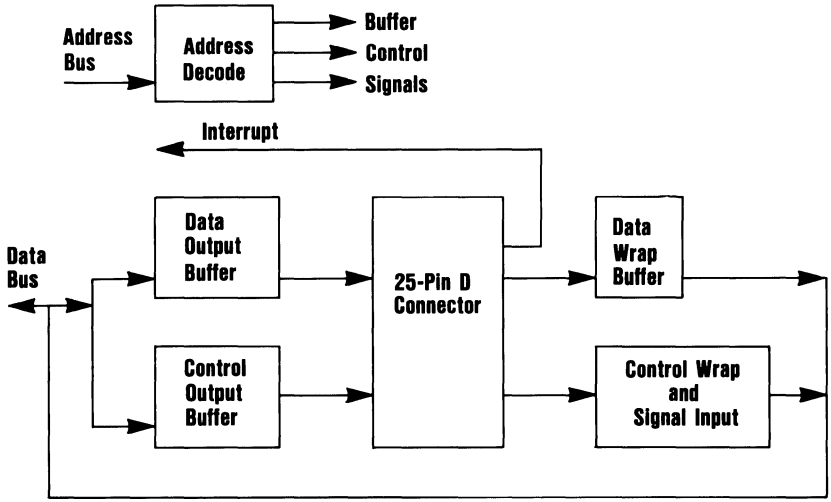


Parallel Portion of the Adapter

The parallel portion of the adapter makes possible the attachment of various devices that accept eight bits of parallel data at standard TTL levels. The rear of the adapter has a 25-pin, D-shell connector. This port may be addressed as either parallel port 1 or 2. The port address is determined by the position of jumper J2, as shown in the following figure.



The following figure is a block diagram of the parallel portion of the adapter.



Parallel Portion Block Diagram

Printer Application

The following discusses the use of the parallel portion of the adapter to connect to a parallel printer. Hexadecimal addresses in this section begin with an X, which is replaced with a 3 to indicate port 1, or a 2 to indicate port 2.

Data Latch (hex X78, X7C)

Writing to this address causes data to be stored in the printer's data buffer. Reading this address sends the contents of the printer's data buffer to the system microprocessor.

Printer Controls (hex X7A, X7E)

Printer control signals are stored at this address to be read by the system microprocessor. The following are bit definitions for this byte.

Bit 7 Not used

Bit 6 Not used

Bit 5 Not used

Bit 4 +IRQ Enable—A logical 1 in this position allows an interrupt to occur when '-ACK' changes from active to inactive.

Bit 3 +SLCT IN—A logical 1 in this bit position selects the printer.

Bit 2 -INIT—A logical 0 starts the printer (50-microsecond pulse, minimum).

Bit 1 +AUTO FD XT—A logical 1 causes the printer to line-feed after a line is printed.

Bit 0 +STROBE—A 0.5-microsecond minimum, high, active pulse clocks data into the printer. Valid data must be present for a minimum of 0.5 microsecond before and after the strobe pulse.

Printer Status - (hex X79, X7D)

Printer status is stored at this address to be read by the microprocessor. The following are bit definitions for this byte.

Bit 7 -BUSY—When this signal is active, the printer is busy and cannot accept data. It may become active during data entry, while the printer is offline, during printing, when the print head is changing positions, or while in an error state.

Bit 6 -ACK—This bit represents the current state of the printer's '-ACK' signal. A 0 means the printer has received the character and is ready to accept another. Normally, this signal will be active for approximately 5 microseconds before '-BUSY' stops.

Bit 5 +PE—A logical 1 means the printer has detected the end of paper.

Bit 4 +SLCT—A logical 1 means the printer is selected.

Bit 3 -Error—A logical 0 means the printer has encountered an error condition.

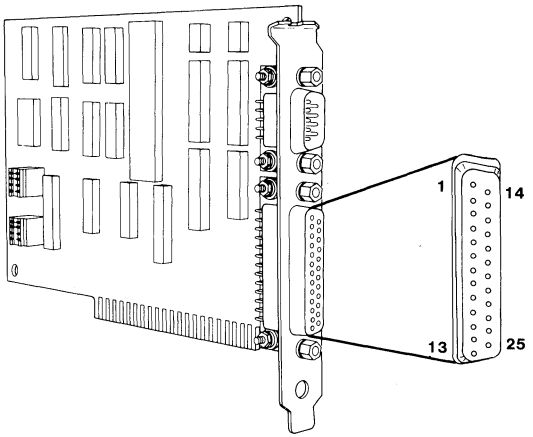
Bit 2 Unused.

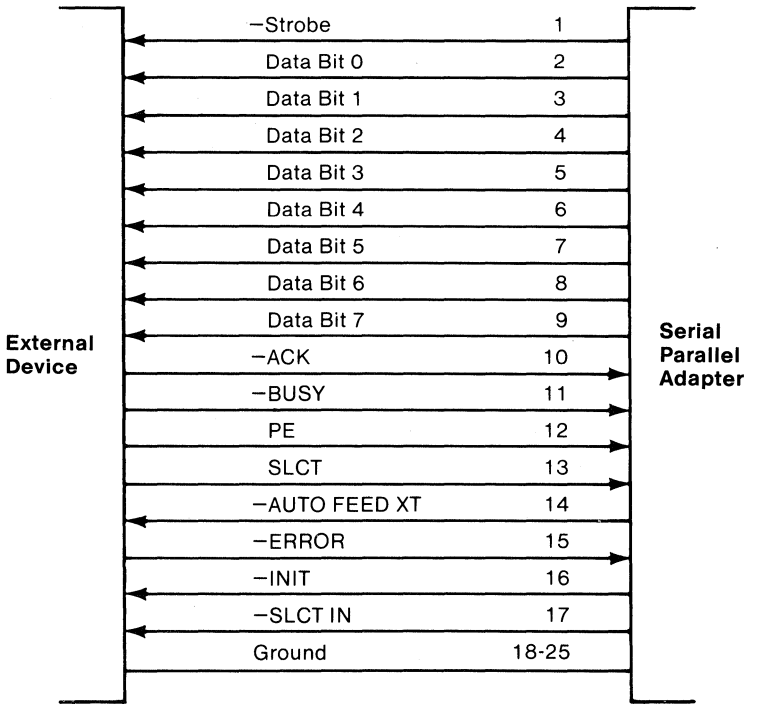
Bit 1 Unused.

Bit 0 Unused.

Parallel Interface

The adapter has a 25-pin, D-shell connector at the rear of the adapter. The following figure shows the signals and their pin assignments. Typical printer input signals also are shown.





Specifications

The following figures list characteristics of the output driver.

Sink current	24 mA	Max
Source Current	-2.6 mA	Max
High-Level Output Voltage	2.4 Vdc	Min
Low-Level Output Voltage	0.5 Vdc	Max

Parallel Data and Processor IRQ

Sink Current	16 mA	Max
Source Current	0.55 mA	Max
High Level Output Voltage	5 Vdc	Minus Pull-Up
Low Level Output Voltage	0.4 Vdc	Max

Parallel Control

Sink Current	24 mA	Max
Source Current	-15 mA	Max
High Level Output Voltage	2.0 Vdc	Min
Low Level Output Voltage	0.5 Vdc	Max

Parallel Processor Interface (Except IRQ)

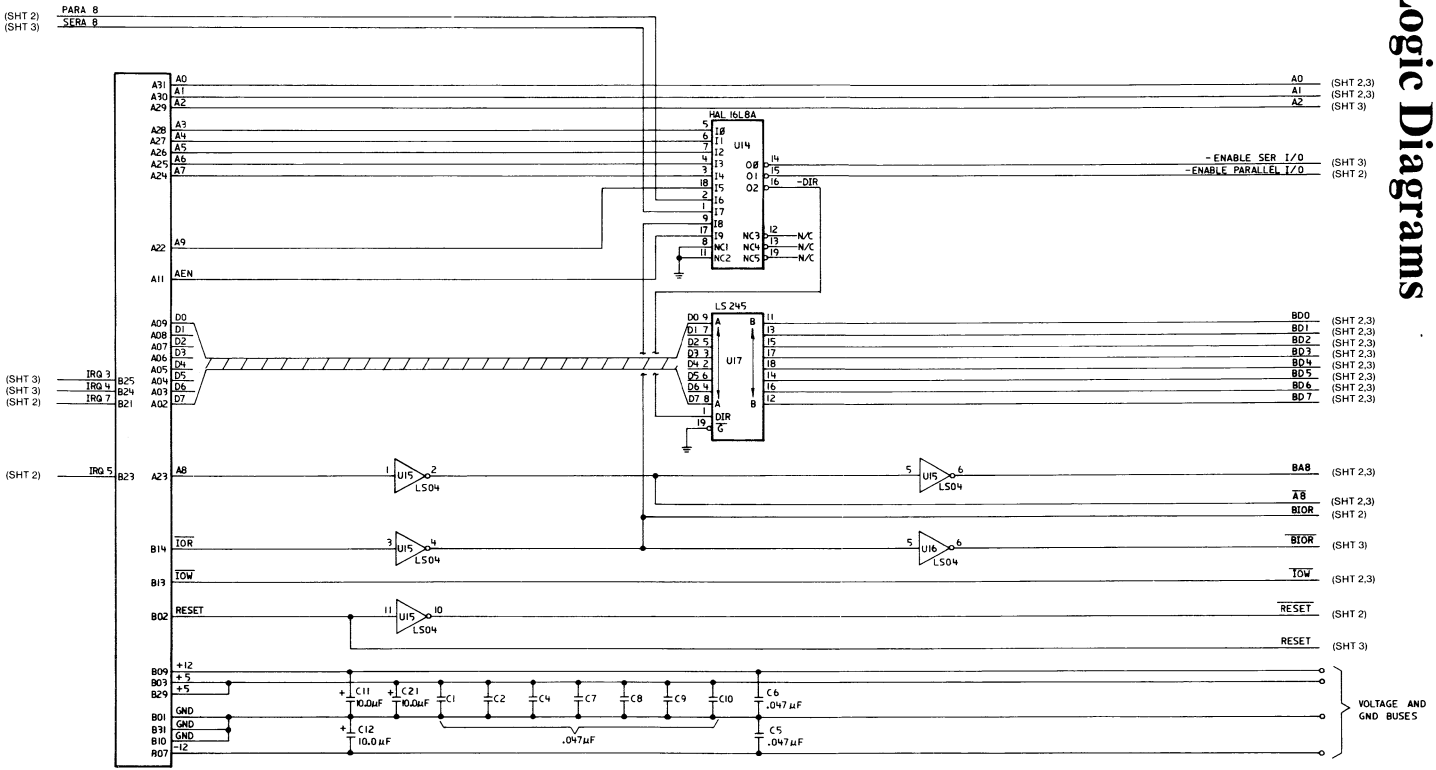
The following are the specifications for the serial interface.

Function	Condition
On	Spacing condition (binary 0, positive voltage).
Off	Marking condition (binary 1, negative voltage).

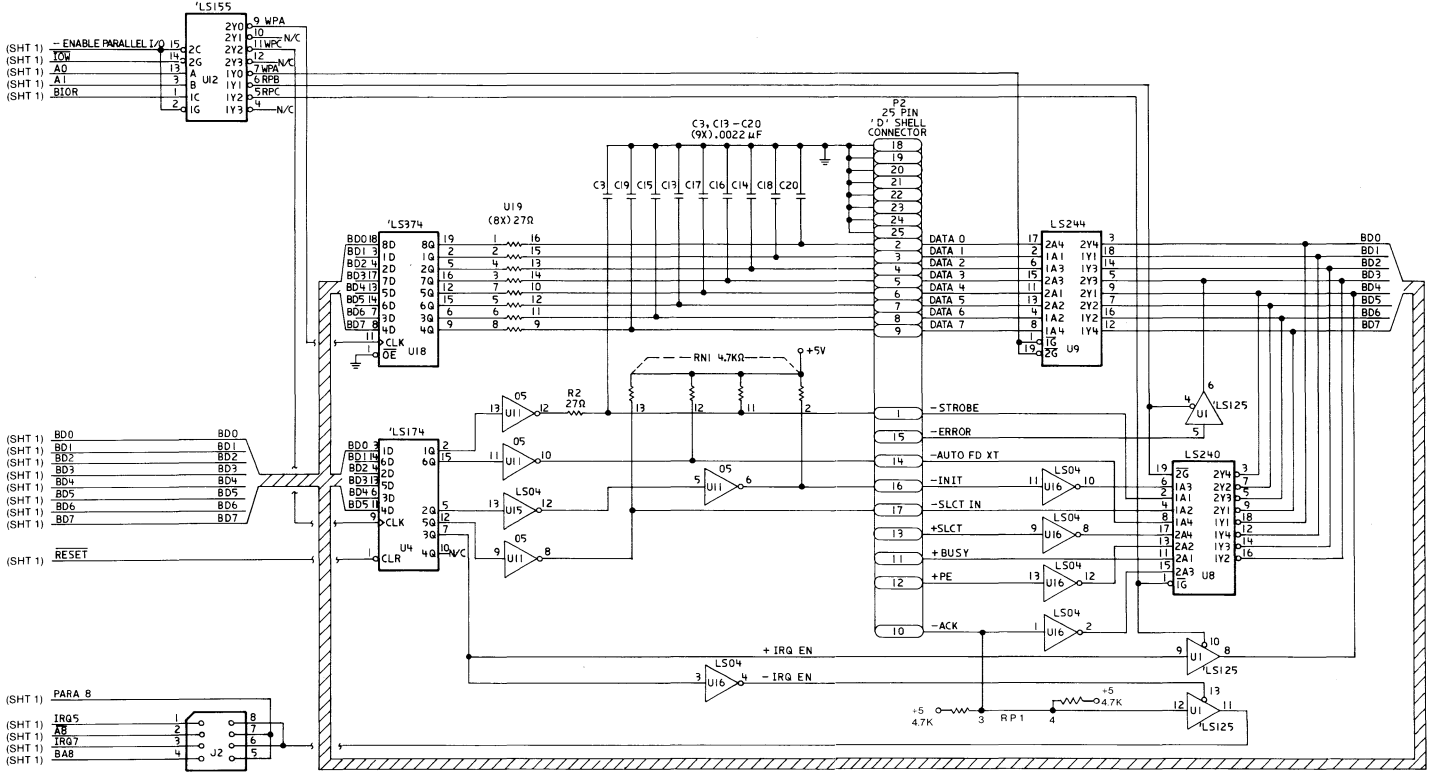
Voltage	Function
Above +15 Vdc	Invalid
+3 Vdc to +15 Vdc	On
-3 Vdc to +3 Vdc	Invalid
-3 Vdc to -15 Vdc	Off
Below -15 Vdc	Invalid

Serial Port Functions

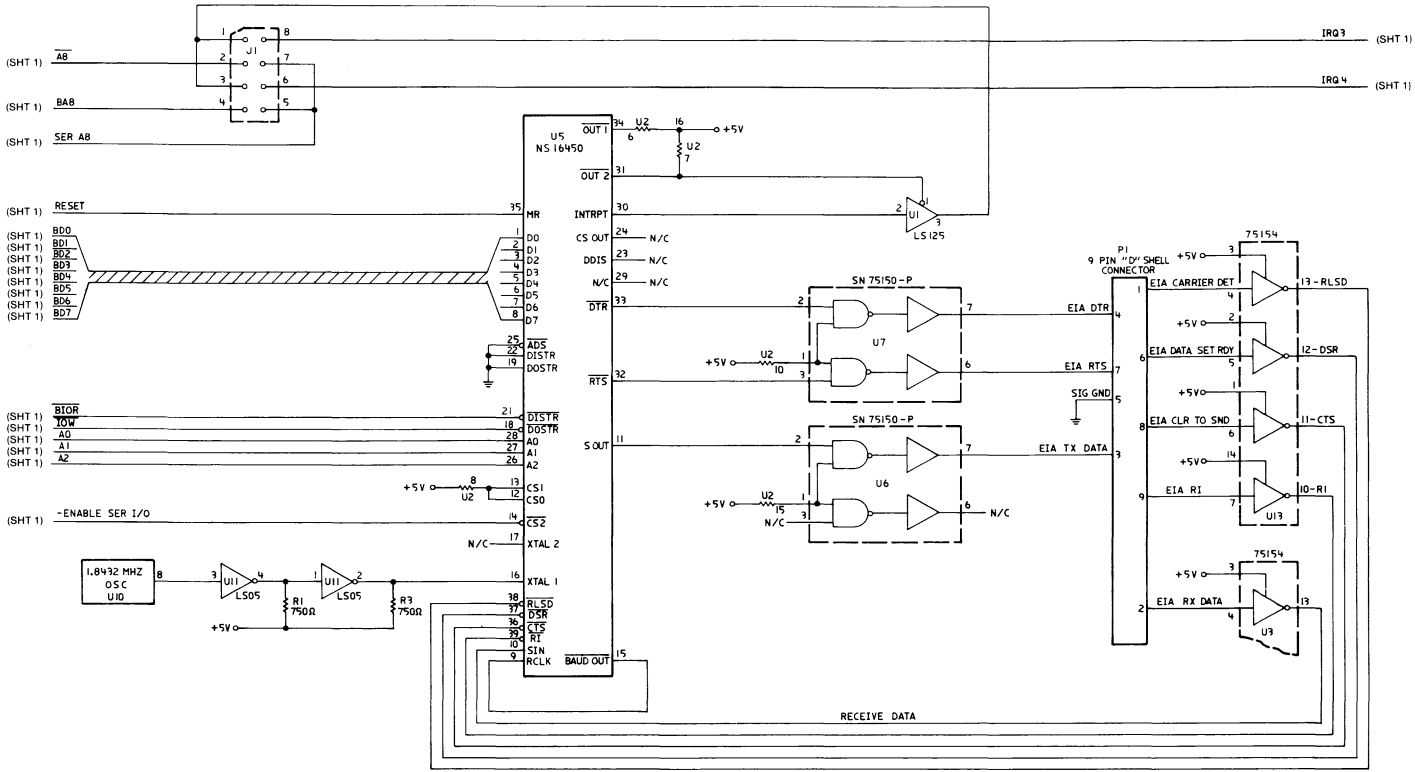
Notes:



Serial/Printer Adapter (Sheet 1 of 3)



Serial/Printer Adapter (Sheet 2 of 3)



Serial/Printer Adapter (Sheet 3 of 3)

Notes:

Personal Computer Hardware Reference Library

IBM Binary Synchronous Communications Adapter

6361499

IBM Binary Synchronous Communications Adapter

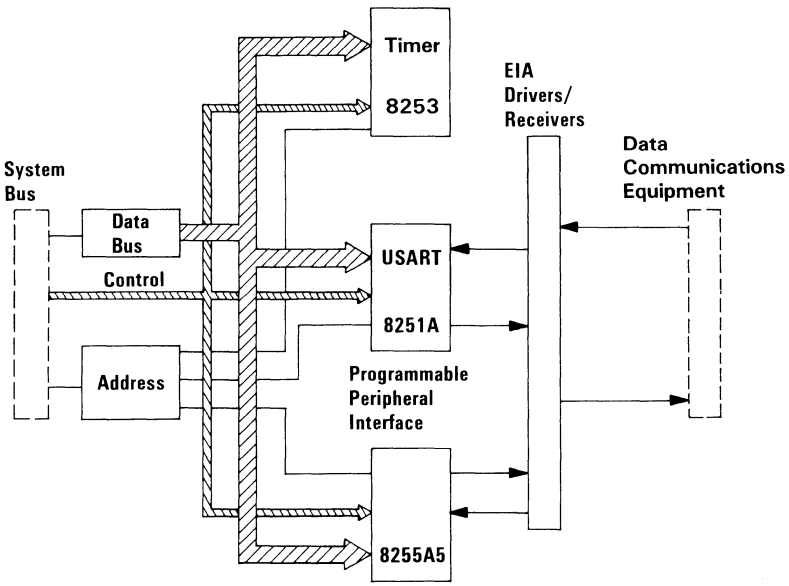
Contents

Description

The IBM Binary Synchronous Communications (BSC) Adapter provides an RS-232C-compatible communications interface for the IBM Personal Computer family of products. All system control, voltage, and data signals are provided through a 2- by 31-position card-edge connector. External interface is in the form of Electronic Industries Association (EIA) drivers and receivers connected to an RS-232C, standard 25-pin, D-shell connector.

The adapter is programmed to operate in a binary synchronous mode. Maximum transmission rate is 9600 bits per second (bps). The main feature of the adapter is an Intel 8251A Universal Synchronous/Asynchronous Receiver/Transmitter (USART). An Intel 8255A-5 Programmable Peripheral Interface (PPI) also is used for expanded modem operation, and an Intel 8253-5 Programmable Interval Timer provides time-outs and generates interrupts.

The following is a block diagram of the BSC adapter.



BSC Adapter Block Diagram

2 BSC Adapter

Programming Considerations

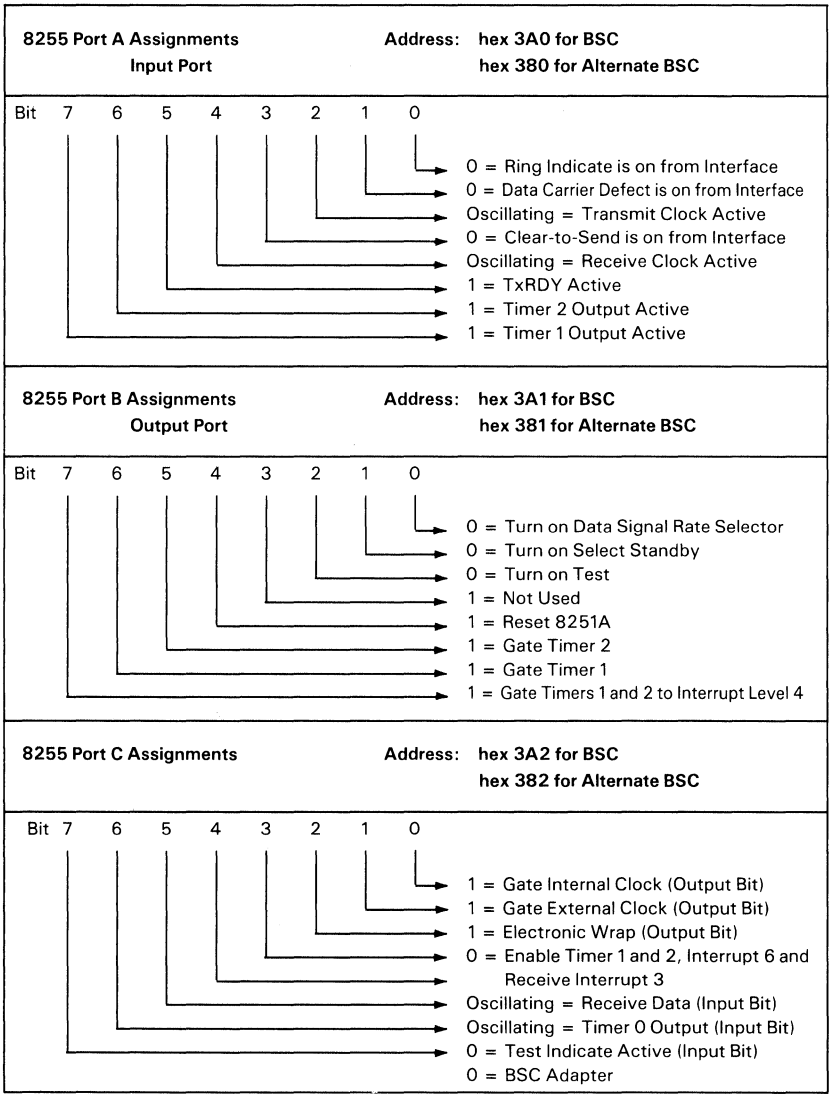
Before starting data transmission or reception, the system unit programs the BSC adapter to define control and gating ports, timer functions and counts, and the communications environment.

