OPERATION MANUAL

MICRO-PRODAC MONITOR SYSTEM

TABLE OF CONTENTS

PAGE

I. MICRO-PRODAC MONITOR SYSTEM INTRODUCTION 2

A. HARDWARE 2
B. SOFTWARE 2

II. MICRO-PRODAC MONITOR SYSTEM KEYBOARD OPERATION 2

A. ON-LINE AUTOMATIC MODE 2
B. ON-LINE MANUAL MODE 3
C. OFF-LINE MANUAL MODE 4

1. INTRODUCTION 4
2. DISPLAY BYTE OR DISPLAY WORD: "DB" OR "DW" 5
3. INPUT BYTE: "IB" 5
4. OUTPUT BYTE: "OB" 5
5. DISPLAY REGISTER: "DR" 5
6. MOVE MEMORY: "MV" 6
7. CHANGE INTERRUPT VECTOR TABLE: "CIV" 7
8. DISPLAY TEST: "DSP TST" 7
9. IMPORTANT NOTE 7
10. SINGLE STEP: "ST" 7
11. GO: "GO" 8

III. APPLICATION NOTES: OFF-LINE MANUAL MONITOR MODE 8,9

IV. APPENDIX 10

A. MICRO-PRODAC MONITOR KEYBOARD 10
B. ON-LINE AUTO KEYBOARD FUNCTIONS 10

1. KEYBOARD 10
2. COMMANDS 10
3. SUMMARY OF COMMAND SEQUENCE 11

C. ON-LINE MANUAL KEYBOARD FUNCTIONS 11

1. KEYBOARD 11
2. COMMANDS 11
3. SUMMARY OF COMMAND SEQUENCE 11,12

D. OFF-LINE MANUAL KEYBOARD FUNCTIONS 13

1. KEYBOARD 13
2. COMMANDS 13
3. SUMMARY OF COMMAND SEQUENCE 13,14

E. PMI (MONITOR INTERFACE) MEMORY MAP 14

F. MONITOR SYSTEM SCHEMATICS . . . . 15 to 20

G. PMI BOARD LAYOUT 21

H. MONITOR PANEL LAYOUT 22

I. MONITOR KEYBOARD & DISPLAY 22,23

EFFECTIVE DATE — APRIL 1982

I. INTRODUCTION

A. Hardware

The MICRO-PRODAC monitor system hardware consists of a plug-in PMI (Monitor Interface) board S#1993A28G01 located in the MICRO-PRODAC regulator cage and connected to a monitor keyboard/display panel S#8449A55G01 via two harnesses S#1999A51G02.

The heart of the system is an 8279 programmable keyboard/display interface chip located on the PMI board. This device serves to detect keypad entries and light appropriate seven-segment displays. Two digital-to-analog converters also located on the PMI are used for monitoring address contents. An 8251 programmable communication interface (USART) is available for future use with an RS232 serial link. Numerous interface chips provide communication between the PMI board and the Intel multibus backplane.

One small printed circuit board mounted on the monitor panel contains the 8 seven-segment displays and attendant decoder and driver chips. An associated daughter board contains the 32-key keypad and four amber mode LED's.

The monitor panel is available with dual keyboard/display boards (S#8449A55G02) for interfacing with two PMI boards in the split-cage MICRO-PRODAC regulator option.

B. Software

Software to operate the monitor system is linked in with drive software and loaded into the PROM's located on the regulator microcomputer board. The MICRO-PRODAC monitor system has the capability of operating in one of up to four modes: on-line automatic, on-line manual, off-line manual and off-line automatic. In the on-line modes, the system main program is executing and control algorithms are running. This is not the case for the off-line modes in which separate program modules are executing.

In the on-line automatic mode, the monitor display indicates either a predetermined drive variable (for example, speed) or an error code loaded in the drive's fault table.

The on-line manual mode allows access to RAM, PROM, Memory Mapped I/O and 8086 Ports while the drive is operating. Data can be changed if the correct data key is entered.

The off-line manual mode has capability similar to the on-line manual mode with the addition of PROM shadowing and selective program execution. In this mode, the main program does not loop; the 8086 waits for a command from the monitor keypad before taking action.

The off-line automatic mode is not presently used but is available for use similar to the on-line auto mode. Diagnostics would be run while the drive is not operating, with appropriate display indications.

II. MICRO-PRODAC MONITOR SYSTEM KEYBOARD OPERATION

A. On-Line Automatic Mode

The MICRO-PRODAC regulator powers up in the on-line automatic monitor mode with the corresponding amber LED on the monitor panel turned on. Should the monitor be in any other mode, it can be changed to on-line automatic by pressing the appropriate mode key.

If no faults are noted in the on-line automatic mode, the left-most four character address field indicates "SPd-." The right-most four characters in the data field are an indication of the machine's speed.

If the regulator detects any system fault conditions, non-zero error codes are loaded into a fault table in the order in which they occur. The address field changes to display "-Err"; the data field displays the two-character byte code for the first fault. This byte is one of 40 possible fault indications that can be displayed. The "+IFLT" (Increment Fault Display and "-DFLT" (Decrement Fault Display) buttons on the keyboard enable the user to examine any of the codes that have been catalogued. Note that once the last byte in the fault table has been displayed pressing "+IFLT" has no effect. Similarly, if the first fault is being displayed, pressing "-DFLT" has no effect.

There are two additional controls in this mode. One, "FDR" (Fault Display Reset), is used to return the fault display to the first fault. The second control, "FTR" (Fault Table Reset), clears the fault table to all zeros, returns the fault pointer to the first fault, and restores the display to a "SPd-" indication.

B. On-Line Manual Mode

Should the monitor be in any other mode, it can be changed to on-line manual by pressing the appropriate mode key. The corresponding amber LED on the monitor panel will turn on. Note that the term "On-Line" indicates that the microprocessor is executing through its main program loop; power may or may not be applied to the motor. The display in the on-line manual mode indicates the selected address in the left-most four-character address field and the associated data contents in the right-most data field. Some keys in this mode operate as either commands or hexadecimal characters; therefore, it is important to key in the proper sequence of buttons in order to achieve the desired effect.

To display the two-character byte contents of an address, press the "DB" (Display Byte) button followed by the four hexadecimal characters of the desired address. When the desired address is displayed, press the "NX" (Next) button and the associated data will appear, updated once every main program cycle. The "NX" button serves to terminate a command sequence and prepare the monitor for the next command (such as "SD" for setting new data).

Four-character word data is displayed in a similar fashion. The "DW" (Display Word) key is used in place of the "DB" key.

Once the byte or word contents of an address have been displayed, the "NX" button may be used to increment the address by one and display its contents (either byte or word, depending upon whether the "DB" or "DW" command was initially used). The "DA" (Decrement Address) operates in similar fashion.

If the HEX/PCT switch on the monitor panel is switched to PCT, data is displayed as a percentage of full scale based on two's complement notation. For example, the positive full scale hexadecimal byte is 7FH; in percent, this is 99.9. The negative full scale hex byte is 80H; in percent, this is -99.9. In the word mode, 7FFFH would be displayed as 99.9 percent and 8000H would be displayed as -99.9 percent.

To sample rapidly changing data at a displayed address, simply press "DB" or "DW." The data field freezes until "NX" is pressed.

The "EX" (clear) button may be used to erase a series of address characters that have been entered and revert to the previously-displayed address and data. The monitor is prepared for the next command.

Note that displaying data contained in a four-character address covers the MICRO-PRODAC RAM and Memory-Mapped I/O address space. In order to display data contained at PROM addresses higher in memory (for example, FA000H to the top of memory FFFFFH), a four-character segment base and four-character offset must be used. Press "DB" or "DW," followed by the four-character segment base, followed by "SEG" (Segment Base), followed by the four-character offset address, followed by "NX." For example, to display the byte contents of address FA100H, press "DB," "F," "A," "0," "0," "SEG," "0," "1," "0," "0," "NX." This causes absolute address FA000H to be the base address until "SEG" is used again and takes 0100H as the current offset from the base address. Any number of bases and offsets may actually be used to access one address. Note that if "SEG" is not used, the default base address is 00000H.

There are three operations that affect the data. They are "ID" (Increment Data By One), "DD" (Decrement Data By One), and "SD" (Set New Data). As a safeguard, following power-up no changes in data can be made until the correct software key is entered. The key is used much as "DB" and "DW." First press the "DK" (Data Key) button, followed by the four characters of the key, followed by the "NX" button. If the correct characters are used (1212H), the "ID," "DD," and "SD" data-changing keys become effective. If any but the correct key characters are used, the software key is reset and data cannot be changed.

To change the data (either byte or word) at a displayed address, press "SD" (Set Data), followed by the desired 2 or 4 characters of data, followed by "NX." For example, to place 7FH in address 0700H, press "DB," followed by "0," "7," "0," "0," "NX," to display the current contents of 0700H, followed by "SD," "7," "F," "NX," to update the data. All data changes must be made in hexadecimal, not percent.

Data at a displayed address may be incremented by one using the "ID" key and decremented by one using the "DD" key. "NX" need not be pressed following "ID" or "DD." Note that data wraparound occurs; in the byte mode, if an address currently contains FFH and "ID" is pressed the data changes to 00H.

The word contents of a RAM or Memory-Mapped I/O address may be converted to an analog signal and monitored at one of the two analog outputs on the monitor panel. Two's complement notation is used: 7FFFH becomes +10.0 volts, 8000H becomes -10.0 volts and 0000H becomes 0.0 volts. To implement this feature, press "SAO" (Set Analog Output), followed by "1" or "2" to select one of the two analog output channels, followed by the four-character address whose contents are to be monitored, followed by "NX" to complete the operation. The data at the specified address is used to refresh the analog outputs each time through the main program loop.

