NorthStar™

ADVANTAGE

Technical Manual

00464B

NorthStar™
North Star Computers, Inc.
14440 Catalina St., San Leandro, CA 94577 USA
(415) 357-8500 TWX/Telex (910) 366-7001

ADVANTAGE™
Technical Manual

00464 B

PREFACE

This manual contains all the technical information required to fully utilize the features of the North Star ADVANTAGE computer. Chapter 1 and 2 contain a brief introduction to the unit and a summary of the operating procedures. Chapter 3 provides the sophisticated user with the programming information and technical details required for writing application programs. Chapter 4 describes the theory of operation of the hardware, and Chapter 5 and 6 support maintenance personnel with maintenance procedures and instructions for using the diagnostic programs. The schematics for the main printed circuit board are found in the appendices, along with other support material.

CONTENTS

CONTENTS

CONTENTS

CONTENTS

CONTENTS

ILLUSTRATIONS

ILLUSTRATIONS

ILLUSTRATIONS

ILLUSTRATIONS

1.1 GENERAL DESCRIPTION

The North Star ADVANTAGE is a high performance Z80 based microcomputer system complete with keyboard, CRT and disk drives housed in a single cabinet. The ADVANTAGE computer is illustrated in Figure 1-1.

The ADVANTAGE Computer

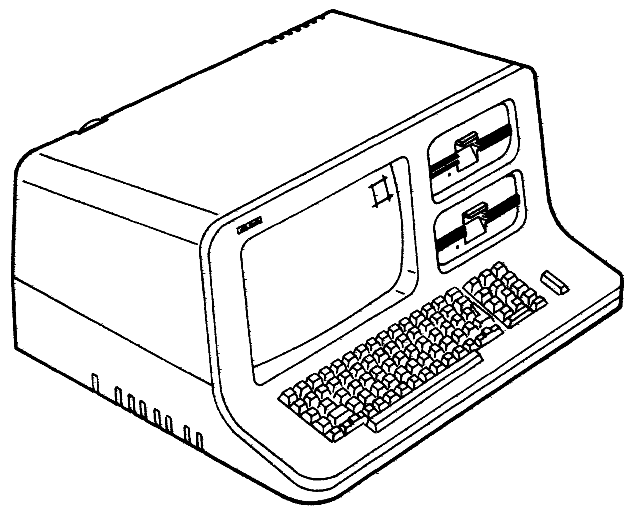


Figure 1-1

The ADVANTAGE contains a 4 MHz Z80A microprocessor with 64K bytes of dynamic RAM for program storage, a separate 20K byte RAM to drive the bit-mapped display and an auxiliary 8035 microprocessor to control the keyboard and floppy disk drives. The display can be operated as a 1920 character display (24 lines by 80 characters) or as a 640 x 240 pixel bit-mapped display, where each pixel is controlled by one bit in the display RAM. The two integrated 5-1/4 inch floppy disks are double-sided, and double-density providing storage of 360K bytes per drive. The keyboard contains 49 standard typewriter keys, 9 symbol or control keys, a 14 key numeric/cursor control pad and 15 user-programmable function keys.

A functional block diagram of the ADVANTAGE computer is shown in Figure 1-2. The blocks are described briefly below. A more detailed description of the ADVANTAGE can be found in Chapter 4, Theory of Operation.

- The Central Control Unit maintains primary control of the system. Contained herein are the Z80 and 8035 processors and the controllers for the I/O devices.
- The 64K Main RAM (Random Access Memory) provides temporary storage of programs and data. Programs are executed while residing in this RAM.
- The Video Monitor and 20K Display RAM produce a high resolution display that can be used for graphics applications, or to display messages for the operator.
- The two Disk Drives use 5-1/4 inch floppy diskettes to store a total of 720K bytes.
- The Speaker produces a tone used to signal the operator. The frequency and duration of the tone is controlled by the program.
- The Keyboard includes the standard typewriter configuration, a numeric keypad and 15 programmable function keys.
- The I/O Board Slots allow the ADVANTAGE to be customized for specific applications. There are six board slots which may contain interface boards for external devices or other boards which expand the computing power of the ADVANTAGE. Two types of boards are presently available for use in this area: the Serial Input/Output (SIO) Board and the Parallel Input/Output (PIO) Board.

Functional Block Diagram

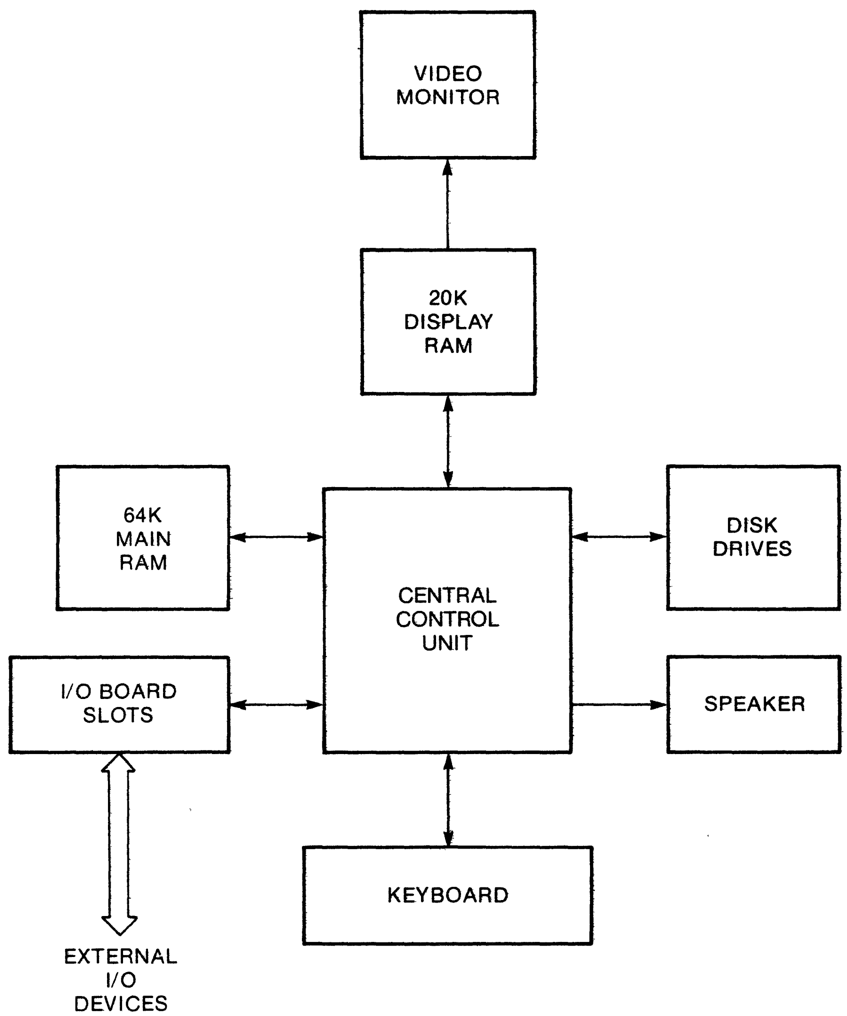


FIGURE 1-2

1.2 SPECIFICATIONS

The ADVANTAGE specifications are given in Table 1-1.

Table 1-1

ADVANTAGE Specifications	
CABINET	
Dimensions	48 cm wide x 51 cm long x 31.5 cm high (18-3/4 in x 20 in x 12-1/2 in)
Net Weight	19.5 kg (43 lbs)
Composition	High impact structural foam
POWER REQUIREMENTS	
External (with Internal Line Filter)	
Domestic	115 VAC, (95 to 135 VAC) 50/60Hz
International	115/230 VAC, (95 to 132 VAC/187 to 265 VAC) 50/60 Hz
Internal Supply Voltages	±5 VDC ±5% ±12 VDC ±5%
Power Consumption	2 amps @ 115V 1 amp @ 230V
TEMPERATURE AND HUMIDITY	
Operating: (with diskette)	$10^\circ C$ to $40^\circ C$ ($50^\circ F$ to $104^\circ F$) 20% to 80% non-condensing
Non-operating	$-40^\circ C$ to $60^\circ C$ ($-40^\circ F$ to $140^\circ F$)
Shipping	$-40^\circ C$ to $52^\circ C$ ($-40^\circ F$ to $125^\circ F$) 5% to 95% non-condensing

Table 1-1 (continued)

PROCESSOR/MEMORY	
CPU	Z80 Microprocessor, operating speed: 4MHz 8035 auxiliary processor for keyboard and disk
Memory	64K byte Main RAM 20K byte Display RAM 2K byte Boot PROM
VIDEO	
Screen	28 cm (11 in) diagonal P31 phosphor (green) High impact, non-glare safety shield
Grid	1920 character display, 24 lines by 80 characters 5X7 character in 8x10 dot matrix
Graphics resolution	240 pixels high x 640 pixels wide
Refresh rate	50 or 60 Hz, depending on line frequency
CRT Anode Voltage	17 KV maximum
KEYBOARD	
Keytops	Sculptured Selectric-compatible N-Key roll-over for fast data entry
Number of Keys:	87
Key Groups	49 Standard Typewriter Keys 14-key Numeric Pad with ENTER key 15 Programmable Function Keys 9 Additional Symbol/Control Keys
Other features	Full Cursor control Special Shift-Lock Keys 5 Shift Modes Auto Repeat

Table 1-1 (continued)

DISK DRIVES	
Number of drives	Two floppy disk drives housed in cabinet
Diskettes	Standard 5-1/4 in floppy diskettes. Recommended type: Dysan part No. 107/2D.
	512 bytes/sector, 10 (hard) sectors/ track 35 tracks/side, 2 sides/diskette
Storage	Quad (double-sided, double-density)
	360K bytes per diskette (formatted)
Transfer Rate	250K bits/second
Latency (average)	100 ms
Access Time Track-to-Track	5 ms
Track Density	48 tpi
Tracks per Side	35
ERROR RATES	
Soft errors	1 per 10^8 bits read
Hard errors	1 per 10^{11} bits read
Seek errors	1 per 10^6 seeks
Disk speed	300 rpm $\pm$ 3.0%

Table 1-1 (continued)

INPUT/OUTPUT	
I/O Bus	Slots for up to six plug-in boards
	Each board addressed by 16 I/O addresses
Serial I/O (SIO)	RS232 Serial Port
	Current loop option
	Asynchronous: 45 baud to 19.2 kilobaud
	Synchronous: 2400 Baud to 51 kilobaud
Parallel I/O	8-bit data in and out with three handshake lines for each port
	Maximum speed is limited by the processor.

2.1 PRELIMINARY INFORMATION

2.1.1 Keyboard

Primary system control is maintained by entering commands and data from the ADVANTAGE keyboard. The keyboard is illustrated in Figure 2-1. There are 87 keys, described in Table 2-1. The keys generate standard ASCII codes, listed in Appendix A.

Display of characters entered from the keyboard is under program control. A program-maintained cursor, the rectangular shaped symbol, marks the position on the

The ADVANTAGE Keyboard

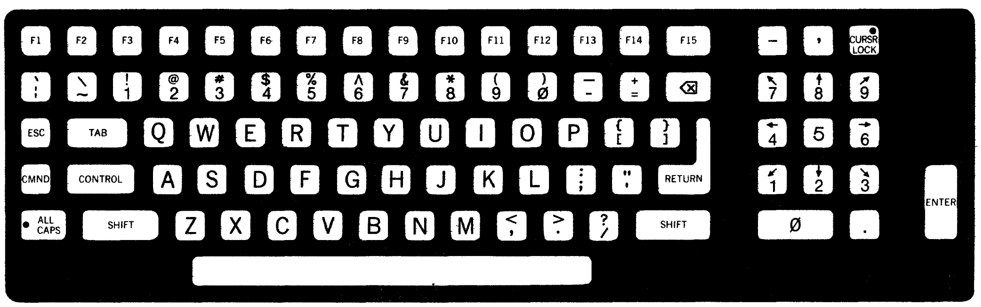


Figure 2-1.

Table 2-1

ADVANTAGE Keys

Key Group	Keys	Description
CHARACTER	ABCDEFGHIJKLM NOPQRSTUVWXYZ 1234567890!@# $%^&*()-_=+;: '",.<>/?[]{} (space)	Alphabetic, numeric, and special symbols. Numbers and three symbols (.,-) are also available on the numeric pad.
KEYBOARD CONTROL	SHIFT	Either of two identical keys which cause most of the other keys to shift into upper case (see Appendix A).
	ALL CAPS	Shifts only alphabetic characters to upper case. Key is a "push on-push off" type with LED to signal when function is active.
	RETURN	Carriage return.
	TAB	Position to next tab set on the line. Setting and releasing tabs is done under program control.
	⌫	Character delete, backspace, or delete and backspace depending upon the program being used.
	ENTER	Numeric pad data entry key.
CURSOR CONTROL	8 direction arrows	All cursor activity is under program control.
	CURSOR LOCK	Shifts only cursor control keys (1-9 on numeric pad) to allow cursor positioning without using SHIFT key. Key is a "push on-push off" type with LED to signal when key is active.

Table 2-1 (continued)

Key Group	Keys	Description
FUNCTION	F1 F2 F3 F4 F5 F6 F7 F8 F9 F10 F11 F12 F13 F14 F15	Special purpose keys entirely under program control. Each Function key can generate up to three ASCII codes.
PROGRAM	ESC	(ESCAPE) key under program control.
	CMND	(Command) operates as a special shift mode for Function keys.
	CONTROL	(CTRL) operates as a special shift for keys.

2.1.2 Rear Panel Controls

A rear view of the ADVANTAGE is shown in Figure 2-2. Table 2-2 describes the controls shown in the figure.

ADVANTAGE Rear View

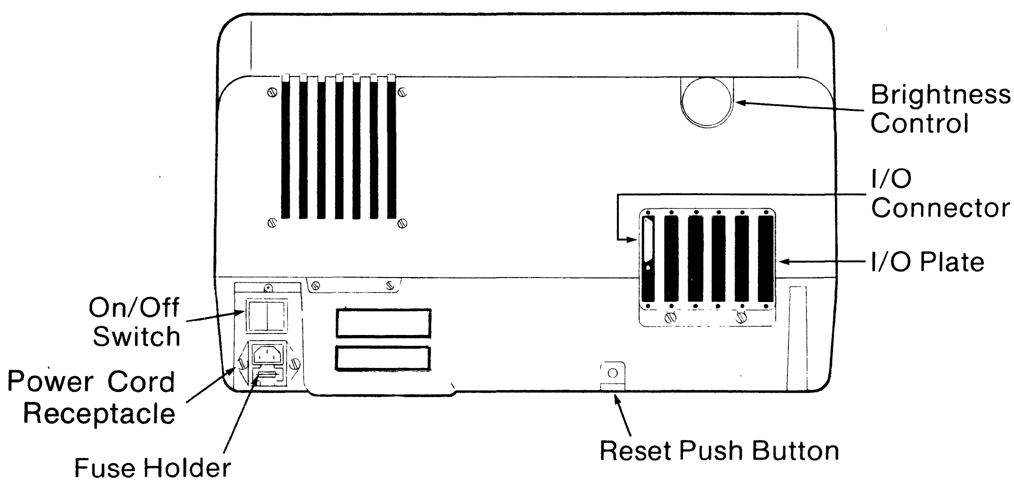


Figure 2-2

Table 2-2

Rear Panel Controls	
Control	Description
ON/OFF Switch	Applies/removes electrical power to the unit.
Power Cord Receptacle	Mates with power cord to provide electric current from AC power source.
Fuse Holder	Contains the AC line fuse. Use 3A fuse for 115V operation and 1.5A fuse for 230V operation.
Reset Push Button	Resets and initializes the system. After reset, data in Main Memory is indeterminate but data on diskettes is not affected.
I/O Plate	Openings in plate allow access to I/O connectors on I/O Boards
Brightness Control	Controls brightness of the display screen. Turn clockwise to increase lightness.

2.1.3 Diskette Loading/Unloading

To load a diskette into one of the disk drives, proceed as follows:

1. Open the latch on the front of the disk drive.
2. Hold the diskette on the label end, with the label facing up and the write protect notch on the left. (see Figure 2-3).
3. Insert the diskette into the drive and push it all the way back until it contacts the rear of the disk slot.
4. Close the latch.

Loading a Diskette

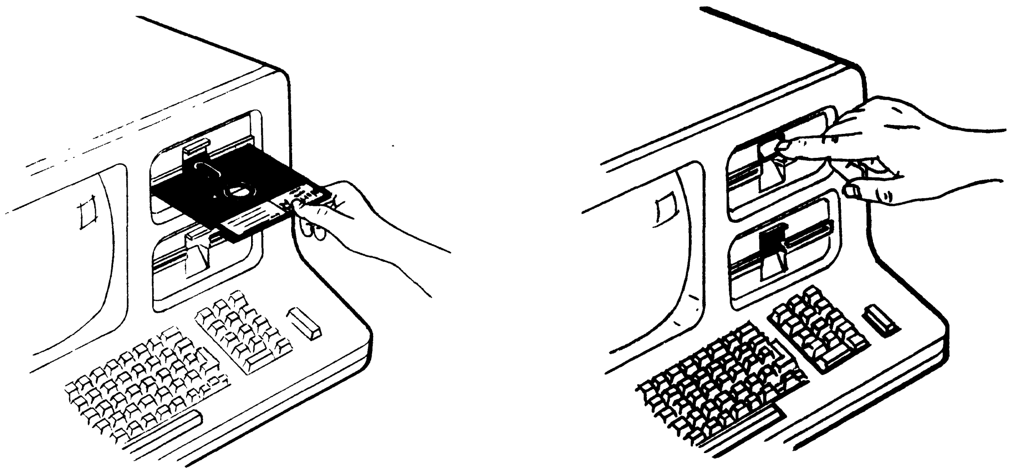


Figure 2-3.

To unload a diskette, proceed as follows:

1. Wait until the red indicator light on the front of the disk drive goes out.

2. Open the latch on the front of the drive.

3. Grasp the edge of the diskette and pull it out.

2.1.4 Keyboard Reset

The ADVANTAGE system may be reset by pressing four keys simultaneously on the keyboard. The keys are: CMND, both SHIFT keys, and $\boxed{\langle x}$. The effect of this reset is equivalent to pushing the Reset Pushbutton on the rear of the ADVANTAGE cabinet (see Section 2.1.2).

When power is first applied to the ADVANTAGE or after the Reset Pushbutton is pressed, the keyboard reset feature is enabled. Thereafter, the feature can be enabled and disabled by the program (see Section 3.3.2 and Table 3-6).

2.2 SYSTEM STARTUP

2.2.1 Booting From Drive 1

To load a program from disk drive 1, proceed as follows:

1. If the ADVANTAGE power is already turned on, skip to step 4.

2. Insure that there are no diskettes in the disk drives. Turning power on or off with diskettes loaded may cause loss of data on the diskettes.

3. Turn on the ADVANTAGE by pressing the ON/OFF switch located at the rear of the cabinet.

4. Load the desired diskette into the upper drive (Drive 1) as described in Section 2.1.3. The diskette must be of the type that can be used for bootstrapping. Typically, a System Diskette or a Diagnostic Diskette is used.

5. Press the RESET button at the rear of the cabinet. The screen displays the message "LOAD SYSTEM" with a cursor positioned below it. This step is not necessary if the ADVANTAGE was just turned on, as the ADVANTAGE automatically resets on power-up.

6. Press the RETURN key. A program is read from Drive 1 and control is turned over to that program.

2.2.2 Booting From Drive 2

The procedure for booting from disk drive 2 is the same as for booting from disk drive 1, except as follows:

1. Load the diskette into drive 2.

2. In step 6 instead of just pressing the RETURN key, press three keys in sequence: D2<RETURN>. Note that when booting from drive 1, the format D1 <RETURN> may also be used.

2.2.3 Booting From Serial Port

To load a program through a serial communication link, proceed as follows:

1. If the ADVANTAGE power is already turned on, skip to step 4.
2. Insure that there are no diskettes in the disk drives. Turning power on or off with diskettes loaded may cause loss of data on the diskettes.
3. Turn on the ADVANTAGE by pressing the ON/OFF switch located at the rear of the cabinet.
4. Press the RESET button at the rear of the cabinet. The screen displays the message "LOAD SYSTEM" with a cursor positioned below it. This step is not necessary if the ADVANTAGE was just turned on, as the ADVANTAGE automatically resets on power-up.
5. Press two keys in sequence: S<RETURN>.

This chapter provides programming information for the various sections of the ADVANTAGE, including the I/O devices. It also explains how to reconfigure the SIO and PIO boards to change their mode of operation.

3.1 MICROPROCESSOR CONTROL

The heart of the ADVANTAGE computer is the Z80 processor. Refer to the Appendix G for the programming details of this integrated circuit.

3.2 MEMORY CONTROL

3.2.1 Memory Mapping

The ADVANTAGE computer uses a memory mapping scheme to expand its memory addressing capabilities from 64K bytes to 256K bytes. This effectively expands the Memory Address bus from 16 bits to 18 bits.

The addressing scheme divides the 256K bytes into 16 pages of 16K bytes each (see Table 3-1). The three major areas of memory in the ADVANTAGE: the Main RAM, the Display RAM, and the Boot PROM, are permanently assigned to the addresses shown in the table.

Table 3-1

256K Address Space Allocation		
Page	18-Bit Address	Contents
0	00000 - 03FFF	16K bytes of Main RAM
1	04000 - 07FFF	16K bytes of Main RAM
2	08000 - 0BFFF	16K bytes of Main RAM
3	0C000 - 0FFFF	16K bytes of Main RAM
4	10000 - 13FFF	Not presently used
5	14000 - 17FFF	
6	18000 - 1BFFF	
7	1C000 - 1FFFF	
8	20000 - 23FFF	First 16K bytes of Display RAM
9	24000 - 27FFF	Last 4K bytes of Display RAM repeated four times
A	28000 - 2BFFF	Not used
B	2C000 - 2FFFF	Not used
C	30000 - 33FFF	2K-byte Boot PROM repeats to fill 64K bytes
D	34000 - 37FFF	
E	38000 - 3BFFF	
F	3C000 - 3FFFF	

Memory mapping is implemented by four Memory Mapping registers. Figure 3-1 shows how these registers work.

First, output instructions are used to load the register with the appropriate bits. Thereafter, each time the memory is accessed, the upper two bits of the program address automatically generate four bits of memory address by selecting one of the four Memory Mapping registers. The remaining 14 bits of the program address are passed through to the memory address without change.

With any one configuration of the Memory Mapping registers, the program has access to only four of the 16 possible pages. In order to change the four pages it wishes to access, the program must change one or more of the Mapping registers.

Memory Mapping Registers

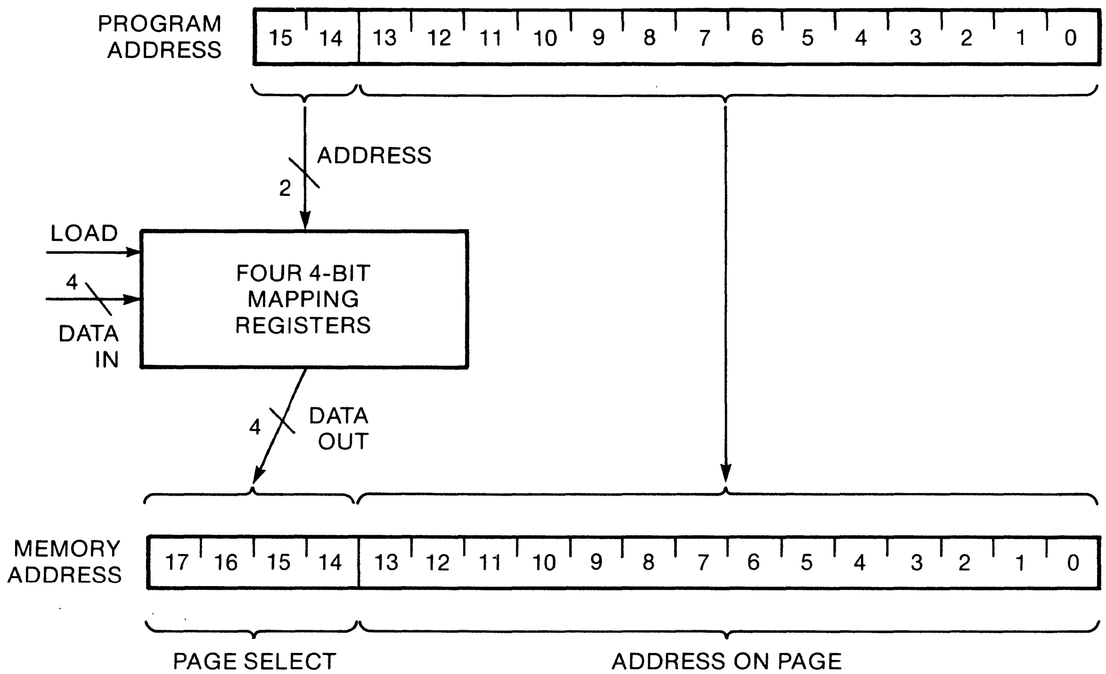


Figure 3-1

The Memory Mapping registers are initialized or changed by executing output instructions. The registers are write-only; their contents cannot be read by the program. Memory mapping I/O addresses are summarized in Table 3-2.

Table 3-2

Memory Mapping I/O Addresses

I/O Address (Hexadecimal)	Operation	Description
A0	OUTPUT only	Memory Map register 0
A1	OUTPUT only	Memory Map register 1
A2	OUTPUT only	Memory Map register 2
A3	OUTPUT only	Memory Map register 3

NOTES

- When these I/O addresses are decoded, bits 2 and 3 are ignored. This produces four addresses for each function that work equally well. For example, addresses A0, A4 and A8 all produce identical results.
- Attempting to read from any of the addresses listed in this table will read indeterminate data, and will load indeterminate data into the corresponding Memory Mapping register.

The bits from the output byte that are used to load any of the Memory Mapping registers are bits 7,2,1 and 0. The format of the output byte is shown in Table 3.3.

As an example of programming the mapping registers, the Display RAM may be mapped into pages 0 and 1 (program addresses 0000H through 4 FFFH by performing the following two steps:

1. Output 80H to I/O address A0H.
2. Output 81H to I/O address A1H.

Table 3-3

Memory Mapping Register Configurations	
Bits of Output Byte 76543210	Memory Reference
0xxxxNNN	Main RAM page NNN
1xxxx00N	Display RAM, N=0 is page 8 N=1 is page 9
1xxxx1xx	Boot Prom
NOTE: xx = ignored bits	

MEMORY MAPPING IN INTERRUPT MODE

When programming the ADVANTAGE computer in interrupt mode, take care to configure the memory mapping registers so that the automatic branch to the interrupt serviceroutine is directed to the correct page of memory. Exactly how this is done depends how the Z80 processor is programmed to respond to interrupts (see Appendix G). If the Z80 processor is programmed for a "Mode 2" response, the I/O ports in the ADVANTAGE respond with an "FF" regardless of which port generated the interrupt.

3.2.2 Memory Parity

The Main RAM has a parity bit associated with each memory location. The display and PROM memories do not have parity. The Main RAM parity bit is automatically written during a write operation and checked during a read operation. If an incorrect parity bit is encountered during a read operation the Parity Error flag is set. A parity error can occur because a memory location was read before any data was stored at that location.

The handling of parity errors can be controlled through the use of the status and control bytes shown in Table 3-5. The address of these bytes is given in Table 3-4.

Table 3-4

Memory Parity I/O Address

I/O Address (Hexadecimal)	Operation	Description
60	READ	Read Memory Parity Status byte
60	WRITE	Load Memory Parity Control byte

NOTE: When I/O address 60 is decoded, address bits 0,1,2 and 3 are ignored. This permits addresses 61 through 6F to work as well as 60.

Table 3-5

Memory Parity Status and Control Bytes

STATUS BYTE

7 6 5 4 3 2 1 0

Parity Error Flag . A zero indicates that a parity error was detected.

Indeterminate

CONTROL BYTE

7 6 5 4 3 2 1 0

Interrupt Enable. A one enables parity errors to generate either maskable or non-maskable interrupts (see Section 3.3) A zero disables interrupts due to parity errors.

Clear Parity Error. A one clears the Parity Error flag.

Ignored

3.3 INTERRUPTS

The Z80 processor has two interrupt inputs: a Maskable Interrupt (INT) and a Non-Maskable Interrupt (NMI). Refer to the data sheet in Appendix G for information about how these inputs affect the Z80 processor.

3.3.1 Maskable Interrupts

The sources of maskable interrupts are as follows:

1. The Keyboard. See Section 3.5.
2. The Video Controller. See Section 3.6.
3. I/O Boards. See Section 3.8.
4. Memory parity error. See Section 3.2.2.

A parity error in the Main RAM may cause a maskable interrupt or a non-maskable interrupt, depending upon jumper W4 on the Main PC Board. As shipped, the parity error is connected to the maskable interrupt. North Star software does not support its connection to the non-maskable interrupt.

3.3.2 Non-Maskable Interrupts

The sources of non-maskable interrupts are as follows:

1. Power Reset. This reset occurs whenever power is turned on, or whenever power is interrupted. The power reset also resets the Z80 processor.

2. Reset Pushbutton. This control is located on the rear panel of the ADVANTAGE.

3. Keyboard Reset. This reset is under program control (see Section 2.1.4).

4. Memory Parity Error. See the paragraph above describing jumper W4 on the Main PC Board.

3.4 SHARED I/O INTERFACE REGISTERS

The Z80 processor uses several status and control registers in order to communicate with other system components. Most of these registers are dedicated to a particular I/O device, but three of them, the I/O Control register, Status register 1 and Status register 2 are shared by more than one device. Figure 3-2 shows the relationship of these registers to the devices which they serve.

These three 'shared registers' are introduced and briefly described in this section. Their use on a particular device such as the keyboard or video monitor is covered in the section for that device.

Table 3-6

Shared Register Addresses

I/O Address (Hexadecimal)	Operation	Description
F0	WRITE only	Load I/O Control register
E0	READ only	Read Status register 1
D0	READ only	Read Status register 2

NOTES

- When these I/O addresses are decoded, address bits 0,1,2 and 3 are ignored. This produces 16 addresses for each function that work equally as well. For example, addresses F0 through FF all produce identical results.
- The I/O Control register is in an indeterminate state when power is turned on, and is not affected by any reset. Reading from this address at any time will cause indeterminate data to be read and to be loaded into the I/O Control register.
- Do not write to Status Register 1 or Status Register 2 as it causes bus conflicts.

The Three Shared I/O Interface Registers

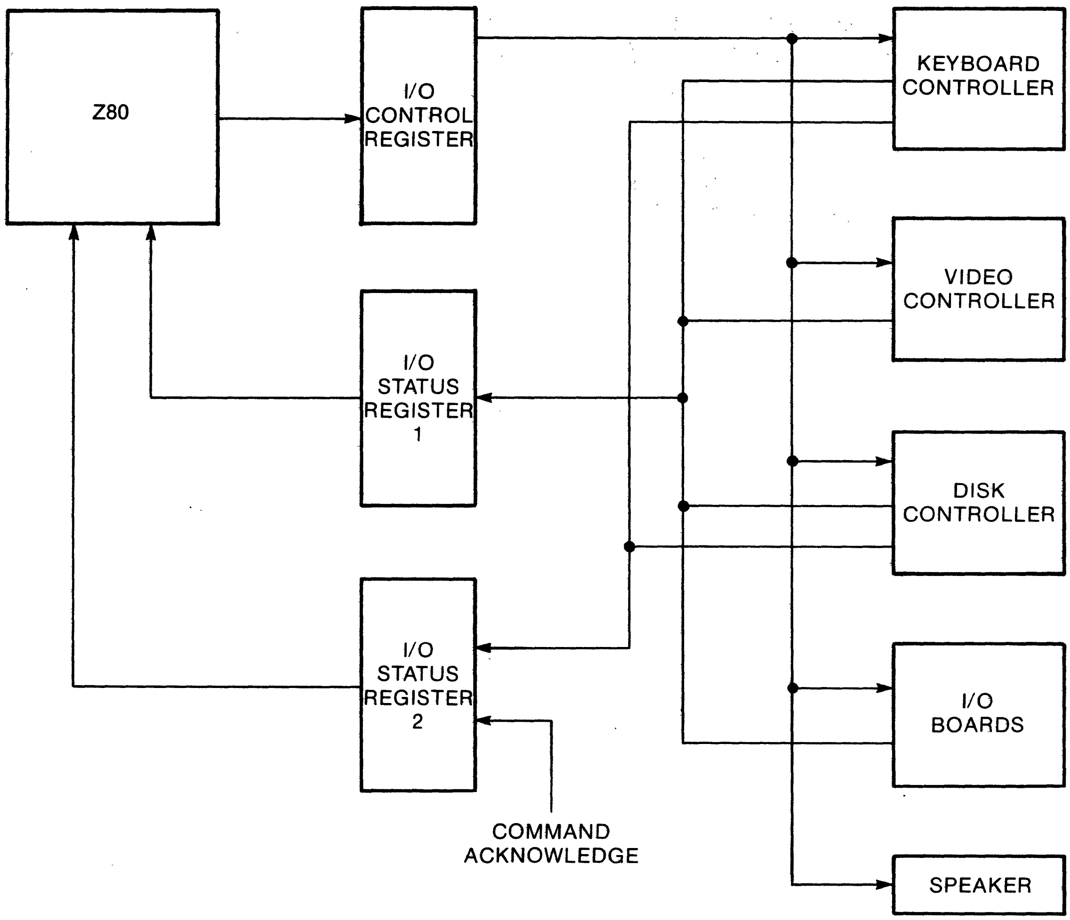


Figure 3-2

Table 3-7

I/O Control Register Format	
7 6 5 4 3 2 1 0	
2 1 0	<u>I/O Commands.</u> See Table 3-8.
3	<u>Acquire Mode.</u> This bit is set to zero at the beginning of each disk read operation to select Acquire mode (See Section 3.7)
4	<u>I/O Reset.</u> A zero causes a reset to be sent to the keyboard controller, the disk controller, and all the I/O boards. A one removes that reset.
5	<u>Blank Display.</u> A one causes the display to be blanked. A zero enables the data presently in the Display RAM to be displayed.
6	<u>Speaker Data.</u> This bit is fed directly into the speaker control circuitry. Complementing the bit at an audio frequency produces a corresponding tone in the speaker.
7	<u>Enable Display Interrupt.</u> A one enables bit 2 of the I/O Status Register 1 to generate a maskable interrupt (see Section 3.6).

The three shared registers are addressed as shown in Table 3-6. Their formats are given in Table 3-7, 3-9 and 3-10. Table 3-8 defines the I/O Commands, which are generated by the low-order three bits of the I/O Control register.

Table 3-8

I/O Commands

Command Number	Bits 0-2 of Control Register	Description
0	000	Show Sector. Place disk sector number into bits 0-3 of I/O Status register 2. The sector number has a range of 0-9, or one of two special codes: E = disk drive motors off, and F= index pulse detected. This function is also performed by command 5.
1	001	Show Char LSB's. Place low-order four bits of keyboard character into I/O Status register 2, bits 0-3.
2	010	Show Char MSB's. Place high-order four bits of keyboard character into I/O Status register 2, bits 0-3. Reset Keyboard flag, bit 6 of the same register.
3	011	Keyboard MI Flag. Complement the state of the Keyboard Maskable Interrupt flag. Following execution of the command 3, the state of this flag appears in bit 0 of I/O Status register 2. One=on, zero=off. The KB MI flag allows the Keyboard Data flag, bit 6 of I/O Status register 2, to generate a maskable interrupt.

Table 3-8 (continued)

Command Number	Bits 0-2 of Control Register	Description
4	100	Cursor Lock. Change the state of the Curson Lock flag, and place that flag into bit 0 of I/O Status register 2. One = on, zero = off.
5	101	Start Disk Drive Motors. Turn on both disk drive motors. Motors remain on for 3 seconds after the command is removed. Also perform "Show Sector" command (see above).
6	110	Used only as part of the command 6, command 7 sequence (see below).
6,7	110,111	Keyboard NMI Flag. This 2-command sequence complements the state of the Keyboard Non-maskable Interrupt flag. Following execution of this command sequence, the KB NMI flag appears in bit 0 of I/O Status register 2. One=on, zero=off. When this flag is on, the keyboard reset feature is enabled (see Section 2.1.4).
7	111	All Caps. When used alone, this command changes the state of the All Caps flag, and places that flag in bit 0 of I/O Status register 2. One = on, zero = off.

NOTE: In order for the I/O Commands to be effective, they must remain in the I/O Control register until the Command Acknowledge bit changes state. This bit is number 7 in I/O Status Register 2.

Table 3-9

I/O STATUS REGISTER 1 FORMAT

0	Keyboard Interrupt. A one indicates that a valid keyboard entry has caused a maskable interrupt.
1	I/O Interrupt. A zero indicates that one or more of the I/O Boards is generating a maskable interrupt. The bit is is set by clearing the interrupting condition.
2	Display Flag. This bit is set at the end of each vertical scan. It is cleared by accessing I/O address BOH. (see Section 3.6.1).
3	Non-Maskable Interrupt. A zero indicates that a non-maskable interrupt is present. See Section 3.3 for interrupting conditions.
4	Disk Write Protect. A one indicates that the selected disk drive is write protected. If no drives are selected or if the selected drive has no write protect option this bit is indeterminate.
5	Track 0. A one indicates that the selected disk drive is positioned at Track 0. If no drive is selected, this bit is indeterminate.
6	Sector Mark. This bit changes momentarily from a one to a zero at the end of each sector on the selected disk drive.
7	Disk Serial Data. This bit is connected directly to the serial data stream coming from the selected disk drive. It is used by the program to synchronize disk read operations (see Section 3.7).

Table 3-10

I/O STATUS REGISTER 2 FORMAT

7 6 5 4 3 2 1 0

3–0	These bits depend on the I/O command in the I/O Control register (see Table 3-8 above).
4	Auto-Repeat. A one indicates that a key has been held down for longer than 800 milliseconds. The bit goes to a zero when the key is released.
5	Character Overrun. Set to a one if an attempt is made to put more than seven characters into the keyboard buffer. (i.e., a character was received from the keyboard when the buffer was full). The bit is cleared whenever the upper nibble of a character is read (command 2).
6	Keyboard Data Flag. Set whenever one or more characters is available from the 7-characters keyboard buffer. Cleared when the upper nibble of the character is read (command 2).
7	Command Acknowledge. This bit is complemented each time an I/O Control Code is executed (for codes, see Table 3.8 above).

NOTES

- Bits 0-3 are only valid after bit 7 changes state to acknowledge that the command has been executed.
- When bits 0-3 contain the disk sector number, they have a range of 0-9 for the 10 sectors, or one of the following special codes:

  E = disk drive motors off
  F = index pulse detected

3.5 KEYBOARD CONTROL

This section contains the programming information for the ADVANTAGE keyboard. Refer to the diagrams and tables in section 3.4 for the following discussion.

3.5.1 RESET

When the I/O Reset bit (I/O address F0H, bit 4) is set on, then off, it has the following effect on the operation of the keyboard.

1. If there is an active maskable interrupt from the keyboard, it is reset.

2. The Keyboard Maskable Interrupt flag is reset. This disables maskable interrupts from the keyboard.

3. The Keyboard Data flag is reset. This flag is bit 6 of I/O Status register 2.

4. The Cursor Lock feature is reset (see Section 3.5.4).

5. The All Caps feature is reset (see Section 3.5.5).

6. The Auto-Repeat flag is reset. This flag is bit 4 of I/O Status register 2.

7. The Character Overrun flag is reset. This flag is bit 5 of I/O Status register 2.

3.5.2 Interrupt or Polled

The keyboard may be serviced in the interrupt mode, or it may be polled by the program.

If the interrupt mode is used, the program must set the Keyboard Maskable Interrupt (KB MI) flag. The following procedure may be used for this purpose.

1. Input and record the state of the Command Acknowledge bit (I/O address D0H, bit 7).

2. Issue command 3 to the I/O control register (I/O address F0H).

3. Wait for the Command Acknowledge bit to complement. This delay is in the range of 0.5 to 1.5 milliseconds.

4. Input from I/O Status register 2 and check bit 0. If this bit is on, the KB MI flag is already set.

5. If the KB MI flag is reset, repeat step 2 above.

When the keyboard causes an interrupt, the program can verify the source of the interrupt by imputting from I/O address E0H and checking bit 0. This bit is on if the keyboard is interrupting.

To clear the interrupt, the program must input keyboard characters (see Section 3.5.2) until the Keyboard Data flag is reset. This flag is bit 6 of I/O address D0H.

If the keyboard is to be polled rather than operated in interrupt mode, the KB MI flag must be reset. This flag is reset when the ADVANTAGE power is turned on, or when the ADVANTAGE Reset Button is pushed. The program may reset the KB MI flag by executing the following sequence:

1. Input and record the state of the Command acknowledge bit (I/O address D0H, bit 7).

2. Issue command 3 to the I/O Control register (I/O address F0H).

3. Wait for the Command Acknowledge bit to complement. This dellay is in the range of 0.5 to 1.5 milliseconds.

4. Input from I/O Status register 2 and check bit 0. If this bit is off, the KB MI flag is already reset.

5. If the KB MI flag is set, repeat step 2 above.

The program polls the keyboard by periodically imputting from Status register 2 (I/O address D0H) and checking bit 6. If the bit is on, the program reads the keyboard character(s) as described below.

3.5.3 Read Keyboard

Characters are read from the keyboard by performing the sequence given below. A sample subroutine for reading keyboard data without using interrupts is given in Table 3-11.

1. Input and record the state of the Command Acknowledge bit (I/O address D0H, bit 7).

2. Issue command 1 to the I/O Control register (I/O address F0H).

3. Wait for the Command Acknowledge bit to complement. This delay is in the range of 0.5 to 1.5 milliseconds.

4. Input the low-order nibble of the character from I/O address D0H.

5. Issue command 2 to I/O address F0H.

6. Wait for the Command Acknowledge bit to toggle.

7. Input the high-order nibble of the character from I/O address D0H.

Keyboard character ASCII codes are given in Appendix A. There are six keys that affect the values received from other keys: Left SHIFT, right SHIFT, CONTROL, COMMAND, ALL CAPS and CURSOR LOCK. Combinations of none, one, or two of these keys produce the five variations of keyboard codes: Unshifted, Shifted, CONTROL, CONTROL- Shifted, and CMND, as sown in the table "Keyboard ASCII Codes by Key" of Appendix A.

TABLE 3-11

Sample Routine for Reading Characters

```
KEYBOARD INPUT EXAMPLE

00D0    ==      SPRCS   ==      0D0H       ; STATUS REG 2 ADDR
0040    ==      CHRDY   ==      040H       ; KEYBOARD STATUS MASK
00F8    ==      CNTRG   ==      0F0H       ; CONTROL REGISTER ADDR
0038    ==      NORM    ==      038H       ; NORMAL CONTROL REG VALUE
0001    ==      CHDSL   ==      001H       ; COMMAND TO SHOW LOWER NIBBLE
0002    ==      CHDSU   ==      002H       ; COMMAND TO SHOW UPPER NIBBLE
                ;
0000'   DBD0    KEY:    IN      SPRCS      ; STATUS REG 2
0002'   E640            ANI     CHRDY      ; TEST FOR CHARACTER READY
0004'   28FA            JRZ     KEY        ; WAIT FOR KEYSTROKE
0006'   DBD0            IN      SPRCS      ; RESPONSE TO CURRENT COMMAND
0008'   6F              MOV     L,A        ; SAVE FOR COMMAND ACK TEST
0009'   3E19            MVI     A, NORM+CHDSL      ; LOWER NIBBLE COMMAND
000B'   D3F0            OUT     CNTRG      ; REQUEST LOWER NIBBLE FIRST
000D'   DBD0    KEY1:   IN      SPRCS
000F'   AD              XRA     L          ; TEST FOR COMMAND ACK
0010'   F2 000D'        JP      KEY1       ; WAIT FOR COMMAND ACK
0013'   DBD0            IN      SPRCS      ; GET LOWER NIBBLE
0015'   E60F            ANI     15         ; MASK TO NIBBLE ONLY
0017'   67              MOV     H,A
0018'   3E1A            MVI     A, NORM+CHDSU      ; UPPER NIBBLE COMMAND
001A'   D3F0            OUT     CNTRG      ; ALSO ADJUSTS FIFO AND STATUS
001C'   DBD0    KEY2:   IN      SPRCS
001E'   AD              XRA     L
001F'   FA 001C'        JM      KEY2       ; WAIT FOR ANOTHER ACK
0022'   DBD0            IN      SPRCS      ; GET UPPER NIBBLE
0024'   87              ADD     A          ; X2
0025'   87              ADD     A          ; X4
0026'   87              ADD     A          ; X8
0027'   87              ADD     A          ; X16
0028'   B4              ORA     H          ; COMBINE THE TWO NIBBLES
                ;
                        .END
```

3.5.4 Cursor Lock

The CURSOR LOCK key alters the codes that are produced by some of the keys on the numeric keypad as defined in Appendix A.

The CURSOR LOCK key has a built-in light that indicates whether the feature is on or off. This feature can be set or reset by pressing the key, or by issuing a command from the program.

To change the state of the CURSOR LOCK feature, perform the following sequence:

1. Input and save the state of the Command Acknowledge bit (I/O address D0H, bit 7).

2. Issue command 4 to I/O address F0H.

3. Wait for the Command Acknowledge bit to complement. This delay is in the range of 0.5 to 1.5 milliseconds.

4. If desired, confirm the new state of CURSOR LOCK by inputting I/O address D0H and checking bit 4. One = on, zero = off.

3.5.5 All Caps

The ALL CAPS key alters the codes that are produced by the alphabetic keys as defined in Appendix A.

The ALL CAPS key has a built-in light that indicates whether the feature is on or off. This feature can be set or reset by pressing the key, or by issuing a command from the program.

To change the state of the ALL CAPS feature, perform the following sequence:

1. Input and save the state of the Command Acknowledge bit (I/O address D0H, bit 7).

2. Issue command 7 to F0H.

3. Wait for the Command Acknowledge bit to complement. This delay is in the range of 0.5 to 1.5 milliseconds.

4. If desired, confirm the new state of ALL CAPS by inputting I/O address D0H and checking bit 7. One = on, zero = off.

3.5.6 Auto-Repeat

If any key or legal combination of keys is held down for more than 800 milliseconds, the Auto-Repeat bit in Status register 2 is set. It will remain set until the key(s) is released. In addition, a special character (FFH) is inserted by the keyboard following the one that is to be repeated. The keyboard sends the character to be repeated only once.

If the program is to implement the Auto-Repeat feature, it should perform the following procedure:

1. Input I/O address D0H and check bit 4. A "one" indicates repeat.

2. If this bit is set, start inputting keyboard characters until the FFH character is encountered.

3. When FFH is found, the preceeding character will be the one that should be repeated.

4. Discard the FFH character.

5. Continue to repeat the character until the Auto-Repeat bit is reset.

If the program is not to implement the Auto-Repeat feature, it should simply discard the FFH character.

3.5.7 Character Overrun

I/O address D0H should be input and bit 5 checked each time a character is input from the keyboard. If the bit is a one, it indicates that the seven-character keyboard buffer was overfilled, resulting in the loss of one or more characters.

3.6 VIDEO DISPLAY CONTROL

3.6.1 Screen Mapping

The video display consists of a matrix of contiguous dot positions that is 640 dots wide and 240 dots high. There is a one-to-one correspondence between each dot position and a bit in memory.

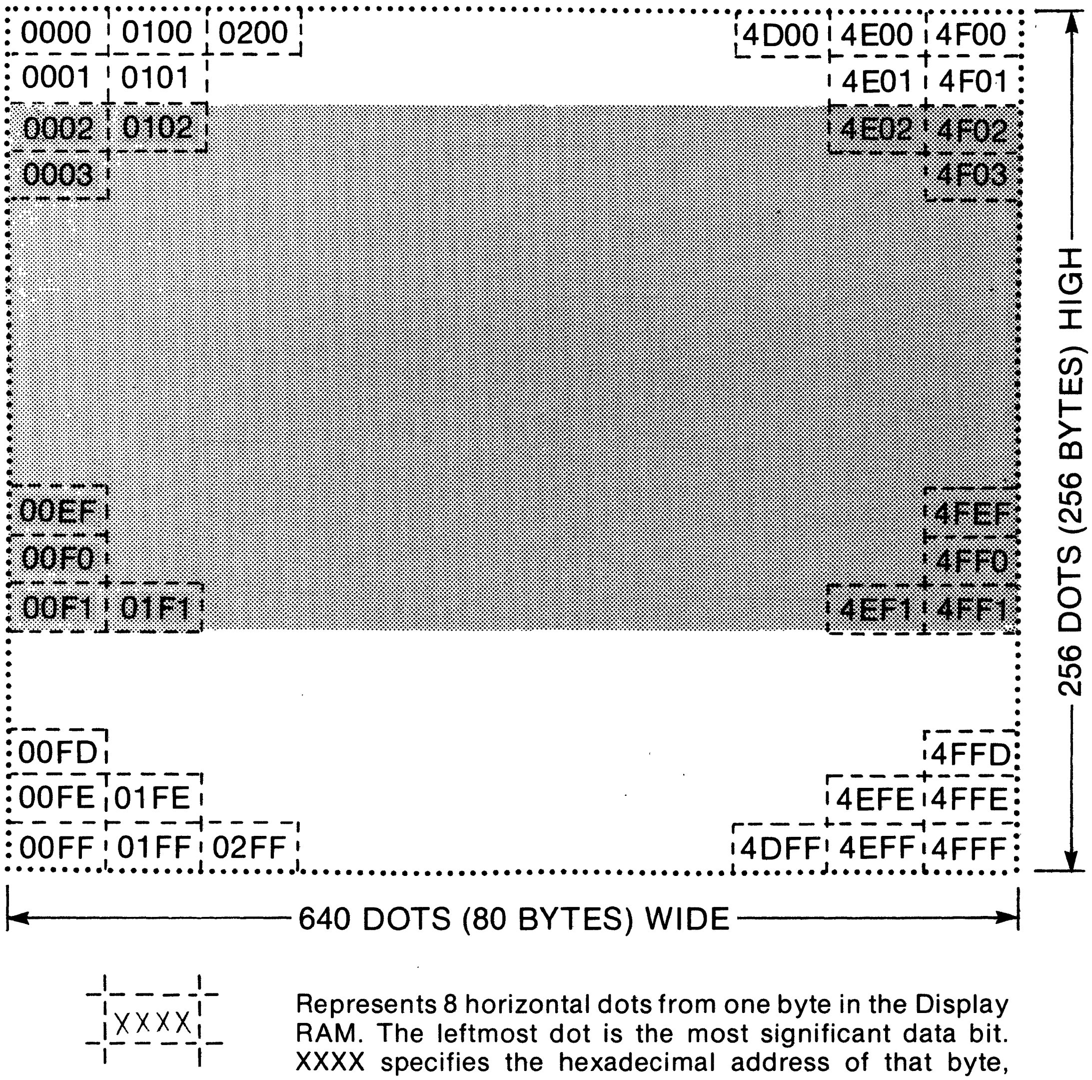
Data to be displayed on the screen is stored in the Display RAM. This RAM contains enough data to produce a display that is the same width as the screen format (640 dots) but is 256 dots high (see Figure 3-3).

The screen can be made to scroll vertically through the Display RAM in a wrap-around fashion. For example, if the screen is scrolled down so that the 50th horizontal row of dots in the RAM format is displayed at the top of the screen, then row 51 will be next, then 52, etc., until row 256 is encountered somewhere near the bottom of the screen. At that point the display continues with row 1 of dots in the RAM format, then row 2, row 3, etc., until the bottom of the screen is encountered.

The Display RAM is physically located between memory addresses 20000H and 24FFFH. The actual program addresses used to access this RAM depend on the state of the Memory Mapping registers (see Section 3.2.1). For the purpose of this discussion, assume that the Display RAM has been mapped into pages 0 and 1, i.e., 80H has been output to I/O address A0H, and 81H has been output to I/O address A1H.

The data in the Display RAM is organized as shown in Figure 3-3. To write into any dot or group of dots on the screen load the appropriate bit pattern into the correct locations of Display RAM, and insure that the screen is scrolled into position so that the bits are displayed.

Data Format in Display RAM



Represents 8 horizontal dots from one byte in the Display RAM. The leftmost dot is the most significant data bit. XXXX specifies the hexadecimal address of that byte, provided that the Display RAM is mapped into page 0.

NOTE: The shaded area indicates the data that would be displayed if the Start Scan register contained 02H.

Figure 3-3

To scroll the screen, change the number in the Start Scan register. Table 3-12 gives the I/O addresses of the register. The binary number in this register indicates how far down the screen image will be positioned relative to the top of the Display RAM format (see Figure 3-3). For example, if 02H is output to this register, the data for the top row of dots on the screen will come from RAM locations 0002H, 0102H, 0202H, etc.

3.6.2 Forming Letters and Symbols

The flexibility of the display screen format allows the user to form characters of virtually any style or size. For convenience, a set of standard character shapes is stored in the Boot PROM. When these characters are used, the display may contain 24 horizontal rows of characters with 80 characters per row. Instructions for accessing these characters are given in Section 3.6.5.

Table 3-12

Video I/O Addresses		
I/O Address (Hexadecimal)	**Operation**	**Description**
90	OUTPUT	<u>Load Start Scan Register.</u> This 8-bit register specifies which display line is to be on top of the screen.
B0	INPUT or OUTPUT	<u>Clear Display Flag.</u> This flag marks the period between automatic scans of the display screen (see Section 3.6.3 below).

NOTES

- When these I/O addresses are decoded, address bits 0,1,2 and 3 are ignored. This produces 16 addresses for each function that work equally well. For example, addresses 90 through 9F all produce identical results.
- When inputting from address B0, the input data is indeterminate.
- When outputting to address B0, the output data is ignored.

3.6.3 Display Flag

The Display flag is bit 2 in I/O Status Register 1 (I/O address EOH). This flag allows the program to synchronize data transfers to the Display RAM. This prevents the momentary flicker which occurs when RAM data is changed while it is being refreshed on the screen.

The flag is set each time the automatic refresh circuitry completes a scan of the display screen, or approximately every 17 milliseconds. The flag is reset by the program (see Table 3-12). When the Display flag is set, it marks the beginning of a 0.50 millisecond period, during which time the screen is not being scanned. After this period, scanning resumes at the top of the screen and moves toward the bottom.

The Display flag causes a maskable interrupt each time it sets, if bit 7 is set in the I/O Control register (I/O address FOH).

3.6.4 Screen Blanking

The screen may be blanked by setting bit 5 of the I/O Control register (I/O address FOH). Resetting the bit allows the screen to display again the contents of Display RAM.

3.6.5 The Video Driver

The Video Driver is a Z80 processor subroutine within the Boot PROM. It is used to generate character templates for the video display and for controlling the cursor. The generated templates are 8 dots wide and 10 dots high, including the intercharacter and interline spaces.

The user supplies a list of parameters to the Video Driver that includes the current position of the cursor. The user then passes a single character to the Video Driver. If the character corresponds to one of the 96 displayable ASCII characters listed in Appendix A, it is displayed on the screen at the current cursor position. If the character corresponds to one of the control codes listed in Table 3-13, the Video Driver executes the appropriate command.

Table 3-13

Video Driver Control Codes		
Control Code	Hexadecimal Value	Description
CTRL-H	08	Backspace (cursor left)
CTRL-J	0A	Line Feed (cursor down)
CTRL-K	0B	Reverse Line Feed (cursor up)
CTRL-L	0C	Forespace (cursor right)
CTRL-M	0D	Carriage Return
CTRL-N	0E	Clear to End of Line
CTRL-O	0F	Clear to End of Screen
CTRL-X	18	Cursor On
CTRL-Y	19	Cursor Off
CTRL-_	1F	New Line
CTRL-^	1E	Home Cursor (to upper left corner of screen)

Before using the Video Driver, map the Boot PROM into 8000H and map the Display RAM into 0000H and 4000H (see Section 3.2.1). The Video Driver does not use the Z80 processorstack pointer. A block of eleven bytes of data in main RAM must be set up before calling the Video Driver. The calling sequence is shown below and the data block format is shown in Table 3-9.

To invoke the Video Driver:

1. Set up the 11-byte RAM block as described in Table 3-9.
2. Set Z80 processor IX Register to the start address of the RAM block.
3. Place the desired byte in the Z80 processor acumulator.
4. Jump to the Video Driver entry point (JMP 87FDH).

Table 3-14

Video Driver Data Block Format		
Byte	**Name**	**Description**
1	CURSX	<u>Cursor Column Number.</u> There are 80 columns on the screen numbered 00H through 4FH. Each column is one byte wide.
2	CURSY	<u>Cursor Line Number.</u> There are 256 lines numbered 00H through FFH. This number refers to the top line of the cursor template.
3-4	PIXEL	<u>PIXEL Data Table Address.</u> The standard Pixel Data Table is in the PROM at address 8561H.
5	SCRCT	<u>Line Number.</u> The line number which is currently at the top of the screen. This number is incremented or decremented by 10 (decimal) whenever a character causing a scroll is executed.
6	STATS	<u>Status Byte:</u> Bit 0 - Set by Driver if cursor is disabled. Bit 1 - Set by user to disable auto wrap-around of display format. Bit 2 - Set by the user to disable scrolling. Also inhibits automatic carriage return of cursor. Bit 6 - Set by Driver if cursor reaches top of screen and scrolling is inhibited.

Table 3-14 (continued)

Bytes	Name	Description
		Bit 7 - Set by Driver if cursor reaches bottom of screen and scrolling is inhibited. Bits 3,4,5, Not used.
7-8	RETFP	Return Address. The Video Driver does not use the Z80 stack. It returns to the calling program by jumping to the address stored in these two bytes.
9-10	CTEMP	Cursor Template Address. This address must be set up to the start of a 10-byte block containing the cursor template (normally all FHH's).
11	VIDEO	Normal/Reverse. Set this byte to 00 for normal video, FFH for reverse video.

Typical Default Values for RAM Block:

```
CURSX: DB 00        ; Cursor at upper left corner
CURSY: DB 00
PIXEL: DW 8561H     ; Standard character set
SCRCT: DB 00        ; Scan line 0 at top of screen
STATS: DB 00        ; Cursor on, auto wrap-around on, scrolling
                      enabled
RETFP: DW XXXX      ; XXXX is return address from PROM
CTEMP: DW 0FFFFH,0FFFFH,0FFFFH,0FFFFH,0FFFFH  ; Cursor template
```

Note: CURSX, CURSY, SCRCT are automatically updated by the Video Driver.

3.7 FLOPPY DISK DRIVE CONTROL

The Floppy Disk Drive Controller uses a minimum of hardware and requires a sophisticated program to read from and write to the disk drives. Some of the timing and motor control is determined by the program.

The program communicates with the Floppy Disk Controller in the following ways:

1. Through the Shared I/O Interface registers described in Section 3.4.
2. By outputting control bytes to the Drive Control register. The format for the register is shown in Table 3-16, and its I/O address is listed in Table 3-15.
3. By accessing the other I/O addresses given in Table 3-15.

Table 3-15

Floppy Disk I/O Addresses

I/O Address (Hexadecimal)	Operation	Description
80	INPUT	Input Disk Data. Sets the processor into the wait state until the disk data is available, then reads the data. Inputting from this address when data is unavailable puts the processor into a continuous wait state.
80	OUTPUT	Output Disk Data. Sets the processor into the wait state until the Disk Controller writes the data to the diskette. Outputting to this address before setting the Disk Write flag puts the processor into a continuous wait state.

Table 3-15 (continued)

81	INPUT	Input Sync Byte. Sets the processor into the wait state until the sync byte is available, then reads the data. If the disk format is correct, the character read is a BFH. Inputting from this address when a sync byte is not available puts the processor into a continuous wait state.
81	OUTPUT	Load Drive Control Register. See Table 3-16 for the register format.
82	INPUT	Clear Disk Read Flag. Terminates the disk read operation. The data input by this address is indeterminate.
82	OUTPUT	Set Disk Read Flag. This flag is set as one of the steps in initiating a disk read operation. The output data is ignored.
83	OUTPUT	Set Data Write Flag. This flag is set to initiate a disk write operation. The output data is ignored. The Disk Write flag is cleared on the leading edge of the next sector mark.

NOTES

- When these I/O addresses are decoded, bits 2 and 3 are ignored. This produces four address for each function that work equally well. For example, addresses 80, 84, 88 and 8C all produce identical results.
- If a disk operation causes the processor to go into a continuous wait state, the Main RAM refresh cycles are interrupted and data in Main RAM is lost.

Table 3-16

Drive Control Register Format

7 6 5 4 3 2 1 0

<u>Disk Drive 1.</u> A one selects disk drive 1. Bit 1 must be off.

<u>Disk Drive 2.</u> A one selects disk drive 2. Bit 0 must be off.

Not used.

<u>Step Pulse.</u> Setting this bit on, then resetting it, causes the head to step in the selected disk drive. This bit must remain off for at least 5 milliseconds between step pulses.

<u>Step Direction/Precompensation.</u>

A. During stepping of the disk from track to track, this bit determines the step direction.
   - 0 = Step toward outer track (higher track numbers).
   - 1 = Step toward inner track (lower track numbers)

B. During writing, this bit controls write precompensation.
   - 1 = Use precompensation
   - 0 = Use no precompensation.

Precompensation is required on the inside 20 tracks - track 15 through 34 on side 0 and tracks 35 through 49 on side 1.

<u>Diskette Side Select.</u>
   - 0 = Side 0
   - 1 = Side 1

Not used

A disk operation involves selecting the drive, enabling the motor, performing a head seek, selecting a sector, and then performing the read or write operation. These operations are described separately in the following subsections.

3.7.1 Power-On Initialization

The data separation circuitry must be initialized after power is applied to the disk controller but before a read or write operation. This is done by alternately setting and clearing the Disk Read flag (I/O address 82H)) at approximately 100-millisecond intervals for five cycles.

3.7.2 Motor Enable

Both disk drive motors are turned on whenever a command 5 is received (Start Disk Drive Motors, see Table 3-8). If the command 5 is removed for three seconds, the value 0EH is displayed as the sector number. After 100 microseconds both disk drive motors are turned off and the Drive Control register is reset to zeroes. The 100-microsecond delay prevents the motors from being turned off in the middle of a read or write operation.

3.7.3 Drive Selection

After the drive motors are turned on, the program loads the Drive Control register (see Table 3-16) to select one of the two drives. At the same time the other bits of the register may be loaded in preparation for a head seek, read, or write.

3.7.4 Seek

The positioning of the disk drive read/write head is entirely under program control. The program must keep track of the position of the head and generate the timing pulses required to move the head from track to track.

The head is initialized (set on Track 0) by stepping it one track at a time toward the outside of the diskette, and after each step, inputting I/O Status register 1 (I/O address E0H). Bit 5 of the register is on when the selected drive has its head positioned on track 0. There are 35 tracks per side.

The head is stepped by setting and then resetting bit 4 of the Drive Control register (I/O address 81H). When the head is moved by more than one track in either direction, this bit must remain off for at least 5 milliseconds between step pulses. When the head reaches its destination, the program must delay at least 20 milliseconds to allow time for the head to settle.

3.7.5 Sector Selection

The sector number is read by performing the following sequence:

1. Input and record the state of the Command Acknowledge bit (I/O address D0H, bit 7).

2. Issue command 5 to the I/O Control register (I/O address F0H, refer to section 3.4).

3. Wait for the command acknowledge bit to complement. This delay is in the range of 0.5 to 1.5 milliseconds.

4. Input the Sector Mark bit (I/O address E0H, bit 6) until it is found to be zero.

5. Input the sector number (I/O address D0H, bits 0 through 3). This number is valid while the Sector bit is zero, and for 50 microseconds thereafter.

The number obtained by following the above procedure is actually the number of the previous sector. For example, if sector 6 is to be accessed, the program must search for sector 5. If the desired sector is not found on the first attempt, repeat steps 4 and 5 above until it is found.

When the correct sector has been located, the program goes into a loop, waiting for the sector mark to go from a zero to a one. The read or write operation sequence must be initiated on this transition.

3.7.6 Read Data

After the proper sector number is found, the read sequence is as follows:

1. Wait 500 microseconds after the zero-to-one transition of the Sector Mark bit.

2. Set the Disk Read flag by outputting to I/O address 82H.

3. Change the Acquire Mode flag to zero (bit 3 of I/O address F0H).

4. Wait 150 microseconds, then change the Acquire Mode flag to a one.

5. Wait until the Disk Serial Data bit (I/O address E0H, bit 7) changes to a one.

6. Input the sync byte (I/O address 81H). This byte should be FBH.

7. Input from I/O address 80H for the remainder of the data. The next byte read is the second sync byte, which is the sector number plus 16 times the track number, truncated to eight bits. Following this are the 512 data bytes and the CRC byte. The CRC byte is not checked by hardware; a software routine is needed if checking is desired.

8. The program's task is complete at this point. The hardware will reset the Disk Write flag at the zero-to-one edge of the next sector mark. During the sector mark a new write sequence can be started

Read timing is illustrated in Figure 3-4A. Note that the timing of the Sector Mark bit is such that consecutive sectors may be read.

3.7.7 Write Data

After the proper sector number is found, the write sequence is as follows:

1. Input the Write Protect bit (I/O address E0H, bit 4). The bit must be a zero to write on the diskette.
2. If writing to one of the inner tracks, set the Precompensation bit (I/O address 81H, bit 5). Precompensation is required on tracks 15 through 34 on side 0, and tracks 35 through 49 on side 1.
3. Set the Disk Write flag by outputting to I/O address 83H. This must be done within 150 microseconds after the zero-to-one transition of the Sector Mark bit (I/O address E0H, bit 6).
4. Output 33 consecutive bytes of zeros to I/O address 80H. This forms the preamble of the sector.
5. Output two sync bytes to I/O address 80H. The first contains the synchronization byte (0FBH), and the second contains the sector address (see READ DATA).
6. Output 512 data bytes to I/O address 80H.
7. Output the CRC byte to I/O address 80H. Note that the program must calculate the CRC byte.
8. The program's task is complete at this point. The hardware will reset the Disk Write flag at the zero-to-one edge of the next sector mark. During the sector mark a new write sequence can be started.

Note that it is possible to write contiguous sectors by waiting for the Sector Mark bit to return to zero, and starting again with step 3 above.

Write timing is illustrated in Figure 3-4B.

Disk Read/Write Timing

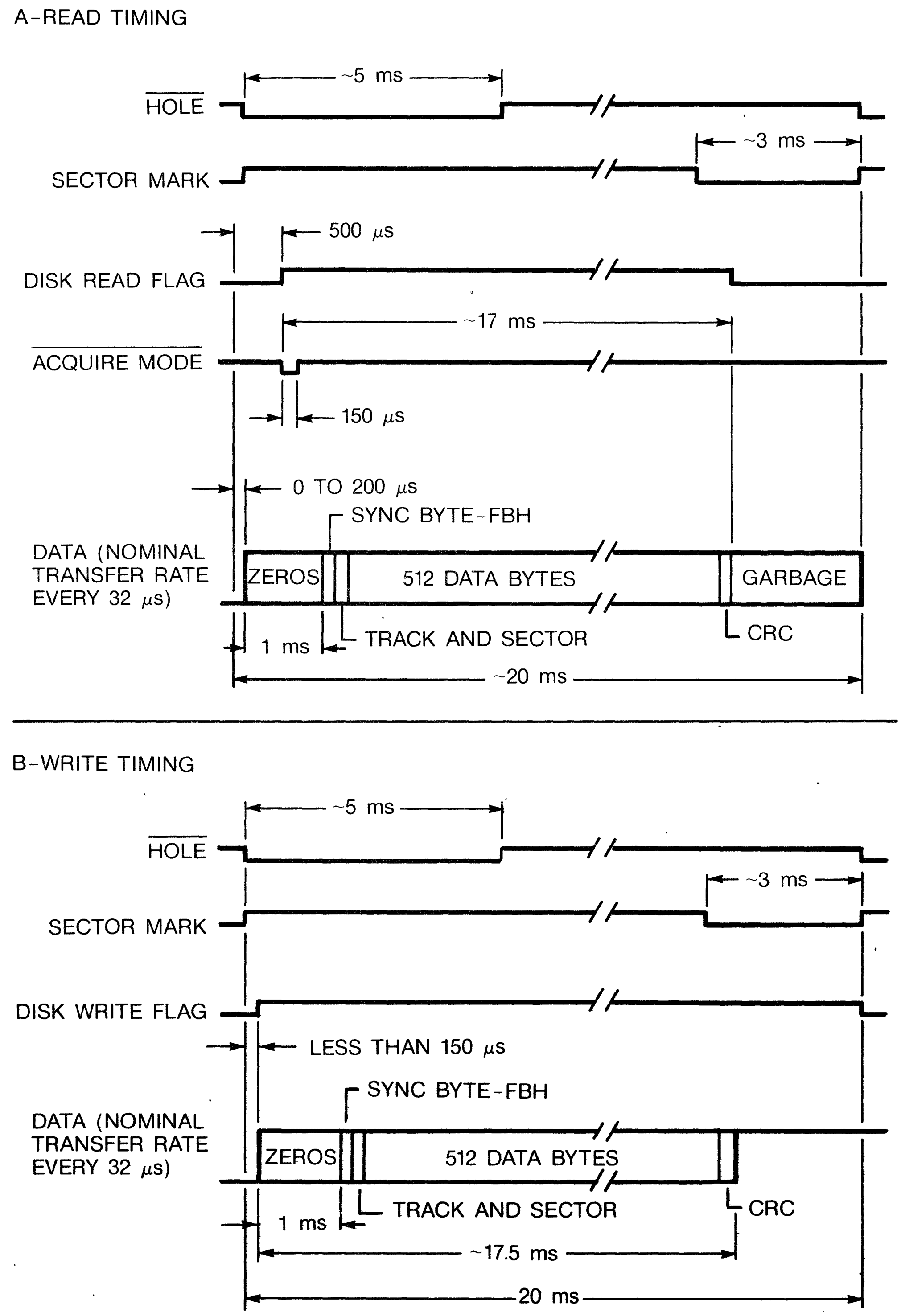


Figure 3-4

3.8 ACCESSING THE I/O BOARDS

The ADVANTAGE computer interfaces with external I/O devices such as printers and communication links by means of printed circuit (PC) boards. These boards plug into the connectors at the rear of the Main PC Board. The connectors all share a common set of signals and a common set of commands which can be sent by the program.

There are two I/O boards which can be used in the I/O board slots: the PIO (Parallel I/O) and the SIO (Serial I/O). This section introduces the I/O commands which can be sent to these boards. The programming information for a particular board will be found in the section pertaining to that board.

3.8.1 Reset

The I/O boards are reset by changing bit number 4 of the I/O Control register first to a zero, then to a one. The I/O address of this register is F0H.

3.8.2 Board ID

A command may be sent to each of the I/O board slots requesting that the board inserted into that slot identify its board type. These commands take the form of I/O instructions. The I/O addresses corresponding to the board slots are given in Table 3-17. The I/O identification codes are given in Table 3-18.

There are six I/O board slots, numbered 1 through 6. Slot 1 is the left-hand board as seen from the rear of the unit. They are numbered in sequence from left to right.

Table 3-17

I/O Board Addresses		
I/O Address (Hexadecimal)	Operation	Description
00 - 0F	INPUT/OUTPUT	Access I/O board in slot 6
10 - 1F	INPUT/OUTPUT	Access I/O board in slot 5
20 - 2F	INPUT/OUTPUT	Access I/O board in slot 4
30 - 3F	INPUT/OUTPUT	Access I/O board in slot 3
40 - 4F	INPUT/OUTPUT	Access I/O board in slot 2
50 - 5F	INPUT/OUTPUT	Access I/O board in slot 1
70 or 78	INPUT	INPUT the ID from slot 6
71 or 79	INPUT	INPUT the ID from slot 5
72 or 7A	INPUT	INPUT the ID from slot 4
73 or 7B	INPUT	INPUT the ID from slot 3
74 or 7C	INPUT	INPUT the ID from slot 2
75 or 7D	INPUT	INPUT the ID from slot 1
76 or 7E	INPUT	Currently unused. Returns all ones.
77 or 7F	INPUT	Currently unused. Returns all ones.

Table 3-18

I/O Board Identification Codes	
Identification Code (Hexadecimal)	I/O Board
7F	FPB - Floating Point Board
F7	SIO - Serial Input/Output Board
BE	HDC - Hard Disk Controller Board
DB	PIO - Parallel Input/Output Board
FF	No board or board with no ID.

3.8.3 Byte Transfers

I/O instructions are used to transfer 8-bit bytes between the program and any one of the I/O boards. These bytes may be data bytes, control bytes or status bytes, depending upon the I/O address that is used and the particular I/O board that decodes the address.

Table 3-17 lists the I/O addresses (00 through 5F) that are used to access a board for a single byte transfer. Each board slot is assigned to a group of 16 I/O addresses. The most significant digit of the address determines which board slot is accessed, and the least significant digit has a meaning determined by the particular board in that slot. The directin of the data transfer depends upon whether the program executes an input or an output instruction.

3.8.4 Interrupt

A maskable interrupt may be generated from any of the I/O board slots. The program may detect this condition by inputting from I/O address E0H and checking bit 1. The bit will be a zero if any of the I/O boards are interrupting. The boards must be polled individually to determine which board caused the interrupt.

3.9 SIO BOARD

The Serial Input/Output (SIO) Board provides a general facility for communicating with serial I/O devices. Synchronous and asynchronous operation are described in separate subsections. This section begins by describing those features of the board that are common to both synchronous and asynchronous operation.

3.9.1 Reset

When the I/O Reset bit (I/O address F0H, bit 4) is set on, then off, it has the following effect on the SIO Board:

1. The Interrupt Mask is cleared to zeros, preventing any interrupts from the board.

2. The Baud Rate register is cleared to zeros. Normally the register would now have to be reloaded to select the desired baud rate. See the appropriate section below.

3. The USART is reset, in preparation for reprogramming.

Note that the I/O Reset bit resets all I/O Boards simultaneously.

3.9.2 Board ID

The 8-bit identification code for the SIO Board is F7H. The I/O address used to input this code is determined by the board slot occupied by the SIO (see Table 3-17).

3.9.3 Data Transfers

The I/O address used to transfer a data byte to or from the SIO Board is X0H, where X is determined by the board slot occupied by the SIO (see Table 3-19). The standard location for the SIO Board is slot 1.

Table 3-19

First Digit of I/O Address	
Board Slot	First Digit of I/O Address
6	0
5	1
4	2
3	3
2	4
1	5

3.9.4 Control

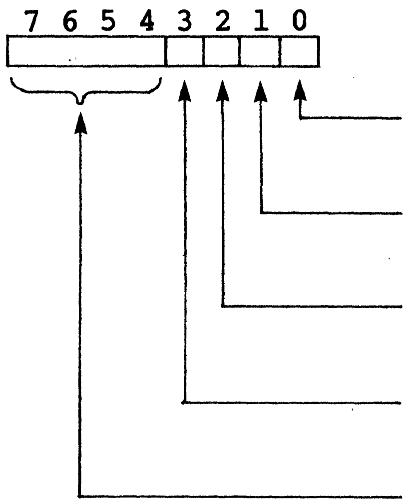
The operation of the SIO Board is controlled by specifying the Interrupt Mask and the baud rate, and by programming the 8251 USART IC (integrated circuit).