Typical Programming Sequence

The 8255A-5 Programmable Peripheral Interface (PPI) is set for the proper mode by selecting address hex 3A3 and writing the control word. This defines port A as an input, port B as an output for modem control and gating, and port C for 4-bit input and 4-bit output. An output to port C sets the adapter to the wrap mode, disallows interrupts, and gates external clocks (address = hex 3A2, data = hex 0D). The adapter is now isolated from the communication interface, and setup continues.

Bit 4 of the PPI's port B brings the USART reset pin high, holds it, then drops it. This resets the internal registers of the USART.

The PPI's port assignments are as follows:



The USART uses the 8253-5 Programmable Interval Timer (PIT) in the synchronous mode for inactivity time-outs to interrupt the system unit after a preselected amount of time has elapsed from the start of a communication operation. Counter 0 is not used for synchronous operation. Counters 1 and 2 connect to

interrupt-level 4 and, being programmed to terminal-count values, provide the desired time delay before generating a level-4 interrupt. These interrupts signal the system that a predetermined amount of time has elapsed without a TxRDY (level 4) or an RxRDY (level 3) interrupt being sent to the system unit.

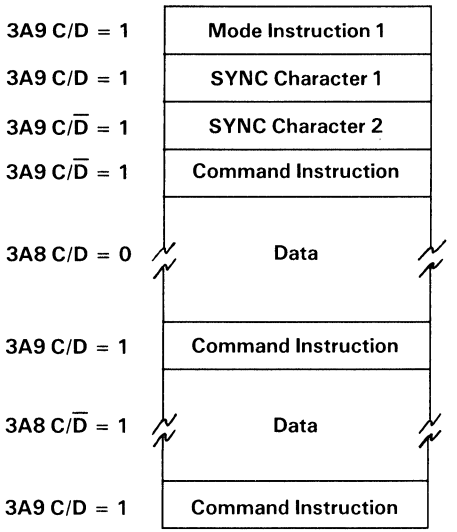
USART Programming

After the support devices on the BSC adapter are programmed, the USART is loaded with a set of control words that defines the communication environment. The control words consist of mode instructions and command instructions.

Both the mode and command instructions must conform to a specified sequence for proper device operation. The mode instruction must be inserted immediately after a reset operation before using the USART for data communications. The required synchronization characters for the defined communication technique are then loaded into the USART (usually hex 32 for BSC). All control words written to the USART after the mode instruction will load the command instruction. Command instructions can be written to the USART in the data block any time during its operation.

To return to the mode instruction, the master reset bit in the command instruction word is set to start an internal reset operation, which places the USART back into the mode instruction. Command instructions must follow the mode instructions or synchronization characters.

The following represents a typical data block and shows the mode instruction and command instruction.



Typical Data Block

The following are the communications interrupt levels.

- Interrupt level 4
  - Transmit
  - Timer 1
  - Timer 2
- Interrupt level 3
  - Receive

The following are device addresses.

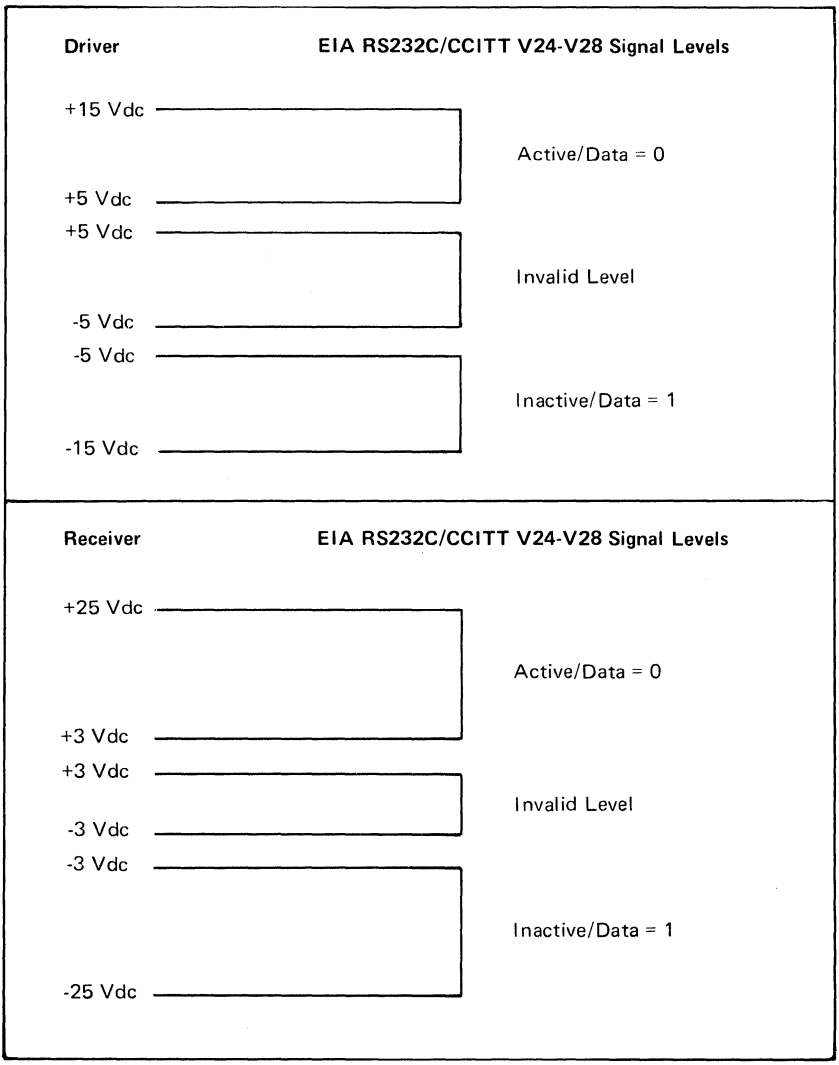
Hex Address		Device	Register Name	Function
Primary	Alternate			
3A0	380	8255	Port A Data	Internal/External Sensing
3A1	381	8255	Port B Data	External Modem Interface
3A2	382	8255	Port C Data	Internal Control
3A3	383	8255	Mode Set	8255 Mode Initialization
3A4	384	8253	Counter 0 LSB	Not Used in Sync. Mode
3A4	384	8253	Counter 0 MSB	Not Used in Sync. Mode
3A5	385	8253	Counter 1 LSB	Inactivity Time Outs
3A5	385	8253	Counter 1 MSB	Inactivity Time Outs
3A6	386	8253	Counter 2 LSB	Inactivity Time Outs
3A6	385	8253	Counter 2 MSB	Inactivity Time Outs
3A7	387	8253	Mode Register	8253 Mode Set
3A8	388	8251	Data Select	Data
3A9	389	8251	Command/Status	USART Status

Device Address Summary

8 BSC Adapter

Interface

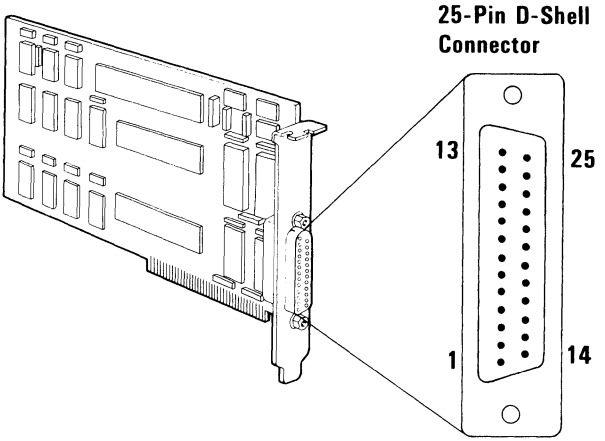
The IBM Binary Synchronous Communications Adapter conforms to interface signal levels standardized by the Electronic Industries Association (EIA) RS-232C Standard. The following figure shows these levels.



Interface Voltage Levels

Pins 11, 18, and 25 on the interface connector are not standardized by the EIA. These lines are designated as 'select standby,' 'test,' and 'test indicate.' 'Select standby' is used to support the switched network backup facility of a modem that provides this option. 'Test' and 'test indicate' support a modem wrap function on modems designated for business-machine, controlled-modem wraps.

Specifications



	Signal Name — Description	Pin	
	No Connection	1	
←	Transmitted Data	2	
→	Received Data	3	
←	Request to Send	4	
→	Clear to Send	5	
→	Data Set Ready	6	
	Signal Ground	7	
→	Received Line Signal Detector	8	
	No Connection	9	
	No Connection	10	Binary Synchronous Communications Adapter
←	Select Standby*	11	
	No Connection	12	
	No Connection	13	
	No Connection	14	
→	Transmitter Signal Element Timing	15	
	No Connection	16	
→	Receiver Signal Element Timing	17	
←	Test (IBM Modems Only)*	18	
	No Connection	19	
←	Data Terminal Ready	20	
	No Connection	21	
→	Ring Indicator	22	
←	Data Signal Rate Selector	23	
	No Connection	24	
→	Test Indicate (IBM Modems Only)*	25	

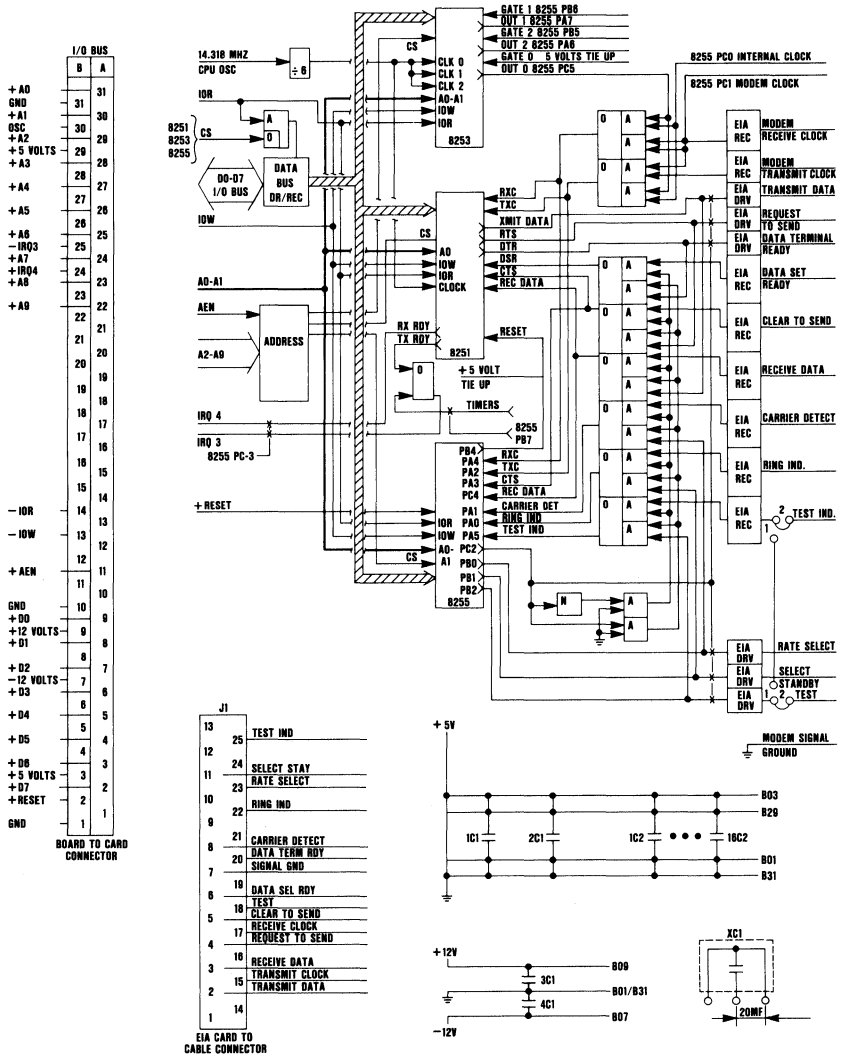
External Device

*Not standardized by EIA (Electronic Industries Association).

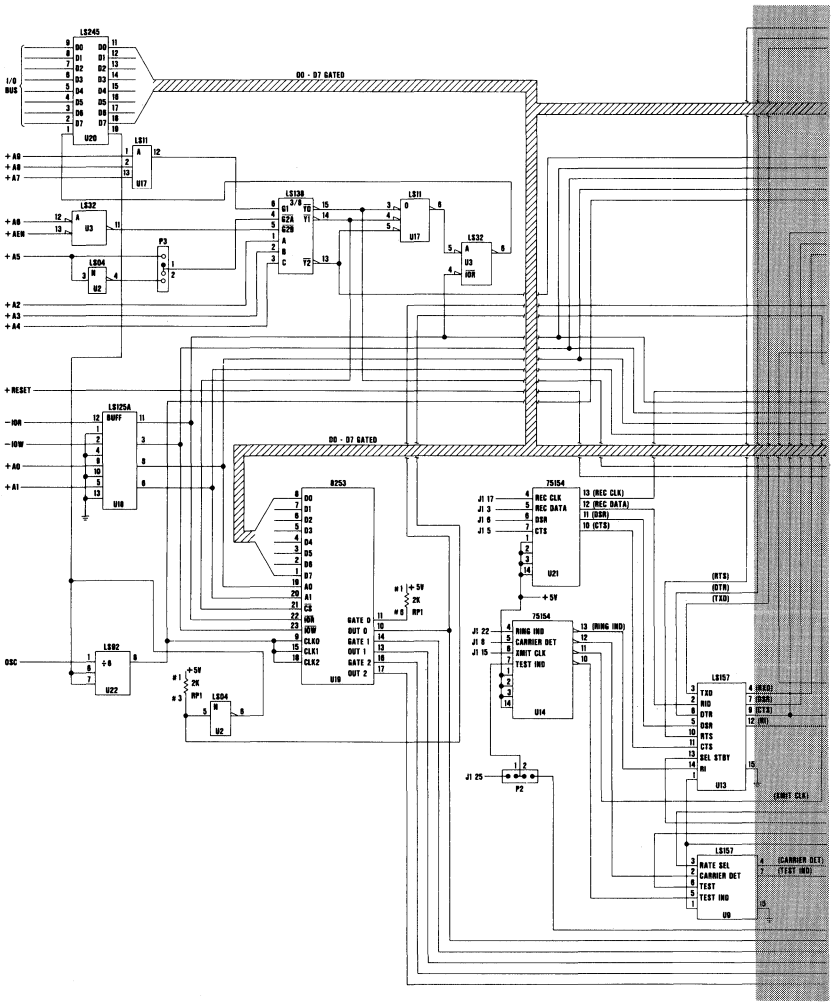
Connector Specifications

12 BSC Adapter

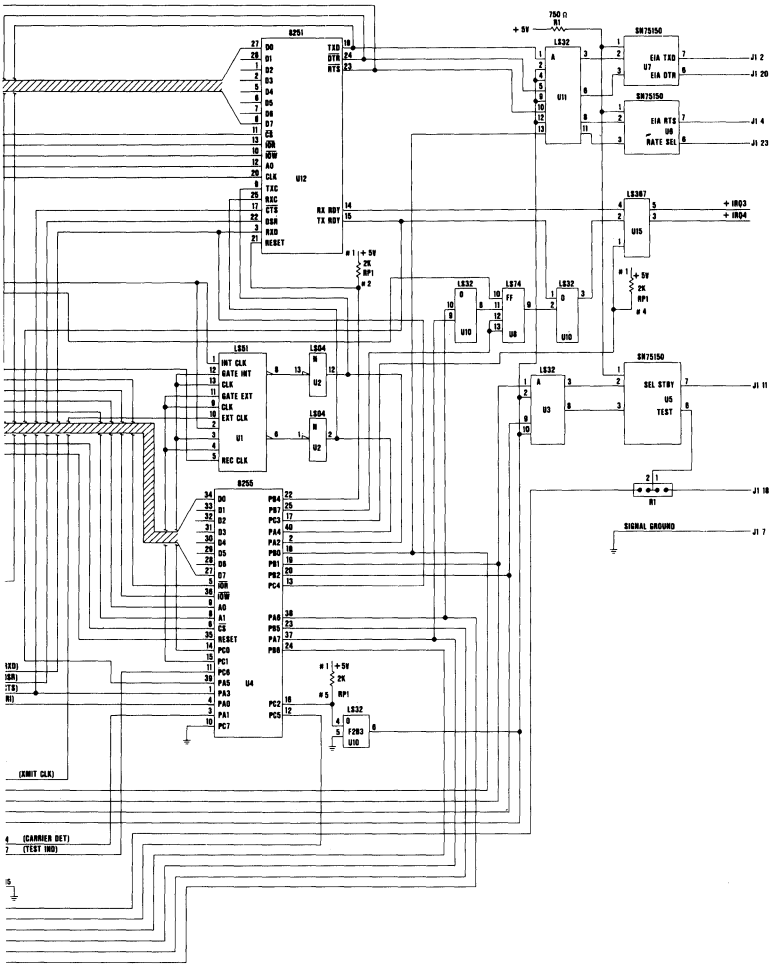
Logic Diagrams



Binary Synchronous Communications Adapter (Sheet 1 of 2)



Binary Synchronous Communications Adapter (Sheet 2 of 2)


8251
U12
LS32
U11
SN75150
EIA TXD
EIA DTR
EIA RTS
RATE SEL
U7
U6
LS367
U15
LS74
FF
U8
U10
LS51
U1
LS04
U2
8255
U4
SEL STBY
TEST
U5
U3
750 Ω
R1
+ 5V
2K
RP1
J1 2
J1 20
J1 4
J1 23
J1 11
J1 18
J1 7
+ IRQ3
+ IRQ4
SIGNAL GROUND
(XMIT CLK)
(CARRIER DET)
(TEST IND)

Personal Computer Hardware Reference Library

IBM Synchronous Data Link Control (SDLC) Communications Adapter

6361497

Contents

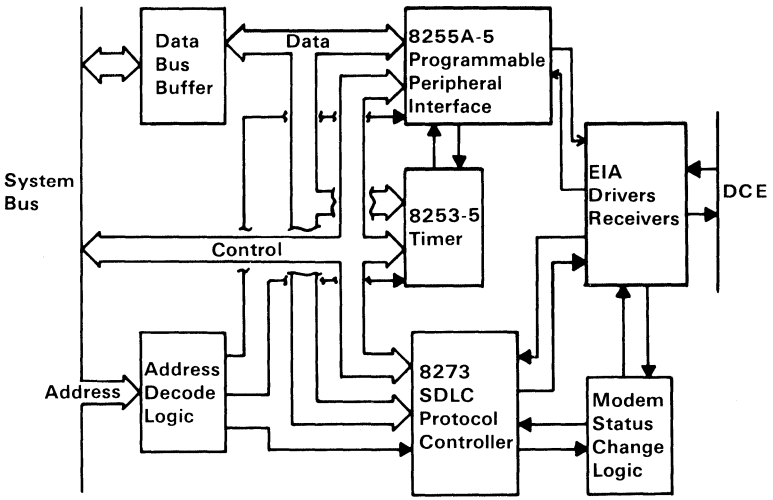
Description

The IBM Synchronous Data Link Control (SDLC) Communications Adapter provides communications support to the system in a half-duplex synchronous mode. The adapter receives address, data, and control signals from the system board through the internal bus. Electronic Industries Association (EIA) drivers and receivers connect to an RS232-C standard 25-pin, D-shell, male connector.

The adapter is programmed by communications software to operate in a half-duplex mode. Maximum transmission rate is 9600 bits per second, as generated by the attached modem or other data communications equipment.

The SDLC adapter uses an Intel 8273 SDLC Protocol Controller and an Intel 8255A-5 Programmable Peripheral Interface (PPI) for an expanded external modem interface. An Intel 8253 Programmable Interval Timer (PIT) generates timing and interrupt signals. Internal test-loop capability is provided for diagnostic purposes.

The following figure is a block diagram of the IBM SDLC Communications Adapter.



SDLC Communications Adapter Block Diagram

8273 SDLC Protocol Controller

The 8273 SDLC Protocol Controller has three operations—transmission, reception, and port read—with each operation consisting of three phases:

Command: Commands and/or requirements for the operation are issued by the system unit's microprocessor.

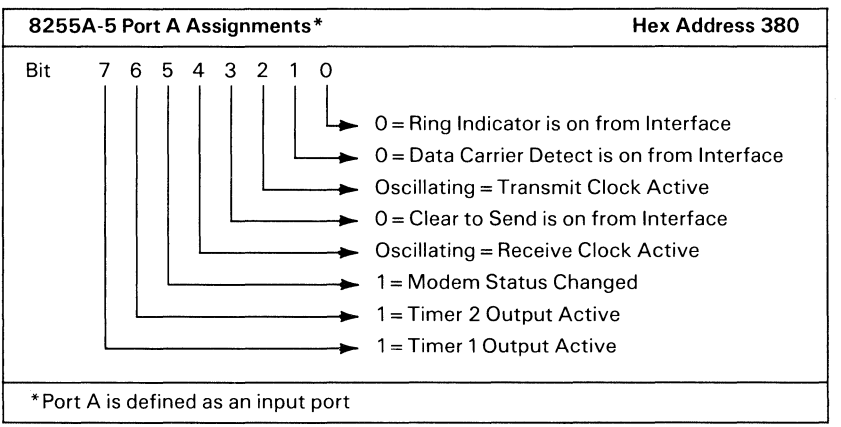
Execution: Executes the command, manages the data link, and may transfer data to or from memory using direct memory access (DMA), and thus freeing the system unit's microprocessor except for minimal interruptions.