The final controls are for exercising any of the 8086 microprocessor I/O ports. To display the input data to a specified port number the "IB" (Input Byte) function may be used. Press "IB," followed by the four-character port number, followed by "NX." The input to the port is displayed in the two right-most characters of the data field. It may be converted to percent using the HEX/PCT switch on the monitor panel if desired.

To output a byte to a specified port, the correct data key must first be entered. Then press "OB" (Output Byte), followed by the four-character port number, followed by "NX," followed by the two characters of desired output data, followed by "NX." The display reverts back to the "DB" command mode following this sequence of keys.

As a note of caution, only access addresses or ports that are physically realized in hardware. Otherwise, 8086 time-out occurs, which may cause problems in main program execution.

C. Off-Line Manual Mode

1. Introduction

A depression of the off-line manual mode key is detected by the on-line monitor procedure and an appropriate software flag is set. It is then up to the main program to examine this flag versus any other conditions of interest before a jump to the off-line manual routine is allowed. If the jump is not allowed, the software flag is reset. As an example, a jump to the off-line manual mode may not be allowed unless the proper mode key is pressed AND power is removed from the motor.

While off line, a depression of an on-line mode key effects an unconditional jump back into the main program loop execution. The "Go" and "Single Step" commands allow selective execution of portions or all of the main program code.

When the monitor has jumped to the off-line manual mode, the corresponding amber LED on the monitor panel turns on and the display indicates "U-PRODAC." The microprocessor is no longer executing through its main program; it is simply waiting for a command input from the keypad.

The display in this mode, as in the on-line manual mode, indicates selected addresses in the left-most four-character address field and the associated data contents in the right-most data field. Some keys in this mode operate as either commands or hexadecimal characters; therefore, it is important to key in the proper sequence of buttons in order to achieve the desired effect.

Note that the off-line monitor is ready to accept a command if the display indicates either "U-PRODAC," "-" (the Command Prompt), or "-Err" (Keyboard Mistake).

2. Display Byte or Display Word: "DB" or "DW"

Addresses entered in the "DB" or "DW" command sequences may take one of two forms. One, a four-character segment base and four-character offset may be specified, separated by the "SEG" key. The segment base applies only for the one "DB" or "DW" sequence. Two, just a four-character offset may be specified. In this case, the current contents of the "CS" register array location acts as the segment base default. "CS" is set to 0000 upon entry into the off-line manual mode but could change depending upon which operations are performed. It can be displayed using the "DR" (Display Register) command discussed later.

To display the two-character byte contents or four-character word contents of an address, press the "DB" (Display Byte) or "DW" (Display Word) button, respectively. A period prompt appears in the address field to indicate that the hexadecimal characters to follow will be displayed in the address field. Then key in the address; upon completion only the offset address will be visible in the address field. Pressing "NX" causes the associated data to appear in the data field.

The period prompt has now shifted to the data field to indicate that the address displayed is now open for optional changing of data; any keyed-in hexadecimal characters to follow will be displayed in the data field. Following this optional key-in of new data, pressing the "NX" key enters the data in memory, increments the address field to the next offset address (+1 for "DB," +2 for "DW"), and displays the corresponding data in the data field.

The "DB" or "DW" command sequence is terminated by pressing the "EX" key. This returns a command prompt "-" to the display.

3. Input Byte: "IB"

The "IB" (Input Byte) command allows access to the 8086 I/O Mapped Ports. Up to 64K (OFFFFH) ports may be examined; only a four-character port number is required.

To display the input to a specific port number, press "IB." A period prompt appears in the address field to indicate that the hexadecimal characters to follow will be displayed in the address field. Then key in the port number. Pressing "NX" causes the associated data to appear in the data field. Further presses of "NX" cause the displayed port number to be resampled and the associated data to be displayed in the data field. Pressing "EX" terminates the "IB" command sequence and restores a command prompt "-" to the display.

4. Output Byte: "OB"

The "OB" (Output Byte) command allows changing of the output to the 64K (OFFFFH) of 8086 I/O Mapped Ports. Only a four-character port number is required.

To output a byte to a specified port number, press "OB." A period prompt appears in the address field to indicate that the hexadecimal characters to follow will be displayed in the address field. Then key in the port number, followed by an "NX." The period prompt now appears in the data field to indicate that the hex characters (2) to follow will be displayed in the data field. Another press of "NX" outputs the data to the port. Data to the same displayed port can be retransmitted by further data entry (optional) and "NX" depressions. Pressing "EX" terminates the "OB" command sequence and restores a command prompt "-" to the display.

5. Display Register: "DR"

The "DR" (Display Register) command allows access to a 14-word array representing the 14-8086 machine registers. This array is initialized upon entry into the off-line manual mode. It is important to note that the contents of this array mirror the actual machine register states only upon return to the off-line manual monitor from a "ST" (Single Step) or "GO"-with-breakpoint command execution. These two conditions activate interrupt procedures which push the machine states onto the stack and then pop them into the respective array locations.

Similarly, the machine register states mirror the array contents only when the "ST" or "GO" commands are initiated. In this case, the contents of the array are pushed onto the stack and then popped into the appropriate machine registers before the execution of main program code begins.

The array contents which may be displayed and optionally altered are "AX," "BX," "CX," "DX," "SP," "BP," "SI," "DI," "CS," "DS," "SS," "ES," "IP" and "FL," in that order. Note that they will be referred to within quotation marks to indicate that they are not necessarily the corresponding 8086 machine register states. Also, these names appear on the same keypad keys as the hex characters 0 through D.

To display the contents of this register array, press "DR." A period prompt appears in the address field. In this case, it indicates that the next button pressed will be interpreted as representing a member of the register array from "AX" (button 0) to "FL" (button D). Press the register button desired. A register abbreviation appears in the address field, the corresponding array word contents appears in the data field, and a period prompt also appears in the data field to indicate that the register array displayed is open for optional data change. Any keyed-in hexadecimal characters to follow will be displayed in the data field. Following this optional key-in of new data, pressing the "NX" key enters the data in memory, increments the address field to the next register abbreviation in the series, and displays the corresponding data in the data field.

The "DR" command sequence may be terminated in one of two ways. Pressing "EX" causes the command prompt "-" to return to the display. Alternatively, pressing "NX" following a display and optional change of data in "FL" also causes a return of the command prompt.

6. Move Memory: "MV"

The "MV" (Move Memory) command allows blocks of memory (either PROM or RAM) to be moved from one location to another location in RAM. This is particularly useful for PROM shadowing, i.e. copying the main program into RAM and executing out of RAM.

Press "MV." Three period prompts appear in the address field, indicating that three address inputs are required. As each address is entered, one period is removed.

Key in the starting address of the block of data to be moved. As with "DB" or "DW," this address may be of a (segment base):(offset) format or simply an (offset) format, in which case the default segment base is the current contents of the "CS" register array location. Press "NX" when the desired offset address is displayed in the address field, entering the starting address information.

Now key in the ending offset address of the block of data to be moved. Only 64K of memory may be moved at a time, so the ending address uses the same segment base as the starting address. Again press "NX" to enter this information.

Finally, key in the destination address to which the block of memory (beginning with the starting address) is to be moved. This address format may again take either the (segment base):(offset) or (offset) with "CS" default form.

Press the "EX" button to enter the destination address and initiate the memory move. Upon completion, the command prompt "-" is again displayed.

Note that the memory move is carried out one byte at a time. Accordingly, the "MV" command may be used to fill a block of memory with a constant. Specify a destination address one byte above the starting address (which contains the constant in question). The ending offset should then be specified one byte below the last memory address to be filled.

7. Change Interrupt Vector Table: "CIV"

The "CIV" (Change Interrupt Vector Table) command is used in conjunction with PROM shadowing execution. Pressing "CIV" causes the PROM shadowing code segment (which must be predetermined) to be overwritten into the segment base address locations in the interrupt vector table; the instruction pointer address locations are not affected. Thus, after "CIV" is pressed, words at absolute address locations 2H, 6H, AH, EH, ..., 3FAH, 3FEH have been updated. The effect is to cause interrupts occurring during PROM shadow execution to be serviced by interrupt procedures within the executing RAM block. This assumes that the instruction pointer to these interrupt procedures has not changed. Note that pressing "CIV" has no effect on the display; the command prompt "-" should not change.

8. Display Test: "DSP TST"

Pressing "DSP TST" causes all eight seven-segment displays to indicate "8" as an operational test. Again pressing "DSP TST" blanks the 8's and returns the command prompt "-" to the display.

9. Important Note

Before attempting to make use of the "ST" and "GO" commands, please read the application notes in Section III as well as the following sections concerning "ST" and "GO." This will acquaint the user with some of the subtleties of operation.

10. Single Step: "ST"

The "ST" (Single Step) command is used to execute one main program machine language instruction at a time, whether it is located in PROM or RAM.

Before using the "ST" command, one must use the "DR" (Display Register) command to load the correct machine register contents into the register array. This array is pushed onto the stack and popped into the corresponding machine registers just prior to the return to the main program.

Following the machine language instruction execution, a software interrupt 1 (single-step, trap flag set) occurs. The machine register contents are pushed onto the stack, popped into the register array accessed by "DR," and control returns to the off-line manual monitor. The registers may then be viewed and another single step executed if desired.

There are two restrictions on single-stepping. First, if an interrupt occurs during the execution of a single step, the "CS" (Code Segment) and "IP" (Instruction Pointer) register array locations will contain the address of the referenced interrupt procedure upon return from the single step. Further stepping executes this interrupt procedure. If it is a type 3 interrupt (breakpoint interrupt), further stepping eventually returns control to the off-line monitor since a type 3 software interrupt is used by the "GO"-with-breakpoint command.

Second, a series of instructions changing the contents of SS (Stack Segment) and SP (Stack Pointer) cannot be single-stepped.