The format of the Interrupt Mask is shown in Table 3-20. A one in any of the bit positions 0 through 3 allows the SIO Board to generate a maskable interrupt if the stated condition occurs. The program defines this mask by outputting the appropriate bit pattern to I/O address XAH, where X is determined by the board slot occupied by the SIO Board (see Table 3-19).

The baud rate is specified by loading the Baud Rate register as described in the appropriate section: 3.9.7 for asynchronous mode, and 3.9.8 for synchronous mode.

Table 3-20

SIO Interrupt Mask Format



Tx Empty. Transmitter has finished sending characters.

Tx Ready. Buffer is ready to receive the next character.

Rx Ready. Receiver has a character ready.

Sync Detect. Synchronization has been achieved (synchronous mode only).

Unused.

NOTE: A one in any bit position 0 through 3 allows the stated condition to generate a maskable interrupt.

Programming the 8251 USART is done by resetting the SIO Board (see Section 3.9.1), then outputting a series of control bytes to the SIO. These bytes are output to I/O address X1H, where X depends upon the board slot occupied by the SIO Board. The control bytes necessary to configure the SIO for a particular mode of operation such as synchronous/asynchronous, number of bits per character, etc., are defined in the specification sheets for this IC, which can be found in Appendix H.

3.9.5 Status

A status byte may be read from the SIO Board by inputting I/O address X1H, where X depends upon the board slot occupied by the SIO Board (see Table 3-19). The composition of this status byte is given in the specification sheets for the 8251 USART, which can be found in Appendix H.

Table 3-21

Serial I/O Addresses

I/O Address (Hexadecimal)	Operation	Description
X0	INPUT/OUTPUT	USART data
X1	INPUT/OUTPUT	USART Status/Command
X8	OUTPUT	Baud Rate Register
XA	OUTPUT	Interrupt Mask

NOTES

- The first digit of these I/O addresses is determined by the board slot occupied by the SIO board (see Table 3-19).
- The Baud Rate register may also be accessed by using I/O address X9.
- The Interrupt Mask may also be accessed by using I/O address XB.
- Inputting from I/O addresses X8, X9, XA or XB causes indeterminate data to be loaded.

3.9.6 Interrupt or Polled

The SIO Board may be serviced in the interrupt mode or it may be polled by the program.

If the interrupt mode is used, one or more bits of the Interrupt Mask must be set to allow the USART to generate interrupts. The Interrupt Mask is discussed in Section 3.9.4.

When the SIO Board causes an interrupt, the program must determine the source of the interrupt. It does this by inputting from I/O address E0H and checking bit 1. The bit is a zero if any of the I/O boards including the SIO are interrupting. The program then inputs the status of all I/O boards to determine which board(s) is interrupting.

The program decides whether the SIO Board has interrupted by comparing the status bits to the bits in the Interrupt Mask. The program can respond by inputting or outputting a data byte, as appropriate, or by simply masking the interrupting condition.

If the SIO Board is to be polled, the Interrupt Mask must be loaded with zeros. The program polls the SIO by perodically reading the status byte from the 8251 USART (see Section 3.9.5) and taking appropriate action.

3.9.7 SIO in Asynchronous Mode

A. Asynchronous Modem Configuration

To establish a communication link between two electronic devices, one device must simulate a modem while the other simulates a terminal. If the ADVANTAGE is to communicate with a serial terminal such as an external CRT, a teletype, or a serial printer, the SIO must be configured to simulate a modem. Similarly, if the ADVANTAGE is to communicate with a modem, the SIO must simulate a terminal.

As shipped, the SIO is configured as a modem; it is ready for immediate connection to an asynchronous RS-232 terminal or a North Star-supplied printer. Connection to most asynchronous terminals and printers requires no configuration changes.

If the SIO has ever been reconfigured as a terminal, it can be restored to its original configuration as follows:

1. Remove the Clock Header in board locatin 1A, if one is present.
2. Remove the Configuration Header, board location 3A, and replace it with a 16-pin header wired as shown in Figure 3-5.

Asynchronous Modem Configuration Header

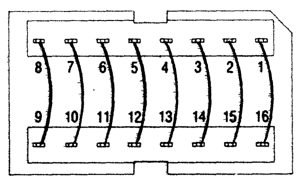


Figure 3-5

B. Asynchronous Terminal Configuration

If the ADVANTAGE is to communicate with a modem (or with another computer simulating a modem) the interfacing SIO port must be configured to simulate a terminal.

To configure the SIO as a terminal, proceed as follows:

1. Remove the Clock Header in board location 1A, if one is present.

2. Remove the Configuration Header from board location 3A and replace it with a 16-pin header wired as shown in Figure 3-6.

Asynchronous Terminal Configuration Header

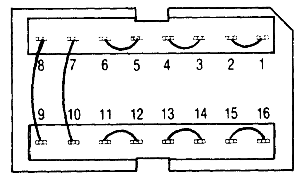


Figure 3-6

C. Current Loop Operation

Whereas most computers, terminals, and printers use RS-232 signal levels, some terminals, such as teletypes, use 20 mA current loop signals.

A teletype is a passive device; it does not supply current, but relies on current supplied by the SIO. The SIO is not equipped to accommodate active current loop devices such as computers that produce current loop signals.

As shipped, each SIO board is configured to use RS-232 signals.

To configure an SIO for current loop operation, perform the following procedure:

1. Remove the Configuration Header, board location 3A, and replace it with a 16-pin header wired as shown in Figure 3-7.

Current Loop Configuration Header

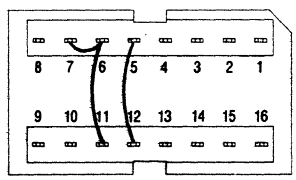


Figure 3-7

2. Remove the 1488 in location 4A and replace it with the Current Loop circuit built on a 14-pin header. This circuit is shown in Figure 3-8 and is constructed as follows:

   a. Connect a 2N3904 transistor to the 14-pin header with the emitter (E) lead connected to pin 7, the base (B) lead connected to pin 5 and the collector (C) lead connected to pin 6.

   b. Solder a 5.6K ohm 1/4 Watt resistor between pin 4 and pin 12 on the header.

   c. Solder a 1K ohm 1/4 Watt resistor between pin 8 and pin 14 on the header.

Current Loop Circuit

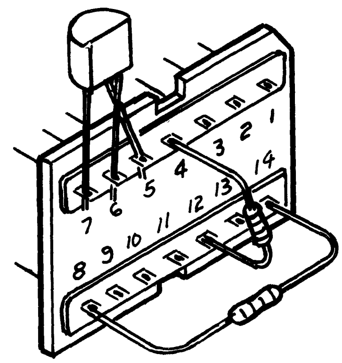


Figure 3-8

3. Connect a 25-pin D-type connector to the terminal cable as follows:

   pin 9 to the printer +lead
   pin 3 to the printer -lead
   pin 2 to the keyboard +lead
   pin 10 to the keyboard -lead

   The procedure is then complete.

D. Asynchronous Printers

As noted earlier, most asynchronous printers can be connected to the SIO with no configuration changes. For a few printers, however, the buffer full status signal may be on an alternate pin.

The SIO supports printers that indicate buffer full status on Pin 20 (DTR) or on pin 19 (SCA). Consult the manual for your printer to determine which pin is used to indicate buffer full status. Depending on the manufacturer, this signal may be identified as "Printer Ready" or "Buffer Full."

As shipped, the SIO expects the buffer full signal on pin 20. If this signal is on pin 19, the SIO Board must be modified as shown in Figure 3-9.

Buffer Full Modification

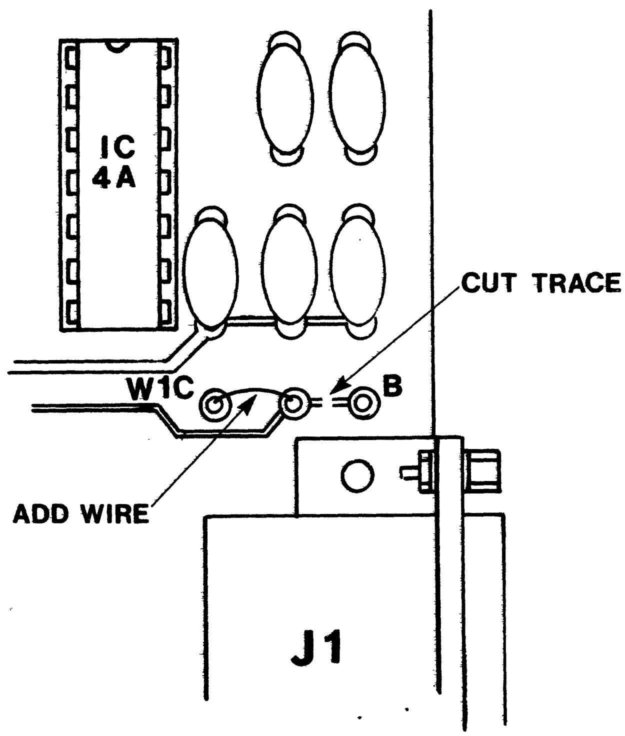


Figure 3-9

E. Asynchronous Baud Rate Selection

The baud rate is selected by a combination of the USART command to "divide by 16" or to "divide by 64" and the value placed in the Baud Rate register. This register is loaded via I/O address X8H, where X is determined by the board slot occupied by the SIO board (see Table 3-19). Table 3-22 shows the values that produce the commonly used baud rates.

Table 3-22

Asynchronous Baud Rate Selection

Baud Rate	USART set to ÷ 16		USART set to ÷ 64	
	Baud Rate Register		Baud Rate Register	
	Decimal	Hexadecimal	Decimal	Hexadecimal
19200	127	7F	--	--
9600	126	7E	--	--
4800	124	7C	127	7F
2400	120	78	126	7E
1200	112	70	124	7C
600	96	60	120	78
300	64	40	112	70
200	32	20	104	68
150	0	00	96	60
110	--	--	84	54
75	--	--	64	40
50	--	--	32	20
45	--	--	22	16

F. Asynchronous Programming Examples

Table 3-23 illustrates a method of programming the SIO Board for asynchronous operation.

Table 3-23

Sample Asynchronous I/O Routines for SIO Board

```
0000            ;
0000            ;
0030            PORTA  EQU   30H     ; Set for SIO boardlet in slot three.
0038            BAUD   EQU   PORTA+8 ; Set Baud rate for channel
0030            DATA   EQU   PORTA   ; USART data address
0031            CTRL   EQU   PORTA+1 ; USART control/status.
007F            BDRT   EQU   127     ; Set Baud rate of 19.2K Baud
0000            ;
0000            ;
0000            ;
0000            ;              Input and output routines
0000            ;
0000  DB31      CINA   IN    CTRL             ; Check USART status
0002  E602             ANI   2                ; Get RxReady bit
0004  28FA             JRZ   CINA             ; Wait till character ready
0006  DB30             IN    DATA             ; Read character
0008  E67F             ANI   7FH              ; Mask off top bit
000A  C9               RET
000B            ;
000B  DB31      COUTA  IN    CTRL             ; Check USART status
000D  E601             ANI   1                ; Get TxReady bit
000F  28FA             JRZ   COUTA            ; Wait till ready
0011  78               MOV   A,B              ; Output char is in B reg
0012  D330             OUT   DATA             ; Output character
0014  C9               RET
0015            ;
0015            ; SIO Boardlet initialization routine
0015            ;
0015  3E7F      INIT   MVI   A,BDRT
0017  D338             OUT   BAUD             ; Set baud rate
0019            ;
0019            ; Interrupt masks are cleared at power up
0019            ;
0019  3E03             MVI   A,3              ; Give USART commands
001B  D331             OUT   CTRL             ; to reset.
001D  D331             OUT   CTRL
001F  3E40             MVI   A,40H
0021  D331             OUT   CTRL
0023  3ECE             MVI   A,0CEH           ; Give mode command
0025  D331             OUT   CTRL             ; 2 STOP BITS, 16*CLK,
0027  3E27             MVI   A,27H            ; Give command.
0029  D331             OUT   CTRL             ; CMD: RTS,ER,RXF,DTR,TXEN
002B  CD2E00           CALL  INJNK            ; Read junk twice
002E  DB30      INJNK  IN    DATA
0030  C9               RET
0031            ;
0031                   END
SYMBOL TABLE

BAUD  0038 00  BDRT  007F 00  CINA  0000 01  COUTA 000B 01  CTRL  0031 00  DATA  0030 00
```

3.9.8 SIO in Synchronous Mode

A. Synchronous Modem Configuration

As shipped, the SIO is configured for operation as an asynchronous modem. It can be reconfigured for synchronous operation as described below.

1. Wire an 8-pin header as shown in Figure 3-10, and install it in the Clock Header socket, board location 1A.

Synchronous Modem Clock Header

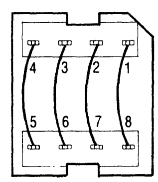


Figure 3-10

2. Remove the Configuration Header, board location 3A, and replace it with a 16-pin header wired as shown in Figure 3-11.

Synchronous Modem Configuration Header

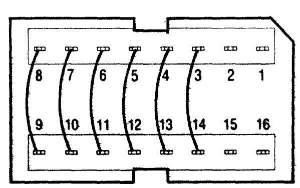


Figure 3-11

B. Synchronous Terminal Configuration

As shipped, the SIO is configured for operation as an asynchronous modem. It can be reconfigured as a synchronous terminal as described below.

1. Wire an 8-pin header as shown in Figure 3-12, and install it in the Clock Header socket, board location 1A.

Synchronous Terminal Clock Header

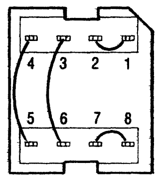


Figure 3-12

2. Remove the Configuration Header, board location 3A, and replace it with a 16-pin header wired as shown in Figure 3-13.

Synchronous Terminal Configuration Header

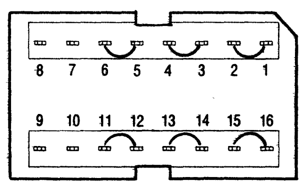


Figure 3-13

C. Synchronous Baud Rates

During synchronous operation, the receiving port speed is determined by the clock signal generated by the transmitting port. Thus, the SIO baud rate selection determines only the transmission speed for a particular port, not the receiving baud rate.

The baud rate is programmed by outputting a value to the Board Rate register. This register is loaded via I/O address X8H, where X is determined by the board slot occupied by the SIO Board (see Table 3-19). Table 3-24 shows the values that produce the commonly used baud rates. The lowest rate is 2400 baud and the highest rate is 51K baud. Rates higher than 51K baud should not be used as this exceeds the upper frequency limit of the 8251 USART.

Table 3-24

Synchronous Baud Rate Selection

Baud Rate	Baud Rate Register	
	Decimal	Hexadecimal
51000	122	7A
38400	120	78
19200	112	70
9600	96	60
4800	64	40
2400	0	00

D. Synchronous Programming Example

Table 3-25 provides an example of programming the SIO to communicate with a synchronous device.

Table 3-25

Sample Synchronous I/O Routines for SIO Board

```
0000              ;
0000              ;
0000              ;
0000              ;
0000              ;     INIT initializes the USART for synchronous operation.
0000              ;
0000              ;     SYNI loads a received message into RAM starting
0000              ;          at the address given in HL.
0000              ;
0000              ;     SYNO transmits a message from RAM starting at the
0000              ;          address given in HL.  The number of bytes of
0000              ;          the message is given in BC.
0000              ;
0000              ; As the data transferred is binary and may contain any character,
0000              ; an escape character must be used to indicate the presence of
0000              ; control characters such as End-of-text, Start-of-text  and Sync.
0000              ; The escape character used is DLE, 10H.  If a DLE character
0000              ; occurs in the data this is replaced by two DLEs in sequence.
0000              ;
0000              ;
0002              STX    EQU   2        ; Start of text character
0003              ETX    EQU   3        ; End of text character
0010              DLE    EQU   10H      ; Data Link Escape character
0016              SYN    EQU   16H      ; Sync character
0000              ;
0001              TXRDY  EQU   1        ; USART status bits
0002              RXRDY  EQU   2
0000              ;
0030              PORTA  EQU   30H      ; Set for SIO boardlet in slot three.
0038              BAUD   EQU   PORTA+8  ; Set Baud rate for channel
0030              DATA   EQU   PORTA    ; USART data address
0031              CTRL   EQU   PORTA+1  ; USART control/status.
0000              ;
0078              BDRT   EQU   120      ; Set Baud rate of 38.4 Khz
0000              ;
0000  3E78        INIT   MVI   A,BDRT   ; Set Baud rate
0002  D338               OUT   BAUD     ; for SIO boardlet
0004  3E80               MVI   A,80H    ; Ensure USART is cleared
0006  D331               OUT   CTRL     ; as specified by manufacturers
0008  D331               OUT   CTRL
000A  3E40               MVI   A,40H    ; do reset
000C  D331               OUT   CTRL
000E              ;
000E  3E0C               MVI   A,0CH    ; Double sync, no parity
0010  D331               OUT   CTRL
0012  3E10               MVI   A,DLE    ; Sync character #1
0014  D331               OUT   CTRL
0016  3E16               MVI   A,SYN    ; Sync character #2
0018  D331               OUT   CTRL
001A  3EB7               MVI   A,0B7H   ; Hunt,RTS,Error reset,RxE,DTR,TxE
001C  D331               OUT   CTRL
001E  DB30               IN    DATA     ; Read junk
0020  C9                 RET
0021              ;
0021              ; Synchronous input routine (RAM address in HL)
0021              ;
0021  CD0000      SYNI   CALL  INIT     ; Set USART into hunt mode and
0024  CD5100             CALL  GETCH    ; reset errors
```

Table 3-25 (continued)

```
0027  FE10          CPI   DLE
0029  20F6          JRNZ  SYNI      ; Wait for DLE to appear
002B  CD5100        CALL  GETCH
002E  FE16          CPI   SYN       ; If SYNC, try again
0030  28EF          JRZ   SYNI
0032  FE02          CPI   STX       ; Check for start of text,
0034  20EB          JRNZ  SYNI      ; if bad, try again
0036         ;
0036         ; Transfer message into RAM
0036         ;
0036  CD5100 SDATA  CALL  GETCH
0039  FE10          CPI   DLE
003B  2010          JRNZ  RAMLD     ; If not DLE then data
003D  CD5100        CALL  GETCH     ; Get second char of DLE seq
0040  FE10          CPI   DLE       ; If DLE-DLE then use one
0042  2809          JRZ   RAMLD     ; of them as data
0044  FE16          CPI   SYN       ; Check for padding (SYNC chars)
0046  28EE          JRZ   SDATA     ; ignore if it is
0048  FE03          CPI   ETX       ; End yet ?
004A  C8            RZ    ;         ; If not done, then bad DLE
004B  18E9          JR    SDATA     ; sequence found, ignore it
004D         ;
004D  77     RAMLD  MOV   M,A       ; Insert byte into RAM at (HL)
004E  23            INX   H
004F  18E5          JR    SDATA     ; Get next byte
0051         ;
0051  DB31   GETCH  IN    CTRL      ; Get char from serial port
0053  E602          ANI   RXRDY
0055  28FA          JRZ   GETCH     ; Wait till done
0057  DB30          IN    DATA
0059  C9            RET
005A         ;
005A         ; Synchronous output routine
005A         ; Outputs BC characters starting at address in HL
005A         ;
005A  CD0000 SYNO   CALL  INIT      ; Reset USART
005D  C5            PUSH  B         ; Save byte count
005E  0600          MVI   B,0       ; Send 255 DLE-SYNCs
0060  3E10   HEADR  MVI   A,DLE     ; before message
0062  CD9100        CALL  OPCH
0065  3E16          MVI   A,SYN
0067  CD9100        CALL  OPCH
006A  10F4          DJNZ  HEADR
006C  C1            POP   B         ; Restore byte count
006D         ;
006D  3E10          MVI   A,DLE     ; Send message header of
006F  CD9100        CALL  OPCH      ; DLE STX
0072  3E02          MVI   A,STX
0074  CD9100        CALL  OPCH
0077         ;
0077         ; Transfer message contents
0077         ;
0077  7E     NCHO   MOV   A,M
0078  CD9100        CALL  OPCH      ; Output byte of data
007B  3E10          MVI   A,DLE     ; DLE for comparison
007D  EDA1          CPII  ;         ; Check if char was DLE and count
007F  CC9100        CZ    OPCH      ; Output second DLE if it was
0082  EA7700        JPE   NCHO      ; Loop till done
0085  CD9100        CALL  OPCH      ; Output DLE from A
```

Table 3-25 (continued)

```
0088  3E03            MVI   A,ETX   ; Send End of text
008A  CD9100          CALL  OPCH    ;
008D  CD0000          CALL  INIT    ; Stop SYNC characters
0090  C9              RET   ;       ; Return to calling program
0091            ;
0091  F5        OPCH  PUSH  PSW     ; Output Character
0092  DB31      WTX   IN    CTRL    ; Get USART status
0094  E601            ANI   TXRDY   ; Check if ready for character
0096  28FA            JRZ   WTX     ; Wait till it is
0098  F1              POP   PSW     ; Get character back and
0099  D330            OUT   DATA    ; output
009B  C9              RET
009C            ;
009C                  END
SYMBOL TABLE

BAUD  0038 00  BDRT  0078 00  CTRL  0031 00  DATA  0030 00  DLE   0010 00  ETX   0003 00
```

3.10 PIO Board

The PIO (Parallel Input Output) Board is used to drive parallel printers and other devices requiring transfer of data in 8-bit parallel form.

The PIO Board contains a configuration header which allows it to adapt to many different device interfaces. This header changes the way that the components on the board are connected. Since the header can be wired in many ways, only one configuration is discussed here, i.e., with the header wired as shown in Figure 3-14.

Standard PIO Configuration Header

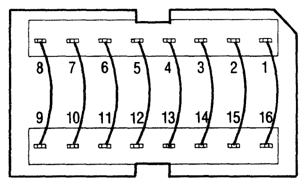


Figure 3-14

This is the standard North Star configuration. To determine the affect that other configurations would have on the operation and programming of the PIO board, refer to the PIO board schematic in Appendix I.

3.10.1 Reset

When the I/O Reset bit (I/O address F0H, bit 4) is set on, then off, its only effect on the PIO Board is to reset the Interrupt Mask to all zeros. See CONTROL heading below. Note that the I/O Reset bit resets all I/O boards simultaneously.

3.10.2 Board ID

The 8-bit identification code for the PIO board is DBH. The I/O address used to input this code depends on the board slot occupied by the PIO board (see Table 3-19).

3.10.3 Data Transfers

The I/O address used to transfer a data byte to or from the PIO board is X0H, where X is determined by the board slot occupied by the PIO (see Table 3- 9). The standard location for the PIO Board is slot 2.

3.10.4 Control

The operation of the PIO Board is controlled by specifying the Interrupt Mask, and by setting and resetting the Input and Output flags. These flags are input as part of the status byte and may be used to generate maskable interrupts.

The format of the Interrupt Mask is shown in Table 3-26. A one in any of the bit positions 4 through 7 enables the PIO Board to generate a maskable interrupt if the stated condition is true. The program defines this mask by outputting the appropriate bit pattern to I/O address X2H, where X is determined by the board slot occupied by the PIO Board (see Table 3-19).

The program initializes the Input flag by resetting it. The input device sets the flag when an input byte is ready at the device interface. After the byte is input, the program again resets the flag, and the cycle is repeated.

The Input flag is reset by accessing I/O address X6H, where X is determined by the board slot occupied by the PIO Board (see Table 3-19). In the standard configuration of the PIO Board, the Input flag is not normally set by the program. The flag could be set by accessing I/O address X7H, where X is determined in the same manner as for resetting the flag.

The program initalizes the Output flag by resetting it. The output device sets the flag when it is ready to receive a byte. After the byte is transferred, the program again resets the flag, and the cycle is repeated.

Table 3-26

PIO Interrupt Mask Format

7	Input Flag. The Input flag is set.
6	Output Flag. The Output flag is set.
5	Input Strobe. The device interface signal IN STROBE is at a low logic level.
4	Output Acknowledge. The device interface signal OUT ACK is at a low logic level.
3 2 1 0	Ignored

NOTE: A one in any bit position 4 through 7 allows the stated condition to generate a maskable interrupt.

The Output flag is reset by accessing I/O address X4H, where X is determined by the board slot occupied by the PIO Board (see Table 3-19). In this configuration of the PIO Board, the Output flag is not normally set by the program, although it could be set by accessing I/O address X5H, where X is determined in the same manner as for resetting the flag.

3.10.5 Status

A status byte may be read from the PIO Board by inputting from I/O address X1H, where X is determined by the board slot occupied by the PIO Board, (see Table 3-19). Table 3-27 shows the format of the Status byte. The operation of the Input and Output flags is discussed under the CONTROL heading above.

Table 3-27

PIO Status Byte Format

7 6 5 4 3 2 1 0

Output Acknowledge. A one indicates that the device interface signal OUT ACK is at a low logic level.

Input Acknowledge. A one indicates that the device interface signal IN ACK is at a low logic level.

Output Flag. The Output flag is set.

Input Flag. The Input flag is set.

All ones.

3.10.6 Interrupt or Polled

The PIO Board may be serviced in the interrupt mode or it may be polled by the program.

If the interrupt mode is used, one or more bits of the Interrupt Mask must be set on to allow the PIO Board to generate interrupts. The Interrupt Mask is discussed in Section 3.10.4.

When the PIO Board causes an interrupt, the program must determine the source of the interrupt. It does this by inputting from I/O address E0H and checking bit 1. The bit is a zero if any of the I/O boards including the PIO is interrupting. The program then inputs the status of all I/O boards to determine which board(s) is interrupting. The program decides that the PIO Board has interrupted if one of the four status bits is a one, and the corresponding bit in the Interrupt Mask is also a one.

If the PIO Board is to be polled, the Interrupt Mask must be loaded with zeros. The program polls the PIO by perodically reading the board status and taking appropriate action.

Table 3-28

Parallel I/O Addresses		
I/O Address (Hexadecimal)	Operation	Description
XO	INPUT	Input Data Byte.
X0	OUTPUT	Output Data Byte.
X1	INPUT	Input Status Byte (see format in Table 3-27).
X2	OUTPUT	Output to Interrupt Mask (see format in Table 3-26).
X3		Not used.
X4	INPUT/OUTPUT	Reset Output flag.
X5	INPUT/OUTPUT	Set Output flag.
X6	INPUT/OUTPUT	Reset Input flag.
X7	INPUT/OUTPUT	Set Input flag.

NOTES

- The first digit of these I/O addresses is determined by the board slot occupied by the PIO board (see Table 3-19).
- Addresses X8 through XF function the same as addresses XO through X7 respectively.

3.10.7 Programming Example

The subroutine in Table 3-29 provides an example of programming the standard configuration PIO Board to output data.

Table 3-29

Sample Routine For Outputting PIO Data

```
0040   ==     PIO     ==     40H     ; PIO BASE PORT ADDRESS
0040   ==     PDATA   ==     PIO     ; PIO DATA PORT ADDRESS
0041   ==     PSTAT   ==     PIO+1   ; PIO STATUS PORT ADDRESS
0004   ==     POBIT   ==     4       ; PO FLAG BIT MASK
0044   ==     RSFLG   ==     PIO+4   ; ADDR RO RESET OUTPUT FLAG
              L
0000'  DB41   POUT:   IN     PSTAT   ; PIO STATUS
0002'  E604           ANI    POBIT   ; TEST OUTPUT
0004'  28FA           JRZ    POUT    ; WAIT FOR DEVICE READY
0006'  D344           OUT    RSFLG   ; RESET OUTPUT FLAG
0008'  78             MOV    A,B     ; CHARACTER TO SEND
0009'  F680           ORI    80H     ; SET STROBE BIT FALSE
000B'  D340           OUT    PDATA   ; SET UP DATA
000D'  EE80           XRI    80H     ; TOGGLE STROBE
000F'  D340           OUT    PDATA
0011'  EE80           XRI    80H     ; TOGGLE STROBE
0013'  D340           OUT    PDATA
0015'  E67F           ANI    7FH     ; CLEAR STROBE BIT
0017'  C9             RET
              ;
                      .END
```

3.11 Speaker Control

The speaker produces sounds that are used to signal the operator of the ADVANTAGE. The program can either produce a standard 'beep' sound, or a programmable sound.

The standard 'beep' sound is a 1920 Hz tone with a duration of one-half second. This sound is produced by inputting from I/O address 83H. The input data is indeterminate.

The programmable sound is produced by manipulating bit 6 of the I/O Control register (I/O address 0FH). When this bit is complemented at the proper rate, a tone is produced in the speaker. For example, complementing the bit once every millisecond will produce a 500 Hz tone. The tone is maintained as long as the bit is being complemented. Note that complex sounds may be generated by complementing the bit at an irregular rate.

3.12 BOOTSTRAP FIRMWARE

The Bootstrap program is contained in the Boot PROM (see Section 4.1.3). The Bootstrap program loads other programs from diskette or from a serial port via an SIO Board.

3.12.1 Startup

The Bootstrap program may be entered by generating a non-maskable interrupt (see Section 3.3.2), or by executing the following two instructions:

1. Output 84H to I/O address A2H.
2. Jump to address 8066H.

When the Bootstrap program is entered, it performs the following sequence:

1. The Z80 processor registers are pushed into the existing stack in the following sequence: AF,B,D,H, alternate AF, alternate B, alternate D, alternate H, alternate IX and alternate IY. Finally, the interrupt vector is pushed.

2. The stack pointer is put in register IY. If the Bootstrap program was entered as the result of a power reset, register IY contains 0001H.

3. The Display RAM is mapped into 0000H through 7FFFH, the Boot PROM is mapped into 8000H through BFFFH, and the first 16K bytes of Main RAM are mapped into C000H through FFFFH.

4. A beep sounds, and the message 'LOAD SYSTEM' is displayed.

The Bootstrap program then waits for instructions entered from the keyboard. These instructions may cause it to boot from drive 1, boot from drive 2, or boot from a serial port (see Section 2.2).

3.12.2 Boot from Disk Drive

If the Bootstrap program is directed to boot from one of the disk drives, it performs the following sequence:

1. Sectors 4,5,6 and 7 on track 0 are read into Main RAM. The first data byte in sector 4 determines the starting location of the area in Main RAM in which the program is stored.

   For example, if the first data byte is C0H, this byte is stored in location C000H, and remaining data bytes in sectors 4,5,6 and 7 are stored sequentially from that point. This first byte must be in the range C0H through F8H.

2. The first 16K bytes of Main RAM are mapped into 0000H through 3777H and 4000H through 7FFFH.

3. A jump is made to the load address + 10. This location must contain the op code for a jump instruction.

If the boot attempt is unsucessful, a beep sounds and the 'LOAD SYSTEM' message is redisplayed. There are five ways that a failure may occur:

1. Diskette not loaded.

2. Machine malfunction.

3. Uncorrectable read error (wrong CRC byte). The CRC byte is calculated by the routine shown in Table 3-30.

4. Wrong sync byte. The first sync byte is FBH. The second sync byte is the sector number plus 16 times the track number, truncated to eight bits.

5. The first byte of sector 4 is not in the range C0H through F8H, or the tenth byte of sector 4 is not C3H.

Table 3-30

Boot PROM CRC Routine

```
814E   DB80     READL  IN     RDATA  ;GET BYTE
8150   FEC0            CPI    0C0H
8152   D8              RC
8153   FEF9            CPI    0FSH
8155   D0              RNC
8156   57              MOV    D,A ; MSB OF STORE ADDRESS
8157   12              STAX   D ;STORE IT ALSO
8158   13              INX    D
8159   07              RLC
815A   4F              MOV    C,A ;START OF CRC VALUE
815B   216581          LXI    H,BLOOP ;SET NEW RETURN ADDRESS
815E   DB80            IN     RDATA ;GET SECOND BYTE
8160   12              STAX   D
8161   13              INX    D
8162   A9              XRA    C
8163   07              RLC    ;CRC CALC
8164   4F              MOV    C,A
8165   DB80     BLOOP  IN     RDATA ;READ DATA LOOP
8167   12              STAX   D
8168   A9              XRA    C ;FORM CRC
8169   07              RLC
816A   4F              MOV    C,A
816B   13              INX    D ;UPDATE STORE ADDRESS
816C   DB80            IN     RDATA ;SECOND BYTE
816E   12              STAX   D
816F   A9              XRA    C
8170   07              RLC
8171   4F              MOV    C,A
8172   13              INX    D
8173   10F0            DJNZ   BLOOP
8175            ;HAVE COMPLETED A BLOC, GET CRC
8175   DB80            IN     RDATA ;CRC BYTE
8177   A9              XRA    C ;SEE IF IT MATCHES COMPUTED CRC
8178   DB82            IN     RENBL ;CLEAR READ ENABLE
817A   20A1            JRNZ   READA ;IF NOT, GO READ AGAIN
```

3.12.3 Boot from Serial Port

In order to use this feature, an SIO board must be installed in I/O slot 3, and the board ID must be in the range F0H through F7H. The board must be connected to a synchronous communication link.

If the Bootstrap program is directed to boot from serial port, it configures the USART as follows:

Synchronous Mode
2400 baud
Two sync bytes - DLE,SYN
Eight bits per word
Two stop bits
Parity off

After the USART is configured, it should be receiving sync bytes. If sync is not detected within 1 second, a beep sounds and 'LOAD SYSTEM' is redisplayed. If sync is detected, the following 'diaglogue' should occur:

```
Other system:DLE,SYN,ENQ,PAD         "WHAT DO YOU WANT?
ADVANTAGE:DLE,SYN,EOT,NUM,ENQ PAD "I WANT THE PROGRAM"
Other system:STX,<data>,ETX,SUMLO,       "HERE IT IS"
                        SUMHI, PAD
```

STX=02H,ETX=03H,EOT=04H,ENQ=05H,DLE=10H,SYN=16H,PAD=0FFH
NUM = boot type number (01H for the ADVANTAGE)
SUMHI,SUMLO=checksum computed as((sum of all data bytes) +1) mod 65536

The Boot program can wait indefinitely for the "What do you want?" message. When it is received, it sends the "I want the program" message. Then it can wait indefinitely for the STX. When the STX arrives, the Boot program assumes that subsequent data is the program.

The first byte after the STX determines the starting location of the area in Main RAM into which the program is loaded. For example, if the first byte is C0H this byte is stored in location C000H, and the remainder of the program is stored sequentially from that point. This first byte must be in the range C0H through F8H.

The DLE character has special significance in the data stream as follows:

1. Two DLE's in a row are stored as one DLE.

2. Pairs of sync bytes DLE, SYN are dropped.

3. DLE,DLE,SYN is stored as DLE,SYN.

4. Single DLE's not followed by SYN or ETX are dropped.

5. The pair DLE,ETX signals end of program and is not stored.

Only those bytes that are stored in the RAM are included in the checksum. The checksum is computed as ((sum of all data bytes)+1) mod 65536. If the computed checksum does not match the checksum in the message, a beep sounds and the message 'LOAD SYSTEM' is redisplayed. If the checksums match, the first 16K bytes of Main RAM is mapped into locations 0000H through 3FFFH and 4000H through 7FFFH, and a jump is made to the load address + 10.

This chapter discusses the theory of operation of the Main PC Board, the Serial Input Output (SIO) Board and the Parallel Input Output (PIO) Board.

The block diagrams in the chapter are coordinated with the schematics in Appendix I. Each block that represents circuitry on a PC board corresponds to a page of the schematics or to a shaded section of a page. In addition, the names used in the blocks are the same as those used in the schematics.

4.1 MAIN PC BOARD

Figure 4-1 is a block diagram of the ADVANTAGE computer system. The shaded blocks represent the elements of the system which are on the Main PC Board.

The ADVANTAGE System Block Diagram

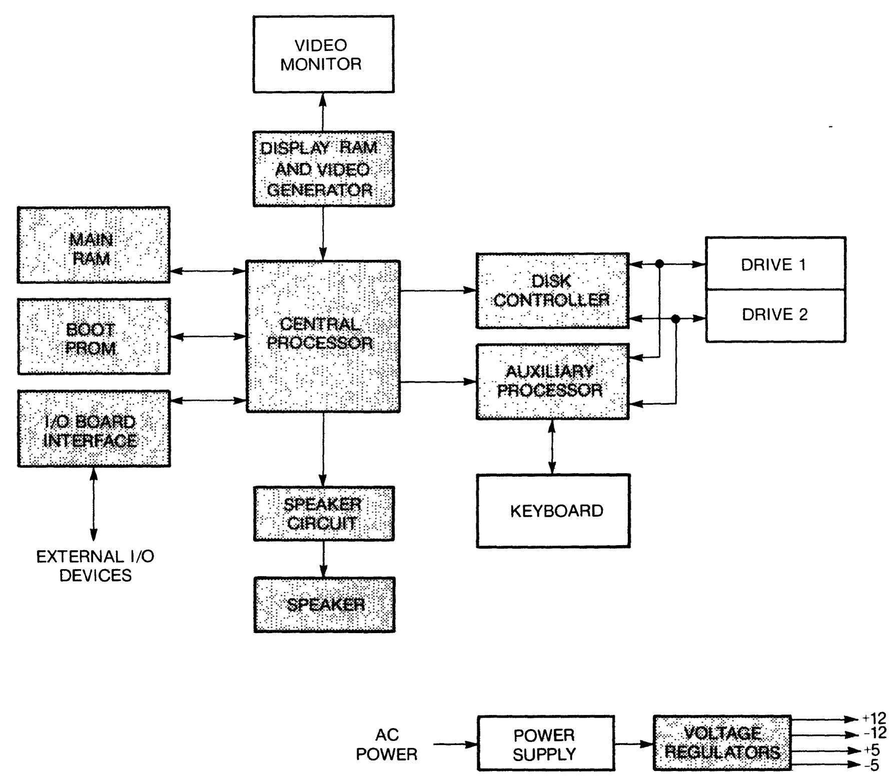


Figure 4-1

The Central Processor is in primary control of the ADVANTAGE system. It controls the flow of data between the I/O devices and the Main RAM. It also checks status on these devices, issues commands, and responds to interrupts.

The Central Processor performs its duties by executing the programs residing in the Boot PROM and the Main RAM. The programs contain Z80 processor instructions. See Appendix G for a list of these instructions and a description of the Z80 microprocessor.

The Boot PROM contains the bootstrap routine that loads programs into the Main RAM. Programs may be loaded from diskette or from a serial port connected to the I/O board interface. The Boot PROM also contains a video driver routine and a monitor routine. See Sections 3.6.5 and 6.4 for additional information on these routines.

The Main RAM is used to store programs and data. The storage capacity is 64K bytes by nine bits including parity. Parity checking is used to insure the integrity of the stored information.

The Display RAM stores data to be displayed on the Video Monitor. The capacity of this RAM is 20K bytes by eight bits with no parity. The Display Controller serializes the data and sends it to the Video Monitor. It also provides the Monitor with horizontal and vertical sync signals.

The Disk Controller performs most of the control functions for the disk drives. It selects the drive, selects a side of the diskette, positions the read/write head, and performs the read or write operation.

The Auxiliary Processor performs the remaining disk operations. It turns the drive motors on and off, keeps track of the sector number, and determines the width of the sector pulse. The Auxiliary Processor also controls the keyboard. It scans the keyboard, converts the scanning information to the correct character code, and notifies the Central Processor when keyboard data is available.

The Speaker is a small audio transducer located on the Main PC Board. The Speaker circuit can produce either a standard 'beep' sound or a programmable sound. The I/O Board Interface consists of six PC board connectors and associated bus drivers and command decoders. The PC boards used in this area can interface external I/O devices to the Central Processor, or they can expand the computing power of the Central Processor.

The voltage regulators receive DC power from the Power Supply and produce four regulated DC supply voltages that are used throughout the ADVANTAGE system. The voltages are: +12, -12, +5 and -5.

4.1.1 Central Processor

A block diagram of the Central Processor is shown in figure 4-2. The Central Processor uses two address buses and three data buses. Multiple buses are required because the Z80 processor interfaces with a large number of circuits.

Any address placed on the Address (ADR) bus automatically appears on the Buffered Address (BA) bus. The same is true of data placed on the Data bus - it automatically appears on the RAM data (RD) bus. Transfers between the Data bus and the Buffered Data (BD) are controlled by the I/O Select PROM, and depend upon the direction of data flow.

The Z80 processor is the heart of the Central Processor. When it fetches instructions it places the instruction address on the Address bus and reads the instruction from the Data bus. It reads status by inputting from the I/O controller, the Auxiliary Processor and I/O Status register 1. It issues commands by outputting to the I/O controllers, and to the I/O Control Register. See Appendix G for more information about this microprocessor.

Central Processor Block Diagram

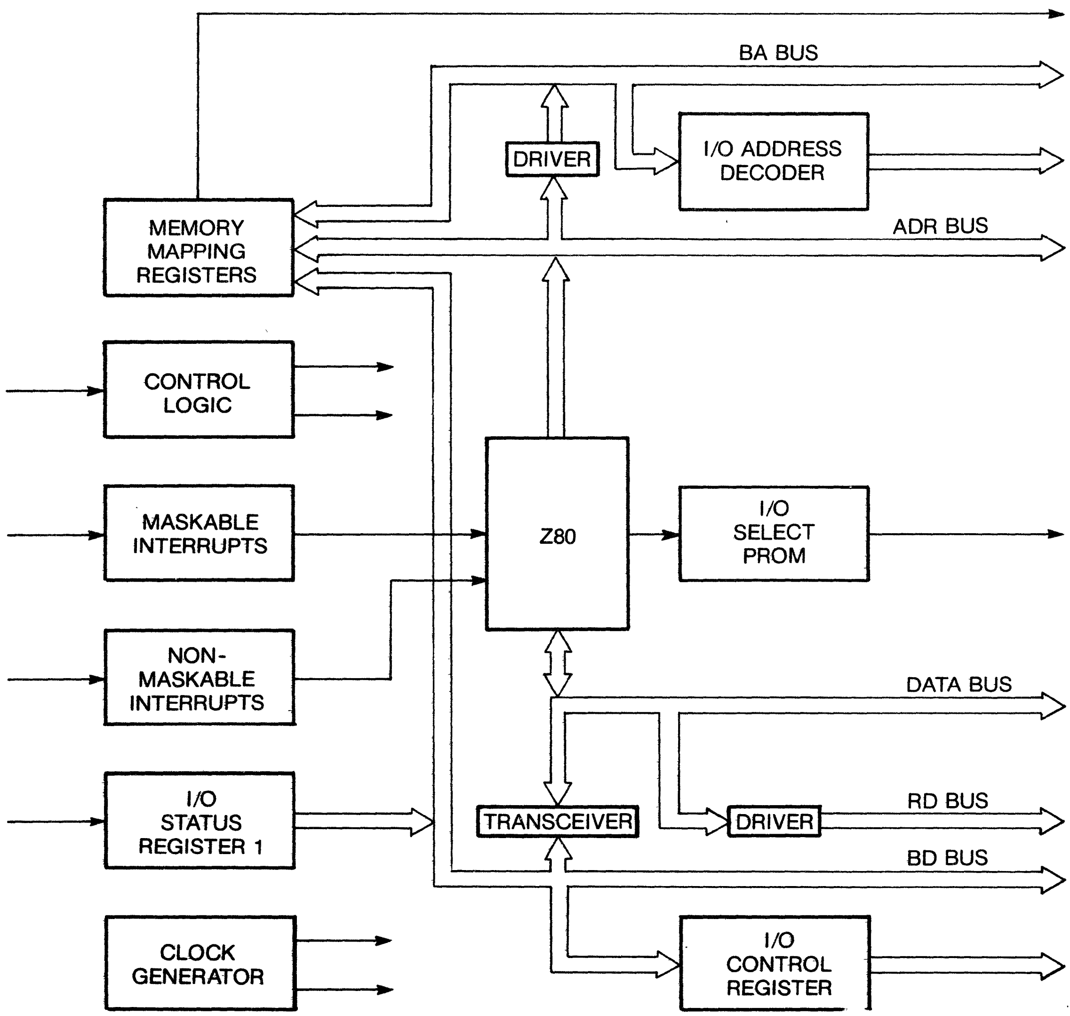


Figure 4-2

The Memory Mapping registers expand the memory addressing capabilities of the ADVANTAGE computer from 64K bytes to 256K bytes. See Section 3.2.1 for detailed information on their use.

The Memory Mapping registers are implemented by a 74LS670 scratch pad RAM. The RAM contains four locations with four bits per location. Each location represents one mapping register.

When data is written into a register, the BA bus selects the register, and the BD bus contains the data to be written. When data is read from a register, the ADR bus selects the register, and the contents of the register are used to select the 16K section of memory to be accessed. Note that it is possible to select a non-existent section of memory, because some of the allocated address space is not used (see Table 3-1).

The Control Logic maintains the Display flag, and controls the wait input signal to the Z80 processor.

The Display flag is set at the end of each vertical scan (signal PL SYNC) and reset when the program executes an input or output instruction to I/O address B0H.

Two conditions may cause the Z80 processor to go into a wait state:

1. The program has initiated an access to the Display RAM and data is not yet available (signal WAIT A).

2. The program has initiated a disk operation and the Disk Controller has not completed the operation (signal $\overline{\text{WAIT 1}}$, $\overline{\text{WAIT 2}}$ and $\overline{\text{WAIT 3}}$).

The maskable interrupt circuitry generates a maskable interrupt to the Z80 processor if any of the following conditions are true:

1. Keyboard data is available (signal KB INT).

2. The Display flag is set.

3. One of the I/O boards is interrupting.

4. A parity error occurs in Main RAM (signal $\overline{\text{PINT}}$).

The non-maskable interrupt circuitry generates a non-maskable interrupt to the Z80 processor when any of the following conditions are true:

1. Power has just been turned on, or power has been interrupted (signal $\overline{\text{PWR RES}}$).

2. The reset pushbutton is pressed. This is the momentary contact switch located on the rear panel of the ADVANTAGE cabinet (signal $\overline{\text{PNMI}}$).

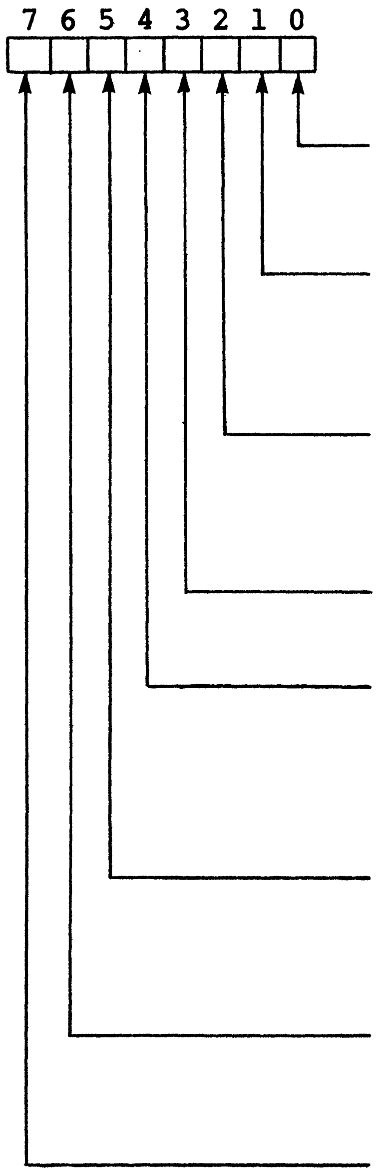
3. The keyboard reset is active (signal INT 48). This reset is under program control (see Section 2.1.4).

The Main RAM parity error can be made to generate a non-maskable interrupt instead of a maskable interrupt by changing the position of jumper W4 on the Main PC Board, but this connector is not supported by North Star software.

I/O status register 1 is an 8-bit bus driver through which eight status signals are input from various parts of the system. When an input instruction is executed from any of the I/O addresses E0H through EFH, the status signals are transferred to the BD bus and from there into the Z80 processor. Table 4-1 defines the signals that are input.

Table 4-1

I/O Status Register 1 Format



Keyboard Interrupt. A one indicates that a valid keyboard entry has caused a maskable interrupt.

I/O Interrupt. A zero indicates that one or more of the I/O boards is generating a maskable interrupt. The bit is set by clearing the interrupting condition.

Display Flag. This bit is set at the end of each vertical scan. It is cleared by accessing I/O address B0H (See Section 3.6.3).

Non-Maskable Interrupt. A zero indicates that a non-maskable interrupt is present.

Disk Write Protect. A one indicates that the selected disk drive is write protected. If no drives are selected or if the selected drive has no write protect option this bit in indeterminate.

Track 0. A one indicates that the selected disk drive is positioned at Track 0. If no drive is selected, this bit is indeterminate.

Sector Mark. This bit changes momentarily from a one to a zero at the beginning of each sector on the selected disk drive.

Disk Serial Data. This bit is connected directly to the serial data stream coming from the selected disk drive. It is used by the program to synchronize disk read operations.

The Clock Generator consists of a crystal oscillator, two flip flops, and a divide-by-16 counter. These circuits generate the following clocks which are used throughout the Main PC board: 8 MHz, 4 MHz, 2 MHz, 0.5 MHz, 0.25 MHz and 0.125 MHz.

The I/O Address Decoder produces some of the individual signals required to carry out I/O instructions. These signals are listed in Table 4-2 along with the corresponding decoder output.

Table 4-2

I/O Address Decoder Signals

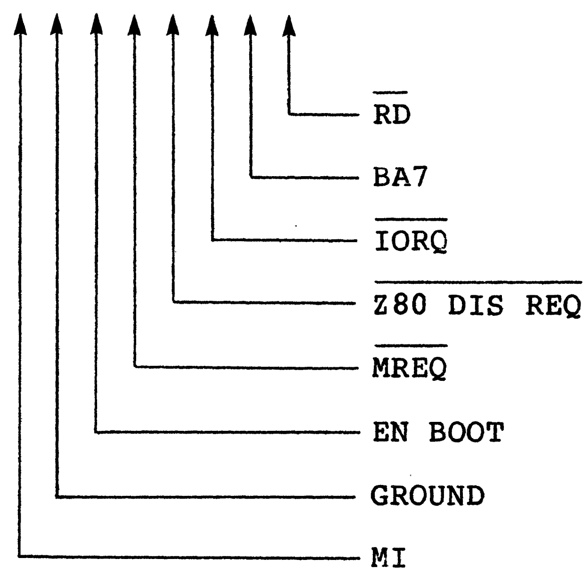
Output	Description
0	Partial decode of disk I/O instructions and instruction to produce the standard 'beep' sound.
1	Load Start Scan register located in the Video Generator.
2	Load Memory Mapping register. Bits 0 and 1 of the BA bus specify which register is loaded.
3	Clear Display flag.
4	Clear non-maskable interrupt.
5	Input from I/O Status Register 2 located in the Auxiliary Processor.
6	Input from I/O Status Register 1.
7	Load I/O Control Register.

The I/O Select PROM produces four control signals which make data available to the Z80 processor by transferring data to the Data bus. Each control signal transfers the data from a different source. Table 4-3 defines the contents of this PROM and summarizes its input and outputs. The four output signals are described below.

Table 4-3

I/O Select PROM Summary

ADDRESS BITS 7 6 5 4 3 2 1 0	DATA BITS 4 3 2 1	ACTIVE SIGNAL
X X 1 0 X 1 X 0	0 1 1 1	$\overline{\text{RD PROM}}$ - Read Boot PROM
X X X 0 1 1 X 0	1 0 1 1	$\overline{\text{RD RAM}}$ - Read Main RAM
0 X X 1 1 0 0 0	1 1 0 1	$\overline{\text{I/O to Z80}}$ - Input instruction
1 X X 1 1 0 X 1	1 1 0 1	$\overline{\text{I/O to Z80}}$ - Interrupt response
X X X 1 1 0 1 0	1 1 1 0	$\overline{\text{BD to Z80}}$ - Input instruction
X X X 0 0 1 X 0	1 1 1 0	$\overline{\text{BD to Z80}}$ - Reading Display RAM



NOTE: All locations not defined in this table contain all ones and produce no active output signals.

1. $\overline{\text{RD PROM}}$. Transfers data from the Boot PROM to the Data bus. The Z80 processor supplies the address of the data. This transfer can occur if the Memory Mapping registers select the Boot PROM, or if a non-maskable interrupt occurs.

2. $\overline{\text{RD RAM}}$. Transfers data from the Main RAM to the Data bus. The Z80 processor supplies the address of the data.

3. $\overline{\text{I/O to Z80}}$. Transfers data from the I/O board interface to the Data bus. This transfer occurs when the Z80 processor executes an input I/O instruction addressed to an I/O board. It also occurs when the Z80 processor is responding to a maskable interrupt (mode 2 response) and is reading the address vector from the I/O board interface. Note that the address vector from the I/O boards is always FFH.

4. $\overline{\text{BD to Z80}}$. Transfers data from the BD bus to the Data bus. This transfer occurs when the Z80 processor reads from the Display RAM, or when the Z80 processor executes an input instruction addressed to the Disk Controller, to Status Registers 1 or to Status Register 2.

The I/O Control register stores commands that are used throughout the ADVANTAGE system. When the program executes an output instruction to any I/O address from F0H through FFH, the eight control bits are transferred from the Z80 processor, through the BD bus and into the I/O Control register.

Table 4-4 defines the bits of the I/O Control register. The low-order three bits of the register form a command code which is sent to the Auxiliary Processor. The commands are defined in Table 4-5.

Table 4-4

I/O Control Register Format

Bit(s)	Function
2 1 0	I/O Commands. See Table 4-5.
3	Acquire Mode. This bit is momentarily set to zero at the beginning of each disk read operation (see Section 4.1.5).
4	I/O Reset. A zero causes a reset to be sent to the Auxiliary Processor and to the I/O Board Interface. A one removes that reset.
5	Blank Display. A one causes the display to be blanked. A zero enables the data presently in the Display RAM to be displayed.
6	Speaker Data. This bit is fed directly into the speaker control circuitry. Changing the bit at the proper rate will produce a corresponding tone in the speaker.
7	Enable Display Interrupt. A one enables bit 2 of the I/O Status Register 1 to generate a maskable interrupt.

Table 4-5

I/O Commands

Command Number	Bits 0-2 of Control Register	Description
0	000	Show Sector. Place disk sector number into bits 0-3 of I/O Status register 2.
1	001	Show Char LSB's. Place low-order four bits of keyboard character into I/O Status register 2, bits 0-3.
2	010	Show Char MSB's. Place high-order four bits of keyboard character into I/O Status register 2, bits 0-3. Reset Keyboard flag, bit 6 of the same register.
3	011	Keyboard MI Flag. Complement the state of the Keyboard Maskable Interrupt flag. Following execution of the command 3, the state of this flag appear in bit 0 of I/O Status register 2. One=on, zero=off. The KB MI flag allows the Keyboard Data flag, bit 6 of I/O Status register 2, to generate a maskable interrupt.
4	100	Cursor Lock. Change the state of the Cursor Lock flag, and place that flag into bit zero of I/O Status register 2. One = on, Zero = off.

Table 4-5 (continued)

Command Number	Bits 0-2 of Control Reg.	Description
5	101	Start Disk Drive Motors. Turn on both disk drive motors. Motors remain on for 3 seconds after the command is removed. Also perform "Show Sector" command (see above).
6	110	Command Prefix. Used only as part of the command 6, command 7 sequence (see below).
6,7	110,111	Keyboard NMI Flag. This 2-command sequence complements the state of the Keyboard Non-maskable Interrupt flag. Following execution of this command sequence, the KB NMI flag appears in bit 0 of I/O Status register 2. One = on, Zero = off. When this flag is on, the keyboard reset feature is enabled (see Section 2.1.4)
7	111	All Caps. When used alone, this command changes the state of the Cap Lock flag, and places that flag in bit zero of I/O Status register 2. One = on, zero = off.

4.1.2 Main RAM

The Main RAM is a dynamic memory array with a storage capacity of 64K bytes. Each byte contains nine bits, eight for data and one for parity. The parity is odd.

A block diagram of the Main RAM is shown in Figure 4-3.

The address MUX outputs 14 bits of memory address to the RAM, seven bits at a time. These 14 bits select four memory locations, one in each 16K section of the RAM. The Control Logic completes the address decode by selecting one of the four 16K sections. Expressed in terms of the RAM integrated circuits (ICs) the Control Logic selects one of four rows of ICs: row F, row G, row H and row J.

Main RAM Block Diagram

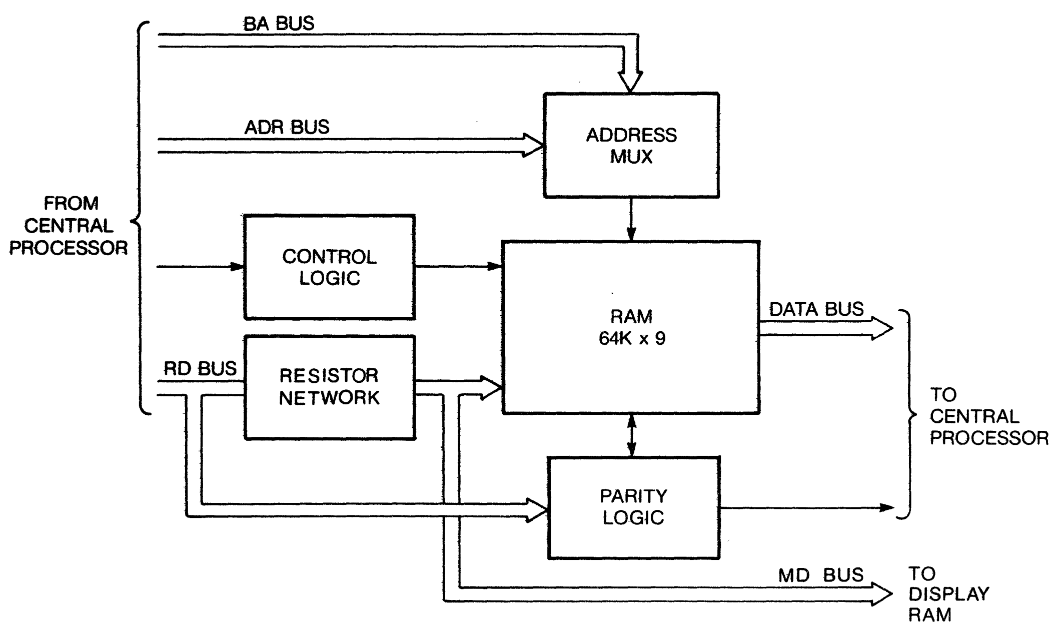


Figure 4-3

When the RAM is accessed for a read or a write, the address bits are latched into the RAM in two steps. First, the seven most significant address bits are latched with the row address strobe (RAS) signals. Then the seven least significant address bits are latched with the column address strobe (CAS) signals.

The RMBWR signal determines whether data is read from or written into the RAM. If this signal is high, data is read from the RAM and placed into an 8-bit latch. The RD RAM signal transfers this data to the Data bus. When RMBWR is low, data is written into the RAM. Data enters the RAM from the RD bus.

Figure 4-4 shows the Main RAM timing for an op code fetch and for a non-op code memory read.

The Main RAM is refreshed only after an op code fetch. The second half of Figure 4-4 shows the timing of the refresh cycle. During refresh, the Z80 supplies the refresh address, and all RAS signals are active,thereby selecting all the RAM ICs simultaneously.

The Resistor Network removes electrical noise from the data imputs of the Main RAM and the Display RAM. This network filters the signals as they pass from the RD bus to the MD bus.

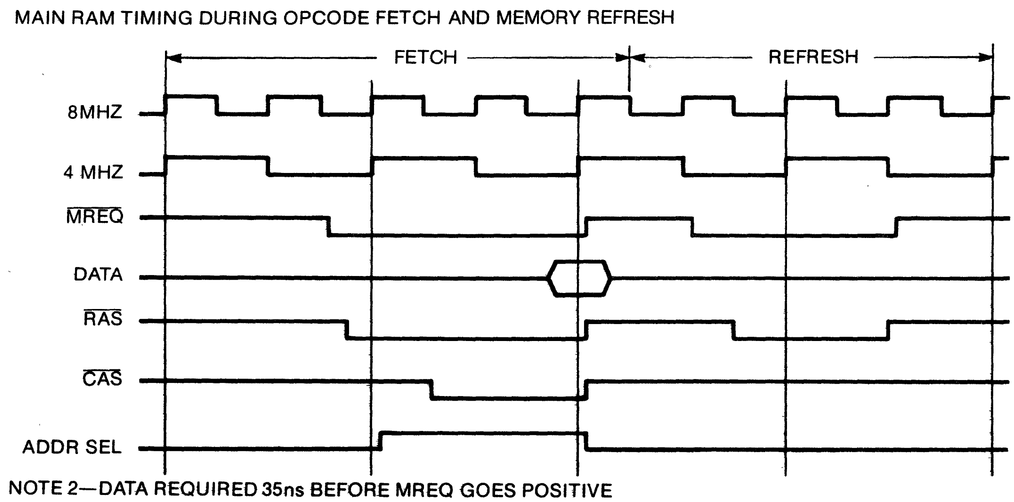
The Parity Logic automatically stores a parity bit in the RAM each time data is written, and checks the parity each time data is read. The Parity Logic may be programmed to generate an interrupt if a parity is detected (see Section 3.2.2).

When a byte is written into the RAM, the Parity Logic computes parity on the RD bus and supplies an odd parity bit to the RAM. When a byte is read from the RAM the Parity Logic computes parity on nine bits-eight bits from the RD bus, and the single parity bit from the RAM. At this time the RD bus contains data read from the RAM, because the RD bus is always a direct copy of the Data bus.

If a parity error is detected, the Parity Error flag is set. If the Parity Logic is programmed to generate interrupts, the Parity Error flag will generate either a maskable or a non-maskable interrupt depending upon the connection of jumper W4. The standard connection for W4 is to allow maskable interrupts. North Star software does not support the alternate connection.

The Parity Error flag may be tested and/or reset by the program (see Section 3.2.2).

Main Ram Timing



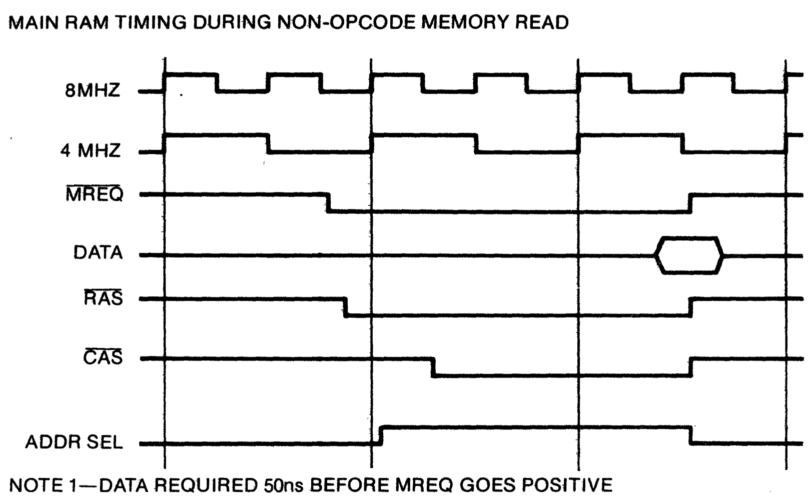


Figure 4-4

4.1.3 Boot Prom

The storage capacity of the Boot Prom is 2K bytes. Contained in the PROM are the Bootstrap routine, the Mini Monitor and the Video Driver.

The Bootstrap routine performs the primary function of the Boot PROM, i.e., to load programs from the disk or from the serial port. Programming information relating to the Bootstrap routine is given in Section 3.12.

The Mini Monitor allows the operator of the ADVANTAGE to perform some elementary commands from the keyboard, such as examining a single location in Main RAM. The operating instructions for the Mini Monitor are in Section 6.4.

The Video Driver controls the position of the cursor and provides a set of standard templates for forming character images on the screen. The Video Driver is described in Section 3.6.5.

4.1.4 Auxiliary Processor and Keyboard

The Auxiliary Processor interfaces the keyboard to the Central Processor, and controls some of the disk drive functions.

A block diagram of the Auxiliary Processor is shown in Figure 4-5. The heart of the Auxiliary Processor is the 8035 microprocessor, which executes the fixed program located in the Auxiliary PROM. The 8035 operates as a slave to the Central Processor. It responds to commands from the Central Processor and responds to data input from the keyboard.

The 8035 maintains a 7-character buffer for storing keyboard characters. It also maintains various status bits associated with the keyboard and debounces the keyboard signals.

Auxiliary Processor Block Diagram

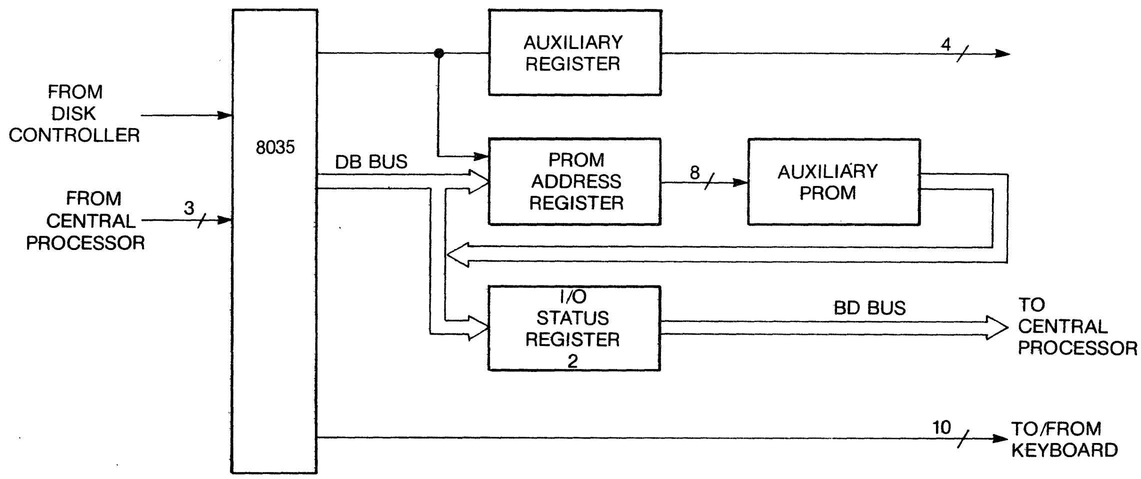


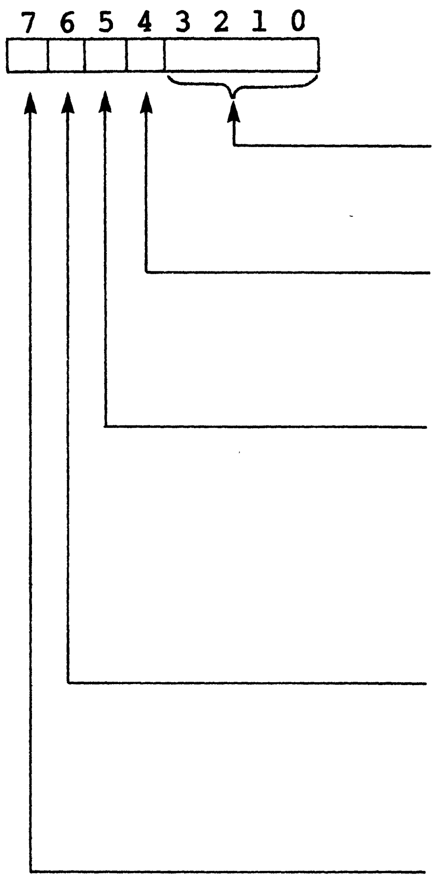
Figure 4-5

The Auxiliary Register stores four control bits which are output by the 8035. Two of them, SPW1 and SPW2, are used by the Disk Controller to determine the width of the sector pulse. The third bit turns the disk drive motors on and off, and the fourth bit causes a maskable interrupt in the Central Processor when keyboard data is available.

I/O Status register 2 stores data and control bits which are loaded by the 8035 and read by the Central Processor. Table 4-6 shows the format of this register.

Table 4-6

I/O Status Register 2 Format



These bits depend on the I/O command in the I/O Control register (see Table 4-5 above).

Auto-Repeat. A one indicates that a key has been held down for longer than 800 milliseconds. The bit goes to a zero when the key is released.

Character Overrun. Set to a one if an attempt is made to put more than seven characters into the keyboard buffer (i.e., a character was received from the keyboard when the buffer was full). The bit is cleared whenever the upper nibble of a character is read (command 2).

Keyboard Data Flag. Set whenever one or more characters is available from the 7-character keyboard buffer. Cleared when the upper nibble of a character is read (command 2).

Command Acknowledge. This bit is complemented each time an I/O Control Code is executed (for codes, see Table 4-5 above).

The 8035 performs the following functions:

1. It monitors the sector pulse signal from the Disk Controller, SPULSE, and sends two signals back to the controller that are used to determine the width of the sector pulse. These signals pass through the Auxiliary Register.

3. It scans the keyboard to determine if any key(s) is pressed. Keyboard scanning proceeds as follows:

   The 8035 outputs a repeating sequence of addresses to the keyboard on signals KBD D0/AD0 through KBD D3/AD3. As each new address is output, it is accompanied by a pulse on the KBD STB signal. If a key is pressed, the keyboard responds by placing the code for the active key onto signals KBD D0/AD0 through KBD D7, immediately after the KBD STB signal expires. The 8035 pauses momentarily to input the code and then proceeds to scan.

   If the entered key is a data key, the 8035 stores the appropriate ASCII code in its 7-character buffer. If the data key is pressed for more than 800 milliseconds, the 8035 also stores a special repeat code in the buffer.

   If the entered key is the CURSOR LOCK or ALL CAPS key, the 8035 interrupts the scan momentarily to change the state of the light in the corresponding key. It does this by pulsing one of four signals (KBD D4 through KBD D7) coincident with the KBD STB signal. These four signals allow for four commands: cursor lock on, cursor lock off, all caps on and all caps off.

4. It executes the command indicated by signals CI0 through CI2. These signals form a 3-bit command code which originates in the Central Processor. The commands are defined in Table 4-5.

When the Central Processor changes the command code the 8035 executes the new command, and acknowledges that the command has been performed by changing the state of the Command Acknowledge bit, bit 7 of I/O Status register 2 (see Table 4-6). The time interval between a change in the Command Code and a change in the Command Acknowledge bit is in the range of 0.5 to 1.5 milliseconds.

4.1.5 Disk Controller

The Disk Controller performs most of the control functions for the disk drives. It selects the drive, selects a side on the diskette, positions the read/write head and performs the read or write operation.

The Auxiliary Processor performs the remaining disk operations, controlling of the disk motors and keeping track of the sector number.

A block diagram of the Disk Controller is shown in Figure 4-6.

The Data Separation Circuitry receives a signal from the selected disk drive which contains both data and clocks. It synchronizes with the clocks, removes the clocks from the signal, and sends the data in serial form to the Control Logic. Three major signals control the Data Separation Circuitry: DISK READ FLAG, ACQUIRE and BUFACQUIRE. The DISK READ FLAG enables the Data Separation Circuitry. The ACQUIRE and BUFACQUIRE signals are set only during the preamble of the sector when there are clock pulses but no data pulses. They allow the phase lock loop in the Data Separation circuitry to quickly synchronize with the clock.

Disk Controller Block Diagram

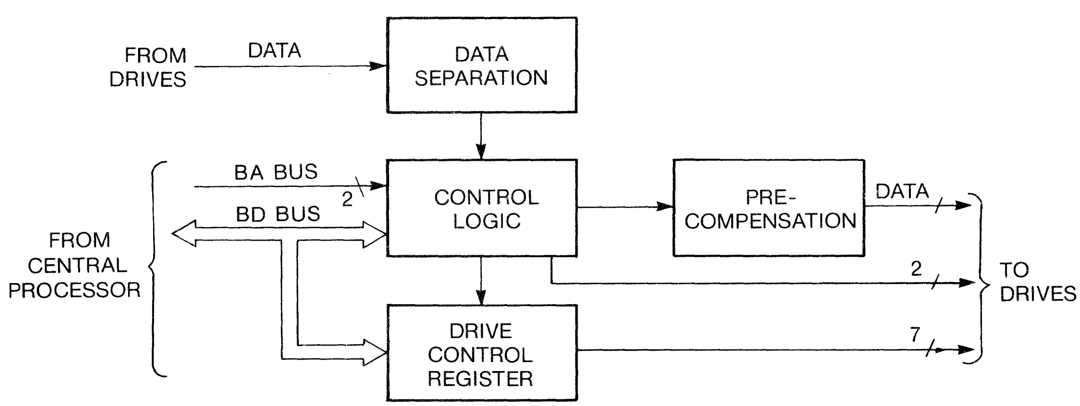


Figure 4-6

The Control Logic responds to the eight I/O instructions listed in Table 4-7. The Control Logic detects these instructions by comparing bits 0 and 1 of the BA bus, and signals $\overline{WR}$, $\overline{RD}$ and $\overline{\text{DISK I/O}}$ from the Central Processor. 8-bit bytes are transferred between the Control Logic and the Central Processor via the BD bus.

Table 4-7

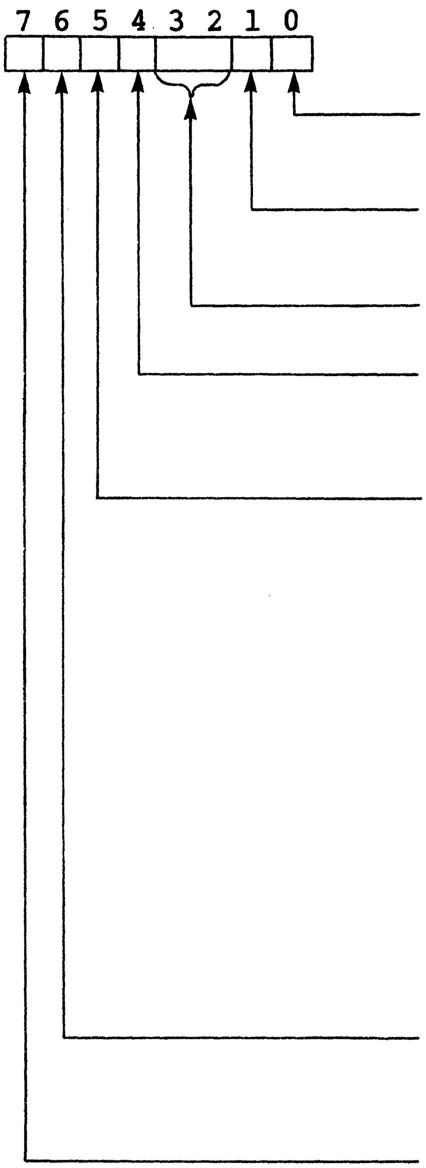
Disk I/O Instructions

I/O Address (Hexadecimal)	Operation	Description
80	INPUT	Input disk data.
80	OUTPUT	Output disk data.
81	INPUT	Input sync byte.
81	OUTPUT	Load drive control register.
82	INPUT	Clear Disk Read flag.
82	OUTPUT	Set Disk Read Flag.
83	INPUT	Produce the standard 'beep' sound. The decoded signal is sent to the Speaker Circuit (see Figure 4-1).
83	OUTPUT	Set Disk Write flag.

The Drive Control Register stores a control byte which comes from the Central Processor and is sent directly to the disk drives. Table 4-8 shows the format of the register.

