Technical Reference Manual

Part No. 2223216-0001, Rev. A
May 1984

TEXAS INSTRUMENTS

INSERT LATEST CHANGED PAGES, DESTROY SUPERSEDED PAGES

LIST OF EFFECTIVE PAGES

Technical Reference Manual
Part No. 2223216-0001
Preliminary Issue: January 1983
Revision A: May 1984

Total number of pages in this publication is consisting of the following:

Page No.	Change No.	Page No.	Change No.	Page No.	Change No.
Front Cover	0	C-1 — C-12	0		
Effective Pages	0	D-1	0		
Preface	0	E-1 — E-6	0		
v — xiv	0	F-1 — F-3	0		
1-1 — 1-6	0	G-1 — G-9	0		
2-1 — 2-61	0	H-1 — H-7	0		
3-1 — 3-72	0	Index-1 — Index-9	0		
4-1 — 4-76	0	User's Response	0		
5-1 — 5-202	0	Mailer	0		
6-1 — 6-44	0	Back Cover	0		
A-1 — A-5	0				
B-1 — B-2	0				

CHANGE NOTICES

Revision Letter	Date	ECN Number	ECN Level	Description
A	5/3/84	518860	B	Update manual.

Address all correspondence regarding orders to:

Texas Instruments Incorporated
Data Systems Group
P.O. Box 1444, M/S 7793
Houston, TX 77251

PREFACE

The Technical Reference Manual contains detailed information on the design and function of the Texas Instruments Professional Computer and is intended for use by software and hardware designers and other technical persons.

This manual is divided into six major sections:

Section 1. Introduction - Provides a general description of the Texas Instruments Professional Computer and identifies its various configurations, options, and accessories. This section also includes tables listing environmental requirements for the system.

Section 2. System Hardware - Provides a detailed description of each component of the system including specifications and interface information. This section also includes hardware programming data such as coding tables, registers, and signal pin-outs.

Section 3. Hardware Options - Provides a detailed description of the options available for the system. This section contains specifications, interface information, and hardware programming data such as coding tables, registers, and signal pin-outs.

Section 4. Device Service Routines - Describes the ROM, gives interrupt vector lists, and a keyboard scan coding table.

Section 5. Assembly Drawings and Lists of Materials - Includes detailed drawings for all field replaceable assemblies and options. A List of Materials, identifying all components and piece parts, accompanies each assembly drawing.

Section 6. Schematics and Logic Drawings - Provides logic diagrams and schematics for each component and field replaceable assembly of the Texas Instruments Professional Computer.

The appendixes provide reference information, such as definitions of all I/O addresses, and a complete memory map (covering the motherboard, all memory connected to the expansion bus, and the memory expansion bus). Also included are complete information on the character sets furnished with the computer and a breakdown of the power allocation between the various options and printed wiring boards.

Table of Contents

List of Illustrations

List of Tables

Section 1

INTRODUCTION

1.1 SYSTEM COMPONENTS

The basic Texas Instruments Professional Computer system consists of three major parts: the keyboard, the system unit (including the diskette drive), and a monochrome display unit. A general description of each is given in this section. The available options are also briefly described in this section. For more detailed information, refer to Section 2, "System Hardware", and to Section 3, "Hardware Options."

1.1.1 Keyboard

The low-profile keyboard is easy to use. The large, sculptured, typewriter-like keys grouped on the main keyboard are used to enter alphanumeric data. The smaller numeric keypad on the right side of the keyboard can be used as a calculator. A five-key cluster between these two groups controls the display cursor movement. Twelve programmable function keys are arranged in three groups of four keys each across the top of the keyboard.

Other keyboard features include:

* A full-length tilt-bar, adjustable from 5 degrees to 15 degrees.

* The sculptured, low-profile keys, which comply with the European 30-millimeter (mm) home row height requirements.

* Tactile-designed F and J keys, which help to locate the "home" position on the alphanumeric keys.

* A raised dot on the numeric keypad 5, indicating the center key.

* A keyboard microprocessor, which converts keystrokes into character information and conducts keyboard diagnostics on every power-up.

1.1.2 System Unit

The system unit is the heart of the computer. The basic configuration includes the central processing unit (CPU), the floppy disk controller (FDC), a parallel printer port, a power supply, a read-only memory (ROM), and 64K bytes (K = 1024) dynamic random-access memory (RAM). A cathode-ray tube (CRT) controller board is standard equipment.

The system unit board is a 361.95 x 215.9-millimeter (mm) (14.25 x 8.5-inch (in)) printed wiring board (PWB) mounted horizontally on the bottom of the system unit chassis. This board houses the microprocessor and control logic. It also supports an expansion bus with five card-edge connectors for option boards and another connector for a memory expansion option.

The system unit power supply is a switching-type, 110-watt (W) unit with three output levels. It will sustain a system equipped with every combination of options.

The 5 1/4-in diskette drive is a mass storage device for reading or writing data to a removable diskette. The Texas Instruments Professional Computer uses a double-density, modified frequency modulation (MFM) recording format. This format requires certified double-sided, dual-density, soft-sectored 5 1/4-in diskettes. The data separation logic uses a phase-lock loop technique for reliability. The computer is equipped with one diskette drive, which can store approximately 320K bytes of data.

1.1.3 Display Unit

The display unit furnished with the Texas Instruments Professional Computer is a high-resolution (720 x 300 pixels), composite video, green phosphor monochrome unit. The standard CRT controller contained in the system unit supports eight intensity levels for the display. The display presents information in a 25-line x 80-column alphanumeric format, which works well with the bit-mapped graphics option. The display unit is specially adapted to accomodate the horizontal scan rate of 19 200 lines per second .

1.2 OPTIONAL COMPONENTS

There are several options available for the Texas Instruments Professional Computer. These options include additional 320K-byte diskette drives, a Winchester disk drive, expansion memory boards (which can expand the system memory to 768K bytes), a synchronous-asynchronous communications board, internal modem boards, a graphics video controller board, and a high-resolution color display unit. A general description of each of these options is given in the following paragraphs. If more detailed information is needed, refer to Section 3, Hardware Options.

1.2.1 Diskette Drive

One internal diskette drive is standard equipment for The Texas Instruments Professional Computer. Enough internal space is available to install either a second diskette drive or a Winchester disk drive. You can also install two external drives.