To execute a single step, press "ST." The address field now displays an offset address with respect to the current contents of the "CS" register array location; the data field displays the contents of this address, representing a machine language instruction. If it is desired to specify a different address from which to begin single-stepping, key in the address. The (Segment Base):(Offset) or (Offset) with "CS" default address format may be used. Be sure to specify the correct "CS," i.e., a segment base representing the starting address of all executable code. This causes the register array contents to be updated. The address field now displays the revised offset; the data field is blanked.

A press of "NX" following the optional change in the single-step start address causes the main program machine language instruction at the displayed offset address to execute. The address field now contains the offset of the next machine language instruction. Further presses of "NX" continue the single-stepping.

To terminate single-stepping and allow other commands to be used, press "EX." The command prompt "-" is again displayed. The "DR" command may now be used to examine/update machine register contents, if desired.

11. Go: "GO"

The "GO" command is used to execute the entire main program, whether it is located in PROM or RAM. Optionally, if the main program has been loaded into RAM for PROM shadowing, a main program breakpoint may be specified, returning control to the off-line manual monitor when the breakpointed instruction is reached. In placing this breakpoint, the user must be careful to select an address containing an actual machine instruction, not an address reference which could foul program execution.

As with single-stepping, before using the "GO" command, one must use the "DR" (Display Register) command to load the correct machine register contents into the register array. This array is pushed onto the stack and popped into the corresponding machine registers just prior to the return to the main program.

If the optional breakpoint is used, a machine instruction OCCH is located at the address specified (the actual instruction at that address is saved for future restoration). When this instruction is executed, a software interrupt 3 (breakpoint) occurs. The machine register contents are pushed onto the stack, popped into the register array accessed by "DR," and control returns to the off-line manual monitor, with a command prompt "-" and breakpoint indication "br" displayed. The registers may then be viewed and another "GO" executed if desired. The original instruction at the breakpoint has been restored; thus, each time the "GO"-with-breakpoint command is used, a breakpoint address must be specified.

To use the "GO" command, press "GO." The address field now displays an offset address with respect to the current contents of the "CS" register array location; the data field displays the contents of this address, representing a machine language instruction. If it is desired to specify a different address from which to begin execution, key in the address. The (Segment Base):(Offset) or (Offset) with "CS" default address format may be used. Be sure to specify the correct "CS," i.e., a segment base representing the starting address of all executable code. This causes the register array contents to be updated. The address field now displays the revised offset; the data field is blanked.

A press of "EX" following the optional change in the execution start address causes an "E" to be displayed in the address field just prior to the transfer of control to the main program. The main program should continue to execute (either out of PROM or RAM) indefinitely since no breakpoint has been specified.

To specify a breakpoint, the "NX" button would have been pressed instead of the "EX" button. Then the offset of the desired breakpoint address with respect to the starting address segment base would be specified, followed by a press of the "EX" button. The main program should continue to execute until the breakpoint is reached, at which point control returns to the off-line manual monitor, as discussed. Note that since the breakpoint option requires the writing of OCCH at the specified address, it can only be used if the main program is executed out of RAM.

III. APPLICATION NOTES: OFF-LINE MANUAL MONITOR MODE

Before attempting to use the "GO" or "ST" commands, the user should be somewhat familiar with main program structure, assembly language, and machine code. Also, the standard MICRO-PRODAC system contains 24K of PROM and 4K of RAM; PROM shadowing of the PROM-based code into RAM would therefore require an additional RAM-Memory board.

Note that in order to be able to access and change constants for PROM shadowing execution, all main program modules must have been compiled with the "ROM" option. This allows constants to be accessed via the code segment instead of the data segment (which should not be changed in the off-line mode), providing the capability to change constants in RAM.

The following application tips are provided to aid the user in successful operation of the "GO" and "ST" functions:

(1) PLM86 causes a prologue to be inserted before the first executable statement in the main program. This prologue initializes 8086 registers. It is shown below for CS=FC40H, SS=DS=FC00H, SP=4500H:

```
             ADDRESS    MACHINE CODE         ASSEMBLY LANGUAGE
             FC40:00    00                   -
                  01    FC                   -
FIRST             02    FA                   CLI
STATEMENT         03    2E 8E 16 00 00       MOV  SS, CS:@@STACK$FRAME
TO                08    BC 00 45             MOV  SP, @@STACK$OFFSET
EXECUTE           0B    8B EC                MOV  BP, SP
                  0D    16                   PUSH SS
                  0E    1F                   POP  DS
                  0F    FB                   STI
```

@@STACK$FRAME refers to the 00 00 at FC40:06 and 07. CS:@@STACK$FRAME thus refers to the contents of FC40:00 and 01. @@STACK$OFFSET refers to the 00 45 at FC40:09 and 0A. This register initialization is determined during the "LOCATE" step of program development and affects SS, SP, BP and DS The CS is initialized by the power-up "BOOTSTRAP" located at FFFF:0 in memory. The user may want to avoid these initializations and specify his own register contents. However, DS and SS should NEVER be changed from the initialization specified in the prologue.

(2) To insure proper "ST" and "GO" command operation, register array locations "SP" and "BP" must be specified away from (preferably, below) the addresses to which the main program prologue normally initializes these registers. For example, if BP and SP are normally initialized to 4500H, specify "BP" and "SP" as, say, 44E0H. Otherwise, the "ST" and "GO" commands cause register array data placed on the stack to be overwritten before it can be popped into the appropriate machine registers. The off-line manual monitor requires at least 7H addresses below the initialized SP. Therefore, for use with "GO" and "ST," "SP" and "BP" should be specified at least 7H addresses lower.

(3) In PLM86 public labels or labels that are jumped to from within procedures located in a main module are given a prologue at the label which resets the stack and base pointers (SP and BP) to their original, initialized values. This is shown below:

```
LABEL: MOV  SP, @@STACK$OFFSET
       MOV  BP, SP
```

@@STACK$OFFSET is immediate data located within the code and mirroring the initialized prologue data.

Thus, if such a label is executed, the revised "USER" stack pointer and base pointer discussed in point #2 will be lost. (This should not matter if the "GO"-without-breakpoint is being used; proper execution would continue.) Try to avoid executing such label prologues.

A prime example of such a label in the main program is "BUFFLP," the public label to which the off-line monitor normally returns to the on-line mode when the "ON-LINE AUTO" or "ON-LINE MANUAL" mode key is pressed.

Also, "OFFLN" is the public label to which program execution jumps when an off-line manual monitor request is honored. It marks the beginning of a separate main module of off-line manual monitor code and causes BP and SP to be reinitialized to the same values as the system main program prologue discussed in point #1.

(4) Main programs normally include a RAM-initialization section before the main loop is reached. If the "ST" or "GO"-with-breakpoint command causes execution through this sectio of code, some RAM variables used in the associated interrupt procedures 1 and 3 are zeroed. This is undesirable, because it causes the SP and SS machine registers to be zeroed during the interrupt procedure execution. Try to avoid executing these RAM-initialization sections or else make sure the main program does not initialize off-line monitor RAM variables.

(5) Do not initiate the "ST" or "GO" commands from a point after a main program procedure call and before the procedure return. This causes the stack to contain erroneous information for the procedure return.

IV. APPENDIX

A. MICRO-PRODAC Monitor Keyboard

OFF LINE MAN	+ IFLT	- DFLT	CIV	C DA/IP	D FL	E DD	F ID
OFF LINE AUTO	SEG	DSP TST	SAO	8 CS	9 DK/DS	A SD/SS	B ES
ON LINE MAN	NX	EX		4 IB/SP	5 OB/BP	6 MV/SI	7 DW/DI
ON LINE AUTO	FTR	FDR		0 DB/AX	1 DR/BX	2 GO/CX	3 ST/DX

B. On-Line Auto Keyboard Functions: Keyboard

1.

OFF LINE MAN	+ IFLT	- DFLT					
OFF LINE AUTO							
ON LINE MAN							
ON LINE AUTO	FTR	FDR					

2. On-Line Automatic Keyboard Functions: Commands

(1) +IFLT : Increment to next byte in fault table
(2) -DFLT : Decrement to previous byte in fault table
(3) FDR : Fault display reset to first byte in fault table
(4) FTR : Fault table reset to all zeros; "SPd-" variable is displayed if the first fault table location contains zero.

3. On-Line Automatic Keyboard Functions: Summary of Command Sequence†

Increment the fault table display by one: [(+IFLT)]

Decrement the fault table display by one: [(-DFLT)]

Reset the fault display to the first fault: [(FDR)]

Clear the fault table to all zeros: [(FTR)]

C. On-Line Manual Keyboard Functions: Keyboard

1.

OFF LINE MAN				C DA/	D	E DD	F ID
OFF LINE AUTO	SEG		SAO	8	9 DK/	A SD/	B
ON LINE MAN	NX	EX		4 IB/	5 OB/	6	7 DW/
ON LINE AUTO				0 DB/	1	2	3

2. On-Line Manual Keyboard Functions: Commands

(1) DK : DATA KEY; enter key in order to change address contents
(2) DB : DISPLAY BYTE; two-character address contents displayed
(3) DW : DISPLAY WORD; four-character address contents displayed
(4) DA : DECREMENT ADDRESS by one and display contents
(5) EX : Clear display to previously displayed address/data
(6) NX : Complete instruction and prepare for next command; also, increment address by one and display data
(7) DD : Decrement Data by one
(8) ID : Increment Data by one
(9) SD : SET DATA; used to change byte or word data
(10) SEG : Segment base can be changed from the 0000 default
(11) SAO : SET ANALOG OUTPUT to convert address contents from a hexadecimal number to an analog voltage using two's complement notation and ±10.0 volts full scale
(12) IB : INPUT BYTE from I/O mapped port displayed
(13) OB : OUTPUT BYTE to I/O mapped port
(14) HEX/PCT SWITCH: Display address contents in either hexadecimal or percent; 7FFFH corresponds to positive FULL SCALE OR +99.9 pct and 8000H corresponds to negative full scale or -99.9 pct.