Table 4-8

Drive control Register Format



<u>Disk Drive 1.</u> A one selects disk drive 1. Bit 1 must be off.

<u>Disk Drive 2.</u> A one selects disk drive 2. Bit 0 must be off.

Reserved for future use.

<u>Step Pulse.</u> Setting this bit on, then resetting it, causes the head to step in the selected disk drive.

<u>Step Direction/Precompensation</u>

A. During stepping of the disk from track, this bit determines the step direction.
   - 0 = Step toward outer track (higher track numbers).
   - 1 = Step toward inner track (lower track numbers)

B. During writing, this bit controls write precompensation.
   - 1 = Use precompensation.
   - 0 = Use no precompensation.

Precompensation is required on the inside 20 tracks - track 15 through 34 on side 0 and tracks 35 through 49 on side 1.

<u>Diskette Side Select.</u>
- 0 = Side 0
- 1 = Side 1

Not used.

The Precompensation circuit changes the timing of the data and clock pulses that are written on the inside tracks of the diskette. The pulse timing must be changed because of the higher density of the data on these tracks.

4.1.6 Display RAM and Video Generator

The Display RAM has a storage capacity of 20K bytes, with 8 bits per byte. This RAM stores the data displayed on the ADVANTAGE video monitor. Section 3.6.1 explains the correlation between the bits in memory and the dots (pixels) on the screen.

The Video Generator serializes the data in the Display RAM and sends this data to the Video Monitor, along with horizontal and vertical sync pulses. It also allows the Central Processor to gain access to the Display RAM, and implements vertical scrolling of the displayed data.

Figure 4-7 shows a block diagram of the Display RAM and Video Generator. All blocks in the diagram are part of the Video Generator except the one marked 'RAM'.

When the Central Processor writes data into the Display RAM, the Address Mux (multiplexer) directs address bits from the BA and ADR buses to the RAM. The data to be written enters the RAM from RD bus.

When the Central Processor reads data from the RAM, the Address MUX again directs the address bits from the BA and ADR buses to the RAM, but the data from the RAM is placed on the BD bus.

The RAM is automatically refreshed as a result of reading video data during generation of the video signal.

Display RAM and Video Generator

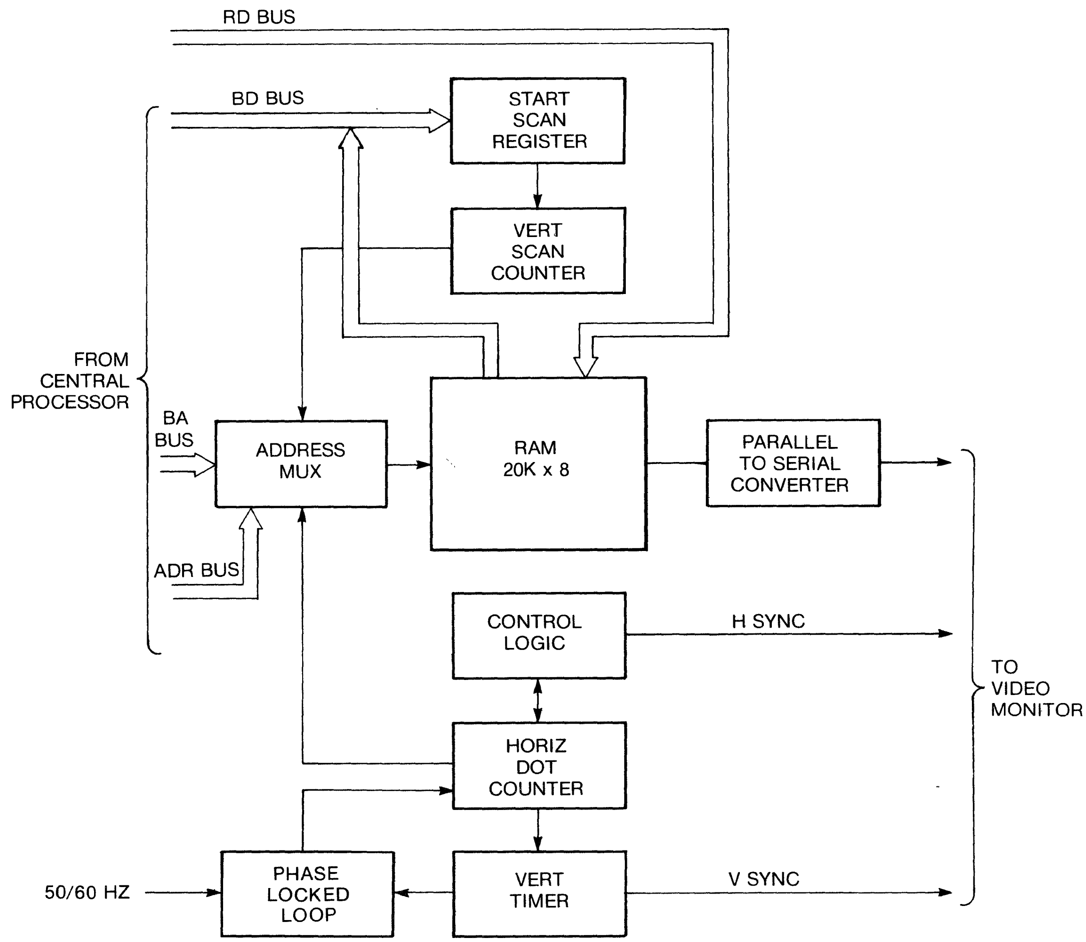


Figure 4-7

When the RAM is supplying data to the Video Monitor, the Address Mux takes RAM address bits from the Vertical Scan Counter and from the Horizontal Dot Counter and sends them to the RAM. These two counters increment as the display screen is scanned so that the correct data is always being sent to the Video Monitor. The RAM data passes through a serial to parallel converter before going to the Video Monitor.

The Start Scan Register controls the vertical position of data on the display screen. When data is output to this register, the data enters the register from the BD bus. At the start of each vertical scan, the number in the Start Scan Register is loaded into the Vertical Scan Counter. This number determines the starting address that is sent to the RAM at the beginning of each vertical scan. The Vertical Scan Counter increments once each horizontal cycle.

The Horizontal Dot Counter increments as the display is scanned in a horizontal direction. It is reset at the beginning of each horizontal scan, and advances once for each dot position. This counter is used in the following ways:

1. It supplies RAM address bits to the Address Mux.

2. It assists the Control Logic in generating certain signals which must repeat in the same way in each horizontal cycle.

3. It provides a clock signal for the Vertical Timing and Control section.

The Control Logic performs the following functions:

1. It controls the Address Mux.

2. It responds to Central Processor request for access to the RAM (signal $\overline{\text{Z80 DIS REQ}}$) and grants the request with signal Z80 CYC.

3. It generates the row address and column address strobes for the RAM (signals $\overline{\text{RASA}}$, $\overline{\text{RASB}}$, $\overline{\text{CASA}}$ and $\overline{\text{CASB}}$).

4. It generates the 'load' signal for the Parallel to Serial Converter.

5. It generates the HORIZ SYNC signal. This signal keeps the Video Monitor horizontal sweep circuits in synchronization with the serial video data.

6. It blanks the display when the Central Processor $\overline{\text{DISP ON}}$ signal is high.

7. It generates a synchronization signal (PS SYNC) for the Ramp Generator in the Voltage Regulator section (see Section 4.1.9).

The Control Logic contains two PROMs, HTIML and HTIMH which are used to generate a repeating pattern of signals. The PROM address is supplied by the Horizontal Dot Counter. The contents of these PROMs is defined in Tables 4-9 and 4-10. Figure 4-8 shows the timing of the signals derived from the PROMs.

Table 4-9

HTIML Horizontal Scan PROM

Address (Hexadecimal)	Output Bits PD3	PD2	PD1	PD0	Description
00	0	0	0	0	ENDIS-Get display data
01	0	0	0	0	Wait
02	0	0	1	1	LDVSR-Load Shift Register
03	0	0	1	0	ENDIS-Get display data
04	0	0	0	0	Wait
05	0	0	0	1	ENZ80-Allow Z80 memory cycle
06	0	0	1	1	LDVSR-Load Shift register
07	0	0	0	0	Wait
08 . . . FF					The above pattern repeated 31 times.

Table 4-10

HTIMH Horizontal Scan PROM

Address (Hexadecimal)	Output Bits PD3	PD2	PD1	PD0	Description
00	0	0	1	0	ENDIS-Get display data
01	0	0	0	0	Wait
02	0	0	1	1	LDVSR-Load Shift register
03	0	0	1	0	ENDIS-Get display data
04	0	0	0	0	Wait
05	0	0	0	1	ENZ80-Allow Z80 memory cycle
06	0	0	1	1	LDVSR-Load Shift register
07	0	0	0	0	Wait
08 . . . 3F					The above pattern repeated 7 times
40	0	0	0	0	Wait
41	0	0	0	0	Wait
42	0	0	0	0	Wait
43	0	0	0	0	Wait
44	0	0	0	0	Wait
45	0	0	0	1	ENZ80-Allow Z80 memory cycle
46	0	0	0	0	Wait
47	0	0	0	0	Wait
48	0	1	0	0	HZSYNC-Horizontal Sync time
49	0	1	0	0	HZSYNC
4A	0	1	0	0	HZSYNC
4B	0	1	0	0	HZSYNC
4C	0	1	0	0	HZSYNC
4D	0	1	0	1	HZSYNC and ENZ80
4E	0	1	0	0	HZSYNC
4F	0	1	0	0	HZSYNC
50 . . . 97					The above pattern in addresses 48 through 4F repeated 9 times

Table 4-10 (continued)

98	0	0	0	0	Wait
99	0	0	0	0	Wait
9A	0	0	0	0	Wait
9B	0	0	0	0	Wait
9C	0	0	0	0	Wait
9D	0	0	0	1	ENZ80-Allow Z80 memory cycle
9E	0	0	0	0	Wait
9F	0	0	0	0	Wait
A0	0	0	0	0	Wait
A1	0	0	0	0	Wait
A2	0	0	0	0	Wait
A3	0	0	0	0	Wait
A4	0	0	0	0	Wait
A5	0	0	0	1	ENZ80-Allow Z80 memory cycle
A6	0	0	0	0	Wait
A7	0	0	0	0	Wait
A8	0	0	0	0	Wait
A9	0	0	0	0	Wait
AA	0	0	0	0	Wait
AB	0	0	0	0	Wait
AC	0	0	0	0	Wait
AD	0	0	0	0	Wait
AE	0	0	0	0	Wait
AF	1	0	0	0	Clear Horizontal Column Counter
B0	1	1	1	1	
B1 . . . FF					The above pattern repeated 79 times

Horizontal Scan Timing

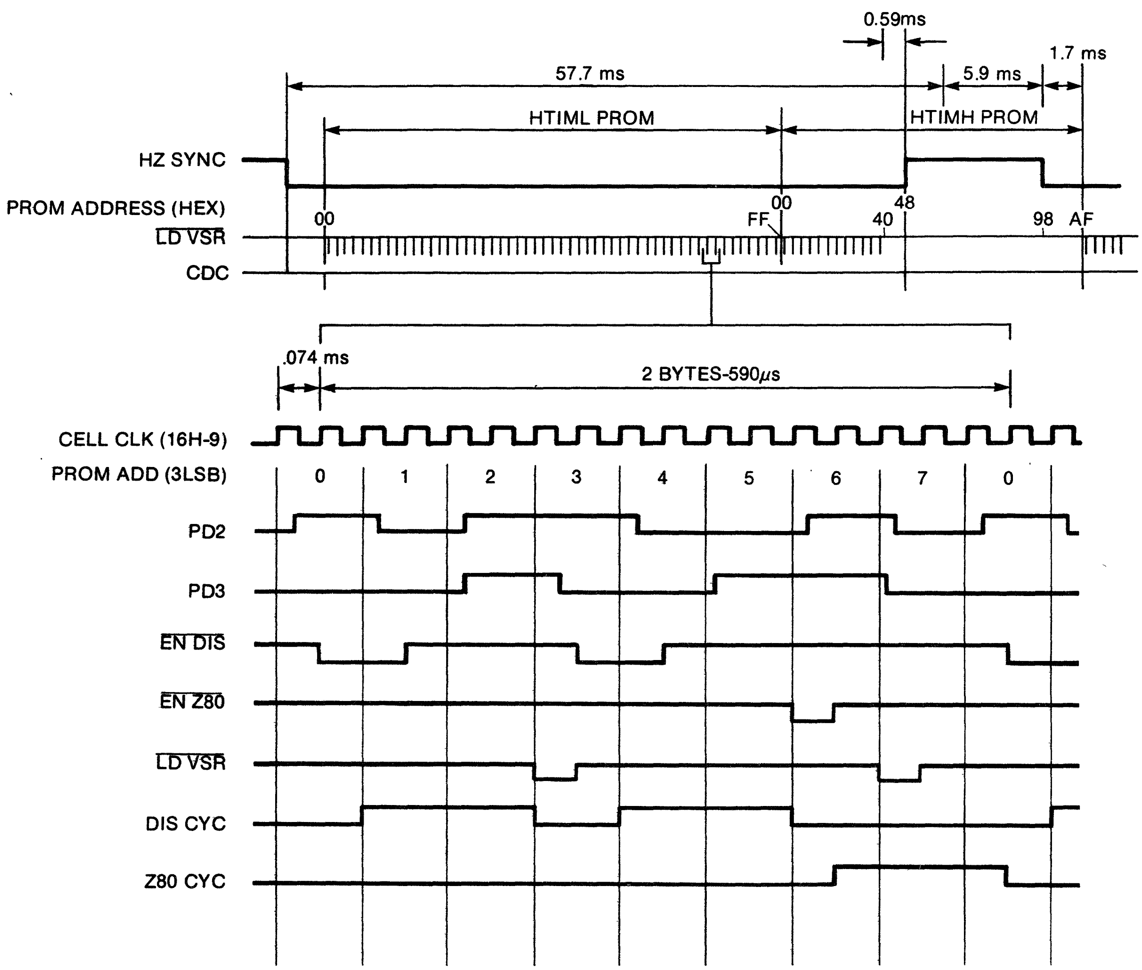


Figure 4-8

The Vertical Timer performs the following functions:

1. It generates the vertical sync (VSYNC) signal. This signal keeps the Video Monitor vertical sweep circuits in synchronization with the serial video data.

2. It generates the vertical blanking (VBL) signal. This signal causes the serial video data to be all zeros during vertical retrace.

3. It generates a synchronization signal, PL SYNC, that is used by the Phase Locked Loop and by the Control Processor. In the Central Processor it sets the Display flag.

4. It loads the contents of the Start Scan Register into the Vertical Scan Counter at the beginning of each vertical scan.

The repetitive control signals required to perform these four functions are generated by means of one of two PROMs: VTM60 or VTM50. The first of these PROMs is used when the power line frequency is 60Hz and the second PROM is used when the power line frequency is 50Hz. Table 4-11 and 4-12 define the contents of the PROMs. Figure 4-9 shows the timing of the generated signals.

The Phase Locked Loop keeps the Video Generator in synchronization with the power line frequency. It compares signal PL SYNC from the Vertical Timer with the power frequency, and generates an output signal, CELL CLK, which varies in frequency according to the phase of the two compared signals. CELL CLK drives the Horizontal Dot Counter which in turn drives the Vertical Timer, establishing the feedback loop.

Table 4-11

60Hz Vertical Timing PROM (VTIM60)

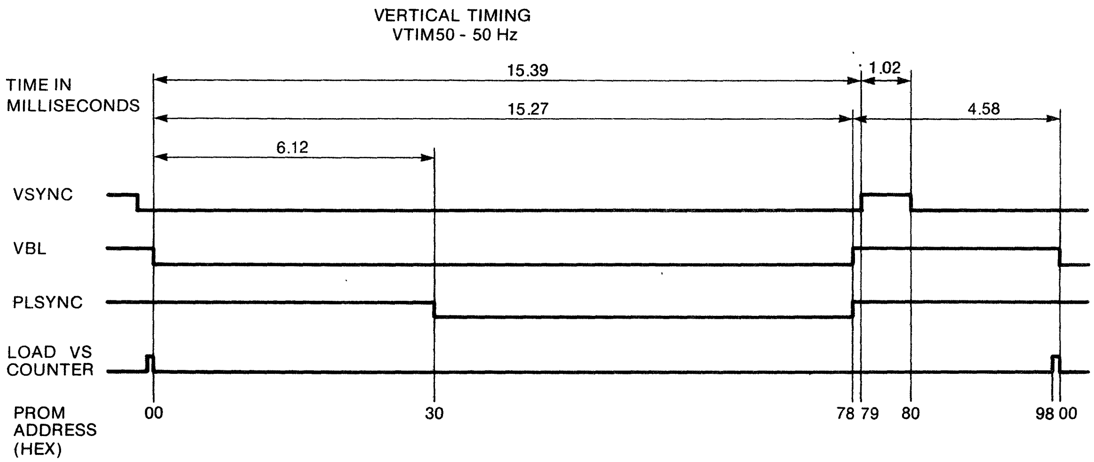
Address (Hexadecimal)	Output Bits VD3	VD2	VD1	VD0	Description
00	0	0	1	0	
.	0	0	1	0	
.	0	0	1	0	PLSYNC on for 100 lines
.	0	0	1	0	
32	0	0	1	0	
33	0	0	0	0	
.	0	0	0	0	
.	0	0	0	0	Wait for 139 lines
.	0	0	0	0	
77	0	0	0	0	
78	1	0	1	0	VBL + PLSYNC
79	1	1	1	0	VBL + PLSYNC + VSYNC
7A	1	1	1	0	VBL + PLSYNC + VSYNC
7B	1	1	1	0	VBL + PLSYNC + VSYNC
7C	1	1	1	0	VBL + PLSYNC + VSYNC
7D	1	1	1	0	VBL + PLSYNC + VSYNC
7E	1	1	1	0	VBL + PLSYNC + VSYNC
7F	1	1	1	0	VBL + PLSYNC + VSYNC
80	1	0	1	0	VBL + PLSYNC
81	1	0	1	0	VBL + PLSYNC
82	1	0	1	1	VBL + PLSYNC + Load Vertical Scan Counter
83	1	1	1	1	
84					
.					
.					The above pattern repeated 124 times
.					
FF					

Table 4-12

50Hz Vertical Timing PROM (VTIM50)

Address (Hexadecimal)	Output Bits VD3	VD2	VD1	VD0	Description
00	0	0	1	0	
.	0	0	1	0	
.	0	0	1	0	PLSYNC on for 96 lines
.	0	0	1	0	
29	0	0	1	0	
2A	0	0	0	0	
.	0	0	0	0	
.	0	0	0	0	Wait for 156 lines
.	0	0	0	0	
77	0	0	0	0	
78	1	0	1	0	VBL + PLSYNC
79	1	1	1	0	VBL + PLSYNC + VSYNC
7A	1	1	1	0	VBL + PLSYNC + VSYNC
7B	1	1	1	0	VBL + PLSYNC + VSYNC
7C	1	1	1	0	VBL + PLSYNC + VSYNC
7E	1	1	1	0	VBL + PLSYNC + VSYNC
7F	1	1	1	0	VBL + PLSYNC + VSYNC
80	1	0	1	0	VBL + PLSYNC + VSYNC
.	1	0	1	0	VBL + PLSYNC + VSYNC
.	1	0	1	0	VBL + PLSYNC + VSYNC
.	1	0	1	0	VBL + PLSYNC + VSYNC
9A	1	0	1	0	VBL + PLSYNC + VSYNC
9B	1	0	1	1	VBL + PLSYNC + Load Vertical Scan Counter
9C	1	1	1	1	
9D					
.					
.					The above pattern repeated 99 times
.					
FF					

Vertical Scan Timing



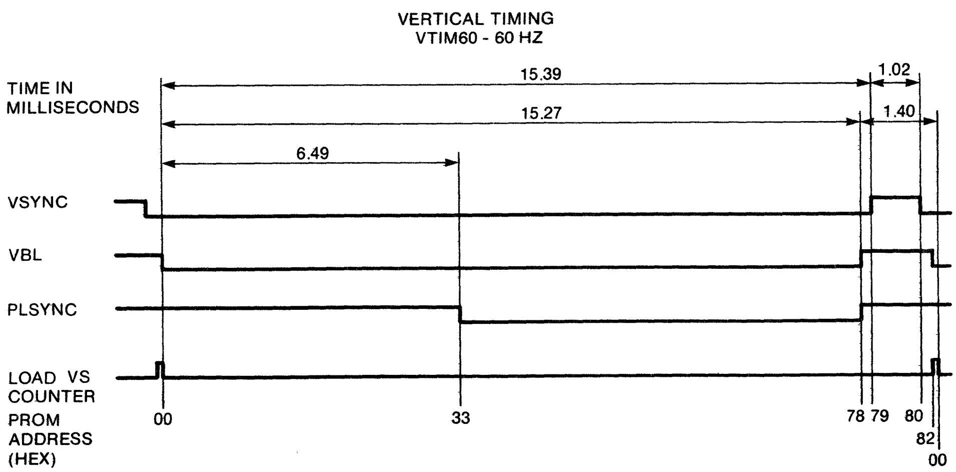


Figure 4-9

4.1.7 I/O Board Interface

The I/O Board Interface consists of six PC board connectors and associated bus drivers and decoders. The I/O boards inserted in these connectors respond to I/O instructions from the Central Processor. The boards may communicate only with the Central Processor, or they may interface the Central Processor to an external device.

Figure 4-10 is a block diagram of the I/O Board Interface.

I/O Board Interface Block Diagram

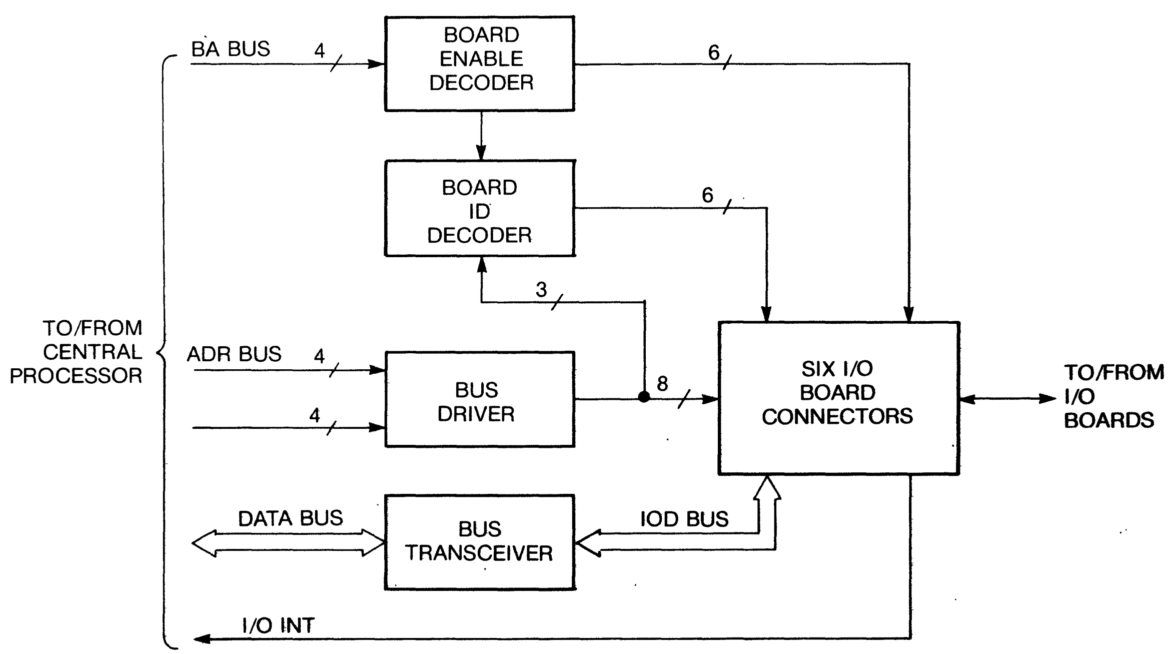


Figure 4-10

The Board Enable Decoder decodes the upper four bits of the I/O address, taken from the BA bus. It provides each of the board connectors with an enable signal (ENA I/O 1 through ENA I/O 6). Each board must complete the decoding of the I/O address and the recognition of I/O instructions by comparing signals sent to it from the Bus Driver.

The Board ID Decoder responds to I/O instructions with an I/O address of 70 through 75 and 78 through 7D. These instructions input the identification code of the board in a particular board connector. The decoder provides one ID REQ signal for each connector. The ID code returns to the Central Processor via the IOD and DATA buses.

The Bus Driver continually transfers the lower four bits of the address bus and four control and timings signals from the Central Processor to all board connectors. The I/O boards use these signals, in conjunction with those sent from the Board Enable Decoder and the Board ID Decoder to complete the recognition of specific I/O instructions.

The Bus transceiver transfers 8-bit bytes of data between the Central Processor and the I/O Boards. The Central Processor controls the direction of data flow.

The I/O Boards use the $\overline{\text{I/O INT}}$ signal to send interrupt requests to the Central Processor.

The signals on the six I/O Board connectors are defined in Table 4-13. All signals are common to all connectors, except the signals on pin 3 and pin 29. These are the individual 'board select' signals from the Board Enable Decoder and the Board ID Decoder.

Table 4-13

I/O Board Pin Assignments

Pin	Signal Name	Signal Direction	Function
1	Ground		Power/signal ground
2			Not used.
3	$\overline{\text{ID REQ}}$	OUTPUT	Input board identification code
4	+5V	OUTPUT	DC power
5	+12V	OUTPUT	DC power
6			Not used
7	$\overline{\text{IO INT}}$	INPUT	Maskable interrupt request
8			Not used
9	IOA2	OUTPUT	Buffered Address bus, bit 2
10	IOA1	OUTPUT	Buffered Address bus, bit 3
11	IOA1	OUTPUT	Buffered Address bus, bit 1
12	Ground		Power/signal ground
13	$\overline{\text{BRD}}$	OUTPUT	Buffered Z80 processor $\overline{\text{RD}}$ signal
14	IOA0	OUTPUT	Buffered Address bus, bit 0
15	IO8MHz	OUTPUT	8 MHz clock
16	$\overline{\text{BWR}}$	OUTPUT	Buffered Z80 processor $\overline{\text{WR}}$ signal
17	IOD3	BIDIREC-TIONAL	I/O Data bus, bit 3
18	$\overline{\text{BIORES}}$	OUTPUT	Resets I/O boards
19	IOD2	BIDIREC-TIONAL	I/O Data bus, bit 2
20	IOD4	BIDIREC-TIONAL	I/O Data bus, bit 4

Table 4-13 (continued)

Pin	Signal Name	Signal Direction	Function
21	Ground		Power/signal ground
22	IOD5	BIDIREC-TIONAL	I/O Data bus, bit 5
23	IOD6	BIDIREC-TIONAL	I/O Data bus, bit 6
24	IOD1	BIDIREC-TIONAL	I/O Data bus, bit 1
25	IOD0	BIDIREC-TIONAL	I/O Data bus, bit 0
26	-12V	OUTPUT	DC power
27	+5V	OUTPUT	DC power
28	IOD7	BIDIREC-TIONAL	I/O Data bus, bit 7
29	$\overline{\text{ENA I/O}}$	OUTPUT	Selects board for I/O operation
30	Ground		Power/signal ground

Figure 4-11 shows the timing of the I/O Board signals. Both read and write cases are shown, although the WR and DATA OUT signals would only be active during an output instruction and the RD and DATA IN signals would only be active during an input instruction.

I/O Board Timing

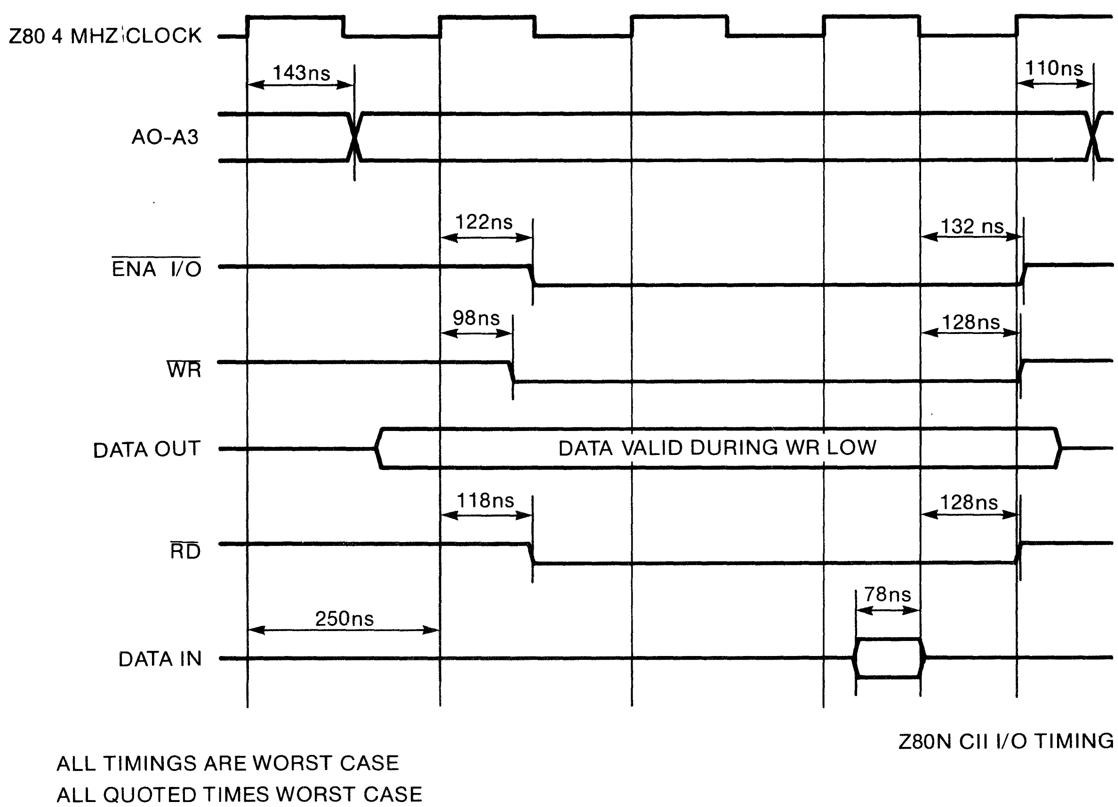


Figure 4-11

4.1.8 Speaker Circuit

The speaker is a small transducer located on the Main PC Board.

The speaker circuit produces two kinds of sounds in the speaker: a standard 'beep' sound with a fixed pitch and duration, and a programmable sound which can be varied in pitch and duration.

The standard beep sound is triggered when signal TRIG BEEP pulses low. This fires a one-shot which allows a free-running oscillator to produce a 1920 Hz tone for one-half second.

The programmable sound is generated from signal SPK DATA which represents bit 6 of the I/O Control Register. To produce the sound, the program turns the bit on and off at an audible rate. The program can produce any desired tone and maintain it for any length of time.

4.1.9 Voltage Regulators

There are five DC voltage regulators on the Main PC Board that provide regulated DC power for the ADVANTAGE system. These regulators are shown in Figure 4-12, along with their associated circuits.

The +12V and Main +5V regulators receive power from the unregulated +23V supplied to the Main PC Board. These regulators are of the switching type and use transistors and op amps as active elements.

The Ramp Generator produces a pulse which synchronizes the +12V and +5V regulators so that they switch on during the horizontal retrace of the Video Monitor. The pulse is triggered by the positive going edge of the PS SYNC signal.

The Ramp Generator receives a signal from the Video Generator (PS SYNC) and from it develops a pulse that synchronizes the +12V and +5V regulators. The regulators are triggered to switch on during the horizontal retrace of the Video Monitor in order to minimize the effect of switching noise on the display screen.

Voltage Regulators Block Diagram

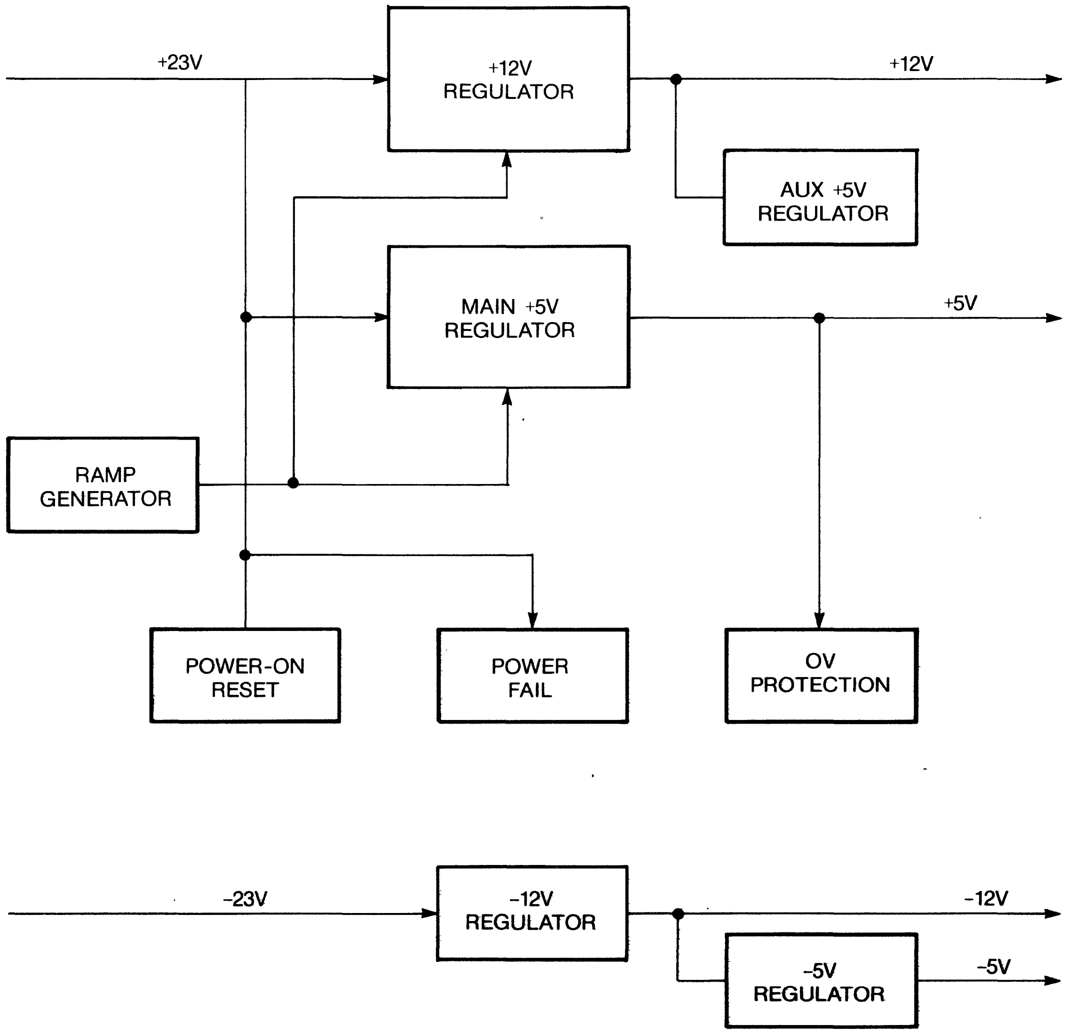


Figure 4-12

The Auxiliary +5V regulator is an integrated circuit linear regulator and is used only by the Video Phase Locked Loop circuit.

The Power-on Reset circuit and the Power Fail circuit both monitor the unregulated +23V input to the Main PC Board. The Power-on Reset circuit produces the $\overline{\text{PWR RES}}$ signal when power is first turned on. This signal resets the ADVANTAGE system. The Power Fail circuit produces the $\overline{\text{PWR FAIL}}$ signal if the +23V power is interrupted.

The Over-Voltage (OV) Protection circuit monitors the output of the Main +5V regulator. It pulls the +5V line to ground and blows the Main PC Board fuse if +5V rises above +7.8V.

The -12V regulator receives power from the unregulated -23V supplied to the Main PC Board. The -5V regulator receives power from the -12V regulator. Both of these are integrated circuit linear regulators.

4.2 SIO BOARD

The SIO (Serial Input/Output) Board interfaces the Main PC Board with serial printers and communication links. The board's serial interface can be configured to support the RS232 standard or current loop operation. A block diagram of the SIO Board is shown in Figure 4-13.

The heart of the SIO board is the 8251 USART. Refer to the manufacturer's data sheet in Appendix H for information concerning this integrated circuit.

The Main Board Interface responds to I/O instructions from the Main PC Board. All but one of these instruction are listed in Table 4-14. The unlisted instruction is directed to the I/O board connector rather than the SIO board. It requests that the board in that connector place its board ID code on the IOD bus. When this instruction is active, the ID REQ signal goes low. The ID code for the SIO Board is F7H.

The Interrupt Mask is a 4-bit register contained in the Main Board Interface. It determines the conditions under which a maskable interrupt is sent to the Main PC Board. Each bit of the register is associated with an output bit of the USART. When the mask bit is a one and the associated USART signal is true, the interrupt is generated. Figure 4-15 shows the format of the register.

SIO Board Block Diagram

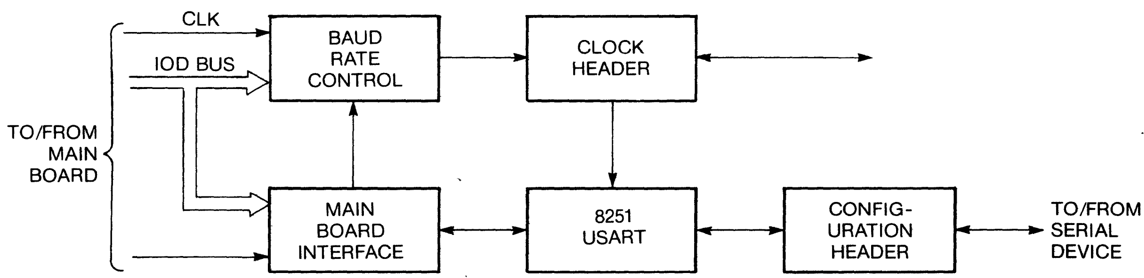


Figure 4-13

Table 4-14

SIO Board I/O Instructions

I/O Address (Hexadecimal)	Operation	Description
X0	INPUT	Transfer a data byte from the USART to the Main PC Board.
X0	OUTPUT	Transfer a data byte from the Main PC Board to the USART.
X1	INPUT	Transfer a status byte from the USART to the Main PC Board.
X1	OUTPUT	Transfer a control byte from the Main PC Board to the USART.
X8 or X9	OUTPUT	Load the Baud Rate register.
XA or XB	OUTPUT	Load the Interrupt Mask register.
NOTE: The first digit of these I/O addresses selects one of the six I/O board connectors. If the connector is enabled, signal $\overline{\text{ENA IO}}$ is low.		

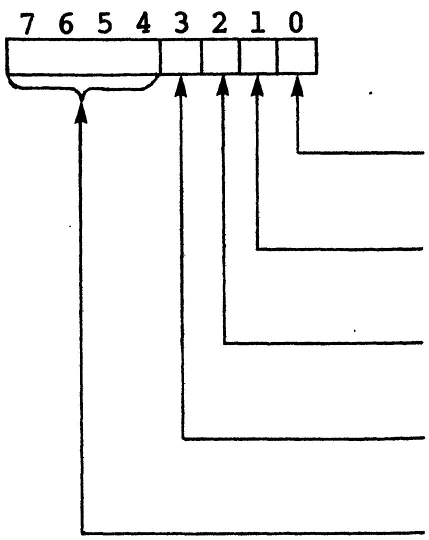
The Baud Rate Control section provides two clocks for the USART: the USART clock and the baud clock.

The USART clock is the fixed frequency basic clock signal for the USART. It is produced by dividing the Main PC Board 8MHz clock signal by 4.33.

The baud clock is used by the USART to determine its transmitting and receiving frequency. The baud clock is generated by a combination of the Baud Rate register and a 9-bit counter. The Baud Rate register provides the pre-load value for the low order 8 bits of the counter. The counter clock is developed by dividing the Main PC Board 8MHz clock signal by 13.

Table 4-15

SIO Interrupt Mask Format



- Tx Empty. Transmitter has finished sending characters.
- Tx Ready. Buffer is ready to receive the next character.
- Rx Ready. Receiver has a character ready.
- Syn Detect. Synchronization has been achieved (synchronous mode only).
- Unused.

The Clock Header is an 8-pin jumper plug which mates with an 8-pin IC socket on the SIO board. This header is used only for synchronous operation. It allows the receive and transmit clocks to be rerouted so the receive clock originates from the serial device (connector J1) and the transmit clock is supplied to that device.

The Configuration Header is a 16-pin jumper plug which mates with a 16-pin IC socket on the SIO board. This header allows the interface signals between the USART and the serial device to be wired so as to conform to the requirements of the device.

4.3 PIO BOARD

The PIO (Parallel Input Output) Board interfaces the Main PC Board with devices that input or output data in 8-bit parallel form.

A block diagram of the PIO Board is shown in Figure 4-14.

The Control Logic contains a programmable configuration header which allows the PIO Board to adapt to many different I/O devices. The configuration header is a 16-pin jumper plug which mates with a 16-pin IC connector on the PIO Board. The header determines the routing of critical control signals in the Control Logic.

This discussion is based on a PIO Board with a standard configuration header, i.e, one that is wired as shown in Figure 4-15. For other possible configurations, consult the schematic drawings in Appendix I.

The Control Logic responds to I/O instructions from the Main PC Board. All but one of these instruction are listed in Table 4-16. The unlisted instruction is directed to the I/O board connector rather than the PIO board. It requests that the board in that connector place its board ID code on the IOD bus. When this instruction is active, the $\overline{\text{ID REQ}}$ signal goes low. The ID code for the PIO Board is DBH.

PIO Board Block Diagram

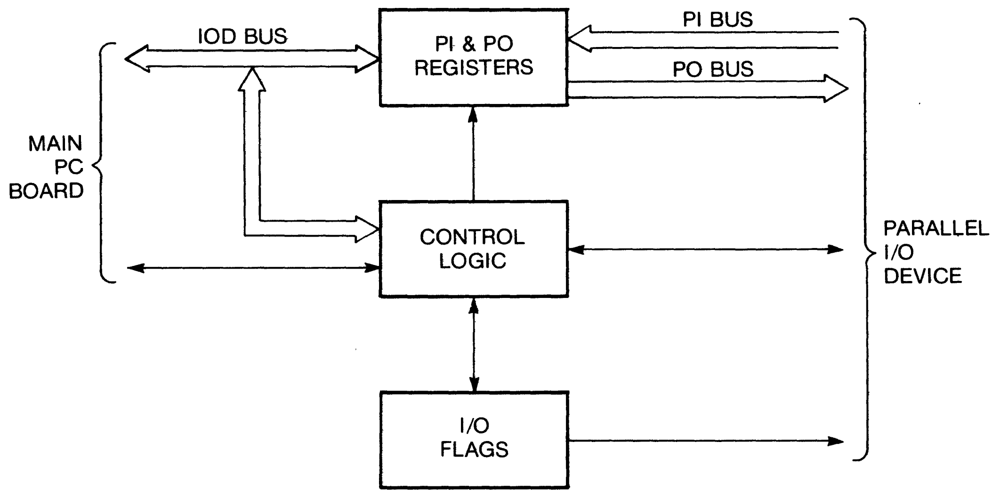


Figure 4-14

The I/O flags are used by the Main PC Board and the I/O device to signal the availability of data. The I/O device sets the Input flag when an input byte has been place on the PI bus. The Main PC Board resets this flag when it inputs the data. Similarly, the Main PC Board resets the Output flag when an output byte has been placed on the PO bus. The I/O device sets the Output flag when it accepts the data.

Standard PIO Configuration Header

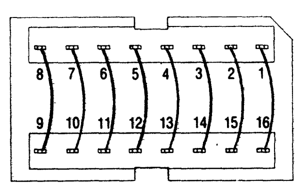


Figure 4-15

Table 4-16

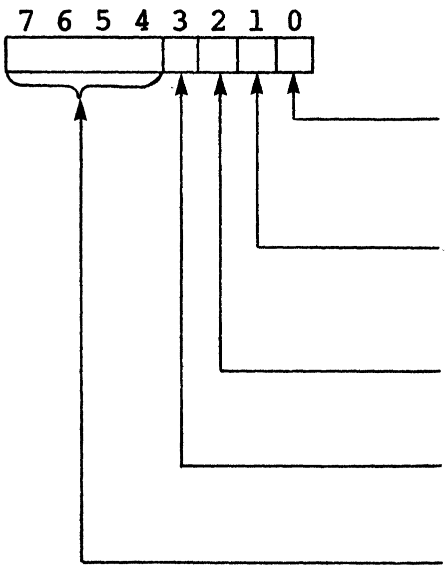
PIO Board I/O Instructions

I/O Address (Hexadecimal)	Operation	Description
X0 or X8	INPUT	Input a data byte from the I/O device to the Main PC Board via the PI register.
X0 or X8	OUTPUT	Output a data byte from the Main PC Board to the I/O device via the PO register.
X1 or X9	INPUT	Input a status byte from the Control Logic. The format of the Status Byte is shown in Table 4-16.
X2 or XA	OUTPUT	Load the Interrupt Mask register in the Control Logic. The format of this register is shown in Table 4-17.
X3 or XB		Not used.
X4 or XC	INPUT/ OUTPUT	Reset the Output flag.
X5 or XD	INPUT/ OUTPUT	Set the Output flag.
X6 or XE	INPUT/ OUTPUT	Reset the Input flag.
X7 or XF	INPUT/ OUTPUT	Set the Input flag.

NOTE: The first digit of these I/O addresses selects one of the six I/O board connectors. If the connector is enabled, signal $\overline{\text{ENA I/O}}$ is low.

Table 4-17

PIO Status Byte Format



Output Acknowledge. A one indicates that the device interface signal OUT ACK is at a low logic level.

Input Acknowledge. A one indicates that the device interface signal IN ACK is at a low logic level.

Output Flag. A one if the output flag is set.

Input Flag. A one if the input flag is set.

All ones.

Table 4-18

PIO Interrupt Mask Format

Bit	Description
3 2 1 0	Ignored.
4	<u>Output Acknowledge.</u> The device interface signal OUT ACK is at a low logic level.
5	<u>Input Strobe.</u> The device interface signal IN STROBE is at a low logic level.
6	<u>Output Flag.</u> The Output flag is set.
7	<u>Input Flag.</u> The Input flag is set.

NOTE: A one in any of the bit positions 4 through 7 allows the stated condition to generate a maskable interrupt.

The ADVANTAGE requires only minimal preventive maintenance for long-term operation. Suggested preventive maintenance procedures are listed in Table 5-1. Instructions for opening the ADVANTAGE cabinet can be found in Section 6.5.2.

WARNING
ALWAYS UNPLUG THE POWER CORD FROM THE BACK OF THE UNIT BEFORE OPENING THE CABINET.

Table 5-1

Preventive Maintenance Schedule

Activity	Schedule *	Comments
Clean exterior of cabinet	As needed	Dust with a soft cloth. Clean CRT screen with glass cleaner. For persistent dirt use a damp sponge or towel. Do not allow cleaning water to drip down into unit.
Examine/replace diskettes	Weekly or after approximately 150 hours of use.	Examine diskettes for excessive wear. A new disk has a smooth surface of magnetic film. As it wears, concentric lines appear on the magnetic surface. Replace any diskette that appears worn by copying the data to a new diskette and discarding the old one. The standard life of a diskette varies by brand, handling, and other usage factors.

Table 5-1 (continued)

Clean printed circuit boards	During drive servicing or as needed	Clean printed circuit boards with compressed air or similar means.
Check internal connections	During drive servicing	Visually check all interior wires and connectors, making sure connectors are properly seated.
Run Diagnostic programs	Monthly	The Diagnostic programs may detect the beginning of a maintenance problem before it has become evident to the operator. Diagnostic programs are described in Section 6.2.
* Schedule more often if unit is being used in a dirty environment.		

6.1 LOCATING THE CAUSE OF FAILURE

This chapter describes how to locate and replace a failed part at the subassembly level. If the system is operating minimally, the extensive diagnostic programs can be used to test the machine. These programs are described in Section 6.2. If the failure is serious enough to prevent diagnostic programs from being loaded, the troubleshooting procedure in Section 6.3 or the Mini Monitor in Section 6.4 can be used. When the failed subassembly is located, it may be replaced by using the procedures provided in Section 6.5.

6.2 THE DIAGNOSTIC PROGRAMS

The ADVANTAGE diagnostic programs provide comprehensive testing of the ADVANTAGE system. They are loaded from the Dealer Diagnostics Diskette and may be run at three different levels:

1. Integrity Test. Automatically performs low level testing when cold starting the system from any ADVANTAGE System Diskette.

2. Default Mode. User-level diagnostic. More extensive than the Integrity Test, but requires only minimal operator interaction.

3. Single Block Mode. Most detailed diagnostic level level. Provides individual tests of ADVANTAGE subassemblies.

The remainder of this section describes how to run the diagnostics in Single Block mode. Note that the diagnostic programs are self-prompting; this description is included for general reference.

6.2.1 Single Block Mode

1. Load the Dealer Diagnostic diskette as described in Section 2.2. The screen will display "North Star Test System - Option Menu" with a version number ending in 'B' and the following menu:

   [1] Run the Default test
   [2] Go into Single Block mode

2. Press the '2' key to enter Single Block mode. Data is read from the diskette and the screen changes to the format shown in Figure 6-1.

3. To select one of the tests, press the corresponding key (1 through 6). A diagnostic monitor loads the selected test. Control is returned to the monitor when the test is completed.

```
            Single Block Mode - Display Format


         North Star Test System - Ver. 1.0-A
            SINGLE BLOCK MENU

   Please make your choice from the following:

         [1] Disk Subsystem Test
         [2] Executable Memory Test
         [3] Video Memory Test
         [4] SIO Board Test
         [5] Keyboard Test
         [6] Display Monitor Test


   Input your desired choice:

Ctl-C to exit            (c) NorthStar Computers, inc. 1981
```

Figure 6-1.

6.2.2 Disk Subsystem Test

The Disk Subsystem Test requires two 'scratch' diskettes, one for each of the two disk drives. They must be in very good condition to ensure the validity of the test. They may be formatted, although this is not required.

CAUTION
This test destroys any data that was previously stored on the scratch diskettes.

The diskettes are inserted according to machine prompt. When the test is started, the screen will display a format similar to that shown in Figure 6-2. The test begins immediately and runs continuously on both drives, incrementing the pass number, track number, etc. and indicating any errors.

Three passes represent a complete test. To terminate the test, press CONTROL-C and the display returns to the Single Block Menu.

Disk Subsystem Test - Display Format

```
              NORTH STAR TEST SYSTEM - VER. 1.0-B

   MODE: Single          BLOCK: Disk          SECT: Verify
         Block                  Subsystem
         Continuous

                              PASS:    1
                             DRIVE:    2
                             TRACK:   12
                              SIDE:    1
                          FUNCTION: READ
                           PATTERN:  95H

   ============================ ERRORS ==================

 Unit 1: CRC: 0  Verify Comp: 0  Index Pulse: 0 Wrt Prot:0
        Seek: 0    Sync Byte: 0         Read: 0 Status: Passing

 Unit 2: CRC: 0  Verify Comp: 0  Index Pulse: 0  Wrt Prot: 0
        Seek: 0    Sync Byte: 0         Read: 0  Status: Passing

 Ctl-C to exit              (c) NorthStar Computers Inc. 1981
```

Figure 6-2.

6.2.3 Executable Memory Test

This test exercises the Executable Memory (Main RAM) by writing various test patterns, reading them back and checking for discrepancies. This not only tests for failures of individual bits, but checks for cross-talk between the address bits and the data bits. In addition, the memory is tested for its ability to contain a running program.

The test is composed of six sections. The first five sections are identical, except that each of these uses a different test pattern. The sixth section verifies that instructions can be executed from the portion of memory under test.

When the Executable Memory Test is loaded, the screen displays a format similar to Figure 6-3. The pattern of Ms and *s in the center of the screen represents the total area of Main RAM (64K). Each vertical column of single characters represents a 1K portion of the memory, starting with the lowest portion on the left (physical address 0000H) and going in ascending order to the highest portion on the right (physical address 0FFFH). The horizontal rows of characters each represent a different bit in the memory, starting with bit zero on the top, and going down in order to bit 7 on the bottom. The portions of memory marked by Ms are tested by the program. The portions marked by *s are not tested, as they are needed to store the GDOS program, the test program, and various parameter fields.

As the test is running, the display indicates which section of the test is currently being executed. During section 6, a row of characters is displayed across the bottom half of the screen, one character at a time, from left to right. Each new character marks the current 1K portion of memory that is being tested. Each time section 6 is completed, the pass counter is incremented. The counter advances thusly: AA, AB, AC, etc.

Executable Memory Test - Display Format

```
                NORTH STAR TEST SYSTEM - VER. 1.0 - B

          MODE: Single        BLOCK: Executable       SECT: 4
                Block                Memory
                Continuous           PASS: AA

                                 RAM MATRIX
MMMMMMMM MMMMMM*M MMMMMMMM MMMMMMMM MMMMMMMM MMMMMMMM M*M*MMMM ********
MMMMMMMM MMMMMM*M MMMMMMMM MMMMMMMM MMMMMMMM MMMMMMMM M*M*MMMM ********
MMMMMMMM MMMMMM*M MMMMMMMM MMMMMMMM MMMMMMMM MMMMMMMM M*M*MMMM ********
MMMMMMMM MMMMMM*M MMMMMMMM MMMMMMMM MMMMMMMM MMMMMMMM M*M*MMMM ********
MMMMMMMM MMMMMM*M MMMMMMMM MMMMMMMM MMMMMMMM MMMMMMMM M*M*MMMM ********
MMMMMMMM MMMMMM*M MMMMMMMM MMMMMMMM MMMMMMMM MMMMMMMM M*M*MMMM ********
MMMMMMMM MMMMMM*M MMMMMMMM MMMMMMMM MMMMMMMM MMMMMMMM M*M*MMMM ********
MMMMMMMM MMMMMM*M MMMMMMMM MMMMMMMM MMMMMMMM MMMMMMMM M*M*MMMM ********



CTL-C to exit                          (c) NorthStar Computers Inc. 1981
```

Figure 6-3.

If a failure occurs, and the program does not jump sequence, a question mark (?) replaces one of the Ms in the displayed RAM MATRIX, and a exclamation mark (!) is displayed next to the pass count.

Figure 6-4 can be used to find the PC board location of the failing RAM chip. First, note the position of the (?) in the display. Then, read the coordinates of that position from the figure. These coordinates indicate the PC board location of the RAM chip. Figure 6-4 indicates a bad chip at board location J7.

The Executable Memory Test runs continuously, completing a pass approximately every four minutes. To exit from the test and return to the diagnostic monitor, press CONTROL-C.

Locating a defective Main RAM Chip

```
                                                                    PC BOARD
                                                                   COORDINATES
        G                    H                      J                  F
MMMMMMMM MMMMMM*M MMMMMMMM MMMMMMMM MMMMMMMM MMMMMMMM M*M*MMMM ******** 9
MMMMMMMM MMMMMM*M MMMMMMMM MMMMMMMM MMMMMMMM MMMMMMMM M*M*MMMM ******** 8
MMMMMMMM MMMMMM*M MMMMMMMM MMMMMMMM MMMMMMMM MM?MMMMM M*M*MMMM ******** 7
MMMMMMMM MMMMMM*M MMMMMMMM MMMMMMMM MMMMMMMM MMMMMMMM M*M*MMMM ******** 6
MMMMMMMM MMMMMM*M MMMMMMMM MMMMMMMM MMMMMMMM MMMMMMMM M*M*MMMM ******** 5
MMMMMMMM MMMMMM*M MMMMMMMM MMMMMMMM MMMMMMMM MMMMMMMM M*M*MMMM ******** 4
MMMMMMMM MMMMMM*M MMMMMMMM MMMMMMMM MMMMMMMM MMMMMMMM M*M*MMMM ******** 3
MMMMMMMM MMMMMM*M MMMMMMMM MMMMMMMM MMMMMMMM MMMMMMMM M*M*MMMM ******** 2
```

Figure 6-4.

6.2.4 Video Memory Test

The Video Memory Test exercises the portion of memory (Video RAM) that provides data for the video screen. It operates as described above for the Executable Memory Test, with the following exceptions:

1. There is no section 6 (Instruction Fetch Test), since instructions are never fetched from the Video RAM.
2. The test patterns used to exercise the memory are displayed on the screen. They move across the screen from left to right as the testing proceeds.
3. The 'RAM MATRIX" displayed in the center of the screen is 20 columns wide instead of 64.

If an error occurs, a question mark replaces one of the Ms in the displayed RAM MATRIX. Figure 6-5 may be used to locate the defective RAM chip. The figure indicates a bad chip at board location 5K.

Locating a Defective Video RAM Chip

```
        K                 L        PC BOARD
                                   COORDINATES
MMMMMMMM MMMMMMMM MMMM   9
MMMMMMMM MMMMMMMM MMMM   8
MMMMMMMM MMMMMMMM MMMM   7
MMMMMMMM MMMMMMMM MMMM   6
MMMMMMMM MMMMM?MM MMMM   5
MMMMMMMM MMMMMMMM MMMM   4
MMMMMMMM MMMMMMMM MMMM   3
MMMMMMMM MMMMMMMM MMMM   2
```

Figure 6-5.

6.2.5 SIO Board Test

The SIO Test Diagnostic checks for the presence of an SIO Board and performs rudimentary testing. Before running this diagnostic, connect a special test plug to the RS-232 connector of the SIO Board. This connector is located on the rear panel of the ADVANTAGE. The test plug can be made with a male RS-232 connector as follows:

```
Connect:  pin 2 to pin 3 (RxD to TxD)
          pin 4 to pin 5 (DSR to DTR)
          pin 8 to pin 20 (CTS low)
```

A sample display is shown in Figure 6-6, indicating one SIO Board in connector J5.

```
SIO Board Test - Display Format


NO SIO BOARD IN SLOT 6
TESTING SIO BOARD IN SLOT 5
  BOARD PASSED AT 9600 BAUD
NO SIO BOARD IN SLOT 4
NO SIO BOARD IN SLOT 3
NO SIO BOARD IN SLOT 2
NO SIO BOARD IN SLOT 1
I'm done now!

Type any character to continue.
```

Figure 6-6.

6.2.6 Keyboard Test

The Keyboard Test confirms the operation of every function of the keyboard: that the scan lines are functional; that there is no cross-talk and that N-key rollover is operational; that the auto repeat function is in working order; that all the shift modes scan properly; that the ALL CAPS and CURSOR LOCK lights work correctly. If desired, the Keyboard Test can test every character code which can be generated.

The Keyboard Test is divided into modules, which are in turn divided into sections (see Figure 6-7). The modules and sections are normally executed in the order shown in the figure by following video prompts. However, it is possible to jump to other areas of the test from any given section, as shown by the arrows leaving the '3rd Row' segment of module 'CASE III". This option is discussed in a later section titled 'Changing Sequence'.

Keyboard Test Modules and Sections

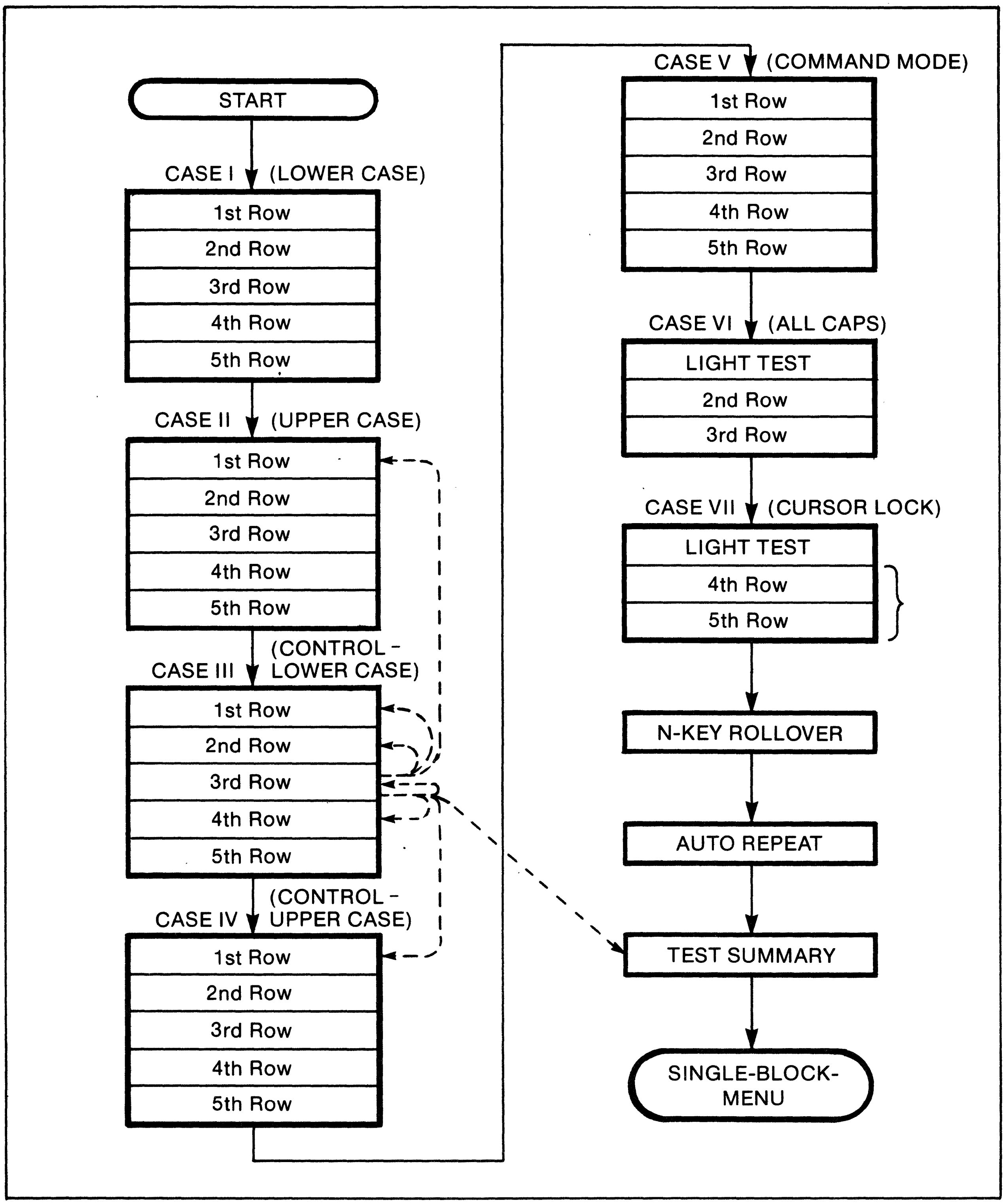


Figure 6-7

A description of the modules in the Keyboard Test is given below:

A. Case I - Lower Case

The Case I through Case VII modules verify correct ASCII coding from the keyboard. The Case I module takes three to six minutes, depending on the speed of the operator.

Specified keys are pressed in a left-to-right sequence, one row at a time, and the characters are echoed on the screen. The abbreviation codes used to represent the characters on the screen are listed in Table 6-1. The Case I module requires the entry of four rows of keys.

The easiest way to input a line of keys is to glide the finger across the keytops from left to right for the designated row. Be sure to include the first indicated key, be it "1", "ESC", etc.

There is a short beep after each line has been entered correctly and its codes verified. If there is an error, a longer beep sounds. This usually indicates that one or more keys in the row was hit incorrectly; a question mark is displayed under the incorrect key entry on the screen display. The Keyboard Test allows three chances to input a row correctly. Then it logs the error, which appears in the summary display at the end of the test, and proceeds to the next row. You cannot correct an error when keying in a row. Quickly finish the row with dummy entries (e.g., spaces) and re-enter the line on the next try. If this was the last try, go to the beginning of the section and try again.

Table 6-1

Keyboard Test - Abbreviation Codes	
Code	Description
2	Lower case 2 on the typewriter keyboard
@	Upper case 2 on the typewriter keyboard. (The SHIFT key is pressed with the 2 key.)
^2	Control 2 on the typewriter keyboard. (The CONTROL key is pressed with the 2 key.
^@	Control SHIFT 2 on the typewriter keyboard. (The CONTROL and SHIFT keys are pressed with the 2 key.)
!2	CMND-2 on the typewriter keyboard. (The CMND key is pressed with the 2 key.)
	Lower case 2 on the numeric pad.
N2	Upper case 2 on the numeric pad. (The SHIFT key is pressed with the 2 key.)
^n2	Control 2 on the numeric pad. (The CONTROL key is pressed with the 2 key.)
!n2	CMND-2, on the numeric pad. (The CMND key is pressed with the 2 key.)
<X]	Delete key
ESC	ESCAPE key
TAB	TAB key
RET	RETURN key
ENT	ENTER KEY
SP	Space bar

B. Case II - Upper Case

This is the same as Case I, except that the SHIFT key is held down while the other keys are entered.

C. Case III - Control, Lower Case

This is the same as Case I, except that the CONTROL key is held down while the other keys are entered.

D. Case IV - Control, Upper Case

This is the same as Case I, except that the CONTROL and SHIFT keys are held down while the other keys are entered.

E. Case V - Command

This is the same as Case I, except that the CMND key is held down while the other keys are entered.

F. Case VI - All Caps

This module requires that the operator verify the correct operation of the ALL CAPS light and key in two rows for code verification.

G. Case VII - Cursor Lock

This module requires that the operator verify the correct operation of the CURSOR LOCK light and key in two rows on the numeric key pad for code verification.

H. N-Key Rollover

This module checks for interference between keyboard signals when multiple keys are pressed. The module is summarized in Figure 6-8. The test procedure is given below:

1. Four keys must be held down with the left hand while pressing a sequence of keys with the right hand. First,starting with the little finger of the left hand, press and hold down the "2" on the main keyboard, then with the next finger press and hold down the "E", and so forth with the "F" and "B" keys. The display will show a repeating sequence of:

   BBBBBBBBBBB...

2. While keeping the '2EFB' keys pressed, with the right hand press the RETURN key several times until blanks are printed on the screen:

   (spaces)...

3. While still keeping the '2EFB' keys pressed, with the right hand press these four keys on the main keyboard once each, releasing each in turn: nine "9", oh "o", el "l" and RETURN.

4. Now, repeat the procedure, except that four keys on the numeric pad must be held down with the right hand while pressing a sequence of keys with the left hand. Starting with the index finger of the right hand, press the "7", "5", and "3" keys, then with the thumb press the zero "0" key--all on the numeric pad. The display will show a repeating sequence of:

   0000000000...

5. While keeping the '7530' keys pressed, with the left hand press the RETURN key several times until blanks are indicated on the screen:

   (spaces)...

6. While still keeping the '7530' keys pressed, press and release each of these four keys with the left hand: nine "9", oh "o", el "l" and RETURN.

Each of the two parts (main keyboard test and numeric pad test) may be repeated up to three times if errors were made in performing the test. If the test was performed successfully, the following message is printed:

N-KEY ROLLOVER Passed

I. Auto Repeat

For this test simply press and hold down any key, as instructed by the video prompts. There are three tries to perform this function.

N-Key Rollover Test

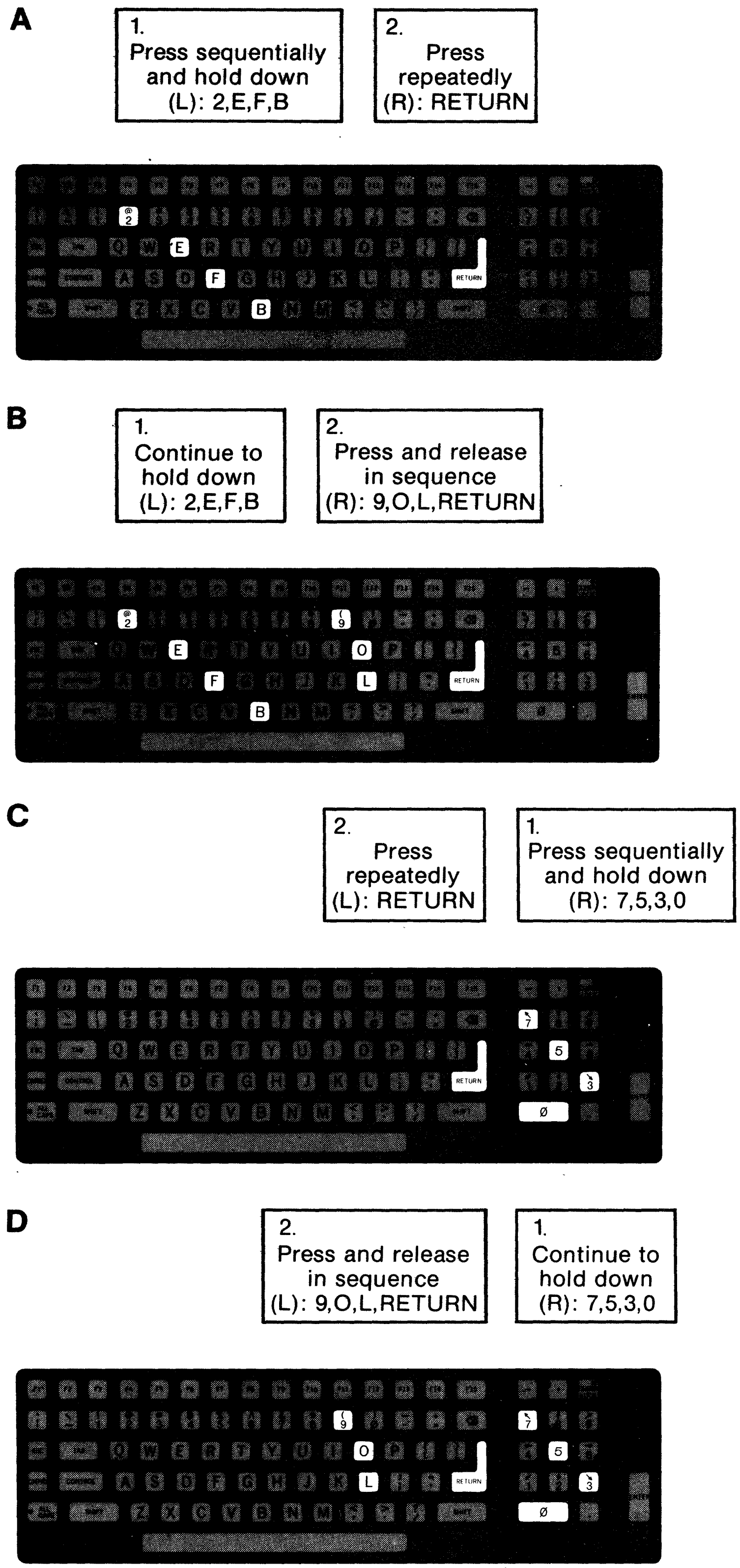


Figure 6-8.

J. Test Summary

On completion of the auto repeat module, a summary of the Keyboard Test Diagnostic is displayed. A sample display is shown in Figure 6-9. This display consists of a matrix that represents the main keyboard and numeric pad with a "status matrix" to the right indicating which cases were tested for which rows (the *s show what was tested.)

The keyboard matrix is composed of squares representing individual keys. The squares enclose either a question mark ("?") or a number. The question mark means that that key was never tested for any case. The numbers indicate the number of errors received at each key for the cases indicated on the status matrix to the right. They should all be zeroes for a good keyboard.

If there are any errors, a message is printed in the summary directly after the keyboard matrix, and the suspect keys are shown in the matrix itself. Errors are also announced during the test procedure.

Keyboard Test Summary

```
[0] [0] [0] [0] [0] [0] [0] [0] [0] [0] [0] [0] [0] [0] [0]     [0] [0]             |*|*|*|*|*|
[0] [0] [0] [0] [0] [0] [0] [0] [0] [0] [0] [0] [0] [0] [0]     [0] [0] [0]         |*|*|*|*|*|
[0]  [0] [0] [0] [0] [0] [0] [0] [0] [0] [0] [0] [0] [0]        [0] [0] [0]         |*|-|-|-|-|
      [0] [0] [0] [0] [0] [0] [0] [0] [0] [0] [0] [0]  [ 0 ]    [0] [0] [0]         |*|-|-|-|-|

       [0] [0] [0] [0] [0] [0] [0] [0] [0] [0] [0]          [   0 ] [0]  [ 0 ]     |*|-|-|-|-|
                         [     0     ]
Notes on the Keyboard Summary:
      [0]  : Means that there is nothing wrong                Lower case
      [n]  : Indicates that n tests failed at this key        Upper case
      [?]  : Indicates that this key was not tested           Control case
ALL-CAPS Feature Test-                                         Shifted Ctl case
      Passed                                                   Command case

N-Key rollover Test -                                          |*| = case was tested
      Passed                                                   |-| = case wasn't tested

Automatic Repeat Feature / Buffer-Full Flag Test -
      Passed
KEYBOARD PASSED THIS SUBSET OF THE EXHAUSTIVE TEST

Type any character to continue.
```

Figure 6-9.

K. Changing Sequence

Instead of executing the sections of the Keyboard Test in their normal sequence, it is possible to execute only the desired sections (or modules) in any sequence. Figure 6-2 illustrates the possible moves that can be made from any given section. They are:

1. Skip to previous module (in this case, the 'Case II' module, section '1st Row').
2. Repeat current module.
3. Skip to previous section.
4. Repeat current section.
5. Skip to next section.
6. Skip to next module.
7. Skip to Test Summary.

In order to perform any of these moves, the Keyboard Test must be waiting for the <u>first</u> response to any test for that section.) At this point, when the CONTROL - C or 'left arrow' is entered, control returns to the Shell Monitor, and any of the control keys listed in Table 6-2 may be entered to perform the desired move. Note that return may be easily made to the Single Block Menu by pressing CONTROL - C twice, then pressing any other key.

Table 6-2

Keyboard Test Control Keys

DESIRED MOVE	KEY(S)	ALTERNATE KEY(S)
Return to Shell Monitor	CONTROL-C	← or Shifted ← (row section only)
AFTER RETURNING TO SHELL MONITOR		
Skip to previous module	Shifted ↑	
Repeat current module	Shifted →	M R
Skip to previous section	Unshifted ↑	S U
Repeat current section	Unshifted →	S R
Skip to next section	Unshifted ↓	S D
Skip to next module	Shifted ↓	M D
Skip to Test Summary	CONTROL-C	

6.2.7 Display Monitor Test

The Display Monitor Diagnostic Test is used primarily to check resolution, screen rippling, blooming and distortion of the screen or characters.

This test places a pattern on the screen along with some text (see Figure 6-10 for text). It then performs a series of disk accesses to instigate screen rippling (if any), after which it reverses the screen five times to bring out blooming (if any). This pattern is repeated continuously until the operator terminates the test.