Diskettes used with the Texas Instruments Professional Computer must be certified double-sided, dual-density, soft-sectored, 5 1/4-in diskettes.

1.2.2 Winchester Disk Drive

The Winchester disk drive and controller option is available in 5- or 10-megabyte capacities. You can install the Winchester disk drive in the space set aside for the second diskette drive.

1.2.3 Expansion Memory Boards

The system unit board contains 64K bytes of dynamic RAM. Adding expansion RAM boards can increase the system memory to a total of 768K bytes. First, use the expansion RAM option boards that plug into the memory connector on the motherboard. These boards are available in 64K-, 128K-, or 192K-byte capacities. After adding the 192K-byte board (bringing the total to 256K bytes), further expansion requires that you add a 256K-byte board that plugs into the expansion bus. To reach the 768K-byte total, another 256K-byte board attaches (piggyback style) to the board on the expansion bus.

1.2.4 Synchronous-Asynchronous Communications Board

The synchronous-asynchronous communications (sync-async comm) board option allows either synchronous or asynchronous communications through an RS-232-C interface. The sync-async comm board supports asynchronous data rates from 50 bits per second (bps) to 19 200 bps.

1.2.5 Internal Modem Boards

Two versions of the internal modem board option are available: a 300-bps board providing Bell 103-compatible communication, and a 300/1200-bps board providing Bell 212A-compatible communications.

1.2.6 Graphics Video Controller Board

The graphics video controller board option is available in either one or three planes. It provides a resolution of 720 horizontal by 300 vertical picture elements (pixels).

1.2.7 Color Display Unit

The 13-in color display unit permits the display of high-resolution (720 x 300 pixels) colors. The standard CRT controller located on the system unit board supports eight colors for the unit, which presents information in a 25-line x 80-column format. Used with the graphics video controller board option, the color display unit produces high-quality raster and character graphics.

1.3 ENVIRONMENTAL CONDITIONS

The next four tables list environmental conditions for the Texas Instruments Professional Computer. Table 1-1 lists the storage conditions for a standard system. (Storage assumes that the system is enclosed in the shipping container.) Table 1-2 lists the operating conditions for a standard system. Table 1-3 lists the storage conditions for a system that includes a Winchester disk. Table 1-4 lists the operating conditions for a system that includes a Winchester disk.

Table 1-1 Storage Conditions, Standard System

Temperature	-30 C to +70 C (50 C maximum for diskette)
Relative humidity	10% to 90%, no condensation
Shock	30 Gs, half-sinusoidal pulse with 30 ms duration along X and Y axes. 20 Gs, half-sinusoidal pulse with 30 ms duration along Z axis.
Vibration	Sinusoidal, 5 to 250 Hz linear sweep at 1 octave/minute with 0.50 input. Dwell 15 minutes at resonant points (2X input level.)
Altitude	45 000 feet maximum

Table 1-2 Operating Conditions, Standard System

Temperature	+10 C to +40 C with gradient less than 10 C per hour
Relative humidity	20% to 80%, no condensation
Shock	5 Gs, half-sinusoidal pulse with 10 ms duration along any of the three perpendicular axes.
Vibration	0.5 Gs peak accelleration in the range of 5 to 250 Hz, linear sweep at 1 octave/minute.
Altitude	10 000 feet maximum

NOTE

Derate the upper limit of the operating temperature by 1 C for every 1000 feet above the first 500 feet.

Table 1-3 Storage Conditions, System with Winchester Disk

Temperature	-30 C to +60 C with gradient less than 10 C per hour
Relative humidity	20% to 80%, no condensation
Shock	30 Gs, half-sinusoidal pulse with 11 ms duration.
Vibration	20 Gs, half-sinusoidal pulse with 11 ms duration.
Altitude	30 000 feet maximum 10 000 feet unpressurized

Table 1-4 Operating Conditions, System with Winchester Disk

Temperature	+10 C to +40 C with gradient less than 10 C per hour
Relative humidity	20% to 80%, no condensation
Shock	5 Gs, half-sinusoidal pulse with 10 ms duration along any of the three perpendicular axes.
Vibration	0.5 Gs peak acceleration in the range of 5 to 250 Hz, linear sweep at 1 octave/minute.
Altitude	10 000 feet maximum

NOTE

Derate the upper limit of the operating temperature by 1 C for every 1000 feet above the first 500 feet.

Section 2

SYSTEM HARDWARE

2.1 INTRODUCTION

This section describes the design and functions of the hardware in the standard Texas Instruments Professional Computer system. Hardware described in this section includes the keyboard, the system unit board and its two logical subdivisions, and the display unit. Figure 2-1 is a block diagram of the system showing the separate hardware components, including some options. The option hardware is described in Section 3, "Hardware Options."