3. On-Line Manual Keyboard Functions: Summary of Command Sequence†

Display Address Byte Contents:
[(DB)(4-Character Offset Address)(NX)]

Display Address Word Contents:
[(DW)(4-Character Offset Address)(NX)]

† See bottom of page 13 for notation key.

Change Segment Base and Display Address Byte Contents:
[(DB)(4-Character Segment Base)(SEG)
(4-Character Offset Address)(NX)]

Change Segment Base and Display Address Word Contents:
[(DW)(4-Character Segment Base)(SEG)
(4-Character Offset Address)(NX)]

Change Data at a Displayed Address:
[(SD)(New Data - 2 or 4 Characters)(NX)]

Increment Address by One:
[(NX)]

Decrement Address by One:
[(DA)]

Enter the Data Key:
[(DK)(4-Character Key)(NX)]

Increment Data by One:
[(ID)]

Decrement Data by One:
[(DD)]

Clear Display to the Previously-Displayed Address:
[(EX)]

Hold Currently-Sampled Data at a Displayed Address:
[(DB)]* or [(DW)]*

Set Analog Output:
[(SAO)(1 or 2 Channel)
(4-Character Absolute Address)(NX)]

Input a Byte From an I/O Mapped Port:
[(IB)(4-Character Port Number)(NX)]

Output a Byte to an I/O Mapped Port:
[(OB)(4-Character Port Number)(NX)
(2-Character Port Data)(NX)]

HEX/PCT Switch:
HEX - Displays address or port contents in hexadecimal
PCT - Displays address or port contents as a percentage of two's complement full scale

* Punch [(NX)] to continue sampling or go to next command.

D. Off-Line Manual Keyboard Functions: Keyboard

1.

OFF LINE MAN			CIV	C /IP	D FL	E	F
OFF LINE AUTO	SEG	DSP TST		8 CS	9 /DS	A /SS	B ES
ON LINE MAN	NX	EX		4 IB/SP	5 OB/BP	6 MV/SI	7 DW/DI
ON LINE AUTO				0 DB/AX	1 DR/BX	2 GO/CX	3 ST/DX

2. Off-Line Manual Keyboard Functions: Commands

(1) DB : DISPLAY BYTE; two-character address contents displayed
(2) DW : DISPLAY WORD; four-character address contents displayed
(3) NX : COMPLETE INSTRUCTION; opens address or register for optional change; also, increment address by one
(4) EX : Terminate a sequence of commands
(5) SEG : Segment base can be changed from the "CS" register default
(6) IB : Input byte from I/O mapped port displayed
(7) OB : Output byte to I/O mapped port
(8) DR : Display "REGISTER" contents (AX, BX, CX, DX, SP, BP, SI, DI, CS, DS, SS, ES, IP, FL); actual registers are only updated in the "GO" or "SINGLE STEP" function; the array displayed by "DR" mirrors the actual registers following an interrupt 1 (single step) or interrupt 3 (breakpoint of "GO")
(9) GO : Execute code from a specified address until an optional breakpoint (PROM shadowing only) is reached
(10) ST : Single step through code from a specified point
(11) MV : Move memory between the addresses specified to a third specified address; used for PROM shadowing

(12) CIV : Change interrupt vector table for PROM shadowing operation
(13) DSP TST : DISPLAY TEST; all displays read "8"

3. Off-Line Manual Keyboard Functions: Summary of Command Sequence

All "addresses" may be either a (Segment Base):(Offset) format specification in which the (SEG) key operates as a colon or an (Offset) format specification in which the current "CS" register contents acts as the default segment base.

In the following information, the format is as follows:

[ ]	Encloses a command string
( )	Encloses keyboard entries
(_)	Indicates a keyboard key
< >	Encloses optional keyboard entries
< >*	Encloses multiple optional keyboard entries

Display/Modify Address Byte Contents:
[(DB)(Address)(NX)<<(New Data)>(NX)>*(EX)]

Display/Modify Address Word Contents:
[(DW)(Address)(NX)<<(New Data)>(NX)>*(EX)]

Input a Byte From an I/O Mapped Port:
[(IB)(Port Number)(NX)<(NX)>*(EX)]

Output a Byte to an I/O Mapped Port:
[(OB)(Port Number)(NX)(Data)<(NX)(Data)>*(EX)]

Display/Modify 8086 Register Contents:
[(DR)(Reg. Key)<<(New Data)>(NX)>*<(EX)>]

Move Data for PROM Shadowing:
[(MV)(Start Address)(NX)(End Offset Address)(NX)
(Destination Address)(EX)]

Execute Through Program One Instruction at a Time:
[(ST)<(Start Address)>(NX)
<<(Start Address)>(NX)>*(EX)]

Execute Through Program Until Optional Breakpoint is Reached:
[(GO)<(Start Address)><(NX)(Breakpoint Offset)>(EX)]

Change Interrupt Vector Table for PROM Shadow Execution:
[(CIV)]

Test The Display by Showing All 8's:
[(DSP TST)(DSP TST)]

E. PMI (Monitor Interface) Memory Map

ADDRESS	TYPE	NAME	FUNCTION
9000_H	Word	PMI$BOARD$ID	Contains board ID of 0090 or 0190 for hex/pct switch
9002_H	Word	PMI$DAC1	Digital-to-analog converter output
9004_H	Word	PMI$DAC2	Digital-to-analog converter output
9006_H	Byte	PMI$LEDS	Writing a "0" to the following bits turns on the noted LED: Bit 0: OFF LINE MANUAL Bit 1: OFF LINE AUTO Bit 2: ON LINE MANUAL Bit 3: ON LINE AUTO Bit 7: I'M OK ON PMI
9008_H	Byte	USART$DATA$PORT	RS232 serial data link using 8251 USART - data port

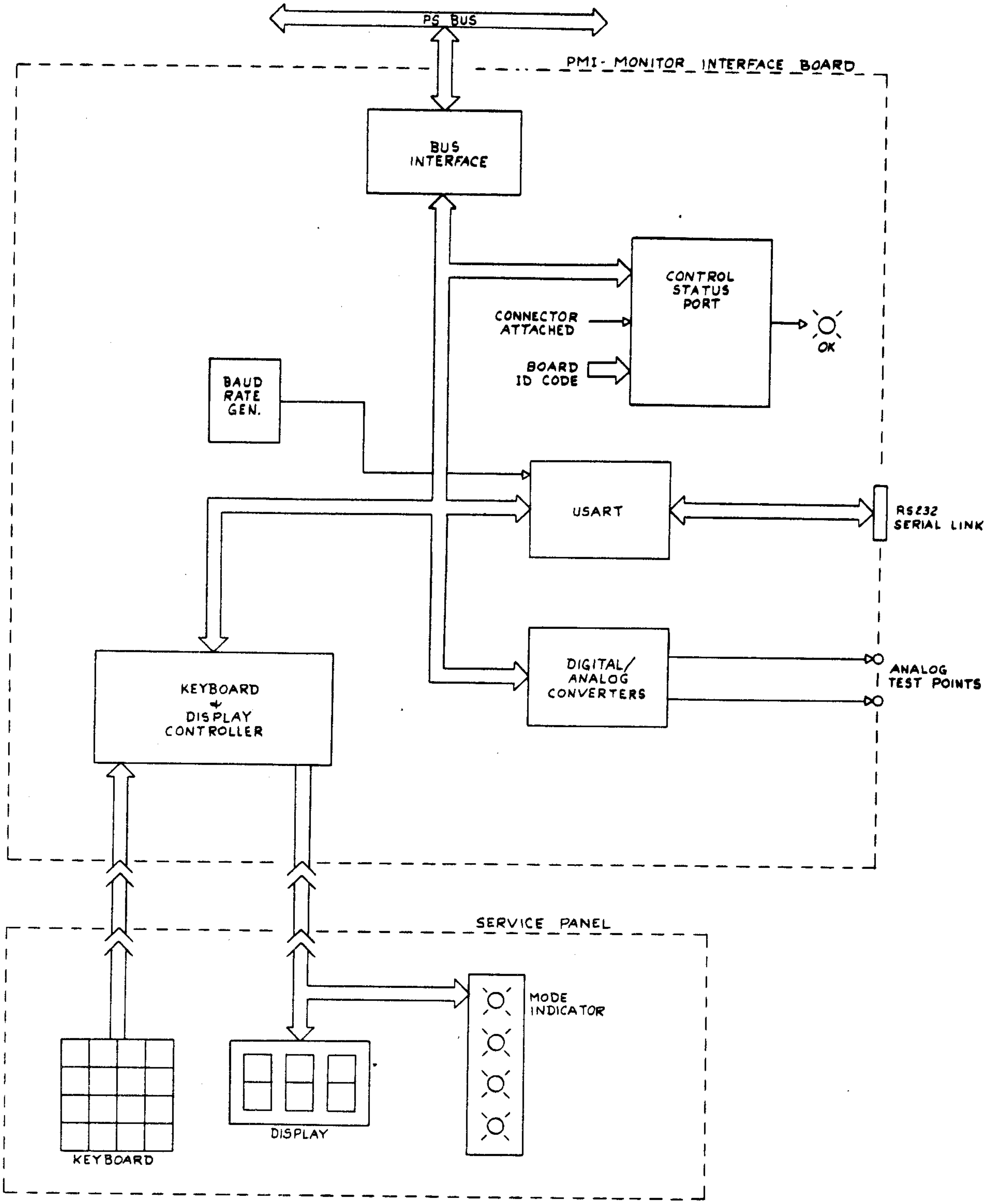
$900A_H$	Byte	USART$STAT$PORT	RS232 serial data link using 8251 USART - status port - initialize in software with the following PLM86 sequence:
			USART$STAT$PORT = 65H;
			USART$STAT$PORT = 0CFH;
			USART$STAT$PORT = 25H;*