Result: Shows the effect of the command by returning the interrupt results.

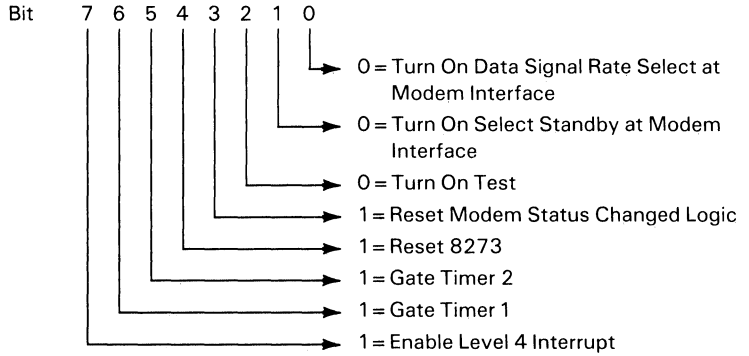
Support of these phases is through the internal registers and control blocks of the controller.

8255A-5 Programmable Peripheral Interface

The 8255A-5 PPI has three 8-bit ports—A, B, and C. Descriptions of each bit of these ports follow.

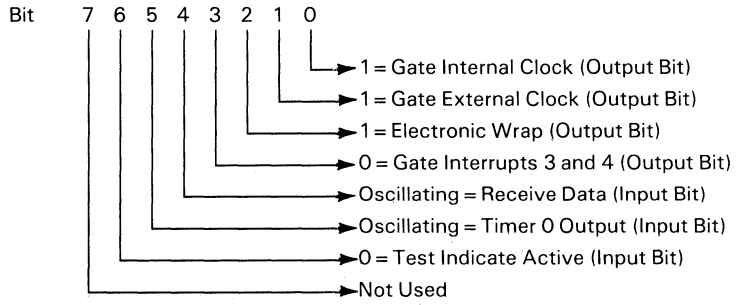


8255A-5 Port B Assignments* **Hex Address 381**



*Port B is defined as an output port

8255A-5 Port C Assignments* **Hex Address 382**



*Port C is defined for internal control and gating functions. It has three input and four output bits. The four output bits are defined during initialization, but only three are used.

8253-5 Programmable Interval Timer

The 8253-5 PIT is driven by a microprocessor clock signal that is divided by 2. The PIT's three counters provide the following output:

Counter 0 Programmed to generate a square-wave signal that is used as an input to timer 2. Also connected to port C, bit 5 of the PIT.

Counter 1 Connected to PPI port A, bit 7, and interrupt-level 4.

Counter 2 Connected to PPI port A, bit 6, and interrupt-level 4.

Programming Considerations

Initializing the Adapter (Typical Sequence)

Before the 8273 SDLC Protocol Controller is started, the support devices on the adapter must be set to the correct modes of operation.

Setup of the 8255A-5 Programmable Peripheral Interface is accomplished by selecting the mode set address for the PPI and by writing the appropriate control word to hex 98 to set ports A, B, and C to the modes described previously in this section.

Next, a bit pattern sent to port C disallows interrupts, sets wrap mode on, and gates the external clock pins (address is hex 382, data is hex 0D). The adapter is now isolated from the communications interface.

The controller reset line is brought high through bit 4 of port B, held, then dropped. This action resets the internal registers of the controller.

8253-5 Programmable Interval Timer

The PIT's counters 1 and 2 terminal-count values are set to values that will provide the desired time delay before a level-4 interrupt is generated. These interrupts may be used to indicate to the communication programs that a predetermined amount of time has elapsed without a result interrupt (interrupt-level 3). The terminal-count values for these counters are set for any time delay the programmer requires. Counter 0 also is set to mode 3 (generates square-wave signal used to drive counter 2 input).

The counter modes are set up by selecting the address for the PIT's counter-mode register and by writing the control word for each individual counter to the device separately.

When the support devices are set to the correct modes and the 8273 SDLC Protocol Controller is reset, it is ready to be set up for the operating mode that defines the communications environment in which it will be used.

Address and Interrupt Information

The following tables provide address and interrupt information for the SDLC adapter.

Hex Code	Device	Register Name	Function
380	8255	Port A Data	Internal/External Sensing
381	8255	Port B Data	External Modem Interface
382	8255	Port C Data	Internal Control
383	8255	Mode Set	8255 Mode Initialization
384	8253	Counter 0 LSB	Square Wave Generator
384	8253	Counter 0 MSB	Square Wave Generator
385	8253	Counter 1 LSB	Inactivity Time-Outs
385	8253	Counter 1 MSB	Inactivity Time-Outs
386	8253	Counter 2 LSB	Inactivity Time-Outs
386	8253	Counter 2 MSB	Inactivity Time-Outs
387	8253	Mode Register	8253 Mode Set
388	8273	Command/Status	Out = Command In = Status
389	8273	Parameter/Result	Out = Parameter In = Status
38A	8273	Transmit INT Status	DMA/INT
38B	8273	Receive INT Status	DMA/INT
38C	8273	Data	DPC (Direct Program Control)

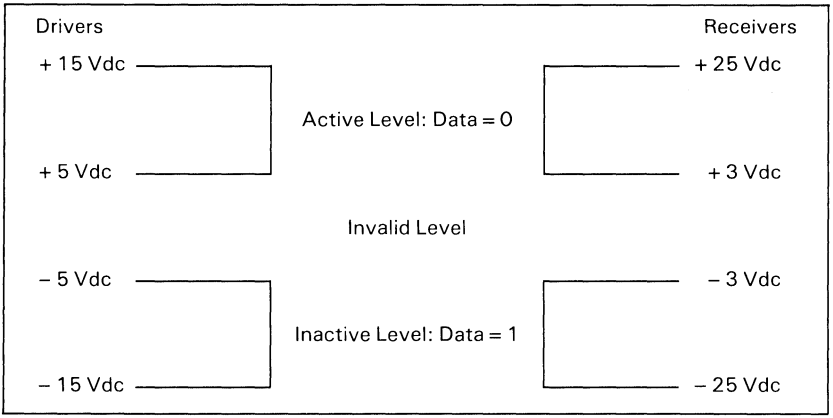
SDLC Communications Adapter Device Addresses

Interrupt Level 3	Transmit/Receive Interrupt
Interrupt Level 4	Timer 1 Interrupt Timer 2 Interrupt Clear to Send Changed Data Set Ready Changed
DMA Level 1 is used for Transmit and Receive	

Interrupt Information

Interface

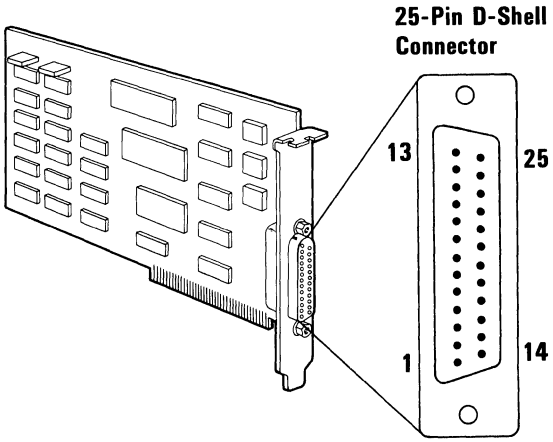
The SDLC Communications Adapter conforms to interface signal levels standardized by the Electronic Industries Association (EIA) RS232-C Standard. These levels are shown in the following figure.



Additional lines used but not standardized by the EIA are pins 11, 18, and 25. These lines are designated as 'select standby,' 'test,' and 'test indicate,' respectively. 'Select standby' supports the switched network backup facility of a modem that has this option. 'Test' and 'test indicate' support a modem-wrap function for modems that are designed for business-machine controlled modem-wraps. Two jumpers on the adapter (P1 and P2) connect 'test' and 'test indicate' to the interface.

8 SDLC Communications Adapter

Specifications



	Signal Name — Description	Pin	
	No Connection	1	
←	Transmitted Data	2	
→	Received Data	3	
←	Request to Send	4	
→	Clear to Send	5	
→	Data Set Ready	6	
	Signal Ground	7	
→	Received Line Signal Detector	8	
	No Connection	9	
	No Connection	10	Synchronous Data Link Control Communications Adapter
←	Select Standby*	11	
	No Connection	12	
	No Connection	13	
	No Connection	14	
→	Transmitter Signal Element Timing	15	
	No Connection	16	
→	Receiver Signal Element Timing	17	
←	Test (IBM Modems Only)*	18	
	No Connection	19	
←	Data Terminal Ready	20	
	No Connection	21	
→	Ring Indicator	22	
←	Data Signal Rate Selector	23	
	No Connection	24	
→	Test Indicate (IBM Modems Only)*	25	

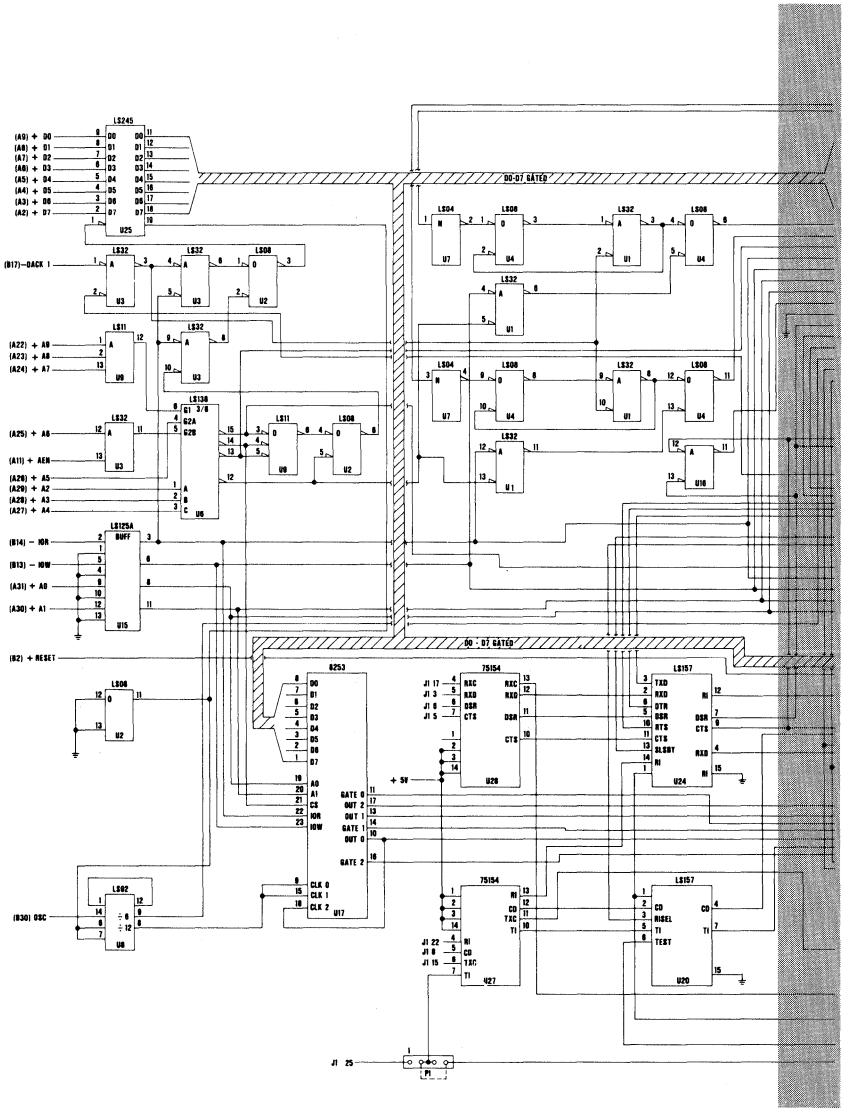
External Device

*Not standardized by EIA (Electronic Industries Association).

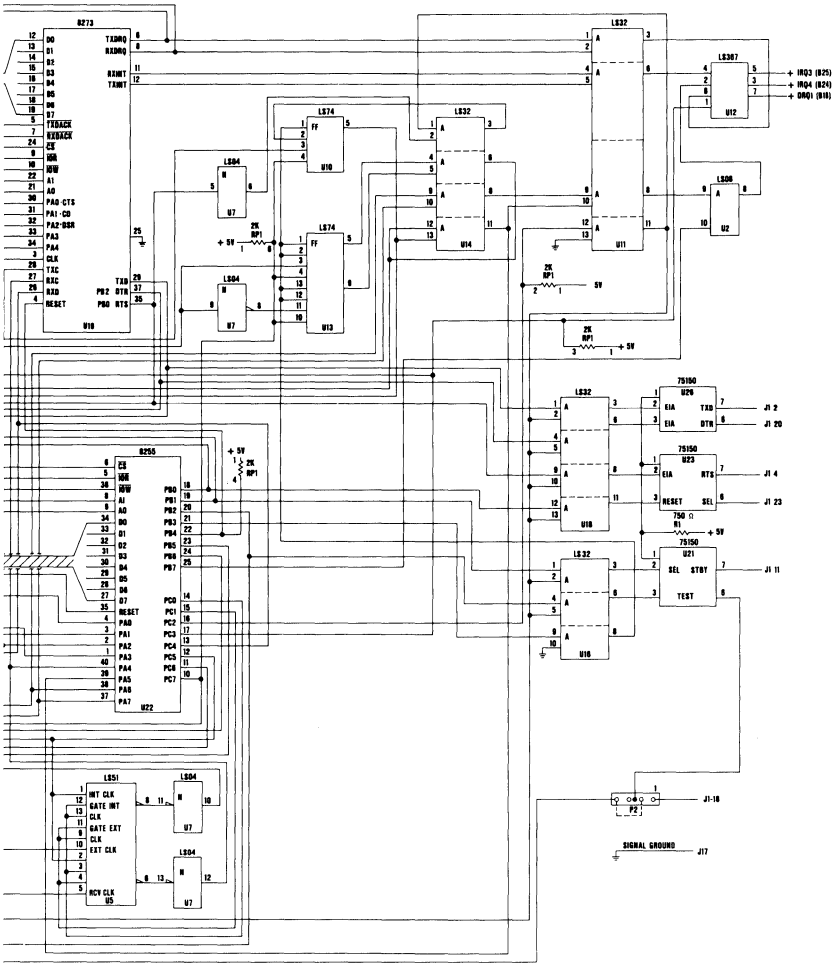
Connector Specifications

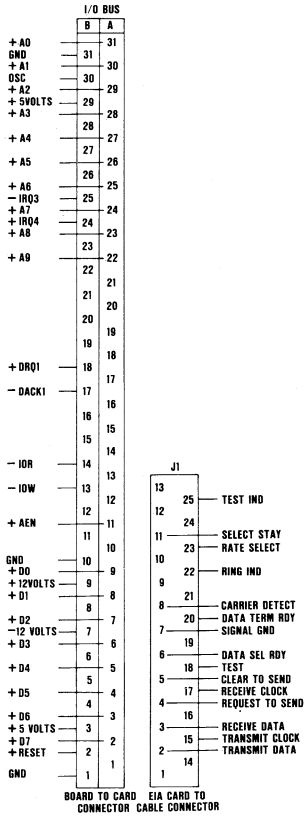
Logic Diagrams

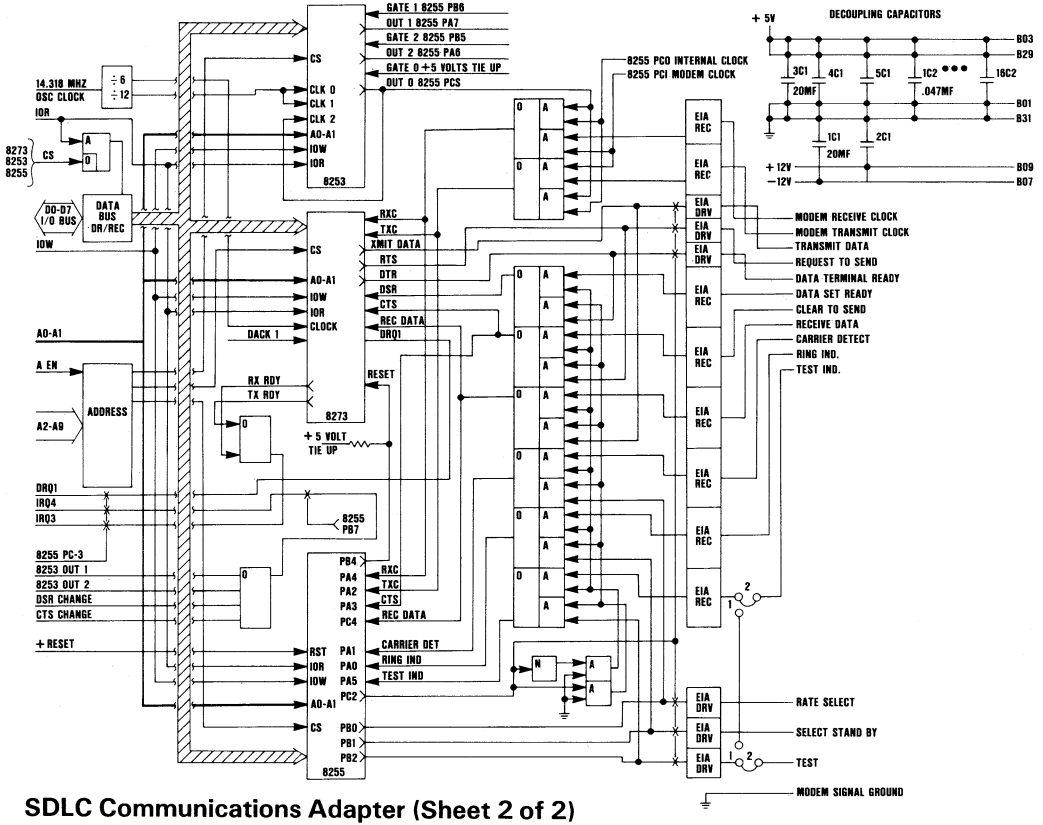
The following pages contain the logic diagrams for the IBM Synchronous Data Link Control (SDLC) Adapter.



SDLC Communications Adapter (Sheet 1 of 2)


8273
TXDRQ
RXDRQ
RXINT
TXINT
TXBACK
RXBACK
CS
RD
WR
A1
A0
PA0-CTS
PA1-CD
PA2-DSR
PA3
PA4
CLK
TXC
RXC
RXD
RESET
TXD
PB2 DTR
PB0 RTS
U18
LS74
U10
U13
LS04
U7
LS32
U14
U11
U16
LS367
U12
LS06
U2
2K RP1
5V
8255
PB0
PB1
PB2
PB3
PB4
PB5
PB6
PB7
PC0
PC1
PC2
PC3
PC4
PC5
PC6
PC7
RESET
PA0
PA1
PA2
PA3
PA4
PA5
PA6
PA7
U22
LS51
INT CLK
GATE INT
CLK
GATE EXT
CLK
EXT CLK
RCV CLK
U5
75150
U29
U23
U21
EIA
TXD
DTR
RTS
RESET
SEL
STBY
TEST
750 Ω R1
J1 2
J1 20
J1 4
J1 23
J1 11
J1-18
P2
SIGNAL GROUND
J17
IRQ3 (B25)
IRQ4 (B24)
DRQ1 (B18)





SDLC Communications Adapter (Sheet 2 of 2)

Personal Computer Hardware Reference Library

IBM Cluster Adapter

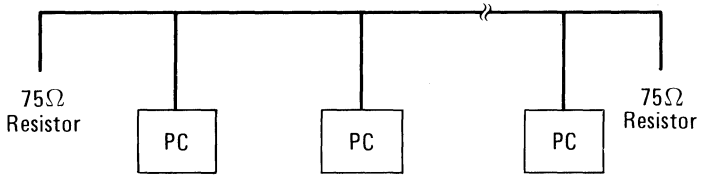
6361495

Contents

Description

The Cluster Adapter is a 10.16 cm (4 inch) high by 25.4 cm (10 inch) wide communication adapter used for linking up to 64 IBM Personal Computers (PCs). The transmission rate is 375,000 bits per second (bps). A multi-drop bus architecture passively links (cluster operation is unaffected if the power to any station is off) the PCs to a coaxial cable. The coaxial cable bus can be a maximum length of 1 kilometer (3280 feet) and requires a 75-ohm (Ω) terminating resistor at both ends to minimize signal reflection. The coaxial cable drop can be a maximum length of 5 meters (16.4 feet) and a minimum length of 1 meter (3.3 feet).

The following is an example of a cluster:



CLUSTER EXAMPLE

The PCs share the bus through a distributed-access protocol called carrier sense multiple access with collision avoidance (CSMA/CA). With this protocol, each PC (station) that wants to transmit, calculates its own access-window wait time after no signal is sensed on the bus. The wait time differs for each station and changes with each transmission to prevent collisions (two stations transmitting at the same time). If cluster traffic is light (no signal is on the coaxial cable for approximately 2.8 milliseconds), a station that wants to transmit establishes cluster synchronization by transmitting all 1's (111 . . . 1) for 150

microseconds (μs), thereby forcing a carrier sense transition (On-to-Off). The station can then calculate its access-window wait time.

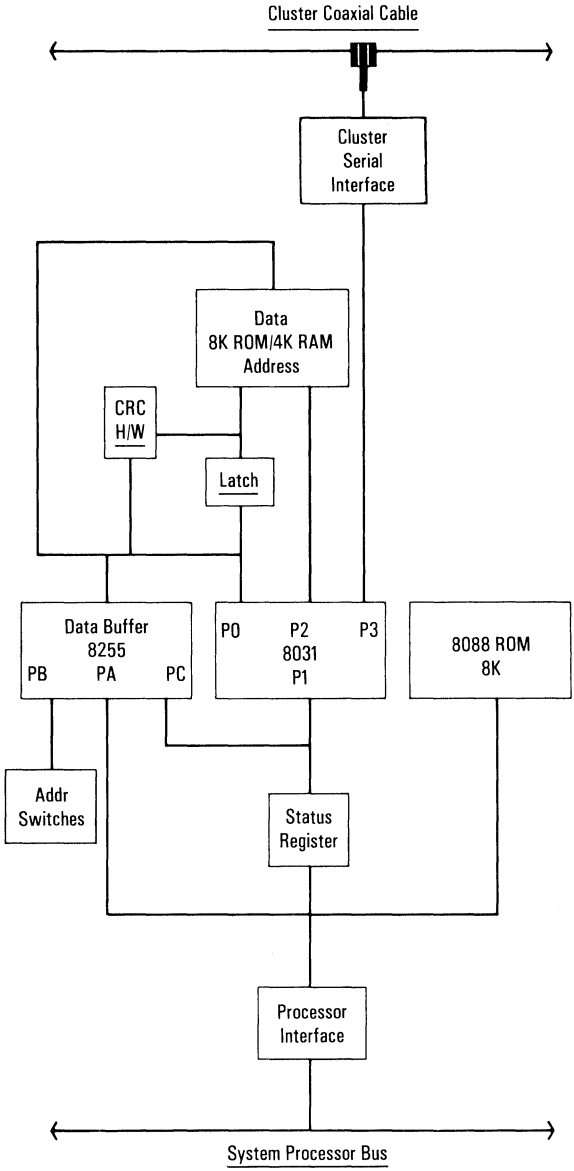
Because the PCs are passively connected and operate under a distributed-access protocol, the operation of the cluster is unaffected if the power to any single station is off.