Display Format for Display Monitor Test

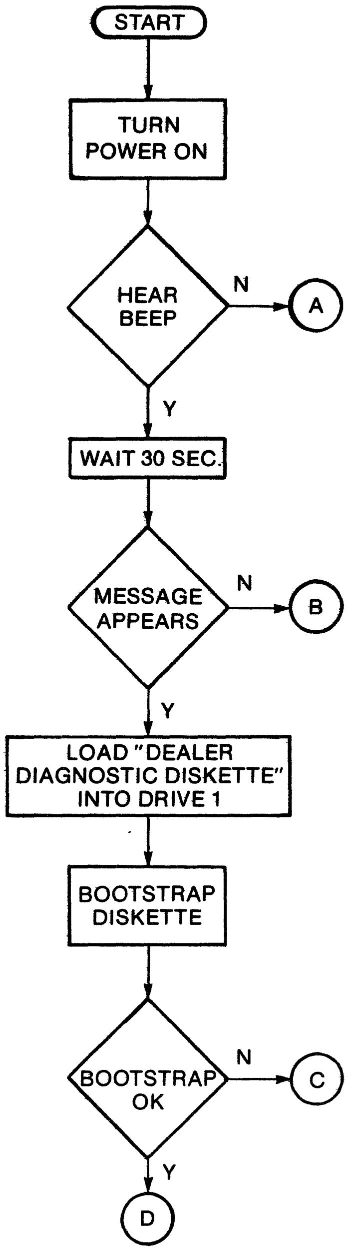
```
################################################################################
################################################################################
################################################################################
################################################################################
################################################################################
#####                                                                      #####
#####          Display Monitor Test - Ver. 1.0-B - 04/20/81                #####
#####                                                                      #####
#####                Please check this display for:                        #####
#####                    1) Good Resolution                                #####
#####                    2) Low distortion - esp. in corners               #####
#####                    3) Minimal Blooming on Reverse Video              #####
#####                    4) No Changing Shapes or Ripple                   #####
#####                    5) No Erratic Dots on the Screen                  #####
#####                                                                      #####
#####                HIT RETURN TO EXIT                                    #####
#####                                                                      #####
#####                                                                      #####
#####                                                                      #####
################################################################################
################################################################################
################################################################################
################################################################################
################################################################################
```

Figure 6-10

6.3 TROUBLESHOOTING CHART

The troubleshooting chart on the following pages is intended to assist service personnel in isolating a machine failure to a replaceable subassembly. The troubleshooting chart is used when the machine failure is serious enough to prevent loading of the diagnostic programs.

Instructions for opening the ADVANTAGE cabinet and removing subassemblies are in Section 6.5.

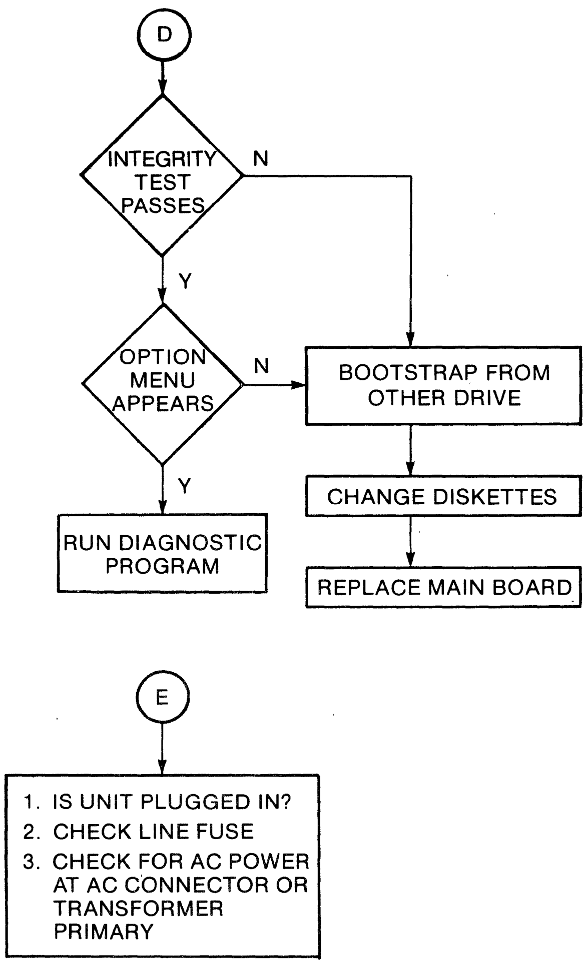


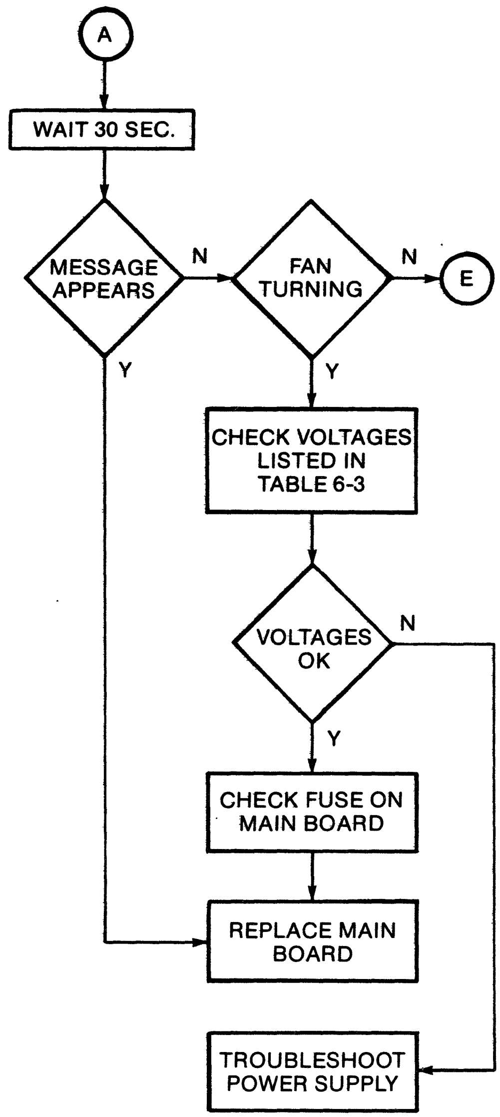
COMMENTS

Message consists of the words "LOAD SYSTEM" accompanied by a cursor. The rest of the screen is blank.

Instructions for loading diskettes and bootstrapping are in Section 2.2.

Bootstrap is successful if screen message changes to "Integrity Test".


D
INTEGRITY TEST PASSES
N
Y
OPTION MENU APPEARS
N
Y
BOOTSTRAP FROM OTHER DRIVE
CHANGE DISKETTES
REPLACE MAIN BOARD
RUN DIAGNOSTIC PROGRAM
E
1. IS UNIT PLUGGED IN?
2. CHECK LINE FUSE
3. CHECK FOR AC POWER AT AC CONNECTOR OR TRANSFORMER PRIMARY



COMMENTS

Message consists of the words "LOAD SYSTEM" accompanied by a cursor. The rest of the screen is blank.

Fuse is located in right rear corner of board.

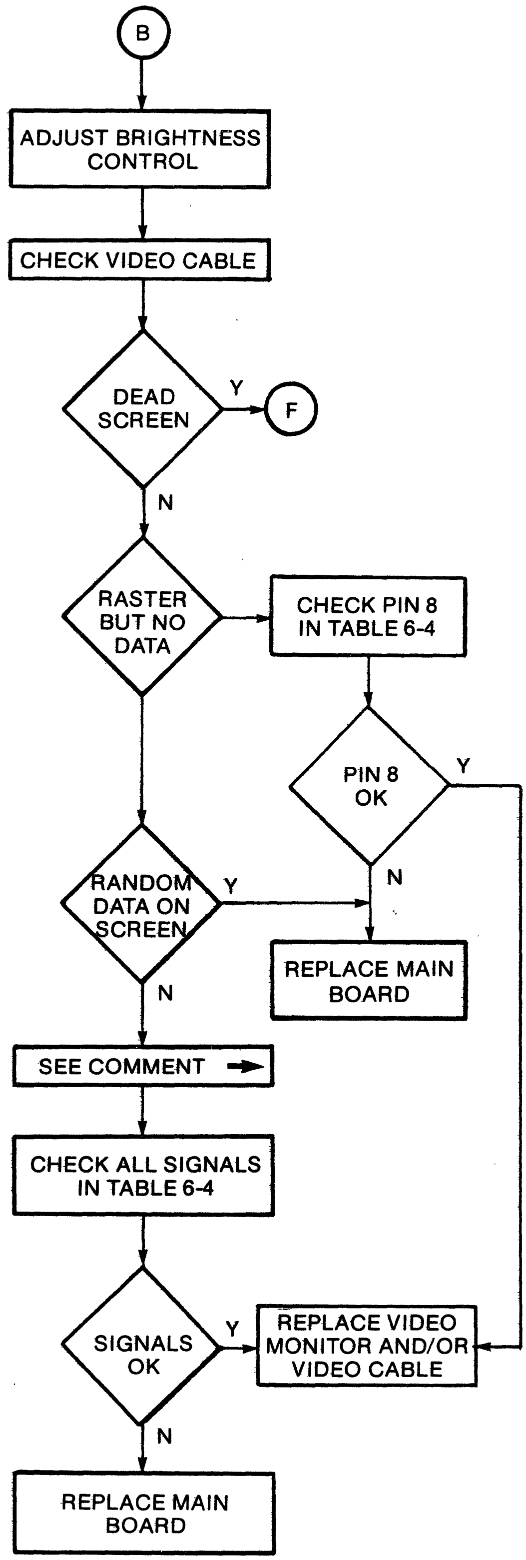
See Appendix F, Main PCB Schematic.

Table 6-3

Main Board Input Power (J11)	
Pin Number	**Description**
1	-23 VDC ± 10%
2	Not Used
3	+23 VDC ± 10%
4	Power/signal ground
5	Chassis ground
6	17 VAC ± 10%

Table 6-4

Main Board Video Interface (J7)	
Pin Number	**Description**
1	Power/signal ground
2-4	Not used.
5	Power/signal ground
6	Horizontal sync. Positive going pulses at TTL levels.
7	+12 VDC ± 10%
8	Video data at TTL levels. High=light, low=dark.
9	Vertical sync. Negative going pulses at TTL levels.
10	Power/signal ground



COMMENTS

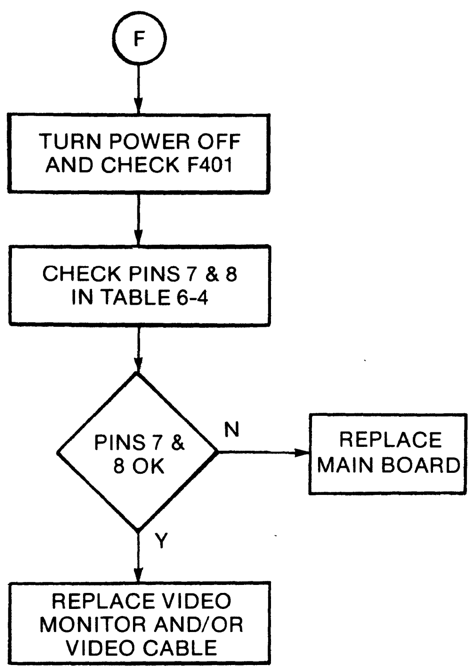
Ensure that the video cable is properly plugged into J7 of the Main Board and into the Video PCB. See warning below.

Raster is present when there is a uniform brightness throughout most of the screen area. If the brightness is turned all the way up, periodic lines of higher intensity can also be seen

The Video PCB contains various potentiometers which can be used to adjust display width, vertical linearity, etc.

WARNING

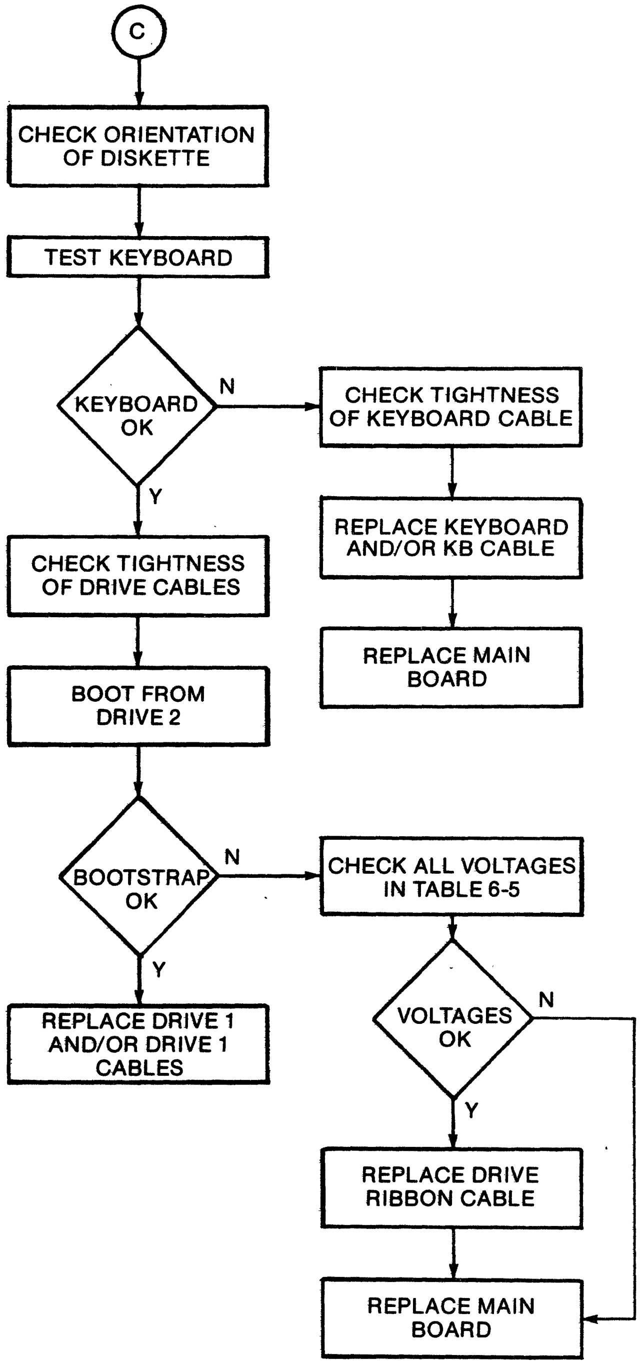
WHEN REACHING INSIDE THE VIDEO MONITOR, KEEP HANDS AWAY FROM THE RED HIGH VOLTAGE WIRE ON THE SIDE OF THE CRT. THIS WIRE CARRIES A POTENTIALLY FATAL CHARGE OF 10,000 VOLTS.



COMMENTS

F401 is a 2A miniature fuse located on the Video PCB.

WARNING
WHEN REACHING INSIDE THE VIDEO MONITOR, KEEP HANDS AWAY FROM THE RED HIGH VOLTAGE WIRE ON THE SIDE OF THE CRT. THIS WIRE CARRIES A POTENTIALLY FATAL CHARGE OF 10,000 VOLTS.



COMMENTS

To test the keyboard:

1. Check operation of the ALL CAPS key. The light in the key-cap should go on and off, alternating each time the key is pressed.
2. Press several alpha-numeric keys (except "D") one at a time. Each key should cause a "beep".

Bootstrap is succesful if screen message changes to "Integrity Test".

Table 6-5.

Main Board - Floppy Disk Power (J10)	
Pin Number	Description
1	+12 VDC ± 10%
2	Ground
3	Not used
4	Ground
5	+5 VDC ± 10%
6	+12 VDC ± 10%
7	Ground
8	Ground
9	+5 VDC ± 10%

6.4 THE MINI-MONITOR

The Mini-Monitor is a primitive diagnostic routine located in the Boot PROM. It provides the ability to examine and change individual bytes of RAM, execute single input and output instructions, and jump to a memory location. When the Mini-Monitor is entered (see Section 2.2) an asterisk (*), used as a prompt, appears on the screen. Commands may then be entered to the Mini-Monitor. The commands are listed in Table 6.6.

There is practically no error checking of the commands. If a hexadecimal digit is expected but some other character is received, the results are unpredictable. However, CTRL-C can be used to abort entry of a command before the RETURN key is pressed. The command letters must be given in upper case only; lower case letters are not recognized.

Table 6-6

Mini-Monitor Commands		
Command	**Name**	**Description**
Dxxxx	DISPLAY	Display contents of address xxxx. When the byte is displayed, type: YY = Replace in xxxx and display next byte <RETURN> = Exit from command
Ixx	INPUT	Read data byte from Port xx amd display it.
Oyyxx	OUTPUT	Write data byte yy to Port xx.
R	READ	Read program from the DIAGNOSTIC board slot 3.
Jxxxx	JUMP	Go to address xxxx. Before jumping, the Mini Monitor loads the address of its re-entry point into the Z80 H and L registers.
Q	QUIT	Exit from Mini-Monitor. A beep sounds and the message 'LOAD SYSTEM' is re-displayed.

NOTES

- xxxx = 4-digit hexadecimal number
xx = 2-digit hexadecimal number
yy = 2-digit data byte in hexadecimal
- Control-C can be used to cancel a command if entered before <RETURN>.

6.5 ASSEMBLY REMOVAL AND INSTALLATION PROCEDURES

SUMMARY OF PRECAUTIONS

1. Service should be performed only by trained personnel.
2. Before servicing the ADVANTAGE, disconnect the power source by disconnecting the power cord from the back of the unit.
3. Handle PC boards carefully, as the underside has sharp pin protrusions.
4. Be extremely careful of the high voltage lead on the CRT; be sure to discharge it properly before disconnecting the high voltage connector. Instructions for discharging the high voltage lead are given in Section 6.5.7.
5. Be extremely careful not to bump the CRT, which is attached to the Cover Assembly. Pay particular attention to this when opening and closing the ADVANTAGE cabinet (Section 6.5.2). Handle the CRT with extreme care. If the glass is fractured, the CRT may implode and create a hazard because of flying glass.

6.5.1 Tools Required

The following tools are used in the removal and installation procedures:

- Screwdrivers: One 6-inch flatblade
  One 4-inch flatblade
  One 4-inch Phillips (disk only)
  One 90-degree angled Phillips
  One small thin flatblade (I/O boards only)
- One insulated grounding probe (video only)
- Safety goggles (video only)

6.5.2 Opening and Closing the ADVANTAGE Cabinet

To open the ADVANTAGE cabinet, proceed as follows:

1) Disconnect the AC power source. Turn the Power ON/OFF switch to OFF. Unplug the power cord from the back of the machine, as shown in Figure 6-11.

WARNING
MAKE SURE THAT THE AC POWER SOURCE IS DISCONNECTED BEFORE PROCEEDING.

Power Cord Removal

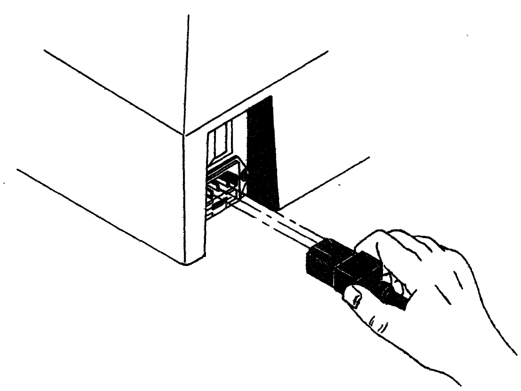


Figure 6-11

2) Disconnect any I/O cables which may be connected to the rear of the ADVANTAGE cabinet.

3) Remove mounting screws. To reach the mounting screws on the bottom of the ADVANTAGE grasp the unit firmly and carefully turn it upside down. Unscrew the four mounting screws near the front of the base (1 through 4 in Figure 6-12). Unscrew the remaining two mounting screws, which are recessed at the back of the unit (5 and 6 in the figure). When the screws are removed, grasp the unit firmly and carefully return it to the upright position.

Bottom View of the ADVANTAGE

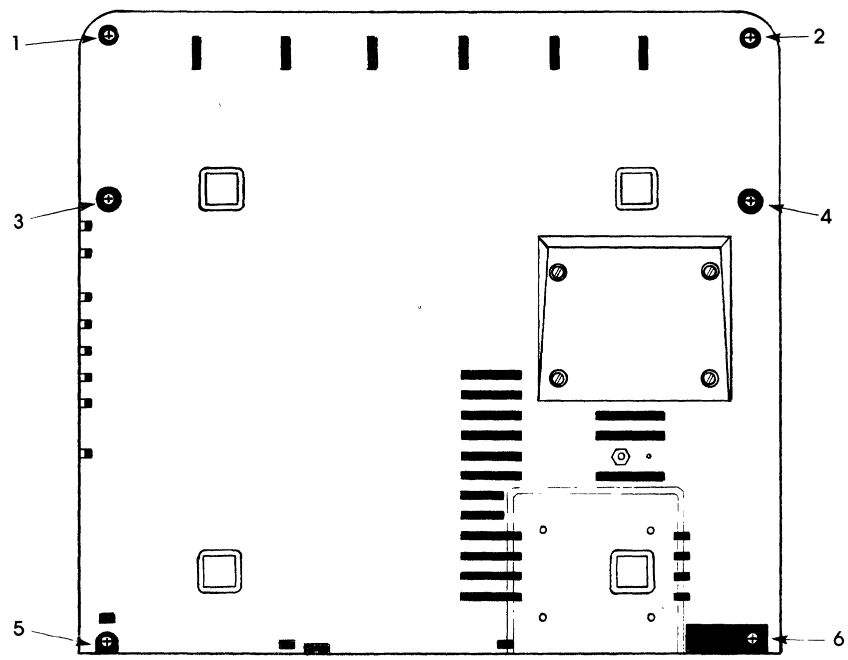


Figure 6-12

4) Clear away the area behind the ADVANTAGE cabinet, to provide space for the Cover Assembly (see Figure 6-13d).

CAUTION
While performing the next 2 steps, do not allow the Cover Assembly to drift too far to the left or right, as the CRT tube socket may be damaged by striking other internal components.

5) Carefully lift the Cover Assembly straight up to the position shown in Figure 6-13b.

6) Carefully rotate the Cover Assembly toward the rear, and allow it to rest on its rear surface, with the CRT screen facing up (Figure 6-13d).

Cabinet Separation Sequence

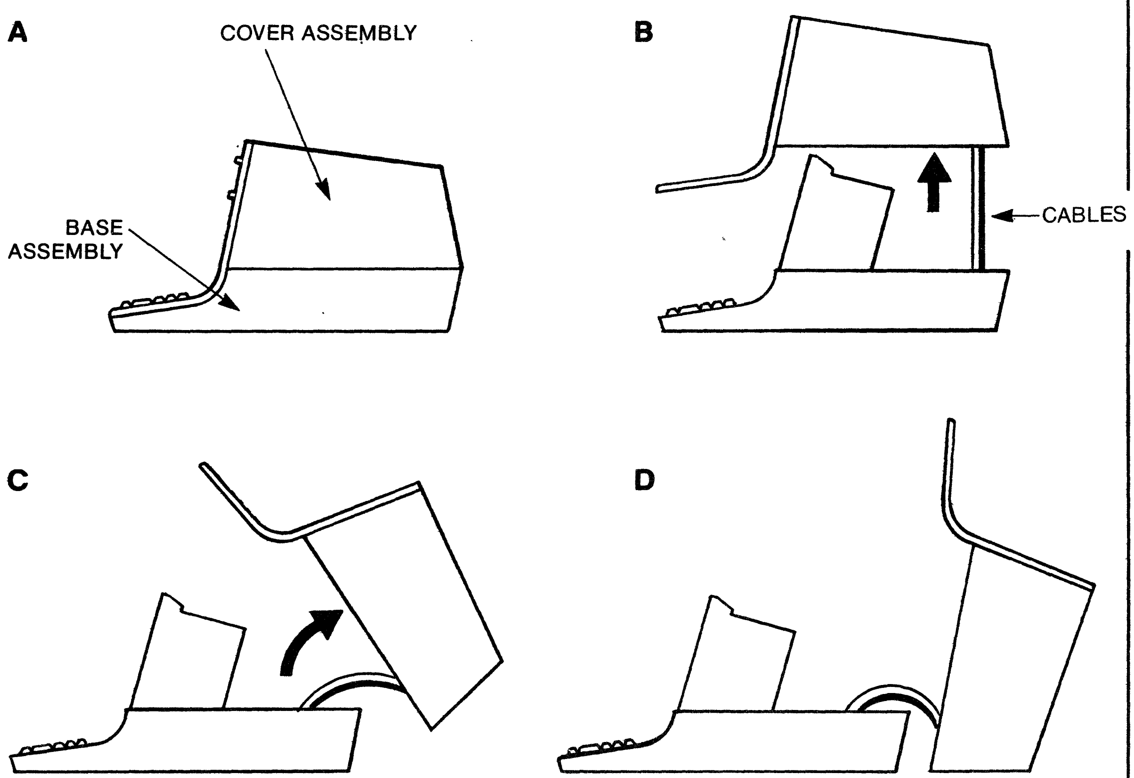


Figure 6-13

When the Base Assembly and the Cover Assembly have been separated, the major components of the system are exposed. These components are shown in figure 6-14.

Inside the Base Assembly are four major components:

1. Main PC Board
2. Keyboard
3. Disk Drive Assembly
4. Transformer

The Cover Assembly holds three major components:

1. CRT
2. Video PC Board
3. Fan

Major Components Inside the ADVANTAGE

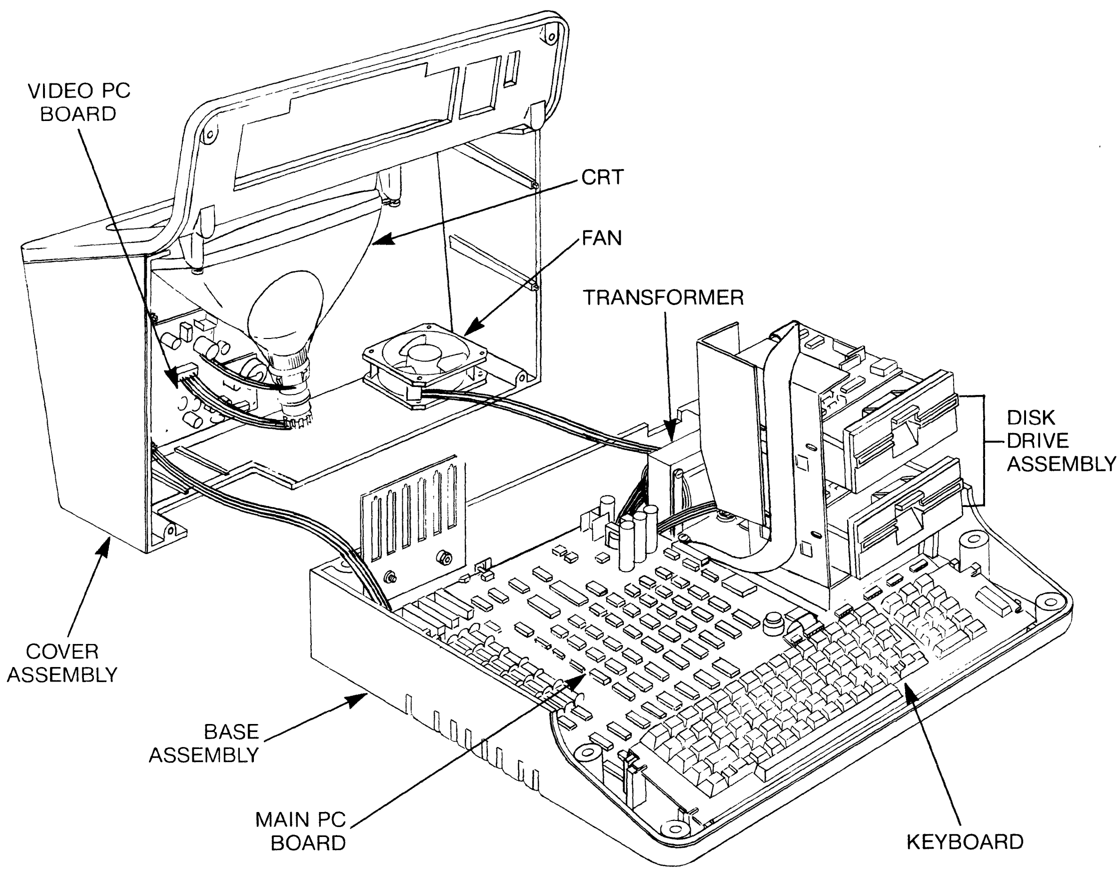


Figure 6-14

The procedure for closing the ADVANTAGE cabinet is essentially the reverse of the procedure for opening it.

6.5.3 Removing and Installing the Keyboard

To remove the keyboard, proceed as follows:

1) Open the ADVANTAGE cabinet as described in Section 6.5.2.

2) Lift the keyboard out of the Base Assembly and place it in front of the Base Assembly as shown in Figure 6-15.

Base Assembly

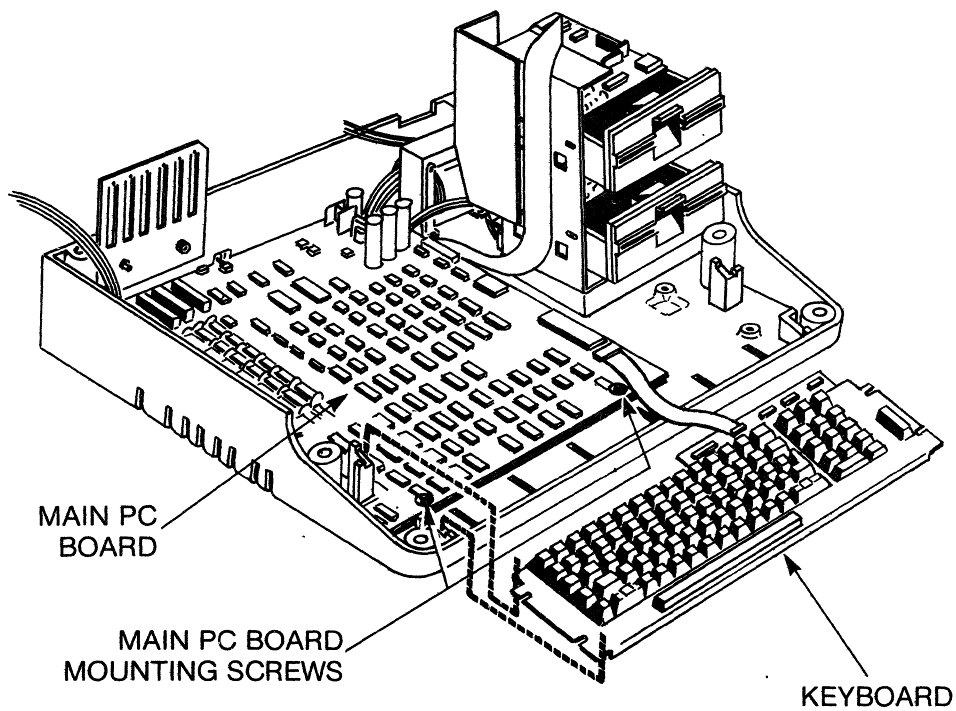


Figure 6-15

3) Disconnect the keyboard cable from J8 on the Main PC Board (see Figure 6-16). To remove the cable, pull straight up on the cable connector.

4) Remove the keyboard, which is now free.

To install the keyboard, reverse the above procedure.

6.5.4 Removing and Instaling The Main PC Board

Refer to Figure 6-16 for the positions of the components referenced in this procedure.

Cable Connections

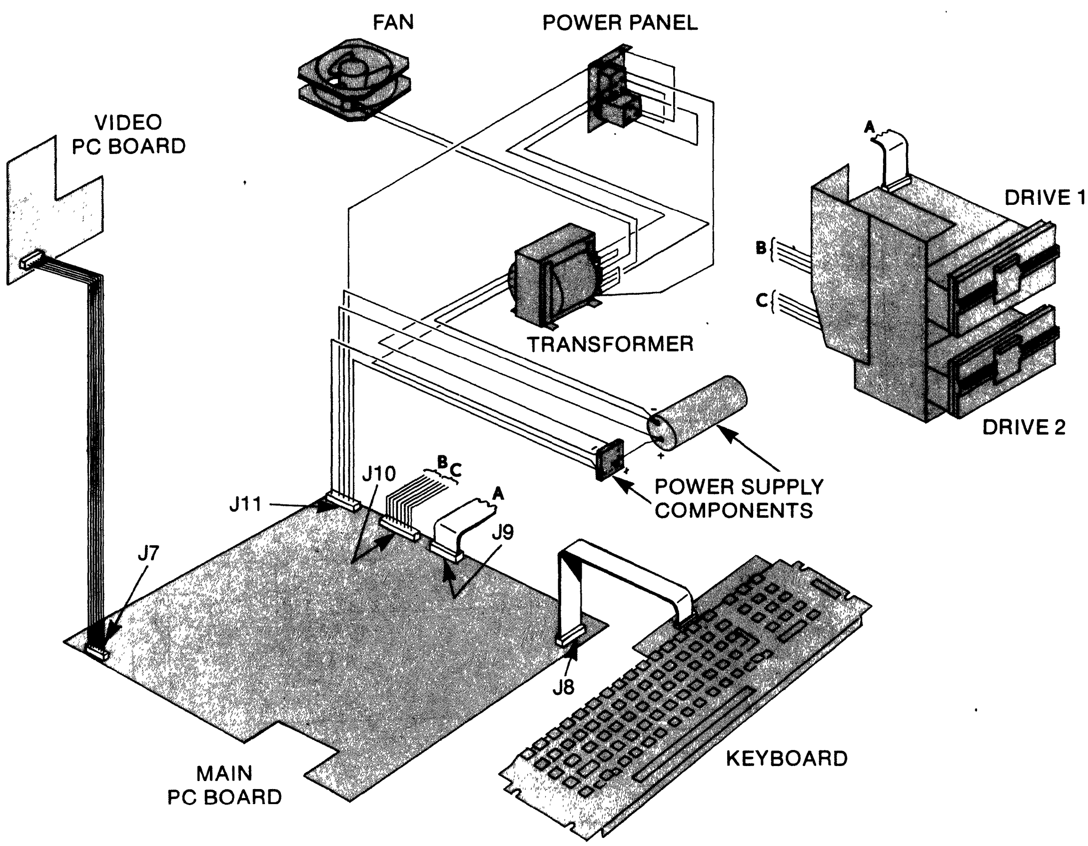


Figure 6-16

A. To remove the Main PC Board, proceed as follows:

1) Open the ADVANTAGE cabinet as described in Section 6.5.2.

2) Remove the keyboard as described in Section 6.5.3.

3) Disconnect the video cable from J7 on the Main PC Board (see Figure 6-16). To remove the cable, pull straight up on the cable connector.

4) If any I/O Boards are installed in the Main PC Board, record their slot positions. When they are reinstalled, they must be returned to these same positions.

5) Remove the I/O Boards. For each board, remove the retaining screw (if any). Gently pull board toward the front of the system and upward, removing it from its connector.

6) Remove the Main PC Board mounting screws. Unscrew the retaining screws located along the front edge of the main PC board (see Figure 6-15).

7) Lift up the front edge of the Main PC Board as shown in Figure 6-17a, and pull forward until the rear edge of the PC board is free of the base plate. (The cables along the right-hand edge of the PB board are still connected at this time.)

8) Maneuver the Main PC Board into the position shown in Figure 6-17b.

9) Remove the connectors from J8 through J11 by pulling them straight up. Do not pull on the wires.

10) The Main PC Board can now be lifted out of the base plate.

B. To install the Main PC Board, proceed as follows:

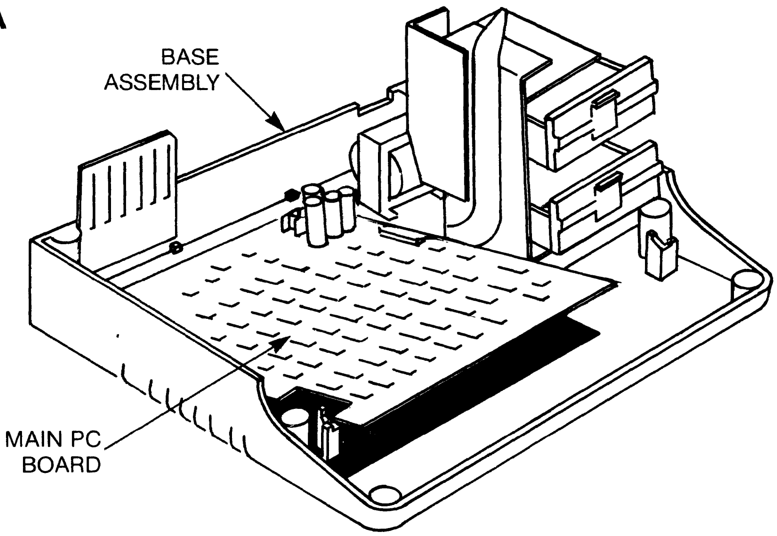
1) Place the Main PC Board in the position shown in Figure 6-17a.

2) Clear away any cables or connectors that may be under the PC board.

3) Slide the rear edge of the Main PC Board under the three tabs at the rear of the base plate.

Main PC Board Removal

A



B

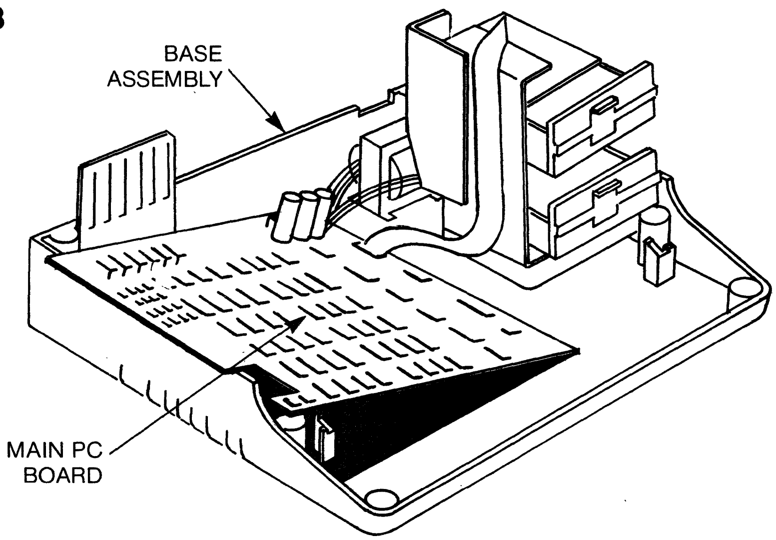


Figure 6-17

4) Lower the PC board to the horizontal position.

5) Install the cables in J7 through J11 as shown in Figure 6-16.

6) Install the two Main PC Board mounting screws at the locations shown in Figure 6-15, and tighten the screws.

7) Reinstall the I/O Boards (if any). Insert them into their connectors at the left rear corner of the Main PC Board. These boards must be returned to the same connectors from which they were removed. Reinstall the retaining screws (if any) associated with any of these boards.

8) Install the keyboard as described in Section 6.5.3.

9) Close the ADVANTAGE cabinet as described in Section 6.5.2.

Disk Drive Shield Removal

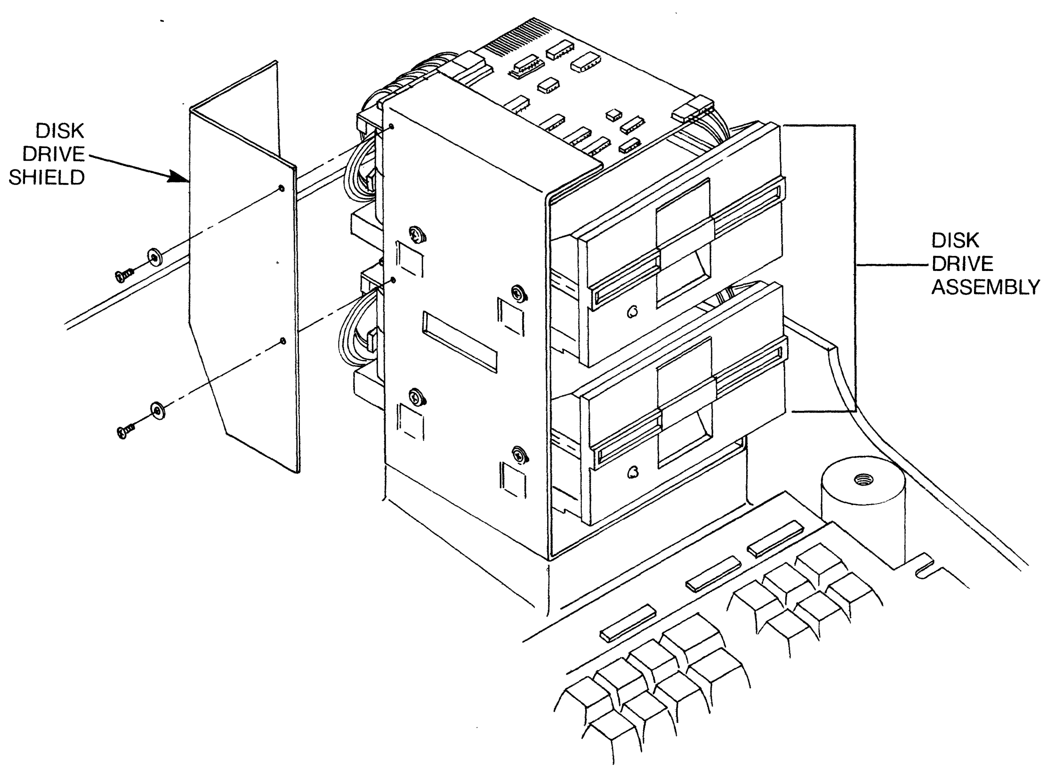


Figure 6-18

6.5.5 Removing and Installing a Disk Drive

The following steps cover the removal of the upper disk drive. To remove the <u>lower</u> disk drive apply these instructions to the corresponding parts of the lower drive.

1) Open the ADVANTAGE cabinet as described in Section 6.5.2.

2) Remove the two screws securing the Disk Drive Shield, and remove the shield (see Figure 6-18). Avoid dropping the screws into the base plate, as they may roll under the Main PC Board and be difficult to retrieve.

Disk Drive Cabling

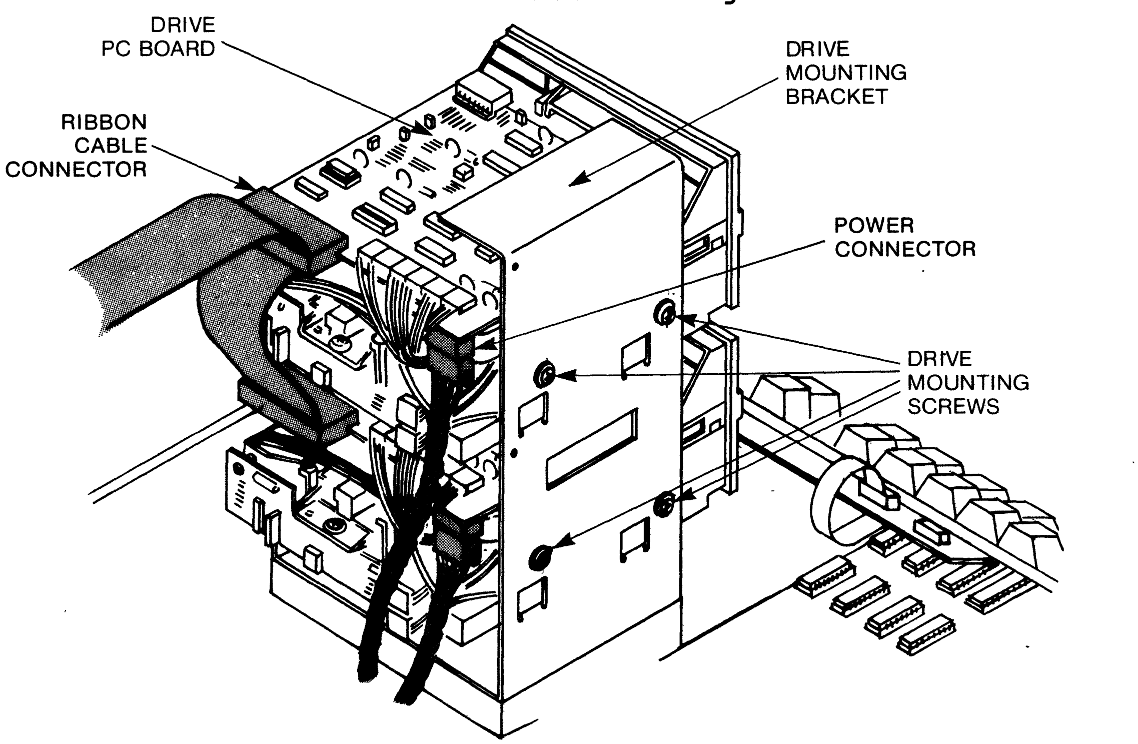


Figure 6-19

3) Disconnect the power connector shown in Figure 6-19. Hold onto the edge of the Drive PC Board while pulling down on this connector.

4) Disconnect the ribbon cable connector shown in Figure 6-19 by pulling the connector straight off the rear of the drive PC board.

5) Remove the drive mounting screws. There are four screws, two at each side, holding the drive to the drive mounting bracket.

6) Remove the upper drive by sliding it forward as shown in Figure 6-20.

Disk Drive 1 Removed

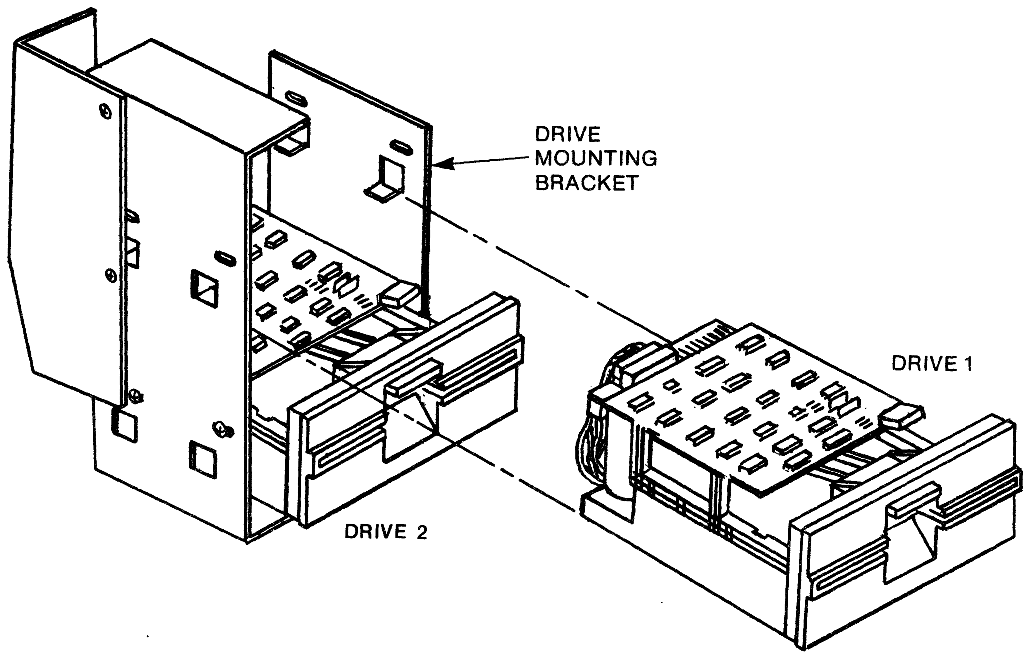


Figure 6-20

The installation procedure for either disk drive is essentially the reverse of the procedure given for its removal, except that the position of the drive may have to be adjusted, so that the front panel of the drive mates properly with the front of the cabinet.

6.5.6 Removing and Installing the Power Supply Components

This section explains how to remove and install the diode bridge and capacitor located behind and below the disk drives (see Figure 6-21). To remove either of these components proceed as follows:

1) Open the ADVANTAGE cabinet as described in Section 6.5.2.

2) Remove both disk drives following the procedure described in Section 6.5.5. The drives must be removed to gain access to the mounting bracket for the power supply components.

3) From the top of the chassis, remove the two screws which secure the power supply mounting bracket shown in Figure 6-21.

4) Remove the wires from the desired component (either the diode bridge or the capacitor), carefully marking their location so that they may be re-connected later.

5) Remove the component from its mounting bracket.

To install either of the power supply components, reverse the above procedure. When installing the diode bridge, insure that the (+) and (-) corners of the bridge are positioned as shown in Figure 6-21.

Power Supply Components

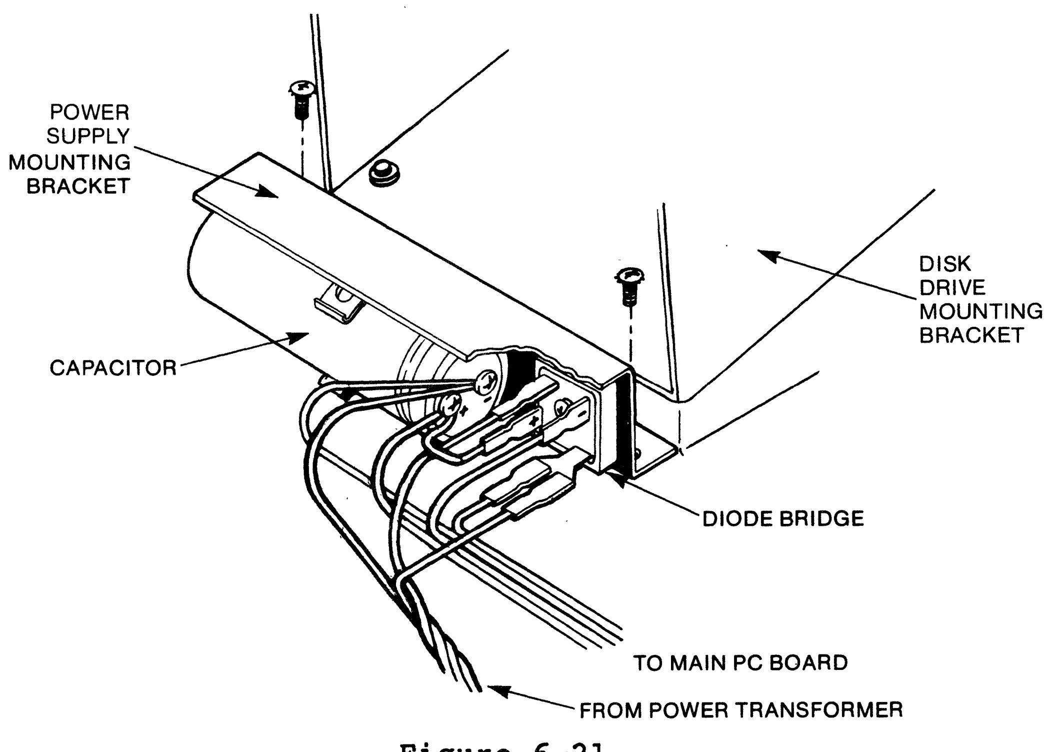


Figure 6-21

6.5.7 Removing and Installing the CRT and Video PC Board

If either the CRT or the Video PC board needs to be replaced, then both of these assemblies should be replaced. The CRT and Video PC Board are factory aligned and stocked as matched pairs. Replacing just one assembly may result in a misaligned video display.

WARNING
THIS PROCEDURE SHOULD BE PERFORMED ONLY BY QUALIFIED PERSONNEL. WEAR SAFETY GLASSES OR EQUIVALENT EYE PROTECTION WHEN PERFORMING THIS PROCEDURE. BE EXTREMELY CAREFUL NOT TO STRIKE ANY OBJECT AGAINST THE CRT, OR TO PUT PRESSURE ON THE NECK OF THE CRT. IF THE CRT IS BROKEN IT MAY IMPLODE AND CREATE A HAZARD BECAUSE OF FLYING GLASS.

A. To remove these assemblies proceed as follows:

1) Open the ADVANTAGE cabinet as described in Section 6.4.2. The video components described in this section are shown in Figures 6-22, 6-23 and 6-24.

2) Disconnect the two wires from the fan by grasping the wire terminals and pulling them off as shown in Figure 6-23.

COVER ASSEMBLY

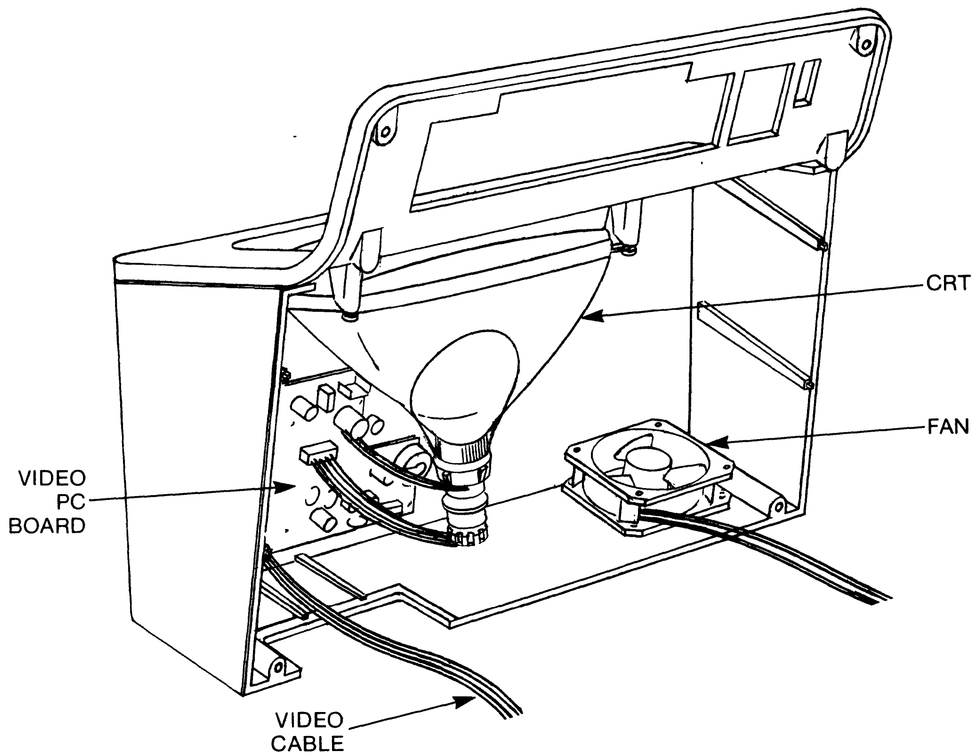


Figure 6-22

Fan Cable Removal/Installation

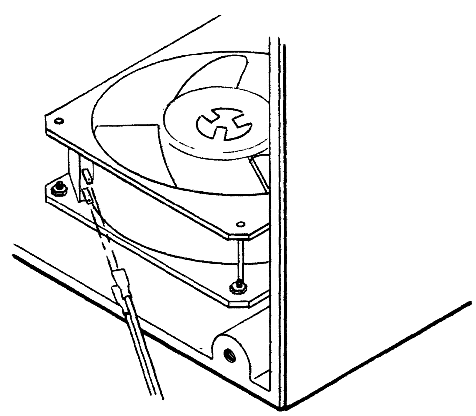


Figure 6-23

3) Disconnect the Video Cable by pulling the cable connector off the Video PC Board (see Figure 6-24).

4) On completion of step 3, the Cover Assembly is completely separated from the Base Assembly. Turn the Cover Assembly upside down so that the Video PC Board is in the horizontal position

WARNING
THE RED SUCTION CUP/CLIP CONNECTED TO THE SIDE OF THE CRT IS THE HIGH VOLTAGE CONNECTOR, WHICH MAY CARRY A POTENTIALLY FATAL CHARGE OF 10,000 VOLTS, EVEN WITH THE POWER TURNED OFF. THE HIGH VOLTAGE CONNECTOR MUST BE DISCHARGED BEFORE DISCONNECTING IT FROM THE CRT. THIS CAN ONLY BE DONE BY A QUALIFIED TECHNICIAN.

5) Discharge the high voltage connector. Connect one end of a well insulated grounding probe to the wire loop on the side of the CRT (see Figure 6-24). Push the other end of the probe down between the side of the CRT and the high voltage connector until the probe touches the metal contact.

6) Disconnect the high voltage lead. Peel back the rubber portion of the high voltage connector and observe the two metal contacts underneath. Slide the connector to the side and pull to release the first contact. Slide the connector in the opposite direction to release the second contact.

7) Remove the CRT socket cable by pulling the cable connector straight off the end of the CRT neck (see Figure 6-24).

Video Components

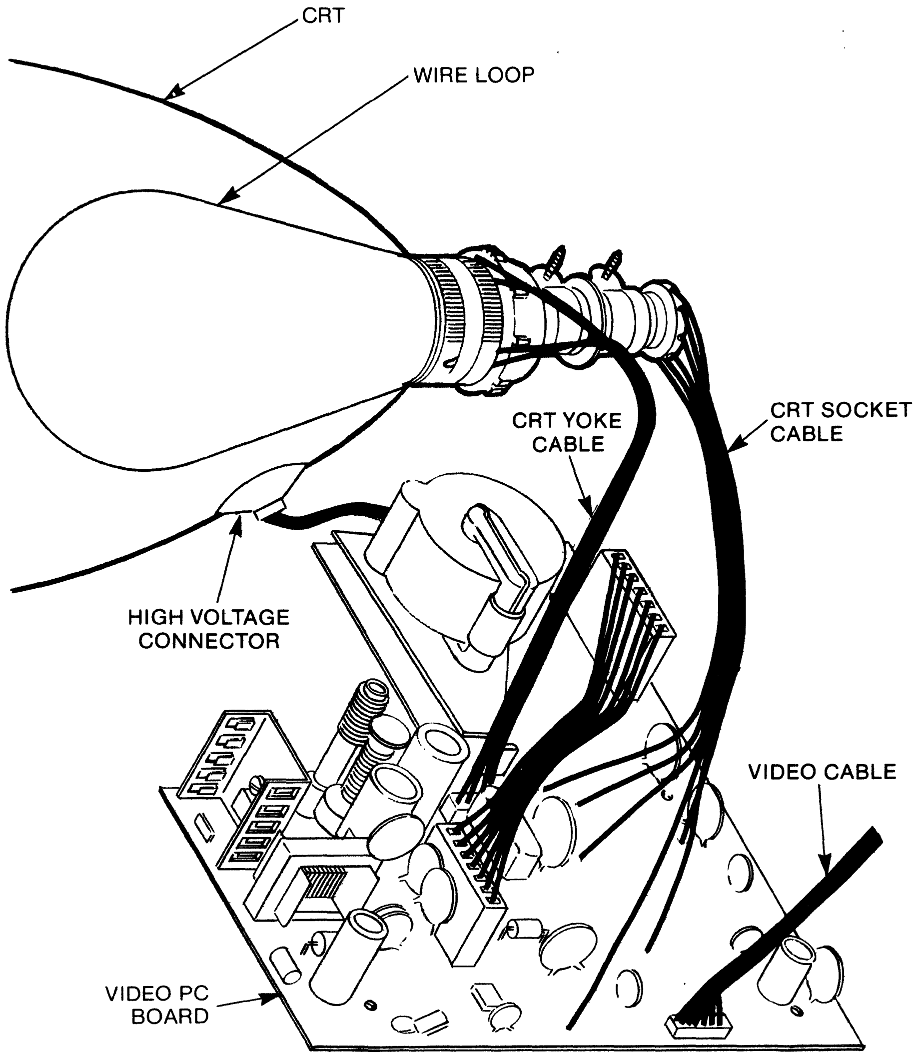


Figure 6-24

8) Disconnect the CRT yoke cable from the Video PC Board by removing the connector attached to the PC board (see Figure 6-24)

9) Remove the Video PC Board mounting screws. Unscrew the five retaining screws shown in Figure 6-25.

Video PC Board

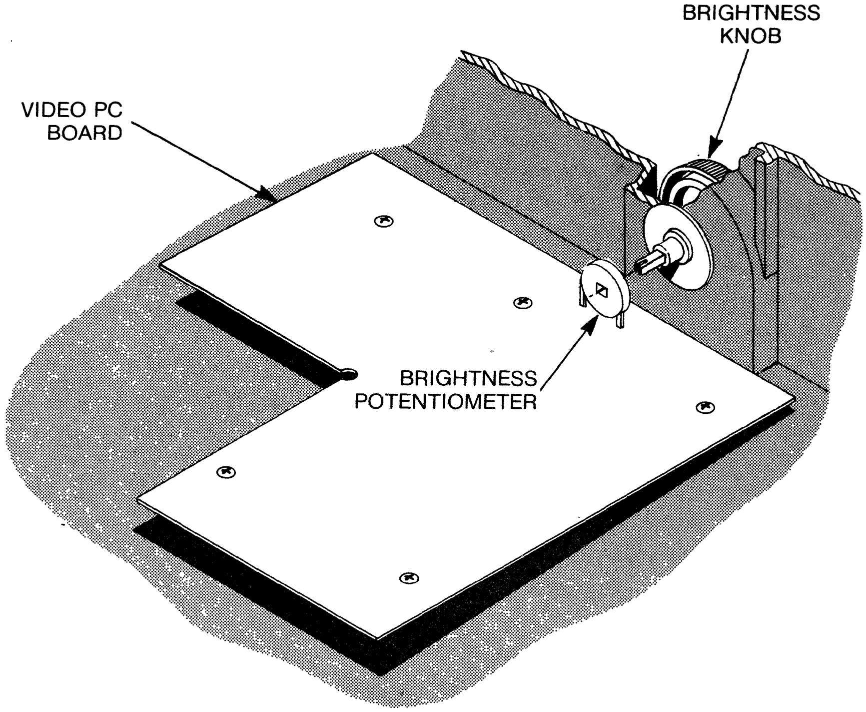


Figure 6-25

10) Pull the Video PC Board away from the brightness knob, until the brightness knob shaft disengages from the brightness potentiometer (see Figure 6-25).

11) The Video PC Board is now completely free and may be lifted out of the Cover Assembly.

12) Place the Cover Assembly on the edge of a work bench as shown in Figure 6-26. Use padding on the work bench to prevent the cabinet from being scratched.

CRT REMOVAL

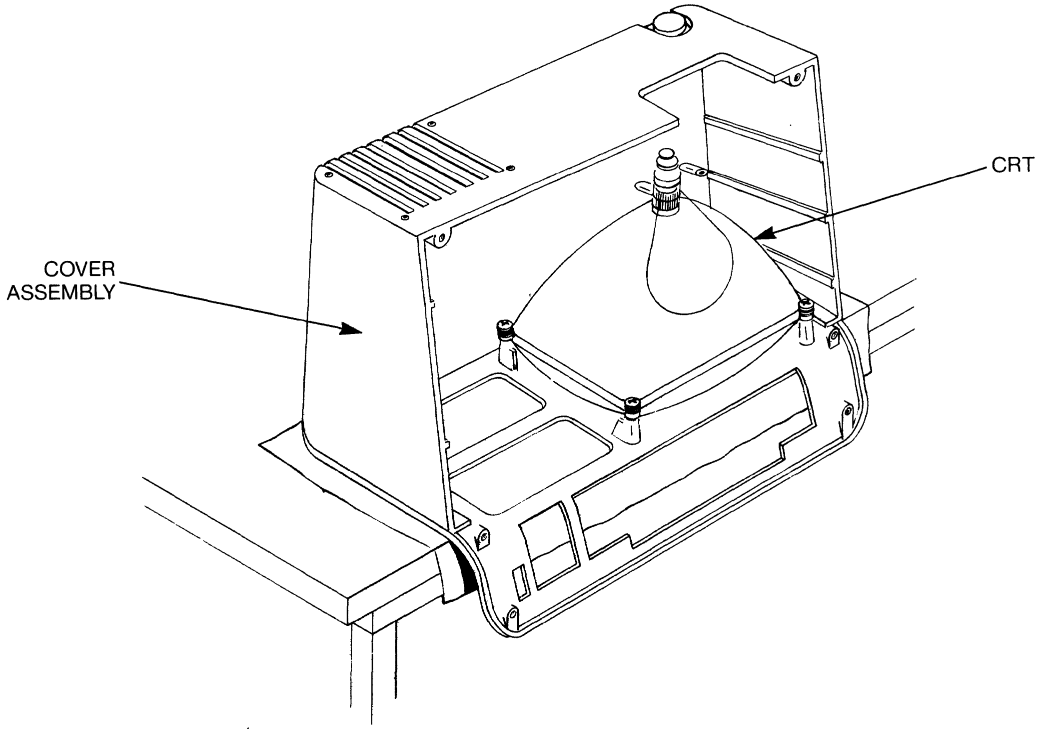


Figure 6-26

CAUTION
The following two steps are best performed by two people, one person to steady the Cover Assembly and the other person to remove the CRT. Handle the CRT yoke with care, to avoid breaking the tiny magnets glued to the outside of the yoke. The yoke is shown in Figure 6-24.

WARNING
BE SURE TO WEAR SAFETY GLASSES OR EQUIVALENT EYE PROTECTION WHEN PERFORMING THIS PROCEDURE. INSURE THAT NOTHING STRIKES THE CRT WHILE IT IS BEING REMOVED OR PLACED ON A WORK BENCH. IF THE CRT IS BROKEN, IT MAY IMPLODE AND CREATE A HAZARD BECAUSE OF FLYING GLASS.

13) Remove the four mounting screws which secure the CRT to the Cover Assembly.

14) The CRT is now free, and can be lifted carefully out of the cover.

B. To install the CRT and Video PC Board, proceed as follows:

1) Place the Cover Assembly on the edge of a work bench as shown in Figure 6-27. Use padding on the work bench to avoid scratching the cabinet.

CRT Installation

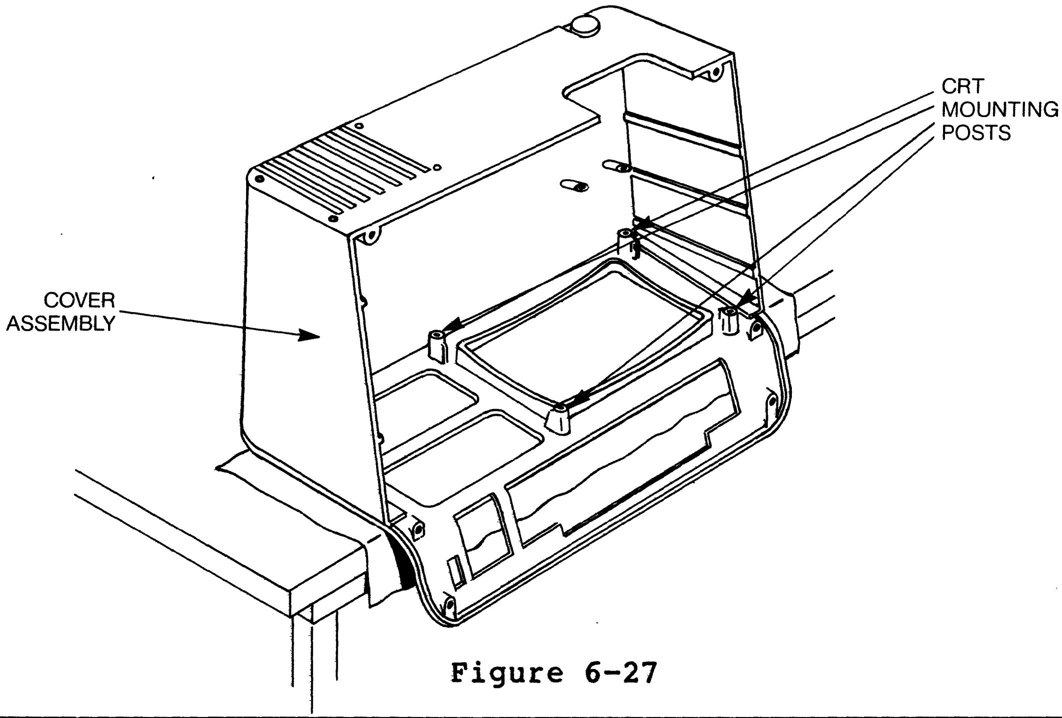


Figure 6-27

2) Find four 1/4 inch standoffs in the CRT mounting hardware. Place these standoffs on the four CRT mounting posts (see Figure 6-27).

CAUTION
The following two steps are best performed by two people, one person to steady the Cover Assembly and the other person to install the CRT. Handle the CRT yoke with care, to avoid breaking the tiny magnets glued to the outside of the yoke. The yoke is shown in Figure 6-24.

WARNING
BE SURE TO WEAR SAFETY GLASSES OR EQUIVALENT EYE PROTECTION WHEN PERFORMING THIS PROCEDURE. INSURE THAT NOTHING STRIKES THE CRT WHILE IT IS BEING INSTALLED. IF THE CRT IS BROKEN IT MAY IMPLODE AND CREATE A HAZARD BECAUSE OF FLYING GLASS.

3) Carefully place the CRT in the Cover Assembly so that the CRT high voltage connector is on the right hand side and the mounting tabs on the CRT rest on top of the 1/4 inch standoffs.

4) Slide a locking washer and a flat washer onto each of the four mounting screws. Drop these screws into the four mounting holes and start the screws by hand.

5) Adjust the position of the CRT so that it is centered on the mounting posts.

6) Tighten the four CRT mounting screws.

7) Rotate the Cover Assembly to the horizontal position, with the CRT facing to the side.

8) Lower the Video PC Board into the Cover Assembly and position it as shown in Figure 6-25.

9) Position the Video PC Board so that the hole in the brightness potentiometer presses lightly against the shaft of the brightness knob.

10) Rotate the brightness knob until the shaft clicks into position in the brightness potentiometer.

11) Press the brightness potentiometer and the brightness knob together so that the shaft is fully engaged in the potentiometer.

12) Install and tighten the five mounting screws for the Video PC Board (see Figure 6-25).

13) Connect the CRT yoke cable (see Figure 6-24). Align pin 1 on the cable connector with pin 1 on the PC board connector, and push the cable connector straight down onto the board.

14) Connect the CRT socket cable (See Figure 6-24). Align the seven pins on the CRT with the seven holes in the socket and press the socket onto the CRT.

15) Connect the high voltage lead (see Figure 6-24). Observe the two metal contacts on the high voltage connector. Hook these contacts into the hole in the side of the CRT, one contact at a time.

16) Place the Cover Assembly behind the Base Assembly as shown in Figure 6-13d.

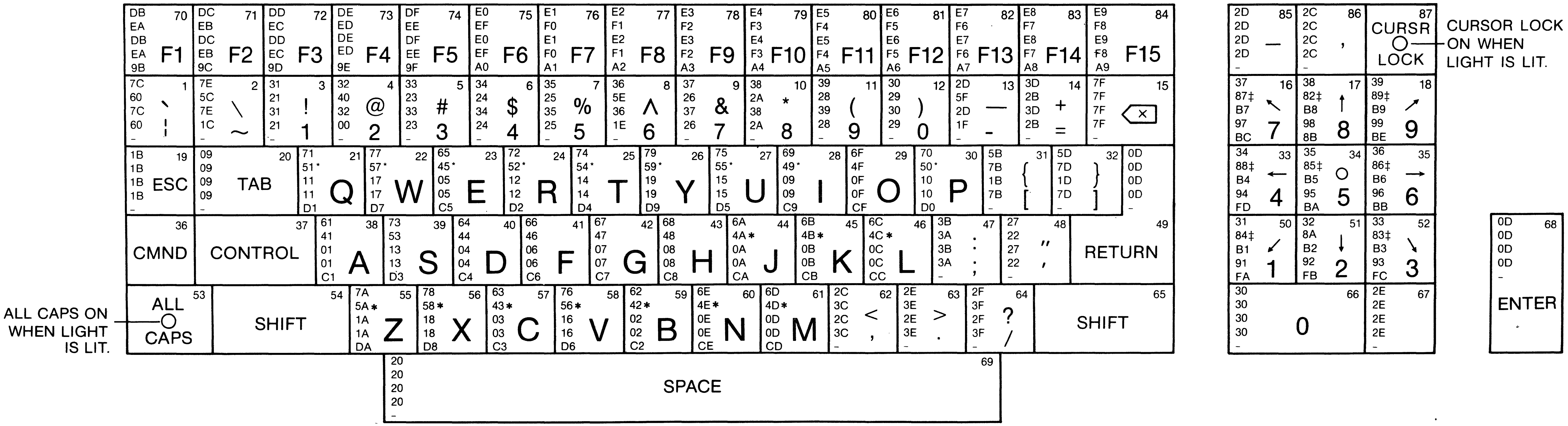
17) Install the video cable (see Figure 6-22 and 6-24). Align pin 1 on the cable connector with pin 1 on the PC board connector and push the cable connector straight down onto the board.

18) Install the fan cable (see Figure 6-16 and 6-23). Push the cable connectors straight onto the fan terminals.

19) Close the ADVANTAGE cabinet as described in Section 6.5.2.

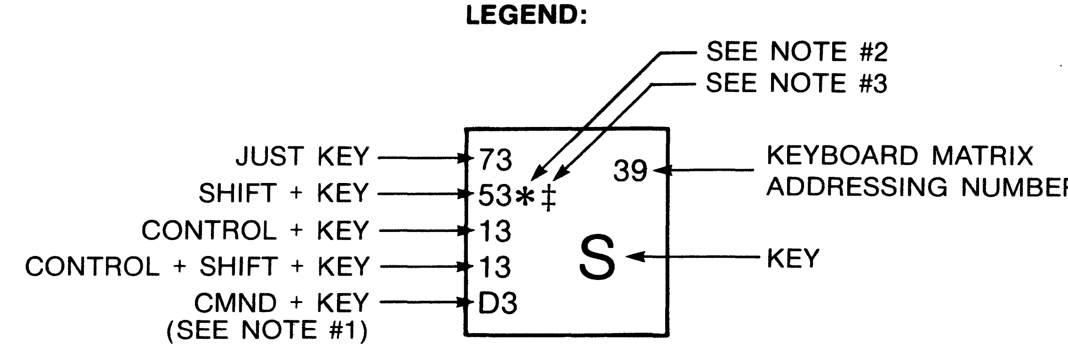
This appendix contains the following sections:

1. KEYBOARD PHYSICAL LAYOUT
2. KEYBOARD ASCII CODES BY KEY
3. KEYBOARD ASCII CODES IN NUMERIC ORDER
4. DECIMAL-HEX-BINARY-ASCII CONVERSION TABLE



NOTES:
1. A DASH (-) IN THE 5th LOCATION MEANS IGNORE CMND KEY IF DEPRESSED. ANYTHING ELSE MEANS IGNORE SHIFT AND/OR CONTROL KEY IF DEPRESSED.
2. ONLY THOSE KEYS WITH AN ASTERISK (*) ARE AFFECTED BY THE ALL CAPS KEY. WHEN ALL CAPS IS OFF THE CODES ARE AS SHOWN. WHEN ALL CAPS IS ON THE "JUST KEY" CODE CHANGES TO THE "SHIFT + KEY" CODE.
3. ONLY THOSE KEYS WITH ‡ ARE AFFECTED BY THE CURSOR LOCK KEY. WHEN CURSOR LOCK IS OFF THE CODES ARE AS SHOWN. WHEN CURSOR LOCK IS ON THE "JUST KEY" CODES CHANGE TO THE "SHIFT + KEY" CODES.