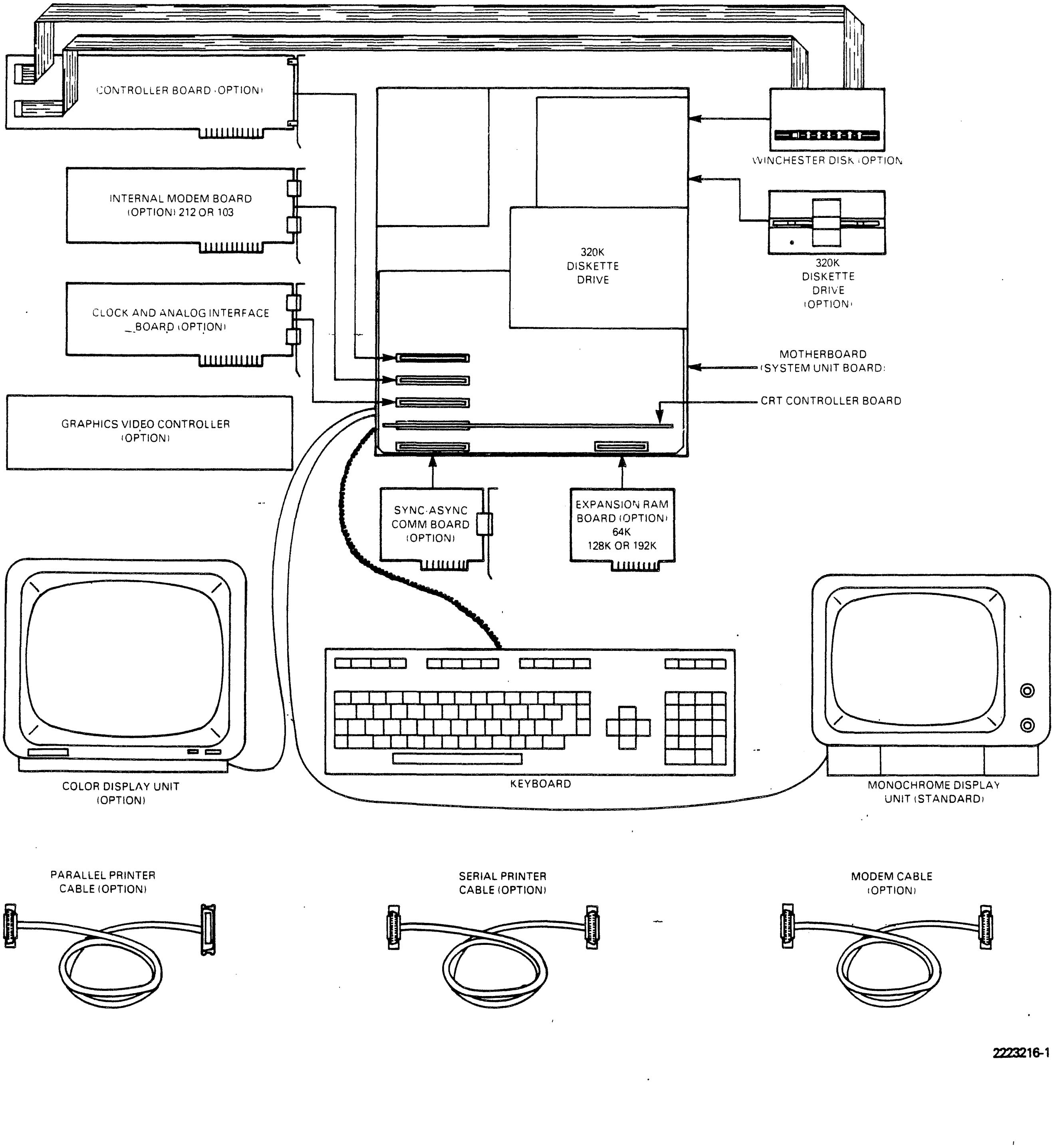


Figure 2-1 System Block Diagram

2.2 KEYBOARD

The electronic functions of the keyboard include:

* Scanning the key matrix and encoding keys depressed by the operator

* Transmitting data to the system unit

* Receiving and responding to commands from the system unit

* Implementing a software-switchable repeat-action function

* Performing n-key rollover

* Locking/unlocking the keyboard

* Performing a self-test

2.2.1 Encoding Keystrokes

The encoder scans the keyswitch matrix, detects valid keyswitch state changes, looks up the proper key code, and transmits the keycode as part of an 11-bit stream to the system unit. Each key causes either 1 or 2 bytes to be transmitted, based on the status of the SHIFT, ALT, CAPS LOCK, and CTRL keys. For specific details on byte definitions, refer to subsection 4.12.

Some user-programming of the function keys is possible at the application level. See the paragraph in Section 2 entitled, "Custom Encoding."

2.2.2 Transmission

The keyboard transmits data to the system unit at 2440 baud ± 1.50 percent. The keyboard transmits when one of the following conditions is met:

* When a valid key depression has been detected

* When a system command is understood and acted upon

When the user presses a key, the keyboard responds by sending the proper keycode byte or bytes across the keyboard transmit line. Keycodes are explained in detail in subsection 4.12 entitled "Keyboard DSR." Pressing some keys can signal repeat-action transmissions.

2.2.3 Receiving and Responding to System Unit Commands

The system unit transmits to the keyboard at 305 baud ± 1.50 percent. To respond to a system unit command, the keyboard transmits a response code to the system unit, indicating that the required action has been taken. The keyboard responds to every valid command. For certain conditions, such as parity errors, unknown commands, and start bit errors, the keyboard ingores the system unit commands and sends no response. If this happens, the system unit retries the command.

System unit commands and keyboard responses are listed, in hexadecimal form, in Table 2-1. In this table, the "Command Code" column lists the codes sent to the keyboard. The "Keyboard Response" column lists the code returned by the keyboard microprocessor. Typically, the microprocessor returns Self-test OK (code 70) to the system unit (except in the case of a failure during self-test).

NOTE

Throughout this manual, the symbol H denotes a hexadecimal address or value.

Table 2-1 Keyboard Commands and Responses

System Unit Command	Command Code (H)	Keyboard Response (H)	Response Meaning
Perform a power-up self-test and install default parameters	00*	70	Self-test OK
		71	Keyboard ROM error
		72	Keyboard RAM error
Turn repeat action <u>ON</u>	01*	70	Self-test OK
Turn repeat action <u>OFF</u>	02	70	Self-test OK
Lock keyboard	03	70	Self-test OK
Unlock keyboard	04*	70	Self-test OK
Turn keyclick <u>ON</u>	05**	70	Self-test OK
Turn keyclick <u>OFF</u>	06**	70	Self-test OK
Reset (same as 00)	07	70	Self-test OK
		71	Keyboard ROM error
		72	Keyboard RAM error
Return version (of keyboard ROM).	08	70,73	(2-byte code)

* Indicates default values.
** Keyclick requires a hardware modification. It is not presently supported.

2.2.4 Implementing a Software-Switchable Repeat-Action Function

A repeat-action key is one that automatically repeats when depressed for one-half second (s) or longer. As long as the key is held down, repeat-action transmissions from the keyboard to the system unit continue at a rate of 15 per second.

2.2.5 Performing n-Key Rollover

Repeat-action interacts with n-key rollover in the following manner. Pressing more than one nonmode key does not cause repeat-action. Instead, the most recent key pressed transmits to the system unit. When repeat-action is enabled and one key is pressed, that key is acted upon by the repeat-action function. The following examples clarify the relationship between rollover, repeat-action, and mode byte changes.