The Cluster Adapter sends and receives frames consisting of link-control and information fields to and from other Cluster Adapters in the cluster.

The Cluster Adapter consists of the following components:

- 8031 8-bit Microcomputer
- 8031 Accessible ROM
- 8031 Accessible RAM
- System Processor (8088) Interface
- Adapter Status Register
- 8088 Accessible ROM
- 8255 Programmable Peripheral Interface (PPI)
- Cyclic Redundancy Checking (CRC) Hardware
- Cluster Interface

The following is a block diagram of the Cluster Adapter:



Cluster Adapter Block Diagram

DANGER
TO HELP PROTECT FROM LIGHTNING AND OTHER SOURCES OF ELECTRICAL SHOCK, IBM REQUIRES THAT THE COAXIAL CABLE SHIELDING BE GROUNDED, AND NEITHER THE FRAME NOR COVERS OF THE IBM PERSONAL COMPUTER CAN BE USED AS THE GROUNDING POINT.

- To ensure proper operation of the cluster, the shielding of the coaxial cable cannot be grounded at more than one point.
- If compliance to electrical codes require multiple ground points, then triaxial cable (double shielded) must be used. In using the triaxial cable, only the outer shielding can be grounded and under no circumstances should the outer shield be connected to the inner shield.
- This installation should be performed by a licensed electrician.

8031 Microcomputer

The 8031 Microcomputer is the controlling processor for the Cluster Adapter. The 8031 has an 8K x 8-bit ROM, and a 4K x 8-bit static RAM.

The 8031 consists of the following:

- A processor
- A dynamic 128 x 8-bit read/write data memory
- 32 I/O lines
- 2 16-bit timer/event counters
- A five-source, two-priority-level, nested interrupt structure
- A serial I/O port for multiprocessor communications
- I/O expansion or a full duplex Universal Synchronous/Asynchronous Receiver/Transmitter (USART)
- An on-chip oscillator and clock circuits.

The 8031 also provides addressing for up to 64K bytes of program memory and 64K bytes of data memory.

The 8031 is operated at 12 Megahertz (MHz), yielding a single-cycle time of 1 μs.

Program and data address spaces on the adapter are combined into a 64K-byte address space by ORing -Program Store Enable (-PSEN) and -Read (-RD). The memory address space includes not only the 8K x 8-bit ROM and 4K x 8-bit static RAM, but also the 8255 port and control registers and the 2653 registers necessary for CRC calculation.

8031 Ports

The 8031 on the Cluster Adapter provides external memory addresses through ports 0 and 2.

- Port 0 is an 8-bit, open-drain, bidirectional, I/O port used as the multiplexed low-order address and data bus.
- Port 2 is a bidirectional I/O port and provides the high-order address byte for the external memory.

Port 1 of the 8031 is an 8-bit, bidirectional, I/O port used on the adapter for status conditions.

Port 3 is an 8-bit, bidirectional, I/O port used as a serial port and as a source for external memory and serial-transmission control lines.

The following is a summary of the 8031 port signals:

Bits	Port 0	Port 2	Port 3	Port 1
	External Memory Address			
	Low Order Byte and Data Bus	High Order Address Byte Only	Transmission and Control Lines	Status
7	A7/D7	A15	-RD	Direction to 8031
6	A6/D6	A14	-WR	Error
5	A5/D5	A13	-CRC INT	Communication Port Busy
4	A4/D4	A12	-RTS	RX Virtual I/O Frame Available
3	A3/D3	A11	+Internal Loop	RX Frame in (FIFO)
2	A2/D2	A10	-Carrier Sense	Data Available for 8088 (0 = Active)
1	A1/D1	A9	+TXD	Command or Data Available for 8031
0	A0/D0	A8	+RXD	Command in Progress

Summary of 8031 Port Signals

Serial Transmission and Control Lines

The serial transmission lines are:

+Receive Data (+RXD)	The +RXD line provides the serial port's receiver data input.
+Transmit Data (+TXD)	The +TXD line provides the serial port's transmitter data output.

The serial transmission control lines are:

-Request to Send (-RTS)	The -RTS signal enables the adapter's transmitter to send data on the cluster cable bus.
+Internal Loop	The +Internal Loop line is used in the diagnostic mode. When high, it activates the internal loopback feature so the Cluster Adapter can receive the data it is transmitting without interference or being attached to the bus.
-Carrier Sense	-Carrier Sense is an input signal to port 3 that indicates the current state of the cluster; it is low (0) when the cluster is busy.

The memory control lines are:

-Write (-WR)	The -WR line latches the data byte from port 0 into the external data memory.
-Read (-RD)	The -RD line enables external data memory to port 0.

The interrupt line is:

-CRC Interrupt (-CRC INT)	The -CRC INT line is used to indicate a successful or unsuccessful comparison in CRC values. The signal source is -INT from the 2653 Polynomial Checker Generator.

8K x 8-Bit ROM

The 8K x 8-bit ROM contains the 8031 code necessary for hardware initialization and the data link control program (DLCP). The DLCP is the lowest level of software for the Cluster Adapter. The DLCP resides in the 8K by 8-bit ROM on the Cluster Adapter, which is accessible by the 8031 Microcomputer.

4K x 8-Bit Static RAM

The 4K x 8-bit static RAM is available to the 8031 for read/write storage necessary to implement the DLCP. The available space is used to buffer frames and to store control and cluster information. The 4K x 8-bit static RAM is implemented using two 2K x 8-bit static RAM modules.

The following is the 8031 memory map:

Start Address (Hex)	Function
0000	DLCP ROM
2000	RAM
3000	8255 Port A
3001	8255 Port B
3002	8255 Port C
3003	8255 Control
3004	2653 Character Register
3005	2653 Status Register
3006	2653 Mode Register
3007	2653 CRC Upper/Lower Registers

8031 Memory Map

8088 Accessible ROM

The 8088 (System Processor) accessible ROM is an 8K x 8-bit ROM and contains the 8088 code necessary to perform the remote initial program load (IPL) and power-on diagnostic functions.

2653 Polynomial Generator Checker

The 2653 Polynomial Generator Checker is used by the 8031 Microcomputer to compute the Cyclic Redundancy Check (CRC) value for transmitted or received data blocks for error checking.

The 2653 is programmed by the 8031 in the automatic mode to generate the American National Standards Institute (ANSI) CRC-16 values. Two 8-bit characters are read from the 2653 character register into the Block Check Character (BCC) generation unit to calculate the 16-bit check character.

Programming is achieved as follows:

- The Clear CRC command, hex 02, is issued to the 2653 command register at address hex 3005.
- The Automatic Accumulation Mode command, hex 49, is issued to the 2653 mode register at address hex 3006.
- The Start Accumulation command, hex 01, is issued to the 2653 command register at address hex 3005.
- Characters to be accumulated are written to the character register at address hex 3004.

The accumulated CRC value may be read by the 8031 from address hex 3007 (the 2653 CRC upper and lower registers) in two read operations. The 2653 alternately provides the upper and lower values.

The 2653 is activated upon proper decoding of addresses in the range of hex 3004 through hex 3007 and the occurrence of -Read Strobe (-RS) or -Write (-WR). This allows the input to the -Read/Write (-R/W) pin of the 2653 to become stable prior to the fall of -Clear Entry 1 (-CE1), as required.

Cluster Adapter I/O Register Definitions

The following defines the Cluster Adapter I/O registers:

Adapter	I/O Address (Hex)	Device
Adapter 1	0790	Adapter Status Register
	0791	Adapter Command/Data (Output)
		Adapter Result/Data (Input)
	0792	Adapter Interrupt Register
	0793	Adapter Reset Control
Adapter 2	0B90	Adapter Status Register
	0B91	Adapter Command/Data (Output)
		Adapter Result/Data (Input)
	0B92	Adapter Interrupt Register
	0B93	Adapter Reset Control
Adapter 3	1390	Adapter Status Register
	1391	Adapter Command/Data (Output)
		Adapter Result/Data (Input)
	1392	Adapter Interrupt Register
	1393	Adapter Reset Control
Adapter 4	2390	Adapter Status Register
	2391	Adapter Command/Data (Output)
		Adapter Result/Data (Input)
	2392	Adapter Interrupt Register
	2393	Adapter Reset Control

Cluster Adapter I/O Registers

Adapter Status Register

The adapter status is provided to the system data bus by a 74LS373 transparent latch.

The following are the bit assignments:

Bit	Definition (1 = Active Unless Noted)
7	Direction (1 = data expected from 8088 to 8031)
6	Error
5	Communication Port Busy
4	RX Virtual I/O Frame Available
3	RX Frame in First in First Out (FIFO)
2	Data Available for 8088 (0 = active)
1	Command/Data Available for 8031
0	Command in Progress

Status Register Bit Definitions

The outputs of the transparent latch, though not enabled on the bus, continuously follow the inputs provided by the 8031 and 8255. Upon decoding of the read-status I/O address, the latch-enable input to the transparent latch goes low, latching the inputs of the current state and enabling the data onto the bus.

The status bits are latched during the active read time to preserve the integrity of the data. When the outputs are disabled and the latch-enable input to the latch goes high at the end of the read cycle, the outputs of the transparent latch again monitor the inputs in real time.

Definition of Bits at Port 0791 (for Adapter 1) (Command or Parameters for 8031)	
Bit	**Definition**
7	Command or Data Bit 7
6	Command or Data Bit 6
5	Command or Data Bit 5
4	Command or Data Bit 4
3	Command or Data Bit 3
2	Command or Data Bit 2
1	Command or Data Bit 1
0	Command or Data Bit 0

Cluster Adapter Command/Data Register (Output)

Definition of Bits at Port 0791 (for Adapter 1) (Result or Data from 8031)	
Bit	**Definition**
7	Result or Data Bit 7
6	Result or Data Bit 6
5	Result or Data Bit 5
4	Result or Data Bit 4
3	Result or Data Bit 3
2	Result or Data Bit 2
1	Result or Data Bit 1
0	Result or Data Bit 0

Cluster Adapter Result/Data Register (Input)

Definition of Bits at Port 0792 (for Adapter 1)	
Bit	**Definitions**
7-2	Not used.
1	Received Frame(s) Available. One or more information frames have been received and may be read using either the BIOS Receive Frame or Receive Virtual I/O Frame command (1 = active).
0	Cluster BIOS Command Complete. The Cluster BIOS command intiated with the Initiate Transmit bit set is complete. The result must be obtained by issuing the same Cluster BIOS command with the Finish Transmit bit set (1 = active).

Cluster Interrupt Status Bits

Note: Both bits 1 and 0 are set to indicate interrupt due to Cluster Status command complete.

Definition of Bits at Port 0793 (for Adapter 1)	
Bit	**Definitions**
7-1	Not used.
0	Reset Cluster Adapter. The adapter microprocessor as well as all other logic on the adapter will be held in a reset condition until there is an output with this Reset Adapter bit set to zero (1 = active).

Note: Any output to the reset register will also disable the adapter from generating interrupts.

Cluster Adapter Reset Register Bit Definitions

Cluster Adapter Interrupts

The Cluster Adapter may be set (one jumper selectable) to allow interrupts on either interrupt-level 3 or interrupt-level 7. An adapter error detected by diagnostic tests is reported if the interrupt jumper is missing. The received frames must be available or the Transmit operation complete (if initiated by a Transmit command with the Initiate Transmit bit set).

Up to four Cluster Adapters can be installed at a station. Each adapter can be enabled/disabled and all are similar in operation. If enabled, the adapter generates interrupts on levels 3 or 7 provided one of the following conditions is met:

- A received frame is available.
- The Transmit Frame command is complete.
- The Cluster Status command is complete.

The following description is for adapter 1:

1. Interrupts are enabled by executing an output instruction to the adapter's interrupt enable register.

2. Interrupts are disabled by writing the hex 00 instruction to the adapter's reset register. Also, additional interrupts are disabled by generating the interrupt request. The adapter must be re-enabled after each interrupt if additional interrupts are desired.

3. To avoid resetting the adapter, data bit 0 must be set to a 0 when an output is sent to the adapter's reset register.

   No interrupt handler is provided for the cluster, and must be provided by the user who requires interrupt capability.

   The interrupt condition is provided in the adapter's interrupt register, as described in the Cluster Adapter Interrupt Status Bits table.

Programming Considerations

The data link control program (DLCP) is the lowest level of software for the Cluster Adapter. The DLCP resides in the 8K by 8-bit ROM, which is accessible by the 8031 Microcomputer.

The Cluster Adapter basic input/output system (BIOS) code resides in an 8K-byte 8088 accessible ROM on the Cluster Adapter at address hex D0000.

> **Note:** The Cluster Adapter decodes a 32K-byte range starting at hex D0000. High-level cluster BIOS commands are processed by the cluster BIOS into the appropriate low-level commands and parameters. The low-level commands and parameters are then passed to the 8031 Microcomputer, which performs the requested command. After the command is complete, the 8031 Microcomputer transfers the results back to the DLCP BIOS routine, which fills in the requester's link control block (LCB) with the results and then return through an interrupt return (IRET) to the requester that issued the INT hex 5A.

The cluster BIOS level interface allows the higher layer communication program to transmit to and receive data from the specified destination through the bus. The basic unit of information transmitted using DLCP is a frame. A frame consists of a control field and an optional data field.

The following functions are implemented in the DLCP to interface with the higher layer communication program and to ensure reliable data transfer between stations on the bus:

- Higher layer communication program BIOS interface to the communication software
- Frame assembly, reception and transmission
- CRC generation and checking
- Carrier sense multiple access with collision avoidance (CSMA/CA)
- Error detection and recovery
- Cluster status monitoring
- Remote IPL

Higher Layer Communication Program BIOS Interface

When the Power switch is set to On, the hex 5A software interrupt vector is set to the address of the Cluster Adapter BIOS by the adapter's self-test diagnostic code.

Notes:

1. The DLCP must be initialized before it can process most of its commands.

2. Interrupt hex 5A is reserved for the Cluster Adapter BIOS and should not be changed.

The higher layer communication program must access the Cluster Adapter BIOS through an interrupt hex 5A instruction. The program must set the Extra Segment (ES) Register output to the segment and the Base Index (BX) Register output to the offset of the Link Control Block (LCB) before invoking the cluster DLCP BIOS. All parameters, the return code, and the cluster status are passed through the LCB.

The format of the LCB is shown below:

Link Control Block (LCB)	Number of Bytes
Destination Station Physical Address	1
Source Station Physical Address	1
Command	1
Buffer 1 Length	2
Buffer 1 Address	2 (Offset)
	2 (Segment)
Buffer 2 Length	2
Buffer 2 Address	2 (Offset)
	2 (Segment)
Return Code	1
Cluster Status	1
Select Adapter	1

Structure of Link Control Block (LCB)

Notes:

1. The internal variables and buffers of the DLCP are in the RAM resident on the adapter and are not directly accessible from the higher layer communication program.

2. Select Adapter is used to select the adapter for which the command is intended (0 for adapter 1, 1 for adapter 2, 2 for adapter 3, and 3 for adapter 4).

3. For the length and address fields, the word values are ordered least-significant byte first.

The contents of buffer 1 and buffer 2 together form the information field of the frame. For example, buffer 1 can be used to store header bytes while buffer 2 can be used to store the actual data to be transferred.

The return code indicates the success or failure of the function requested, and the error code if the function fails. The LCB status indicates the current status of the cluster. This field is valid as a result of the DLCP Status command. The LCB status field is also used by some commands to store an extended return code.

Frame Transmission

Transmit or Transmit Virtual Information frames are sent by the DLCP to complete the corresponding DLCP BIOS commands. The DLCP on its own initiative transmits various frames. The following response frames are issued in response to a received frame:

Ack	Reception OK with no problems
Frame Reject	All receive buffers full
Not Connected	Not connected to sending station
Bad Error	Frame out of sequence (rejected)
Duplicate Address	Duplicate station address exists on the cluster

The following control frame is transmitted by the DLCP when the Power switch is set to On or at initialization:

Initializing	Broadcast to all stations to indicate that the source station is in the process of initializing and all connections to that station should be set to the disconnected state. Also, if any station has the same station address, it sends a duplicate-address response back to the initializing station.

In addition, the DLCP determines if it is necessary to send a connect frame to establish connection with the destination station. If this station's Cluster Status table indicates that it is not connected to the destination station, the DLCP transmits a connect frame to establish connection and then transmits the information frame. If a not-connected control frame is received in response to the transmission of a frame, the DLCP transmits a connect frame to establish connection, then transmits the information frame.

Frame Format

The basic unit of information transmitted is a frame. The On-to-Off transition of the 'carrier sense' signal identifies the beginning of a frame, and the Off-to-On transition identifies the end. A frame consists of fixed control fields and an optional variable length information field. The following shows the format of a frame:

Field	Number of Bytes	Note
Destination Address	1	Control
Source Address	1	Field
Transmit Window Token	1	
Control	1	
Sequence	1	
Byte Count	2	
Control CRC	2	
Information	1 to 578	Information
Data CRC	2	Field

Frame Format

Note: The minimum and maximum total number of bytes transmitted for a frame is 9 and 587, respectively. The transmission time for a frame ranges from approximately 1 millisecond (ms) for a minimum length frame up to approximately 16.5 ms for the maximum length frame. However, additional time may be required to gain access to the cluster before a frame can be sent.

Control Field Format

The control field consists of the following:

Destination Address - The destination address can be any number from hex 00 through hex 3F; that is, 64 station addresses are supported. Address hex FF is reserved as the broadcast address that all stations respond to. Addresses hex FE through hex F0 are reserved for use as multicast addresses.

Source Address - The source address is used to tell the DLCP the senders station address. The DLCP uses the source address as an index into a Cluster Status Table, which is used to maintain the status of connected stations and sequence numbers for each possible sender. Station addresses hex 00 through hex 3F are the only supported source addresses.

Transmit Window Token - This value is updated for every transmission and is used in an algorithm to determine how long each station must wait after Carrier Sense Off before transmitting.

Control Byte - The control byte is used to identify the function of a frame. There are two basic types of frames used in the cluster, information frames and control frames. Information frames are used to transfer information from one station to another, and control frames are used to assure reliable transfer of information across the cluster bus.

The following types of frames are used by the DLCP:

Acknowledge (hex 10) Confirm receipt of a frame.

Initializing (hex 21) Indicates that the source station is re-initializing. Existing connections to this station should be cleared.

Virtual Disk (hex 82) Identifies that this frame contains a data block and was transmitted as a result of the source station issuing a Transmit Virtual Frame DLCP command. One buffer is reserved for this frame.

Information (hex 83) Signifies the frame contains a data block and was transmitted as a result of the source station issuing a Transmit Frame DLCP command. There is a first-in-first-out (FIFO) buffer set aside for this frame.

Connect (hex 04) Establishes the virtual point-to-point connection between a pair of stations.

Broadcast (hex 45) Signifies that the frame is a broadcast or multicast frame.

Not Connected (hex 16) Indicates that the receiving station is not connected to the sending station.

Frame Reject (hex 17) Sent by the receiving station when it has received an information frame or a virtual disk frame and the DLCP does not have buffer space available to store the frame.

Bad Error (hex 18) Sent by the receiving station to indicate that a frame is out of sequence.

Duplicate Address (hex 19) Sent by the receiving station in response to an initializing control frame to indicate that more than one station has the same address.

Are You There? (hex 1A) Sent to each station to poll for status in the cluster. Each station that is on sends a response to this query. An Acknowledge response frame is sent by stations that are initialized. A Frame Reject response is sent by stations that are not initialized.

Note: The most-significant four bits of the frame-control byte have the following meaning:

Sequenced Information	Bit 7
Broadcast Information	Bit 6
Broadcast Control	Bit 5
Response	Bit 4

Frame Sequence Byte - If one of the acknowledge frames did not reach the transmitting station, the frame sequence byte is used to make sure that no duplicate information frames are received. The least-significant four bits in the Cluster Status Entry are used for maintaining a sequence number for transmitted and received frames. The first two bits are used for the sequence number for received frames. The two least-significant bits are used for the sequence number for transmitted frames. The sequence numbers are incremented each time a transmitting station sends an information frame and each time the receiving station accepts an information frame. If a mismatch occurs between the two stations, the sender marks the destination station in the disconnected state and sends a connect frame to try to reconnect with the destination station. If the connection attempt is successful, the frame is transmitted again.

Byte Count - The byte count is the number of information bytes to be transmitted. If the frame is a control frame, the byte count is zero. There are two bytes allocated for the byte count.

Control CRC - A 16-bit cyclic redundancy check (CRC) is calculated and appended to the end of the control block. A hardware CRC generator is used. The receiving station compares the control CRC received with the CRC calculated from the received data and makes sure they are the same. If they are not the same, the receiving station ignores the rest of the frame.

Data CRC - A 16-bit CRC is calculated and appended to the end of the data block. The receiving station compares the data CRC received with the CRC calculated from the received information bytes and makes sure they are the same. If they are not the same, the receiving station ignores the received frame.

Information Field

This field is for an information frame only. The information field is absent in the control frames. The maximum number of information bytes that can be transmitted in a frame is 578.

Cluster Access Protocol

Collision avoidance is used with the Cluster Adapter. To avoid collisions, each station waits a different amount of time after 'carrier sense' goes inactive before transmitting.

Stations get access to the cluster by timing from the end of the current transmission (-Carrier Sense On-to-Off transition) until its transmit time is reached, and then it may transmit. See also "Collision Avoidance (Medium or High Activity)" on page 30.

Each station maintains two flags to determine that it is permitted to transmit.