LEGEND:



1. KEYBOARD ASCII CODES BY KEY

KEY	NORMAL	SHIFT	CONTROL	CONTROL/ SHIFT	CMND
TAB	09	09	09	09	-
RETURN	0D	0D	0D	0D	-
ESC	1B	1B	1B	1B	-
Space	20	20	20	20	-
0)	30	29	30	29	-
1 !	31	21	31	21	-
2 @	32	40	32	00	-
3 #	33	23	33	23	-
4 $	34	24	34	24	-
5 %	35	25	35	25	-
6 ^	36	5E	36	1E	-
7 &	37	26	37	26	-
8 *	38	2A	38	2A	-
9 (	39	28	39	28	-
' "	27	22	27	22	-
, <	2C	3C	2C	3C	-
- _	2D	5F	2D	1F	-
>	2E	3E	2E	3E	-
/ ?	2F	3F	2F	3F	-
; :	3B	3A	3B	3A	-
= +	3D	2B	3D	2B	-
A	61	41	01	01	C1
B	62	42	02	02	C2
C	63	43	03	03	C3
D	64	44	04	04	C4
E	65	45	05	05	C5
F	66	46	06	06	C6
G	67	47	07	07	C7
H	68	48	08	08	C8
I	69	49	09	09	C9
J	6A	4A	0A	0A	CA
K	6B	4B	0B	0B	CB
L	6C	4C	0C	0C	CC
M	6D	4D	0D	0D	CD
N	6E	4E	0E	0E	CE
O	6F	4F	0F	0F	CF
P	70	50	10	10	D0
Q	71	51	11	11	D1
R	72	52	12	12	D2
S	73	53	13	13	D3
T	74	54	14	14	D4
U	75	55	15	15	D5
V	76	56	16	16	D6
W	77	57	17	17	D7

KEY	NORMAL	SHIFT	CONTROL	CONTROL/ SHIFT	CMND
X	78	58	18	18	D8
Y	79	59	19	19	D9
Z	7A	5A	1A	1A	DA
[{	5B	7B	1B	7B	-
] }	5D	7D	1D	7D	-
¦ `	7C	60	7C	60	-
~ \	7E	5C	7E	1C	-
⌫	7F	7F	7F	7F	-
,	2C	2C	2C	2C	-
-	2D	2D	2D	2D	-
.	2E	2E	2E	2E	-
0	30	30	30	30	-
1 ↙	31	84	91	91	FA
2 ↓	32	8A	92	92	FB
3 ↘	33	83	93	93	FC
4 ←	34	88	94	94	FD
5	35	85	95	95	BA
6 →	36	86	96	96	BB
7 ↖	37	87	97	97	BC
8 ↑	38	82	98	98	8D
9 ↗	39	89	99	99	BE
Enter	0D	0D	0D	0D	-
F1	DB	EA	DB	EA	9B
F2	DC	EB	DC	EB	9C
F3	DD	EC	DD	EC	9D
F4	DE	ED	DE	ED	9E
F5	DF	EE	DF	EE	9F
F6	E0	EF	E0	EF	A0
F7	E1	F0	E1	F0	A1
F8	E2	F1	E2	F1	A2
F9	E3	F2	E3	F2	A3
F10	E4	F3	E4	F3	A4
F11	E5	F4	E5	F4	A5
F12	E6	F5	E6	F5	A6
F13	E7	F6	E7	F6	A7
F14	E8	F7	E8	F7	A8
F15	E9	F8	E9	F8	A9

NOTES

* Single dash means ignore CMND key if pressed.

* The ALL CAPS key only affects the 26 alphabetic keys. When the light in the ALL CAPS key is on, the alphabetic keys produce the codes shown in the SHIFT column.

* The CURSOR LOCK key only affects keys 1 through 9 on the numeric keypad. When the light in the CURSOR LOCK key is on, keys 1 through 9 produce the codes shown in the SHIFT column.

3. KEYBOARD ASCII CODES IN NUMERIC ORDER

ASCII CODE (HEX)	KEY(S)
00	CTL @
01	CTL a
	CTL A
02	CTL b
	CTL B
03	CTL c
	CTL C
04	CTL d
	CTL D
05	CTL e
	CTL E
06	CTL f
	CTL F
07	CTL g
	CTL G
08	CTL h
	CTL H
09	CTL i
	CTL I
	TAB
	SHIFT TAB
	CTL TAB
	CTL-SHIFT TAB
0A	CTL j
	CTL J
0B	CTL k
	CTL K
0C	CTL l
	CTL L
0D	CTL m
	CTL M
	RETURN
	SHIFT RETURN
	CTL RETURN
	CTL-SHIFT RETURN
	ENTER
	SHIFT ENTER
	CTL ENTER
	CTL-SHIFT ENTER
0E	CTL n
	CTL N
0F	CTL o
	CTL O
10	CTL p
	CTL P
11	CTL q
	CTL Q
12	CTL r
	CTL R
13	CTL s
	CTL S
14	CTL t
	CTL T
15	CTL u
	CTL U
16	CTL v
	CTL V
17	CTL w
	CTL W
18	CTL x
	CTL X
19	CTL y
	CTL Y
1A	CTL z
	CTL Z
1B	ESC
	SHIFT ESC
	CTL ESC
	CTL-SHIFT ESC
	CTL [
1C	CTL \
1D	CTL]
1E	CTL ^
1F	CTL _
20	SPACE
	SHIFT SPACE
	CTL SPACE
	CTL-SHIFT SPACE
21	!
	CTL !
22	"
	CTL "
23	#
	CTL #
24	$
	CTL $
25	%
	CTL %
26	&
	CTL &

ASCII CODE (HEX)	KEY(S)
27	'
	CTL '
28	(
	CTL (
29	)
	CTL)
2A	*
	CTL *
2B	+
	CTL +
2C	,
	CTL ,
2D	-
	CTL -
2E	.
	CTL .
2F	/
	CTL /
30	0
	SHIFT 0 (numeric pad)
	CTL 0
	CTL-SHIFT 0 (numeric pad)
31	1
	CTL 1 (typewriter key)
32	2
	CTL 2 (typewriter key)
33	3
	CTL 3 (typewriter key)
34	4
	CTL 4 (typewriter key)
35	5
	CTL 5 (typewriter key)
36	6
	CTL 6 (typewriter key)
37	7
	CTL 7 (typewriter key)
38	8
	CTL 8 (typewriter key)
39	9
	CTL 9 (typewriter key)
3A	:
	CTL :
3B	;
	CTL ;
3C	<
	CTL <

ASCII CODE (HEX)	KEY(S)
3D	=
	CTL =
3E	>
	CTL >
3F	?
	CTL ?
40	@
41	A
42	B
43	C
44	D
45	E
46	F
47	G
48	H
49	I
4A	J
4B	K
4C	L
4D	M
4E	N
4F	O
50	P
51	Q
52	R
53	S
54	T
55	U
56	V
57	W
58	X
59	Y
5A	Z
5B	[
5C	\
5D	]
5E	^
5F	_
60	`
	CTL `
61	a
62	b
63	c
64	d
65	e
66	f

ASCII CODE (HEX)	KEY(S)
67	g
68	h
69	i
6A	j
6B	k
6C	l
6D	m
6E	n
6F	o
70	p
71	q
72	r
73	s
74	t
75	u
76	v
77	w
78	x
79	y
7A	z
7B	{
	CTL {
7C	¦
	CTL ¦
7D	}
	CTL }
7E	~
	CTL ~
7F	⌫
	SHIFT ⌫
	CTL ⌫
	CTL-SHIFT ⌫
80	
81	
82	↑
83	↘
84	↙
85	∘
86	→
87	↖
88	←
89	↗
8A	↓
8B	
8C	
8D	
8E	

ASCII CODE (HEX)	KEY(S)
8F	
90	
91	CTL ↙
92	CTL ↓
93	CTL ↘
94	CTL ←
95	CTL ∘
96	CTL →
97	CTL ↖
98	CTL ↑
99	CTL ↗
9A	
9B	CMND F1
9C	CMND F2
9D	CMND F3
9E	CMND F4
9F	CMND F5
A0	CMND F6
A1	CMND F7
A2	CMND F8
A3	CMND F9
A4	CMND F10
A5	CMND F11
A6	CMND F12
A7	CMND F13
A8	CMND F14
A9	CMND F15
AA	
AB	
AC	
AD	
AE	
AF	
B0	
B1	CTL 1 (numeric pad)
B2	CTL 2 (numeric pad)
B3	CTL 3 (numeric pad)
B4	CTL 4 (numeric pad)
B5	CTL 5 (numeric pad)
B6	CTL 6 (numeric pad)
B7	CTL 7 (numeric pad)
B8	CTL 8 (numeric pad)
B9	CTL 9 (numeric pad)
BA	CMND 5 (numeric pad)
BB	CMND 6 (numeric pad)
BC	CMND 7 (numeria pad)
BD	CMND 8 (numeric pad)

ASCII CODE (HEX)	KEY(S)
BE	CMND 9 (numeric pad)
BF	
C0	
C1	CMND a
C2	CMND b
C3	CMND c
C4	CMND d
C5	CMND e
C6	CMND f
C7	CMND g
C8	CMND h
C9	CMND i
CA	CMND j
CB	CMND k
CC	CMND l
CD	CMND m
CE	CMND n
CF	CMND o
D0	CMND p
D1	CMND q
D2	CMND r
D3	CMND s
D4	CMND t
D5	CMND u
D6	CMND v
D7	CMND w
D8	CMND x
D9	CMND y
DA	CMND z
DB	F1
	CTL F1
DC	F2
	CTL F2
DD	F3
	CTL F3
DE	F4
	CTL F4
DF	F5
	CTL F5
E0	F6
	CTL F6
E1	F7
	CTL F7
E2	F8
	CTL F8
E3	F9
	CTL F9
E4	F10
	CTL F10

ASCII CODE (HEX)	KEY(S)
E5	F11
	CTL F11
E6	F12
	CTL F12
E7	F13
	CTL F13
E8	F14
	CTL F14
E9	F15
	CTL F15
EA	SHIFT F1
	CTL-SHIFT F1
EB	SHIFT F2
	CTL-SHIFT F2
EC	SHIFT F3
	CTL-SHIFT F3
ED	SHIFT F4
	CTL-SHIFT F4
EE	SHIFT F5
	CTL-SHIFT F5
EF	SHIFT F6
	CTL-SHIFT F6
F0	SHIFT F7
	CTL-SHIFT F7
F1	SHIFT F8
	CTL-SHIFT F8
F2	SHIFT F9
	CTL-SHIFT F9
F3	SHIFT F10
	CTL-SHIFT F10
F4	SHIFT F11
	CTL-SHIFT F11
F5	SHIFT F12
	CTL-SHIFT F12
F6	SHIFT F13
	CTL-SHIFT F13
F7	SHIFT 14
	CTL-SHIFT F14
F8	SHIFT 15
	CTL-SHIFT 15
F9	
FA	CMND 1 (numeric pad)
FB	CMND 2 (numeric pad)
FC	CMND 3 (numeric pad)
FD	CMND 4 (numeric pad)
FE	
FF	

DECIMAL-ASCII-HEX-BINARY CONVERSION TABLE

The following table is intended to ease the task of conversion between the various numeric representations commonly used in programming, as well as between numbers (of any kind) and the ASCII character code.

Note that the ASCII character set only goes as far as decimal 127 (7FH, 01111111 B). Also, many "characters" in ASCII are nonprinting CONTROL CHARACTERS. Whenever a code corresponds to a printable character, that will be given. In the case of control characters, a description or name for the special character will be given in parentheses.

DECIMAL	HEX	BINARY	ASCII
0	00H	00000000	(NUL)
1	01H	00000001	(CONTROL-A)
2	02H	00000010	(CONTROL-B)
3	03H	00000011	(CONTROL-C)
4	04H	00000100	(CONTROL-D)
5	05H	00000101	(CONTROL-E)
6	06H	00000110	(CONTROL-F)
7	07H	00000111	(CONTROL-G, RINGS BELL)
8	08H	00001000	(CONTROL-H, BACKSPACE)
9	09H	00001001	(CONTROL-I, TAB)
10	0AH	00001010	(CONTROL-J, LINEFEED)
11	0BH	00001011	(CONTROL-K)
12	0CH	00001100	(CONTROL-L, FORMFEED)
13	0DH	00001101	(CONTROL-M, CARRIAGE RETURN)
14	0EH	00001110	(CONTROL-N)
15	0FH	00001111	(CONTROL-O)
16	10H	00010000	(CONTROL-P)
17	11H	00010001	(CONTROL-Q)
18	12H	00010010	(CONTROL-R)
19	13H	00010011	(CONTROL-S)
20	14H	00010100	(CONTROL-T)
21	15H	00010101	(CONTROL-U)
22	16H	00010110	(CONTROL-V)
23	17H	00010111	(CONTROL-W)
24	18H	00011000	(CONTROL-X)
25	19H	00011001	(CONTROL-Y)
26	1AH	00011010	(CONTROL-Z)
27	1BH	00011011	(ESCAPE)
28	1CH	00011100	(NON-PRINTING)
29	1DH	00011101	(NON-PRINTING)
30	1EH	00011110	(NON-PRINTING)
31	1FH	00011111	(NON-PRINTING)
32	20H	00100000	(SPACE)
33	21H	00100001	!
34	22H	00100010	"
35	23H	00100011	#
36	24H	00100100	$
37	25H	00100101	%
38	26H	00100110	&
39	27H	00100111	'

Conversion Table continued

DECIMAL	HEX	BINARY	ASCII
40	28H	00101000	(
41	29H	00101001	)
42	2AH	00101010	*
43	2BH	00101011	+
44	2CH	00101100	,
45	2DH	00101101	-
46	2EH	00101110	.
47	2FH	00101111	/
48	30H	00110000	0
49	31H	00110001	1
50	32H	00110010	2
51	33H	00110011	3
52	34H	00110100	4
53	35H	00110101	5
54	36H	00110110	6
55	37H	00110111	7
56	38H	00111000	8
57	39H	00111001	9
58	3AH	00111010	:
59	3BH	00111011	;
60	3CH	00111100	<
61	3DH	00111101	=
62	3EH	00111110	>
63	3FH	00111111	?
64	40H	01000000	@
65	41H	01000001	A
66	42H	01000010	B
67	43H	01000011	C
68	44H	01000100	D
69	45H	01000101	E
70	46H	01000110	F
71	47H	01000111	G
72	48H	01001000	H
73	49H	01001001	I
74	4AH	01001010	J
75	4BH	01001011	K
76	4CH	01001100	L
77	4DH	01001101	M
78	4EH	01001110	N
79	4FH	01001111	O
80	50H	01010000	P
81	51H	01010001	Q
82	52H	01010010	R
83	53H	01010011	S
84	54H	01010100	T
85	55H	01010101	U
86	56H	01010110	V
87	57H	01010111	W
88	58H	01011000	X
89	59H	01011001	Y
90	5AH	01011010	Z
91	5BH	01011011	[
92	5CH	01011100	\
93	5DH	01011101	]

Conversion Table continued

DECIMAL	HEX	BINARY	ASCII
94	5EH	01011110	↑ OR ^
95	5FH	01011111	_
96	60H	01100000	`
97	61H	01100001	a
98	62H	01100010	b
99	63H	01100011	c
100	64H	01100100	d
101	65H	01100101	e
102	66H	01100110	f
103	67H	01100111	g
104	68H	01101000	h
105	69H	01101001	i
106	6AH	01101010	j
107	6BH	01101011	k
108	6CH	01101100	l
109	6DH	01101101	m
110	6EH	01101110	n
111	6FH	01101111	o
112	70H	01110000	p
113	71H	01110001	q
114	72H	01110010	r
115	73H	01110011	s
116	74H	01110100	t
117	75H	01110101	u
118	76H	01110110	v
119	77H	01110111	w
120	78H	01111000	x
121	79H	01111001	y
122	7AH	01111010	z
123	7BH	01111011	{
124	7CH	01111100	|
125	7DH	01111101	}
126	7EH	01111110	~
127	7FH	01111111	(DELETE, RUB OUT)
128	80H	10000000	
129	81H	10000001	
130	82H	10000010	
131	83H	10000011	
132	84H	10000100	
133	85H	10000101	
134	86H	10000110	
135	87H	10000111	
136	88H	10001000	
137	89H	10001001	
138	8AH	10001010	
139	8BH	10001011	
140	8CH	10001100	
141	8DH	10001101	
142	8EH	10001110	
143	8FH	10001111	
144	90H	10010000	
145	91H	10010001	
146	92H	10010010	
147	93H	10010011	

Conversion Table continued

DECIMAL	HEX	BINARY	ASCII
148	94H	10010100	
149	95H	10010101	
150	96H	10010110	
151	97H	10010111	
152	98H	10011000	
153	99H	10011001	
154	9AH	10011010	
155	9BH	10011011	
156	9CH	10011100	
157	9DH	10011101	
158	9EH	10011110	
159	9FH	10011111	
160	A0H	10100000	
161	A1H	10100001	
162	A2H	10100010	
163	A3H	10100011	
164	A4H	10100100	
165	A5H	10100101	
166	A6H	10100110	
167	A7H	10100111	
168	A8H	10101000	
169	A9H	10101001	
170	AAH	10101010	
171	ABH	10101011	
172	ACH	10101100	
173	ADH	10101101	
174	AEH	10101110	
175	AFH	10101111	
176	B0H	10110000	
177	B1H	10110001	
178	B2H	10110010	
179	B3H	10110011	
180	B4H	10110100	
181	B5H	10110101	
182	B6H	10110110	
183	B7H	10110111	
184	B8H	10111000	
185	B9H	10111001	
186	BAH	10111010	
187	BBH	10111011	
188	BCH	10111100	
189	BDH	10111101	
190	BEH	10111110	
191	BFH	10111111	
192	C0H	11000000	
193	C1H	11000001	
194	C2H	11000010	
195	C3H	11000011	
196	C4H	11000100	
197	C5H	11000101	
198	C6H	11000110	
199	C7H	11000111	
200	C8H	11001000	
201	C9H	11001001	

Conversion Table continued

DECIMAL	HEX	BINARY	ASCII
202	CAH	11001010	
203	CBH	11001011	
204	CCH	11001100	
205	CDH	11001101	
206	CEH	11001110	
207	CFH	11001111	
208	D0H	11010000	
209	D1H	11010001	
210	D2H	11010010	
211	D3H	11010011	
212	D4H	11010100	
213	D5H	11010101	
214	D6H	11010110	
215	D7H	11010111	
216	D8H	11011000	
217	D9H	11011001	
218	DAH	11011010	
219	DBH	11011011	
220	DCH	11011100	
221	DDH	11011101	
222	DEH	11011110	
223	DFH	11011111	
224	E0H	11100000	
225	E1H	11100001	
226	E2H	11100010	
227	E3H	11100011	
228	E4H	11100100	
229	E5H	11100101	
230	E6H	11100110	
231	E7H	11100111	
232	E8H	11101000	
233	E9H	11101001	
234	EAH	11101010	
235	EBH	11101011	
236	ECH	11101100	
237	EDH	11101101	
238	EEH	11101110	
239	EFH	11101111	
240	F0H	11110000	
241	F1H	11110001	
242	F2H	11110010	
243	F3H	11110011	
244	F4H	11110100	
245	F5H	11110101	
246	F6H	11110110	
247	F7H	11110111	
248	F8H	11111000	
249	F9H	11111001	
250	FAH	11111010	
251	FBH	11111011	
252	FCH	11111100	
253	FDH	11111101	
254	FEH	11111110	
255	FFH	11111111	

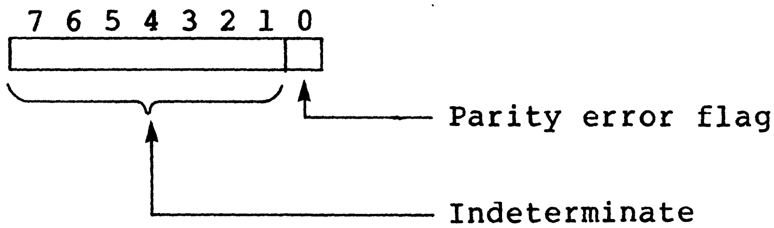
This appendix lists all I/O addresses that can be used in Z80 processor INPUT or OUTPUT instructions when programming the ADVANTAGE computer. The addresses are listed in numeric order. More detailed programming information can be found in Chapter 3.

I/O ADDRESS SUMMARY

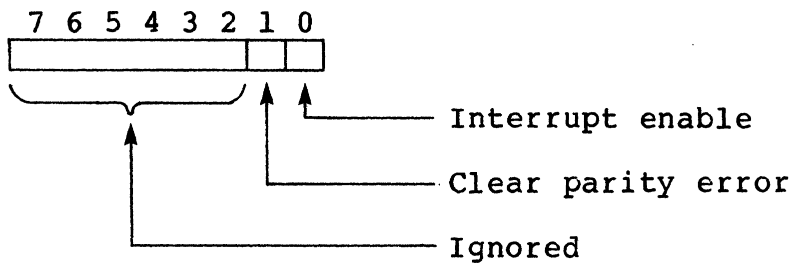
Hexadecimal Address	Operation	Description
00 - 0F	INPUT/OUTPUT	Access I/O board in slot 6. The first digit of these addresses defines the board slot being accessed. The second digit has a meaning defined by the type of board in that slot. Refer to Section 3.9, 3.10, or 3.11.
10 - 1F	INPUT/OUTPUT	Access I/O board in slot 5. The first digit of these addresses defines the board slot being accessed. The second digit has a meaning defined by the type of board in that slot. Refer to Section 3.9, 3.10, or 3.11.
20 - 2F	INPUT/OUTPUT	Access I/O board in slot 4. the first digit of these addresses defines the board slot being accessed. The second digit has a meaning defined by the type of board in that slot. Refer to Section 3.9, 3.10, or 3.11.

30 - 3F	INPUT/OUTPUT	Access I/O board in slot 3. The first digit of these addresses defines the board slot being accessed. The second digit has a meaning defined by the type of board in that slot. Refer to Section 3.9, 3.10, or 3.11.
40 - 4F	INPUT/OUTPUT	Access I/O board in slot 2. The first digit of these addresses defines the board slot being accessed. The second digit has a meaning defined by the type of board in that slot. Refer to Section 3.9, 3.10, or 3.11.
50 - 5F	INPUT/OUTPUT	Access I/O board in slot 1. The first digit of these addresses defines the board slot being accessed. The second digit has a meaning defined by the type of board in that slot. Refer to Section 3.9, 3.10, or 3.11.
60	INPUT	Input Main RAM Parity Status byte. The byte format is shown below.
60	OUTPUT	Output Main RAM Parity Control byte. The byte format is shown below.
61 - 6F	INPUT/OUTPUT	Same as I/O address 60.

MEMORY PARITY STATUS BYTE



MEMORY PARITY CONTROL BYTE



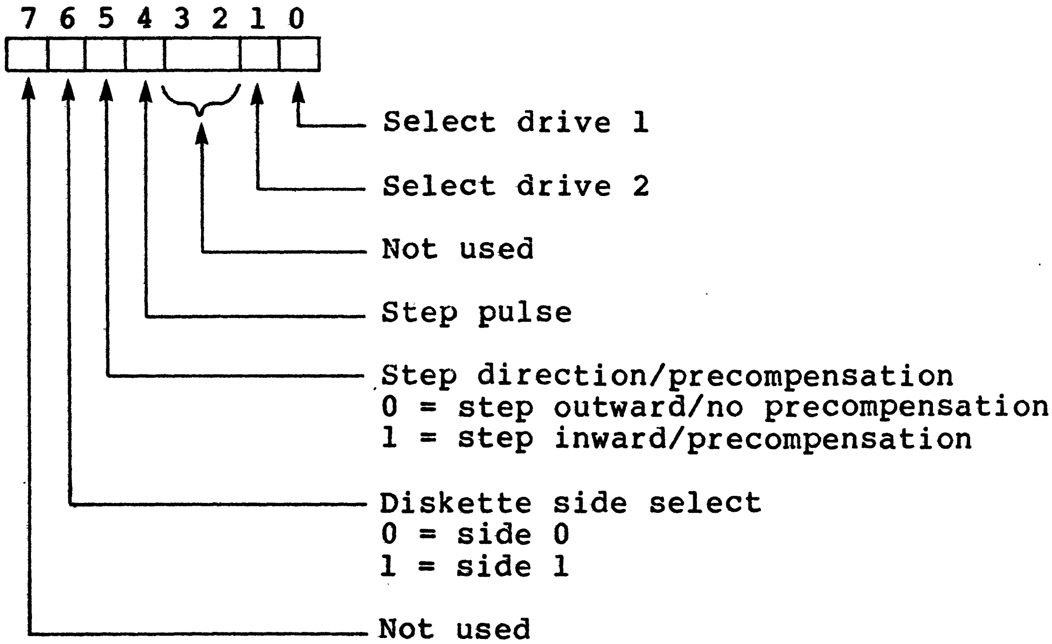
Hexadecimal Address	Operation	Description
70	INPUT only	Input the ID code for board in slot 6. The ID codes are shown below.
71	INPUT only	Input the ID code for board in slot 5. The ID codes are shown below.
72	INPUT only	Input the ID code for board in slot 4. The ID codes are shown below.
73	INPUT only	Input the ID code for board in slot 3. The ID codes are shown below.
74	INPUT only	Input the ID code for board in slot 2. The ID codes are shown below.
75	INPUT only	Input the ID code for board in slot 1. The ID codes are shown below.
76		Unused. Inputting from this address returns all ones.
77		Unused. Inputting from this address returns all ones.
78 - 7D	INPUT only	Same as I/O addresses 70 through 75 respectively.
7E		Unused. Inputting from this address returns all ones.
7F		Unused. Inputting from this address returns all ones.

<u>ID CODES:</u>

7F - Floating Point Board
F7 - SIO Board
BE - Hard Disk Controller Board
DB - PIO Board
FF - No board installed.

Hexadecimal Address	Operation	Description
80	INPUT	Input a data byte from the selected disk drive.
80	OUTPUT	Output a data byte to the selected disk drive.
81	INPUT	Input a sync byte from the selected disk drive.
81	OUTPUT	Load the Drive Control register. The format of the register is shown below.
82	INPUT	Clear Disk Read flag.
82	OUTPUT	Set Disk Read flag.
83	INPUT	Produce a 'beep' sound.
83	OUTPUT	Set Disk Write flag.
84 - 8F		Same as I/O Addresses 80 through 83 respectively.

DRIVE CONTROL REGISTER



Hexadecimal Address	Operation	Description
90	OUTPUT only	Load Start Scan register. Inputting from this address returns indeterminate data and loads indeterminate data into the Start Scan register.
91 - 9F		Same as I/O address 90.
A0 - A3	OUTPUT only	Memory Mapping registers 0 through 3 respectively. The format of the output byte is shown below. Inputting from any of these addresses returns indeterminate data and loads indeterminate data into the corresponding Memory Mapping register.
A4 - AF		Same as I/O addresses A0 through A3 respectively.

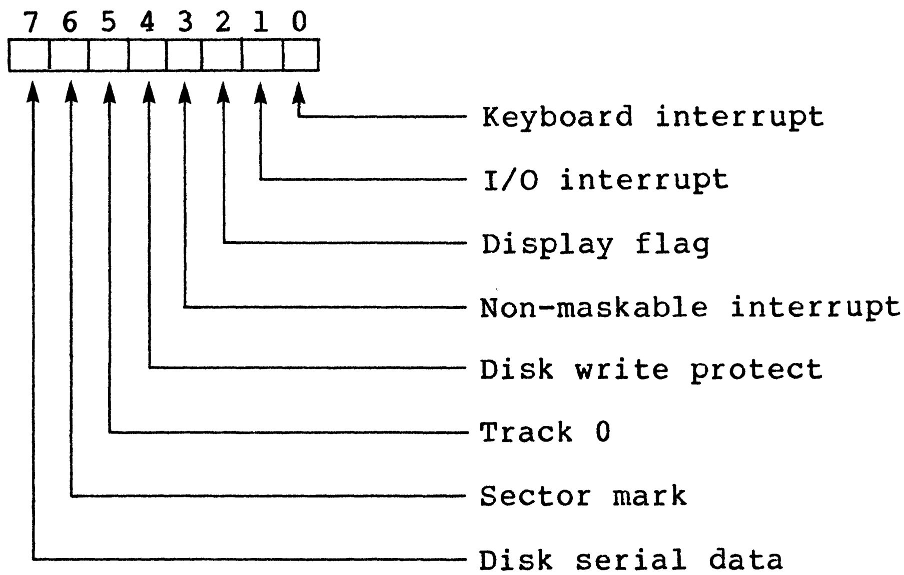
MAPPING REGISTER OUTPUT BYTE

7 6 5 4 3 2 1 0

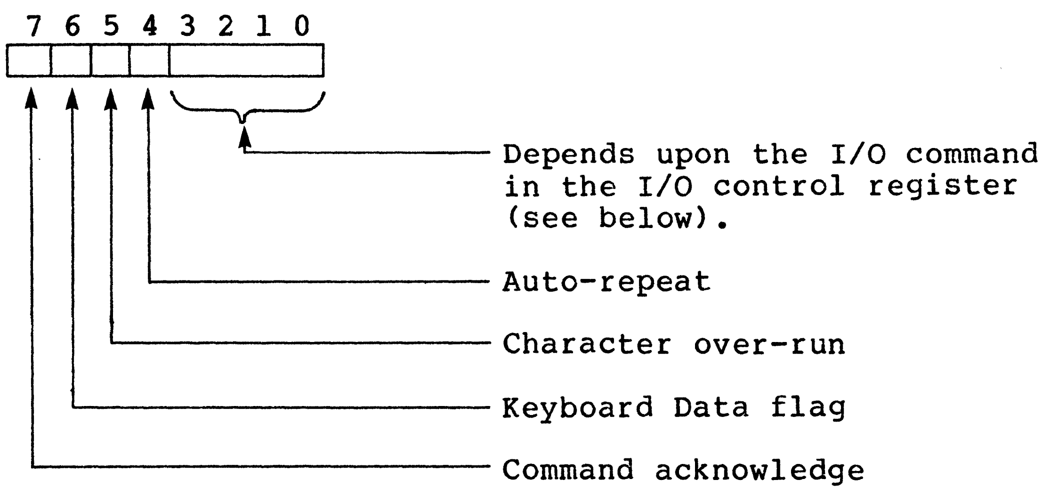
0 X X X X N N N	Main RAM page NNN
1 X X X X 0 0 N	Display RAM, N=0 for page 8 N=1 for page 9
1 X X X X 1 X X	Boot PROM

Hexadecimal Address	Operation	Description
B0	INPUT/OUTPUT	Clear Display flag. Inputting from this address returns indeterminate data.
B1 - BF		Same as I/O address B0.
C0 - CF	INPUT/OUTPUT	Clear non-maskable interrupt to Z80 processor. Inputting from this address returns indeterminate data.
D0	INPUT only	Input from I/O Status Register 2. The format of this register is shown below.
D1 - DF		Same as I/O address D0
E0	INPUT only	Input from I/O Status Register 1. The format of this register is shown below.
E1 - EF		Same as I/O address E0.
F0	OUTPUT only	Output to I/O Control Register. The format of this register is shown below. Inputting from this address returns indeterminate data and loads indeterminate data into the I/O Control register.
F1 - FF		Same as I/O address F0.

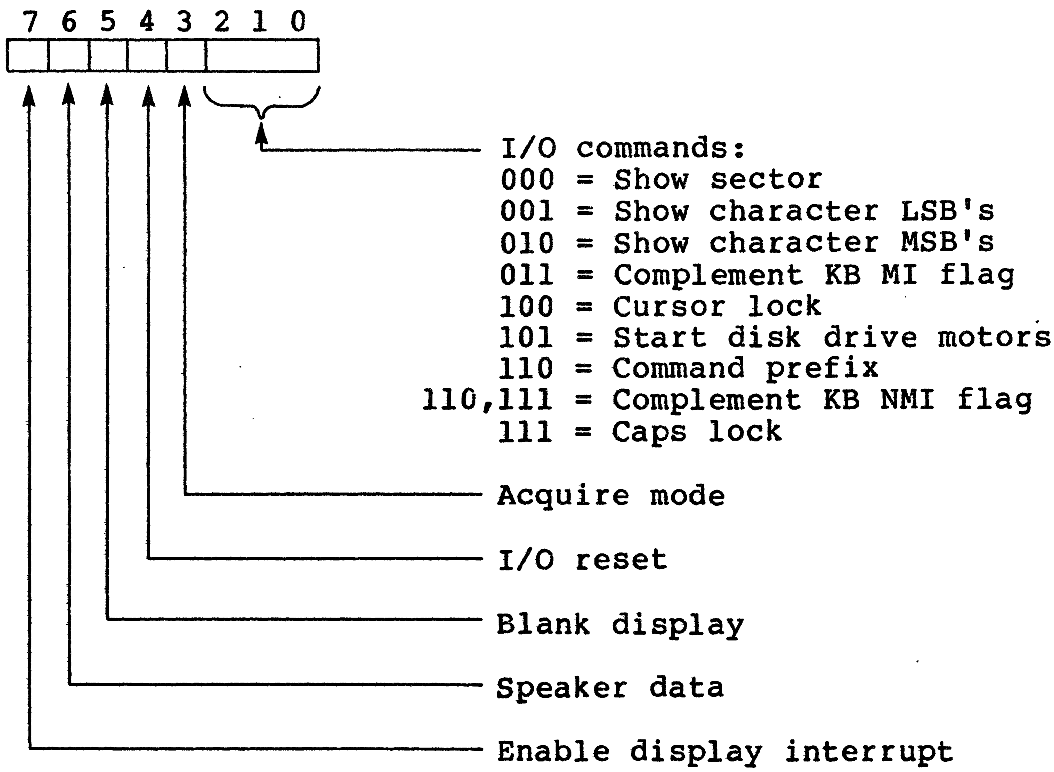
I/O STATUS REGISTER 1



I/O STATUS REGISTER 2



I/O CONTROL REGISTER



This appendix lists the jumpers on the printed circuit boards that allow the connection of certain signals to be modified. Some signals are open-circuited by removing a jumper plug from the board. Other signals are re-routed by cutting a trace and soldering a wire to the board.

1. MAIN PC BOARD JUMPERS

Jumper Number	Board Location	Description
W1	1L	Determines the polarity of the vertical sync pulse going to the Video Monitor. The PC board trace makes the connection for positive sync pulses. The alternate connection produces negative sync pulses.
W2	1L	Determines the polarity of the video data going to the Video Monitor. The PC board trace causes one bits to produce positive data pulses. The alternate connection causes one bits to produce negative data pulses.
W3	1L	Determines the polarity of the horizontal sync pulse going to the Video Monitor. The PC board trace makes the connection for negative sync pulses. The alternate connection produces positive sync pulses.
W4	9E	If parity errors are allowed to generate interrupts (see Section 3.2.2) this jumper determines whether they will be maskable or non-maskable. The PC board trace makes the connection for maskable interrupts. The alternate connection produces non-maskable interrupts.

Jumper Number	Board Location	Description
W5	18C	Determines the type of integrated circuit used for the Auxiliary Processor at board location 18C. The PC board trace is used with an 8035 processor. The alternate connection is used with an 8048 or 8049 processor.
W6	11K	This jumper plug determines the type of integrated circuit used for the Boot PROM at board location 11K. The plug isinserted in the position farthest from the PROM if the PROM is a type 2716. The plug is inserted in the position closest to the PROM if it is a type 2732.
W7	17C	This jumper plug is removed for testing purposes. It disconnects the sector pulse signal from the Auxiliary Processor.
W8	17D	When this jumper plug is inserted, it allows the simultaneous depression of four keys on the keyboard to generate a non-maskable interrupt (see Section 2.1.4). When the jumper plug is removed, the interrupt is not generated.
W9 W10 W11	16K 16K 16K	These jumper plugs are removed for testing purposes. They disconnect the output of the +5V regulator.
W12	10M	This jumper plug is removed for testing purposes. It disconnects +12V power from the I/O interface connectors.
W13	10M	This jumper plug is removed for testing purposes. It disconnects input power from the -5V regulator.
W14	18K	This jumper plug is removed for testing purposes. It disconnects +12V power from the Speaker Circuit and from the Disk Data Separation Circuit.

2. SIO BOARD JUMPERS

JUMPER NUMBER	BOARD LOCATION	DESCRIPTION
W1	4A	Allows the "buffer full" signal to be wired to one of two pins on the device interface connector (see Section 3.9.7). The PC board trace is connected to pin 20. The trace may be cut and a wire soldered to change the connection to pin 19.

3. PIO BOARD JUMPERS

JUMPER NUMBER	BOARD LOCATION	DESCRIPTION
J1 J2 J3	3A 3A 3A	These jumpers improve the ground connection to the output device by connecting ground to pins 13,14, and 15 of the output device cable. One or more of these jumpers may be disconnected so that the corresponding pin(s) may be used to supply power to the output device.
J4 J5 J6	3D 3D 3D	These jumpers improve the ground connection to the input device by connecting ground to pins 13,14, and 15 of the input device cable. One or more of these jumpers may be disconnected so that the corresponding pin(s) may be used to supply power to the input device.

DISK SUBSYSTEM TEST

CRC Error	Cyclic Redundance check: drive may not be working; media may be bad; possible programming error.
No Index Pulse Error	No index pulses are coming from selected drive.
Read Error	Data read was not as expected: read/write circuitry failure or bad media.
Seek Error	Goes to track and reads data; from data determines that track is wrong. If accompanied by read error, may indicate bad media. Seek stepping motor may be bad. Diskette may be stuck at a single track if usual accessing clicks are not audible.
No Sync Byte Found Error	Errors may indicate improper read, write, or bad media.
Verify Compare Error	Indicates probable write error: write circuitry may be defective, or media may be bad.
Write Protect Error	Write protect switch may be bad.

DISPLAY TEST

No error messages - visual assessment only

EXECUTABLE MEMORY TEST

Cursor fails to flash approximately every 5 sec	Test has died at section and pass indicated.
! after PASS counter	Defective Memory (see below).
? in RAM MATRIX	Defective Memory. Indicates location of bad RAM chip.

KEYBOARD TEST

Long beep - audio message Not to be confused with short beep indicating successful completion of row	Defective key or wrong key pressed
? under key entry on screen	Defective key or wrong key pressed. Next to (?) displays actual character entered.

SIO TEST

Bad Character	Character received does not agree with character transmitted. Insure that SIO is in the standard configuration (see Section 3.9.1)
Detected a board in port- but it looks like home is there	Another board in machine with wrong address set on it; or possible problem with shorting, or open wires on data bus to that port.
SIO board in port - won't get ready to receive at character____(between 0 and 255)	Could be SIO board is bad; USART bad; receive circuitry incorrect; strobe signal bad.
SIO board in port - won't get ready to transmit at character ____(between 0 and 255)	If bad character message also occurs could be bad SIO circuitry, or test jumper wired wrong.
There were ___bad characters (maximum of 255)	

VIDEO MEMORY TEST

! after PASS counter	Defective memory (see below).
? in RAM MATRIX	Defective memory: indicates suspected location of bad RAM.
Stationary vertical bar	Defective memory

MAIN PARTS LIST

ITEM	P/N	QTY	DESCRIPTION	REF
1	00113	1	SIO PCB ASSY	
2	38064	3	SCREWLOCK ASSEMBLY, FEMALE (SET CONTAINS 2 JACK SCREWS, 2 SPLIT WASHING & 2 HEX NUTS)	1 EA FOR SIO 2 EA FOR PIO
3	38065-04	6	WASHER, FLAT, #4	(2)SIO, (4)PIO
4	38065-10	6	WASHER, FLAT, #10	BASE TO COVER
5	38075-10	6	WASHER, #10 SPLIT LOCK	BASE TO COVER
6	38091-10	6	SCREW, MACHINE, #10-32 X 5/8 PAN HEAD, XREC	BASE TO COVER
7	49002	2	QUAD DISK DRIVES	
8	00353	1	SHIELD, DISK DRIVE	
9	00333	1	CABLE, DISK DRIVE POWER	
10	00335	1	CABLE, DISK DRIVE SIGNAL	
11	38065-06	10	WASHER, #6 FLAT	DRIVE TO BRKT (8)
12				SHIELD TO BRKT (2)
13	38036-	10	SCREW, #6-32 X 3/8",PHMS XREC, SEMS	
14	00102	1	MAIN PCB ASSEMBLY	
15	00347	1	PLATE, DRIVE CABLE	
16	00352	1	BRACKET, DISK DRIVES MOUNTING	
17	00355	1	BASE, FAB	
18	00356	1	PLATE, I/O MOUNTING	
19	00357	1	KEYBOARD	
20	00358	1	CABLE, KEYBOARD	
21	00364	1	LABEL, MODEL & SERIAL NUMBER	
22	00370	4	FEET, RUBBER	BASE(4),
23	38007	4	WASHER, FLAT #10	XRMR-BASE
24	38009	2	NUT,HEX, #4-40	HOLE COVER
25	38083-08	4	SCREW,MACHINE, PH, BLACK, #4-40 X 1/2"	I/O PLATE-COVER
26	38065-08	4	WASHER, FLAT, #8	DRV BKT-BASE
27	38071	4	SCREW,BLUNT PT, #6 X 3/8"	MAINT PCB-BASE(2) RECT BKT-BASE(2)
28	38074-06	3	WASHER,LOCK,BLACK, #6	I/O PLATE (2) PWR PANEL (1)
29	38075-04	4	WASHER,LOCK,SPLIT,#4	I/O PLATE COVER
30	38075-06	2	WASHER,LOCK,SPLIT,#6	RECT. BKT-BASE
31	38075-10	4	WASHER,LOCK,SPLIT,#10	XRMR-BASE

MAIN PARTS LIST (continued)

ITEM	P/N	QTY	DESCRIPTION	REF
32	38082-08	3	SCREW,MACHINE,BLACK #6-32 X 1/2"	I/O PLATE (2) PWR PANEL (1)
33			#4-40 X 1/2"	
34	38084-06	3	WASHER, FLAT, BLACK, #6	I/O PLATE(2) PWR PANEL(1)
35	38089-10	4	SCREW,MACHINE, #8 X 5/8"	DRV BKT-BASE
36	38091-08	4	SCREW,MACHINE,#10-32 X 1/2"	XFMR-BASE
37	77045	4	BRACKET,MOUNTING,CABLE TIE, 3/4 SQ. ADHESIVE BACK	
38	77046	1	CABLE TIE, 5.5" LONG	
39	00336	1	CABLE MONITOR	
40	00363	1	LOGO, NORTH STAR	
41	00365	1	CRT & VIDEO PCB	
42	00366	1	TOP FAB	
43	00368	1	BEZEL, FAB	
44	00369	2	TAPE, SELF-ADHESIVE FOAM	
45	31001	1	FAN	
46	38007	9	WASHER, #10 FLAT	CRT MTG.
47	38065-06	4	WASHER, #6 FLAT	
48	38075-06	4	WASHER, #6 LOCK, SPLIT	
49	38075-10	9	WASHER, #10 LOCK, SPLIT	CRT MTG.
50	38091-08	5	SCREW, #10-32 X 1/2"	
51	38091-12	4	SCREW, #10-32 X 3/4"	
52	38082-12	4	SCREW, #6-32 X 3/4" BLACK	
53	38071	5	SCREW, #6 X 3/8" PAN HEAD	
54	38010	4	NUT, #6-32 HEX	
55		4	SPACER, 1/4" THICK	
56	77045	2	CABLE TIE, 3/4" BRACKET	(1) MONITOR CABLE
57	77046	2	CABLE TIE, 5.5" LONG, 40 LB	(1) CRT LEAD
58	00372	1	KNOB, CONTROL	BRIGHTNESS CNTR.
59	00154	1	TRANSFORMER, POWER	
60	00334	1	HARNESS, SEC. POWER SUPPLY	
61	38075-06	1	WASHER, LOCK, SPLIT, #6	PWR PNL GND
62		2	SCREW, #6-32 X 3/8"	CAPACITOR
63	38088-12	1	SCREW,MACHINE, #6-32 X 3/4"	PWR PNL GND
64	68016	1	FUSE, 3A, FAST BLOW	PWR PNL
65	77097-10	2	WIRE, #20 AWG,STRANDED,BLACK, 16" LG.	XMFR TO FAN
66	13067	2	TERMINAL FISO, 3/16" X .032	XFMR
67	13098	2	TERMINAL FISO, .110 X.020	FAN
68	00359	1	POWER PANEL ASSEMBLY	
69	00360	1	PLATE, POWER	

MAIN PARTS LIST (continued)

ITEM	P/N	QTY	DESCRIPTION	REF
70	34006	1	FILTER, LINE	LF1
71	68007	1	SWITCH, POWER	S1
72	00361	1	HARNESS, PRIMARY POWER SUPPLY	
73	38088-08	1	SCREW, #6-32 X 1/2"	PWR PNL
74	38083-08	2	SCREW, #4-40 X 1/2", BLACK	LINE FILTER
75	38075-04	2	WASHER, #4 LOCK, SPLIT	LINE FILTER
76	38075-06	1	WASHER, #6 LOCK, SPLIT	PWR PNL
77	38009	2	NUT, #4 HEX	LINE FILTER
78	00373	1	RECTIFIER AND CAPACITOR ASSEMBLY	
79	00354	1	RECT. & CAP. MTG. BRACKET	
80	01052	1	CAPACITOR, 12000 uF, 30 WVDC	
81	65001	1	RECTIFIER, BRIDGE, 100V, 25A	
82	13097	1	ADAPTER, TERMINAL, 1 TO 2 TABS	RECTIFER
83	38059	1	CLAMP, CAPACITOR	
84	38088-06	1	SCREW, MACH. PH, #6-32 X 3/8"	CAPACITOR
85	38088-12	1	SCREW, MACH. PH.,#6-32 X 3/4"	RECTIFIER
86	38075-06	2	WASHER, LOCK, SPLIT, #6	CAP & RECT
87	00148	1	PIO PCB ASSY	
88	00393	1	I/O MTG PLATE COVER	
89	38075	4	WASHER,LOCK,SLIT, #8	DRIVE BKT-BASE
90	00105	1	COVER ASSY	
91	00372-02	1	RING,CONTROL KNOB RETAINING	BRIGHTNESS CTRL

MAIN PC BOARD PARTS LIST

ITEM	P/N	QTY	DESCRIPTION	REF
110	01001	59	CAP, 0.047uF, CERAMIC DISK	*BYPASS,C39,C40, C50
111	01012	1	" 33pF, DIPPED MICA	C20
112	01013	2	" 100pF	C24,C16
113	01015	2	" 330pF	C22,C28
114	01016	2	" 470pF, DIPPED MICA	C15,C19
115	01018	3	" 0.047uF, DIPPED MYLAR	C4,C14,C25
116	01020	4	" 0.01uF, DIPPED MYLAR	C17,C23,C26,C27
117	10300	3	" 820uF, 15V, LOW ESR	C6,C10,C11
118	01039	72	" 0.01uF,16V,20%, CERAMIC	*BYPASS,C7,C9, C43-47
119	01041	8	" 22uF,20V, DIPPMED TANTALUM	C37,31,32,33,34, 35,36,41
120	04043	2	" 2.2uF, 35V	C48,C42
121	01044	1	" 62pF, 300V, 5%	C13
122	01045	1	" 0.0033uF,100V,10%	C12
123	01046	1	" 0.015uF,100V,10%	C2
124	01047	1	" 820pF, 300V	C3
125	01038	1	" 1000uF, 35V	C8
126	01014	2	" 200pF,15V, 5%, MICA	C29,C30
127	01050	2	" 0.1uF, 50V	C1,C5
128	01053	1	" 10pF,50VDC,+/- 0.5pF	C38
129	01056	1	" .47uF,35V, DIPPED TANT.	C18
130	01055	1	" .22uF, 10%,SOLID DIELETRIC	C21
131	13024	1	SOCKET, IC-8 PIN	14MB
132	13028	60	" " -16 PIN	1F-9F,1G-9G, 1H-9H,1J-9J, 1K-9K,1L-9L, J8,10B,13D, 13M,4B,15J, 16J,1A
133	13030	1	" " -20 PIN	17F
134	13032	2	" " -24 PIN	11K,18F
135	13036	2	SOCKET, IC-40 PIN	13K,18C
136	13081	1	CONNECTOR, 34 PIN	J9
137	13084	6	" EDGE-30 PIN	J1-J6
138	13094	1	" 6POST,1.56CTR,L/R	J11
139	13087	9	" PCB-"MINI JUMPERS"	W6-W14
140	13093	1	CONNECTOR,9POST,1.56CTR,L/R	J10
141	13091	8	HEADER, SINGLE ROW - 2 PIN	W7-W14
142	13095	1	CONNECTOR,10POST,0.1CTR,L/R	J7
143	15002	1	CRYSTAL, 8MHZ	Y1
144	43001	2	IC, 74 LS 00	10A
145	43002	2	" 74 LS 02	14E,4E

MAIN PC BOARD PARTS LIST (continued)

ITEM	P/N	QTY	DESCRIPTION	REF
146	43004	5	IC 74 LS 04	10K,2A,9C, 14D,15C
147	43006	1	" 74 LS 08	12E
148	43009	1	" 74 LS 14	16G
149	43012	4	" 74 LS 32	9A,12B,17H,7C
150	43015	6	" 74 LS 74	3A,7A,14C, 16B,16C,17C
151	43018	1	" 74 LS 123	9B
152	43021	2	" 74 LS 138	9,5L,12C
153	43022	1	" 74 LS 139	6C
154	43027	6	" 74 LS 161	14H,11C,14G, 14J,15G,15H
155	43028	1	" 74 LS 164	13G
156	43031	3	" 74 LS 175	11B,13E,15A
157	43034	4	" 74 LS 253	12H,12J,13H,13J
158	43039	1	" 74 LS 273	17G
159	43043	4	" 74 LS 373	9D,10D,10F,17E
160	43044	2	" 74 LS 393	8B, 14A
161	43045	4	" 74 S 00	8A,11H,13A,1E
162	43046	1	" 74 S 08	11J
163	43050	6	" 74 S 74	7B,7D,10C,11A, 13B,15B
164	43059	1	", 74 38	16E,18H,18J,6E
165	43025	2	" 74 LS 157	11G,12G
166	43068	1	" Z80A	13K
167	43069	1	" LF356	17A-A
168	43073	1	" CA3080	17A-B
169	43136	1	", 74L5123 (MFG TI)	17B
170	43079	1	" PROM-DWE	10B
171	43106	1	" 74 LS 20	8C
172	43109	3	" LM393N	16D,16L,14MB
173	43110	1	" LM358N	14MA
174	43112	9	" 74 LS 244	10E,10M,11E, 11F,11M,12M, 13F,13K,16F
175	43114	1	" 74 LS 279	13C
176	43115	3	" 74 LS 374	14F,15F,17F
177	42116	1	" 74 LS 670	13D
178	43117	1	" 74 S 04	6B
179	43118	1	" 74 S 86	1K
180	43120	2	" 74 S 139	12D,17J
181	42121	2	" 74 S 174	12A,16H
182	00145	1	" PROM, 'F-KYBD	18F
183	43123	1	" 8035	18C

MAIN PC BOARD PARTS LIST (continued)

ITEM	P/N	QTY	DESCRIPTION	REF
184	43124	52	IC 4116	1F-9F,1G-9G 1H-9H,1J-9J, 2K-9K,2L-9L
185	00117	1	" PROM, 2716-1	11K
186	43139	1	" PROM, HTIMH	16J
187	43140	1	" PROM, HTIML	15J
188	43141	1	" PROM, IOSEL	13M
189	43142	PBO	" PROM, VTIM	14B
190	43143	PBO	" PROM, VTIM50	14B
191	43144	1	" 74 LS 156	1M
192	61002	3	RES NET, 1K, SIP, 10 PIN	RN4,RN6,RN6
193	61003	1	" " , 2.2K, SIP, 6 PIN	RN3
194	61004	1	" " , 2.2K, SIP, 10 PIN	RN7
195	61007	4	" " , 47K, DIP, 10 PIN	RN10G,10H, 10J,10L
196	61009	1	RESISTOR, 3.3, 1/4W, 5%	R56
197	61010	2	" 22, 1/4W, 5%	R6,R37
198	61011	10	" 100 " "	R62-65,R11, 78,79,80,81, 82
199	61014	1	" 330 " "	R42
200	61015	1	" 470 " "	R24,39,40,48,53
201	61018	13	" 1K " "	R38,R69-73,R75, R76,R92-R95,R97
202	61021	2	" 3.3K " "	R52,R88
203	61022	1	" 3.6K " "	R50
204	61024	8	" 4.7K " "	R25,R1,12,21, 34,36,41,45
205	61025	2	" 5.6K " "	R51,R55
206	61026	2	" 6.8K " "	R60,R61
207	61027	1	" 9.1K " "	R43
208	61028	4	" 10K " 10%	R7,R17,R22,R85
209	61029	2	" 13K " 5%	R18,R58
210	61030	2	" 15K " "	R10,R23
211	61032	5	" 27K " "	R86,R46,R49, R54,R59
212	61034	3	" 47K " "	R5,R3,R74
213	61038	2	" 6.19K, 1%, RN55D	R47,R57
214	61042	2	" 220K, 1/4W, 5%	R35,R77
215	61054	1	" 33 " "	R9
216	61055	2	" 120 " "	R2,R13
217	61056	1	" 2.7K " "	R8
218	61057	1	" 30K " "	R27
219	61058	2	" 56K " "	R29,R84
220	61060	2	" 680, 1W, 10%	R3,R15

MAIN PC BOARD PARTS LIST (continued)

ITEM	P/N	QTY	DESCRIPTION	REF
221	61061	2	RESISTOR, 3.3 ,1/2W, 10%	R4,R15
222	61064	1	" 4.99K, 1%, RN55D	420
223	61065	1	" 6.98K, 1%, RN55D	R19
224	61067	2	" 47 , 1/4W, 5%	R66,R67
225	61068	1	RES NET,150, SIP, 8 PIN	RN2
226	61073	3	RESISTOR, 470K, 1/4W, 5%	R33,R44,R83
227	61017	2	RESISTOR, 680, 1/4W, 5%	R68,R101
228	61078	1	RES NET, 1K, SIP, 8 PIN	RN1
229	61079	1	RESISTOR, 620, 1/4W, 5%	R28
230	61082	1	" 330K " "	R16
231	61059	1	" 360K " "	R87
232	61031	1	" 18K " "	R89
233	61063	1	" 1.96K,1/8W, 1%	R90
234	61071	1	" 2.87K, " "	R91
235	61088	1	" 22K, 1/4W, 5%	R30
236	61087	1	" 100, 1/2W, 20%	R96
237	61085	1	POTENTIOMETER, 5K,MULTI TURN	R26
238	61089	1	RES NET, 100, DIP-16 PIN	7E
239	38002	3	WASHER, LOCK - #6	
240	38010	3	NUT, HEX #6-32	
241	38041	2	HEAT SINK, #6030	
242	38043	1	HEAD SINK, #6107	
243	38073	3	SCREW,MACH,#6-32X3/8,PAN HD	
244	65002	1	REGULATOR, 7805	VR3
245	65006	1	REGULATOR, 79L05	VR2
246	65009	15	DIODE, 1N4148	CR4,CR5, CR7-CR19
247	65014	5	TRANSISTOR,2N2222A	Q3,Q7,Q8, Q10,Q11
248	65015	2	TRANSISTOR,2N2907A	Q4,Q9
249	65018	1	REGULATOR, 7912 (TO-220)	VR1
250	65020	2	TRANSISTOR,D44H5,GE	Q1,Q5
251	65021	2	TRANSISTOR,D45C5,GE	Q2,Q6
252	65022	1	RECTIFIER,C122F,GE SCR	SCR1
253	65024	2	DIODE,M4820,50V,8A	CR1,CR3
254	65025	2	DIODE,IN823,6.2V,ZENER	CR6,CR2
255	68004	1	FUSE, 5AMP, FAST-BLOW	F1
256	68013	2	CLIP, FUSE-BUSSMAN	
257	68015	1	SWITCH, PUSH BUTTON TOGGLE	S1
258	74007	2	INDUCTOR, 250uH, 10%, 5A	L1,L2
259	74009	1	INDUCTOR, 3.3uH, 10%	L3
260	82017	1	LOUDSPEAKER, MINI	IS1
261	38081	2	SPACER, NYLON PUSH	

MAIN PC BOARD PARTS LIST (continued)

ITEM	P/N	QTY	DESCRIPTION	REF
262	77046	1	CABLE TIE, 5.5" LONG, 40LBS	
263	43017	1	IC, 74 LS 109	2E
264	43040	1	" 74 LS 280	8E
265	43146	2	" 74 LS 166	8D,11D
			(NOTE: 74166 IS ALTERNATE, #43063)	
266	43147	1	IC, 74S124	1A
267	13092-03	1	HEADER,single row-3 pin	W6
268	01061	76	CAP, 0.1uf, 16V,20%, ceramic	**BYPASS
269	01022	1	CAP, 6.8uf, 35V,TANTALUM	C49
270	68016	1	FUSE,3A,20MM X 5MM	
271	68017	1	FUSE,1.5A,20MM X 5MM	
272	00364-02	1	LABEL,POWER RATING (115V,60HZ)	
273	00364-04	1	LABEL,POWER RATING (230V,50HZ)	
274	00364-05	1	LABEL,POWER RATING (230V,60HZ)	

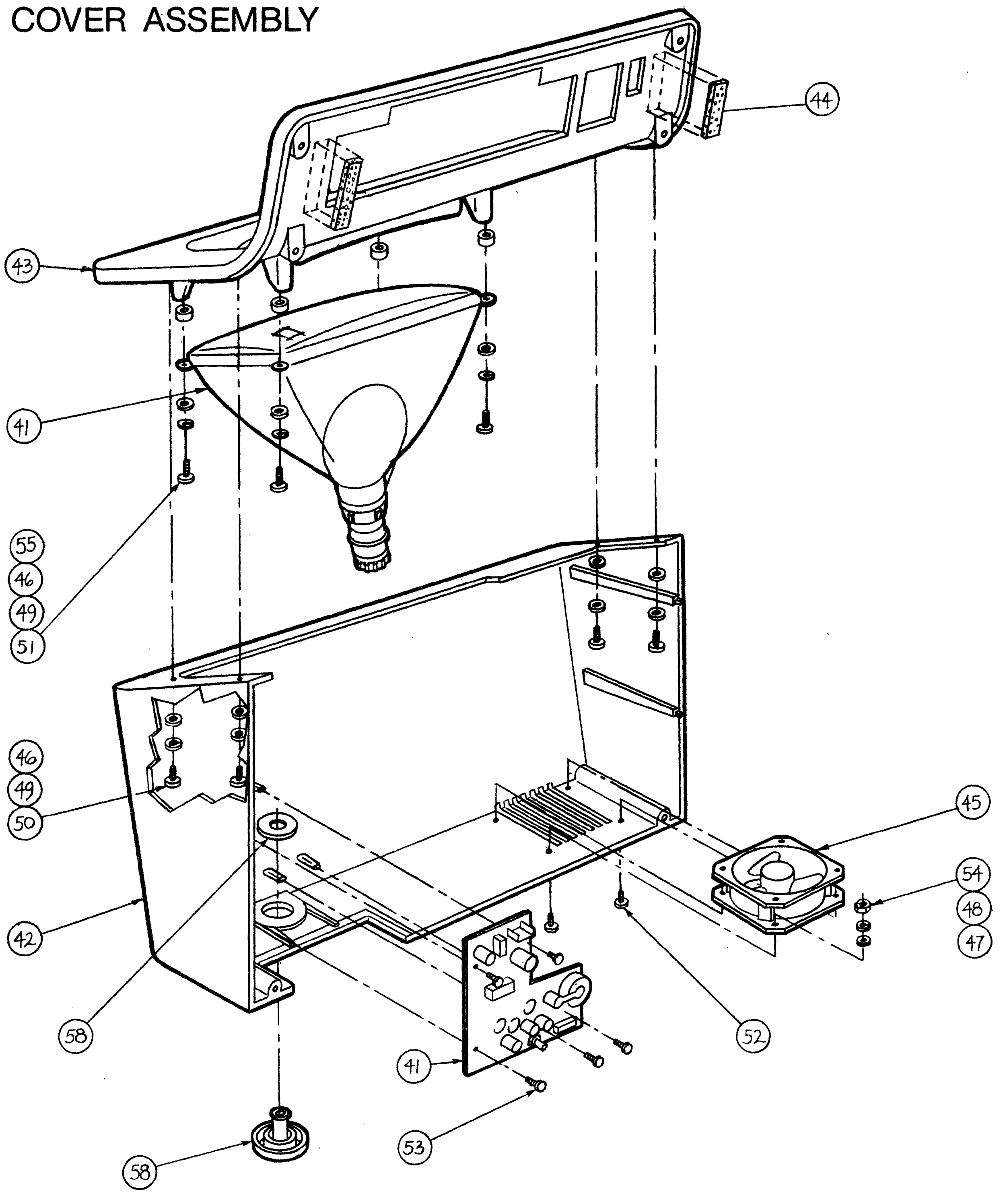
SIO BOARD PARTS LIST

ITEM	P/N	QTY	DESCRIPTION	REF
280	01001	10	CAP. .047uF - CERAMIC	* BY PASS
281	01005	4	" 470pf - CERAMIC	
282	01022	3	" 6.8uF - TANTALUM 35V	C1,C2,C3
283	13017	1	CONNECTOR, "D" TYPE-RIGHT ANGLE, 25 PIN	P1
284	13025	1	SOCKET, IC - 8 PIN	1A
285	13026	1	" " -14 PIN	4A
286	13029	1	" " -16 PIN	3A
287	13034	1	" " -28 PIN	4E
288	13064	1	SHUNT, 16 PIN (CONFIG)	3A
289	43001	1	IC, 74 LS 00	3B
290	43003	1	" 74 LS 03	3D
291	43004	1	" 74 LS 04	2B
292	43017	1	" 74 LS 109	3C
293	43021	1	" 74 LS 138	2C
294	43027	3	" 74 LS 161	1B,1C,1D
295	43030	1	" 74 LS 174	1E
296	43031	1	" 74 LS 175	2D
297	43070	1	" MC 1488	4A
298	43071	1	" MC 1489	2A
299	43095	1	" 8251, USART	4E
300	43135	2	" 74 LS 243	2E,3E
301	61013	1	RESISTOR, 220, 1/4W, 5%	R3
302	61016	1	" 560 " "	R2
303	61019	1	" 1.2K " "	R1
304	61025	1	" 5.6K " "	R5
305	61027	1	" 9.1K " "	R4
306	61035	1	" 1K , 1/2W, 5%	R6
307	61024	1	" 4.7K, 1/4W, 5%	R7

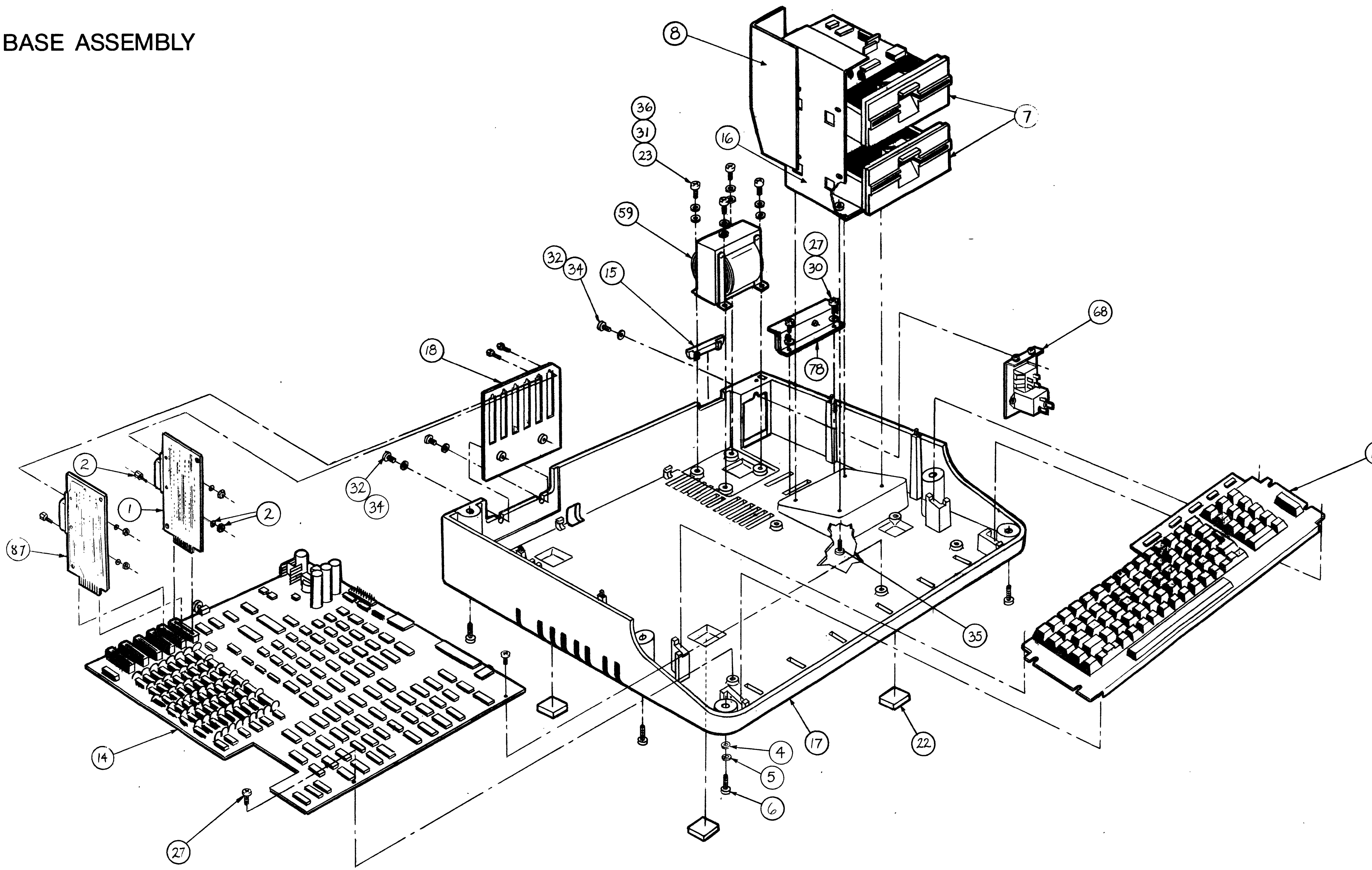
PIO BOARD PARTS LIST

ITEM	P/N	QTY	DESCRIPTION	REF
320	01001	8	CAP, .047uF, CERAMIC-DISK	*BYPASS
321	01022	2	" 6.8uF, TATALUM 35V	C1,C2
322	13016	2	CONNECTOR "D" TYPE - 15 PIN	P1,P2
323	13029	1	SOCKET, IC - 16 PIN	1A
324	13064	1	SHUNT, 16 PIN	1A
325	43031	1	IC, 74 LS 175	2C
326	43065	1	" 74367	1D
327	43003	1	" 74 LS 03	1C
328	43009	1	" 74 LS 14	3A
329	43012	1	" 74 LS 32	2A
330	43015	1	" 74 LS 74	1B
331	43021	1	" 74 LS 138	2B
332	43043	2	" 74 LS 373/68 LS 373	3C,3D
333	43058	1	" 7437	3B
334	61003	1	RESISTOR NETWORK, 2.2K,SIP, 6 PIN	RN1
335	61013	2	RESISTOR, 220 -1/4W, 5%	R1,R3
336	61014	2	RESISTOR, 330 -1/4W, 5%	R2,R4