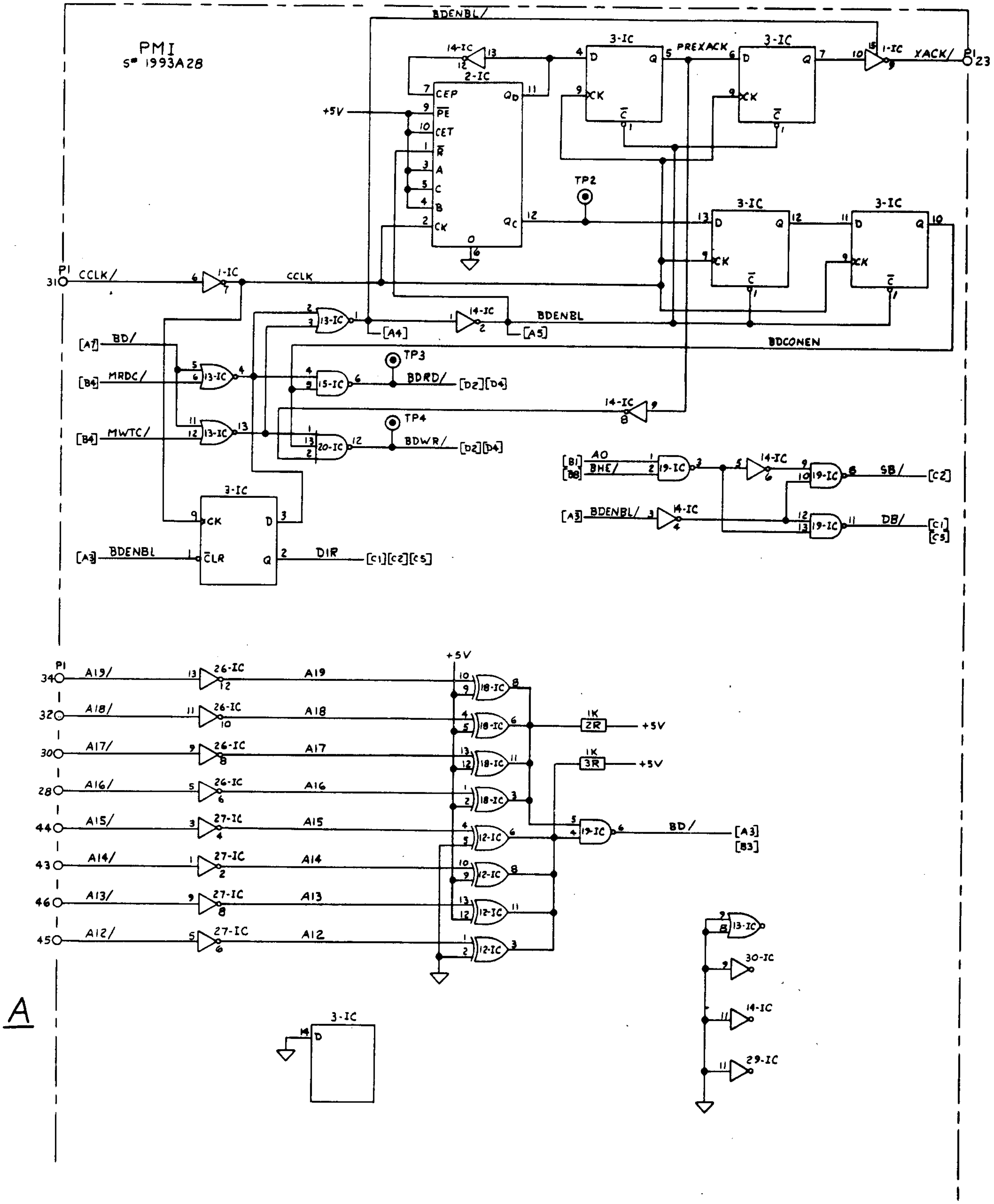
$900C_H$	Byte	KB$DATA$PORT	8279 keyboard/display controller data port

$900E_H$	Byte	KB$STAT$PORT	8279 keyboard/display controller status port - initialize in software with the following PLM86 sequence:
			KB$STAT$PORT = 00H;
			KB$STAT$PORT = 39H;*

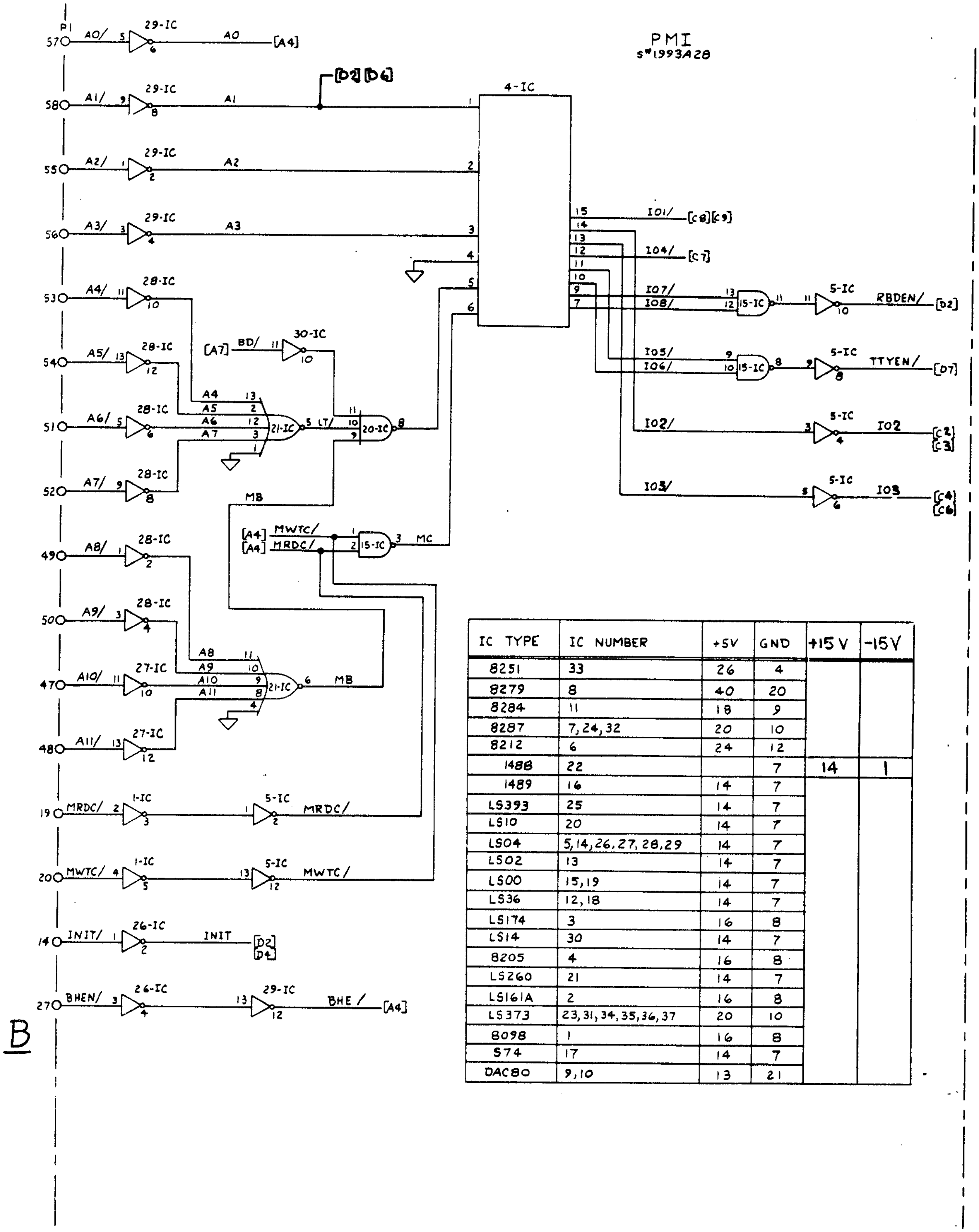
*See MONITOR$INIT procedure in MONINI.SRC file