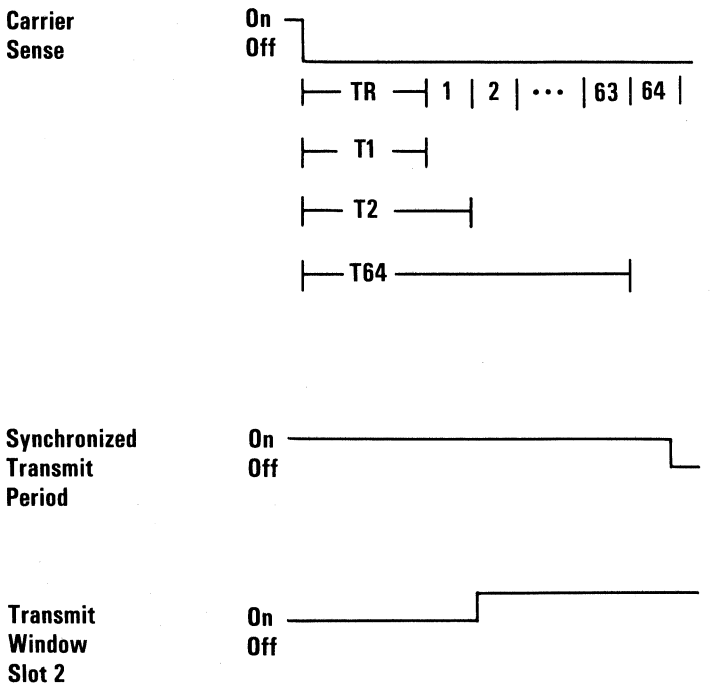
1. Synchronized Transmit Period.

2. Transmit Window.

The Synchronized Transmit Period is set and the Transmit Window is cleared when the Carrier Sense Interrupt routine is entered. Also, timer 0 is reloaded with the count corresponding to this station's calculated Transmit Access Window. Timer 0 counts while 'carrier sense' is off and overflows when this station's Transmit Access Window is reached. Timer 0's overflow causes an interrupt that sets the Transmit Window flag and then reloads timer 0 with the count corresponding to the end of the synchronized transmit period. When timer 0 interrupts again on overflow, the Synchronized Transmit Period flag is reset to indicate that the synchronized transmit period is finished.

Collision Avoidance (Medium or High Activity)

The following shows the timing during medium or high activity in the cluster:



Collision Avoidance (Medium or High Activity)

TR = Time allocated for a receiver to start transmitting a response.
T1 = Time delay for 1st Transmit Access Window.
T2 = Time delay for 2nd Transmit Access Window.

.
.
.

T64 = Time delay for 64th Transmit Access Window.
SN = Station N's address with the bits in reverse order.
Token = Transmit Window Token which is decremented by 2 for each transmitted frame.

Delay time for Station N = TR + ((Token + SN) mod 128) x transmit window/2.

Notes:

1. TR is approximately 200 μs.

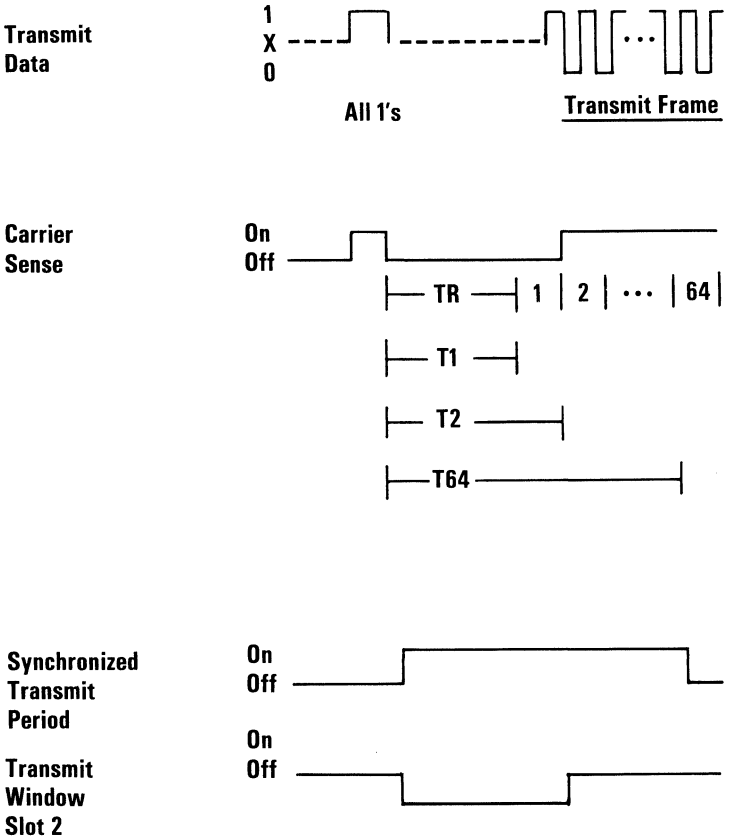
2. Transmit Access Window is approximately 40 μs.

A station must see its Transmit Window flag change from Off to On before it is permitted to transmit. The case where it does not see the change is covered in the next section.

Collision Avoidance (Light Activity)

If cluster activity is light (1480 μs average access time since the previous transmission on the cluster) enough that the Synchronized Transmit Period (STP) flag is reset, then synchronization needs to be re-established to avoid collisions.

The method used to re-establish synchronization is to transmit all 1's in the cluster for approximately 150 μs and then to time the carrier sense On-to-Off condition to this station's transmit slot time. (See also "Collision Avoidance (Light Activity)" on page 33).



Collision Avoidance (Light Activity)

TR = Time allocated for a receiver to start transmitting a response.
T1 = Time delay for 1st Transmit Access Window.
T2 = Time delay for 2nd Transmit Access Window.
.
.
.
T64 = Time delay for 64th Transmit Access Window.

Note: Average cluster access time is 1480 μs if the cluster is lightly loaded.

A station that is initializing waits the time of two complete synchronization periods before sending its broadcast initializing frame to allow it to become synchronized with the cluster. If no frames are received in that time, it uses the procedure above to establish a synchronized transmit period.

Frame Reception

The leading edge of the 'carrier sense' signal is used to interrupt the 8031 Microcomputer. The 8031 interrupt service routine updates its Transmit Window Token to the value transmitted with the frame, and also sets the timer 0 counter to the calculated Transmit Access Window based on the new token value. If the frame is not addressed to this station, the DLCP ignores the rest of the frame and leaves the interrupt routine.

If the frame is addressed to the station, the DLCP checks the Cluster Adapter status to see if it can accept the frame. If this station is not connected to the source station then a not-connected control frame is transmitted to the source station. If the frame is out of sequence, an bad error control frame is transmitted to the source station.

If the DLCP can accept the frame, a check is made that a receive buffer is available. If a buffer cannot be obtained, a frame- reject control frame is sent back to the transmitting station. This indicates that the frame cannot be accepted at this time and another attempt should be made. If the frame is received

correctly, DLCP transmits an Acknowledge frame to the transmitting station and return the control to the interrupted 8031 program.

Error Detection and Recovery

The DLCP can detect various cluster errors and tries to recover from them. If it is not able to recover after a specified number of retries, it notifies the calling program with the returned error code. The list of errors that can be detected is in the figure below:

Type of Error	Action Taken	Retry Count	(Seconds)
Cluster Busy Timeout	Report Error	N/A	1.0
Cluster Access Timeout	Report Error	N/A	13.0
No Response	Retransmit Frame	8	0.20
Frame Reject	Retransmit Frame after Delay	1 2 3 4 5 6 7	0.24 0.09 0.16 0.25 0.36 0.49 0.64
Not Connected or Bad Error	Transmit Connect Control Frame and If Successful Retransmit Frame	N/A	N/A
Command Timeout	Reset Adapter and Report Error	N/A	120.0

Detectable Errors and Recovery

After correctly receiving a control frame or an information frame, the receiving station sends a response frame. If all receive buffers are in use, a Frame Reject response frame is transmitted. If the frame is out of sequence, a Bad Error response frame is transmitted.

If the transmitting station did not get a confirmation of receipt after a certain time period, it assumes that the receiving station never got the frame and it transmits the same frame again. If the transmitting station still does not get a reply after eight retries, it assumes that the receiving station is not available and resets the Connected bit in the corresponding Cluster Adapter status entry.

Cluster Status Table

The DLCP keeps track of the status and sequence numbers for connection with stations 0 through 63 in the Cluster Table in the Cluster Adapter's RAM space. Offset 0 in the Cluster Table corresponds to the status for connection to station 0, offset 1 for station 1, and so on. The offset corresponding to a station's own address is used to store a duplicate-station address indicator.

The bits for each Cluster Status Table byte are designated in the following chart:

Cluster Status Entry (1 Byte)		
C	7	1 = Connected
RB1	6	Response ID
RB0	5	Response ID
P	4	1 = Response Pending
RS1	3	Received Frame Sequence
RS0	2	Received Frame Sequence
TS1	1	Transmitted Frame Sequence
TS0	0	Transmitted Frame Sequence

Cluster Status Table Entry

Bit 7 - Connected (C) is set to 1 when your station has sent a connect frame and an acknowledge frame has been received, or when a connect frame has been received and an acknowledge has been sent. Connected is reset when a not-connected, bad error, or initializing control frame is received.

Bit 6, 5 - Response ID

The following table defines the meaning of these two bits:

Bit 6 RB1	Bit 5 RB0	Type of Response
0	0	Acknowledge
0	1	Frame Reject
1	0	Not Connected

Response ID in Cluster Table Status Entry

Bit 4 - Pending (P) is set to 1 by the transmitting station to indicate that it is waiting for an acknowledge frame from the destination station , and is reset by the interrupt handler when a response is received or upon a time-out.

Bit 3, 2 - Received Frames Sequence Number is incremented every time a new data-sequenced information frame is successfully received. This sequence number and the transmitted frame sequence number are reset to 0 when a connection is established between two stations.

Bit 1, 0 - Transmitted Frames Sequence Number is incremented every time a sequenced information frame is successfully transmitted; that is, an acknowledge is received from the destination station.

Remote IPL

A vector is established at bootstrap vector INT hex 19 to the Remote System Reset Program Loader for the cluster, which is located in adapter 1's ROM. The original contents of the bootstrap vector are stored at vector INT hex 5B. The disk server station address is stored at the least-significant byte of vector INT hex E1. The number of the adapter from which to IPL is stored at the word corresponding to the segment at vector INT hex E1.

The following actions are performed by the Remote System Reset Program Loader:

1. The Remote System Reset Program Loader (in the Cluster Adapter's 8088 accessible ROM) uses a portion of the top 1K bytes of memory for variable and buffer space.

2. The bootstrap vector is restored with its original vector (which was temporarily saved at INT hex 5B). The INT hex 5B vector is set to point to the adapter's diagnostic routines.

3. The variables of DLCP are initialized by executing a DLCP BIOS Cluster Initialization command (hex 00) with parameters provided by a table of constants in the adapter's 8088 ROM.

4. The user timer-interrupt vector at vector hex 1C is saved at interrupt vector hex E2 and replaced with the address of a routine to update a timer count variable used for time-outs by the Remote System Reset Program Loader. It is restored before this routine is left.

5. A broadcast frame requesting IPL is sent using the DLCP BIOS command's Transmit Broadcast Frame (hex 08) to all stations in the cluster. The format of the data portion of the frame is:

Command =	hex 91	(Request for IPL)
Session ID =	hex 0000	(2 bytes)

6. An acknowledge information frame is expected with the following data:

Command =	hex 92	(Response to IPL request)
Session ID =	hex xxxx	(2 bytes)
Status =	hex 00	(non-zero is irrecoverable error)

xxxx = any hexidecimal number

The server station's address is saved at the least-significant byte of vector INT hex E1.

Up to eight retries are made unless a response from the disk server station is received. Approximately 4 seconds are allowed between retries. After the eight retries have been used, the user timer-interrupt vector is restored and then control is passed to the bootstrap routine.

Note: If a Keep-Alive command is received from the disk server station, an additional 30 seconds is allowed.

7. Next, the Remote IPL program requests a data block containing program code from the disk server station. The request has the following form:

Command =	hex 93	Request IPL data block
Session ID =	hex xxxx	(2 bytes)
Status =	hex 00	(Non-zero is a irrecoverable error)

xxxx = any hexidecimal number

The request is sent using the DLCP BIOS command's Transmit Frame (hex 03). Retries are made for up to 20 seconds if the return code indicates a Frame Reject or a No Response error.

8. The disk server sends a response containing the next data block. The response has the following form:

Command =	hex 94	Response with IPL data block
Session ID =	hex xxxx	(2 bytes)
Status =	hex 00	(Non-zero is a irrecoverable error)
Sector # =	hex xxxx	Relative sector number
Data Block =	[0-512 bytes]	Data Block containing program code.

xxxx = any hexidecimal number

The DLCP BIOS Receive Frame command (hex 02) is issued to read the response frame containing the block of program code. Approximately 20 seconds are allowed to receive a valid response from the disk server station. If a Keep-Alive command is received from the disk server station, an additional 30 seconds are allowed. There is no limit to the number of Keep-Alive commands that are accepted. On time-out, the user timer-interrupt vector is restored and control is passed to the Bootstrap Loader by INT hex 19.

The received sector number must start at zero and increment for each block of program code received. If the received sector number is incorrect or if the status is non-zero, then the user timer-interrupt vector is restored and control is passed to the bootstrap vector by INT hex 19. The sector number is two bytes long with the least-significant byte first in the received data.

The received program code is inserted in memory starting at location hex 07C0:0000 and continuing upward. The end of the program code is determined when a frame is received that does not contain 512 bytes of program code.

9. The above two steps are repeated until the end of the program code is received. The user timer-interrupt vector is restored and control passes to the loaded program by a jump to hex 07C0:0000.

Notes:

1. The Remote IPL function is performed, even if local drives are attached, if the Remote IPL switch on Cluster Adapter 1 is On. Remote IPL is supported only for Cluster Adapter 1. The Remote IPL function can be stopped by pressing Control Break, and normal loading from local diskette drives occurs.

2. For every block of data received, an arrow rotates in a clockwise direction on the screen.

3. After power on or system reset, the cursor is moved to the right three columns for about 1 second. Special ROM diagnostic tests for the adapter can be executed by immediately pressing "Ctrl D" on the keyboard. Also, a request to load a general diagnostic program over the cluster can be selected by pressing "Ctrl L" at which time a blinking L is displayed. The adapter sends out a broadcast frame requesting a diagnostic program load. (The first data byte of the request frame is set to hex 90.)

DLCP BIOS Commands

The DLCP BIOS commands are issued by the higher layer communication program to send and receive information through the cluster. The following are the DLCP BIOS commands:

Command Number (Hex)	Command Name
00	Cluster Initialization
01	Receive Virtual Frame
02	Receive Frame
03	Transmit Frame
04	Reserved
05	Display Cluster Status
06	Cluster Status
07	Status
08	Broadcast Frame
09	Transmit Virtual Frame
0A	Stop DLCP
0B	Read Station Address
0C	Set Multicast Address
0D	Check Command In Progress
0E	Read IPL Switch
0F	Start DLCP
10	Dump Statistics
11	Diagnostic Function 1
12	Diagnostic Function 2
13	Diagnostic Function 3
14	Diagnostic Function 4
15	Diagnostic Function 5
16	Diagnostic Function 6
17	Diagnostic Function 7

DLCP BIOS Commands

DLCP Return Codes

The following table indicates the Return Codes that are defined for the cluster DLCP:

DLCP Return Codes	
Return Code	**Meaning**
Hex 00	Successful Completion
Hex 30	Initialization failed
Hex 31	Cluster busy timeout (carrier sense active for 2 seconds)
Hex 32	Duplicate station address on cluster
Hex 33	No response from destination
Hex 34	Frame rejected at destination
Hex 35	Reserved
Hex 36	Cluster access timeout (could not gain access to cluster within a 13 second timeout)
Hex 37	Information field too long (more than 578 bytes)
Hex 38	Information field empty
Hex 39	DLCP command in progress
Hex 3A	Initialization required
Hex 3B	Received frame not available
Hex 3C	Error detected with 8031 (due to command timeout or other processor interface error
Hex 3D	Extended return code in cluster status field
Hex 3E	Invalid initialization parameters (too many or too large buffers specified)
Hex 3F	Previous DLCP BIOS command initiated with Initiate Transmit bit set is not complete

Cluster DLCP Return Codes

Note: A return code of hex 00 indicates successful completion of the DLCP BIOS command. Most of the other return codes indicate error conditions.

Cluster Initialization (DLCP) = Hex 00

Function: This command initializes the DLCP and also transmits an initializing frame to inform others in the cluster. If another station in the cluster has the same address as this station, it sends a response frame indicating duplicate station address, and the return code is hex 32. The Initialization Control Block (ICB) must be built by the calling program with the initialization values indicated by the following:

Return Code	Definition
hex 00	Successful completion
hex 30	Initialization failed
hex 32	Duplicate station address in the cluster
hex 39	Command in progress
hex 3C	Error with 8031
hex 3E	Invalid initialization parameters

Link Control Block (LCB)		
Field	**Value at Entry**	**Value at Exit**
Destination	Don't Care	Unchanged
Source	Don't Care	Unchanged
Command	= 00 (Hex)	Unchanged
Buffer 1 Length	= 0F (Hex)	Unchanged
Buffer 1 Address	Address of Initialization Control Block (ICB)	Unchanged
Buffer 2 Length	Don't Care	Unchanged
Buffer 2 Address	Don't Care	Unchanged
Return Code	Don't Care	Set to Return Code
Cluster Status	Don't Care	Unchanged
Select Adapter	= 0 for Adapter 1 = 1 for Adapter 2 = 2 for Adapter 3 = 3 for Adapter 4	Unchanged

Cluster Initialization (DLCP) = Hex 00

Initialization Control Block (ICB)

The calling program must set the buffer 1 address field in the LCB to the address of an initialization control block (ICB). The figure below shows the composition and bytes that make up the ICB:

Byte	Byte Definition	Value
0	(Bits) 7 6 5 4 3 2 1 0 (Value) 0 0 0 0 0 NVB MM1 MM2	0
1	Number of Large Buffers	4
2	Number of Small Buffers	10
3	Large Buffer Size	584 ÷ 8
4	Small Buffer Size	40
5	Maximum Number of Retries for No Response	8
6	Maximum Number of Retries for Rejected Frame	2
7	Transmit Access Window (TAW)	40 ÷ 2
8	Time Period Reserved for Response	200 ÷ 20
9	Time from Frame Start to First Byte	150 ÷ 2
10	Time Between Control Field and Data Field	100 ÷ 2
11	Timeout Waiting for Response to be Received	300 ÷ 6
12	Timeout Waiting for Next Byte to be Received	300 ÷ 6
13	Timeout Waiting for Command to Complete	7
14	Timeout Waiting to Access Cluster	200

Initialization Control Block (ICB)

Byte 0 - Bits 7, 6, 5, 4, and 3 are reserved and must be set to 0.

Bit 2 - No Virtual Buffer (NVB), when set to zero, allocates a receive buffer for Virtual Frames.

Bit 1 and 0 - These bits are set to enable the first portion of all frames to be received (even if they are not addressed to this station).

The following figure shows the Monitor Mode (MM) bit definitions:

MM1	MM0	Monitor Mode Condition
0	0	Normal Mode
0	1	Receives All Frames on Cluster
1	0	Invalid Setting
1	1	Receives Only Frames from or to Multicast Address or This Station Address

Monitor Mode Bit Definitions

Note: In Monitor Mode, only the first portion of a frame (up to the size of the small buffer minus 7 bytes) is received. The first byte is set to the value of Transmit Window Token, and the second byte corresponds to the first data byte of the information field of the frame.

Byte 1 - This byte indicates the number of large buffers allocated in the 8031 RAM for incoming frames.

Byte 2 - This byte indicates the number of small buffers allocated in the 8031 RAM for incoming frames.

Byte 3 - This byte indicates the large buffer size (each unit represents 8 bytes). Six bytes of the large buffer are reserved for control information.

Byte 4 - This byte indicates the small buffer size (each unit represents 1 byte). Six bytes of the small buffer are reserved for control information.

Byte 5 - This byte indicates the maximum number of times a frame is transmitted with no response from the destination station.

Byte 7 - This byte is used to specify the Transmit Access Window (TAW) time period in μs. For a 40 μs TAW, set this byte to 20. After every transmitted frame, an Access Time Period is allocated, which is 64 times the TAW time period.

Byte 8 - The value of this byte times TAW divided by 2 equals the amount of time (μs.) reserved after each frame for a response frame to be transmitted.

Byte 9 - The value of this byte times 2 equals the delay in μs after the start of a transmit frame before the first byte (destination) is transmitted.

Byte 10 - The value of this byte times 2 equals the delay in μs between the control field and data field of a frame.

Byte 11 - The value of this byte times 6 equals the time allowed in μs for a response frame to be received.

Byte 12 - The value of this byte times 6 equals the time allowed in μs for the next byte of a frame to be received.

Byte 13 - The value of this byte times 16.7 equals the number of seconds allowed for any command in progress to finish before the 8031 indicates error hex 3C to the Cluster Adapter BIOS code.

Byte 14 - The value of this byte times 67 milliseconds equals the amount of time allowed waiting to access the cluster before error hex 36 is returned.

Receive Virtual Frame = Hex 01

Function: This command is used to retrieve a data frame sent by the disk server (using Transmit Virtual Frame).

Notes:

1. There is only one virtual frame buffer for this type of data frame.

2. The destination, command, and cluster status fields in the LCB are modified.

Return Code	Destination
hex 00	Successful completion
hex 32	Duplicate station address in the cluster
hex 37	Information field too long
hex 38	No information field present
hex 39	Command in progress
hex 3A	Initialization required
hex 3B	No receive frame exists
hex 3C	Error detected with 8031

Link Control Block (LCB)		
Field	**Value at Entry**	**Value at Exit**
Destination	Don't Care	Destination
Source	Don't Care	Source
Command	= 01 (Hex)	Frame Control
Buffer 1 Length	Length of Calling Program's Buffer 1	Length of Received Data if Less Than Buffer 1 Length
Buffer 1 Address	Points to Calling Program's Buffer 1	Unchanged
Buffer 2 Length	Length of Calling Program's Buffer 2	Length of Received Data Placed in This Buffer
Buffer 2 Address	Points to Calling Program's Buffer 2	Unchanged
Return Code	Don't Care	Set to Return Code
Cluster Status	Don't Care	Frame Sequence
Select Adapter	= 0 for Adapter 1 = 1 for Adapter 2 = 2 for Adapter 3 = 3 for Adapter 4	Unchanged

Receive Virtual I/O Frame = Hex 01

Receive Frame (from FIFO queue) = Hex 02

Function: This command is used to retrieve a data frame sent from another station (using Transmit Frame) from the First-In-First-Out (FIFO) queue.