Example 1:

Assume that the following sequence of events occurs:

1. No mode bits are on.
2. The a key is depressed and held down for more than one-half second.
3. The b key is depressed.
4. The SHIFT key is depressed. (The SHIFT key can be held or released without altering the characters transmitted to the system unit.)
5. The b key is released.
6. The a key has not yet been released.

The result transmitted to the system unit and displayed is:

aaaaaaaaaaaaaabaaaaaaaaaaa...

Example 2:

Assume that the following sequence of events occurs:

1. No mode bits are on.
2. The a key is depressed and held down for more than one-half second.
3. The SHIFT key is depressed and held.
4. The b key is depressed. (At this point, the SHIFT key can be held or released without altering the characters transmitted to the system unit.)
5. The b key is released.
6. The a key has not yet been released.

The result transmitted to the system unit and displayed is:

aaaaaaaaaaaaaBAAAAAAAAAAA...

2.2.6 Locking/Unlocking the Keyboard

At certain times during system operation, the keyboard locks. During these times, all normal functions of the keyboard are suspended. That is, the keyboard does not scan, encode, or transmit data to the system unit. The keyboard locks if:

* The self-test is in progress.
* The self-test fails.
* The keyboard receives the LOCK KEYBOARD command.

The keyboard remains locked until one of the following conditions occurs:

* The self-test successfully completes.
* The keyboard receives the UNLOCK KEYBOARD command.

2.2.7 Performing a Self-Test

The keyboard performs a self-test when it receives code 00 from the system unit, interrupting any keyboard operation in progress. The self-test completely checks the keyboard system RAM and ROM, then transmits the results to the system unit using a code explained in paragraph 2.2.3, entitled "Receiving and Responding to System Unit Commands."

2.3 SYSTEM UNIT BOARD

The system unit board, or motherboard, is the heart of the computer. It is mounted on the bottom of the system unit chassis. The motherboard is divided into two logical function areas, one for system support and one for the expansion bus. Refer to Section 5, drawing 2223005, for logic diagrams of the system unit board. Figure 2-2 is a block diagram of the separate subsystems of the motherboard.

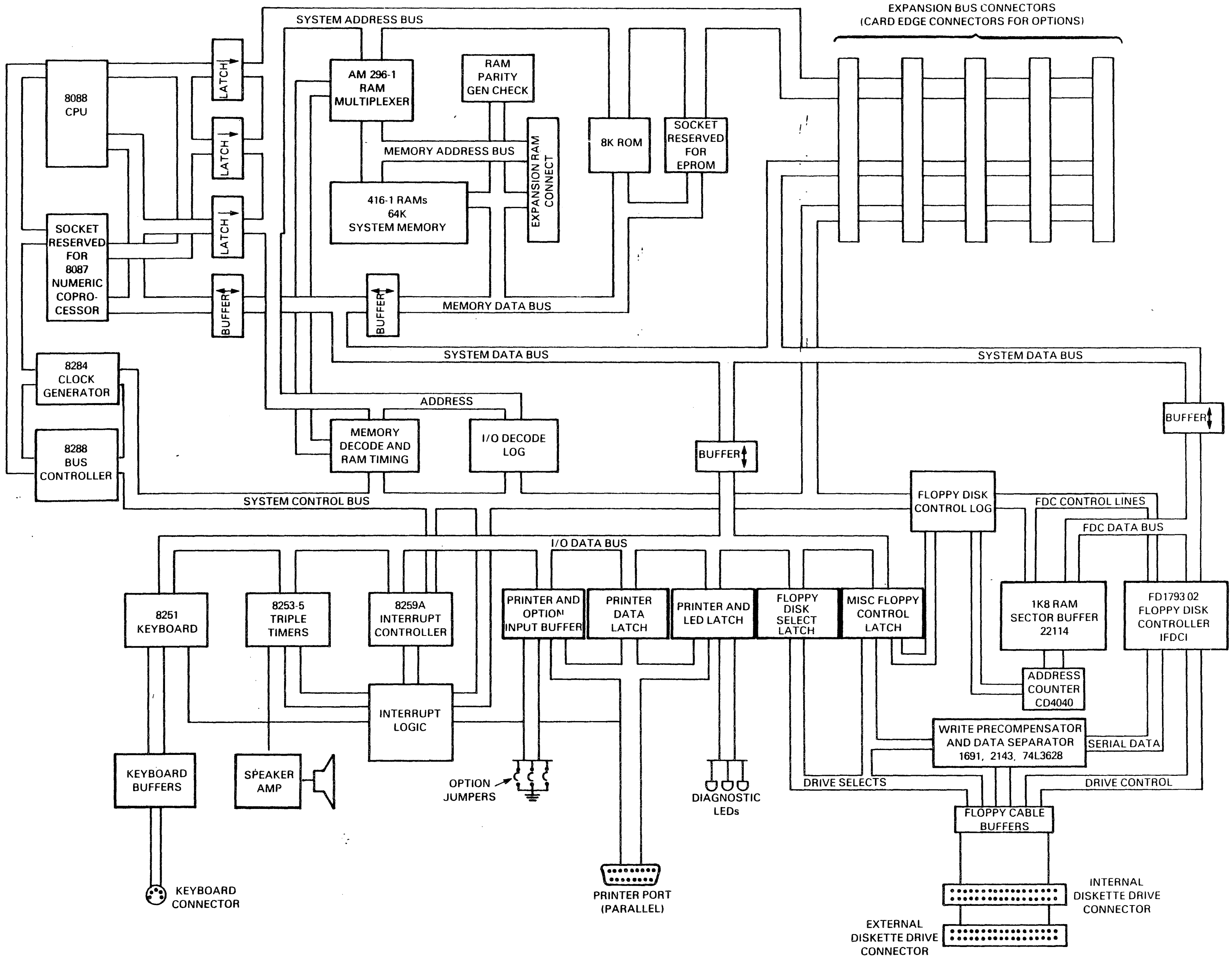


Figure 2-2 Motherboard Block Diagram

2.4 SYSTEM SUPPORT

That section of the motherboard dedicated to system support contains hardware and logic for the:

* Keyboard port
* System CPU (including microprocessors, clocks, bus controllers, and buffers)
* Motherboard input/output (I/O) system
* Motherboard interrupt system
* Motherboard memory system
* FDC subsystem (including buffers, write precompensation, and diskette drive interface)
* CRT controller

2.4.1 Keyboard Port

The Intel 8251A, a universal asynchronous receiver-transmitter (UART), is the port for serial data transmission between the motherboard and the keyboard. Data received by the UART always generates an interrupt to the interrupt controller. The transmit ready line does not generate an interrupt unless the transmitter in the UART is enabled. The keyboard port interrupt is ORed with the "interrupt request 7" line from the numeric coprocessor.