F. Monitor System Schematics

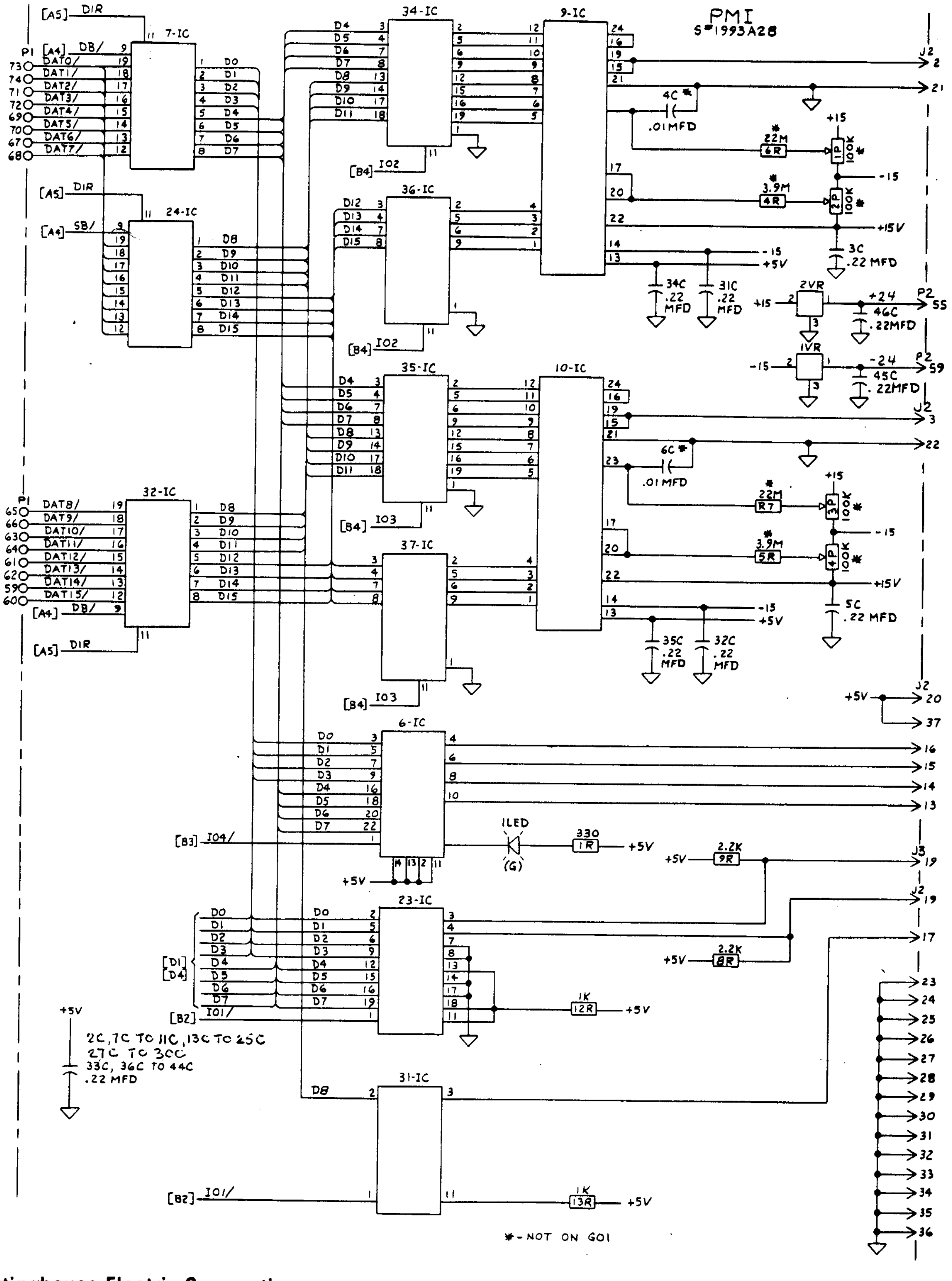



PMI
S# 1993A28
BDENBL/
14-IC
2-IC
3-IC
PREXACK
1-IC
XACK/
P1
23
+5V
CEP
PE
CET
R
A
C
B
CK
QD
QC
D
Q
C
TP2
P1
31
CCLK/
CCLK
13-IC
[A4]
[A5]
BDENBL
BDCONEN
[A1]
BD/
[B4]
MRDC/
15-IC
TP3
BDRD/
[D2][D4]
MWTC/
TP4
20-IC
BDWR/
[D2][D4]
[B1]
A0
[B8]
BHE/
19-IC
SB/
[C2]
[A3]
BDENBL/
DB/
[C1]
[C5]
CLR
[A3]
BDENBL
DIR
[C1][C2][C5]
P1
34
A19/
26-IC
A19
+5V
18-IC
32
A18/
A18
1K
2R
+5V
30
A17/
A17
1K
3R
+5V
28
A16/
A16
44
A15/
27-IC
A15
12-IC
17-IC
BD/
[A3]
[B3]
43
A14/
A14
46
A13/
A13
45
A12/
A12
30-IC
29-IC
A

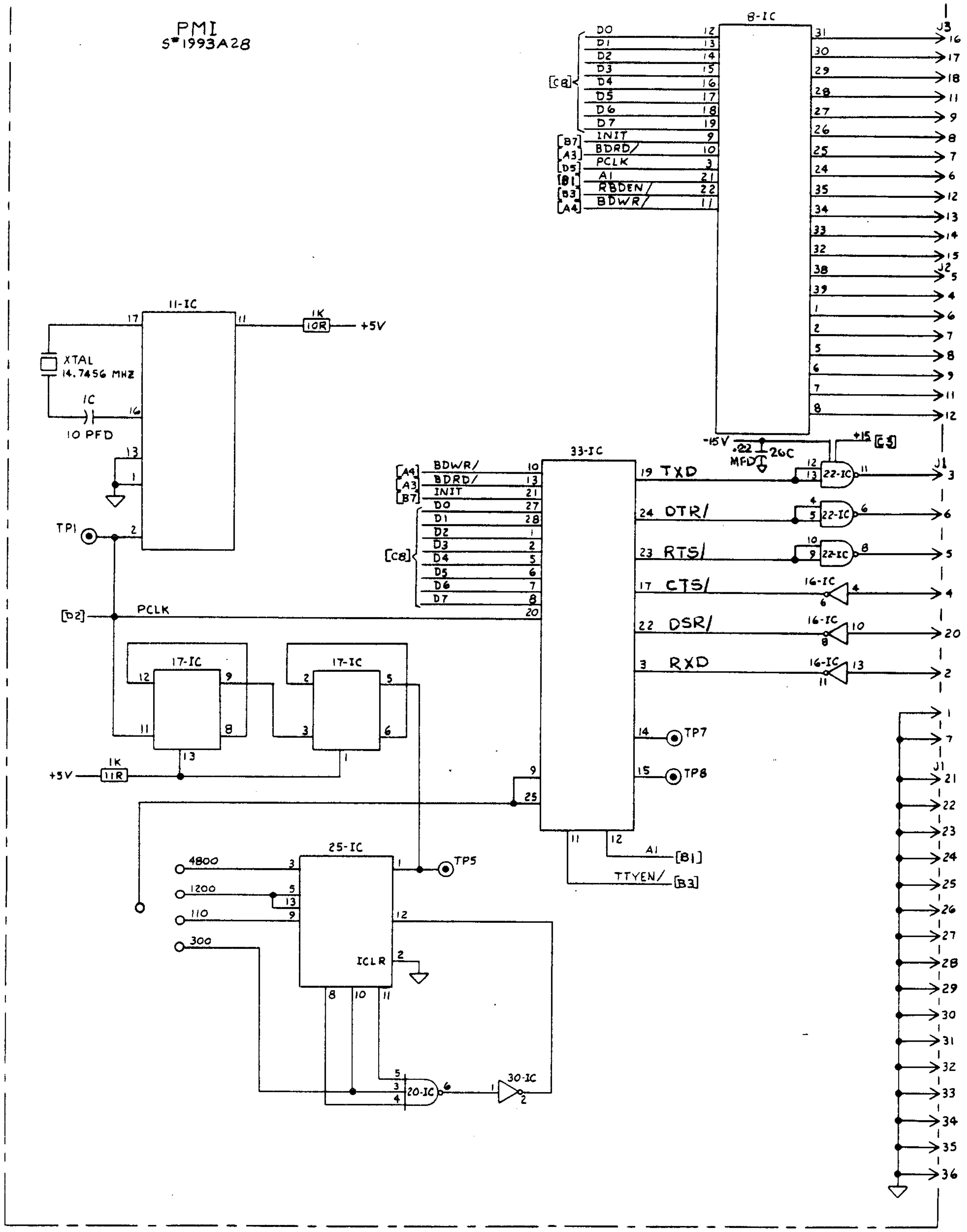
PMI
S#1993A28



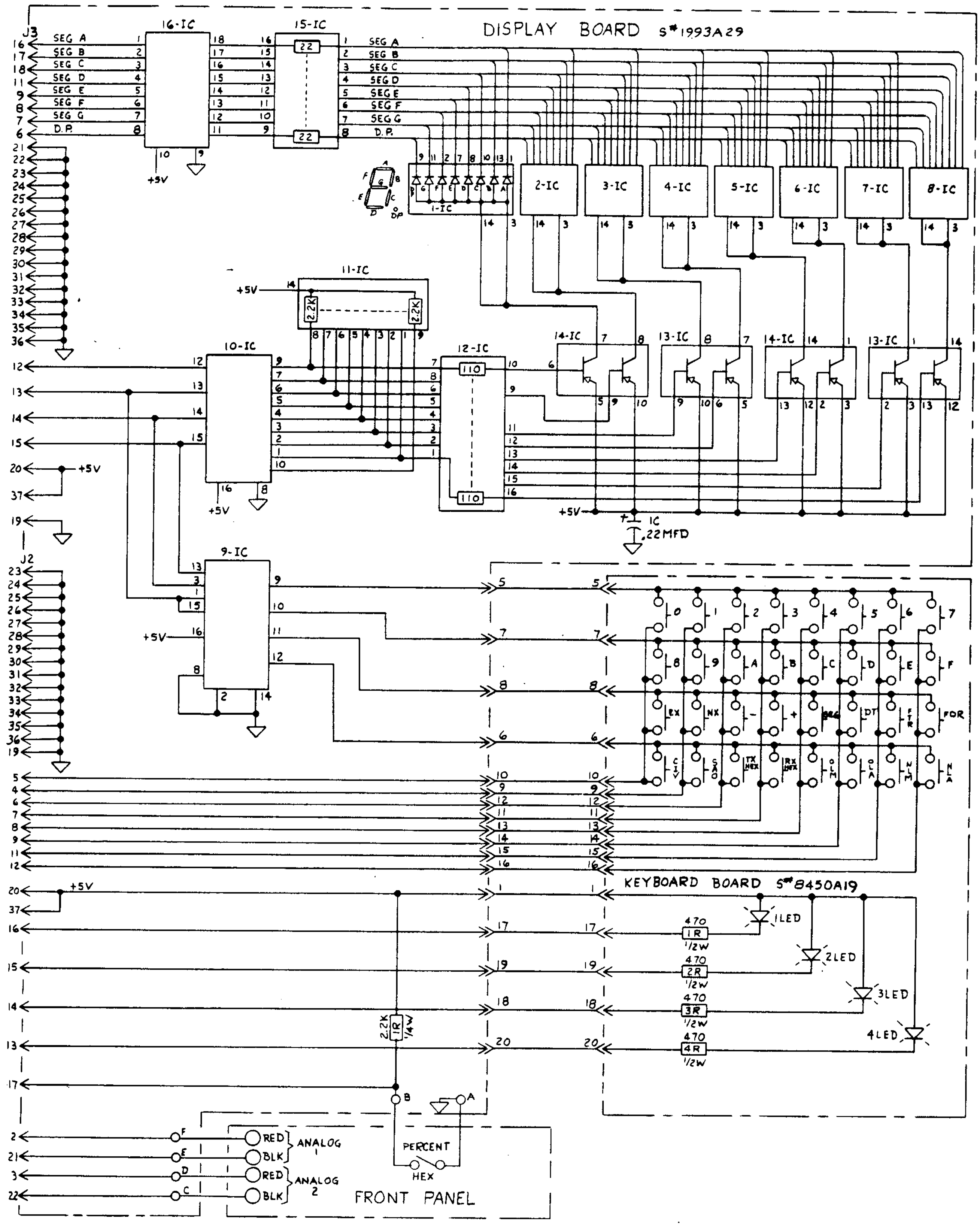
IC TYPE	IC NUMBER	+5V	GND	+15 V	-15 V
8251	33	26	4		
8279	8	40	20		
8284	11	18	9		
8287	7, 24, 32	20	10		
8212	6	24	12		
1488	22		7	14	1
1489	16	14	7		
LS393	25	14	7		
LS10	20	14	7		
LS04	5, 14, 26, 27, 28, 29	14	7		
LS02	13	14	7		
LS00	15, 19	14	7		
LS36	12, 18	14	7		
LS174	3	16	8		
LS14	30	14	7		
8205	4	16	8		
LS260	21	14	7		
LS161A	2	16	8		
LS373	23, 31, 34, 35, 36, 37	20	10		
8098	1	16	8		
S74	17	14	7		
DAC80	9, 10	13	21		