The FIFO queue can contain four full size frames and 10 small frames.

Note: The field's destination, command, and cluster status in the LCB are modified.

Note: If the adapter is in Monitor mode, the first byte returned is the Transmit Window Token. The second byte is the first data byte of the information field of the received frame.

Return Code	Definition
hex 00	Successful completion
hex 32	Duplicate station address in the cluster
hex 37	Information field too long
hex 38	No information field present
hex 39	Command in progress
hex 3A	Initialization required
hex 3B	No receive frame exists
hex 3C	Error detected with 8031

Link Control Block (LCB)		
Field	**Value at Entry**	**Value at Exit**
Destination	Don't Care	Destination
Source	Don't Care	Source
Command	= 02 (Hex)	Frame Control
Buffer 1 Length	Length of Calling Program's Buffer 1	Length of Received Data if Less Than Buffer 1 Length
Buffer 1 Address	Points to Calling Program's Buffer 1	Unchanged
Buffer 2 Length	Length of Calling Program's Buffer 2	Length of Received Data Placed in This Buffer
Buffer 2 Address	Points to Calling Program's Buffer 2	Unchanged
Return Code	Don't Care	Set to Return Code
Cluster Status	Don't Care	Frame Sequence
Select Adapter	= 0 for Adapter 1 = 1 for Adapter 2 = 2 for Adapter 3 = 3 for Adapter 4	Unchanged

Receive Frame (from FIFO Queue) = Hex 02

Transmit Frame = Hex 03

Function: This command is used to transmit a data frame to another station where it can be retrieved by using the Receive Frame command.

Note: See also "Special Transmit Mode Command Bits" on page 81

Return Code	Definition
hex 00	Successful completion
hex 31	Cluster always busy
hex 32	Duplicate station address in the cluster
hex 33	No response from destination
hex 34	Exceed allowed number of rejected frames
hex 36	Cluster access time-out
hex 37	Information field too long (frame is not sent)
hex 38	No information field present (frame is not sent)
hex 39	Command in progress
hex 3A	Initialization required
hex 3C	Error detected with 8031

Link Control Block (LCB)		
Field	**Value at Entry**	**Value at Exit**
Destination	Destination	Unchanged
Source	Don't Care	Unchanged
Command	= 03 (Hex)	Unchanged
Buffer 1 Length	Length of Calling Program's Buffer 1	Unchanged
Buffer 1 Address	Points to Calling Program's Buffer 1	Unchanged
Buffer 2 Length	Length of Calling Program's Buffer 2	Unchanged
Buffer 2 Address	Points to Calling Program's Buffer 2	Unchanged
Return Code	Don't Care	Set to Return Code
Cluster Status	Don't Care	Unchanged
Select Adapter	= 0 for Adapter 1 = 1 for Adapter 2 = 2 for Adapter 3 = 3 for Adapter 4	Unchanged

Transmit Frame = Hex 03

Display Cluster Status = Hex 05

Function: This command is used to determine and then display the cluster status. The On/Off status of 64 stations is displayed. Stations that have the Power switch set to On are displayed in reverse video. Your station is displayed in reverse video and blinking. If another station in the cluster has the same address as your station, a long beep sounds. Only those stations that are initialized can be displayed.

Note: The screen should be cleared before issuing this command.

Note: Type of status (destination field):

hex 00 = report stations that are On

hex FF = report stations that are initialized

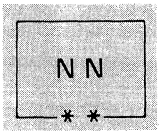
Return Code	Definition
hex 00	Successful completion
hex 31	Cluster always busy
hex 36	Cluster access time-out
hex 39	Command in progress
hex 3C	Error detected with 8031

Link Control Block (LCB)		
Field	**Value at Entry**	**Value at Exit**
Destination	Type of Status	Unchanged
Source	Don't Care	Unchanged
Command	= 05 (Hex)	Unchanged
Buffer 1 Length	Number of Stations to Display	Unchanged
Buffer 1 Address	Don't Care	Unchanged
Buffer 2 Length	Don't Care	Unchanged
Buffer 2 Address	Don't Care	Unchanged
Return Code	Don't Care	Set to Return Code
Cluster Status	Don't Care	Extended Return Code on Error
Select Adapter	= 0 for Adapter 1 = 1 for Adapter 2 = 2 for Adapter 3 = 3 for Adapter 4	Unchanged

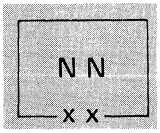
Display Cluster Status = Hex 05

This page explains the cluster status that may appear on your screen.

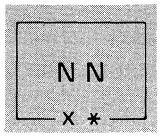
NN is any station address from 0 to 63.



The station you are using is indicated on the screen in blinking reverse video, and the box is marked by two asterisks.



Stations that have their Power switches set to On are displayed in reverse video, and their boxes are marked by two Xs.



Another station has the same address as your station; a long beep sounds every 3 seconds, the box is displayed in blinking reverse video, and is marked by an X and an asterisk.

A station address not in the cluster is indicated by a box displayed in normal video and not marked with Xs or asterisks.

Cluster Status = Hex 06

Function: This command determines the stations' On/Off status. The status bytes are stored in buffer 1 (as determined by the buffer 1 pointer in the LCB). The first byte's least-significant bit is the status of station 0. Bit 1 represents station 1. The least-significant bit of the second byte is the status of station 8, and so on. The number of stations checked is a parameter of this command. Only those stations that are initialized are reported.

Notes:

1. Type of status (destination field):
   hex 00 = report stations that are On
   hex FF = report stations initialized
2. See also "Special Transmit Mode Command Bits" on page 81
3. The size of the buffer required to store the cluster status bytes is (number of stations to check + 7)÷8.

Return Code	Definition
hex 00	Successful completion
hex 31	Cluster always busy
hex 36	Cluster access time-out
hex 39	Command in progress
hex 3C	Error detected with 8031

Link Control Block (LCB)		
Field	**Value at Entry**	**Value at Exit**
Destination	Type of Status	Unchanged
Source	Don't Care	Unchanged
Command	= 06 (Hex)	Unchanged
Buffer 1 Length	Number of Stations to Check	Unchanged
Buffer 1 Address	Points to Calling Program's Buffer 1	Unchanged
Buffer 2 Length	Don't Care	Unchanged
Buffer 2 Address	Don't Care	Unchanged
Return Code	Don't Care	Set to Return Code
Cluster Status	Don't Care	Extended Return Code on Error
Select Adapter	= 0 for Adapter 1 = 1 for Adapter 2 = 2 for Adapter 3 = 3 for Adapter 4	Unchanged

Cluster Status = Hex 06

Status = Hex 07

Function: This command is used to return the status of the connection with a particular station.

Return Code	Definition
hex 00	Successful completion
hex 39	Command in progress
hex 3C	Error detected with 8031

Link Control Block (LCB)		
Field	**Value at Entry**	**Value at Exit**
Destination	Stations for Which Status is Desired	Unchanged
Source	Don't Care	Unchanged
Command	= 07 (Hex)	Unchanged
Buffer 1 Length	Don't Care	Unchanged
Buffer 1 Address	Don't Care	Unchanged
Buffer 2 Length	Don't Care	Unchanged
Buffer 2 Address	Don't Care	Unchanged
Return Code	Don't Care	Set to Return Code
Cluster Status	Don't Care	Cluster Status
Select Adapter	= 0 for Adapter 1 = 1 for Adapter 2 = 2 for Adapter 3 = 3 for Adapter 4	Unchanged

Status = Hex 07

Transmit Broadcast Frame = Hex 08

Function: This command is used to transmit a data frame to another station where it can be retrieved by using the Receive Frame command. No acknowledgment to the frame is sent by the receiving stations.

Note: Transmit Frame and Transmit Virtual Frames are converted to Broadcast Frames if the destination station number is greater than 127.

Note: See also "Special Transmit Mode Command Bits" on page 81..

Return Code	Definition
hex 00	Successful completion
hex 31	Cluster always busy
hex 32	Duplicate station address in cluster
hex 36	Cluster access time-out
hex 37	Information field too long (frame is not sent)
hex 38	No information field present (frame is not sent)
hex 39	Command in progress
hex 3A	Initialization required
hex 3C	Error detected with 8031

Link Control Block (LCB)		
Field	**Value at Entry**	**Value at Exit**
Destination	Destination	Unchanged
Source	Don't Care	Unchanged
Command	= 08 (Hex)	Unchanged
Buffer 1 Length	Length of Calling Program's Buffer 1	Unchanged
Buffer 1 Address	Points to Calling Program's Buffer 1	Unchanged
Buffer 2 Length	Length of Calling Program's Buffer 2	Unchanged
Buffer 2 Address	Points to Calling Program's Buffer 2	Unchanged
Return Code	Don't Care	Set to Return Code
Cluster Status	Don't Care	Unchanged
Select Adapter	= 0 for Adapter 1 = 1 for Adapter 2 = 2 for Adapter 3 = 3 for Adapter 4	Unchanged

Transmit Broadcast Frame = Hex 08

Transmit Virtual Frame = Hex 09

Function: This command is used to transmit a data frame containing sector information from the disk server station. The information can be retrieved only by using the Receive Virtual Frame command.

Note: See "Special Transmit Mode Command Bits" on page 81

Return Code	Definition
hex 00	Successful completion
hex 31	Cluster always busy
hex 32	Duplicate station address in cluster
hex 33	No response from destination
hex 34	Frame rejected at destination
hex 36	Cluster access time-out
hex 37	Information field too long (frame is not sent)
hex 38	No information field present (frame is not sent)
hex 39	Command in progress
hex 3A	Initialization required
hex 3C	Error detected with 8031

Link Control Block (LCB)		
Field	**Value at Entry**	**Value at Exit**
Destination	Destination	Unchanged
Source	Don't Care	Unchanged
Command	= 09 (Hex)	Unchanged
Buffer 1 Length	Length of Calling Program's Buffer 1	Unchanged
Buffer 1 Address	Points to Calling Program's Buffer 1	Unchanged
Buffer 2 Length	Length of Calling Program's Buffer 2	Unchanged
Buffer 2 Address	Points to Calling Program's Buffer 2	Unchanged
Return Code	Don't Care	Set to Return Code
Cluster Status	Don't Care	Unchanged
Select Adapter	= 0 for Adapter 1 = 1 for Adapter 2 = 2 for Adapter 3 = 3 for Adapter 4	Unchanged

Transmit Virtual Disk Frame = Hex 09

Stop DLCP = Hex 0A

Function: This command is used to temporarily inhibit the DLCP from receiving or transmitting frames. Issue a Start DLCP command to leave the stopped state.

Return Code	Definition
hex 00	Successful completion
hex 39	Command in progress
hex 3A	Initialization required
hex 3C	Error detected with 8031

Link Control Block (LCB)		
Field	**Value at Entry**	**Value at Exit**
Destination	Don't Care	Unchanged
Source	Don't Care	Unchanged
Command	= 0A (Hex)	Unchanged
Buffer 1 Length	Don't Care	Unchanged
Buffer 1 Address	Don't Care	Unchanged
Buffer 2 Length	Don't Care	Unchanged
Buffer 2 Address	Dont' Care	Unchanged
Return Code	Don't Care	Set to Return Code
Cluster Status	Don't Care	Unchanged
Select Adapter	= 0 for Adapter 1 = 1 for Adapter 2 = 2 for Adapter 3 = 3 for Adapter 4	Unchanged

Read Station Address = Hex 0B

Function: This command is used to return the address and state of the remote IPL switch of this station.

Return Code	Definition
hex 00	Successful completion
hex 39	Command in progress
hex 3C	Error detected with 8031

Link Control Block (LCB)		
Field	**Value at Entry**	**Value at Exit**
Destination	Don't Care	Unchanged
Source	Don't Care	This station's address
Command	= 0B (Hex)	Unchanged
Buffer 1 Length	Don't Care	Unchanged
Buffer 1 Address	Don't Care	Unchanged
Buffer 2 Length	Don't Care	Unchanged
Buffer 2 Address	Don't Care	Unchanged
Return Code	Don't Care	Set to Return Code
Cluster Status	Don't Care	00 = No IPL FF = IPL
Select Adapter	= 0 for Adapter 1 = 1 for Adapter 2 = 2 for Adapter 3 = 3 for Adapter 4	Unchanged

Read Address = Hex 0B

Set Multicast Address = Hex 0C

Function: This command is used to set the desired multicast address. The multicast address is a variation of the broadcast address (hex FF). More than one station may be assigned the same multicast address. A default value of hex FF is set when a cluster Initialization command is issued to the DLCP. A frame sent, using the Transmit Broadcast Frame command (8), to the group multicast address is received by all stations that share the multicast address.

Return Code	Definition
hex 00	Successful completion
hex 39	Command in progress
hex 3C	Error detected with 8031

Link Control Block (LCB)		
Field	**Value at Entry**	**Value at Exit**
Destination	Desired Multicast Address	Unchanged
Source	Don't Care	Unchanged
Command	= 0C (Hex)	Unchanged
Buffer 1 Length	Don't Care	Unchanged
Buffer 1 Address	Don't Care	Unchanged
Buffer 2 Length	Don't Care	Unchanged
Buffer 2 Address	Don't Care	Unchanged
Return Code	Don't Care	Set to Return Code
Cluster Status	Don't Care	Unchanged
Select Adapter	= 0 for Adapter 1 = 1 for Adapter 2 = 2 for Adapter 3 = 3 for Adapter 4	Unchanged

Check Inside DLCP Flag = Hex 0D

Function: This command is used to return an indication that a DLCP command is already in progress. This command is necessary only for programs that call DLCP from inside an interrupt routine. If a DLCP command is already in progress, the interrupt routine should return to the interrupted program to allow the current DLCP command to finish.

Return Code	Definition
hex 00	Command not in progress
hex 39	Command in progress
hex 3C	Error detected with 8031

Link Control Block (LCB)		
Field	**Value at Entry**	**Value at Exit**
Destination	Don't Care	Unchanged
Source	Don't Care	Unchanged
Command	= 0D (Hex)	Unchanged
Buffer 1 Length	Don't Care	Unchanged
Buffer 1 Address	Don't Care	Unchanged
Buffer 2 Length	Don't Care	Unchanged
Buffer 2 Address	Don't Care	Unchanged
Return Code	Don't Care	Set to Return Code
Cluster Status	Don't Care	Unchanged
Select Adapter	= 0 for Adapter 1 = 1 for Adapter 2 = 2 for Adapter 3 = 3 for Adapter 4	Unchanged

Check Inside DLCP Flag = Hex 0D

Read IPL Switch = Hex 0E

Function: This command is used to read the state of the Remote IPL switch on the requesting station.

Return Code	Definition
hex 00	Successful completion
hex 39	Command in progress
hex 3C	Error detected with 8031

Link Control Block (LCB)		
Field	**Value at Entry**	**Value at Exit**
Destination	Don't Care	Unchanged
Source	Don't Care	This station's address
Command	= 0E (Hex)	Unchanged
Buffer 1 Length	Don't Care	Unchanged
Buffer 1 Address	Don't Care	Unchanged
Buffer 2 Length	Don't Care	Unchanged
Buffer 2 Address	Don't Care	Unchanged
Return Code	Don't Care	Set to Return Code
Cluster Status	Don't Care	IPL Switch (00 = No IPL FF = IPL)
Select Adapter	= 0 for Adapter 1 = 1 for Adapter 2 = 2 for Adapter 3 = 3 for Adapter 4	Unchanged

Read IPL Switch = Hex 0E

Start DLCP = Hex 0F

Function: This command is used to release the DLCP from the stopped state. It enables the DLCP to receive and transmit frames.

Return Code	Definition
hex 00	Successful completion
hex 39	Command in progress
hex 3A	Initialization required
hex 3C	Error detected with 8031

Link Control Block (LCB)		
Field	**Value at Entry**	**Value at Exit**
Destination	Don't Care	Unchanged
Source	Don't Care	Unchanged
Command	= 0F (Hex)	Unchanged
Buffer 1 Length	Don't Care	Unchanged
Buffer 1 Address	Don't Care	Unchanged
Buffer 2 Length	Don't Care	Unchanged
Buffer 2 Address	Don't Care	Unchanged
Return Code	Don't Care	Set to Return Code
Cluster Status	Don't Care	Unchanged
Select Adapter	= 0 for Adapter 1 = 1 for Adapter 2 = 2 for Adapter 3 = 3 for Adapter 4	Unchanged

Start DLCP = Hex 0F

Dump Statistics = Hex 10

Function: This command is used to transfer the current communication statistics block from the adapter.

Link Control Block (LCB)		
Field	**Value at Entry**	**Value at Exit**
Destination	Don't Care	Unchanged
Source	Don't Care	Unchanged
Command	= 10 (Hex)	Unchanged
Buffer 1 Length	12 bytes	Unchanged
Buffer 1 Address	Points to Calling Program's Buffer 1	Unchanged
Buffer 2 Length	Don't Care	Unchanged
Buffer 2 Address	Don't Care	Unchanged
Return Code	Don't Care	Set to Return Code
Cluster Status	Don't Care	Unchanged
Select Adapter	= 0 for Adapter 1 = 1 for Adapter 2 = 2 for Adapter 3 = 3 for Adapter 4	Unchanged

Dump Statistics = Hex 10

Communication Statistics Block (CSB)

The Cluster Adapter returns information regarding previous activity in the CSB.

Return Code	Definition
hex 00	Successful completion
hex 39	Command in progress
hex 3C	Error detected with 8031

The figure below shows the composition and definition of the CSB bytes:

Byte	Definition
0	Number of Times No Response Received (LSB)
1	Number of Times No Response Received (MSB)
2	Number of Times Frame Rejects Received
3	Number of Control Frames Correctly Received (LSB)
4	Number of Control Frames Correctly Received (MSB)
5	Number of Data Frames Correctly Received (LSB)
6	Number of Data Frames Correctly Received (MSB)
7	Number of Control Frames with CRC Error
8	Number of Data Frames with CRC Error
9	Number of Duplicate Frames Received
10	Number of Received Frames That Were Rejected
11	Number of Transmit Collisions

Communication Statistic Block

Diagnostic Function 1 = Hex 11

Function: This command is used to run an internal diagnostic test.

(Reserved for diagnostic use only.)

Bit 1 Test adapter processor-to-processor interface
Bit 2 Reserved
Bit 3 Test driver and receiver logic (terminating plug required for diagnostic use)
Bit 4 Test interrupt logic (set transmit interrupt status bit)
Bit 5 Test interrupt logic (set receive interrupt status bit)
Bit 6 Clear transmit and receive interrupt status bits (no interrupt)
Bit 7 Set transmit and receive interrupt status bits (no interrupt)

Return Code	Definition
hex 00	Successful completion
hex 39	Command in progress
hex 3C	Error detected with 8031
hex 3D	Error detected by 8031 diagnostic test (reason for error in Cluster Status field)

Link Control Block (LCB)		
Field	**Value at Entry**	**Value at Exit**
Destination	Test Number **	Unchanged
Source	Don't Care	Unchanged
Command	= 11 (Hex)	Unchanged
Buffer 1 Length	Don't Care	Unchanged
Buffer 1 Address	Don't Care	Unchanged
Buffer 2 Length	Don't Care	Unchanged
Buffer 2 Address	Don't Care	Unchanged
Return Code	Don't Care	Set to Return Code
Cluster Status	Don't Care	Extended Return Code
Select Adapter	= 0 for Adapter 1 = 1 for Adapter 2 = 2 for Adapter 3 = 3 for Adapter 4	Unchanged

Diagnostic Function 1 = Hex 11

Note: ** Test number (Destination field)

Diagnostic Function 2 = Hex 12

Function: This command is used to transfer data to the adapter's RAM from a buffer in system memory. The data in buffer 1 is transferred to the address specified by buffer 2 in the 8031 address space.

(Reserved for diagnostic use only.)

Return Code	Definition
hex 00	Successful completion
hex 39	Command in progress
hex 3C	Error detected with 8031

Link Control Block (LCB)		
Field	**Value at Entry**	**Value at Exit**
Destination	Don't Care	Unchanged
Source	Don't Care	Unchanged
Command	= 12 (Hex)	Unchanged
Buffer 1 Length	Length of Calling Program's Buffer 1	Unchanged
Buffer 1 Address	Points to Buffer 1	Unchanged
Buffer 2 Length	Don't Care	Unchanged
Buffer 2 Address	Set Offset to Address in 8031 RAM Space to Place Data	Unchanged
Return Code	Don't Care	Set to Return Code
Cluster Status	Don't Care	Unchanged
Select Adapter	= 0 for Adapter 1 = 1 for Adapter 2 = 2 for Adapter 3 = 3 for Adapter 4	Unchanged

Diagnostic Function 2 = Hex 12

Diagnostic Function 3 = Hex 13

Function: This command is used to transfer data from the adapter's RAM to a buffer in system memory. The data is transferred starting at the address specified by the buffer 2 address (offset) in 8031 memory to buffer 1 in the main system's memory.

(Reserved for diagnostic use only.)

Return Code	Definition
hex 00	Successful completion
hex 39	Command in progress
hex 3C	Error detected with 8031

Link Control Block (LCB)		
Field	**Value at Entry**	**Value at Exit**
Destination	Don't Care	Unchanged
Source	Don't Care	Unchanged
Command	= 13 (Hex)	Unchanged
Buffer 1 Length	Length of Calling Program's Buffer 1	Unchanged
Buffer 1 Address	Points to Buffer 1	Unchanged
Buffer 2 Length	Don't Care	Unchanged
Buffer 2 Address	Set Offset to Address in 8031 RAM Space from Which to get Data	Unchanged
Return Code	Don't Care	Set to Return Code
Cluster Status	Don't Care	Unchanged
Select Adapter	= 0 for Adapter 1 = 1 for Adapter 2 = 2 for Adapter 3 = 3 for Adapter 4	Unchanged

Diagnostic Function 3 = Hex 13

Diagnostic Function 4 = Hex 14

Function: This command is used to transfer data to the 8031's internal RAM from a buffer in system memory. The data in buffer 1 is transferred to the address specified by buffer 2 address in 8031 memory.

(Reserved for diagnostic use only.)

Note: Extreme care must be used to prevent destroying data in the 8031's stack and registers in this internal chip RAM. Also, there are only 128 bytes of RAM.