An SN75189A line receiver with a slowdown capacitor conditions the receive data signal to protect the signal from transients. The receiver hysteresis is approximately 1 V centered around 1.4 V, which improves the noise immunity. Another SN75189A buffers the transmit data line, providing a good voltage swing and drive to the keyboard cable. This buffer consists internally of an output transistor with a 2-kilo ohms (kohms) pullup resistor.

To improve diagnostics, the data set ready (DSR) line on the universal synchronous/asynchronous receiver transmitter (USART) connects to the keyboard connector through a SN75189A buffer. The transmit data line connects to the DSR line at the keyboard, which allows detection of a disconnected or defective keyboard.

The input clock to the transmit section is 19 531.25 Hz. The 8251 divides this frequency by 64 to generate a baud rate of 305. The input clock for the receiver is 156 250 Hz. This frequency is divided by 64 to generate a baud rate of 2441. Because these baud rates are close to the standard 300- and 2400-baud rates, system test instruments can simulate a keyboard with standard equipment.

2.4.2 System CPU

The system CPU consists of an Intel 8088 16-bit microprocessor, the CPU clock circuits, several CPU bus buffers and latches, a CPU bus controller, and the reset circuit. A special socket on the motherboard makes it easy to add the optional Intel 8087 numeric data processor (also called a numeric coprocessor).

The Intel microprocessors work together and, to attached components, appear to be a single chip. Therefore, the term CPU (as used in this manual) refers to both devices.

2.4.2.1 Optional Numeric Coprocessor. The user can choose to add an 8087 numeric coprocessor to the system unit board at any time. Once the 8087 is inserted into the socket provided, both the 8088 and the 8087 decode the special escape instructions. The 8088 does any memory-access computations required and accesses the first byte of memory according to the instruction. The 8087 decodes the instruction, "catches" the memory address generated by the 8088, requests the bus from the 8088, and completes the required memory access. After finishing with the bus, the coprocessor releases it so that the 8088 can continue with the next instruction. If necessary, the 8088 sends a WAIT instruction to the 8087, ensuring their synchronization.

2.4.2.2 CPU Clock Generator. The CPU clock generator consists of an Intel-designed 8284, a crystal, and some discrete components. To generate the 5.0 MHz clock frequency, the 8284 divides the crystal frequency (15.0 MHz ± 0.01 percent) by 3. The 8284 also contains logic to synchronize the WAIT- line from the expansion bus and memory subsystems with the RESET- line from the power-good circuit.

NOTE

> Signal names followed by a dash, such as WAIT-, are active low signals.

2.4.2.3 CPU Bus Buffering. The CPU operates in the so-called "maximum" mode of this integrated circuit. (For additional information, see the Intel literature on the 8088 and 8087 microprocessors.) The CPU uses a multiplexed address and data bus in order to reduce the number of pins required on the processor chip. For this reason, and to provide adequate buffering for the address and data lines on the expansion bus, a set of address latches (U5, U6, U7) and a data bus buffer (U8) are an integral part of the CPU.

2.4.2.4 CPU Bus Controller. The CPU bus controller chip (U3 8288) receives the status information from the processor and converts it into the lines MRDC- (memory read), AMWC- (advanced memory write), IORC- (I/O read), AIOWC- (advanced I/O write), INTA (interrupt acknowledge), DEN (data buffer enable), and DTR (data buffer direction control).

A simple open-loop signature analysis (SA) arrangement is provided to check out the CPU. Connecting pins E17 and E18 (on the motherboard) with a jumper and resetting the system (power up) causes the processor to execute a 0BFH opcode. The jumper disables the system data bus buffer U8, and the pullup resistors in U66 pull the bus up to a high state. Transistor Q1 pulls down data line AD6 to provide the "0" bit in the opcode. The segmented architecture then causes the processor to cycle from address FFFF0H through address FFFFFH and from 00000H through 0FFF0H during the SA loop.

NOTE

The symbol "H" denotes a hexadecimal address or value.

2.4.2.5 Reset Detection Circuit. The power-good (reset detection) circuit discovers insufficient power conditions on the motherboard by monitoring the 12-volt (V) power line. When the power drops, but does not shut down completely, this circuit causes an automatic restart. If the voltage falls to approximately 11 Vdc, a resistor/capacitor combination and a voltage comparator with transistor inverter hold the RESET line true for at least 3 milliseconds (ms).

2.4.3 Motherboard Input/Output System

The motherboard input/output (I/O) system decodes the I/O addresses for all the devices on the board. The input buffer and the various output latches are also components of the I/O system. Table 2-2 shows a map of the motherboard I/O addresses.

The various I/O devices have available 16 I/O address bits. Only 10 of these bits, a total of 1024 bytes, are decoded. Beginning at address 000H, the motherboard uses 48 bytes of this space. This leaves 976 bytes available for the expansion bus.

Table 2-2 lists the motherboard devices that are decoded and their addresses within the CPU I/O space. Appendix A provides a complete map of all system I/O addresses.