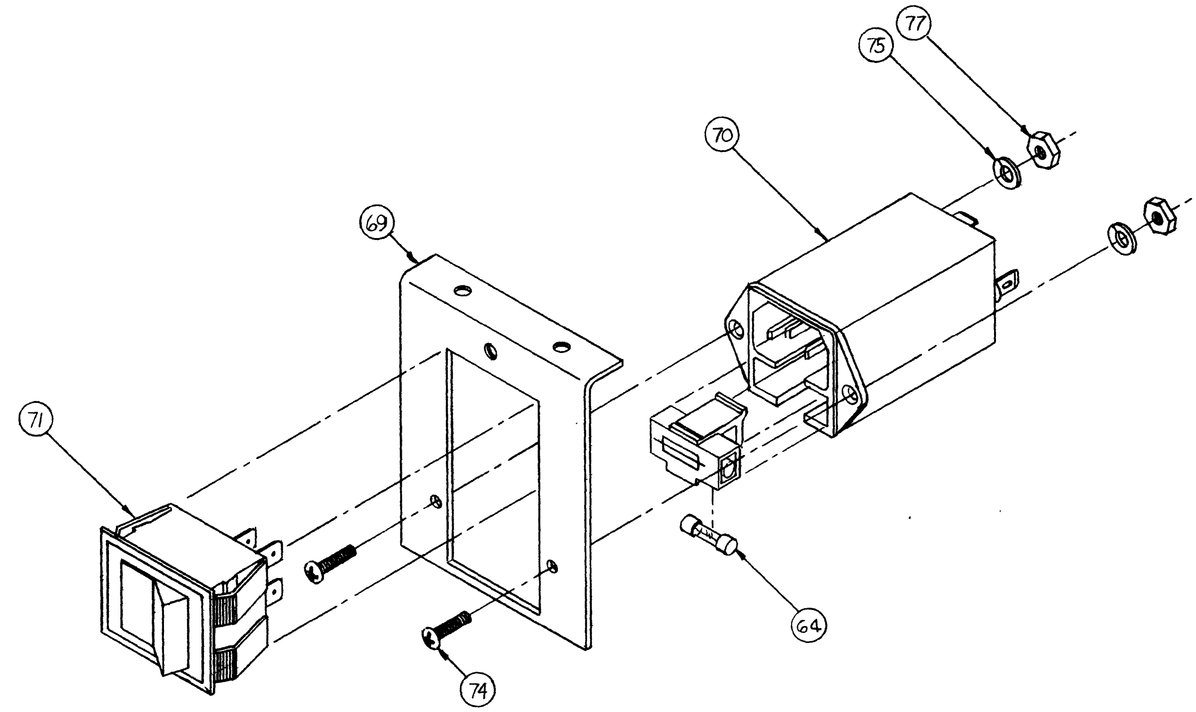
COVER ASSEMBLY



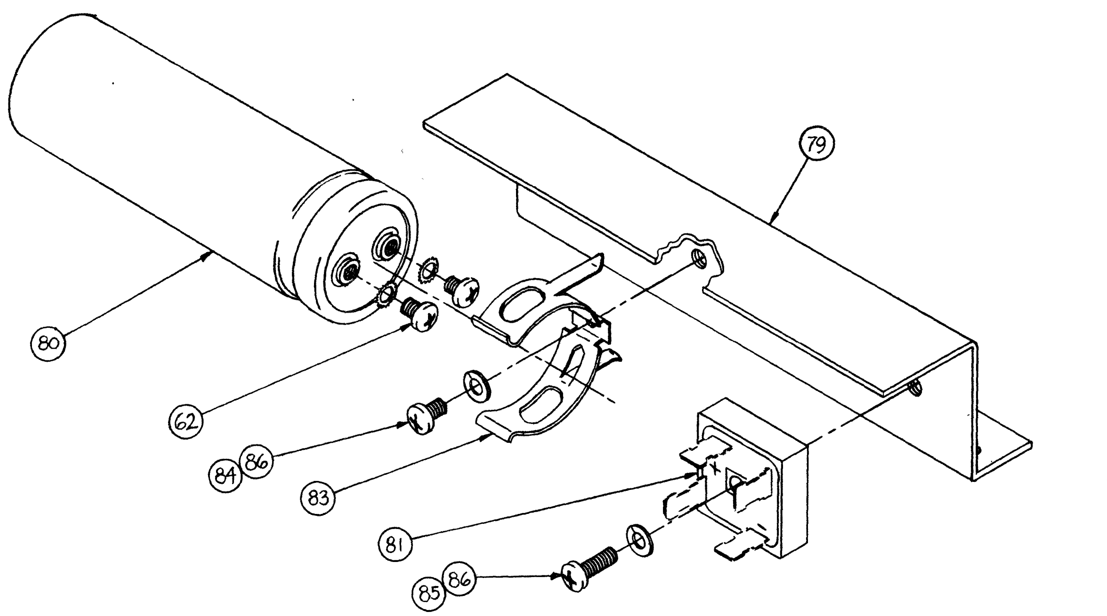
BASE ASSEMBLY

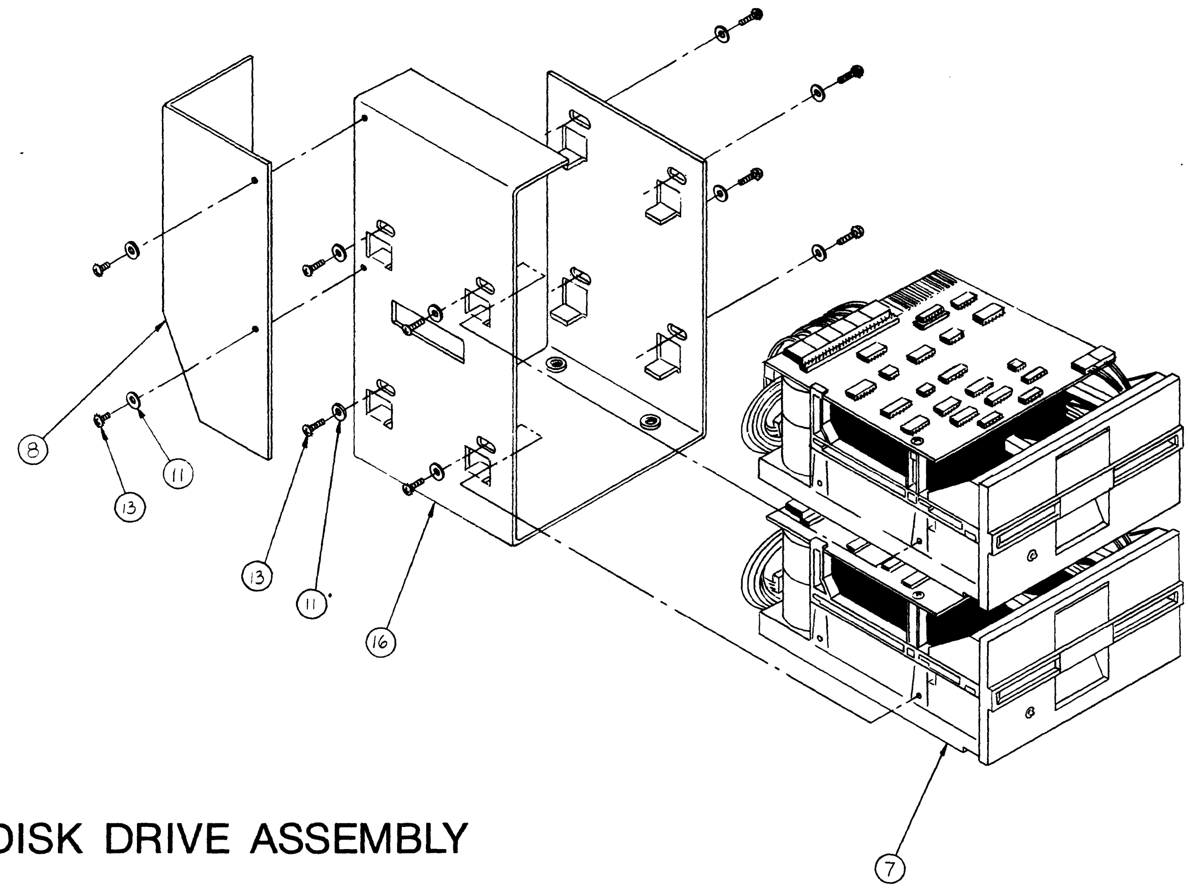


POWER PANEL ASSEMBLY



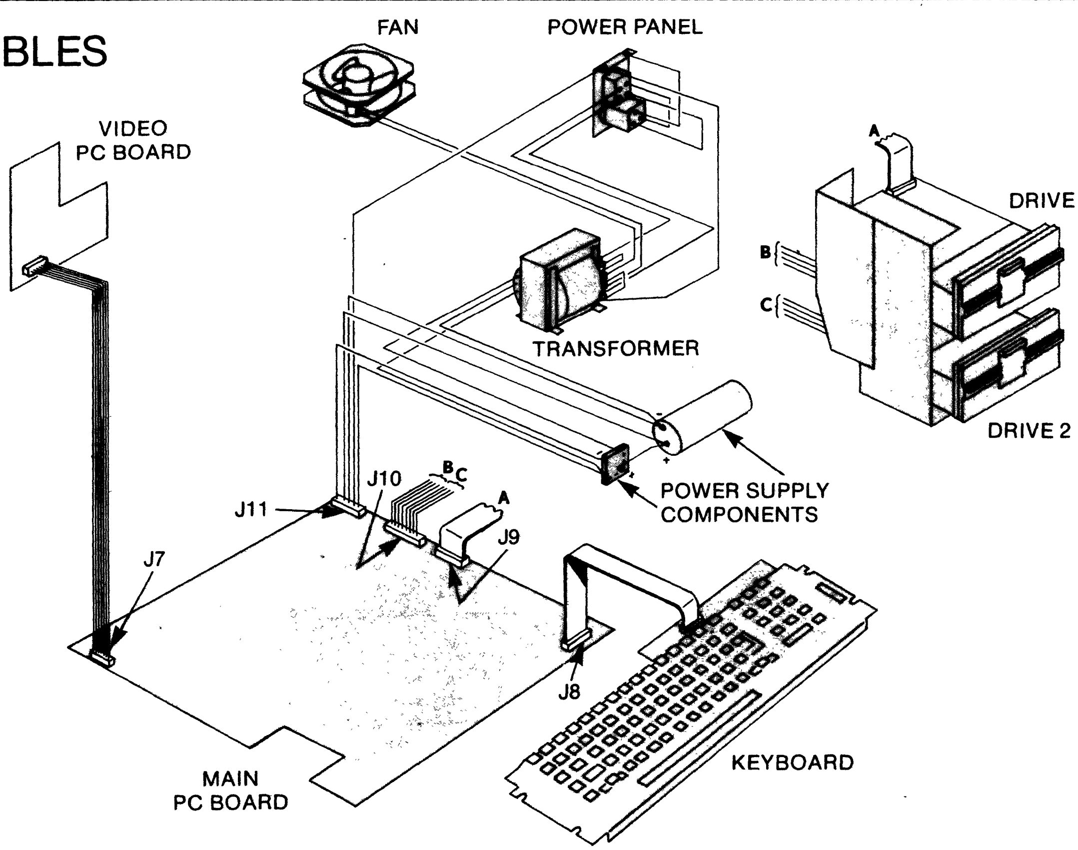
RECTIFIER AND CAPACITOR ASSEMBLY



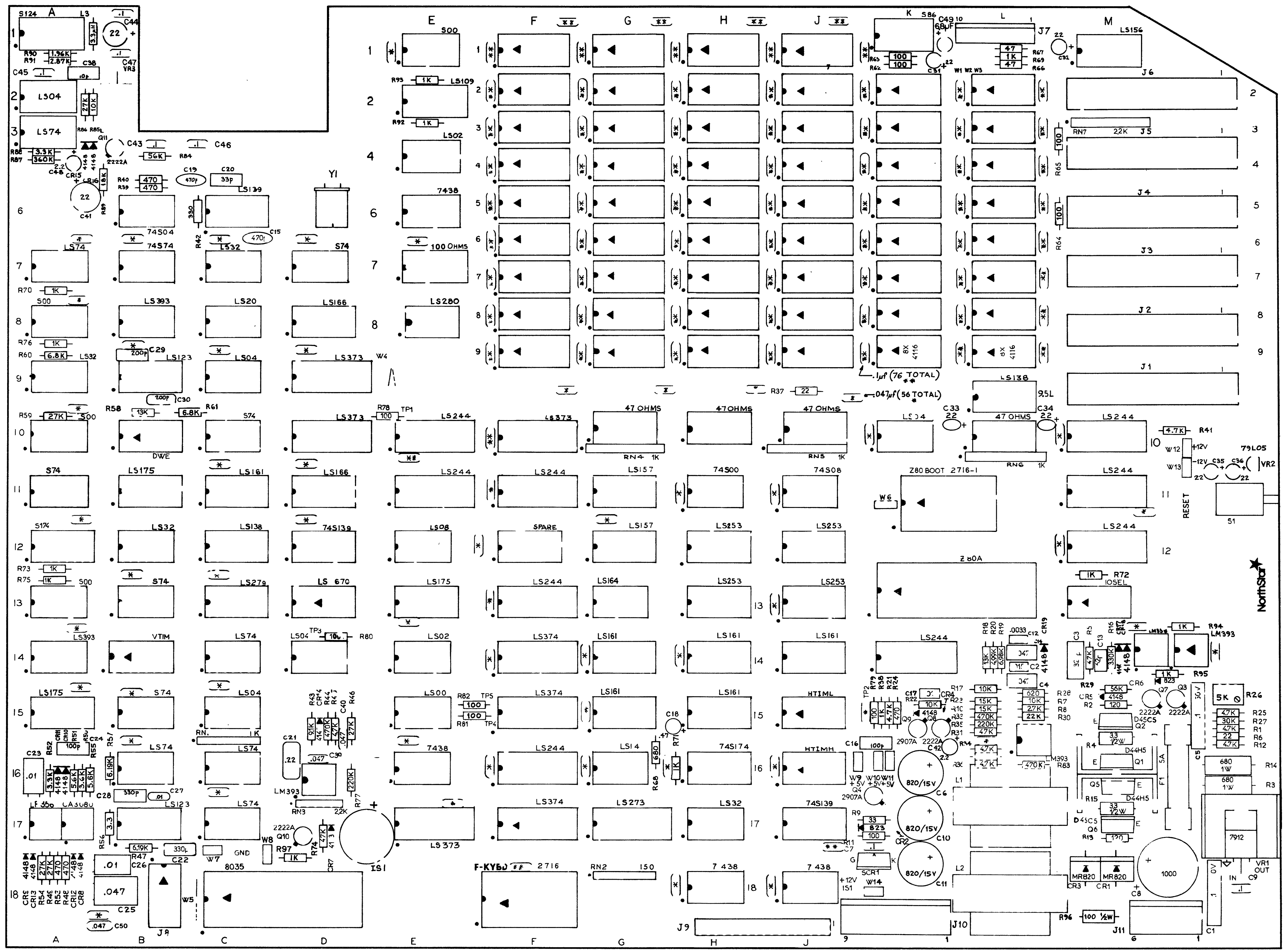


DISK DRIVE ASSEMBLY

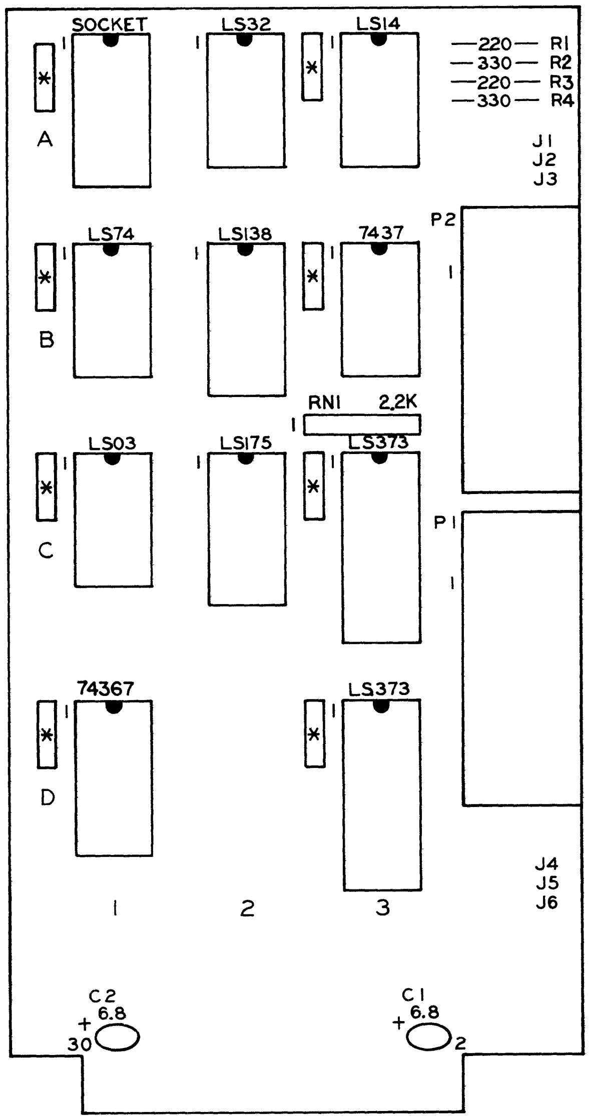
CABLES



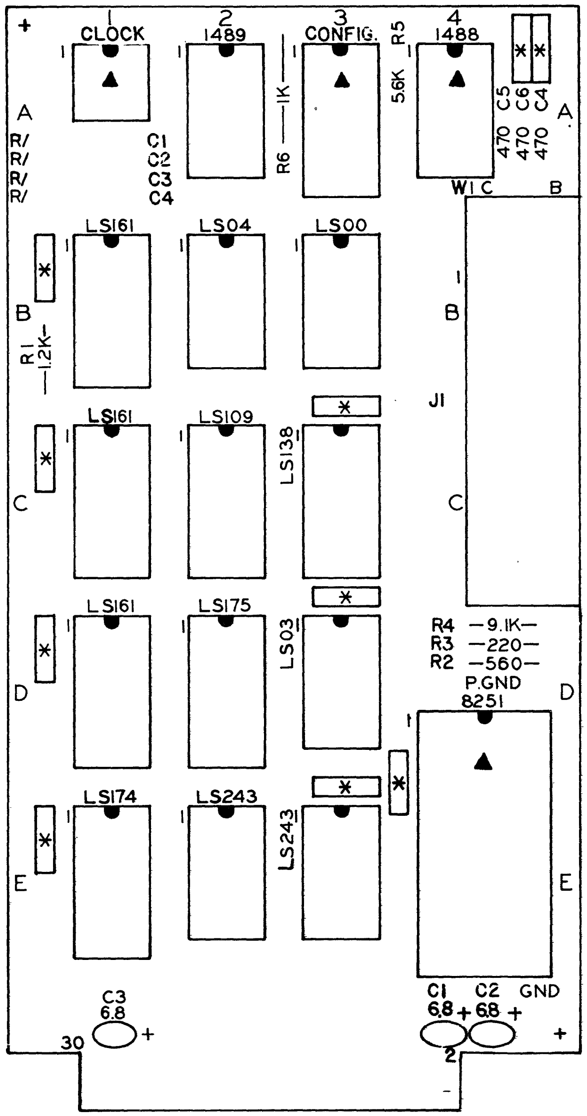
MAIN PC BOARD ASSEMBLY



PIO BOARD ASSEMBLY



SIO BOARD ASSEMBLY



Z80 MICROPROCESSOR DATA SHEET

Z8400
Z80® CPU Central Processing Unit

Zilog

Product Specification

March 1981

Features

- The instruction set contains 158 instructions. The 78 instructions of the 8080A are included as a subset; 8080A software compatibility is maintained.
- Six MHz, 4 MHz and 2.5 MHz clocks for the Z80B, Z80A, and Z80 CPU result in rapid instruction execution with consequent high data throughput.
- The extensive instruction set includes string, bit, byte, and word operations. Block searches and block transfers together with indexed and relative addressing result in the most powerful data handling capabilities in the microcomputer industry.
- The Z80 microprocessors and associated family of peripheral controllers are linked by a vectored interrupt system. This system may be daisy-chained to allow implementation of a priority interrupt scheme. Little, if any, additional logic is required for daisy-chaining.
- Duplicate sets of both general-purpose and flag registers are provided, easing the design and operation of system software through single-context switching, background-foreground programming, and single-level interrupt processing. In addition, two 16-bit index registers facilitate program processing of tables and arrays.
- There are three modes of high speed interrupt processing: 8080 compatible, non-Z80 peripheral device, and Z80 Family peripheral with or without daisy chain.
- On-chip dynamic memory refresh counter.

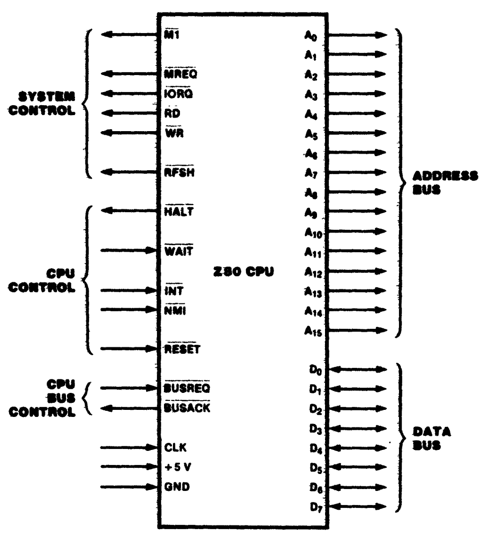


Figure 1. Pin Functions

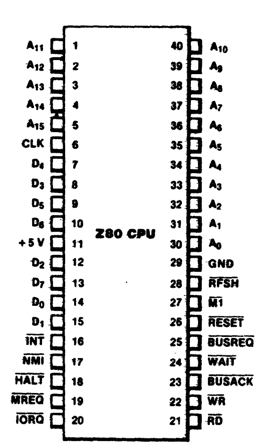


Figure 2. Pin Assignments

General Description

The Z80, Z80A, and Z80B CPUs are third-generation single-chip microprocessors with exceptional computational power. They offer higher system throughput and more efficient memory utilization than comparable second- and third-generation microprocessors. The internal registers contain 208 bits of read/write memory that are accessible to the programmer. These registers include two sets of six general-purpose registers which may be used individually as either 8-bit registers or as 16-bit register pairs. In addition, there are two sets of accumulator and flag registers. A group of "Exchange" instructions makes either set of main or alternate registers accessible to the programmer. The alternate set allows operation in foreground-background mode or it may be reserved for very fast interrupt response.

The Z80 also contains a Stack Pointer, Program Counter, two index registers, a Refresh register (counter), and an Interrupt register. The CPU is easy to incorporate into a system since it requires only a single +5 V power source, all output signals are fully decoded and timed to control standard memory or peripheral circuits, and is supported by an extensive family of peripheral controllers. The internal block diagram (Figure 3) shows the primary functions of the Z80 processors. Subsequent text provides more detail on the Z80 I/O controller family, registers, instruction set, interrupts and daisy chaining, and CPU timing.

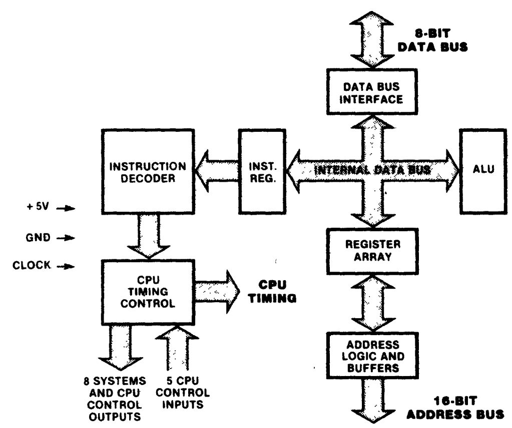


Figure 3. Z80 CPU Block Diagram

Z80 Microprocessor Family

The Zilog Z80 microprocessor is the central element of a comprehensive microprocessor product family. This family works together in most applications with minimum requirements for additional logic, facilitating the design of efficient and cost-effective microcomputer-based systems.

Zilog has designed five components to provide extensive support for the Z80 microprocessor. These are:

- The PIO (Parallel Input/Output) operates in both data-byte I/O transfer mode (with handshaking) and in bit mode (without handshaking). The PIO may be configured to interface with standard parallel peripheral devices such as printers, tape punches, and keyboards.
- The CTC (Counter/Timer Circuit) features four programmable 8-bit counter/timers, each of which has an 8-bit prescaler. Each of the four channels may be configured to operate in either counter or timer mode.
- The DMA (Direct Memory Access) controller provides dual port data transfer operations and the ability to terminate data transfer as a result of a pattern match.
- The SIO (Serial Input/Output) controller offers two channels. It is capable of operating in a variety of programmable modes for both synchronous and asynchronous communication, including Bi-Synch and SDLC.
- The DART (Dual Asynchronous Receiver/Transmitter) device provides low cost asynchronous serial communication. It has two channels and a full modem control interface.

Z80 CPU Registers

Figure 4 shows three groups of registers within the Z80 CPU. The first group consists of duplicate sets of 8-bit registers: a principal set and an alternate set (designated by ' [prime], e.g., A'). Both sets consist of the Accumulator Register, the Flag Register, and six general-purpose registers. Transfer of data between these duplicate sets of registers is accomplished by use of "Exchange" instructions. The result is faster response to interrupts and easy, efficient implementation of such versatile programming techniques as background-foreground data processing. The second set of registers consists of six registers with assigned functions. These are the I (Interrupt Register), the R (Refresh Register), the IX and IY (Index Registers), the SP (Stack Pointer), and the PC (Program Counter). The third group consists of two interrupt status flip-flops, plus an additional pair of flip-flops which assists in identifying the interrupt mode at any particular time. Table 1 provides further information on these registers.

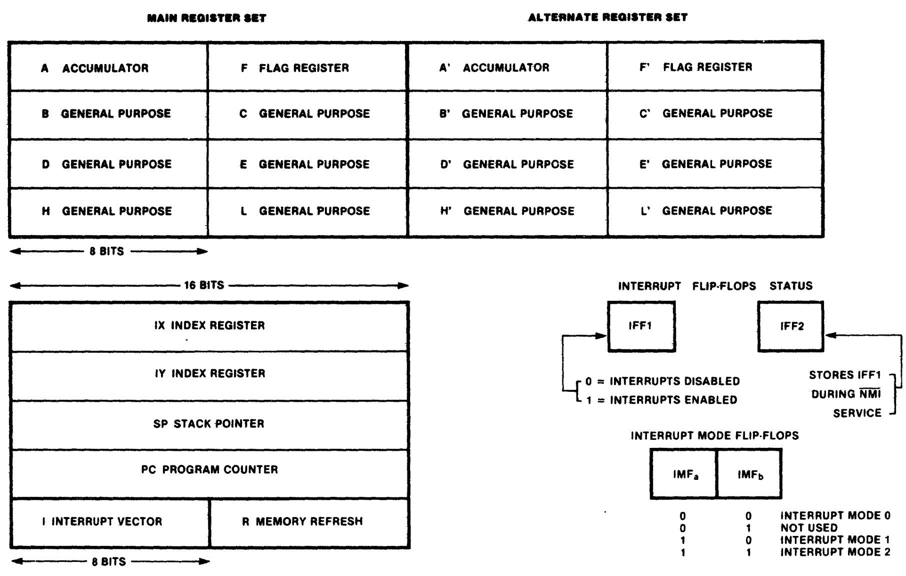


Figure 4. CPU Registers

Z80 CPU Registers (Continued)

Register		Size (Bits)	Remarks
A, A′	Accumulator	8	Stores an operand or the results of an operation.
F, F′	Flags	8	See Instruction Set.
B, B′	General Purpose	8	Can be used separately or as a 16-bit register with C.
C, C′	General Purpose	8	See B, above.
D, D′	General Purpose	8	Can be used separately or as a 16-bit register with E.
E, E′	General Purpose	8	See D, above.
H, H′	General Purpose	8	Can be used separately or as a 16-bit register with L.
L, L′	General Purpose	8	See H, above.
			Note: The (B,C), (D,E), and (H,L) sets are combined as follows: B — High byte C — Low byte D — High byte E — Low byte H — High byte L — Low byte
I	Interrupt Register	8	Stores upper eight bits of memory address for vectored interrupt processing.
R	Refresh Register	8	Provides user-transparent dynamic memory refresh. Automatically incremented and placed on the address bus during each instruction fetch cycle.
IX	Index Register	16	Used for indexed addressing.
IY	Index Register	16	Same as IX, above.
SP	Stack Pointer	16	Stores addresses or data temporarily. See Push or Pop in instruction set.
PC	Program Counter	16	Holds address of next instruction.
IFF_1-IFF_2	Interrupt Enable	Flip-Flops	Set or reset to indicate interrupt status (see Figure 4).
IMFa-IMFb	Interrupt Mode	Flip-Flops	Reflect Interrupt mode (see Figure 4).

Table 1. Z80 CPU Registers

Interrupts: General Operation

The CPU accepts two interrupt input signals: $\overline{NMI}$ and $\overline{INT}$. The $\overline{NMI}$ is a non-maskable interrupt and has the highest priority. $\overline{INT}$ is a lower priority interrupt since it requires that interrupts be enabled in software in order to operate. Either $\overline{NMI}$ or $\overline{INT}$ can be connected to multiple peripheral devices in a wired-OR configuration.

The Z80 has a single response mode for interrupt service for the non-maskable interrupt. The maskable interrupt, $\overline{INT}$, has three programmable response modes available. These are:

- Mode 0 — compatible with the 8080 microprocessor.
- Mode 1 — Peripheral Interrupt service, for use with non-8080/Z80 systems.
- Mode 2 — a vectored interrupt scheme, usually daisy-chained, for use with Z80 Family and compatible peripheral devices.

The CPU services interrupts by sampling the $\overline{NMI}$ and $\overline{INT}$ signals at the rising edge of the last clock of an instruction. Further interrupt service processing depends upon the type of interrupt that was detected. Details on interrupt responses are shown in the CPU Timing Section.

Interrupts: General Operation (Continued)

Non-Maskable Interrupt ($\overline{NMI}$). The non-maskable interrupt cannot be disabled by program control and therefore will be accepted at at all times by the CPU. $\overline{NMI}$ is usually reserved for servicing only the highest priority type interrupts, such as that for orderly shutdown after power failure has been detected. After recognition of the $\overline{NMI}$ signal (providing $\overline{BUSREQ}$ is not active), the CPU jumps to restart location 0066H. Normally, software starting at this address contains the interrupt service routine.

Maskable Interrupt ($\overline{INT}$). Regardless of the interrupt mode set by the user, the Z80 response to a maskable interrupt input follows a common timing cycle. After the interrupt has been detected by the CPU (provided that interrupts are enabled and $\overline{BUSREQ}$ is not active) a special interrupt processing cycle begins. This is a special fetch ($\overline{M1}$) cycle in which $\overline{IORQ}$ becomes active rather than $\overline{MREQ}$, as in a normal $\overline{M1}$ cycle. In addition, this special $\overline{M1}$ cycle is automatically extended by two $\overline{WAIT}$ states, to allow for the time required to acknowledge the interrupt request and to place the interrupt vector on the bus.

Mode 0 Interrupt Operation. This mode is compatible with the 8080 microprocessor interrupt service procedures. The interrupting device places an instruction on the data bus, which is then acted on six times by the CPU. This is normally a Restart Instruction, which will initiate an unconditional jump to the selected one of eight restart locations in page zero of memory.

Mode 1 Interrupt Operation. Mode 1 operation is very similar to that for the $\overline{NMI}$. The principal difference is that the Mode 1 interrupt has a vector address of 0038H only.

Mode 2 Interrupt Operation. This interrupt mode has been designed to utilize most effectively the capabilities of the Z80 microprocessor and its associated peripheral family. The interrupting peripheral device selects the starting address of the interrupt service routine. It does this by placing an 8-bit address vector on the data bus during the interrupt acknowledge cycle. The high-order byte of the interrupt service routine address is supplied by the I (Interrupt) register. This flexibility in selecting the interrupt service routine address allows the peripheral device to use several different types of service routines. These routines may be located at any available location in memory. Since the interrupting device supplies the low-order byte of the 2-byte vector, bit 0 (A_0) must be a zero.

Interrupt Priority (Daisy Chaining and Nested Interrupts). The interrupt priority of each peripheral device is determined by its physical location within a daisy-chain configuration. Each device in the chain has an interrupt enable input line (IEI) and an interrupt enable output line (IEO), which is fed to the next lower priority device. The first device in the daisy chain has its IEI input hardwared to a High level. The first device has highest priority, while each succeeding device has a corresponding lower priority. This arrangement permits the CPU to select the highest priority interrupt from several simultaneously interrupting peripherals.

The interrupting device disables its IEO line to the next lower priority peripheral until it has been serviced. After servicing, its IEO line is raised, allowing lower priority peripherals to demand interrupt servicing.

The Z80 CPU will nest (queue) any pending interrupts or interrupts received while a selected peripheral is being serviced.

Interrupt Enable/Disable Operation. Two flip-flops, IFF_1 and IFF_2, referred to in the register description are used to signal the CPU interrupt status. Operation of the two flip-flops is described in Table 2. For more details, refer to the *Z80 CPU Technical Manual* and *Z80 Assembly Language Manual.*

Action	IFF_1	IFF_2	Comments
CPU Reset	0	0	Maskable interrupt $\overline{INT}$ disabled
DI instruction execution	0	0	Maskable interrupt $\overline{INT}$ disabled
EI instruction execution	1	1	Maskable interrupt $\overline{INT}$ enabled
LD A,I instruction execution	•	•	$IFF_2 \rightarrow$ Parity flag
LD A,R instruction execution	•	•	$IFF_2 \rightarrow$ Parity flag
Accept $\overline{NMI}$	0	IFF_1	$IFF_1 \rightarrow IFF_2$ (Maskable interrupt $\overline{INT}$ disabled)
RETN instruction execution	IFF_2	•	$IFF_2 \rightarrow IFF_1$ at completion of an $\overline{NMI}$ service routine.

Table 2. State of Flip-Flops

Instruction Set

The Z80 microprocessor has one of the most powerful and versatile instruction sets available in any 8-bit microprocessor. It includes such unique operations as a block move for fast, efficient data transfers within memory or between memory and I/O. It also allows operations on any bit in any location in memory.

The following is a summary of the Z80 instruction set and shows the assembly language mnemonic, the operation, the flag status, and gives comments on each instruction. The *Z80 CPU Technical Manual* (03-0029-01) and *Assembly Language Programming Manual* (03-0002-01) contain significantly more details for programming use.

The instructions are divided into the following categories:

- □ 8-bit loads
- □ 16-bit loads
- □ Exchanges, block transfers, and searches
- □ 8-bit arithmetic and logic operations
- □ General-purpose arithmetic and CPU control
- □ 16-bit arithmetic operations
- □ Rotates and shifts
- □ Bit set, reset, and test operations
- □ Jumps
- □ Calls, returns, and restarts
- □ Input and output operations

A variety of addressing modes are implemented to permit efficient and fast data transfer between various registers, memory locations, and input/output devices. These addressing modes include:

- □ Immediate
- □ Immediate extended
- □ Modified page zero
- □ Relative
- □ Extended
- □ Indexed
- □ Register
- □ Register indirect
- □ Implied
- □ Bit

8-Bit Load Group

Mnemonic	Symbolic Operation	S	Z		Flags H		P/V	N	C	Opcode 76 543 210	Hex	No. of Bytes	No. of M Cycles	No. of T States	Comments
LD r, r'	r ← r'	•	•	X	•	X	•	•	•	01 r r'		1	1	4	r, r' Reg.
LD r, n	r ← n	•	•	X	•	X	•	•	•	00 r 110 ← n →		2	2	7	000 B 001 C
LD r, (HL)	r ← (HL)	•	•	X	•	X	•	•	•	01 r 110		1	2	7	010 D
LD r, (IX+d)	r ← (IX+d)	•	•	X	•	X	•	•	•	11 011 101 01 r 101 ← d →	DD	3	5	19	011 E 100 H 101 L
LD r, (IY+d)	r ← (IY+d)	•	•	X	•	X	•	•	•	11 111 101 01 r 110 ← d →	FD	3	5	19	111 A
LD (HL), r	(HL) ← r	•	•	X	•	X	•	•	•	01 110 r		1	2	7	
LD (IX+d), r	(IX+d) ← r	•	•	X	•	X	•	•	•	11 011 101 01 110 r ← d →	DD	3	5	19	
LD (IY+d), r	(IY+d) ← r	•	•	X	•	X	•	•	•	11 111 101 01 110 r ← d →	FD	3	5	19	
LD (HL), n	(HL) ← n	•	•	X	•	X	•	•	•	00 110 110 ← n →	36	2	3	10	
LD (IX+d), n	(IX+d) ← n	•	•	X	•	X	•	•	•	11 011 101 00 110 110 ← d → ← n →	DD 36	4	5	19	
LD (IY+d), n	(IY+d) ← n	•	•	X	•	X	•	•	•	11 111 101 00 110 110 ← d → ← n →	FD 36	4	5	19	
LD A, (BC)	A ← (BC)	•	•	X	•	X	•	•	•	00 001 010	0A	1	2	7	
LD A, (DE)	A ← (DE)	•	•	X	•	X	•	•	•	00 011 010	1A	1	2	7	
LD A, (nn)	A ← (nn)	•	•	X	•	X	•	•	•	0C 111 010 ← n → ← n →	3A	3	4	13	
LD (BC), A	(BC) ← A	•	•	X	•	X	•	•	•	00 000 010	02	1	2	7	
LD (DE), A	(DE) ← A	•	•	X	•	X	•	•	•	00 010 010	12	1	2	7	
LD (nn), A	(nn) ← A	•	•	X	•	X	•	•	•	00 110 010 ← n → ← n →	32	3	4	13	
LD A, I	A ← I	↕	↕	X	0	X	IFF	0	•	11 101 101 01 010 111	ED 57	2	2	9	
LD A, R	A ← R	↕	↕	X	0	X	IFF	0	•	11 101 101 01 011 111	ED 5F	2	2	9	
LD I, A	I ← A	•	•	X	•	X	•	•	•	11 101 101 01 000 111	ED 47	2	2	9	
LD R, A	R ← A	•	•	X	•	X	•	•	•	11 101 101 01 001 111	ED 4F	2	2	9	

NOTES. r, r' means any of the registers A, B, C, D, E, H, L
IFF the content of the interrupt enable flip-flop, (IFF) is copied into the P/V flag
For an explanation of flag notation and symbols for mnemonic tables, see Symbolic Notation section following tables

16-Bit Load Group

Mnemonic	Symbolic Operation	S	Z		H		P/V	N	C	Opcode 76 543 210	Hex	No. of Bytes	No. of M Cycles	No. of T States	Comments
LD dd, nn	dd ← nn	•	•	X	•	X	•	•	•	00 dd0 001 ← n → ← n →		3	3	10	dd Pair 00 BC 01 DE 10 HL 11 SP
LD IX, nn	IX ← nn	•	•	X	•	X	•	•	•	11 011 101 00 100 001 ← n → ← n →	DD 21	4	4	14	
LD IY, nn	IY ← nn	•	•	X	•	X	•	•	•	11 111 101 00 100 001 ← n → ← n →	FD 21	4	4	14	
LD HL, (nn)	H ← (nn+1) L ← (nn)	•	•	X	•	X	•	•	•	00 101 010 ← n → ← n →	2A	3	5	16	
LD dd, (nn)	dd_H ← (nn+1) dd_L ← (nn)	•	•	X	•	X	•	•	•	11 101 101 01 dd1 011 ← n → ← n →	ED	4	6	20	
LD IX, (nn)	IX_H ← (nn+1) IX_L ← (nn)	•	•	X	•	X	•	•	•	11 011 101 00 101 010 ← n → ← n →	DD 2A	4	6	20	
LD IY, (nn)	IY_H ← (nn+1) IY_L ← (nn)	•	•	X	•	X	•	•	•	11 111 101 00 101 010 ← n → ← n →	FD 2A	4	6	20	
LD (nn), HL	(nn+1) ← H (nn) ← L	•	•	X	•	X	•	•	•	00 100 010 ← n → ← n →	22	3	5	16	
LD (nn), dd	(nn+1) ← dd_H (nn) ← dd_L	•	•	X	•	X	•	•	•	11 101 101 01 dd0 011 ← n → ← n →	ED	4	6	20	
LD (nn), IX	(nn+1) ← IX_H (nn) ← IX_L	•	•	X	•	X	•	•	•	11 011 101 00 100 010 ← n → ← n →	DD 22	4	6	20	
LD (nn), IY	(nn+1) ← IY_H (nn) ← IY_L	•	•	X	•	X	•	•	•	11 111 101 00 100 010 ← n → ← n →	FD 22	4	6	20	
LD SP, HL	SP ← HL	•	•	X	•	X	•	•	•	11 111 001	F9	1	1	6	
LD SP, IX	SP ← IX	•	•	X	•	X	•	•	•	11 011 101 11 111 001	DD F9	2	2	10	
LD SP, IY	SP ← IY	•	•	X	•	X	•	•	•	11 111 101 11 111 001	FD F9	2	2	10	
PUSH qq	(SP−2) ← qq_L (SP−1) ← qq_H SP ← SP −2	•	•	X	•	X	•	•	•	11 qq0 101		1	3	11	qq Pair 00 BC 01 DE 10 HL 11 AF
PUSH IX	(SP−2) ← IX_L (SP−1) ← IX_H SP ← SP −2	•	•	X	•	X	•	•	•	11 011 101 11 100 101	DD E5	2	4	15	
PUSH IY	(SP−2) ← IY_L (SP−1) ← IY_H SP ← SP −2	•	•	X	•	X	•	•	•	11 111 101 11 100 101	FD E5	2	4	15	
POP qq	qq_H ← (SP+1) qq_L ← (SP) SP ← SP +2	•	•	X	•	X	•	•	•	11 qq0 001		1	3	10	
POP IX	IX_H ← (SP+1) IX_L ← (SP) SP ← SP +2	•	•	X	•	X	•	•	•	11 011 101 11 100 001	DD E1	2	4	14	
POP IY	IY_H ← (SP+1) IY_L ← (SP) SP ← SP +2	•	•	X	•	X	•	•	•	11 111 101 11 100 001	FD E1	2	4	14	

NOTES: dd is any of the register pairs BC, DE, HL, SP.
qq is any of the register pairs AF, BC, DE, HL.
$(PAIR)_H$, $(PAIR)_L$ refer to high order and low order eight bits of the register pair respectively,
e.g., BC_L = C, AF_H = A.

Exchange, Block Transfer, Block Search Groups

Mnemonic	Symbolic Operation	S	Z		H		P/V	N	C	Opcode 76 543 210	Hex	No. of Bytes	No. of M Cycles	No. of T States	Comments
EX DE, HL	DE ↔ HL	•	•	X	•	X	•	•	•	11 101 011	EB	1	1	4	
EX AF, AF'	AF ↔ AF'	•	•	X	•	X	•	•	•	00 001 000	08	1	1	4	
EXX	BC ↔ BC' DE ↔ DE' HL ↔ HL'	•	•	X	•	X	•	•	•	11 011 001	D9	1	1	4	Register bank and auxiliary register bank exchange
EX (SP), HL	H ↔ (SP+1) L ↔ (SP)	•	•	X	•	X	•	•	•	11 100 011	E3	1	5	19	
EX (SP), IX	IX_H ↔ (SP+1) IX_L ↔ (SP)	•	•	X	•	X	•	•	•	11 011 101 11 100 011	DD E3	2	6	23	
EX (SP), IY	IY_H ↔ (SP+1) IY_L ↔ (SP)	•	•	X	•	X	•	•	•	11 111 101 11 100 011	FD E3	2	6	23	
LDI	(DE) ← (HL) DE ← DE+1 HL ← HL+1 BC ← BC−1	•	•	X	0	X	① ↕	0	•	11 101 101 10 100 000	ED A0	2	4	16	Load (HL) into (DE), increment the pointers and decrement the byte counter (BC)
LDIR	(DE) ← (HL) DE ← DE+1 HL ← HL+1 BC ← BC−1 Repeat until BC = 0	•	•	X	0	X	0	0	•	11 101 101 10 110 000	ED B0	2 2	5 4	21 16	If BC ≠ 0 If BC = 0

NOTE: ① P/V flag is 0 if the result of BC−1 = 0, otherwise P/V = 1

Exchange, Block Transfer, Block Search Groups (Continued)

Mnemonic	Symbolic Operation	S	Z		H		P/V	N	C	Opcode 76 543 210	Hex	No.of Bytes	No.of M Cycles	No.of T States	Comments
LDD	(DE) ← (HL) DE ← DE − 1 HL ← HL − 1 BC ← BC − 1	•	•	X	0	X	↕ ①	0	•	11 101 101 10 101 000	ED A8	2	4	16	
LDDR	(DE) ← (HL) DE ← DE − 1 HL ← HL − 1 BC ← BC − 1 Repeat until BC = 0	•	•	X	0	X	0	0	•	11 101 101 10 111 000	ED B8	2 2	5 4	21 16	If BC ≠ 0 If BC = 0
CPI	A − (HL) HL ← HL + 1 BC ← BC − 1	↕	↕ ②	X	↕	X	↕ ①	1	•	11 101 101 10 100 001	ED A1	2	4	16	
CPIR	A − (HL) HL ← HL + 1 BC ← BC − 1 Repeat until A = (HL) or BC = 0	↕	↕ ②	X	↕	X	↕ ①	1	•	11 101 101 10 110 001	ED B1	2 2	5 4	21 16	If BC ≠ 0 and A ≠ (HL) If BC = 0 or A = (HL)
CPD	A − (HL) HL ← HL − 1 BC ← BC − 1	↕	↕ ②	X	↕	X	↕ ①	1	•	11 101 101 10 101 001	ED A9	2	4	16	
CPDR	A − (HL) HL ← HL − 1 BC ← BC − 1 Repeat until A = (HL) or BC = 0	↕	↕ ②	X	↕	X	↕ ①	1	•	11 101 101 10 111 001	ED B9	2 2	5 4	21 16	If BC ≠ 0 and A ≠ (HL) If BC = 0 or A = (HL)

NOTES ① P/V flag is 0 if the result of BC − 1 = 0, otherwise P/V = 1
② Z flag is 1 if A = (HL), otherwise Z = 0

8-Bit Arithmetic and Logical Group

Mnemonic	Symbolic Operation	S	Z		H		P/V	N	C	Opcode 76 543 210	Hex	No.of Bytes	No.of M Cycles	No.of T States	Comments
ADD A, r	A ← A + r	↕	↕	X	↕	X	V	0	↕	10 000 r		1	1	4	r Reg. 000 B 001 C 010 D 011 E 100 H 101 L 111 A
ADD A, n	A ← A + n	↕	↕	X	↕	X	V	0	↕	11 000 110 ← n →		2	2	7	
ADD A, (HL)	A ← A + (HL)	↕	↕	X	↕	X	V	0	↕	10 000 110		1	2	7	
ADD A, (IX+d)	A ← A + (IX+d)	↕	↕	X	↕	X	V	0	↕	11 011 101 10 000 110 ← d →	DD	3	5	19	
ADD A, (IY+d)	A ← A + (IY+d)	↕	↕	X	↕	X	V	0	↕	11 111 101 10 000 110 ← d →	FD	3	5	19	
ADC A, s	A ← A+s+CY	↕	↕	X	↕	X	V	0	↕	001					s is any of r, n, (HL), (IX+d), (IY+d) as shown for ADD instruction. The indicated bits replace the 000 in the ADD set above.
SUB s	A ← A−s	↕	↕	X	↕	X	V	1	↕	010					
SBC A, s	A ← A−s−CY	↕	↕	X	↕	X	V	1	↕	011					
AND s	A ← A ∧ s	↕	↕	X	1	X	P	0	0	100					
OR s	A ← A ∨ s	↕	↕	X	0	X	P	0	0	110					
XOR s	A ← A ⊕ s	↕	↕	X	0	X	P	0	0	101					
CP s	A−s	↕	↕	X	↕	X	V	1	↕	111					
INC r	r ← r + 1	↕	↕	X	↕	X	V	0	•	00 r 100		1	1	4	
INC (HL)	(HL) ← (HL) + 1	↕	↕	X	↕	X	V	0	•	00 110 100		1	3	11	
INC (IX+d)	(IX+d) ← (IX+d) + 1	↕	↕	X	↕	X	V	0	•	11 011 101 00 110 100 ← d →	DD	3	6	23	
INC (IY+d)	(IY+d) ← (IY+d) + 1	↕	↕	X	↕	X	V	0	•	11 111 101 00 110 100 ← d →	FD	3	6	23	
DEC m	m ← m − 1	↕	↕	X	↕	X	V	1	•	101					m is any of r, (HL), (IX+d), (IY+d) as shown for INC. DEC same format and states as INC. Replace 100 with 101 in opcode.

General-Purpose Arithmetic and CPU Control Groups

Mnemonic	Symbolic Operation	S	Z		H		P/V	N	C	Opcode 76 543 210	Hex	No. of Bytes	No. of M Cycles	No. of T States	Comments
DAA	Converts acc. content into packed BCD following add or subtract with packed BCD operands.	↕	↕	X	↕	X	P	•	↕	00 100 111	27	1	1	4	Decimal adjust accumulator.
CPL	A ← $\overline{A}$	•	•	X	1	X	•	1	•	00 101 111	2F	1	1	4	Complement accumulator (one's complement).
NEG	A ← 0 − A	↕	↕	X	↕	X	V	1	↕	11 101 101 01 000 100	ED 44	2	2	8	Negate acc. (two's complement).
CCF	CY ← $\overline{CY}$	•	•	X	X	X	•	0	↕	00 111 111	3F	1	1	4	Complement carry flag.
SCF	CY ← 1	•	•	X	0	X	•	0	1	00 110 111	37	1	1	4	Set carry flag.
NOP	No operation	•	•	X	•	X	•	•	•	00 000 000	00	1	1	4	
HALT	CPU halted	•	•	X	•	X	•	•	•	01 110 110	76	1	1	4	
DI ✱	IFF ← 0	•	•	X	•	X	•	•	•	11 110 011	F3	1	1	4	
EI ✱	IFF ← 1	•	•	X	•	X	•	•	•	11 111 011	FB	1	1	4	
IM 0	Set interrupt mode 0	•	•	X	•	X	•	•	•	11 101 101 01 000 110	ED 46	2	2	8	
IM 1	Set interrupt mode 1	•	•	X	•	X	•	•	•	11 101 101 01 010 110	ED 56	2	2	8	
IM 2	Set interrupt mode 2	•	•	X	•	X	•	•	•	11 101 101 01 011 110	ED 5E	2	2	8	

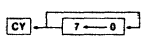
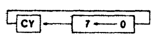
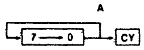
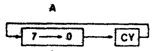
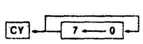
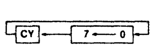
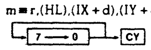
NOTES: IFF indicates the interrupt enable flip-flop.
CY indicates the carry flip-flop.
✱ indicates interrupts are not sampled at the end of EI or DI.

16-Bit Arithmetic Group

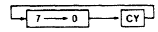
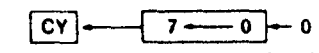
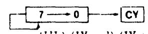
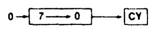
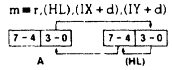
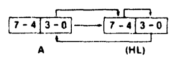
Mnemonic	Symbolic Operation	S	Z		H		P/V	N	C	Opcode 76 543 210	Hex	No. of Bytes	No. of M Cycles	No. of T States	Comments
ADD HL, ss	HL ← HL+ss	•	•	X	X	X	•	0	↕	00 ss1 001		1	3	11	ss Reg. 00 BC 01 DE 10 HL 11 SP
ADC HL, ss	HL ← HL+ss+CY	↕	↕	X	X	X	V	0	↕	11 101 101 01 ss1 010	ED	2	4	15	
SBC HL, ss	HL ← HL−ss−CY	↕	↕	X	X	X	V	1	↕	11 101 101 01 ss0 010	ED	2	4	15	
ADD IX, pp	IX ← IX + pp	•	•	X	X	X	•	0	↕	11 011 101 01 pp1 001	DD	2	4	15	pp Reg. 00 BC 01 DE 10 IX 11 SP
ADD IY, rr	IY ← IY + rr	•	•	X	X	X	•	0	↕	11 111 101 00 rr1 001	FD	2	4	15	rr Reg. 00 BC 01 DE 10 IY 11 SP
INC ss	ss ← ss + 1	•	•	X	•	X	•	•	•	00 ss0 011		1	1	6	
INC IX	IX ← IX + 1	•	•	X	•	X	•	•	•	11 011 101 00 100 011	DD 23	2	2	10	
INC IY	IY ← IY + 1	•	•	X	•	X	•	•	•	11 111 101 00 100 011	FD 23	2	2	10	
DEC ss	ss ← ss − 1	•	•	X	•	X	•	•	•	00 ss1 011		1	1	6	
DEC IX	IX ← IX − 1	•	•	X	•	X	•	•	•	11 011 101 00 101 011	DD 2B	2	2	10	
DEC IY	IY ← IY − 1	•	•	X	•	X	•	•	•	11 111 101 00 101 011	FD 2B	2	2	10	

NOTES: ss is any of the register pairs BC, DE, HL, SP
pp is any of the register pairs BC, DE, IX, SP.
rr is any of the register pairs BC, DE, IY, SP

Rotate and Shift Group

Mnemonic	Symbolic Operation	S	Z		H		P/V	N	C	Opcode 76 543 210	Hex	No. of Bytes	No. of M Cycles	No. of T States	Comments
RLCA		•	•	X	0	X	•	0	↕	00 000 111	07	1	1	4	Rotate left circular accumulator.
RLA		•	•	X	0	X	•	0	↕	00 010 111	17	1	1	4	Rotate left accumulator.
RRCA		•	•	X	0	X	•	0	↕	00 001 111	0F	1	1	4	Rotate right circular accumulator.
RRA		•	•	X	0	X	•	0	↕	00 011 111	1F	1	1	4	Rotate right accumulator.
RLC r		↕	↕	X	0	X	P	0	↕	11 001 011 00 000 r	CB	2	2	8	Rotate left circular register r.
RLC (HL)		↕	↕	X	0	X	P	0	↕	11 001 011 00 000 110	CB	2	4	15	r Reg. 000 B 001 C 010 D 011 E 100 H 101 L 111 A
RLC (IX+d)	 r,(HL),(IX+d),(IY+d)	↕	↕	X	0	X	P	0	↕	11 011 101 11 001 011 – d – 00 000 110	DD CB	4	6	23	
RLC (IY+d)		↕	↕	X	0	X	P	0	↕	11 111 101 11 001 011 – d – 00 000 110	FD CB	4	6	23	
RL m	 m ≡ r,(HL),(IX+d),(IY+d)	↕	↕	X	0	X	P	0	↕	010					Instruction format and states are as shown for RLC's. To form new opcode replace 000 or RLC's with shown code.
RRC m	 m ≡ r,(HL),(IX+d),(IY+d)	↕	↕	X	0	X	P	0	↕	001					

Rotate and Shift Group (Continued)

Mnemonic	Symbolic Operation	S	Z		H		P/V	N	C	Opcode 76 543 210	Hex	No. of Bytes	No. of M Cycles	No. of T States	Comments
RR m	 m ≡ r,(HL),(IX + d),(IY + d)	↕	↕	X	0	X	P	0	↕	$\boxed{011}$					
SLA m	 m ≡ r,(HL),(IX + d),(IY + d)	↕	↕	X	0	X	P	0	↕	$\boxed{100}$					
SRA m	 m ≡ r,(HL),(IX + d),(IY + d)	↕	↕	X	0	X	P	0	↕	$\boxed{101}$					
SRL m	 m ≡ r,(HL),(IX + d),(IY + d)	↕	↕	X	0	X	P	0	↕	$\boxed{111}$					
RLD		↕	↕	X	0	X	P	0	•	11 101 101 01 101 111	ED 6F	2	5	18	Rotate digit left and right between the accumulator and location (HL)
RRD		↕	↕	X	0	X	P	0	•	11 101 101 01 100 111	ED 67	2	5	18	The content of the upper half of the accumulator is unaffected

Bit Set, Reset and Test Group

Mnemonic	Symbolic Operation	S	Z		H		P/V	N	C	Opcode 76 543 210	Hex	No. of Bytes	No. of M Cycles	No. of T States	Comments
BIT b, r	$Z \leftarrow \overline{r}_b$	X	↕	X	1	X	X	0	•	11 001 011 01 b r	CB	2	2	8	r Reg. 000 B 001 C
BIT b, (HL)	$Z \leftarrow \overline{(HL)}_b$	X	↕	X	1	X	X	0	•	11 001 011 01 b 110	CB	2	3	12	010 D
BIT b, $(IX+d)_b$	$Z \leftarrow \overline{(IX+d)}_b$	X	↕	X	1	X	X	0	•	11 011 101 11 001 011 ← d → 01 b 110	DD CB	4	5	20	011 E 100 H 101 L 111 A b Bit Tested
BIT b, $(IY+d)_b$	$Z \leftarrow \overline{(IY+d)}_b$	X	↕	X	1	X	X	0	•	11 111 101 11 001 011 ← d → 01 b 110	FD CB	4	5	20	000 0 001 1 010 2 011 3 100 4 101 5 110 6 111 7
SET b, r	$r_b \leftarrow 1$	•	•	X	•	X	•	•	•	11 001 011 $\boxed{11}$ b r	CB	2	2	8	
SET b, (HL)	$(HL)_b \leftarrow 1$	•	•	X	•	X	•	•	•	11 001 011 $\boxed{11}$ b 110	CB	2	4	15	
SET b, (IX + d)	$(IX+d)_b \leftarrow 1$	•	•	X	•	X	•	•	•	11 011 101 11 001 011 ← d → $\boxed{11}$ b 110	DD CB	4	6	23	
SET b, (IY + d)	$(IY+d)_b \leftarrow 1$	•	•	X	•	X	•	•	•	11 111 101 11 001 011 ← d → $\boxed{11}$ b 110	FD CB	4	6	23	
RES b, m	$m_b \leftarrow 0$ m ≡ r, (HL), (IX+d), (IY+d)	•	•	X	•	X	•	•	•	$\boxed{10}$					To form new opcode replace $\boxed{11}$ of SET b, s with $\boxed{10}$ Flags and time states for SET instruction

NOTES The notation m_b indicates bit b (0 to 7) or location m

Jump Group

Mnemonic	Symbolic Operation	S	Z		H		P/V	N	C	Opcode 76 543 210	Hex	No. of Bytes	No. of M Cycles	No. of T States	Comments
JP nn	PC ← nn	•	•	X	•	X	•	•	•	11 000 011 ← n → ← n →	C3	3	3	10	
JP cc, nn	If condition cc is true PC ← nn, otherwise continue	•	•	X	•	X	•	•	•	11 cc 010 ← n → ← n →		3	3	10	cc Condition 000 NZ non-zero 001 Z zero 010 NC non-carry 011 C carry 100 PO parity odd 101 PE parity even 110 P sign positive
JR e	PC ← PC+e	•	•	X	•	X	•	•	•	00 011 000 ← e−2 →	18	2	3	12	111 M sign negative
JR C, e	If C = 0, continue	•	•	X	•	X	•	•	•	00 111 000 ← e−2 →	38	2	2	7	If condition not met.
	If C = 1, PC ← PC+e											2	3	12	If condition is met.
JR NC, e	If C = 1, continue	•	•	X	•	X	•	•	•	00 110 000 ← e−2 →	30	2	2	7	If condition not met.
	If C = 0, PC ← PC+e											2	3	12	If condition is met.
JP Z, e	If Z = 0 continue	•	•	X	•	X	•	•	•	00 101 000 ← e−2 →	28	2	2	7	If condition not met.
	If Z = 1, PC ← PC+e											2	3	12	If condition is met.
JR NZ, e	If Z = 1, continue	•	•	X	•	X	•	•	•	00 100 000 ← e−2 →	20	2	2	7	If condition not met
	If Z = 0, PC ← PC+e											2	3	12	If condition is met
JP (HL)	PC ← HL	•	•	X	•	X	•	•	•	11 101 001	E9	1	1	4	
JP (IX)	PC ← IX	•	•	X	•	X	•	•	•	11 011 101 11 101 001	DD E9	2	2	8	

Jump Group (Continued)

Mnemonic	Symbolic Operation	S	Z		H		P/V	N	C	Opcode 76 543 210	Hex	No. of Bytes	No. of M Cycles	No. of T States	Comments
JP (IY)	PC ← IY	•	•	X	•	X	•	•	•	11 111 101 11 101 001	FD E9	2	2	8	
DJNZ, e	B ← B−1 If B = 0, continue	•	•	X	•	X	•	•	•	00 010 000 ← e−2 →	10	2	2	8	If B = 0.
	If B ≠ 0, PC ← PC+e											2	3	13	If B ≠ 0.

NOTES: e represents the extension in the relative addressing mode.
e is a signed two's complement number in the range < −126, 129 >.
e−2 in the opcode provides an effective address of pc+e as PC is incremented by 2 prior to the addition of e.

Call and Return Group

Mnemonic	Symbolic Operation	S	Z		H		P/V	N	C	Opcode 76 543 210	Hex	No. of Bytes	No. of M Cycles	No. of T States	Comments
CALL nn	(SP−1) ← PC_H (SP−2) ← PC_L PC ← nn	•	•	X	•	X	•	•	•	11 001 101 ← n → ← n →	CD	3	5	17	
CALL cc, nn	If condition cc is false continue, otherwise same as CALL nn	•	•	X	•	X	•	•	•	11 cc 100 ← n → ← n →		3 3	3 5	10 17	If cc is false. If cc is true.
RET	PC_L ← (SP) PC_H ← (SP+1)	•	•	X	•	X	•	•	•	11 001 001	C9	1	3	10	
RET cc	If condition cc is false continue, otherwise same as RET	•	•	X	•	X	•	•	•	11 cc 000		1 1	1 3	5 11	If cc is false. If cc is true.
RETI	Return from interrupt	•	•	X	•	X	•	•	•	11 101 101 01 001 101	ED 4D	2	4	14	
RETN[1]	Return from non-maskable interrupt	•	•	X	•	X	•	•	•	11 101 101 01 000 101	ED 45	2	4	14	
RST p	(SP−1) ← PC_H (SP−2) ← PC_L PC_H ← 0 PC_L ← p	•	•	X	•	X	•	•	•	11 t 111		1	3	11	

cc	Condition
000 NZ	non-zero
001 Z	zero
010 NC	non-carry
011 C	carry
100 PO	parity odd
101 PE	parity even
110 P	sign positive
111 M	sign negative

t	p
000	00H
001	08H
010	10H
011	18H
100	20H
101	28H
110	30H
111	38H

NOTE: [1]RETN loads IFF_2 ← IFF_1

Input and Output Group

Mnemonic	Symbolic Operation	S	Z		H		P/V	N	C	Opcode 76 543 210	Hex	No. of Bytes	No. of M Cycles	No. of T States	Comments
IN A, (n)	A ← (n)	•	•	X	•	X	•	•	•	11 011 011 ← n →	DB	2	3	11	n to A_0 ~ A_7 Acc. to A_8 ~ A_{15}
IN r, (C)	r ← (C) if r = 110 only the flags will be affected	↕	↕	X	↕	X	P	0	•	11 101 101 01 r 000	ED	2	3	12	C to A_0 ~ A_7 B to A_8 ~ A_{15}
INI	(HL) ← (C) B ← B−1 HL ← HL + 1	X	↕ (1)	X	X	X	X	1	•	11 101 101 10 100 010	ED A2	2	4	16	C to A_0 ~ A_7 B to A_8 ~ A_{15}
INIR	(HL) ← (C) B ← B−1 HL ← HL + 1 Repeat until B = 0	X	1	X	X	X	X	1	•	11 101 101 10 110 010	ED B2	2 2	5 (If B≠0) 4 (If B=0)	21 16	C to A_0 ~ A_7 B to A_8 ~ A_{15}
IND	(HL) ← (C) B ← B−1 HL ← HL−1	X	↕ (1)	X	X	X	X	1	•	11 101 101 10 101 010	ED AA	2	4	16	C to A_0 ~ A_7 B to A_8 ~ A_{15}
INDR	(HL) ← (C) B ← B−1 HL ← HL−1 Repeat until B = 0	X	1	X	X	X	X	1	•	11 101 101 10 111 010	ED BA	2 2	5 (If B≠0) 4 (If B=0)	21 16	C to A_0 ~ A_7 B to A_8 ~ A_{15}
OUT (n), A	(n) ← A	•	•	X	•	X	•	•	•	11 010 011 ← n →	D3	2	3	11	n to A_0 ~ A_7 Acc. to A_8 ~ A_{15}
OUT (C), r	(C) ← r	•	•	X	•	X	•	•	•	11 101 101 01 r 001	ED	2	3	12	C to A_0 ~ A_7 B to A_8 ~ A_{15}
OUTI	(C) ← (HL) B ← B−1 HL ← HL + 1	X	↕ (1)	X	X	X	X	1	•	11 101 101 10 100 011	ED A3	2	4	16	C to A_0 ~ A_7 B to A_8 ~ A_{15}
OTIR	(C) ← (HL) B ← B−1 HL ← HL + 1 Repeat until B = 0	X	1	X	X	X	X	1	•	11 101 101 10 110 011	ED B3	2 2	5 (If B≠0) 4 (If B=0)	21 16	C to A_0 ~ A_7 B to A_8 ~ A_{15}
OUTD	(C) ← (HL) B ← B−1 HL ← HL−1	X	↕ (1)	X	X	X	X	1	•	11 101 101 10 101 011	ED AB	2	4	16	C to A_0 ~ A_7 B to A_8 ~ A_{15}

NOTE: (1) If the result of B − 1 is zero the Z flag is set, otherwise it is reset.

Input and Output Group (Continued)

Mnemonic	Symbolic Operation	S	Z		H		P/V	N	C	Opcode 76 543 210	Hex	No. of Bytes	No. of M Cycles	No. of T States	Comments
OTDR	(C) ← (HL) B ← B−1 HL ← HL−1 Repeat until B = 0	X	1	X	X	X	X	1	•	11 101 101 10 111 011	ED	2 2	5 (If B≠0) 4 (If B=0)	21 16	C to $A_0 \sim A_7$ B to $A_8 \sim A_{15}$

Summary of Flag Operation

Instruction	D_7 S	Z		H		P/V	N	D_0 C	Comments
ADD A, s; ADC A, s	↕	↕	X	↕	X	V	0	↕	8-bit add or add with carry.
SUB s; SBC A, s; CP s; NEG	↕	↕	X	↕	X	V	1	↕	8-bit subtract, subtract with carry, compare and negate accumulator.
AND s	↕	↕	X	1	X	P	0	0	Logical operations
OR s, XOR s	↕	↕	X	0	X	P	0	0	Logical operations
INC s	↕	↕	X	↕	X	V	0	•	8-bit increment.
DEC s	↕	↕	X	↕	X	V	1	•	8-bit decrement.
ADD DD, ss	•	•	X	X	X	•	0	↕	16-bit add.
ADC HL, ss	↕	↕	X	X	X	V	0	↕	16-bit add with carry.
SBC HL, ss	↕	↕	X	X	X	V	1	↕	16-bit subtract with carry.
RLA, RLCA, RRA; RRCA	•	•	X	0	X	•	0	↕	Rotate accumulator.
RL m; RLC m; RR m; RRC m; SLA m; SRA m; SRL m	↕	↕	X	0	X	P	0	↕	Rotate and shift locations.
RLD; RRD	↕	↕	X	0	X	P	0	•	Rotate digit left and right
DAA	↕	↕	X	↕	X	P	•	↕	Decimal adjust accumulator.
CPL	•	•	X	1	X	•	1	•	Complement accumulator.
SCF	•	•	X	0	X	•	0	1	Set carry.
CCF	•	•	X	X	X	•	0	↕	Complement carry.
IN r (C)	↕	↕	X	0	X	P	0	•	Input register indirect.
INI, IND, OUTI; OUTD	X	↕	X	X	X	X	1	•	Block input and output. Z = 0 if B ≠ 0 otherwise Z = 0.
INIR; INDR; OTIR; OTDR	X	1	X	X	X	X	1	•	Block input and output. Z = 0 if B ≠ 0 otherwise Z = 0.
LDI; LDD	X	X	X	0	X	↕	0	•	Block transfer instructions. P/V = 1 if BC ≠ 0, otherwise P/V = 0.
LDIR; LDDR	X	X	X	0	X	0	0	•	Block transfer instructions. P/V = 1 if BC ≠ 0, otherwise P/V = 0.
CPI; CPIR; CPD; CPDR	X	↕	X	X	X	↕	1	•	Block search instructions. Z = 1 if A = (HL), otherwise Z = 0. P/V = 1 if BC ≠ 0, otherwise P/V = 0.
LD A, I, LD A, R	↕	↕	X	0	X	IFF	0	•	The content of the interrupt enable flip-flop (IFF) is copied into the P/V flag.
BIT b, s	X	↕	X	1	X	X	0	•	The state of bit b of location s is copied into the Z flag.

Symbolic Notation

Symbol	Operation
S	Sign flag. S = 1 if the MSB of the result is 1.
Z	Zero flag. Z = 1 if the result of the operation is 0.
P/V	Parity or overflow flag. Parity (P) and overflow (V) share the same flag. Logical operations affect this flag with the parity of the result while arithmetic operations affect this flag with the overflow of the result. If P/V holds parity, P/V = 1 if the result of the operation is even, P/V = 0 if result is odd. If P/V holds overflow, P/V = 1 if the result of the operation produced an overflow.
H	Half-carry flag. H = 1 if the add or subtract operation produced a carry into or borrow from bit 4 of the accumulator.
N	Add/Subtract flag. N = 1 if the previous operation was a subtract.
H & N	H and N flags are used in conjunction with the decimal adjust instruction (DAA) to properly correct the result into packed BCD format following addition or subtraction using operands with packed BCD format.
C	Carry/Link flag. C = 1 if the operation produced a carry from the MSB of the operand or result.

Symbol	Operation
↕	The flag is affected according to the result of the operation.
•	The flag is unchanged by the operation.
0	The flag is reset by the operation.
1	The flag is set by the operation.
X	The flag is a "don't care."
V	P/V flag affected according to the overflow result of the operation.
P	P/V flag affected according to the parity result of the operation.
r	Any one of the CPU registers A, B, C, D, E, H, L.
s	Any 8-bit location for all the addressing modes allowed for the particular instruction.
ss	Any 16-bit location for all the addressing modes allowed for that instruction.
ii	Any one of the two index registers IX or IY.
R	Refresh counter.
n	8-bit value in range < 0, 255 >.
nn	16-bit value in range < 0, 65535 >.

Pin Descriptions

A_0-A_{15}. *Address Bus* (output, active High, 3-state). A_0-A_{15} form a 16-bit address bus. The Address Bus provides the address for memory data bus exchanges (up to 64K bytes) and for I/O device exchanges.

$\overline{\text{BUSACK}}$. *Bus Acknowledge* (output, active Low). Bus Acknowledge indicates to the requesting device that the CPU address bus, data bus, and control signals $\overline{\text{MREQ}}$, $\overline{\text{IORQ}}$, $\overline{\text{RD}}$, and $\overline{\text{WR}}$ have entered their high-impedance states. The external circuitry can now control these lines.

$\overline{\text{BUSREQ}}$. *Bus Request* (input, active Low). Bus Request has a higher priority than $\overline{\text{NMI}}$ and is always recognized at the end of the current machine cycle. $\overline{\text{BUSREQ}}$ forces the CPU address bus, data bus, and control signals $\overline{\text{MREQ}}$, $\overline{\text{IORQ}}$, $\overline{\text{RD}}$, and $\overline{\text{WR}}$ to go to a high-impedance state so that other devices can control these lines. $\overline{\text{BUSREQ}}$ is normally wire-ORed and requires an external pullup for these applications. Extended $\overline{\text{BUSREQ}}$ periods due to extensive DMA operations can prevent the CPU from properly refreshing dynamic RAMs.

D_0-D_7. *Data Bus* (input/output, active High, 3-state). D_0-D_7 constitute an 8-bit bidirectional data bus, used for data exchanges with memory and I/O.

$\overline{\text{HALT}}$. *Halt State* (output, active Low). $\overline{\text{HALT}}$ indicates that the CPU has executed a Halt instruction and is awaiting either a non-maskable or a maskable interrupt (with the mask enabled) before operation can resume. While halted, the CPU executes NOPs to maintain memory refresh.

$\overline{\text{INT}}$. *Interrupt Request* (input, active Low). Interrupt Request is generated by I/O devices. The CPU honors a request at the end of the current instruction if the internal software-controlled interrupt enable flip-flop (IFF) is enabled. $\overline{\text{INT}}$ is normally wire-ORed and requires an external pullup for these applications.

$\overline{\text{IORQ}}$. *Input/Output Request* (output, active Low, 3-state). $\overline{\text{IORQ}}$ indicates that the lower half of the address bus holds a valid I/O address for an I/O read or write operation. $\overline{\text{IORQ}}$ is also generated concurrently with $\overline{\text{M1}}$ during an interrupt acknowledge cycle to indicate that an interrupt response vector can be placed on the data bus.

$\overline{\text{M1}}$. *Machine Cycle One* (output, active Low). $\overline{\text{M1}}$, together with $\overline{\text{MREQ}}$, indicates that the current machine cycle is the opcode fetch cycle of an instruction execution. $\overline{\text{M1}}$, together with $\overline{\text{IORQ}}$, indicates an interrupt acknowledge cycle.

$\overline{\text{MREQ}}$. *Memory Request* (output, active Low, 3-state). $\overline{\text{MREQ}}$ indicates that the address bus holds a valid address for a memory read or memory write operation.

$\overline{\text{NMI}}$. *Non-Maskable Interrupt* (input, active Low). $\overline{\text{NMI}}$ has a higher priority than $\overline{\text{INT}}$. $\overline{\text{NMI}}$ is always recognized at the end of the current instruction, independent of the status of the interrupt enable flip-flop, and automatically forces the CPU to restart at location 0066H.

$\overline{\text{RD}}$. *Memory Read* (output, active Low, 3-state). $\overline{\text{RD}}$ indicates that the CPU wants to read data from memory or an I/O device. The addressed I/O device or memory should use this signal to gate data onto the CPU data bus.

$\overline{\text{RESET}}$. *Reset* (input, active Low). $\overline{\text{RESET}}$ initializes the CPU as follows: it resets the interrupt enable flip-flop, clears the PC and Registers I and R, and sets the interrupt status to Mode 0. During reset time, the address and data bus go to a high-impedance state, and all control output signals go to the inactive state. Note that $\overline{\text{RESET}}$ must be active for a minimum of three full clock cycles before the reset operation is complete.

$\overline{\text{RFSH}}$. *Refresh* (output, active Low). $\overline{\text{RFSH}}$, together with $\overline{\text{MREQ}}$, indicates that the lower seven bits of the system's address bus can be used as a refresh address to the system's dynamic memories.

$\overline{\text{WAIT}}$. *Wait* (input, active Low). $\overline{\text{WAIT}}$ indicates to the CPU that the addressed memory or I/O devices are not ready for a data transfer. The CPU continues to enter a Wait state as long as this signal is active. Extended $\overline{\text{WAIT}}$ periods can prevent the CPU from refreshing dynamic memory properly.

$\overline{\text{WR}}$. *Memory Write* (output, active Low, 3-state). $\overline{\text{WR}}$ indicates that the CPU data bus holds valid data to be stored at the addressed memory or I/O location.

CPU Timing

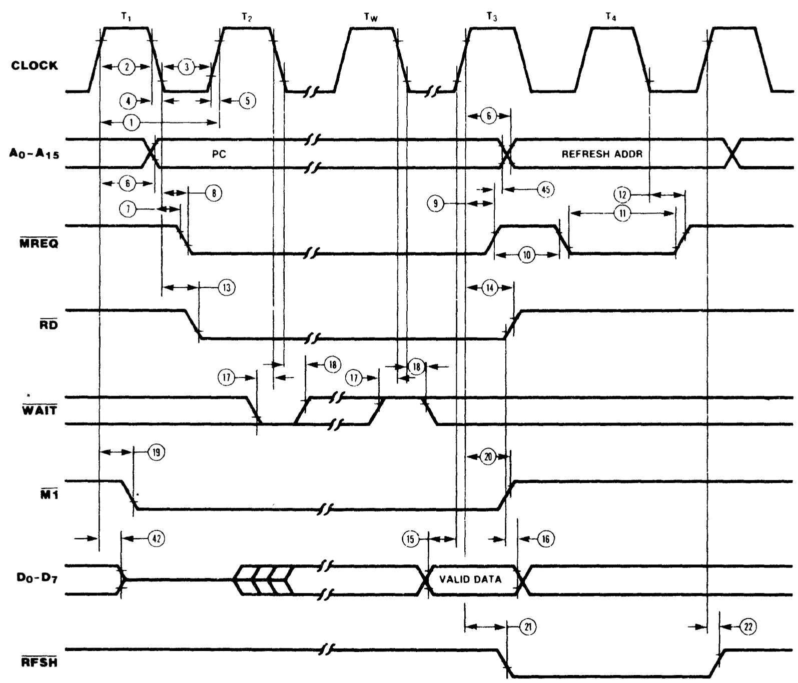
The Z80 CPU executes instructions by proceeding through a specific sequence of operations:

- Memory read or write
- I/O device read or write
- Interrupt acknowledge

The basic clock period is referred to as a T time or cycle, and three or more T cycles make up a machine cycle (M1, M2 or M3 for instance). Machine cycles can be extended either by the CPU automatically inserting one or more Wait states or by the insertion of one or more Wait states by the user.

Instruction Opcode Fetch. The CPU places the contents of the Program Counter (PC) on the address bus at the start of the cycle (Figure 5). Approximately one-half clock cycle later, $\overline{MREQ}$ goes active. The falling edge of $\overline{MREQ}$ can be used directly as a Chip Enable to dynamic memories. When active, $\overline{RD}$ indicates that the memory data can be enabled onto the CPU data bus.

The CPU samples the $\overline{WAIT}$ input with the rising edge of clock state T3. During clock states T3 and T4 of an M1 cycle dynamic RAM refresh can occur while the CPU starts decoding and executing the instruction. When the Refresh Control signal becomes active, refreshing of dynamic memory can take place.



NOTE: T_W-Wait cycle added when necessary for slow ancillary devices.

Figure 5. Instruction Opcode Fetch

CPU Timing (Continued)

Memory Read or Write Cycles. Figure 6 shows the timing of memory read or write cycles other than an opcode fetch ($\overline{M1}$) cycle. The $\overline{MREQ}$ and $\overline{RD}$ signals function exactly as in the fetch cycle. In a memory write cycle, $\overline{MREQ}$ also becomes active when the address bus is stable, so that it can be used directly as a Chip Enable for dynamic memories. The $\overline{WR}$ line is active when the data bus is stable, so that it can be used directly as an R/$\overline{W}$ pulse to most semiconductor memories.

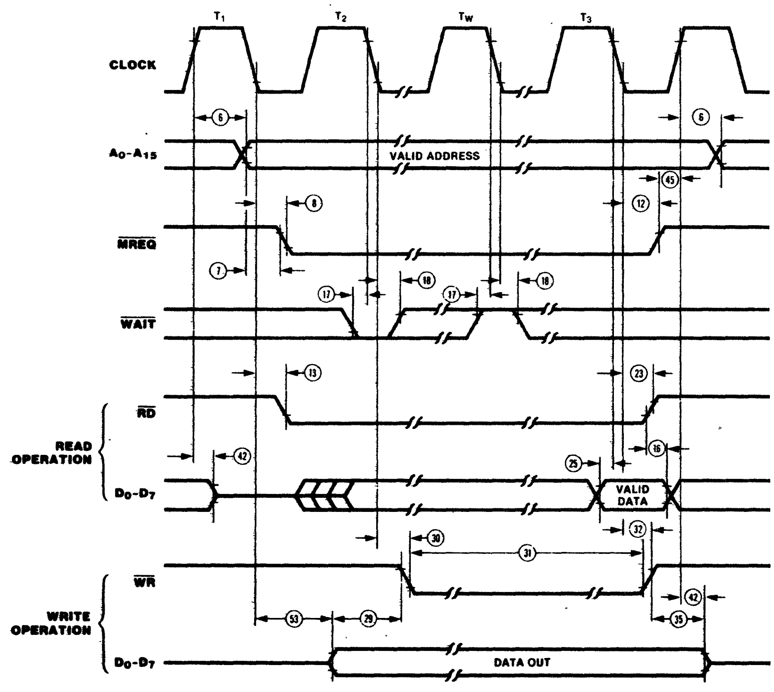
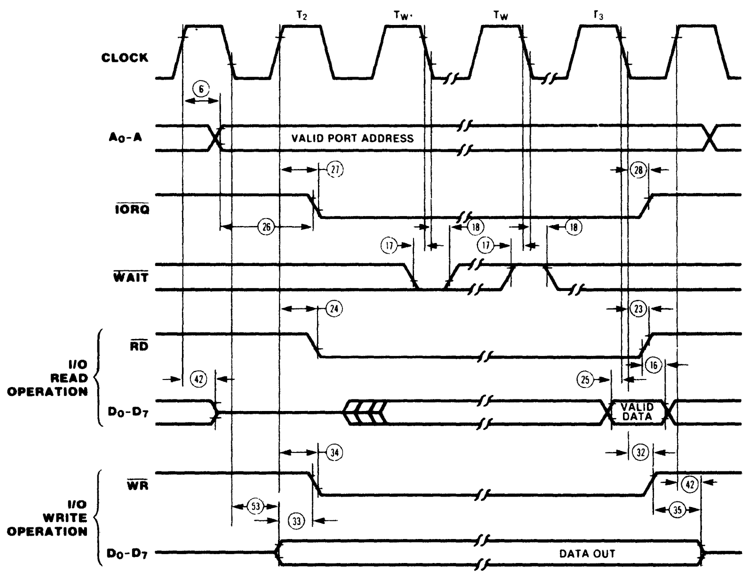


Figure 6. Memory Read or Write Cycles

CPU Timing (Continued)

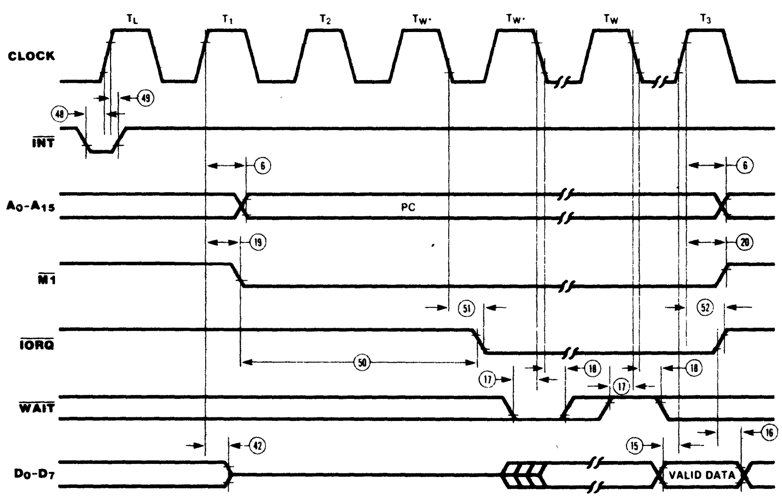
Input or Output Cycles. Figure 7 shows the timing for an I/O read or I/O write operation. During I/O operations, the CPU automatically inserts a single Wait state (T_W). This extra Wait state allows sufficient time for an I/O port to decode the address and the port address lines.



NOTE: T_W* = One Wait cycle automatically inserted by CPU.