Return Code	Definition
hex 00	Successful completion
hex 39	Command in progress
hex 3C	Error detected with 8031

Link Control Block (LCB)		
Field	**Value at Entry**	**Value at Exit**
Destination	Don't Care	Unchanged
Source	Don't Care	Unchanged
Command	= 14 (Hex)	Unchanged
Buffer 1 Length	Length of Calling Program's Buffer 1	Unchanged
Buffer 1 Address	Points to Buffer 1	Unchanged
Buffer 2 Length	Don't Care	Unchanged
Buffer 2 Address	Set Offset to Address in 8031 on Chip Space to Place Data	Unchanged
Return Code	Don't Care	Set to Return Code
Cluster Status	Don't Care	Unchanged
Select Adapter	= 0 for Adapter 1 = 1 for Adapter 2 = 2 for Adapter 3 = 3 for Adapter 4	Unchanged

Diagnostic Function 4 = Hex 14

Diagnostic Function 5 = Hex 15

Function: This command is used to transfer data from the 8031's internal RAM to a buffer in system memory. The data is transferred starting at the address specified by buffer 2 address (offset) in 8031 memory to buffer 1 in the main system's memory.

(Reserved for diagnostic use only.)

Return Code	Definition
hex 00	Successful completion
hex 39	Command in progress
hex 3C	Error detected with 8031

Link Control Block (LCB)		
Field	**Value at Entry**	**Value at Exit**
Destination	Don't Care	Unchanged
Source	Don't Care	Unchanged
Command	= 15 (Hex)	Unchanged
Buffer 1 Length	Length of Calling Program's Buffer 1	Unchanged
Buffer 1 Address	Points to Buffer 1	Unchanged
Buffer 2 Length	Don't Care	Unchanged
Buffer 2 Address	Set Offset to Address in 8031 RAM from Which to Get Data	Unchanged
Return Code	Don't Care	Set to Return Code
Cluster Status	Don't Care	Unchanged
Select Adapter	= 0 for Adapter 1 = 1 for Adapter 2 = 2 for Adapter 3 = 3 for Adapter 4	Unchanged

Diagnostic Function 5 = Hex 15

Diagnostic Function 6 = Hex 16

Function: This command is used to execute an 8031 program at the address specified by the buffer 2 address field. A "Call" is made to that address and it is expected that the called program sets the 8031 accumulator to a return code value before returning. This return code is placed in the Cluster Status field if non-zero.

(Reserved for diagnostic use only.)

Return Code	Definition
hex 00	Successful completion
hex 39	Command in progress
hex 3C	Error detected with 8031
hex 3D	Extended return code in cluster status

Link Control Block (LCB)		
Field	**Value at Entry**	**Value at Exit**
Destination	Don't Care	Unchanged
Source	Don't Care	Unchanged
Command	= 16 (Hex)	Unchanged
Buffer 1 Length	Don't Care	Unchanged
Buffer 1 Address	Don't Care	Unchanged
Buffer 2 Length	Don't Care	Unchanged
Buffer 2 Address	Set Offset to Address in 8031 RAM Space where a Callable Program Exists	Unchanged
Return Code	Don't Care	Set to Return Code
Cluster Status	Don't Care	Extended Return Code
Select Adapter	= 0 for Adapter 1 = 1 for Adapter 2 = 2 for Adapter 3 = 3 for Adapter 4	Unchanged

Diagnostic Function 6 = Hex 16

Diagnostic Function 7 = Hex 17

Function: This command is used to transmit any type of frame to another station. For example, a control frame may be sent to another station.

(Reserved for diagnostic use only.)

Return Code	Definition
hex 00	Successful completion
hex 31	Cluster always busy
hex 32	Duplicate station address in cluster
hex 33	No response from destination
hex 34	Exceeded allowed rejected frames
hex 36	Cluster access time-out
hex 37	Information field too long
hex 39	Command in progress
hex 3A	Initialization required
hex 3C	Error detected with 8031

Link Control Block (LCB)		
Field	**Value at Entry**	**Value at Exit**
Destination	Destination	Unchanged
Source	Frame Type	Unchanged
Command	= 17 (Hex)	Unchanged
Buffer 1 Length	Length of Calling Program's Buffer 1	Unchanged
Buffer 1 Address	Points to Calling Program's Buffer 1	Unchanged
Buffer 2 Length	Length of Calling Program's Buffer 2	Unchanged
Buffer 2 Address	Points to Calling Program's Buffer 2	Unchanged
Return Code	Don't Care	Set to Return Code
Cluster Status	Don't Care	Unchanged
Select Adapter	= 0 for Adapter 1 = 1 for Adapter 2 = 2 for Adapter 3 = 3 for Adapter 4	Unchanged

Diagnostic Function 7 = Hex 17

Special Transmit Mode Command Bits

The three most-significant bits in the command field of the LCB have the following meanings for transmit commands:

Name	Bit	Meaning
Initiate Transmit	7	Initiate transmit operation but return before complete with return code set to immediate result.
Finish Transmit	6	Wait for previously started transmit operation to complete. Return with return code in LCB set to result of transmit operation.
Return Status	5	If the transmit operation is complete, the return code is set to hex 00 (transmit operation complete result available). Otherwise the return code is set to hex 3F (transmit operation not complete).

Special Transmit Command Bits

Notes:

1. These special transmit command bits are valid only for the following DLCP BIOS commands:

Transmit Frame	(hex 03)
Cluster Status	(hex 06)
Transmit Broadcast	(hex 08)
Transmit Virtual Frame	(hex 09)

2. A transmit operation started with the Initiate Transmit bit set to 1 must be finished by issuing the same transmit command, with a different LCB and the Finish Transmit bit set to 1. If the immediate return code was not zero, the transmit operation is already complete.

3. If an interrupt handler is being used for receive frames, an interrupt is also generated when the transmit operation is complete for transmit operations initiated with the Initiate Transmit bit set. The Transmit Interrupt status bit is set to 1 to indicate that the transmit operation is complete. This bit is bit 0 of adapter port hex 0792 (for adapter 1).

Interface

System Processor I/O Interface

Four Cluster Adapters can be installed at each station. The Cluster Adapter number is selected by switch positions 1 through 4 of switch block 2. These positions correspond to I/O address bits 10, 11, 12, and 13. An adapter is selected when a select switch is On, and the adapter receives a high level (1) on the corresponding I/O address bit.

Note: High level is 1 and low level is 0.

If multiple Cluster Adapters are installed at a station, each adapter can have only one address select switch set to On. A station cannot have two Cluster Adapters with the same address.

Notes:

1. When more than one address select switch is On, the Cluster Adapter decodes and responds to all I/O addresses selected.
2. Cluster Adapter 1 is the only adapter that decodes and responds to all memory addresses; therefore, if more than one Cluster Adapter is set as number 1 (C1), undesirable results occur.
3. If a Cluster Adapter does not have a select switch set to On, it does not respond.

Cluster Adapter Switch Settings

Cluster Adapter addresses and functions can be selected by two eight-switch dual in-line package (DIP) switch blocks. The following shows the switch assignments:

Notes:

1. Switch 8 of switch block 1 selects remote initial program load (IPL) when in the On position.
2. Switch 7 of switch block 1 is reserved. It must be in the Off position.

Switch	Legend	Function
SW-8	IPL	Remote IPL
SW-7	N/A	Reserved (Must be Off)
SW-6	A5	Station Address Bit 5
SW-5	A4	Station Address Bit 4
SW-4	A3	Station Address Bit 3
SW-3	A2	Station Address Bit 2
SW-2	A1	Station Address Bit 1
SW-1	A0	Station Address Bit 0

Switch Block 1 Bit Assignments

Switch	Legend	Function
SW-8	N/A	Reserved
SW-7	RDY	I/O Channel Ready
SW-6	N/A	Reserved
SW-5	N/A	Reserved
SW-4	C4	Select Adapter 4
SW-3	C3	Select Adapter 3
SW-2	C2	Select Adapter 2
SW-1	C1	Select Adapter 1

Switch Block 2 Bit Assignments

The following shows the station-address switch settings on switch block 1.

Station	Switch Block 1 Switch Settings					
	SW 1	SW 2	SW 3	SW 4	SW 5	SW 6
0	Off	Off	Off	Off	Off	Off
1	On	Off	Off	Off	Off	Off
2	Off	On	Off	Off	Off	Off
3	On	On	Off	Off	Off	Off
4	Off	Off	On	Off	Off	Off
5	On	Off	On	Off	Off	Off
6	Off	On	On	Off	Off	Off
7	On	On	On	Off	Off	Off
8	Off	Off	Off	On	Off	Off
9	On	Off	Off	On	Off	Off
10	Off	On	Off	On	Off	Off
11	On	On	Off	On	Off	Off
12	Off	Off	On	On	Off	Off
13	On	Off	On	On	Off	Off
14	Off	On	On	On	Off	Off

Notes:
1. Bit switches 7 and 8 are not applicable to the station address.
2. "On" represents the closed/on position.
3. "Off" represents the open/off position.

Station Address Switch Settings

Station	Switch Block 1 Switch Settings					
	SW 1	SW 2	SW 3	SW 4	SW 5	SW 6
15	On	On	On	On	Off	Off
16	Off	Off	Off	Off	On	Off
17	On	Off	Off	Off	On	Off
18	Off	On	Off	Off	On	Off
19	On	On	Off	Off	On	Off
20	Off	Off	On	Off	On	Off
21	On	Off	On	Off	On	Off
22	Off	On	On	Off	On	Off
23	On	On	On	Off	On	Off
24	Off	Off	Off	On	On	Off
25	On	Off	Off	On	On	Off
26	Off	On	Off	On	On	Off
27	On	On	Off	On	On	Off
28	Off	Off	On	On	On	Off
29	On	Off	On	On	On	Off
30	Off	On	On	On	On	Off
31	On	On	On	On	On	Off
32	Off	Off	Off	Off	Off	On

Station Address Switch Settings

Station	Switch Block 1 Switch Settings					
	SW 1	SW 2	SW 3	SW 4	SW 5	SW 6
33	On	Off	Off	Off	Off	On
34	Off	On	Off	Off	Off	On
35	On	On	Off	Off	Off	On
36	Off	Off	On	Off	Off	On
37	On	Off	On	Off	Off	On
38	Off	On	On	Off	Off	On
39	On	On	On	Off	Off	On
40	Off	Off	Off	On	Off	On
41	On	Off	Off	On	Off	On
42	Off	On	Off	On	Off	On
43	On	On	Off	On	Off	On
44	Off	Off	On	On	Off	On
45	On	Off	On	On	Off	On
46	Off	On	On	On	Off	On
47	On	On	On	On	Off	On
48	Off	Off	Off	Off	On	On
49	On	Off	Off	Off	On	On
50	Off	On	Off	Off	On	On

Station Address Switch Settings

Station	Switch Block 1 Switch Settings					
	SW 1	SW 2	SW 3	SW 4	SW 5	SW 6
51	On	On	Off	Off	On	On
52	Off	Off	On	Off	On	On
53	On	Off	On	Off	On	On
54	Off	On	On	Off	On	On
55	On	On	On	Off	On	On
56	Off	Off	Off	On	On	On
57	On	Off	Off	On	On	On
58	Off	On	Off	On	On	On
59	On	On	Off	On	On	On
60	Off	Off	On	On	On	On
61	On	Off	On	On	On	On
62	Off	On	On	On	On	On
63	On	On	On	On	On	On

Station Address Switch Settings

The following I/O addresses are assigned to the Cluster Adapters:

Adapter	I/O Address (Hex)	Device
Adapter 1	0790	Adapter Status Register
	0791	Adapter Command/Data (Output)
		Adapter Result/Data (Input)
	0792	Adapter Interrupt Register
	0793	Adapter Reset Control
Adapter 2	0890	Adapter Status Register
	0891	Adapter Command/Data (Output)
		Adapter Result/Data (Input)
	0892	Adapter Interrupt Register
	0893	Adapter Reset Control
Adapter 3	1390	Adapter Status Register
	1391	Adapter Command/Data (Output)
		Adapter Result/Data (Input)
	1392	Adapter Interrupt Register
	1393	Adapter Reset Control
Adapter 4	2390	Adapter Status Register
	2391	Adapter Command/Data (Output)
		Adapter Result/Data (Input)
	2392	Adapter Interrupt Register
	2393	Adapter Reset Control

Cluster Adapter I/O Summary

The Adapter Reset command resets the 8031 and 8255 on a Cluster Adapter by writing a 1 to that adapter's Adapter Reset/Interrupt Disable port address. This sets a 74LS74 latch, which remains set until a 0 is written to the same port. The latch must remain set for a minimum of 2 μs, which is the minimum reset time of the 8031 operating at 12 MHz.

The interrupts on a Cluster Adapter can be disabled by writing a 0 to the Adapter Reset/Interrupt Disable port, when -IOW is active (0).

The Cluster Adapter can drive the I/O Channel Ready line low in synchronization with the system clock when the processor reads from the adapter card. This enables a longer read cycle from the expansion slots. The option is selected by setting the I/O Channel Ready switch (switch 7 of switch block 2) to On.

System Processor Memory Interface

The memory addresses assigned to the Cluster Adapter are hex D0000 through hex D7FFF. These addresses are fully decoded only on adapter 1, and are selected by setting the C1 select switch (SW2-1) to On. Each station must have one Cluster Adapter selected as number 1.

System Processor Interrupt Interface

The Cluster Adapter provides an interrupt interface to the system processor with Interrupt Request 3 (IRQ3) or Interrupt Request 7 (IRQ7). The desired interrupt is selected using the interrupt select jumper on the Cluster Adapter. The selection of the interrupt is dependent on the programming requirements.

The following is a sequence of the interrupt process for adapter 1:

1. The system processor enables interrupts by writing to the adapter interrupt enable register at address hex 0792.

2. Upon receipt of an interrupt condition, the 8031 sends a negative active (0) pulse of 10 μs on the port C bit 0 (PC0) line of the 8255 which is connected to IRQ3 or IRQ7. The low-to-high transition of this line prevents this adapter and other Cluster Adapters in the system from generating further interrupt requests. The 8031 processor also sets either Port C1 (PC1) or Port C2 (PC2) of the 8255 to indicate the source of the interrupt. PC1 corresponds to a transmit interrupt, and PC2 corresponds to a receive interrupt. If both PC1 and PC2 are set, the source of the interrupt is the completion of a Cluster Status command.

3. The system processor reads I/O addresses hex 0792, 0B92, 1392, and 2392 on each Cluster Adapter to determine the cause of the interrupt. After all pending requests are handled, the system processor re-enables interrupts on all desired adapters.

8255 Programmable Peripheral Interface (PPI)

The 8255 is used to provide an asynchronous interface between the system processor and the 8031 Microcomputer without the use of interrupts or direct memory access (DMA).

Port A

Port A is operated in mode 2 as a strobed, bidirectional, I/O bus. In this mode, all eight bits of Port A (PA0 through PA7) are dedicated to data transfer between the microcomputer (8031) and the system processor (8088).

Port B

Port B is operated in mode 0. The low-order six bits (PB0 through PB5) provide the station address, and the high-order bit (PB7) provides the Remote IPL (On/Off) status. Bit 7 (PB6 is reserved). The source of information for Port B is switch block 1. When a bit switch is On, the bit is active (low). The microprocessor code in the 8031 complements the Port B information to produce logical 1 active bits.

Port C

When port C is operated in mode 2, five lines are dedicated as handshaking signals. The following four handshaking signals are used by the Cluster Adapter:

- -Output Buffer Full (-OBF)

  A low signal on the -OBF (PC7) line indicates that the microcomputer (8031) has written data to Port A. -OBF provides status to the adapter status register.

- -Acknowledge (-ACK)

  A low signal on the -ACK (PC6) line enables the tri-state output buffer of Port A to send out data to the system processor (8088); otherwise the output is in a high impedance state.

- Input Buffer Full (IBF)

  A high signal on the IBF (PC5) output indicates that data from the 8088 has been loaded into Port A. IBF provides input to the adapter status register and to the 8031.

- -Strobe Input (-STB)

  A low signal on the -STB (PC4) loads data from the 8088 into Port A.

The following is a summary of the 8255 port signals:

8255 Port Signals			
Bit	Port A Mode 2	Port B Mode 0	Port C Mode 2
7	Data Bit 7	Remote IPL	-OBF
6	Data Bit 6	Reserved	-ACK
5	Data Bit 5	Station Address Bit 5	+IBF
4	Data Bit 4	Station Address Bit 4	-STB
3	Data Bit 3	Station Address Bit 3	Reserved
2	Data Bit 2	Station Address Bit 2	Receive Frame Interrupt
1	Data Bit 1	Station Address Bit 1	TX Complete Interrupt
0	Data Bit 0	Station Address Bit 0	Interrupt Request

Summary of 8255 Port Signals

Cluster Bus Interface

The bus interface consists of a transmitter, receiver, carrier sense circuitry, and internal loopback-mode logic. They are the interface between the 8031 serial port and the 75Ω coaxial cable.

Cluster Adapter Transmitter

The Cluster Adapter transmitter consists of an Am26LS29 tri-state, single-ended, line driver. This driver features a high capacitive-load drive capability with buffered outputs, individual rise-time control, and output short-circuit protection.

To transmit data to the bus, the microprocessor code in the 8031 must first enable the -RTS signal on the port 3 interface. Data can then be sent to the bus bit-by-bit from +TXD on port 3.

The transmitter is electrically isolated from the logic circuits on the Cluster Adapter by an HCPL-2531 high-speed optocoupler, which uses a light-emitting diode and an integrated light detector to obtain electrical insulation.

Cluster Adapter Receiver

The Cluster Adapter receiver consists of an Am26LS34 high-performance, differential line receiver.

The received signal is amplified by a 5535 Operational Amplifier and is provided to the Am26LS34. To receive the digital data, the microprocessor code in the 8031 must ensure that the +Internal Loop signal on port 3 is inactive. Data can then be received bit-by-bit at port 3 from +RXD.

The receiver is also electrically isolated from the logic circuits on the Cluster Adapter by an HCPL-2531 high-speed optocoupler.

Carrier Sense Circuitry

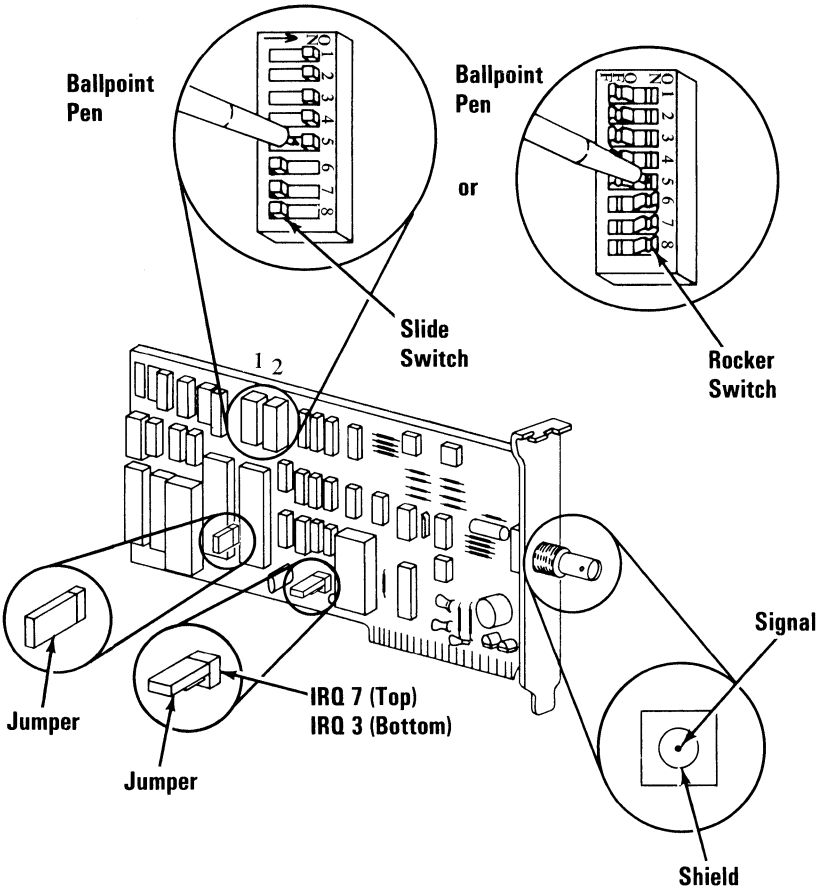
The carrier sense circuitry provides information about the state of the Cluster Adapter. This information is needed to implement the collision avoidance protocol. The amplified signal received from the bus is passed through a comparator to detect the negative voltage state (less than approximately -150 millivolts). This negative portion of the signal is inverted into +NRXD and then ORed with the positive portion (greater than approximately +150 millivolts) of the +RXD signal. The result is then sent to the clear input of a 74LS161 counter. As long as this ORed signal (CLR) is active (0), the counter is held reset. When the signal goes inactive (1), the counter begins counting on the rising edges of the 8031 +ALE signal. On the fourth +ALE pulse, the counter is disabled and the -Carrier Sense signal goes inactive (1). The time delay between the bus going inactive and -Carrier Sense going inactive is 1.5 μs.

Internal Loopback Mode

The Cluster Adapter provides logic to allow the 8031 to receive the data it is transmitting without interference from the bus by wrapping the transmitter to the receiver on the Cluster Adapter.