Table 2-2 Map of the Motherboard I/O Addresses

Hex Address	Device	Bit/Use
00000	U47 Latch	0 Speaker timer enable
		1 Timer 1 interrupt enable
		2 Timer 2 interrupt enable
		3 Single-density (FM) enable
		4 Track greater than 1/2 (TG43)
		5 Diskette side one enable (FSID-)
		6 Diskette mode control (M1)
		7 Diskette mode control (M0)
00001	U48 Input buffer	0 Option jumper E1-E2
		1 Option jumper E3-E4
		2 Option jumper E5-E6
		3 Parity interrupt pending
		4 Printer port BUSY
		5 Printer port paper out
		6 Printer port printer selected
		7 Printer port NO fault
00002	U49 Latch	0-7 Printer port data outputs
00003	U50 Latch	0 LED 1 OFF
		1 LED 2 OFF
		2 LED 3 OFF
		3 Parity interrupt enable
		4 Printer port not autofeed
		5 Printer port not strobe
		6 Printer port not initialize
		7 Printer ACK interrupt enable
00004	U51 Latch	0 Diskette drive SELECT 1
		1 Diskette drive SELECT 2
		2 Diskette drive SELECT 3
		3 Diskette drive SELECT 4
		4 Diskette drive MOTOR 1
		5 Diskette drive MOTOR 2
		6 Diskette drive MOTOR 3
		7 Diskette drive MOTOR 4

Table 2-2. Map of the Motherboard I/O Addresses (Concluded)

Hex Address	Device	Bit/Use
00005--0000F	Reserved	- -
00010	U44 8251 USART	Data register
00011	U44 8251 USART	Control register
00012--00013	Reserved	- -
00014	U45 8253 timer	Counter 0
00015	U45 8253 timer	Counter 1
00016	U45 8253 timer	Counter 2
00017	U45 8253 timer	Control register
00018	U46 8259A interrupt controller	- -
00019	U46 8259A interrupt controller	- -
00020	FDC command register or RAM	- -
00021	FDC track register	- -
00022	FDC sector register or RAM reset	- -
00023	FDC data register	- -
00024--0002F	Reserved	- -

2.4.3.1 I/O Decoding. A combination of three integrated circuits (IC) does the I/O decoding. The first IC is a hard-array-logic (HAL) device HAL12L6. The second is a 74LS138, which is a one-of-eight decoder. The third is one-half of a dual 74LS139, which is a one-of-four decoder.

Table 2-3 gives the array logic device programming. When the logical AND of terms from one row is ORed with the AND of terms from another row in the same section, the output goes active if the result is true. Expressed in Boolean terms,

$$\overline{IORQ} = (\overline{XS2} \times \overline{XS1} \times XS0 \times \overline{IORC}) + (\overline{XS2} \times XS1 \times \overline{XS0} \times \overline{AIOWC})$$

Table 2-3 Input/Output Signals - HAL12L6 Integrated Circuit

Output	XS2	XS1	XS0	DEN-	XA9	XA8	XA7	XA6	XA5	XA4	IORC-	AIOWC-	Comment
	Input												
IEN-	L	L	H	L	L	L	L	L	L	.	.	.	Read I/O
or	L	H	L	L	L	L	L	L	L	.	.	.	Write I/O
or	L	L	L	L	.	.	.	.	.	.	.	.	Interrupt acknowledge
or	.	.	.	.	.	.	.	.	.	.	L	L	Inactive term
XCS-	L	L	H	.	L	L	L	L	L	H	L	.	Read I/O at 74LS139
or	L	H	L	.	L	L	L	L	L	H	.	L	Write I/O at 74LS139
IORQ-	L	L	H	.	.	.	.	.	.	.	L	.	Read I/O
or	L	H	L	.	.	.	.	.	.	.	.	L	Write I/O
FLCS-	L	L	H	.	L	L	L	L	H	L	.	.	Read diskette
or	L	H	L	.	L	L	L	L	H	L	.	.	Write diskette
YCS-	L	L	H	.	L	L	L	L	L	L	L	.	Read I/O at 74LS138
or	L	H	L	.	L	L	L	L	L	L	.	L	Write I/O at 74LS138
XXXX-	H	.	.	.	.	.	.	.	.	.	.	.	Halt
or	.	L	.	.	.	.	.	.	.	.	.	.	
or	.	.	L	.	.	.	.	.	.	.	.	.	
or	.	.	.	L	.	.	.	.	.	.	.	.	

Legend:
- L = Low signal.
- H = High signal.

2.4.3.2 Parallel Printer Port. Printers with Centronics-compatible interfaces use the parallel printer port. This port contains a 25-pin female, D-type connector.

The basic signals are the output data lines from U44, the PTSTR- signal that strobes the data into the printer, and the PBUSY and PACK- lines, which indicate to the CPU the printer's readiness to receive a character. In regular printer operation, the PBUSY line goes high when the printer is not ready to receive a character and low when the printer can accept a character. The PACK- line goes low for a short time when the printer finishes with the current character. The rising edge of this line generates an interrupt when printer interrupts are enabled by the PTEN line. This interrupt is ORed with the "interrupt request 5" line on the expansion bus.

The pin-out of the port is given in Table 2-4. Pin numbers for the 36-pin printer connector (at the printer end of the cable) are given in parentheses. The extra lines are used for various control and status functions associated with the printer port.

Table 2-4 Printer Port Pin-Out

Signal	Return	Signal Name	Source	Signal
1	19	DATA STROBE-	System	Data is sampled when signal is low.
2		DATA 1	System	Data output bit.
3	20(21)*	DATA 2	System	
4		DATA 3	System	
5	21(23)	DATA 4	System	
6		DATA 5	System	
7	22(25)	DATA 6	System	
8		DATA 7	System	
9	23(27)	DATA 8	System	
10		ACKNOWLEDGE-	Printer	Another character can be received.
11	24(29)	BUSY	Printer	No data can be sent when signal is high.
12		PAGE END	Printer	Printer is out of paper when signal is high.
13		SLCT (ON LINE)	Printer	Printer is online when signal is high.
14		AUTO FEED-	System	Printer is to line feed on carriage return when signal is low.
15(32)		FAULT-	Printer	Indicates a fault when signal is low.
16(31)	25(30)	INIT-	System	Resets printer when signal is low.
17(36)	18(33)	SELECTION-	System	Always low.

* The numbers in parentheses are the pin numbers for the 36-pin Centronics-type connector.

2.4.3.3 Timers. The 8253-5 counter/timer IC provides three separate timing units. In this system, one is used as a programmable speaker oscillator, and the other two are programmable interval timers.

The speaker timer is clocked by a square wave of 1.25 MHz. Divisors up to 65 536 can generate output frequencies as low as 19 Hz. The high input frequency creates output tones that are more musically accurate. The speaker timer clock is internally gated with the speaker enable (SPKEN), an output of latch U47. This signal allows the interruption of tones without a reprogramming of the timer.