PMI
S#1993A28
[A5] DIR
[A4] DB/
P1
73 DAT0/
74 DAT1/
71 DAT2/
72 DAT3/
69 DAT4/
70 DAT5/
67 DAT6/
68 DAT7/
7-IC
D0 D1 D2 D3 D4 D5 D6 D7
[A5] DIR
[A4] SB/
24-IC
D8 D9 D10 D11 D12 D13 D14 D15
34-IC
[B4] IO2
36-IC
[B4] IO2
9-IC
J2 2
J2 21
4C*
.01MFD
22M 6R
1P 100K*
+15
-15
3.9M 4R
2P 100K*
+15V
3C .22 MFD
-15
+5V
34C .22 MFD
31C .22 MFD
2VR
+15
+24
P2 55
46C .22MFD
1VR
-15
-24
P2 59
45C .22MFD
35-IC
[B4] IO3
37-IC
[B4] IO3
10-IC
J2 3
22
6C*
.01MFD
22M R7
3P 100K*
3.9M 5R
4P 100K*
5C .22 MFD
35C .22 MFD
32C .22 MFD
P1
65 DAT8/
66 DAT9/
63 DAT10/
64 DAT11/
61 DAT12/
62 DAT13/
59 DAT14/
60 DAT15/
32-IC
[A4] DB/
[A5] DIR
J2 20
37
6-IC
16
15
14
13
[B3] IO4/
1LED
(G)
330 1R
+5V
2.2K 9R
J3 19
J2 19
17
2.2K 8R
23-IC
[D1] [D4]
[B2] IO1/
1K 12R
23 24 25 26 27 28 29 30 31 32 33 34 35 36
+5V
2C, 7C TO 11C, 13C TO 25C
27C TO 30C
33C, 36C TO 44C
.22 MFD
31-IC
D8
[B2] IO1/
1K 13R
*- NOT ON G01

C

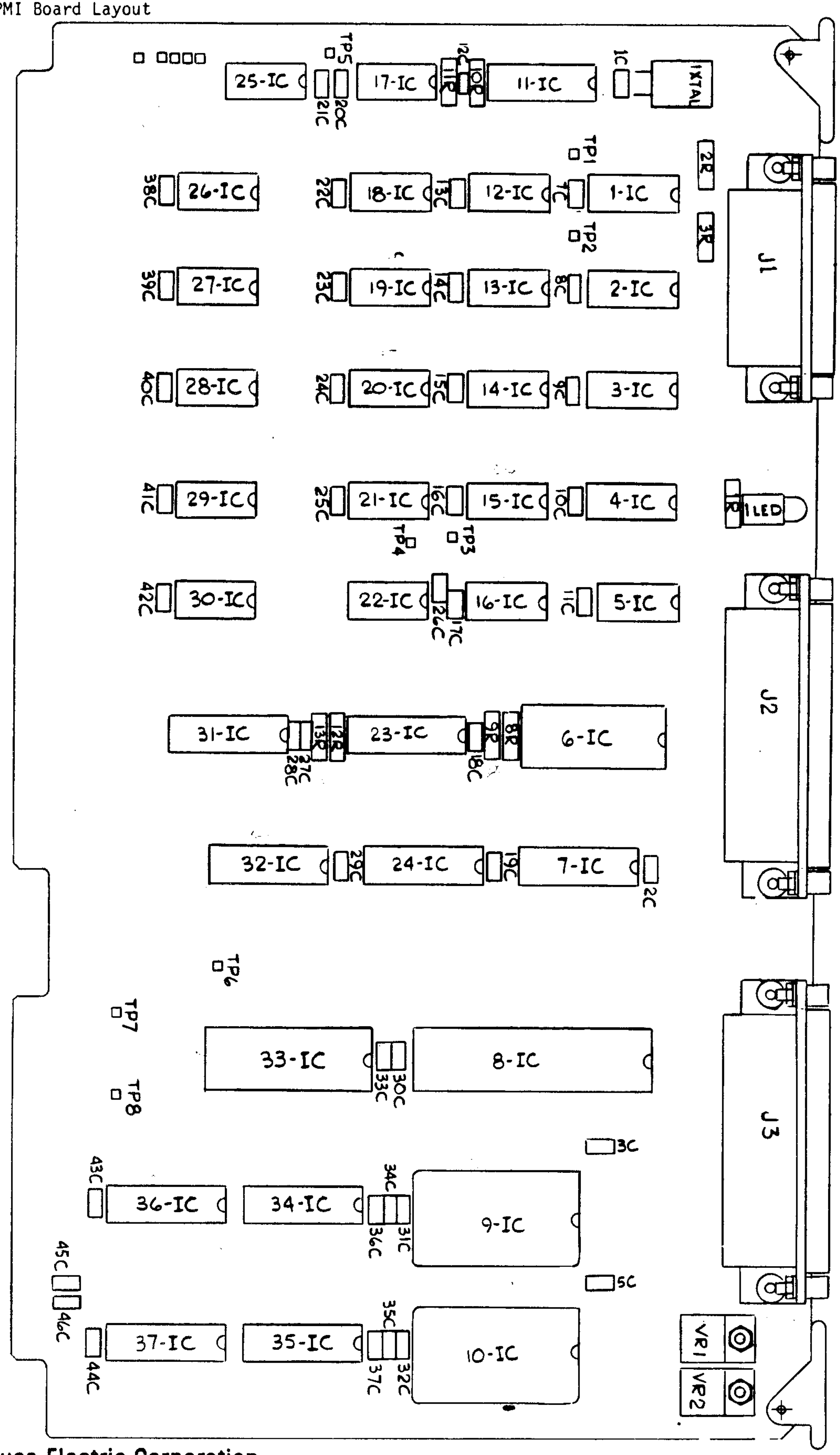

PMI
S# 1993A28
8-IC
D0
D1
D2
D3
D4
D5
D6
D7
INIT
BDRD/
PCLK
A1
RBDEN/
BDWR/
[C8]
[B7]
[A3]
[D5]
[B1]
[B3]
[A4]
12
13
14
15
16
17
18
19
9
10
3
21
22
11
31
30
29
28
27
26
25
24
35
34
33
32
38
39
1
2
5
6
7
8
J3
16
17
18
11
9
8
7
6
12
13
14
15
J2
5
4
6
7
8
9
11
12
-15V
.22 MFD
26C
+15
[C5]
22-IC
J1
3
6
5
4
20
2
11-IC
17
16
13
1
2
11
1K
10R
+5V
XTAL
14.7456 MHZ
1C
10 PFD
TP1
[D2]
PCLK
33-IC
BDWR/
BDRD/
INIT
D0
D1
D2
D3
D4
D5
D6
D7
[A4]
[A3]
[B7]
[C8]
10
13
21
27
28
1
2
5
6
7
8
20
19 TXD
24 DTR/
23 RTS/
17 CTS/
22 DSR/
3 RXD
16-IC
14
TP7
15
TP8
9
25
11
12
A1
[B1]
TTYEN/
[B3]
17-IC
12
9
11
8
17-IC
2
5
3
6
13
1
1K
11R
+5V
25-IC
4800
1200
110
300
3
5
13
9
1
TP5
12
ICLR
2
8
10
11
20-IC
5
3
4
6
30-IC
1
2
J1
1
7
21
22
23
24
25
26
27
28
29
30
31
32
33
34
35
36