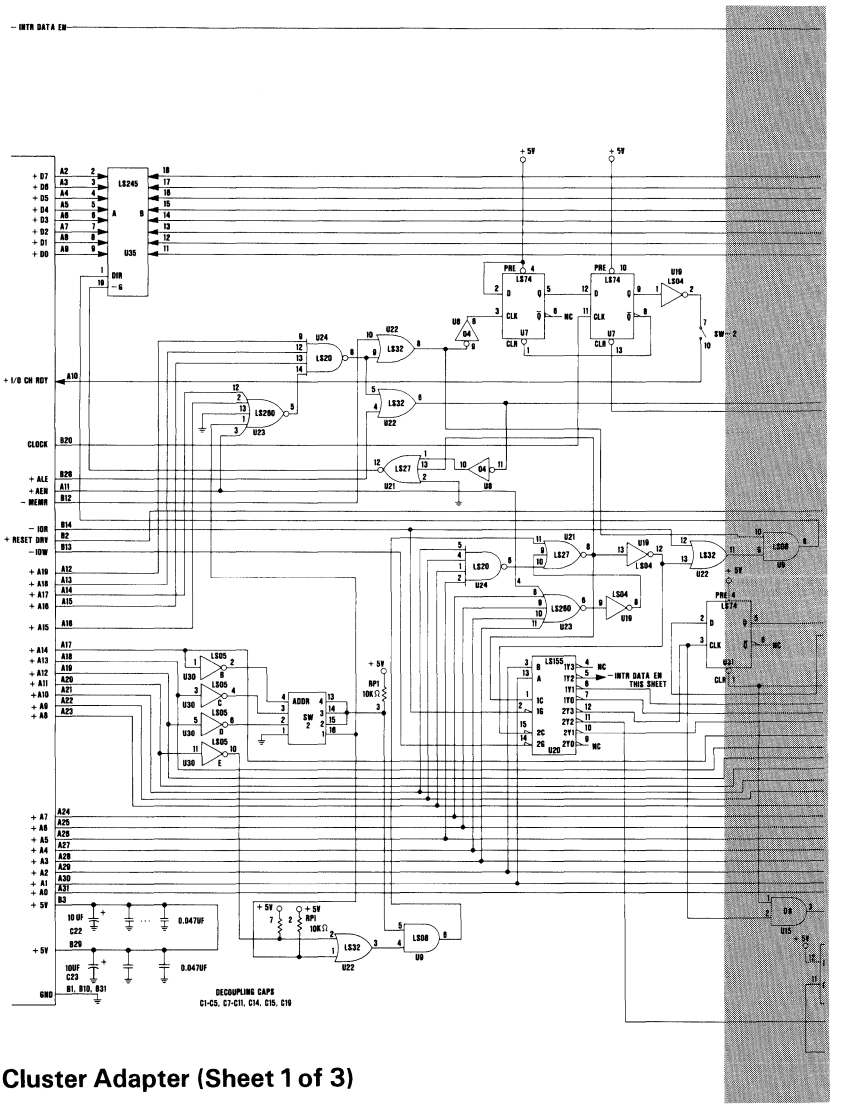
The adapter is placed into internal loopback mode when the 8031 microprocessor code sets the +Internal Loop signal active (1). This mode returns any data transmitted on +TXD to +RXD. Notice that -RTS may or may not be active. If -RTS is active, the data not only returns to +RXD, but also is transmitted to the bus.

Specifications

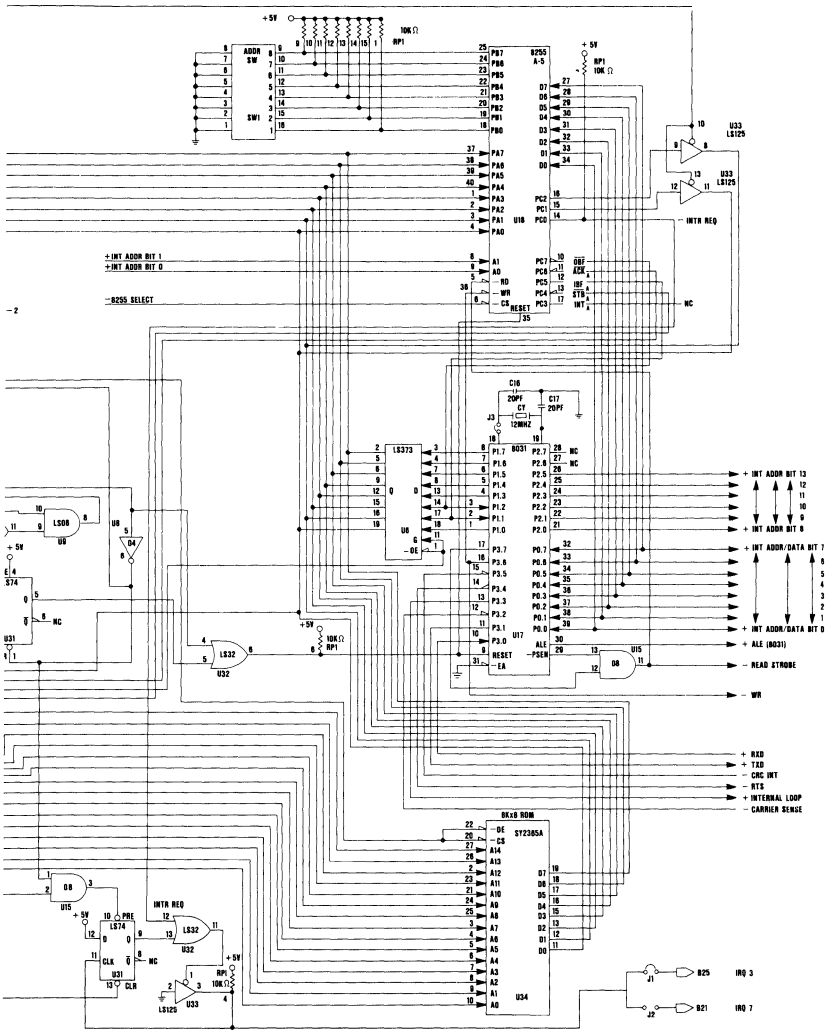


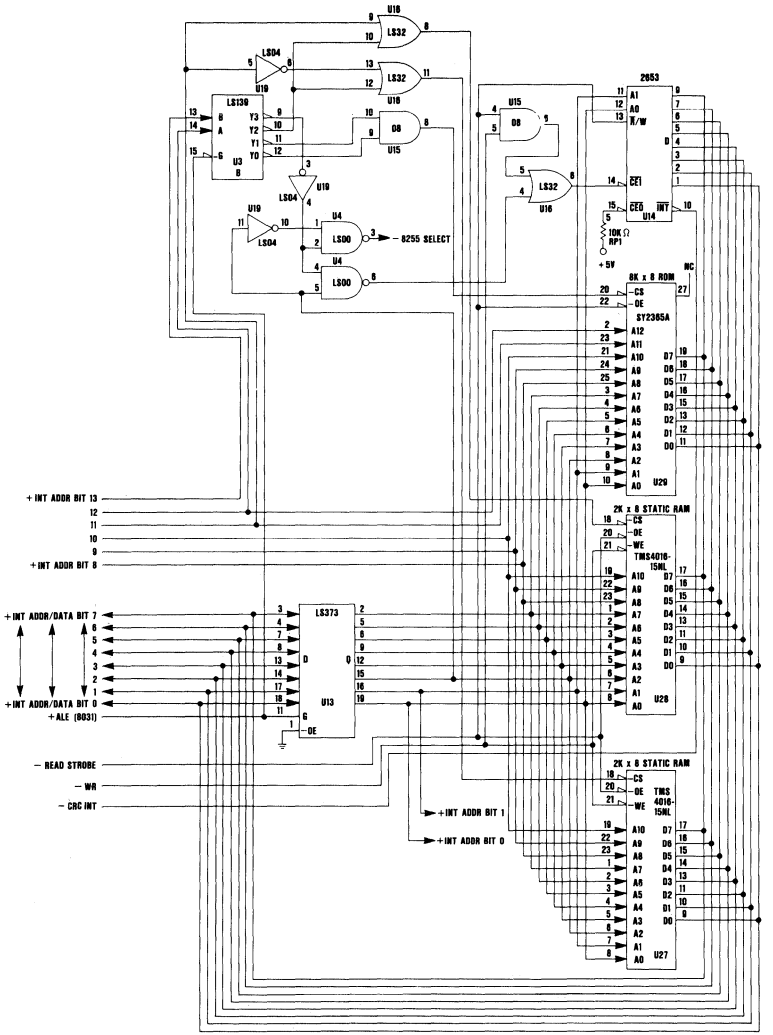
Logic Diagrams

The following pages contain logic diagrams.

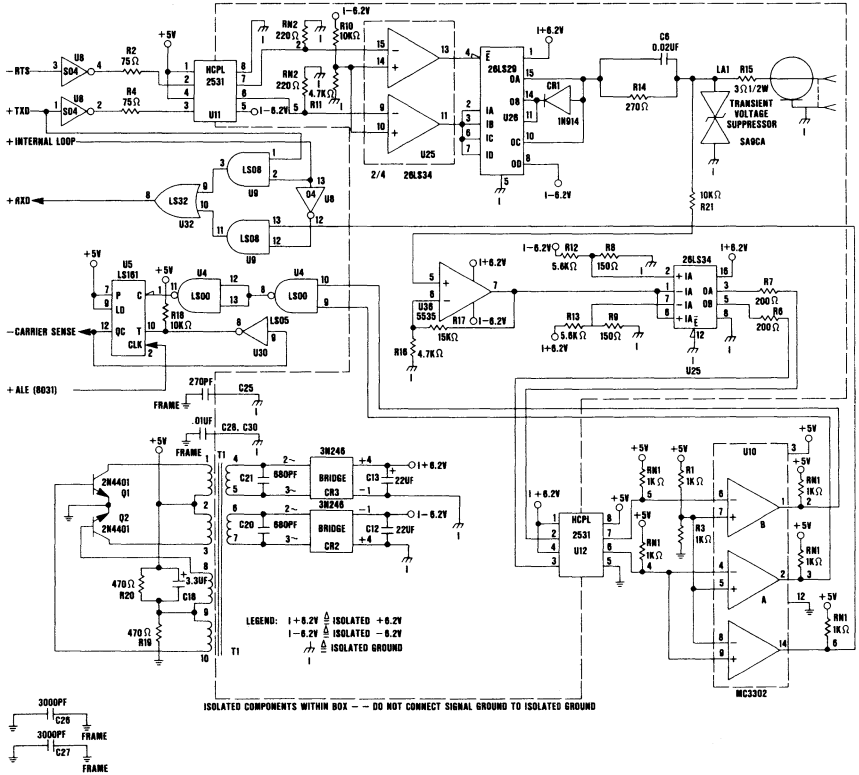


Cluster Adapter (Sheet 1 of 3)


+5V
10K Ω
RP1
ADDR SW
SW1
8255 A-5
U18
RESET
+INT ADDR BIT 1
+INT ADDR BIT 0
-8255 SELECT
INTR REQ
NC
U33 LS125
C16 20PF
C17 20PF
C1 12MHZ
J3
LS373
U6
8031
U17
+INT ADDR BIT 13
+INT ADDR BIT 9
+INT ADDR/DATA BIT 7
+INT ADDR/DATA BIT 0
+ALE (8031)
-READ STROBE
-WR
LS08
U9
LS04
U8
LS74
U31
LS32
U32
8K×8 ROM
SY2365A
U34
U15
+RXD
+TXD
-CRC INT
-RTS
+INTERNAL LOOP
-CARRIER SENSE
J1
J2
B25
B21
IRQ 3
IRQ 7



Cluster Adapter (Sheet 2 of 3)



Cluster Adapter (Sheet 3 of 3)

Index

A

B

C

D

E

F

I

L

M

P

R

S

T

Personal Computer Hardware Reference Library

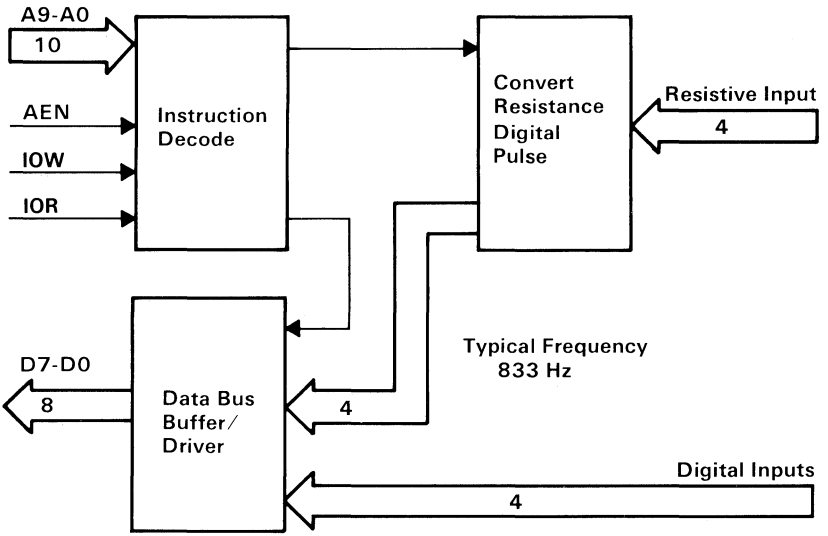
IBM Game Control Adapter

6361493

Contents

Description

The IBM Game Control Adapter allows up to four paddles or two joysticks to be attached to the system. This adapter fits into one of the system board's or expansion board's expansion slots. The game control interface cable attaches to the rear of the adapter. In addition, four inputs for switches are provided. Paddle and joystick positions are determined by changing resistive values sent to the adapter. The adapter, when used with system software, converts the present resistive value to a relative paddle or joystick position. On receipt of an output signal, four timing circuits are started. By determining the time required for the circuit to timeout (a function of the resistance), the paddle position can be determined. This adapter could be used as a general purpose I/O card with four analog (resistive) inputs plus four digital input points.



Game Control Adapter Block Diagram

2 Game Control Adapter

Programming Considerations

Address Decode

The select on the Game Control Adapter is generated by two 74LS138s as an address decoder. AEN must be inactive while the address is hex 201 in order to generate the select. The select allows a write to fire the one-shots, or a read to give the values of the trigger buttons and one-shot outputs.

Data Bus Buffer/Driver

The data bus is buffered by a 74LS244 buffer/driver. For an In from address hex 201, the Game Control Adapter will drive the data bus; at all other times, the buffer is left in the high impedance state.

Trigger Buttons

The trigger button inputs are read by an In from address hex 201. A trigger button is on each joystick or paddle. These values are seen on data bits 7 through 4. These buttons default to an open state and are read as 1. When a button is pressed, it is read as 0. Software should be aware that these buttons are not debounced in hardware.

Joystick Positions

The joystick position is indicated by a potentiometer for each coordinate. Each potentiometer has a range of 0 to 100 kilohms that varies the time constant for each of the four one-shots. As this time constant is set at different values, the output of the one-shot will be of varying durations.

All four one-shots are fired at once by an Out to address hex 201. All four one-shot outputs will go true after the fire pulse and will remain high for varying times depending on where each potentiometer is set.

These four one-shot outputs are read by an In from address hex 201 and are seen on data bits 3 through 0.

I/O Channel Description

A9–A0: Address lines 9 through 0 are used to address the Game Control Adapter.

D7–D0: Data lines 7 through 0 are the data bus.

IOR, IOW: I/O Read and I/O Write are used when reading from or writing to an adapter (In, Out).

AEN: When active, the adapter must be inactive and the data bus driver inactive.

+5 Vdc: Power for the Game Control Adapter.

GND: Common ground.

The following I/O channel lines are not used:

MEMR, MEMW

DACK0–DACK3

IRQ7–IRQ2

DRQ3–DRQ1

I/O CH RDY

I/O CH CK

A19–A10

ALE, T/C

CLK, OSC

-5 Vdc

+12 Vdc

-12 Vdc

RESET DRV

Interface

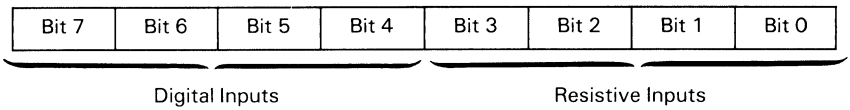
The Game Control Adapter has eight input lines; four digital inputs and four resistive inputs. The inputs are read with one In from address hex 201.

The four digital inputs each have a 1-kilohm pullup resistor to +5 Vdc. With no drives on these inputs, a 1 is read. For a 0 reading, the inputs must be pulled to ground.

The four resistive pullups, measured to +5 Vdc, will be converted to a digital pulse with a duration proportional to the resistive load, according to the following equation:

$$\text{Time} = 24.2\ \mu s + 0.011\ (r)\ \mu s$$

The user must first begin the conversion by an Out to address hex 201. An In from address hex 201 will force the digital pulse to go high and remain high for the duration according to the resistance value. All four bits (bit 3–bit 0) function in the same manner; their digital pulse will all go high simultaneously and will reset independently according to the input resistance value.



The typical input to the Game Control Adapter is a set of joysticks or game paddles.

The joysticks will typically be a set of two (A and B). These will have one or two buttons each with two variable resistances each, with a range of 0 to 100 kilohms. One variable resistance will indicate the X coordinate and the other variable resistance will indicate the Y coordinate.

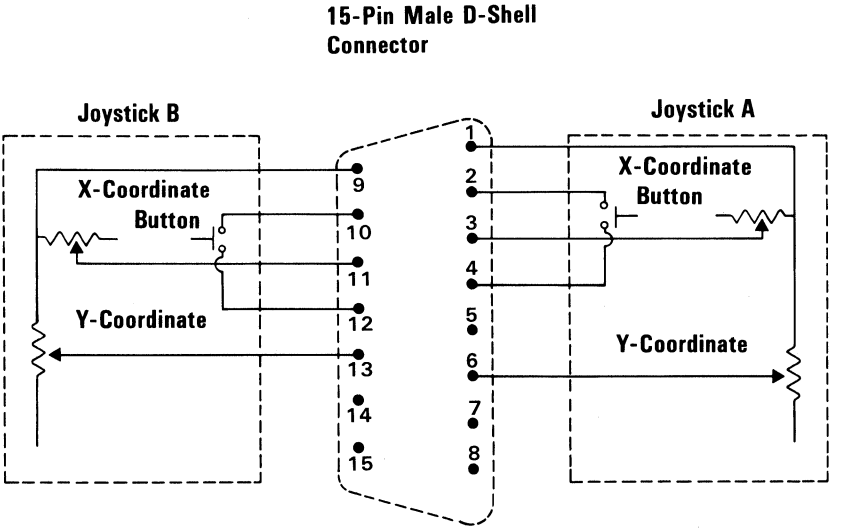
The joystick should be attached to give the following input data:

Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
B-#2 Button	B-#1 Button	A-#2 Button	A-#1 Button	B-Y Coordinate	B-X Coordinate	A-Y Coordinate	A-X Coordinate

The game paddles will consist of two (A and B) or four (A, B, C, and D) paddles. These will have one button each and one variable resistance each, with a range of 0 to 100 kilohms. The game paddles should be attached to give the following input data:

Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
D Button	C Button	B Button	A Button	D Coordinate	C Coordinate	B Coordinate	A Coordinate

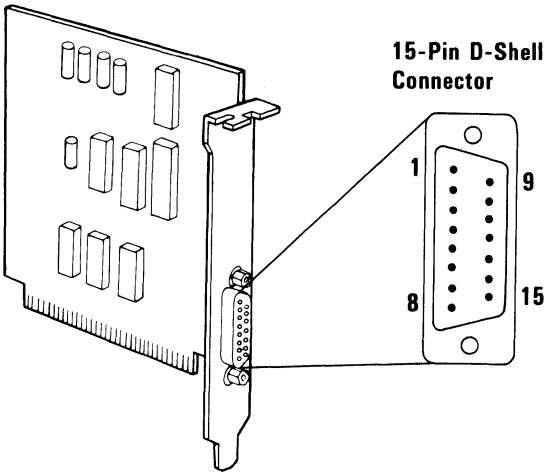
The following "Joystick Schematic Diagram" may be used for attaching game controllers.



Note: Potentiometer for X- and Y-Coordinates has a range of 0 to 100 k-ohms. Button is normally open; closed when pressed.

Joystick Schematic Diagram

Specifications



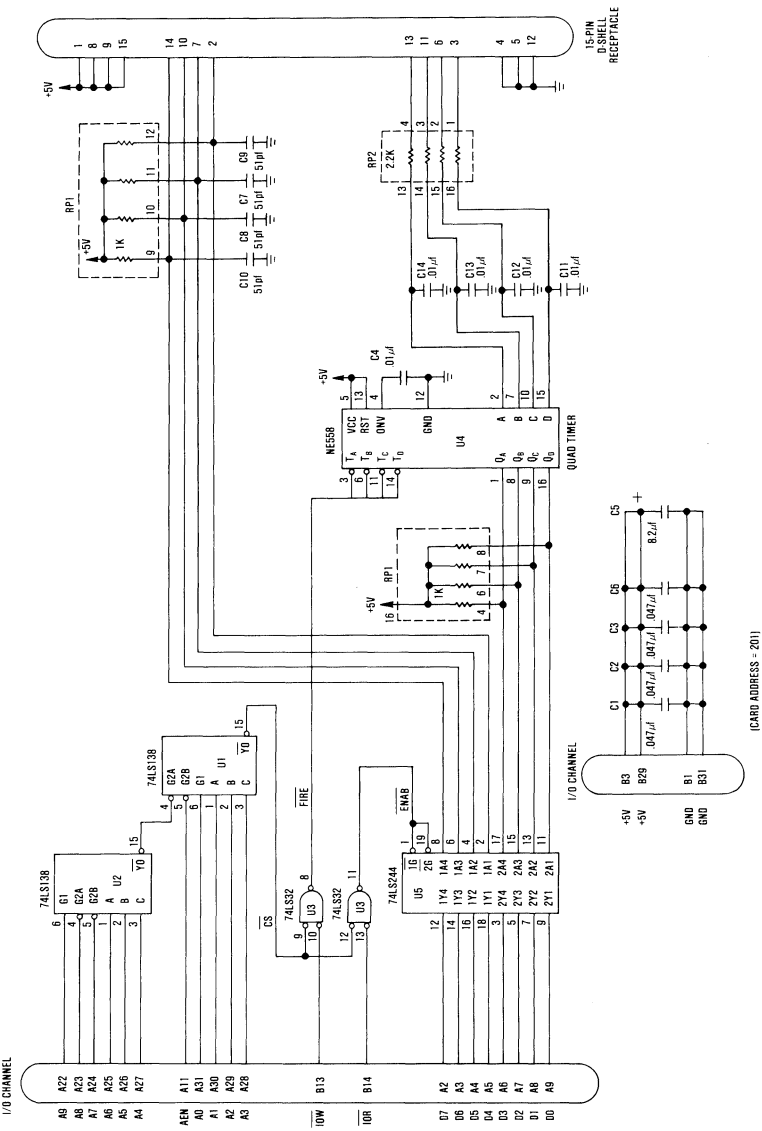
At Standard TTL Levels

	Voltage	Adapter Pin No.	
	+5 Vdc	1	
	Button 4	2	
	Position 0	3	
	Ground	4	
	Ground	5	
	Position 1	6	
	Button 5	7	
External Devices	+5 Vdc	8	Game Control Adapter
	+5 Vdc	9	
	Button 6	10	
	Position 2	11	
	Ground	12	
	Position 3	13	
	Button 7	14	
	+5 Vdc	15	

Connector Specifications

8 Game Control Adapter

Logic Diagram



Game Control Adapter (Sheet 1 of 1)

Notes:

Technical Reference

Options and Adapters
Volume 2

6322509