The second timer (Timer A) is used in system-timing applications and as a real-time clock. It generates an interrupt signal on the rising edge of the timer output when the enable line (address 0 bit 1) is set high. Toggling this line low resets the interrupt; holding this line low disables the interrupt completely. The interrupt level is 3. The input clock frequency to the timer is 625 kilohertz (kHz). A divisor of 62 500 generates a pulsewidth of 100 ms, while a divisor of 15 625 generates a pulsewidth of 25 ms.

The third timer (Timer B) is used for special-purpose timing applications. It generates an interrupt on the rising edge of the timer output when the enable line (address 0 bit 2) is set high. Toggling this line low resets the interrupt; holding this line low disables the interrupt completely. This line is shared with the expansion interrupt line IR2. The interrupt level is 2. The input clock frequency to this timer is 625 kHz.

2.4.3.4 Speaker Amplifier. The speaker timer output goes to an amplifier (LM 386) that drives the 8-ohm speaker, providing sufficient volume and allowing mixing of signals from external sources (option expansion cards). To mix other signals with this signal, connect any other signal source (such as the speech option board) to P12, the summing input.

2.4.4 Motherboard Interrupt System

The motherboard interrupt system can encode eight separate interrupts and vector the central processor to eight separate interrupt routines. A nonmaskable interrupt (NMI) (which produces the highest-priority interrupts) is also available.

The majority of the interrupt logic is contained within the Intel 8259A interrupt controller chip. The 8259A is programmed for level-sensitive input and is the master (only) interrupt controller. During the INTA cycle, the decoding logic array always enables the contents of the I/O data bus onto the system data bus. This information is the vector from the 8929A chip, and the system, therefore, requires only one controller.

The 8259A chip assigns priority to the incoming interrupts, allows masking of interrupts, and provides the vector to the CPU during the interrupt acknowledge (INTA) cycle. A series of OR gates and flip-flops permit some interrupt levels to be shared, cause some inputs to be edge-triggered, and cause others to be level-triggered.

The interrupts that come from the expansion bus are active high and are, therefore, terminated with a 4.7-kohm pulldown resistor to ground. All the pulled-down inputs are connectied to the 8259A chip, either directly or through a CMOS OR gate. This connection prevents the gate input current from raising the input voltage above the legal "low" level through the pulldown resistor.

CAUTION

Even though the system is protected, programmers and designers using interrupts on the expansion bus should be sure to "mask off" unused interrupt lines as a matter of good programming practice.

The NMI detects parity errors on the motherboard RAM system. To generate this interrupt with software, set the DTR line on the 8251A USART. The RAM can then be tested without parity-error interruption.

The interrupt levels and their expected uses are given in Table 2-5.

Table 2-5 Interrupt Level Assignments

Interrupt	Bus Line	Use
NMI	A01	System parity error, CRT interrupt
IR0	B04	Communications port 1
IR1	B24	Communications port 2
IR2	B25	Communications port 3
		System board timer 2
		Local area net board buffer full/empty
IR3	na	System board timer 1 (clock)
IR4	B23	Communications port 4
IR6	B21	Diskette drive, Winchester disk
IR7	na	Keyboard, numeric coprocessor

na = Not applicable.

2.4.5 Motherboard Memory System

The memory system on the motherboard consists of 64K bytes (K = 1024) of dynamic RAM, up to 16K bytes of ROM, decoding logic to establish the addresses, and timing and refresh logic to operate the system. A connector and the necessary logic permit the addition of one of the expansion RAM boards. These boards are available in 64K-, 128K-, and 192K-byte capacities. After adding the 192K-byte board (bringing the total to 256K bytes), further expansion requires the addition of a 256K-byte board that plugs into the expansion bus. (This board and another memory expansion board are fully described in Section 3.)

2.4.5.1 Motherboard Memory Addressing. The memory space of the processor devices used by the motherboard is given in Table 2-6. The balance of the system memory is given in Appendix B.

Table 2-6 Motherboard Memory Map

	Address	Device
Dynamic RAM:		
	00000-0FFFF	64K-bytes motherboard RAM
	10000-1FFFF	64K-bytes expansion RAM board bank 1
	20000-2FFFF	64K-bytes expansion RAM board bank 2
	30000-3FFFF	64K-bytes expansion RAM board bank 3
ROM Usage:		
	FC000-FDFFF	8K ROM space, one wait state (XU62)
	FE000-FFFFF	8K system ROM, one wait state (U63)

2.4.5.2 Memory Control Logic. A bidirectional buffer (U61) separates the main system data bus from the motherboard expansion memory, thereby providing sufficient drive and margin to the data transfers. U28, the memory hard array logic chip HAL16R4, in combination with U53, the 74LS139 decoder, handles decoding and timing for the ROMs. Because ROMs and EPROMs (erasable programmable read-only memories) are generally slow devices, a wait state is added to all accesses to these devices.

The ROM access times are listed in Table 2-7.

Table 2-7 ROM Access Times

Function	Time Required (in Nanoseconds)
CS-ROM access	410
ROM address access	577

I/O Wait States. The HAL chip also contains the logic to add a wait state to all I/O accesses made by the CPU. The wait state is necessary because many of the I/O devices operate too slowly when the system buffer and setup and decode times are included. With the wait state, the control lines are active for approximately 600 nanoseconds (ns).

Memory Refresh Logic. The RAM refresh logic operates synchronously with the accesses to the RAM memory. Refresh cycles begin only when a RAM memory cycle is not in progress. This implies that the RAM refresh can occur at the same time as accesses to other system memory (ROMs) or I/O space. Each time a refresh cycle begins, a refresh timer (one-shot U29) starts. When it times out, it provides the signal to begin another refresh cycle. This timer is set to 15 microseconds (us) maximum, which allows for the worst-case refresh-request latency. To maintain the contents of the RAM under worst-case conditions, the refresh must occur at least 128 times within 2 ms. (The average refresh timing is once per 15.625 us). The worst-case latency for a refresh request is about 600 ns.

Once a refresh cycle has begun, it must be completed (including the precharge) before the next cycle begins. If a RAM access cycle starts before the refresh cycle completes, the HAL state machine puts the CPU into a wait state until the refresh operation completes. In the worst case, this delay could extend the normal memory access time by four wait states, or 800 ns.