D

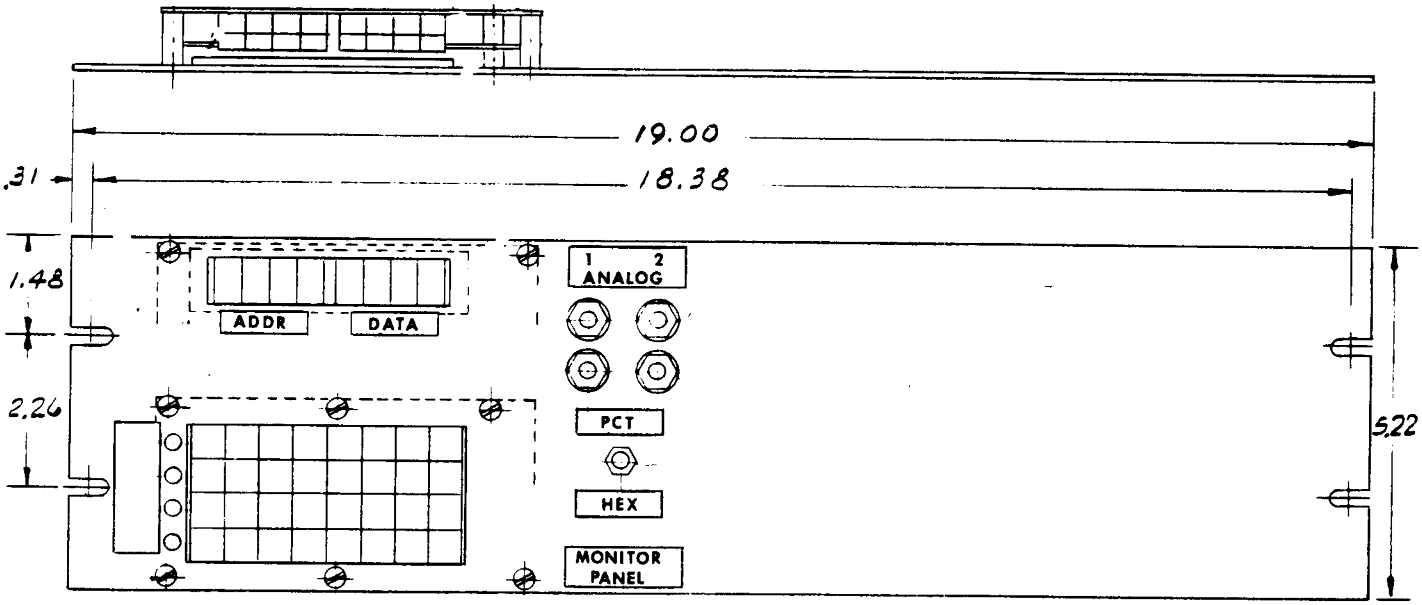

DISPLAY BOARD S#1993A29
J3
SEG A
SEG B
SEG C
SEG D
SEG E
SEG F
SEG G
D.P.
16-IC
15-IC
1-IC
2-IC
3-IC
4-IC
5-IC
6-IC
7-IC
8-IC
11-IC
2.2K
10-IC
12-IC
110
14-IC
13-IC
9-IC
+5V
IC
.22MFD
J2
KEYBOARD BOARD S#8450A19
0
1
2
3
4
5
6
7
8
9
A
B
C
D
E
F
EX
NX
-
+
DT
FOR
TX HEX
RX HEX
470
1R
2R
3R
4R
1/2W
1LED
2LED
3LED
4LED
2.2K
1R
1/4W
B
A
PERCENT
HEX
RED
BLK
ANALOG 1
ANALOG 2
FRONT PANEL

E

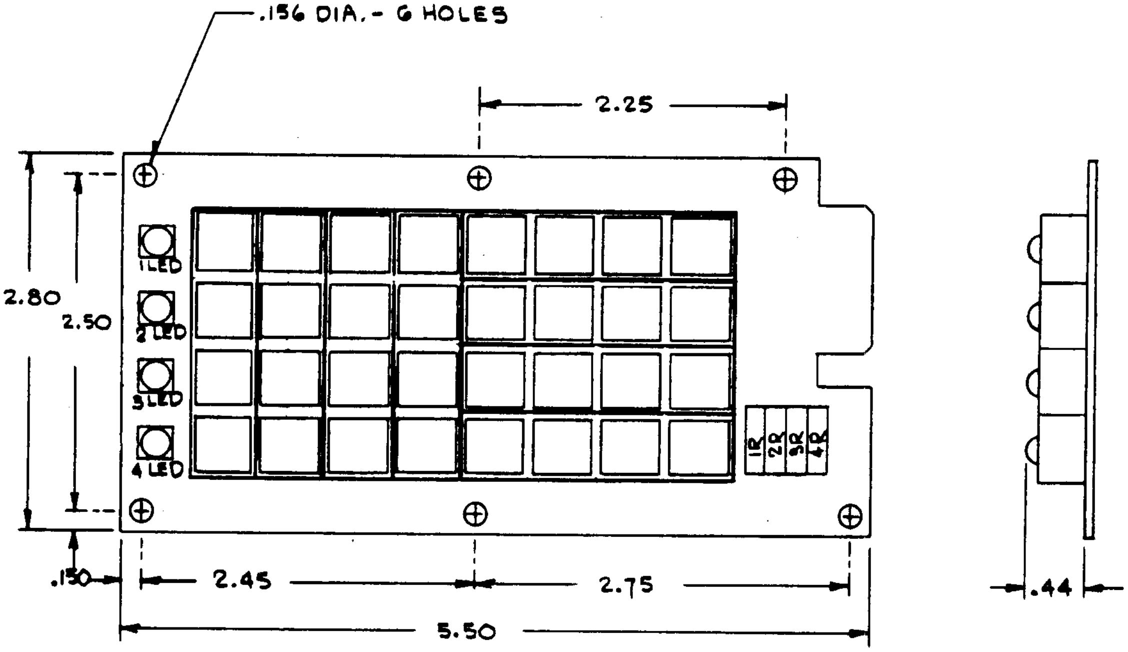
G. PMI Board Layout



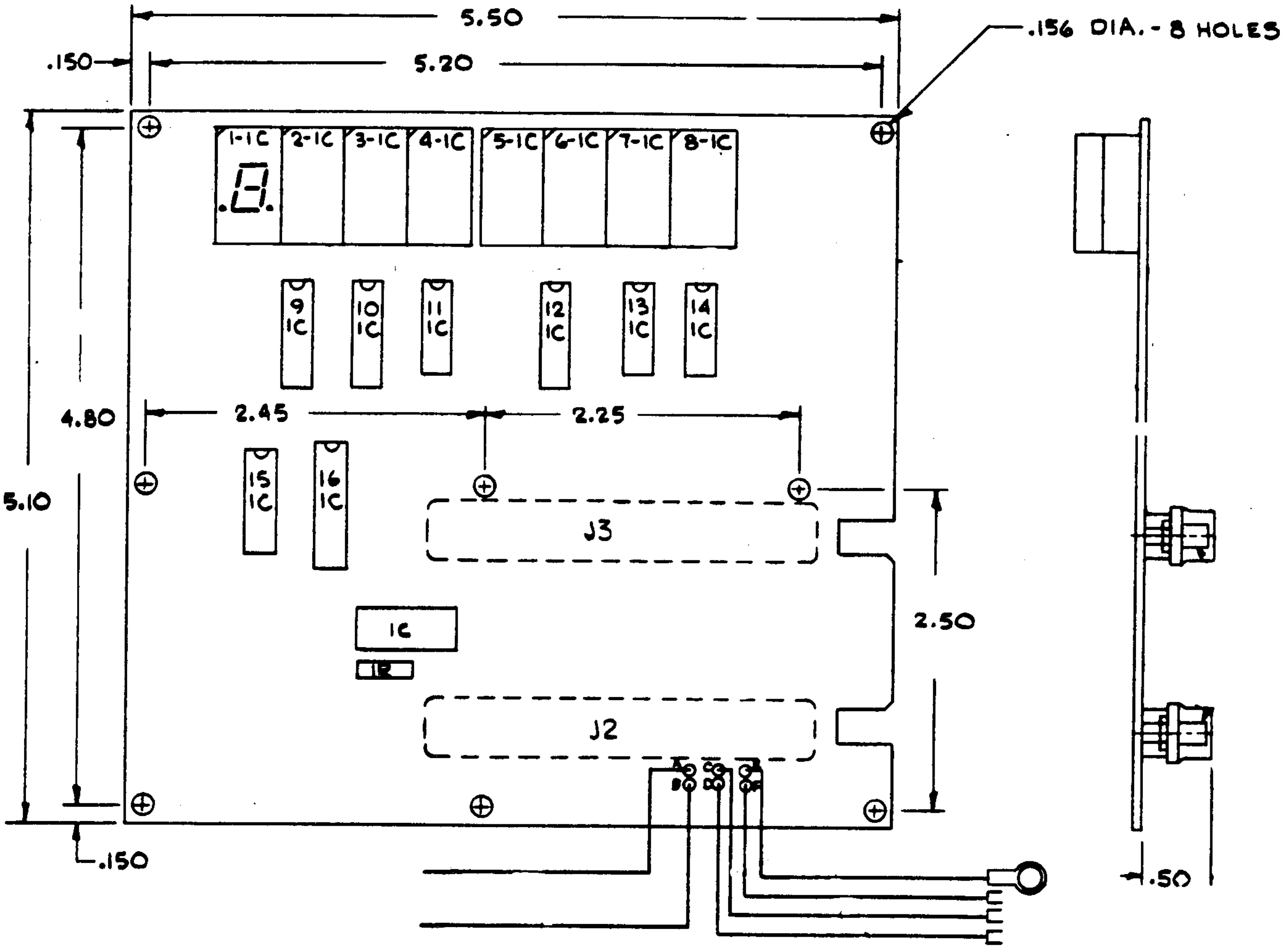
H. Monitor Layout Panel



I. Keyboard and Display



SEE SHEET 10 FOR KEYBOARD INFO.


5.50
.150
5.20
.156 DIA. - 8 HOLES
1-IC
2-IC
3-IC
4-IC
5-IC
6-IC
7-IC
8-IC
9 IC
10 IC
11 IC
12 IC
13 IC
14 IC
4.80
5.10
2.45
2.25
15 IC
16 IC
J3
2.50
IC
J2
A
B
C
D
.150
.50