Figure 7. Input or Output Cycles

Interrupt Request/Acknowledge Cycle. The CPU samples the interrupt signal with the rising edge of the last clock cycle at the end of any instruction (Figure 8). When an interrupt is accepted, a special $\overline{M1}$ cycle is generated. During this $\overline{M1}$ cycle, $\overline{IORQ}$ becomes active (instead of $\overline{MREQ}$) to indicate that the interrupting device can place an 8-bit vector on the data bus. The CPU automatically adds two Wait states to this cycle.

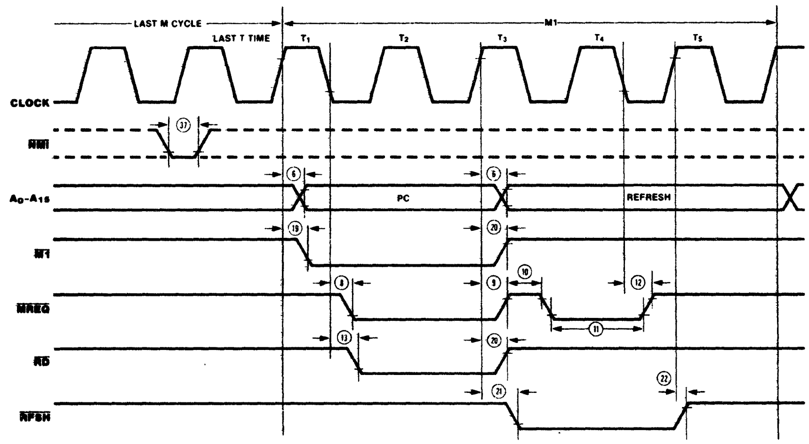


NOTE: 1) T_L = Last state of previous instruction. 2) Two Wait cycles automatically inserted by CPU(*).

Figure 8. Interrupt Request/Acknowledge Cycle

CPU Timing (Continued)

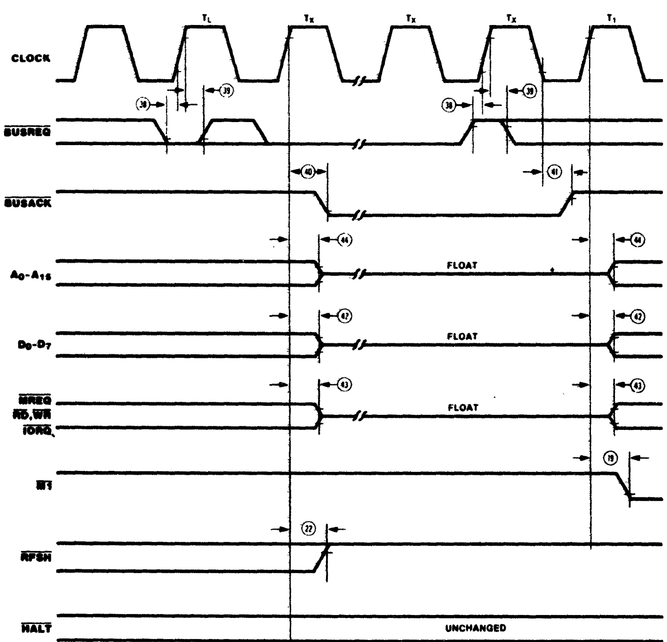
Non-Maskable Interrupt Request Cycle. $\overline{NMI}$ is sampled at the same time as the maskable interrupt input $\overline{INT}$ but has higher priority and cannot be disabled under software control. The subsequent timing is similar to that of a normal memory read operation except that data put on the bus by the memory is ignored. The CPU instead executes a restart (RST) operation and jumps to the $\overline{NMI}$ service routine located at address 0066H (Figure 9).



*Although $\overline{NMI}$ is an asynchronous input, to guarantee its being recognized on the following machine cycle, $\overline{NMI}$'s falling edge must occur no later than the rising edge of the clock cycle preceding T_{LAST}.

Figure 9. Non-Maskable Interrupt Request Operation

Bus Request/Acknowledge Cycle. The CPU samples $\overline{BUSREQ}$ with the rising edge of the last clock period of any machine cycle (Figure 10). If $\overline{BUSREQ}$ is active, the CPU sets its address, data, and $\overline{MREQ}$, $\overline{IORQ}$, $\overline{RD}$, and $\overline{WR}$ lines to a high-impedance state with the rising edge of the next clock pulse. At that time, any external device can take control of these lines, usually to transfer data between memory and I/O devices.



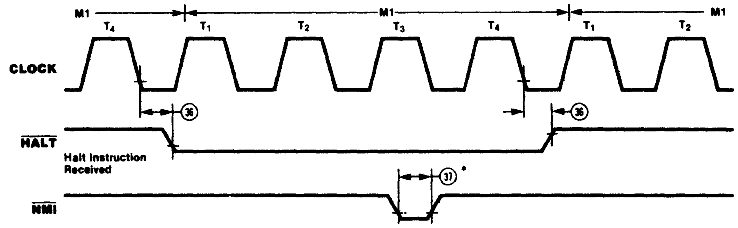
NOTE: T_L = Last state of any M cycle.

T_X = An arbitrary clock cycle used by requesting device.

Figure 10. Bus Request/Acknowledge Cycle

CPU Timing (Continued)

Halt Acknowledge Cycle. When the CPU receives a $\overline{HALT}$ instruction, it executes NOP states until either an $\overline{INT}$ or $\overline{NMI}$ input is received. When in the Halt state, the $\overline{HALT}$ output is active and remains so until an interrupt is processed (Figure 11).



NOTE: INT will also force a Halt exit.

*See note, Figure 9.

Figure 11. Halt Acknowledge Cycle

Reset Cycle. $\overline{RESET}$ must be active for at least three clock cycles for the CPU to properly accept it. As long as $\overline{RESET}$ remains active, the address and data buses float, and the control outputs are inactive. Once $\overline{RESET}$ goes inactive, two internal T cycles are consumed before the CPU resumes normal processing operation. $\overline{RESET}$ clears the PC register, so the first opcode fetch will be to location 0000 (Figure 12).

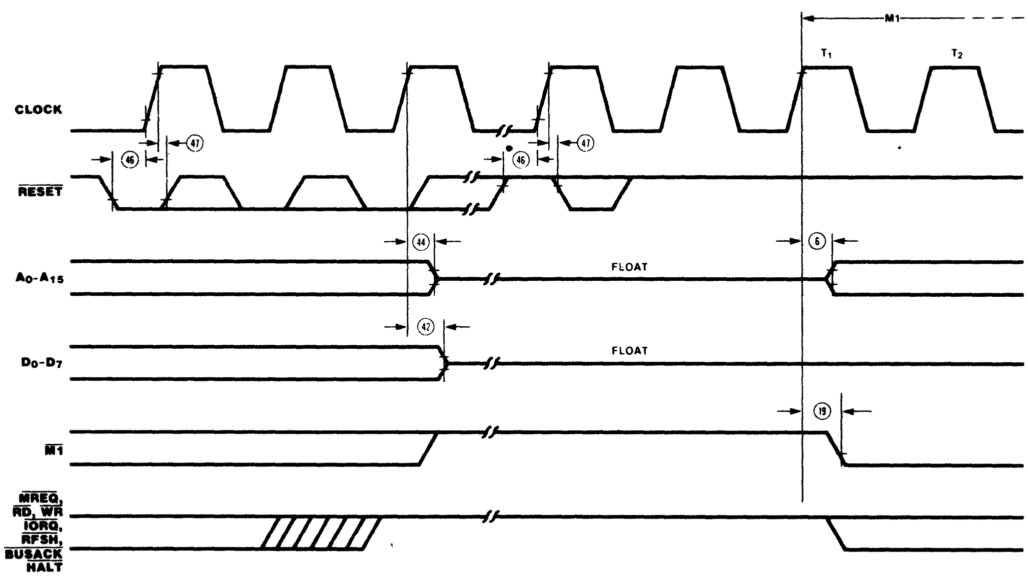


Figure 12. Reset Cycle

AC Characteristics

Number	Symbol	Parameter	Z80 CPU Min (ns)	Z80 CPU Max (ns)	Z80A CPU Min (ns)	Z80A CPU Max (ns)	Z80B CPU Min (ns)	Z80B CPU Max (ns)
1	TcC	Clock Cycle Time	400*		250*		165*	
2	TwCh	Clock Pulse Width (High)	180*		110*		65*	
3	TwCl	Clock Pulse Width (Low)	180	2000	110	2000	65	2000
4	TfC	Clock Fall Time	—	30	—	30	—	20
5	TrC	Clock Rise Time		30		30		20
6	TdCr(A)	Clock ↑ to Address Valid Delay	—	145	—	110	—	90
7	TdA(MREQf)	Address Valid to $\overline{MREQ}$ ↓ Delay	125*	—	65*	—	35*	—
8	TdCf(MREQf)	Clock ↓ to $\overline{MREQ}$ ↓ Delay	—	100	—	85	—	70
9	TdCr(MREQr)	Clock ↑ to $\overline{MREQ}$ ↑ Delay	—	100	—	85	—	70
10	TwMREQh	$\overline{MREQ}$ Pulse Width (High)	170*		110*		65*	
11	TwMREQl	$\overline{MREQ}$ Pulse Width (Low)	360*	—	220*	—	135*	—
12	TdCf(MREQr)	Clock ↓ to $\overline{MREQ}$ ↑ Delay	—	100	—	85	—	70
13	TdCf(RDf)	Clock ↓ to $\overline{RD}$ ↓ Delay	—	130	—	95	—	80
14	TdCr(RDr)	Clock ↑ to $\overline{RD}$ ↑ Delay	—	100	—	85	—	70
15	TsD(Cr)	Data Setup Time to Clock ↑	50		35		30	
16	ThD(RDr)	Data Hold Time to $\overline{RD}$ ↑	—	0	—	0	—	0
17	TsWAIT(Cf)	$\overline{WAIT}$ Setup Time to Clock ↓	70	—	70	—	60	—
18	ThWAIT(Cf)	$\overline{WAIT}$ Hold Time after Clock ↓	—	0	—	0	—	0
19	TdCr(M1f)	Clock ↑ to $\overline{M1}$ ↓ Delay	—	130	—	100	—	80
20	TdCr(M1r)	Clock ↑ to $\overline{M1}$ ↑ Delay		130		100		80
21	TdCr(RFSHf)	Clock ↑ to $\overline{RFSH}$ ↓ Delay	—	180	—	130	—	110
22	TdCr(RFSHr)	Clock ↑ to $\overline{RFSH}$ ↑ Delay	—	150	—	120	—	100
23	TdCf(RDr)	Clock ↓ to $\overline{RD}$ ↑ Delay	—	110	—	85	—	70
24	TdCr(RDf)	Clock ↑ to $\overline{RD}$ ↓ Delay	—	100	—	85	—	70
25	TsD(Cf)	Data Setup to Clock ↓ during M_2, M_3, M_4 or M_5 Cycles	60		50		40	
26	TdA(IORQf)	Address Stable prior to $\overline{IORQ}$ ↓	320*	—	180*	—	110*	—
27	TdCr(IORQf)	Clock ↑ to $\overline{IORQ}$ ↓ Delay	—	90	—	75	—	65
28	TdCf(IORQr)	Clock ↓ to $\overline{IORQ}$ ↑ Delay	—	110	—	85	—	70
29	TdD(WRf)	Data Stable prior to $\overline{WR}$ ↓	190*	—	80*	—	25*	—
30	TdCf(WRf)	Clock ↓ to $\overline{WR}$ ↓ Delay		90		80		70
31	TwWR	$\overline{WR}$ Pulse Width	360*	—	220*	—	135*	—
32	TdCf(WRr)	Clock ↓ to $\overline{WR}$ ↑ Delay	—	100	—	80	—	70
33	TdD(WRf)	Data Stable prior to $\overline{WR}$ ↓	20*	—	-10*	—	-55*	—
34	TdCr(WRf)	Clock ↑ to $\overline{WR}$ ↓ Delay	—	80	—	65	—	60
35	TdWRr(D)	Data Stable from $\overline{WR}$ ↑	120*		60*		30*	
36	TdCf(HALT)	Clock ↓ to $\overline{HALT}$ ↑ or ↓	—	300	—	300	—	260
37	TwNMI	$\overline{NMI}$ Pulse Width	80	—	80	—	70	—
38	TsBUSREQ(Cr)	$\overline{BUSREQ}$ Setup Time to Clock ↑	80	—	50	—	50	—

*For clock periods other than the minimums shown in the table, calculate parameters using the expressions in the table on the following page.

AC Characteristics (Continued)

Number	Symbol	Parameter	Z80 CPU Min (ns)	Z80 CPU Max (ns)	Z80A CPU Min (ns)	Z80A CPU Max (ns)	Z80B CPU Min (ns)	Z80B CPU Max (ns)
39	ThBUSREQ(Cr)	$\overline{BUSREQ}$ Hold Time after Clock ↑	0	—	0	—	0	—
40	TdCr(BUSACKf)	Clock ↑ to $\overline{BUSACK}$ ↓ Delay		120		100		90
41	TdCf(BUSACKr)	Clock ↓ to $\overline{BUSACK}$ ↑ Delay	—	110	—	100	—	90
42	TdCr(Dz)	Clock ↑ to Data Float Delay	—	90	—	90	—	80
43	TdCr(CTz)	Clock ↑ to Control Outputs Float Delay ($\overline{MREQ}$, $\overline{IORQ}$, $\overline{RD}$, and $\overline{WR}$)	—	110	—	80	—	70
44	TdCr(Az)	Clock ↑ to Address Float Delay	—	110	—	90	—	80
45	TdCTr(A)	Address Stable after $\overline{MREQ}$ ↑, $\overline{IORQ}$ ↑, $\overline{RD}$ ↑, and $\overline{WR}$ ↑	160*		80*		35*	
46	TsRESET(Cr)	$\overline{RESET}$ to Clock ↑ Setup Time	90	—	60	—	60	—
47	ThRESET(Cr)	$\overline{RESET}$ to Clock ↑ Hold Time	—	0	—	0	—	0
48	TsINTf(Cr)	$\overline{INT}$ to Clock ↑ Setup Time	80	—	80	—	70	—
49	ThINTr(Cr)	$\overline{INT}$ to Clock ↑ Hold Time	—	0	—	0	—	0
50	TdM1f(IORQf)	$\overline{M1}$ ↓ to $\overline{IORQ}$ ↓ Delay	920*		565*		365*	
51	TdCf(IORQf)	Clock ↓ to $\overline{IORQ}$ ↓ Delay	—	110	—	85	—	70
52	TdCf(IORQr)	Clock ↑ to $\overline{IORQ}$ ↑ Delay	—	100	—	85	—	70
53	TdCf(D)	Clock ↓ to Data Valid Delay	—	230	—	150	—	130

*For clock periods other than the minimums shown in the table, calculate parameters using the following expressions. Calculated values above assumed TrC = TfC = 20 ns.

Footnotes to AC Characteristics

Number	Symbol	Z80	Z80A	Z80B
1	TcC	TwCh + TwCl + TrC + TfC	TwCh + TwCl + TrC + TfC	TwCh + TwCl + TrC + TfC
2	TwCh	Although static by design, TwCh of greater than 200 μs is not guaranteed	Although static by design, TwCh of greater than 200 μs is not guaranteed	Although static by design, TwCh of greater than 200 μs is not guaranteed
7	TdA(MREQf)	TwCh + TfC − 75	TwCh + TfC − 65	TwCh + TfC − 50
10	TwMREQh	TwCh + TfC − 30	TwCh + TfC − 20	TwCh + TfC − 20
11	TwMREQl	TcC − 40	TcC − 30	TcC − 30
26	TdA(IORQf)	TcC − 80	TcC − 70	TcC − 55
29	TdD(WRf)	TcC − 210	TcC − 170	TcC − 140
31	TwWR	TcC − 40	TcC − 30	TcC − 30
33	TdD(WRf)	TwCl + TrC − 180	TwCl + TrC − 140	TwCl + TrC − 140
35	TdWRr(D)	TwCl + TrC − 80	TwCl + TrC − 70	TwCl + TrC − 55
45	TdCTr(A)	TwCl + TrC − 40	TwCl + TrC − 50	TwCl + TrC − 50
50	TdM1f(IORQf)	2TcC + TwCh + TfC − 80	2TcC + TwCh + TfC − 65	2TcC + TwCh + TfC − 50

AC Test Conditions:
V_{IH} = 2.0 V
V_{IL} = 0.8 V
V_{IHC} = V_{CC} −0.6 V
V_{ILC} = 0.45 V
V_{OH} = 2.0 V
V_{OL} = 0.8 V
FLOAT = ±0.5 V

Absolute Maximum Ratings

Storage Temperature .... -65°C to +150°C
Temperature under Bias .... Specified operating range
Voltages on all inputs and outputs with respect to ground . -0.3 V to +7 V
Power Dissipation .... 1.5 W

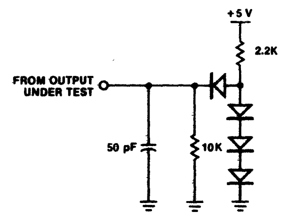
Stresses greater than those listed under Absolute Maximum Ratings may cause permanent damage to the device. This is a stress rating only; operation of the device at any condition above those indicated in the operational sections of these specifications is not implied. Exposure to absolute maximum rating conditions for extended periods may affect device reliability.

Standard Test Conditions

The characteristics below apply for the following standard test conditions, unless otherwise noted. All voltages are referenced to GND (0 V). Positive current flows into the referenced pin. Available operating temperature ranges are:

- 0°C to +70°C, $+4.75\ V \leq V_{CC} \leq +5.25\ V$
- -40°C to +85°C, $+4.75\ V \leq V_{CC} \leq +5.25\ V$
- -55°C to +125°C, $+4.5\ V \leq V_{CC} \leq +5.5\ V$

All ac parameters assume a load capacitance of 50 pF. Add 10 ns delay for each 50 pF increase in load up to a maximum of 200 pF for the data bus and 100 pF for address and control lines.



DC Characteristics

Symbol	Parameter	Min	Max	Unit	Test Condition
V_{ILC}	Clock Input Low Voltage	-0.3	0.45	V	
V_{IHC}	Clock Input High Voltage	V_{CC}-.6	V_{CC}+.3	V	
V_{IL}	Input Low Voltage	-0.3	0.8	V	
V_{IH}	Input High Voltage	2.0	V_{CC}	V	
V_{OL}	Output Low Voltage		0.4	V	I_{OL} = 1.8 mA
V_{OH}	Output High Voltage	2.4		V	I_{OH} = -250 μA
I_{CC}	Power Supply Current				
	Z80		150[1]	mA	
	Z80A		200[2]	mA	
	Z80B		200	mA	
I_{LI}	Input Leakage Current		10	μA	V_{IN} = 0 to V_{CC}
I_{LEAK}	3-State Output Leakage Current in Float	-10	10[3]	μA	V_{OUT} = 0.4 to V_{CC}

1. For military grade parts, I_{CC} is 200 mA.
2. Typical rate for Z80A is 90 mA.
3. A_{15}-A_0, D_7-D_0, $\overline{MREQ}$, $\overline{IORQ}$, $\overline{RD}$, and $\overline{WR}$.

Capacitance

Symbol	Parameter	Min	Max	Unit	Note
C_{CLOCK}	Clock Capacitance		35	pF	Unmeasured pins returned to ground
C_{IN}	Input Capacitance		5	pF	
C_{OUT}	Output Capacitance		10	pF	

T_A = 25°C, f = 1 MHz.

Ordering Information

Product Number	Package/ Temp	Speed	Description	Product Number	Package/ Temp	Speed	Description
Z8400	CE	2.5 MHz	Z80 CPU (40-pin)	Z8400A	DE	4.0 MHz	Z80A CPU (40-pin)
Z8400	CM	2.5 MHz	Same as above	Z8400A	DS	4.0 MHz	Same as above
Z8400	CMB	2.5 MHz	Same as above	Z8400A	PE	4.0 MHz	Same as above
Z8400	CS	2.5 MHz	Same as above	Z8400A	PS	4.0 MHz	Same as above
Z8400	DE	2.5 MHz	Same as above	Z8400B	CE	6.0 MHz	Z80B CPU (40-pin)
Z8400	DS	2.5 MHz	Same as above	Z8400B	CM	6.0 MHz	Same as above
Z8400	PE	2.5 MHz	Same as above	Z8400B	CMB	6.0 MHz	Same as above
Z8400	PS	2.5 MHz	Same as above	Z8400B	CS	6.0 MHz	Same as above
Z8400A	CE	4.0 MHz	Z80A CPU (40-pin)	Z8400B	DE	6.0 MHz	Same as above
Z8400A	CM	4.0 MHz	Same as above	Z8400B	DS	6.0 MHz	Same as above
Z8400A	CMB	4.0 MHz	Same as above	Z8400B	PE	6.0 MHz	Same as above
Z8400A	CS	4.0 MHz	Same as above	Z8400B	PS	6.0 MHz	Same as above

NOTES: C = Ceramic, D = Cerdip, P = Plastic; E = -40°C to +85°C, M = -55°C to +125°C, MB = -55°C to +125°C with MIL-STD-883 Class B processing, S = 0°C to +70°C.

8251/8251A USART
DATA SHEET

NEC Microcomputers, Inc.

PROGRAMMABLE COMMUNICATION INTERFACES

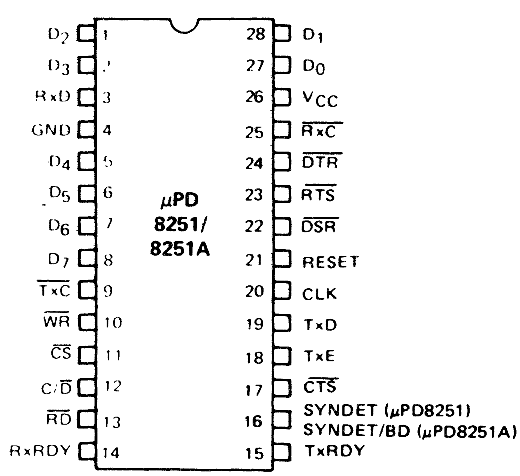
DESCRIPTION

The μPD8251 and μPD8251A Universal Synchronous/Asynchronous Receiver/Transmitters (USARTs) are designed for microcomputer systems data communications. The USART is used as a peripheral and is programmed by the 8080A or other processor to communicate in commonly used serial data transmission techniques including IBM Bi-Sync. The USART receives serial data streams and converts them into parallel data characters for the processor. While receiving serial data, the USART will also accept data characters from the processor in parallel format, convert them to serial format and transmit. The USART will signal the processor when it has completely received or transmitted a character and requires service. Complete USART status including data format errors and control signals such as TxE and SYNDET, is available to the processor at any time.

FEATURES

- Asynchronous or Synchronous Operation
  - Asynchronous:
    - Five 8-Bit Characters
    - Clock Rate – 1, 16 or 64 x Baud Rate
    - Break Character Generation
    - Select 1, 1-1/2, or 2 Stop Bits
    - False Start Bit Detector
    - Automatic Break Detect and Handling (μPD8251A)
  - Synchronous:
    - Five 8-Bit Characters
    - Internal or External Character Synchronization
    - Automatic Sync Insertion
    - Single or Double Sync Characters
- Baud Rate (1X Mode) – DC to 56K Baud (μPD8251)
  – DC to 64K Baud (μPD8251A)
- Full Duplex, Double Buffered Transmitter and Receiver
- Parity, Overrun and Framing Flags
- Fully Compatible with 8080A/8085/μPD780 (Z80™)
- All Inputs and Outputs are TTL Compatible
- Single +5 Volt Supply, ±10%
- Separate Device Receive and Transmit TTL Clocks
- 28 Pin Plastic DIP Package
- N-Channel MOS Technology

PIN CONFIGURATION



PIN NAMES

D_7-D_0	Data Bus (8 bits)
C/$\overline{D}$	Control or Data is to be Written or Read
$\overline{RD}$	Read Data Command
$\overline{WR}$	Write Data or Control Command
$\overline{CS}$	Chip Enable
CLK	Clock Pulse (TTL)
RESET	Reset
$\overline{TxC}$	Transmitter Clock (TTL)
TxD	Transmitter Data
$\overline{RxC}$	Receiver Clock (TTL)
RxD	Receiver Data
RxRDY	Receiver Ready (has character for 8080)
TxRDY	Transmitter Ready (ready for char. from 8080)
$\overline{DSR}$	Data Set Ready
$\overline{DTR}$	Data Terminal Ready
SYNDET	Sync Detect
SYNDET/BD	Sync Detect/Break Detect
$\overline{RTS}$	Request to Send Data
$\overline{CTS}$	Clear to Send Data
TxE	Transmitter Empty
V_{CC}	+5 Volt Supply
GND	Ground

TM: Z80 is a registered trademark of Zilog.

Rev/4

μPD8251/8251A

FUNCTIONAL DESCRIPTION

The μPD8251 and μPD8251A Universal Synchronous/Asynchronous Receiver/Transmitters are designed specifically for 8080 microcomputer systems but work with most 8-bit processors. Operation of the μPD8251 and μPD8251A, like other I/O devices in the 8080 family, are programmed by system software for maximum flexibility.

In the receive mode, the μPD8251 or μPD8251A converts incoming serial format data into parallel data and makes certain format checks. In the transmit mode, it formats parallel data into serial form. The device also supplies or removes characters or bits that are unique to the communication format in use. By performing conversion and formatting services automatically, the USART appears to the processor as a simple or "transparent" input or output of byte-oriented parallel data.

μPD8251A FEATURES AND ENHANCEMENTS

The μPD8251A is an advanced design of the industry standard 8251 USART. It operates with a wide range of microprocessors, including the 8080, 8085, and μPD780 (Z80™). The additional features and enhancements of the μPD8251A over the μPD8251 are listed below.

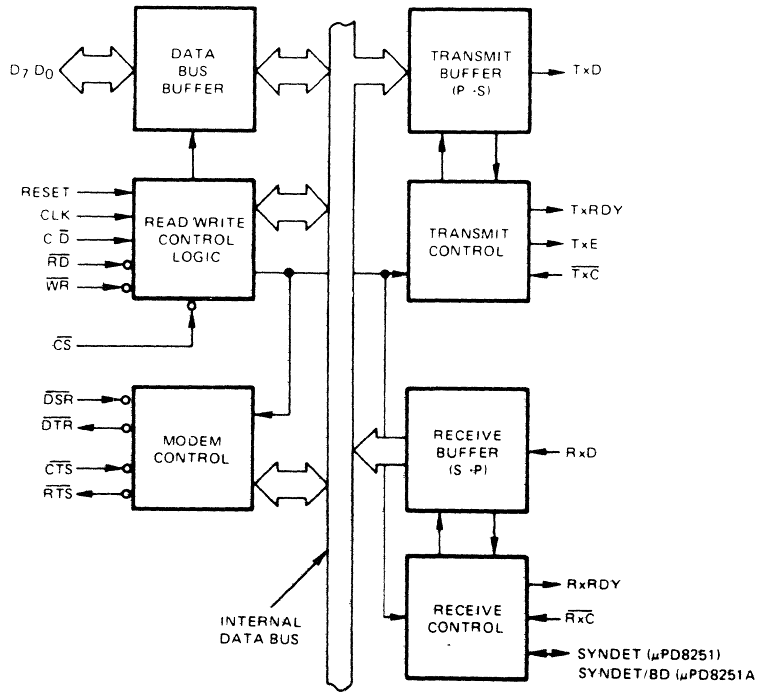
1. The data paths are double-buffered with separate I/O registers for control, status, Data In and Data Out. This feature simplifies control programming and minimizes processor overhead.
2. The Receiver detects and handles "break" automatically in asynchronous operations, which relieves the processor of this task.
3. The Receiver is prevented from starting when in "break" state by a refined Rx initialization. This also prevents a disconnected USART from causing unwanted interrupts.
4. When a transmission is concluded the TxD line will always return to the marking state unless SBRK is programmed.
5. The Tx Disable command is prevented from halting transmission by the Tx Enable Logic enhancement, until all data previously written has been transmitted. The same logic also prevents the transmitter from turning off in the middle of a word.
6. Internal Sync Detect is disabled when External Sync Detect is programmed. An External Sync Detect Status is provided through a flip-flop which clears itself upon a status read.
7. The possibility of a false sync detect is minimized by:
   - ensuring that if a double sync character is programmed, the characters be contiguously detected.
   - clearing the Rx register to all Logic 1s (V_{OH}) whenever the Enter Hunt command is issued in Sync mode.
8. The $\overline{RD}$ and $\overline{WR}$ do not affect the internal operation of the device as long as the μPD8251A is not selected.
9. The μPD8251A Status can be read at any time, however, the status update will be inhibited during status read.
10. The μPD8251A has enhanced AC and DC characteristics and is free from extraneous glitches, providing higher speed and improved operating margins.
11. Baud rate from DC to 64K.

BASIC OPERATION

$C/\overline{D}$	$\overline{RD}$	$\overline{WR}$	$\overline{CS}$	
0	0	1	0	μPD8251/μPD8251A → Data Bus
0	1	0	0	Data Bus → μPD8251/μPD8251A
1	0	1	0	Status → Data Bus
1	1	0	0	Data Bus → Control
X	X	X	1	Data Bus → 3-State
X	1	1	0	

TM:Z80 is a registered trademark of Zilog.

BLOCK DIAGRAM



ABSOLUTE MAXIMUM RATINGS*

Operating Temperature . . . . −0°C to +70°C
Storage Temperature . . . . −65°C to +125°C
All Output Voltages . . . . −0.5 to +7 Volts
All Input Voltages . . . . −0.5 to +7 Volts
Supply Voltages . . . . −0.5 to +7 Volts

COMMENT: Stress above those listed under "Absolute Maximum Ratings" may cause permanent damage to the device. This is a stress rating only and functional operation of the device at these or any other conditions above those indicated in the operational sections of this specification is not implied. Exposure to absolute maximum rating conditions for extended periods may affect device reliability.

*T_a = 25°C

DC CHARACTERISTICS

T_a = 0°C to 70°C; V_{CC} = 5.0V ± 10%; GND = 0V.

PARAMETER	SYMBOL	LIMITS µPD8251 MIN	TYP	MAX	µPD8251A MIN	MAX	UNIT	TEST CONDITIONS
Input Low Voltage	V_{IL}	−0.5		0.8	0.5	0.8	V	
Input High Voltage	V_{IH}	2.0		V_{CC}	2.0	V_{CC}	V	
Output Low Voltage	V_{OL}			0.45		0.45	V	µPD8251: I_{OL} = 1.7 mA µPD8251A: I_{OL} = 2.2 mA
Output High Voltage	V_{OH}	2.4			2.4		V	µPD8251: I_{OH} = −100 µA µPD8251A: I_{OH} = −400 µA
Data Bus Leakage	I_{DL}			−50		−10	µA	V_{OUT} = 0.45V
				10		10		V_{OUT} = V_{CC}
Input Load Current	I_{IL}			10		10	µA	At 5.5V
Power Supply Current	I_{CC}		45	80		100	mA	µPD8251A: All Outputs = Logic 1

CAPACITANCE

T_a = 25°C; V_{CC} = GND = 0V

PARAMETER	SYMBOL	LIMITS MIN	TYP	MAX	UNIT	TEST CONDITIONS
Input Capacitance	C_{IN}			10	pF	fc = 1 MHz
I/O Capacitance	$C_{I/O}$			20	pF	Unmeasured pins returned to GND

AC CHARACTERISTICS

T_a = 0°C to 70°C; V_{CC} = 5.0V ± 10%; GND = 0V

PARAMETER	SYMBOL	LIMITS µPD8251 MIN	LIMITS µPD8251 MAX	LIMITS µPD8215A MIN	LIMITS µPD8215A MAX	UNIT	TEST CONDITIONS
READ							
Address Stable before $\overline{\text{READ}}$, ($\overline{\text{CS}}$, C/$\overline{\text{D}}$)	t_{AR}	50		0		ns	
Address Hold Time for $\overline{\text{READ}}$, ($\overline{\text{CS}}$, C/$\overline{\text{D}}$)	t_{RA}	5		0		ns	
$\overline{\text{READ}}$ Pulse Width	t_{RR}	430		250		ns	
Data Delay from $\overline{\text{READ}}$	t_{RD}		350		200	ns	µPD8251 C_L 100 pF µPD8251A C_L 150 pF
$\overline{\text{READ}}$ to Data Floating	t_{DF}	25	200	10	100	ns	µPD8251 C_L - 100 pF C_L - 15 pF
WRITE							
Address Stable before $\overline{\text{WRITE}}$	t_{AW}	20		0		ns	
Address Hold Time for $\overline{\text{WRITE}}$	t_{WA}	20		0		ns	
$\overline{\text{WRITE}}$ Pulse Width	t_{WW}	400		250		ns	
Data Set Up Time for $\overline{\text{WRITE}}$	t_{DW}	200		150		ns	
Data Hold Time for $\overline{\text{WRITE}}$	t_{WD}	40		0		ns	
Recovery Time Between WRITES ②	t_{RV}	6		6		t_{CY}	
OTHER TIMING							
Clock Period ③	t_{CY}	0 420	1 35	0 32	1 35	µs	
Clock Pulse Width High	$t_{\phi W}$	220	0 7t_{CY}	120	t_{CY} 90	ns	
Clock Pulse Width Low	$t_{\phi W}$			90		ns	
Clock Rise and Fall Time	t_R, t_F	0	50	5	20	ns	
TxD Delay from Falling Edge of TxC	t_{DTx}		1		1	µs	µPD8251 C_L = 100 pF
Rx Data Set Up Time to Sampling Pulse	t_{SRx}	2		2		µs	
Rx Data Hold Time to Sampling Pulse	t_{HRx}	2		2		µs	
Transmitter Input Clock Frequency	f_{Tx}						
1X Baud Rate		DC	56		64	kHz	
16X Baud Rate		DC	520		310	kHz	
64X Baud Rate		DC	520		615	kHz	
Transmitter Input Clock Pulse Width	t_{TPW}						
1X Baud Rate		12		12		t_{CY}	
16X and 64X Baud Rate		1		1		t_{CY}	
Transmitter Input Clock Pulse Delay	t_{TPD}						
1X Baud Rate		15		15		t_{CY}	
16X and 64X Baud Rate		3		3		t_{CY}	
Receiver Input Clock Frequency	f_{Rx}						
1X Baud Rate		DC	56		64	kHz	
16X Baud Rate		DC	520		310	kHz	
64X Baud Rate		DC	520		615	kHz	
Receiver Input Clock Pulse Width	t_{RPW}						
1X Baud Rate		12		12		t_{CY}	
16X and 64X Baud Rate		1		1		t_{CY}	
Receiver Input Clock Pulse Delay	t_{RPD}						
1X Baud Rate		15		15		t_{CY}	
16X and 64X Baud Rate		3		3		t_{CY}	
TxRDY Delay from Center of Data Bit	t_{Tx}		16		8	t_{CY}	µPD8251 C_L 50 pF
RxRDY Delay from Center of Data Bit	t_{RX}		20		24	t_{CY}	
Internal SYNDET Delay from Center of Data Bit	t_{IS}		25		24	t_{CY}	
External SYNDET Set-Up Time before Falling Edge of $\overline{\text{RxC}}$	t_{ES}		16		16	t_{CY}	
TxEMPTY Delay from Center of Data Bit	t_{TxE}		16		20	t_{CY}	µPD8251 C_L 50 pF
Control Delay from Rising Edge of WRITE (TxE, $\overline{\text{DTR}}$, $\overline{\text{RTS}}$)	t_{WC}		16		8	t_{CY}	
Control to READ Set Up Time ($\overline{\text{DSR}}$, $\overline{\text{CTS}}$)	t_{CR}		16		20	t_{CY}	

Notes ① AC timings measured at V_{OH} = 2 0, V_{OL} = 0 8, and with load circuit of Figure 1

② This recovery time is for initialization only, when MODE, SYNC1, SYNC2, COMMAND and first DATA BYTES are written into the USART. Subsequent writing of both COMMAND and DATA are only allowed when TxRDY = 1

③ The TxC and RxC frequencies have the following limitations with respect to CLK
For 1X Baud Rate, f_{Tx} or $f_{Rx} \leq 1/(30\ t_{CY})$
For 16X and 64X Baud Rate, f_{TX} or $f_{Rx} \leq 1/(4.5\ t_{CY})$

④ Reset Pulse Width = 6 t_{CY} minimum

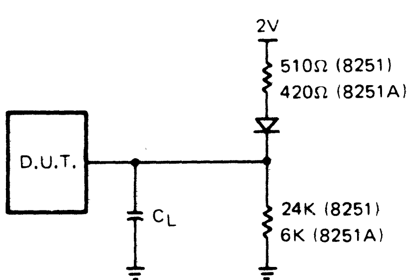
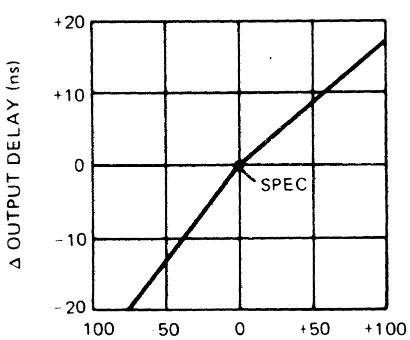


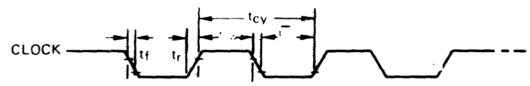
Figure 1.



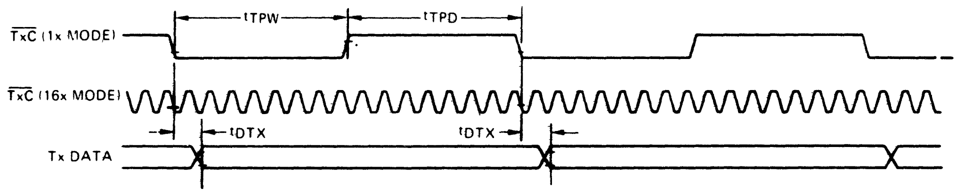
Typical Δ Output Delay Versus Δ Capacitance (pF)

TEST LOAD CIRCUIT

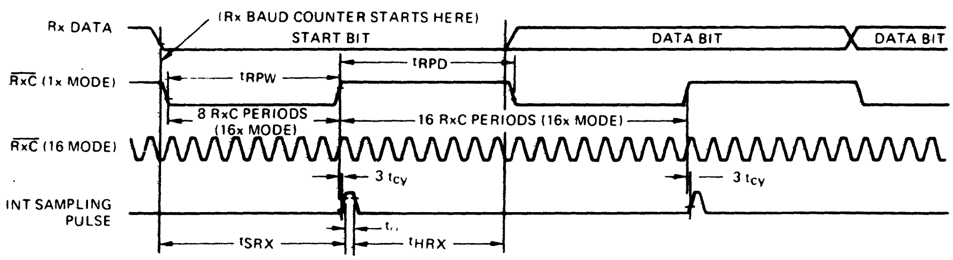
TIMING WAVEFORM



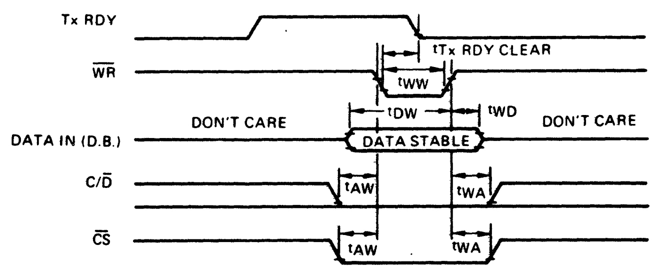
SYSTEM CLOCK INPUT



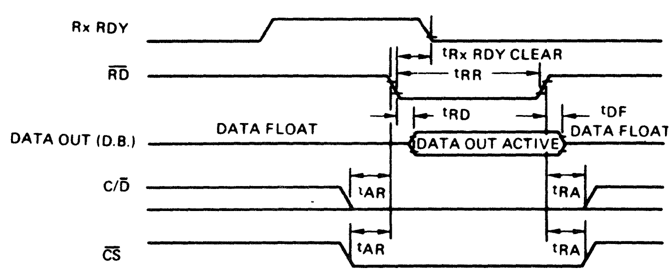
TRANSMITTER CLOCK AND DATA



RECEIVER CLOCK AND DATA

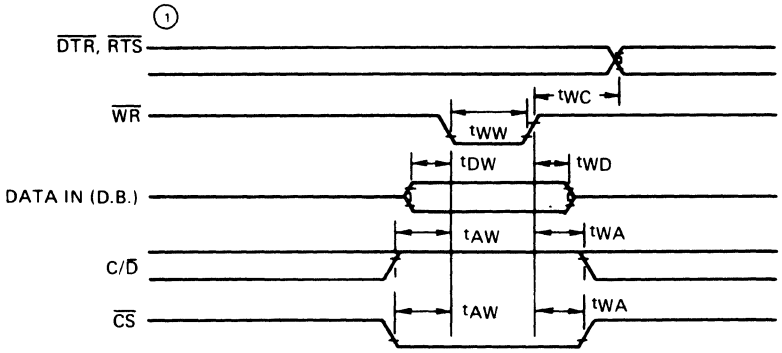


WRITE DATA CYCLE (PROCESSOR → USART)

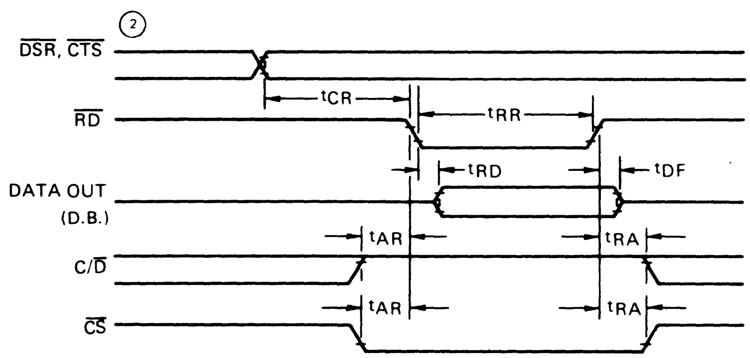


READ DATA CYCLE (PROCESSOR ← USART)

TIMING WAVEFORM (CONT.)



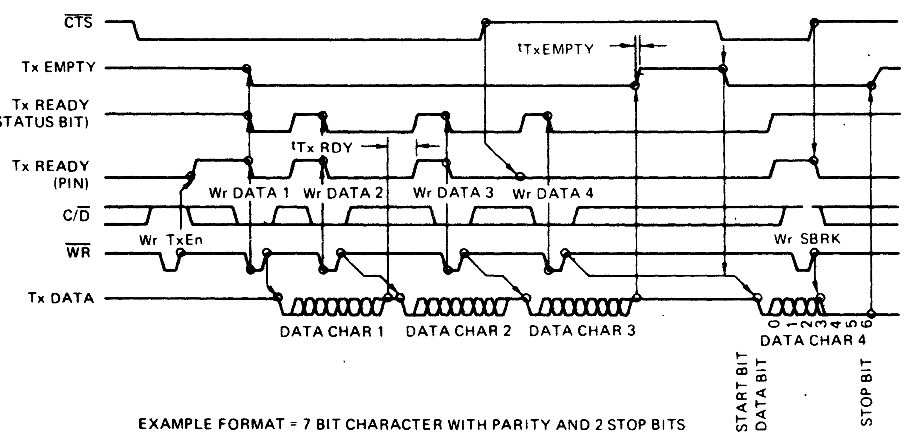
WRITE CONTROL OR OUTPUT PORT CYCLE (PROCESSOR → USART)



READ CONTROL OR INPUT PORT CYCLE (PROCESSOR ← USART)

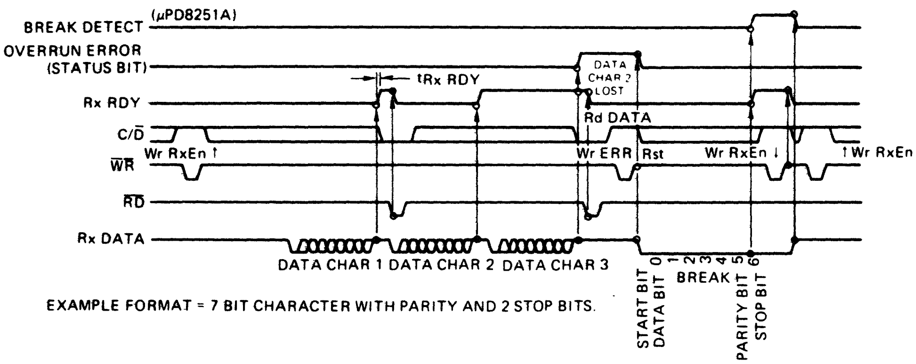
NOTES: (1) T_{WC} Includes the response timing of a control byte

(2) T_{CR} Includes the effect of CTS on the TxENBL circuitry

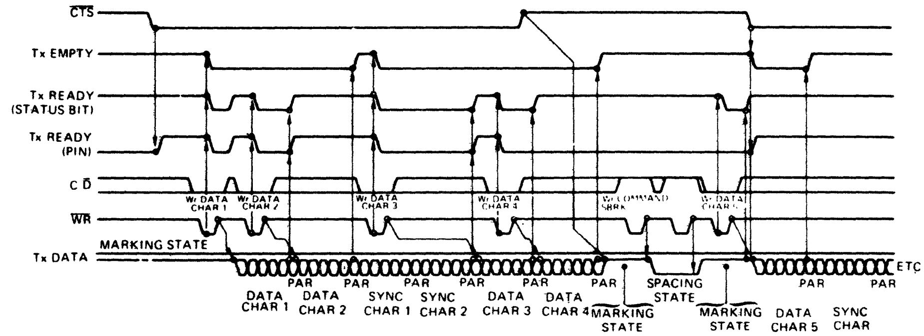


TRANSMITTER CONTROL AND FLAG TIMING (ASYNC MODE)

TIMING WAVEFORM (CONT.)

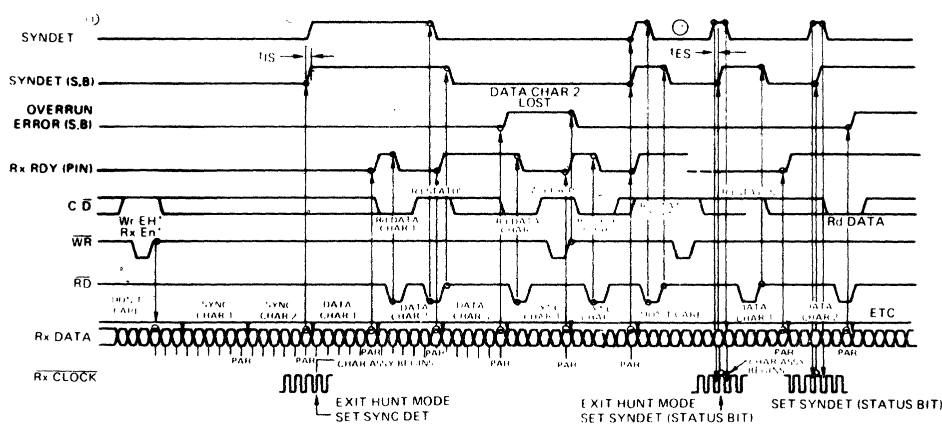


RECEIVER CONTROL AND FLAG TIMING (ASYNC MODE)



EXAMPLE FORMAT = 5 BIT CHARACTER WITH PARITY AND 2 SYNC CHARACTERS.

TRANSMITTER CONTROL AND FLAG TIMING (SYNC MODE)



RECEIVER CONTROL AND FLAG TIMING (SYNC MODE)

Notes: ① Internal sync, 2 sync characters, 5 bits, with parity.
② External sync, 5 bits, with parity.

PIN IDENTIFICATION

PIN NO.	PIN SYMBOL	PIN NAME	FUNCTION
1, 2, 27, 28 5 – 8	$D_7 - D_0$	Data Bus Buffer	An 8-bit, 3-state bi-directional buffer used to interface the USART to the processor data bus. Data is transmitted or received by the buffer in response to input/output or Read/Write instructions from the processor. The Data Bus Buffer also transfers Control words, Command words, and Status.
26	V_{CC}	V_{CC} Supply Voltage	+5 volt supply
4	GND	Ground	Ground
Read/Write Control Logic			This logic block accepts inputs from the processor Control Bus and generates control signals for overall USART operation. The Mode Instruction and Command Instruction registers that store the control formats for device functional definition are located in the Read/Write Control Logic.
21	RESET	Reset	A "one" on this input forces the USART into the "Idle" mode where it will remain until reinitialized with a new set of control words. Minimum RESET pulse width is 6 t_{CY}.
20	CLK	Clock Pulse	The CLK input provides for internal device timing and is usually connected to the Phase 2 (TTL) output of the μPB8224 Clock Generator. External inputs and outputs are not referenced to CLK, but the CLK frequency must be at least 30 times the Receiver or Transmitter clocks in the synchronous mode and 4.5 times for the asynchronous mode.
10	$\overline{WR}$	Write Data	A "zero" on this input instructs the USART to accept the data or control word which the processor is writing out on the data bus.
13	$\overline{RD}$	Read Data	A "zero" on this input instructs the USART to place the data or status information onto the Data Bus for the processor to read.
12	$C/\overline{D}$	Control/Data	The Control/Data input, in conjunction with the $\overline{WR}$ and $\overline{RD}$ inputs, informs the USART to accept or provide either a data character, control word or status information via the Data Bus. 0 = Data; 1 = Control.
11	$\overline{CS}$	Chip Select	A "zero" on this input enables the USART to read from or write to the processor.
Modem Control			The μPD8251 and μPD8251A have a set of control inputs and outputs which may be used to simplify the interface to a Modem.
22	$\overline{DSR}$	Data Set Ready	The Data Set Ready input can be tested by the processor via Status information. The $\overline{DSR}$ input is normally used to test Modem Data Set Ready condition.
24	$\overline{DTR}$	Data Terminal Ready	The Data Terminal Ready output can be controlled via the Command word. The $\overline{DTR}$ output is normally used to drive Modem Data Terminal Ready or Rate Select lines.
23	$\overline{RTS}$	Request to Send	The Request to Send output can be controlled via the Command word. The $\overline{RTS}$ output is normally used to drive the Modem Request to Send line.
17	$\overline{CTS}$	Clear to Send	A "zero" on the Clear to Send input enables the USART to transmit serial data if the TxEN bit in the Command Instruction register is enabled (one).

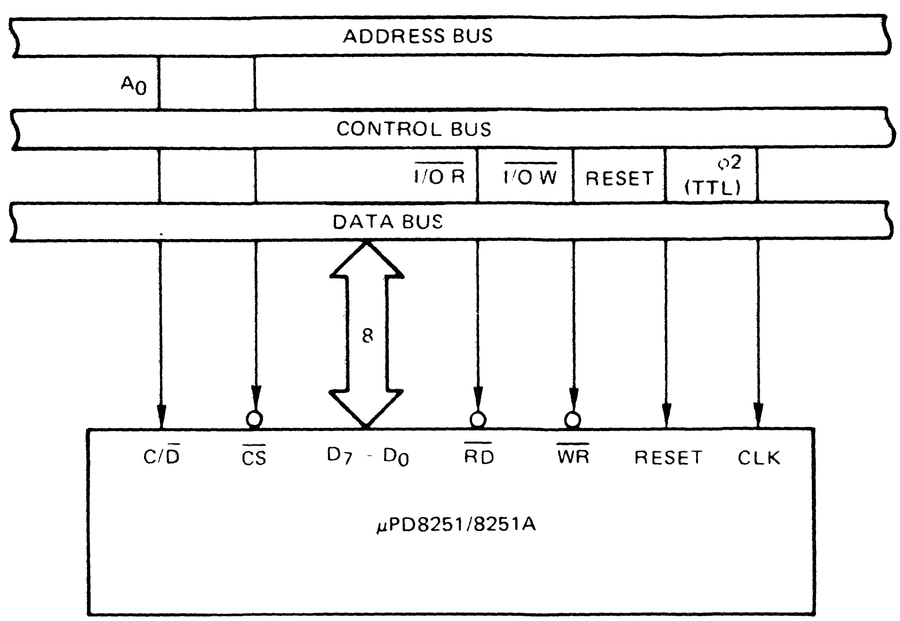
TRANSMIT BUFFER

The Transmit Buffer receives parallel data from the Data Bus Buffer via the internal data bus, converts parallel to serial data, inserts the necessary characters or bits needed for the programmed communication format and outputs composite serial data on the TxD pin.

PIN IDENTIFICATION (CONT.)

PIN NO.	PIN SYMBOL	PIN NAME	FUNCTION
Transmit Control Logic			The Transmit Control Logic accepts and outputs all external and internal signals necessary for serial data transmission.
15	TxRDY	Transmitter Ready	Transmitter Ready signals the processor that the transmitter is ready to accept a data character. TxRDY can be used as an interrupt or may be tested through the Status information for polled operation. Loading a character from the processor automatically resets TxRDY, on the leading edge.
18	TxE	Transmitter Empty	The Transmitter Empty output signals the processor that the USART has no further characters to transmit. TxE is automatically reset upon receiving a data character from the processor. In half-duplex, TxE can be used to signal end of a transmission and request the processor to "turn the line around." The TxEn bit in the command instruction does not effect TxE. In the Synchronous mode, a "one" on this output indicates that a Sync character or characters are about to be automatically transmitted as "fillers" because the next data character has not been loaded.
9	$\overline{TxC}$	Transmitter Clock	The Transmitter Clock controls the serial character transmission rate. In the Asynchronous mode, the $\overline{TxC}$ frequency is a multiple of the actual Baud Rate. Two bits of the Mode Instruction select the multiple to be 1x, 16x, or 64x the Baud Rate. In the Synchronous mode, the $\overline{TxC}$ frequency is automatically selected to equal the actual Baud Rate. Note that for both Synchronous and Asynchronous modes, serial data is shifted out of the USART by the falling edge of $\overline{TxC}$.
19	TxD	Transmitter Data	The Transmit Control Logic outputs the composite serial data stream on this pin.

μPD8251 AND μPD8251A INTERFACE TO 8080 STANDARD SYSTEM BUS



RECEIVE BUFFER

The Receive Buffer accepts serial data input at the $\overline{\text{RxD}}$ pin and converts the data from serial to parallel format. Bits or characters required for the specific communication technique in use are checked and then an eight-bit "assembled" character is readied for the processor. For communication techniques which require less than eight bits, the μPD8251 and μPD8251A set the extra bits to "zero."

PIN IDENTIFICATION (CONT.)

PIN			FUNCTION
NO.	SYMBOL	NAME	
Receiver Control Logic			This block manages all activities related to incoming data.
14	RxRDY	Receiver Ready	The Receiver Ready output indicates that the Receiver Buffer is ready with an "assembled" character for input to the processor. For Polled operation, the processor can check RxRDY using a Status Read or RxRDY can be connected to the processor interrupt structure. Note that reading the character to the processor automatically resets RxRDY.
25	$\overline{\text{RxC}}$	Receiver Clock	The Receiver Clock determines the rate at which the incoming character is received. In the Asynchronous mode, the $\overline{\text{RxC}}$ frequency may be 1.16 or 64 times the actual Baud Rate but in the Synchronous mode the $\overline{\text{RxC}}$ frequency must equal the Baud Rate. Two bits in the mode instruction select Asynchronous at 1x, 16x or 64x or Synchronous operation at 1x the Baud Rate. Unlike $\overline{\text{TxC}}$, data is sampled by the μPD8251 and μPD8251A on the rising edge of $\overline{\text{RxC}}$. ①
3	RxD	Receiver Data	A composite serial data stream is received by the Receiver Control Logic on this pin.
16	SYNDET (μPD8251)	Sync Detect	The SYNC Detect pin is only used in the Synchronous mode. The μPD8251 may be programmed through the Mode Instruction to operate in either the internal or external Sync mode and SYNDET then functions as an output or input respectively. In the internal Sync mode, the SYNDET output will go to a "one" when the μPD8251 has located the SYNC character in the Receive mode. If double SYNC character (bi-sync) operation has been programmed, SYNDET will go to "one" in the middle of the last bit of the second SYNC character. SYNDET is automatically reset to "zero" upon a Status Read or RESET. In the external SYNC mode, a "zero" to "one" transition on the SYNDET input will cause the μPD8251 to start assembling data character on the next falling edge of $\overline{\text{RxC}}$. The length of the SYNDET input should be at least one $\overline{\text{RxC}}$ period, but may be removed once the μPD8251 is in SYNC.
16	SYNDET/BD (μPD8251A)	Sync Detect/ Break Detect	The SYNDET/BD pin is used in both Synchronous and Asynchronous modes. When in SYNC mode the features for the SYNDET pin described above apply. When in Asynchronous mode, the Break Detect output will go high when an all zero word of the programmed length is received. This word consists of: start bit, data bit, parity bit and one stop bit. Reset only occurs when Rx data returns to a logic one state or upon chip reset. The state of Break Detect can be read as a status bit.

Note: ① Since the μPD8251 and μPD8251A will frequently be handling both the reception and transmission for a given link, the Receive and Transmit Baud Rates will be same. $\overline{\text{RxC}}$ and $\overline{\text{TxC}}$ then require the same frequency and may be tied together and connected to a single clock source or Baud Rate Generator.

Examples:

If the Baud Rate equals 110 (Async):	If the Baud Rate equals 300:
$\overline{\text{RxC}}$ or $\overline{\text{TxC}}$ equals 110 Hz (1x)	$\overline{\text{RxC}}$ or $\overline{\text{TxC}}$ equals 300 Hz (1x) A or S
$\overline{\text{RxC}}$ or $\overline{\text{TxC}}$ equals 1.76 KHz (16x)	$\overline{\text{RxC}}$ or $\overline{\text{TxC}}$ equals 4800 Hz (16x) A only
$\overline{\text{RxC}}$ or $\overline{\text{TxC}}$ equals 7.04 KHz (64x)	$\overline{\text{RxC}}$ or $\overline{\text{TxC}}$ equals 19.2 KHz (64x) A only

OPERATIONAL DESCRIPTION

A set of control words must be sent to the μPD8251 and μPD8251A to define the desired mode and communications format. The control words will specify the BAUD rate factor (1x, 16x, 64x), character length (5 to 8), number of STOP bits (1, 1-1/2, 2) Asynchronous or Synchronous mode, SYNDET (IN or OUT), parity, etc.

After receiving the control words, the μPD8251 and μPD8251A are ready to communicate. TxRDY is raised to signal the processor that the USART is ready to receive a character for transmission. When the processor writes a character to the USART, TxRDY is automatically reset.

Concurrently, the μPD8251 and μPD8251A may receive serial data; and after receiving an entire character, the RxRDY output is raised to indicate a completed character is ready for the processor. The processor fetch will automatically reset RxRDY.

Note: The μPD8251 and μPD8251A may provide faulty RxRDY for the first read after power-on or for the first read after receive is re-enabled by a command instruction (RxE). A dummy read is recommended to clear faulty RxRDY. But this is not the case for the first read after hardware or software reset after the device operation has once been established.

The μPD8251 and μPD8251A cannot transmit until the TxEN (Transmitter Enable) bit has been set by a Command Instruction and until the $\overline{CTS}$ (Clear to Send) input is a "zero". TxD is held in the "marking" state after Reset awaiting new control words.

USART PROGRAMMING

The USART must be loaded with a group of two to four control words provided by the processor before data reception and transmission can begin. A RESET (internal or external) must immediately proceed the control words which are used to program the complete operational description of the communications interface. If an external RESET is not available, three successive 00 Hex or two successive 80 Hex command instructions ($C/\overline{D}$ = 1) followed by a software reset command instruction (40 Hex) can be used to initialize the μPD8251 and μPD8251A.

There are two control word formats:

1. Mode Instruction
2. Command Instruction

MODE INSTRUCTION

This control word specifies the general characteristics of the interface regarding the Synchronous or Asynchronous mode, BAUD rate factor, character length, parity, and number of stop bits. Once the Mode Instruction has been received, SYNC characters or Command Instructions may be inserted depending on the Mode Instruction content.

COMMAND INSTRUCTION

This control word will be interpreted as a SYNC character definition if immediately preceded by a Mode Instruction which specified a Synchronous format. After the SYNC character(s) are specified or after an Asynchronous Mode Instruction, all subsequent control words will be interpreted as an update to the Command Instruction. Command Instruction updates may occur at any time during the data block. To modify the Mode Instruction, a bit may be set in the Command Instruction which causes an internal Reset which allows a new Mode Instruction to be accepted.

TYPICAL DATA BLOCK

$C/\overline{D} = 1$	MODE INSTRUCTION	
$C/\overline{D} = 1$	SYNC CHARACTER 1	SYNC MODE ONLY (1)
$C/\overline{D} = 1$	SYNC CHARACTER 2	
$C/\overline{D} = 1$	COMMAND INSTRUCTION	
$C/\overline{D} = 0$	DATA	
$C/\overline{D} = 1$	COMMAND INSTRUCTION	
$C/\overline{D} = 0$	DATA	
$C/\overline{D} = 1$	COMMAND INSTRUCTION	

NOTE (1) The second SYNC character is skipped if MODE instruction has programmed the μPD8251 and μPD8251A to single character Internal SYNC Mode. Both SYNC characters are skipped if MODE instruction has programmed the μPD8251 and μPD8251A to ASYNC mode.

MODE INSTRUCTION DEFINITION

The μPD8251 and μPD8251A can operate in either Asynchronous or Synchronous communication modes. Understanding how the Mode Instruction controls the functional operation of the USART is easiest when the device is considered to be two separate components (one asynchronous and the other synchronous) which share the same support circuits and package. Although the format definition can be changed at will or "on the fly", the two modes will be explained separately for clarity.

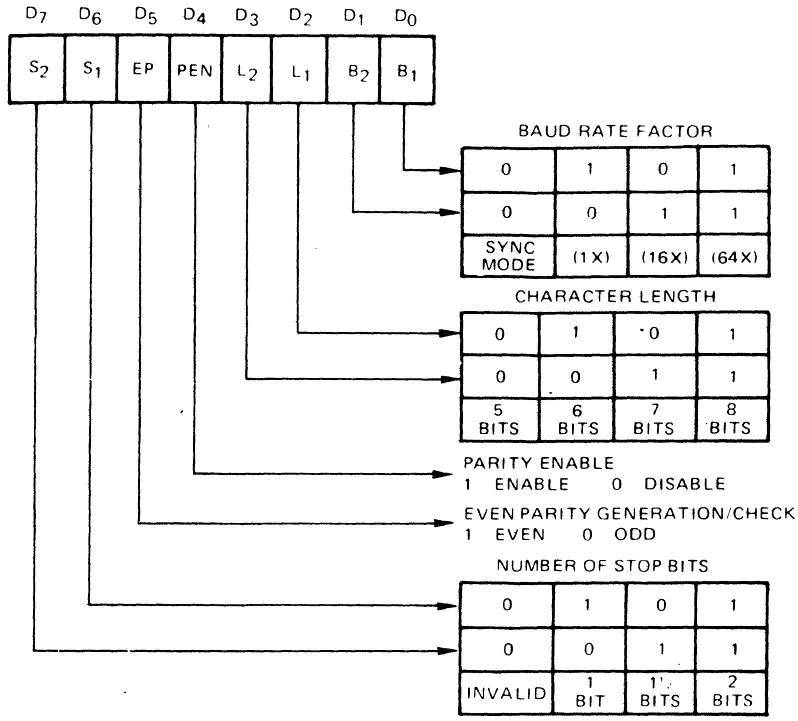
ASYNCHRONOUS TRANSMISSION

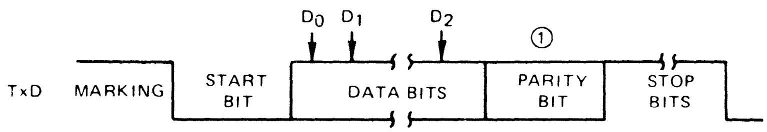
When a data character is written into the μPD8251 and μPD8251A, the USART automatically adds a START bit (low level or "space") and the number of STOP bits (high level or "mark") specified by the Mode Instruction. If Parity has been enabled, an odd or even Parity bit is inserted just before the STOP bit(s), as specified by the Mode Instruction. Then, depending on $\overline{CTS}$ and TxEN, the character may be transmitted as a serial data stream at the TxD output. Data is shifted out by the falling edge of $\overline{TxC}$ at $\overline{TxC}$, $\overline{TxC}/16$ or $\overline{TxC}/64$, as defined by the Mode Instruction.

If no data characters have been loaded into the μPD8251 and μPD8251A, or if all available characters have been transmitted, the TxD output remains "high" (marking) in preparation for sending the START bit of the next character provided by the processor. TxD may be forced to send a BREAK (continuously low) by setting the correct bit in the Command Instruction.

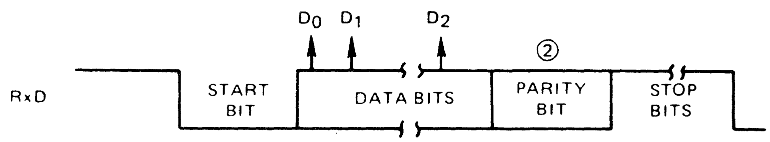
ASYNCHRONOUS RECEIVE

The RxD input line is normally held "high" (marking) by the transmitting device. A falling edge at RxD signals the possible beginning of a START bit and a new character. The START bit is checked by testing for a "low" at its nominal center as specified by the BAUD RATE. If a "low" is detected again, it is considered valid, and the bit assembling counter starts counting. The bit counter locates the approximate center of the data, parity (if specified), and STOP bits. The parity error flag (PE) is set, if a parity error occurs. Input bits are sampled at the RxD pin with the rising edge of $\overline{RxC}$. If a high is not detected for the STOP bit, which normally signals the end of an input character, a framing error (FE) will be set. After a valid STOP bit, the input character is loaded into the parallel Data Bus Buffer of the μPD8251 and μPD8251A and the RxRDY signal is raised to indicate to the processor that a character is ready to be fetched. If the processor has failed to fetch the previous character, the new character replaces the old and the overrun flag (OE) is set. All the error flags can be reset by setting a bit in the Command Instruction. Error flag conditions will not stop subsequent USART operation.

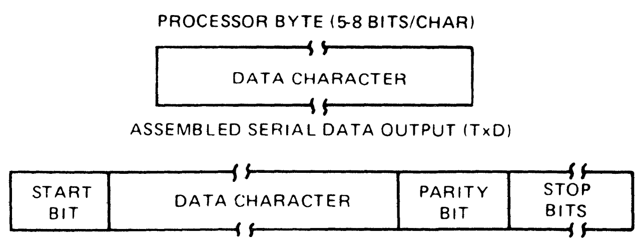




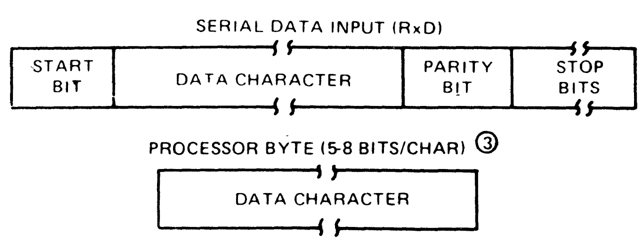
TRANSMITTER OUTPUT



RECEIVER INPUT



TRANSMISSION FORMAT



RECEIVE FORMAT

Notes ① Generated by μPD8251/8251A
② Does not appear on the Data Bus.
③ If character length is defined as 5, 6, or 7 bits, the unused bits are set to "zero."

SYNCHRONOUS TRANSMISSION

As in Asynchronous transmission, the TxD output remains "high" (marking) until the μPD8251 and μPD8251A receive the first character (usually a SYNC character) from the processor. After a Command Instruction has set TxEN and after Clear to Send ($\overline{CTS}$) goes low, the first character is serially transmitted. Data is shifted out on the falling edge of $\overline{TxC}$ and the same rate as $\overline{TxC}$.

Once transmission has started, Synchronous Mode format requires that the serial data stream at TxD continue at the $\overline{TxC}$ rate or SYNC will be lost. If a data character is not provided by the processor before the μPD8251 and μPD8251A Transmit Buffer becomes empty, the SYNC character(s) loaded directly following the Mode Instruction will be automatically inserted in the TxD data stream. The SYNC character(s) are inserted to fill the line and maintain synchronization until new data characters are available for transmission. If the μPD8251 and μPD8251A become empty, and must send the SYNC character(s), the TxEMPTY output is raised to signal the processor that the Transmitter Buffer is empty and SYNC characters are being transmitted. TxEMPTY is automatically reset by the next character from the processor.

SYNCHRONOUS RECEIVE

In Synchronous Receive, character synchronization can be either external or internal. If the internal SYNC mode has been selected, and the Enter HUNT (EH) bit has been set by a Command Instruction, the receiver goes into the HUNT mode.

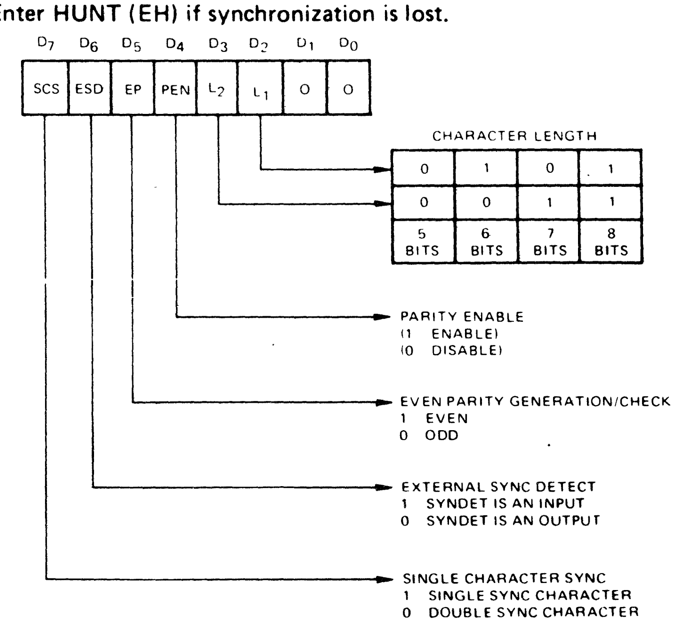
Incoming data on the RxD input is sampled on the rising edge of $\overline{RxC}$, and the Receive Buffer is compared with the first SYNC character after each bit has been loaded until a match is found. If two SYNC characters have been programmed, the next received character is also compared. When the SYNC character(s) programmed have been detected, the μPD8251 and μPD8251A leave the HUNT mode and are in character synchronization. At this time, the SYNDET (output) is set high. SYNDET is automatically reset by a STATUS READ.

If external SYNC has been specified in the Mode Instruction, a "one" applied to the SYNDET (input) for at least one $\overline{RxC}$ cycle will synchronize the USART.

Parity and Overrun Errors are treated the same in the Synchronous as in the Asynchronous Mode. If not in HUNT, parity will continue to be checked even if the receiver is not enabled. Framing errors do not apply in the Synchronous format.

The processor may command the receiver to enter the HUNT mode with a Command Instruction which sets Enter HUNT (EH) if synchronization is lost.

MODE INSTRUCTION FORMAT SYNCHRONOUS MODE



TRANSMIT/RECEIVE FORMAT SYNCHRONOUS MODE

PROCESSOR BYTES (5-8 BITS CHAR)

DATA CHARACTERS

ASSEMBLED SERIAL DATA OUTPUT (TxD)

SYNC CHAR 1	SYNC CHAR 2	DATA CHARACTERS

TRANSMIT FORMAT

SERIAL DATA INPUT (RxD)

SYNC CHAR 1	SYNC CHAR 2	DATA CHARACTERS

PROCESSOR BYTES (5-8 BITS CHAR) ①

DATA CHARACTERS

RECEIVE FORMAT

Note ① If character length is defined as 5, 6 or 7 bits, the unused bits are set to "zero."

COMMAND INSTRUCTION FORMAT

After the functional definition of the μPD8251 and μPD8251A has been specified by the Mode Instruction and the SYNC character(s) have been entered (if in SYNC mode), the USART is ready to receive Command Instructions and begin communication. A Command Instruction is used to control the specific operation of the format selected by the Mode Instruction. Enable Transmit, Enable Receive, Error Reset and Modem Controls are controlled by the Command Instruction.

After the Mode Instruction and the SYNC character(s) (as needed) are loaded, all subsequent "control writes" (C/$\overline{D}$ = 1) will load or overwrite the Command Instruction register. A Reset operation (internal via CMD IR or external via the RESET input) will cause the μPD8251 and μPD8251A to interpret the next "control write", which must immediately follow the reset, as a Mode Instruction.

STATUS READ FORMAT

It is frequently necessary for the processor to examine the status of an active interface device to determine if errors have occurred or if there are other conditions which require a response from the processor. The μPD8251 and μPD8251A have features which allow the processor to read the device status at any time. A data fetch is issued by the processor while holding the C/$\overline{D}$ input "high" to obtain device Status Information. Many of the bits in the status register are copies of external pins. This dual status arrangement allows the μPD8251 and μPD8251A to be used in both Polled and interrupt driven environments. Status update can have a maximum delay of 16 clock periods in the μPD8251 and 28 clock periods in the μPD8251A.

PARITY ERROR

When a parity error is detected, the PE flag is set. It is cleared by setting the ER bit in a subsequent Command Instruction. PE being set does not inhibit USART operation.

OVERRUN ERROR

If the processor fails to read a data character before the one following is available, the OE flag is set. It is cleared by setting the ER bit in a subsequent Command Instruction. Although OE being set does not inhibit USART operation, the previously received character is overwritten and lost.

FRAMING ERROR ①

If a valid STOP bit is not detected at the end of a character, the FE flag is set. It is cleared by setting the ER bit in a subsequent Command Instruction. FE being set does not inhibit USART operation.

Note: ① ASYNC mode only.

COMMAND INSTRUCTION FORMAT

D_7	D_6	D_5	D_4	D_3	D_2	D_1	D_0
EH	IR	RTS	ER	SBRK	RxE	DTR	TxEN

TRANSMIT ENABLE
1 = enable
0 = disable

DATA TERMINAL READY
"high" will force $\overline{DTR}$ output to zero

RECEIVE ENABLE
1 = enable
0 = disable

SEND BREAK CHARACTER
1 = forces TxD "low"
0 = normal operation

ERROR RESET
1 = reset all error flags PE, OE, FE

REQUEST TO SEND
"high" will force $\overline{RTS}$ output to zero

INTERNAL RESET
"high" returns USART to Mode Instruction Format

ENTER HUNT MODE
1 = enable search for Sync Characters ①

STATUS READ FORMAT

D_7	D_6	D_5	D_4	D_3	D_2	D_1	D_0
DSR	SYNDET /BD	FE	OE	PE	TxE	RxRDY	TxRDY

TxRDY → ②

DSR, SYNDET/BD, TxE, RxRDY: SAME DEFINITIONS AS I/O PINS

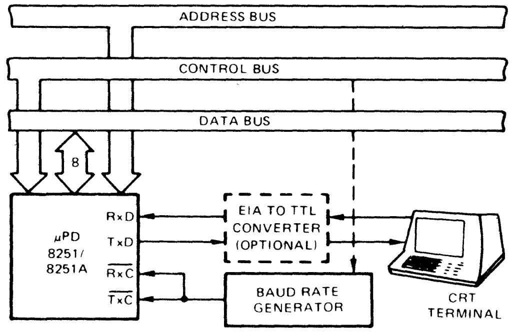
PARITY ERROR
The PE flag is set when a parity error is detected. It is reset by the ER bit of the Command Instruction. PE does not inhibit operation of the μPD8251 and μPD8251A

OVERRUN ERROR
The OE flag is set when the CPU does not read a character before the next one becomes available. It is reset by the ER bit of the Command Instruction. OE does not inhibit operation of the μPD8251 and μPD8251A; but, the previously overrun character is lost.

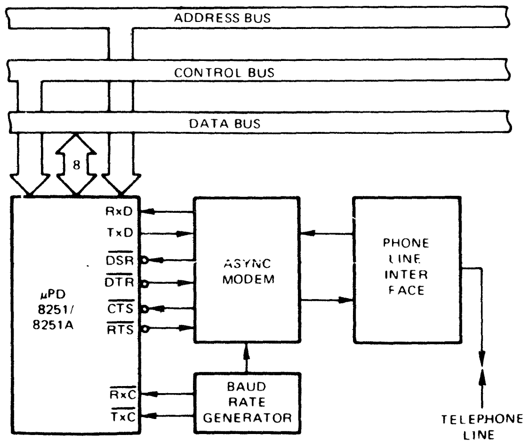
FRAMING ERROR (Async only)
The FE flag is set when a valid Stop bit is not detected at the end of every character. It is reset by the ER bit of the Command Instruction. FE does not inhibit the operation of the μPD8251 and μPD8251A.

Notes: ① No effect in ASYNC mode.
② TxRDY status bit is not totally equivalent to the TxRDY output pin, the relationship is as follows:
TxRDY status bit = DB Buffer Empty
TxRDY (pin 15) = DB Buffer Empty • $\overline{CTS}$ • TxEn

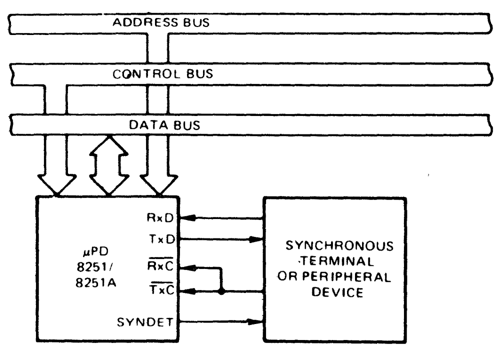
APPLICATION OF THE μPD8251 AND μPD8251A



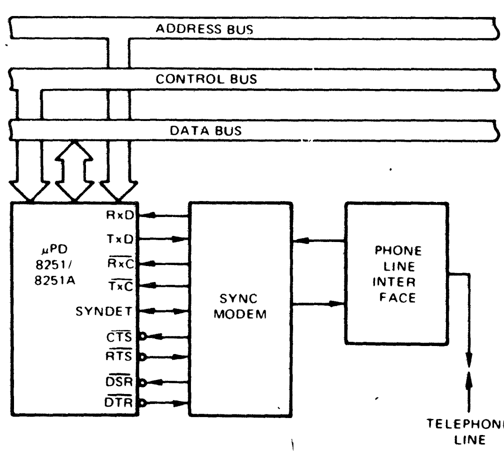
ASYNCHRONOUS SERIAL INTERFACE TO CRT TERMINAL, DC to 9600 BAUD



ASYNCHRONOUS INTERFACE TO TELEPHONE LINES

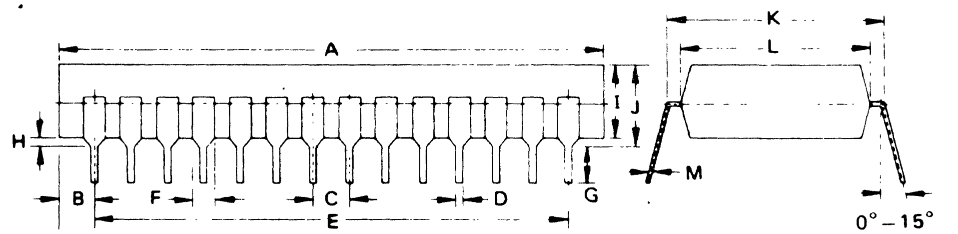


SYNCHRONOUS INTERFACE TO TERMINAL OR PERIPHERAL DEVICE



SYNCHRONOUS INTERFACE TO TELEPHONE LINES

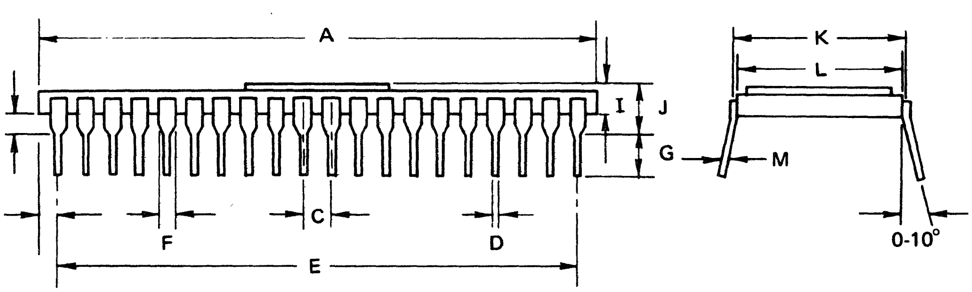
PACKAGE OUTLINES
μPD8251C
μPD8251AC



Plastic

ITEM	MILLIMETERS	INCHES
A	38.0 MAX.	1.496 MAX.
B	2.49	0.098
C	2.54	0.10
D	0.5 ± 0.1	0.02 ± 0.004
E	33.02	1.3
F	1.5	0.059
G	2.54 MIN	0.10 MIN.
H	0.5 MIN.	0.02 MIN.
I	5.22 MAX.	0.205 MAX.
J	5.72 MAX.	0.225 MAX.
K	15.24	0.6
L	13.2	0.52
M	$0.25^{+0.10}_{\;\;0.05}$	$0.01^{+0.004}_{\;\;0.002}$

μPD8251D
μPD8251AD



Ceramic

ITEM	MILLIMETERS	INCHES
A	51.5 MAX.	2.03 MAX.
B	1.62 MAX.	0.06 MAX.
C	2.54 ± 0.1	0.1 ± 0.004
D	0.5 ± 0.1	0.02 ± 0.004
E	48.26 ± 0.1	1.9 ± 0.004
F	1.02 MIN.	0.04 MIN.
G	3.2 MIN.	0.13 MIN.
H	1.0 MIN.	0.04 MIN.
I	3.5 MAX.	0.14 MAX.
J	4.5 MAX.	0.18 MAX.
K	15.24 TYP.	0.6 TYP.
L	14.93 TYP.	0.59 TYP.
M	0.25 ± 0.05	0.01 ± 0.0019

8251/8251ADS-12-80-CAT

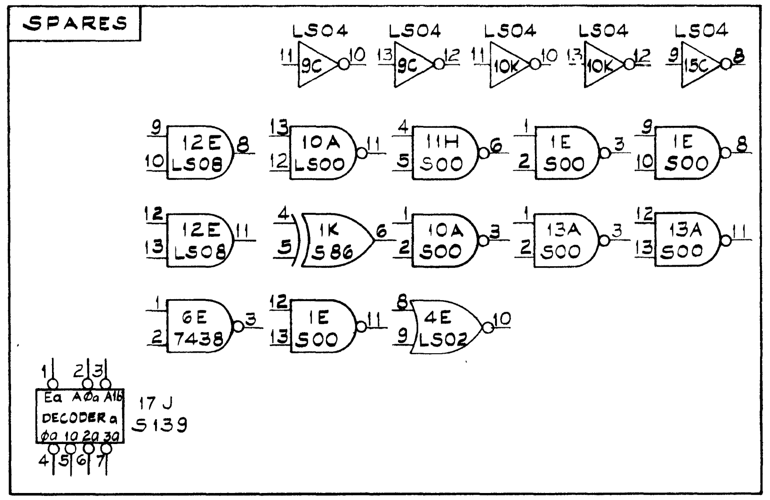
This appendix contains electrical schematics for the following ADVANTAGE PC boards.

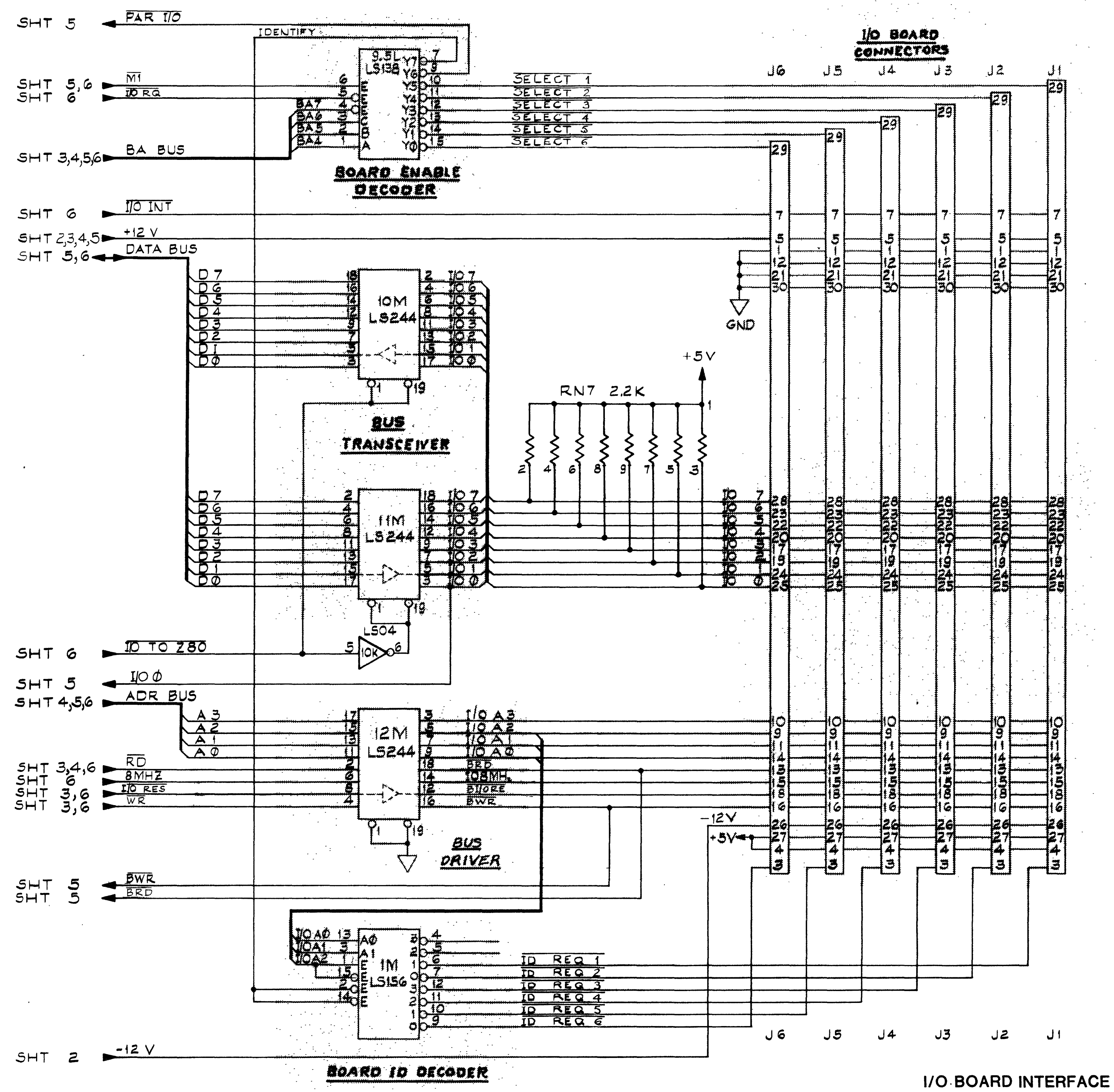
1. Main Board
2. SIO Board
3. PIO Board
4. Keyboard
5. Disk Drive
6. Video

The schematic for the Keyboard PC Board is reprinted herein with the permission from the Key Tronics Corporation.

The schematics for the Disk Drive PC Boards are reprinted herein with permission from the Tandon Corporation.

The schematics for the Video PC Board is reprinted herein with permission from the Elston Electronics Corporation.





9. IC'S AT LOCATION 17E & 18F ARE OMITTED WHEN AN8048 IS USED AT LOCATION 18C
8. (*) .047µF 56 TOTAL
(**) .1µF 76 TOTAL
7. FOR NOTES 7 SEE SHT 2
6. FOR NOTES 6 SEE SHT 2
5.▲ INDICATE SOCKETED IC'S
4.*HEATSINK REQUIRED
3. ALL CAPACITOR VALUE ARE IN MICROFARADS (µF)
2. ALL RESISTOR VALUE ARE IN OHMS(Ω), 1/4W, ±5%, 1% ARE METAL FILM, RN55D.
1. REFERENCE DOCUMENTS ASSY & P/L ARE # 00102

NOTES: UNLESS OTHERWISE SPECIFIED

I/O BOARD INTERFACE

REV	DESCRIPTION	
J	ECO# 263	J.F. 10-6-81
H	ECO# 257	J.F. 8-28-81
F	ECO# 256	P.A 8-25-81
E	ECO# 254	P.A. 8-21-81
D	ECO#	
C	ECO#231	P.A 7-23-81
B	ECO#230	P.A 7-15-81
A	INITIAL REL	6-15-81

UNLESS OTHERWISE SPECIFIED
ALL METAL PARTS MUST BE FREE OF BURRS AND SHARP EDGES
ALL ✓ SURFACES TO BE ✓
TOLERANCES: ANGLES ± 1° FRACTIONS ± 1/16 DECIMALS .X ±.030 .XX ±.020 .XXX ±.010
MATERIAL
FINISH

DO NOT SCALE DRAWING
DIMENSIONS ARE IN INCHES

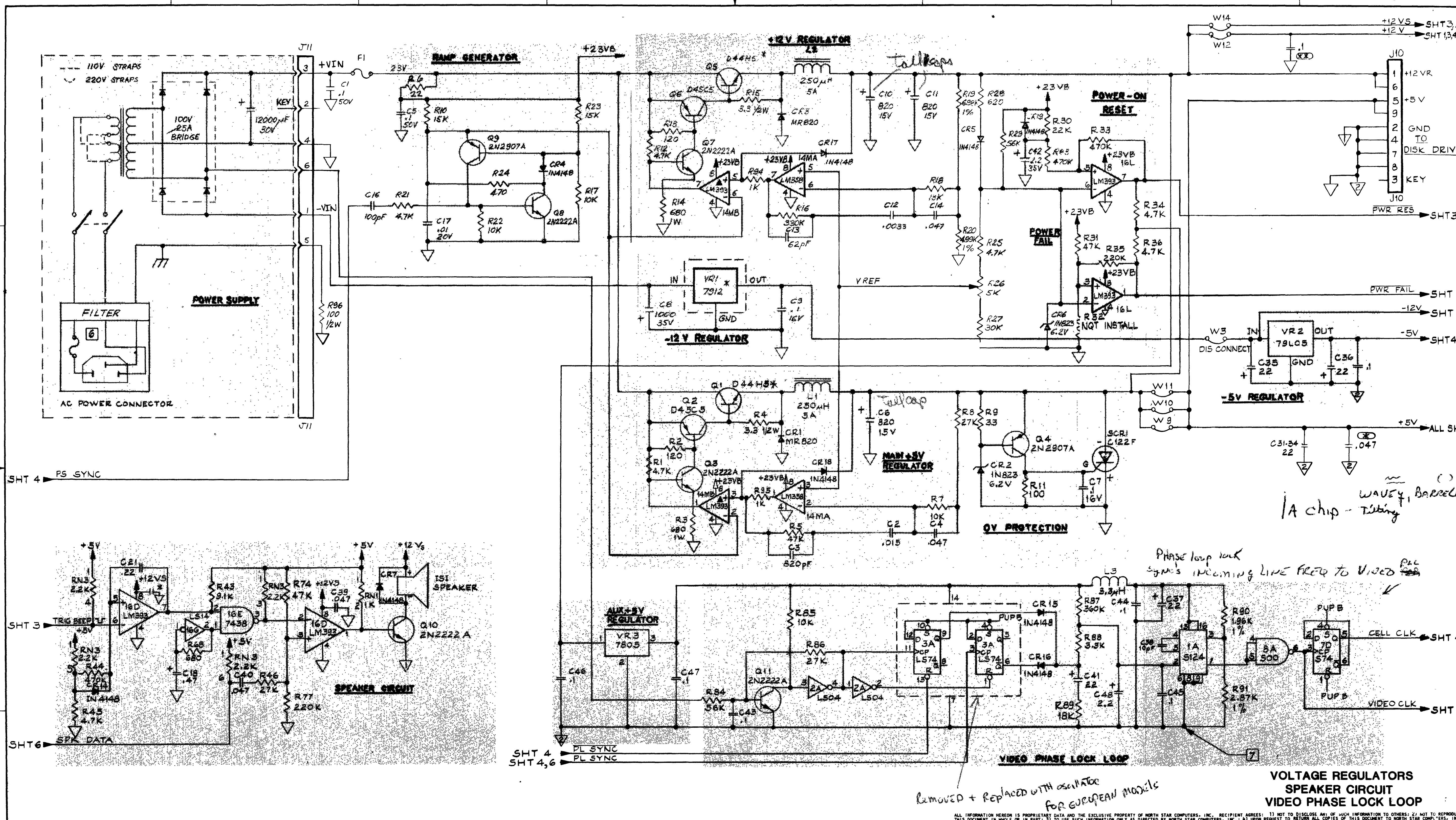
DRAWN	P-A	DATE 4-27-81
CHECKED	R.A.	4-28-81
ENG		
ENG MGR	JES	4-28-81
MFG		
MODEL	NEXT ASSY 00102	

NorthStar™

TITLE ADVANTAGE MAIN PCB SCHEMATIC

SIZE D | DWG NO 00100 | REV J

SCALE 1/1 | SHEET 1 OF 6



VOLTAGE REGULATORS
SPEAKER CIRCUIT
VIDEO PHASE LOCK LOOP

NOTES: UNLESS OTHERWISE SPECIFIED

6. LINE FUSE VALUE: 3A FOR 115 V OPERATION AND 1.5 A FOR 230V OPERATION
7. THESE GROUND CONNECTIONS ARE TO BE SHORT.

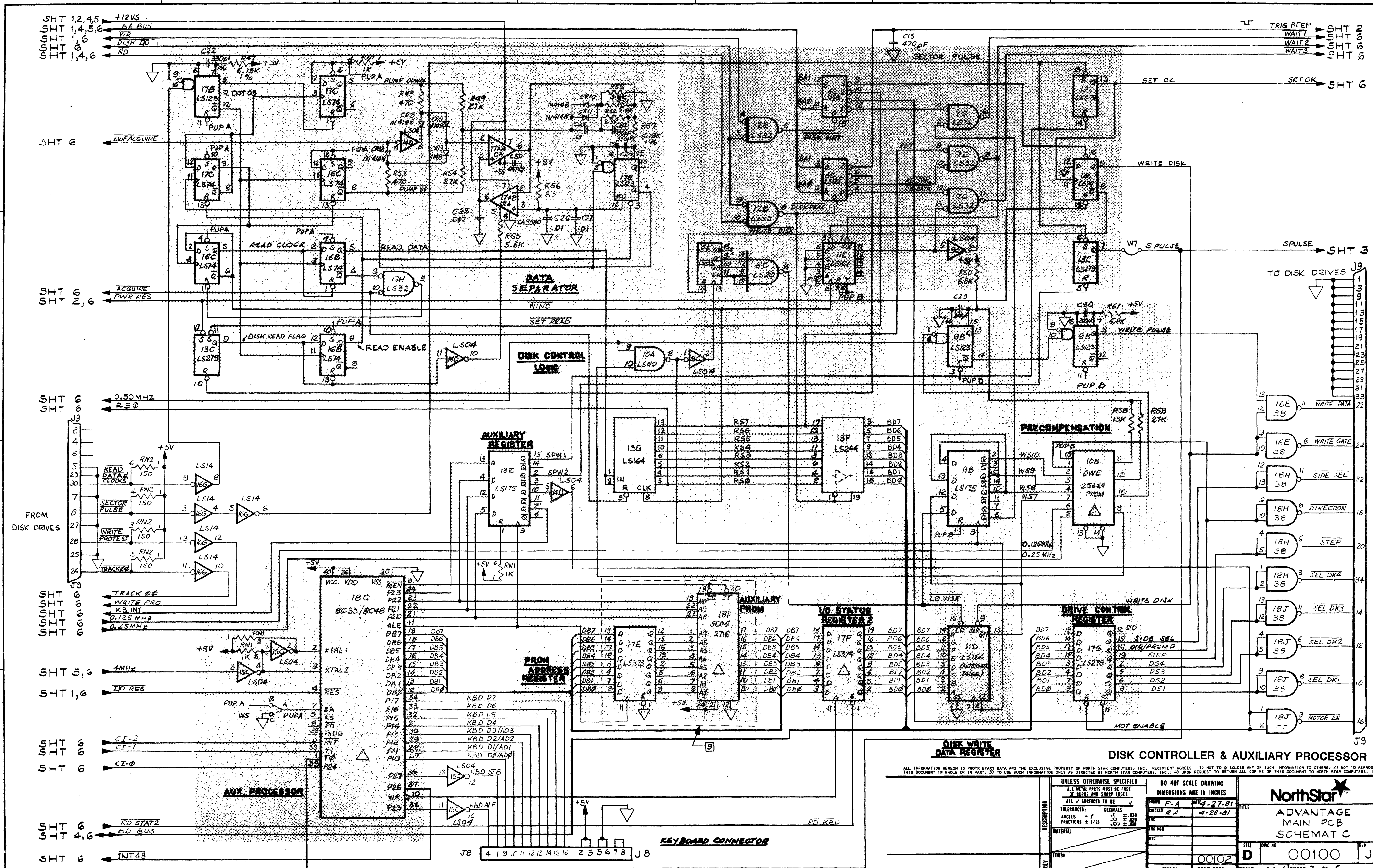
UNLESS OTHERWISE SPECIFIED
ALL METAL PARTS MUST BE FREE OF BURRS AND SHARP EDGES
DO NOT SCALE DRAWING
DIMENSIONS ARE IN INCHES

NorthStar

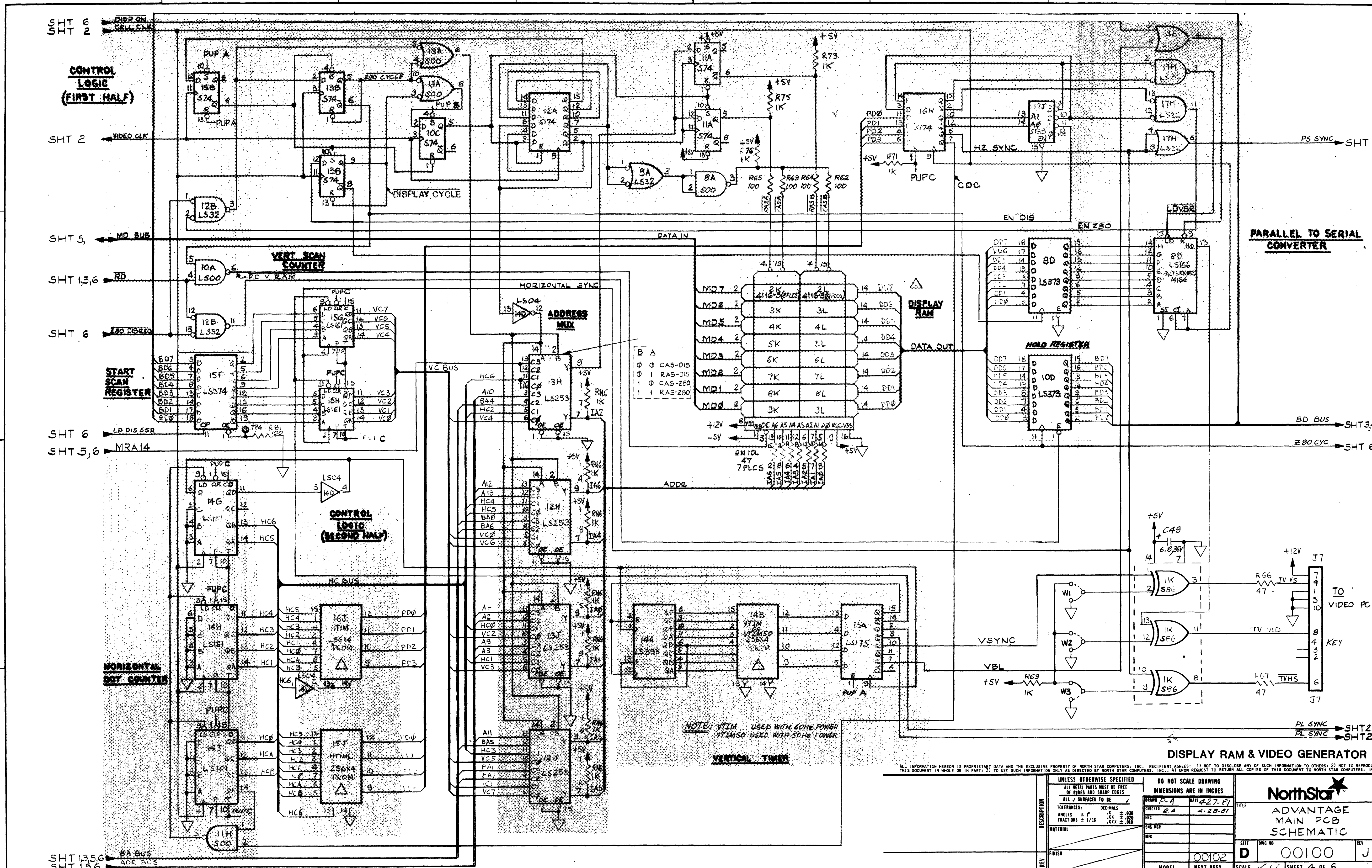
TITLE: ADVANTAGE MAIN PCB SCHEMATIC

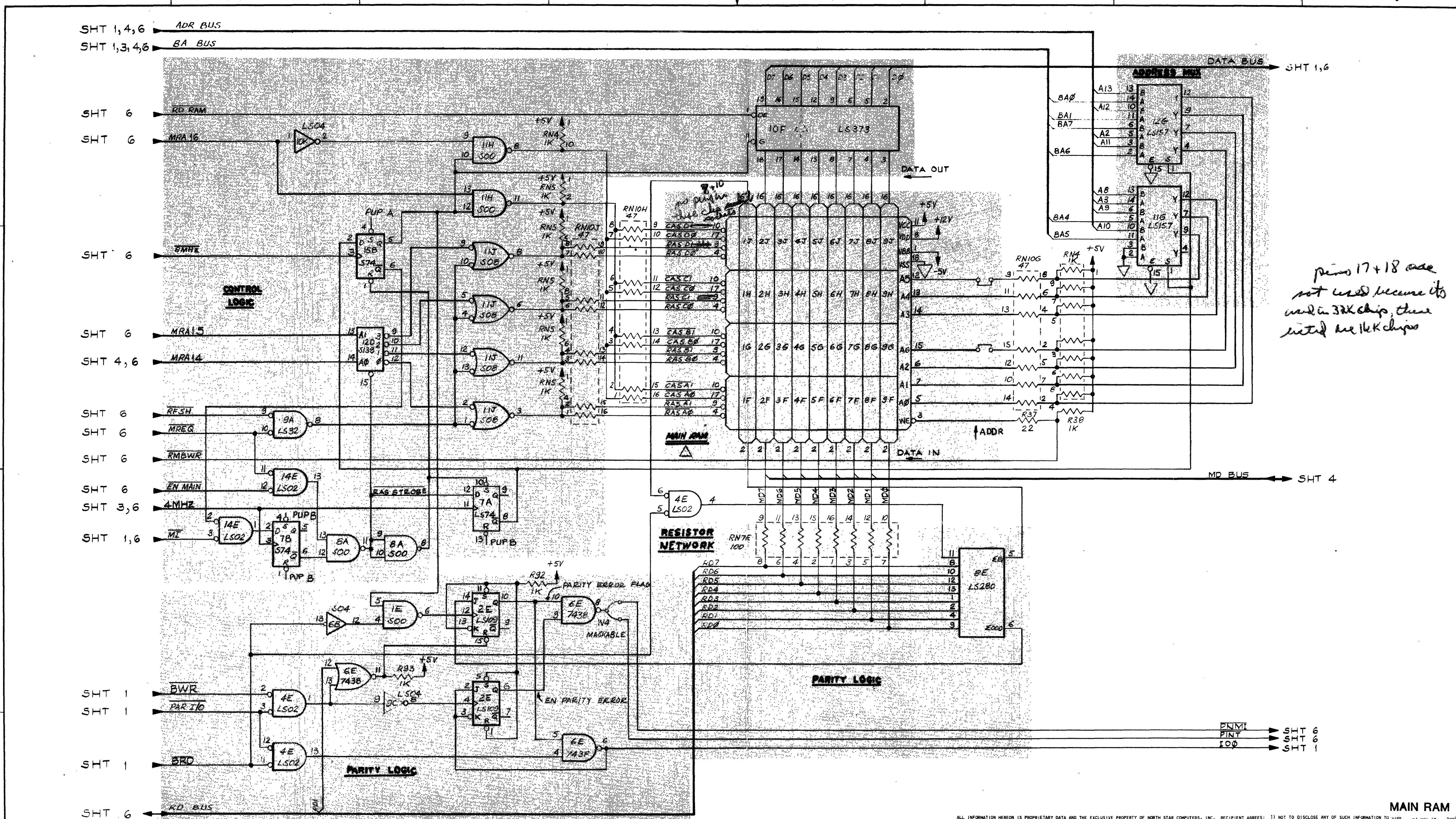
SIZE D | DWG NO. 00100 | REV J

SCALE ✓/✓ | SHEET 2 OF 6



DATA SEPARATOR
DISK CONTROL LOGIC
AUXILIARY REGISTER
PRECOMPENSATION
AUX. PROCESSOR
PROM ADDRESS REGISTER
AUXILIARY PROM
I/O STATUS REGISTER 2
DISK WRITE DATA REGISTER
DRIVE CONTROL REGISTER
KEYBOARD CONNECTOR
FROM DISK DRIVES
TO DISK DRIVES
SECTOR PULSE
WRITE DISK
READ DATA
READ CLOCK
DISK READ FLAG
READ ENABLE
SET OK
SET READ
WIND
WRITE PULSE
S PULSE
PUMP DOWN
PUMP UP
8035/8048
DISK CONTROLLER & AUXILIARY PROCESSOR
ALL INFORMATION HEREON IS PROPRIETARY DATA AND THE EXCLUSIVE PROPERTY OF NORTH STAR COMPUTERS, INC.
UNLESS OTHERWISE SPECIFIED
DO NOT SCALE DRAWING
DIMENSIONS ARE IN INCHES
NorthStar
ADVANTAGE
MAIN PCB
SCHEMATIC
DRAWN P-A 4-27-81
CHECKED R.A 4-28-81
NEXT ASSY 00102
SIZE D
DWG NO 00100
REV J
SCALE
SHEET 3 OF 6


CONTROL LOGIC (FIRST HALF)
CONTROL LOGIC (SECOND HALF)
VERT SCAN COUNTER
START SCAN REGISTER
ADDRESS MUX
DISPLAY RAM
HOLD REGISTER
PARALLEL TO SERIAL CONVERTER
HORIZONTAL DOT COUNTER
VERTICAL TIMER
SHT 6 DISP ON
SHT 2 CELL CLK
SHT 2 VIDEO CLK
SHT 5, MD BUS
SHT 1,3,6 RD
SHT 6 Z80 DISREQ
SHT 6 LD DIS SSR
SHT 5,6 MRA14
SHT 1,3,5,6 BA BUS
SHT 1,5,6 ADR BUS
PS SYNC SHT 2
BD BUS SHT 3,4,6
Z80 CYC SHT 6
PL SYNC SHT 2,6
PL SYNC SHT 2
HORIZONTAL SYNC
DISPLAY CYCLE
Z80 CYCLE
HZ SYNC
EN DIS
EN Z80
LDVSR
DATA IN
DATA OUT
ADDR
VC BUS
HC BUS
VSYNC
VBL
B A
Φ Φ CAS-DIS
Φ 1 RAS-DIS
1 Φ CAS-Z80
1 1 RAS-Z80
4116-3 (8PLCS)
RN 10L 47 7PLCS
TO VIDEO PCB
KEY
J7
NOTE: VTIM USED WITH 60Hz POWER
VTIM50 USED WITH 50Hz POWER
DISPLAY RAM & VIDEO GENERATOR
NorthStar
ADVANTAGE MAIN PCB SCHEMATIC
UNLESS OTHERWISE SPECIFIED
DO NOT SCALE DRAWING
DIMENSIONS ARE IN INCHES
DRAWN P.A 4-27-81
CHECKED P.A 4-28-81
00102
MODEL
NEXT ASSY
SIZE D
DWG NO 00100
REV J
SHEET 4 OF 6



MAIN RAM

ALL INFORMATION HEREON IS PROPRIETARY DATA AND THE EXCLUSIVE PROPERTY OF NORTH STAR COMPUTERS, INC. RECIPIENT AGREES: 1) NOT TO DISCLOSE ANY OF SUCH INFORMATION TO OTHERS; 2) NOT TO REPRODUCE THIS DOCUMENT IN WHOLE OR IN PART; 3) TO USE SUCH INFORMATION ONLY AS DIRECTED BY NORTH STAR COMPUTERS, INC.; 4) UPON REQUEST TO RETURN ALL COPIES OF THIS DOCUMENT TO NORTH STAR COMPUTERS, INC.

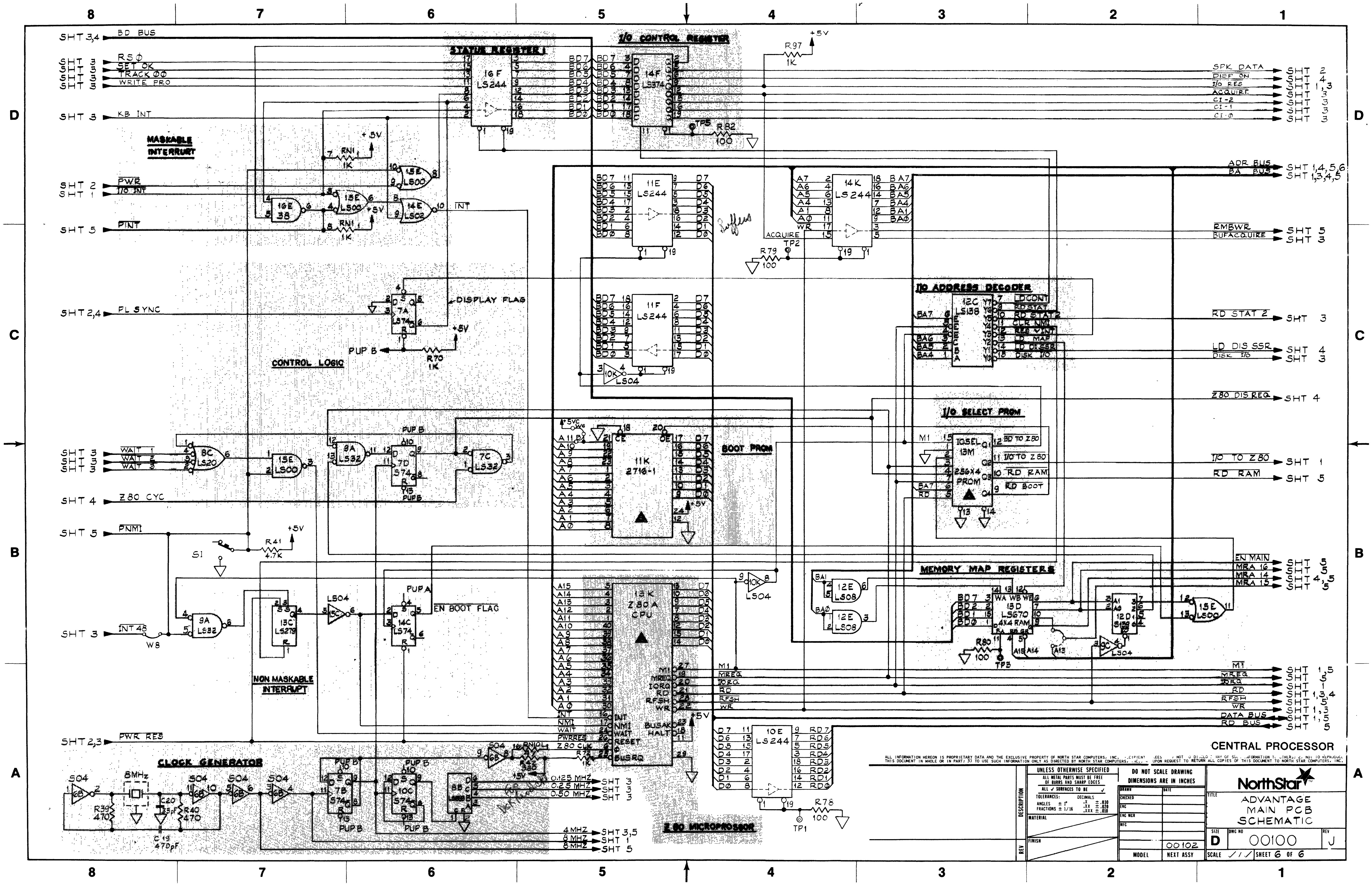
UNLESS OTHERWISE SPECIFIED	ALL METAL PARTS MUST BE FREE OF BURRS AND SHARP EDGES
TOLERANCES: ANGLES ± 1° FRACTIONS ± 1/16	DECIMALS .X ± .030 .XX ± .020 .XXX ± .010
DO NOT SCALE DRAWING	DIMENSIONS ARE IN INCHES
DRAWN P.A.	DATE 4-27-81
CHECKED R.A.	4-28-81
MODEL	NEXT ASSY 00102

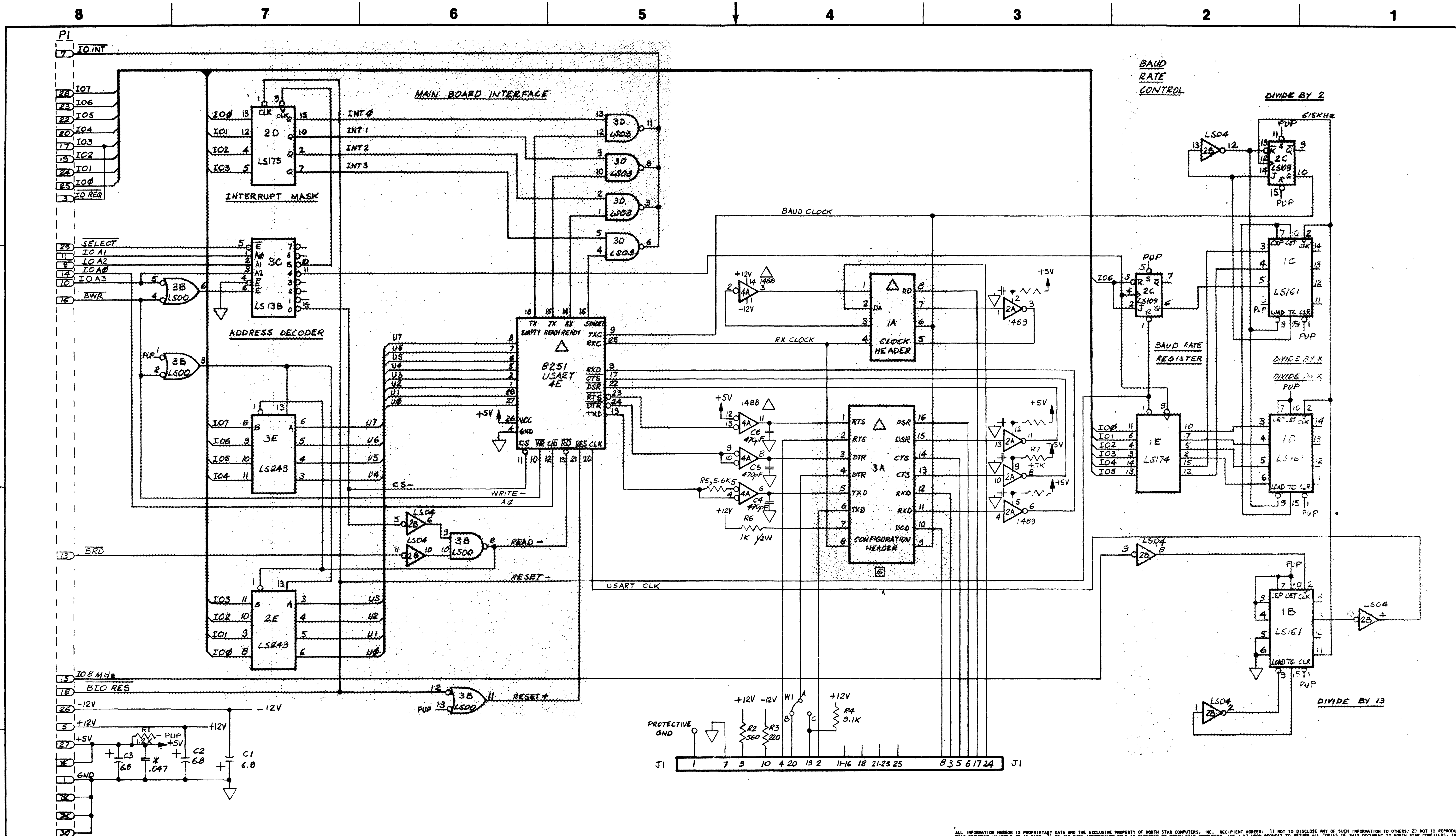
NorthStar

TITLE ADVANTAGE MAIN PCB SCHEMATIC

SIZE D DWG NO 00100 REV J

SCALE 1/1 SHEET 5 OF 6


STATUS REGISTER
16F LS244
I/O CONTROL REGISTER
14F LS374
MASKABLE INTERRUPT
CONTROL LOGIC
DISPLAY FLAG
NON MASKABLE INTERRUPT
CLOCK GENERATOR
8MHz
11E LS244
11F LS244
14K LS244
Buffers
BOOT PROM
11K 2716-1
Z80 MICROPROSSOR
13K Z80A CPU
10E LS244
I/O ADDRESS DECODER
12C LS138
I/O SELECT PROM
13M 256X4 PROM
MEMORY MAP REGISTERS
15D LS670 4X4 RAM
EN BOOT FLAG
BD BUS
ADR BUS
BA BUS
DATA BUS
RD BUS
CENTRAL PROCESSOR
ALL INFORMATION HEREON IS PROPRIETARY DATA AND THE EXCLUSIVE PROPERTY OF NORTH STAR COMPUTERS, INC.
UNLESS OTHERWISE SPECIFIED
DO NOT SCALE DRAWING
DIMENSIONS ARE IN INCHES
NorthStar
ADVANTAGE MAIN PCB SCHEMATIC
D
00100
J
00102
SHEET 6 OF 6



6. STRAP HORIZONTALLY FOR CONNECTION TO TERMINAL
5. △ INDICATES SOCKETED IC'S
4. (X) .047 µF TOTAL 10
3. ALL CAPACITOR VALUES ARE IN MICROFARAD (µF)
2. ALL RESISTOR VALUES ARE IN OHM (Ω) 1/4W 5%
1. REFERENCE DOCUMENTS : ASSY & P/L ARE 00113

NOTES: UNLESS OTHERWISE SPECIFIED

REV	DESCRIPTION
A	INIT REL 10-27-81

UNLESS OTHERWISE SPECIFIED
ALL METAL PARTS MUST BE FREE OF BURRS AND SHARP EDGES
ALL ✓ SURFACES TO BE ✓
TOLERANCES: DECIMALS
ANGLES ± 1° .X ±.030 .XX ±.010
FRACTIONS ± 1/16 .XXX ±.005
MATERIAL
FINISH

DO NOT SCALE DRAWING
DIMENSIONS ARE IN INCHES

		DATE
DRAWN	P-A	5-6-81
CHECKED	R.A	5-7-81
ENG	[illegible]	10/27/81
MFG	[illegible]	10/27/81

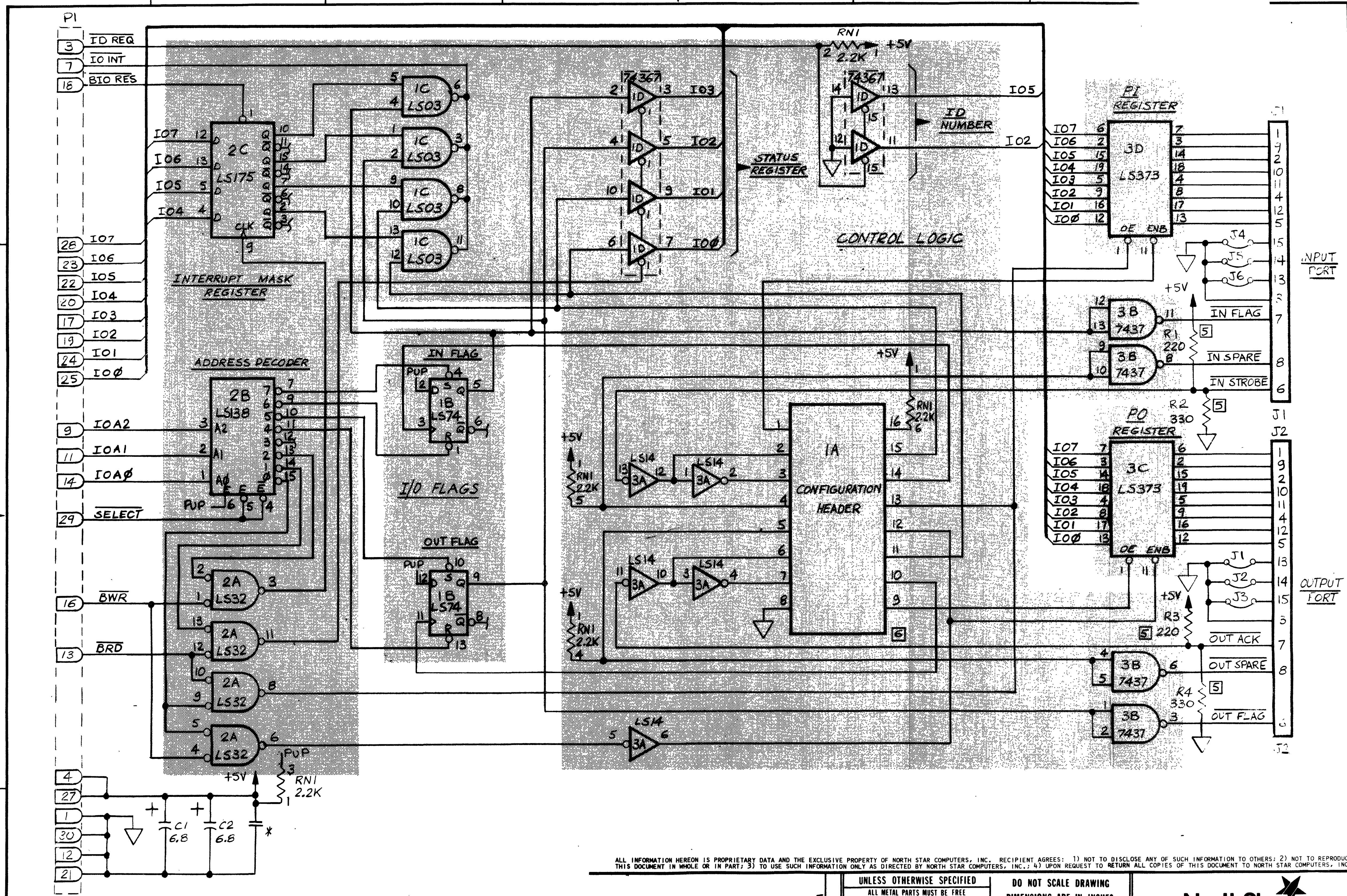
MODEL	NEXT ASSY
ADVANTAGE	00113

NorthStar

TITLE ADVANTAGE SIO PCB SCHEMATIC

SIZE D | DWG NO. 00111 | REV A

SCALE 1/1 | SHEET 1 OF 1



6. STRAP HORIZONTALLY FOR MOST PRINTER
5. R1 TO R4 ARE OPTIONAL TERMINATORS
4. (*) .047 μF TOTAL 5
3. ALL CAPACITOR VALUE ARE IN MICROFARAD (μF)
2. ALL RESISTOR VALUE ARE IN OHM (Ω) 1/4W 5%
1. REFERENCE DOCUMENTS: ASSY & P/L ARE 00148

NOTES : UNLESS OTHERWISE SPECIFIED

REV	DESCRIPTION
A	INIT REL 11-9-81

UNLESS OTHERWISE SPECIFIED
ALL METAL PARTS MUST BE FREE OF BURRS AND SHARP EDGES
ALL ✓ SURFACES TO BE ✓
TOLERANCES: ANGLES ± 1° FRACTIONS ± 1/16
DECIMALS .X ± .030 .XX ± .020 .XXX ± .010
MATERIAL
FINISH

DO NOT SCALE DRAWING
DIMENSIONS ARE IN INCHES

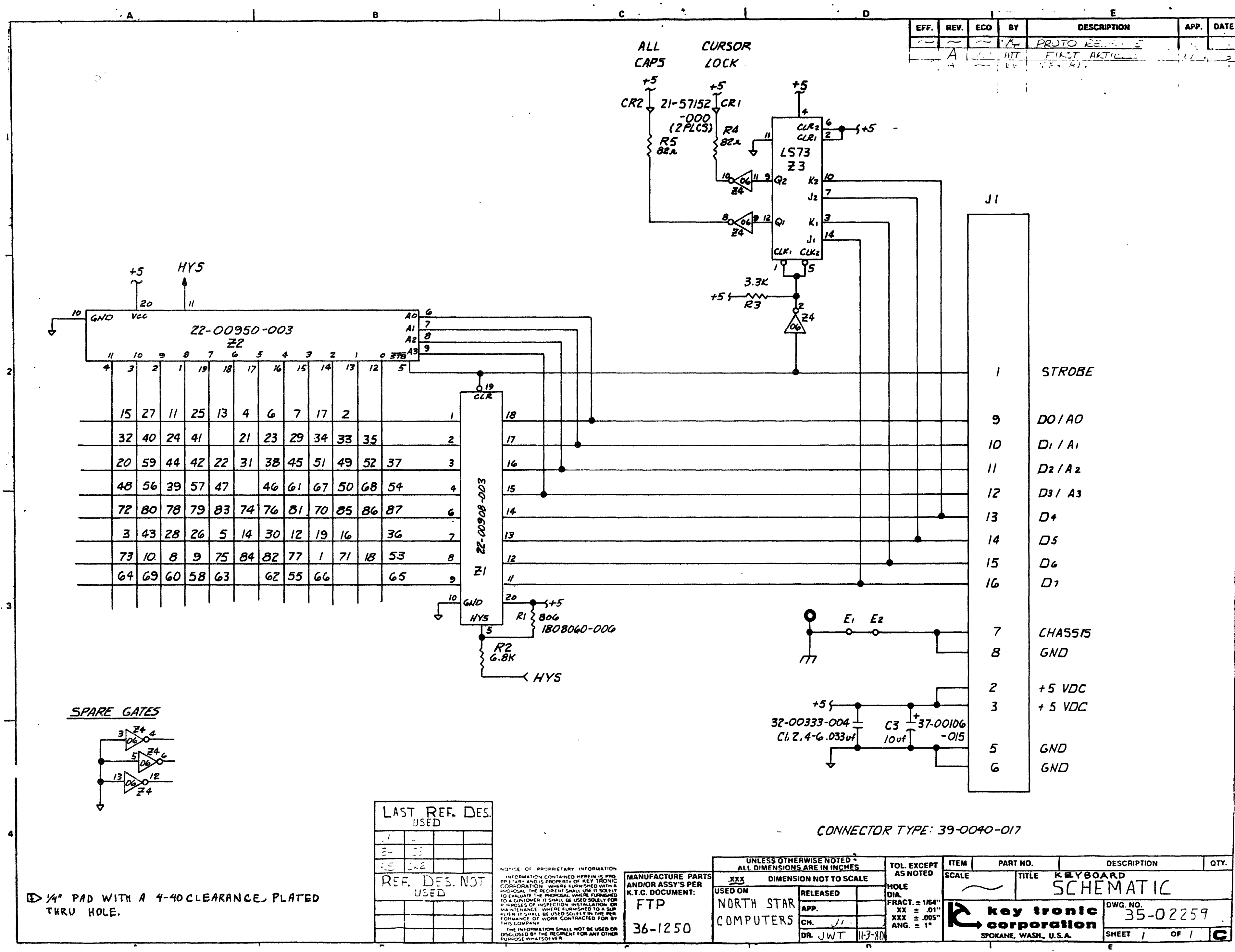
DRAWN	P-A	DATE 5-4-81
CHECKED	R.A.	5-5-81
ENG		
ENG MGR	JeS	5-5-81
MFG		
		00148
	MODEL	NEXT ASSY

NorthStar

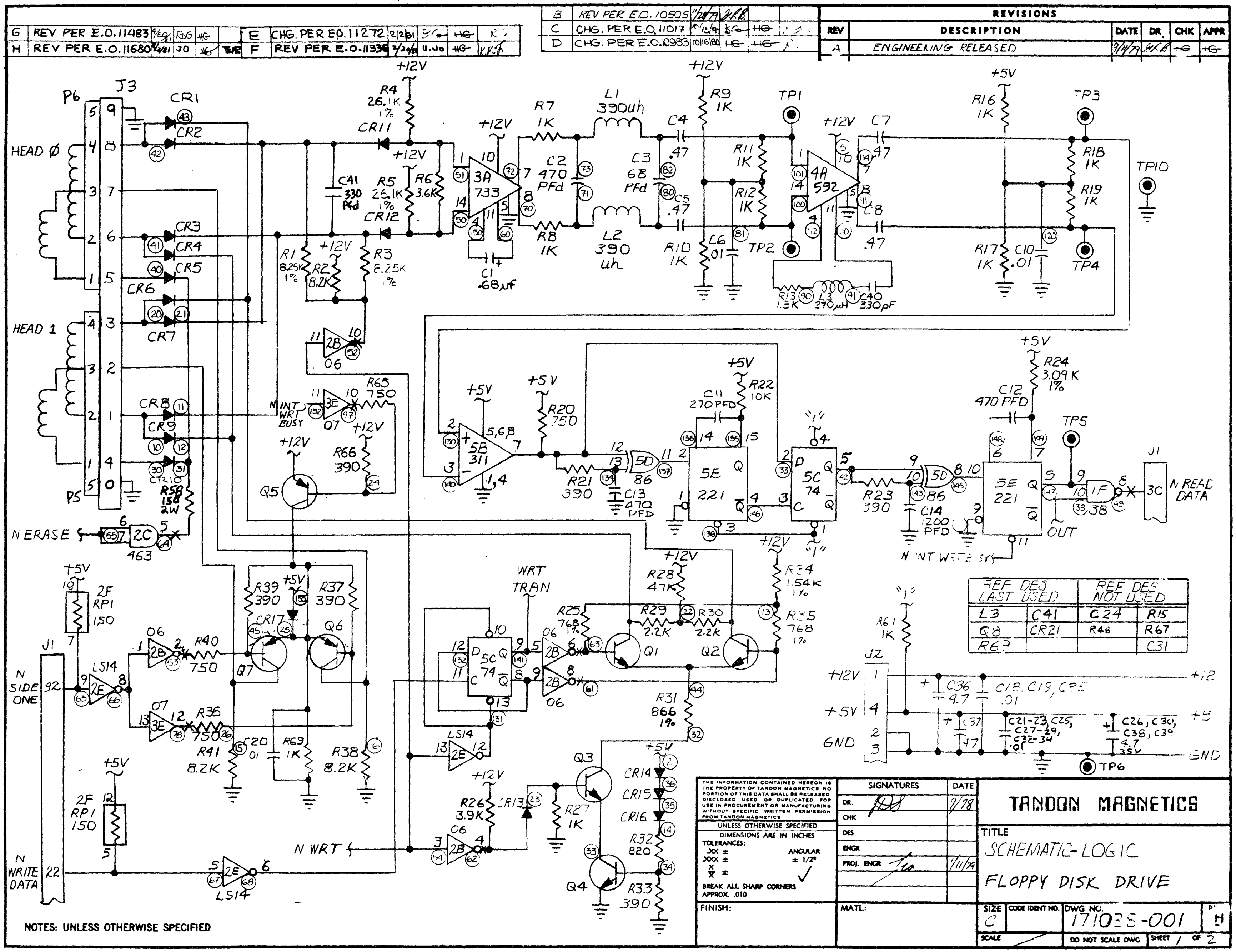
TITLE: ADVANTAGE PIO PCB SCHEMATIC

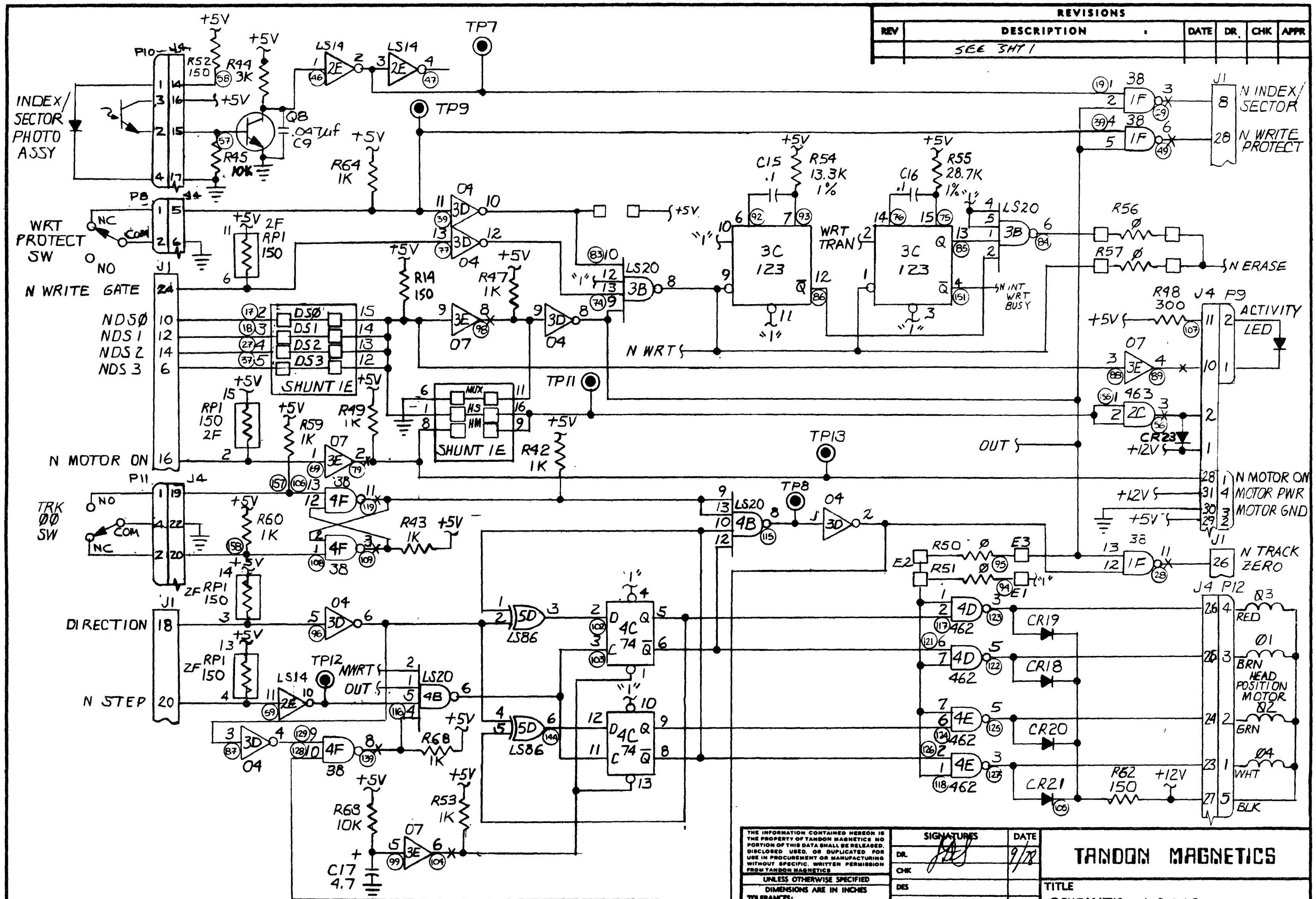
SIZE C | DWG NO. 00146 | REV A

SCALE 1/1 | SHEET 1 OF 1



ALL CAPS
CURSOR LOCK
CR2
21-57152 -000 (2PLCS)
CR1
R5 82Ω
R4 82Ω
LS73 Z3
3.3K R3
Z4
22-00950-003 Z2
HYS
22-00908-003 Z1
R1 806 1808060-006
R2 6.8K
J1
1 STROBE
9 D0 / A0
10 D1 / A1
11 D2 / A2
12 D3 / A3
13 D4
14 D5
15 D6
16 D7
7 CHASSIS
8 GND
2 +5 VDC
3 +5 VDC
5 GND
6 GND
E1 E2
32-00333-004 C1, 2, 4-6 .033uf
C3 10uf 37-00106 -015
SPARE GATES
CONNECTOR TYPE: 39-0040-017
1/4" PAD WITH A 4-40 CLEARANCE, PLATED THRU HOLE.
LAST REF. DES. USED
REF. DES. NOT USED
MANUFACTURE PARTS AND/OR ASSY'S PER K.T.C. DOCUMENT:
FTP
36-1250
UNLESS OTHERWISE NOTED ALL DIMENSIONS ARE IN INCHES
DIMENSION NOT TO SCALE
USED ON
NORTH STAR COMPUTERS
RELEASED
APP.
CH.
DR. JWT 11-3-80
TOL. EXCEPT AS NOTED
HOLE DIA.
FRACT. ± 1/64"
XX ± .01"
XXX ± .005"
ANG. ± 1°
ITEM
PART NO.
DESCRIPTION
QTY.
SCALE
TITLE KEYBOARD SCHEMATIC
key tronic corporation
SPOKANE, WASH., U.S.A.
DWG. NO. 35-02259
SHEET 1 OF 1
C
EFF.
REV.
ECO
BY
DESCRIPTION
APP.
DATE
PROTO RELEASE
A
FIRST ARTICLE


REVISIONS
REV | DESCRIPTION | DATE | DR. | CHK | APPR
A | ENGINEERING RELEASED
B | REV PER E.O. 10505
C | CHG. PER E.O. 11017
D | CHG. PER E.O. 10983
E | CHG. PER E.O. 11272
F | REV PER E.O. 11336
G | REV PER E.O. 11483
H | REV PER E.O. 11680
REF DES LAST USED | REF DES NOT USED
L3 | C41 | C24 | R15
Q8 | CR21 | R48 | R67
R69 | | | C31
TANDON MAGNETICS
TITLE
SCHEMATIC-LOGIC
FLOPPY DISK DRIVE
SIZE C
DWG NO. 171035-001
H
SHEET 1 OF 2
DO NOT SCALE DWG
UNLESS OTHERWISE SPECIFIED
DIMENSIONS ARE IN INCHES
BREAK ALL SHARP CORNERS APPROX. .010
FINISH:
MATL:
NOTES: UNLESS OTHERWISE SPECIFIED
HEAD Ø
HEAD 1
N ERASE
N SIDE ONE
N WRITE DATA
N WRT
WRT TRAN
N READ DATA
N INT WRT BUSY
TP1
TP2
TP3
TP4
TP5
TP6
TP10



REVISIONS					
REV	DESCRIPTION	DATE	DR.	CHK	APPR
	SEE SHT 1				

4. ALL TRANSISTORS NPN ARE 2N4124 & PNP ARE 2N4125
3. ALL DIODES ARE IN4446
2. ALL CAPS ARE IN UFD
1. ALL RESISTORS ARE IN OHMS, 1/4W, 5%

NOTES: UNLESS OTHERWISE SPECIFIED

UNLESS OTHERWISE SPECIFIED DIMENSIONS ARE IN INCHES
TOLERANCES: .XX ± .XXX ± X ± ANGULAR ± 1/2°
BREAK ALL SHARP CORNERS APPROX. .010

SIGNATURES		DATE
DR.		9/78
CHK		
DES		
ENGR		
PROJ. ENGR		1/79

FINISH:

MATL:

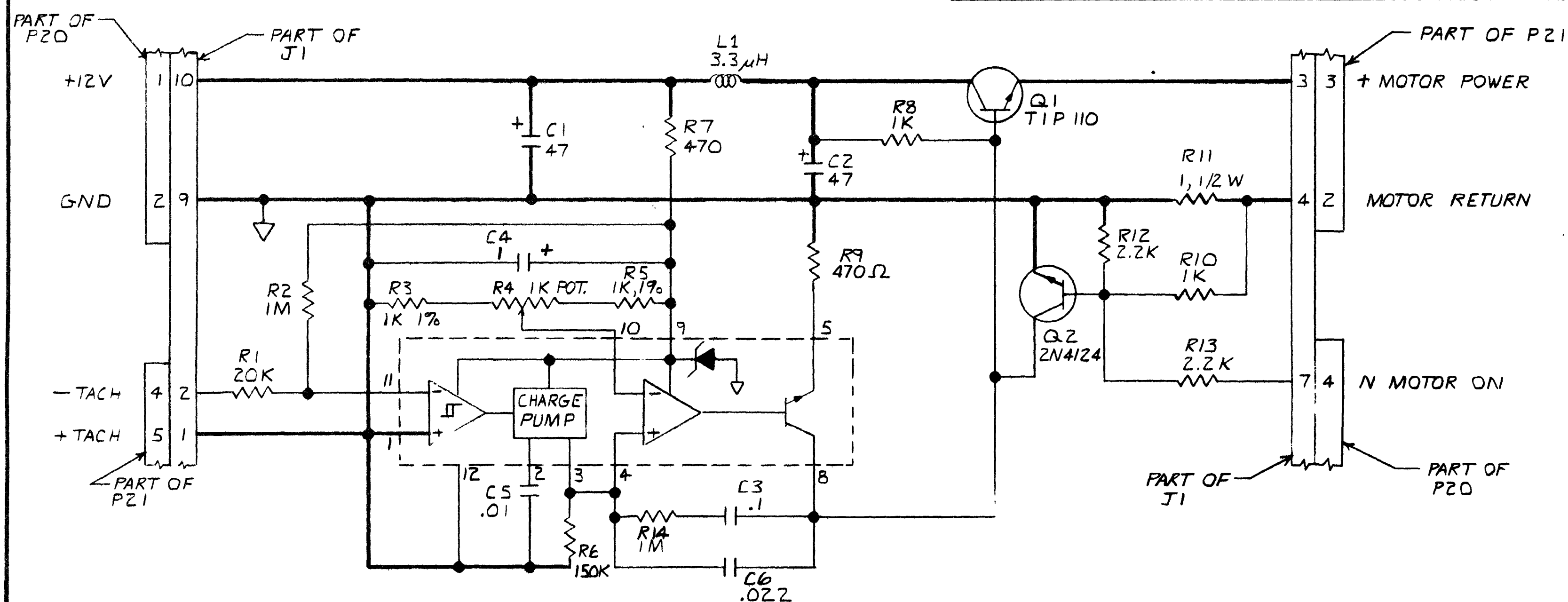
TANDON MAGNETICS

TITLE
SCHEMATIC - LOGIC.
FLOPPY DISK DRIVE

SIZE	CODE IDENT NO.	DWG NO.	REV
C		171038-001	H
SCALE		DO NOT SCALE DWG	SHEET 2 OF 2

P.C.B.A. 171111-001
ART WORK 178900-001
DETAIL 178901-001

REVISIONS					
REV	DESCRIPTION	DATE	DR	CHK	APPR
A	ENGINEERING RELEASE E.O. 10499	11/5/79	G.R.B.		
B	REV. PER E.O. 10547	12/13/79	G.R.B.		
C	REV. PER E.O. 10656	2/26/80	G.R.B.		



3. CAPACITORS ARE IN μF, ±20%, 35V.
2. 1% RESISTORS ARE 1/8 W.
1. RESISTORS ARE IN OHMS, ± 5%, 1/4 W.
NOTES: UNLESS OTHERWISE SPECIFIED

NEXT ASSY	FIRST USE
APPLICATION	

UNLESS OTHERWISE SPECIFIED
DIMENSIONS ARE IN INCHES
TOLERANCES:
.XX ±
.XXX ±
X/X ±
ANGULAR ± 1/2°
BREAK ALL SHARP CORNERS APPROX. .010

FINISH:

SIGNATURES		DATE
DR.	G. Blake	7/9/79
CHK		
DES		
ENGR		
PROJ. ENGR		10/24/79

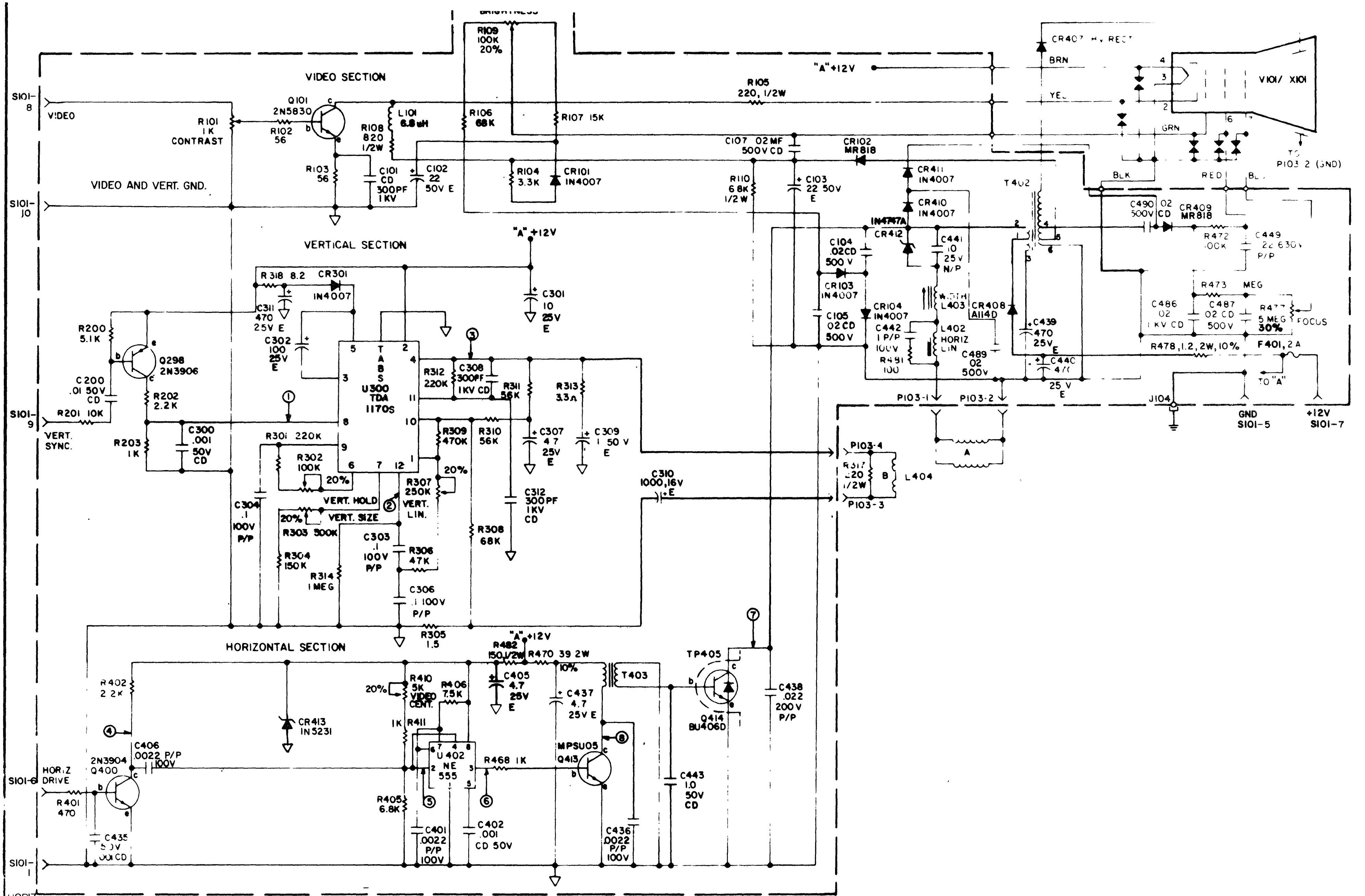
MATL:

Tandon
MAGNETICS CORPORATION

TITLE FLOPPY DISK DRIVE
SCHEMATIC - SERVO, LINEAR

SIZE	CODE IDENT NO.	DWG NO.	REV
B	—	178003-001	C
SCALE		DO NOT SCALE DWG	SHEET 1 OF 1

KLINGLER K-VEL-H 9/79



NOTES

1) ALL RESISTORS 1/4 W 5% UNLESS OTHERWISE SPECIFIED
2). ALL CAPACITORS ARE IN MFD UNLESS OTHERWISE SPECIFIED
3). REQUIRES PC BOARD 02-94168-119

CAP TYPES

E - ELECTROLYTIC
T - TANTALUM
CD - CERAMIC DISC
P/P - PLASTIC PACKAGE
N/P - NONPOLARIZED

NO.	REVISION	BY	DATE	
3	CR408 WAS V334X	DAD	10/14/81	81-115
2	C309 WAS CD,	DAD	9/18/81	ECN
1	ADDED NOTE 3, C301 WAS 20V,T C307 & C405 WERE 25 V, T	DAD	9/4/81	

TITLE VIDEO PC BOARD DM30-12B0-119-C31

MADE FOR NORTH STAR

DRAWN D CAREY DATE 6/17/81 INSP DATE

ELSTON ELECTRONICS CORP

READER COMMENTS

NorthStar™

Please use this mailer to send your comments on this manual and the program it describes. We will carefully consider your suggestions for incorporation in future versions. If you require more space for your comments, please attach a separate sheet.

NAME OF PROGRAM ____________________ VERSION NO. ____________________

MANUAL PART NO. ____________________ MANUAL SERIAL NO. ____________________

What features of the program do you like?

What features of the program don't you like or what features do you feel are missing?

Are there specific points in the manual that need clarification or correction? Give details with page and paragraph references.

Did you find the manual easy to use and understand? Do you think certain aspects should be organized differently? Was any necessary material omitted or was any material unnecessary?

Did you find sufficient information provided in the manual for proper system set up and installation?

Additional comments: ____________________

NAME ____________________ DATE ____________________

COMPANY OR ORGANIZATION ____________________

TYPE OF BUSINESS ____________________

STREET ____________________

CITY, STATE, ZIP ____________________

If you want a reply, check here ☐

Tear along perforation

FOLD FOLD

NO POSTAGE
NECESSARY
IF MAILED
IN THE
UNITED STATES

BUSINESS REPLY MAIL

FIRST CLASS PERMIT NO. 2106 SAN LEANDRO, CA

POSTAGE WILL BE PAID BY ADDRESSEE

NorthStar™

North Star Computers, Inc.
14440 Catalina Street
San Leandro, CA 94577 USA

ATTENTION:
APPLICATION SOFTWARE DEPARTMENT

NorthStar™

North Star Computers, Inc.
14440 Catalina St., San Leandro, CA 94577 USA
(415) 357-8500 TWX/Telex (910) 366-7001