Assuming a refresh timer value of 14 us and an average 600-ns slowdown of the CPU, the refresh overhead is approximately 4.3

percent average or 5.7 percent worst case.

2.4.5.3 CAS and Address Multiplexer Switch. A delay line from the RASI- (row address strobe input) line produces the SWM (the address multiplexer control). SWM ensures an adequate row address hold time (40 ns) and still operates the RAM quickly enough to finish the access within the system cycle time.

The CASI- (column address strobe input) timing depends on whether the cycle is a read or a write. If the cycle is a read, the CASI- signal is taken from the delay line 20 ns after the SWM signal to produce the ACAS- (advance column address strobe). ACAS- ensures an adequate column address setup time to the RAM and still gives fast RAM access. If the cycle is a write, then the CASI- signal is taken from the falling edge of the system clock, which is about 150 ns after the occurrence of RASI-. This delay allows time for the data from the processor to propagate through the data buffers and the parity generator chip (U31 74LS280).

To control the generation of the CASI- pulse, flip-flop U33 is timed with CLK- (the system clock), samples the delay line (ACAS-), and is reset by MRDC- (the memory read signal). The output of the flip-flop is then logically ANDed (U34) with the ACAS- signal to generate the actual CASI- signal. To prevent the generation of a CASI- pulse during refresh, the refresh row address strobe (RRAS-) line holds flip-flop U33 in the preset state during a refresh. This forces the output of OR gate U34 (CASI-) to a high level.

2.4.5.4 Parity Generation and Checking. The parity generator/checker chip (74LS280) generates a "1" to the parity RAM bit whenever there is an even number of 1's in the data byte being written. The parity RAM chip has a separate data bus to drive the output line. A pullup resistor holds this line high when it is not driving the output (as in a write cycle). The parity data is then taken from the "odd sum" output of the parity generator and used to write to the RAM.

This method of parity checking does not cause a parity error when the system attempts to read from nonexistent RAM. (To determine the size of system memory, the system software sometimes "feels" for memory not present.)

When the RAM is read, all of the data bits and the parity bit are presented to the generator/checker and the parity output is sampled at the end of the read cycle. If parity checking is enabled and discovers a parity error, flip-flop U33 is set to interrupt the CPU. Once set, this flip-flop must be reset by software before additional interrupts can be given. If the enable bit (address 3 bit 3) is held low, then no parity interrupts (PINT) are generated. To distinguish the parity interrupt from other NMIs, the PINT line is fed to U48 (address 1 bit 3) and can be tested by software.

2.4.5.5 Memory Control State Machine. A hard array logic device (HAL16R4 U28), set up as a state machine, drives the memory control. This device has four outputs equipped with a set of clocked flip-flops and four outputs that are direct combinations of the inputs. The AND of the terms on a line ORed with the AND of terms on other lines results in low-going outputs. This occurs either directly, on those outputs without registers, or after the clock on those outputs with registers.

The signal RASI- activates RAS- out of the AM2964B RAM address multiplexer. The signal XWAIT- puts the processor into a wait state. The signal MDEN- activates the motherboard memory system data buffer. The signal RMSEL- selects access to the ROMs. The signal RFSH- instructs the AM2964B address multiplexer to put out the refresh address. The signal RRAS- indicates that a refresh RAS is in progress. The signal SY- (used internally to the HAL) indicates refresh states. The signal SX- (used internally to the HAL) cuts off the wait state to the CPU after one cycle.

Table 2-8 gives the logic for the memory control state machine.

A timing diagram of the memory system, shown in Figure 2-3, indicates the major operations of the memory system.

Table 2-8 Memory Control State Machine Logic - HAL16R4

Output	MRD-	MWR-	RFRQ	XA19	XA18	RMX-	IORC-	AIOWC-	RASI-	XWAIT-	MDEN-	RMSEL-	RFSH-	RRAS-	SY-	SX-	Comment
	Input																
RASI-	L	.	.	L	L	.	.	.	.	.	.	.	H	.	H	.	Memory read
or	.	L	.	L	L	.	.	.	.	.	.	.	H	.	H	.	Memory write
or	.	.	.	.	.	.	.	.	.	.	.	.	L	L	.	.	Refresh
or	.	.	.	.	.	.	.	.	.	.	.	.	L	L	.	.	All other OR terms
XWAIT-	L	.	.	L	.	.	.	.	.	.	.	.	L	.	.	.	Refresh+read RF1,2,3
or	L	.	.	L	.	.	.	.	.	.	.	.	.	.	L	.	Refresh+read RF3,4
or	.	L	.	L	.	.	.	.	.	.	.	.	L	.	.	.	Refresh+write RF1,2,3
or	.	L	.	L	.	.	.	.	.	.	.	.	.	.	L	.	Refresh+write RF3,4
or	.	.	.	H	H	L	.	.	.	.	.	L	.	.	.	H	ROM read/write
or	.	.	.	.	.	.	L	.	.	.	.	.	.	.	.	H	I/O read
or	.	.	.	.	.	.	.	L	.	.	.	.	.	.	.	H	I/O write
MDEN-	.	.	.	.	.	.	.	.	L	.	.	.	H	.	H	.	RAM read/write
or	L	.	.	H	H	L	.	.	.	.	.	.	.	.	.	.	ROM read
or	.	L	.	H	H	L	.	.	.	.	.	.	.	.	.	.	ROM write
or	.	.	.	.	.	.	L	L	.	.	.	.	.	.	.	.	All other OR terms
RMSEL-	L	.	.	H	H	L	.	.	.	.	.	.	.	.	.	.	ROM read
or	.	L	.	H	H	L	.	.	.	.	.	.	.	.	.	.	ROM write
or	.	.	.	.	.	.	L	L	.	.	.	.	.	.	.	.	All other OR terms
The following four outputs have flip-flops:																	
RFSH-	H	H	H	.	.	.	.	.	.	.	.	.	.	.	H	.	Refresh RF1; no memory cycle
or	.	.	H	H	.	.	.	.	.	.	.	.	.	.	H	.	Refresh RF1; no RAM cycle
or	.	.	.	.	.	.	.	.	.	.	.	.	L	.	H	.	Refresh RF2,3
or	.	.	.	.	.	.	.	.	.	.	.	.	L	.	H	.	All other OR terms
RRAS-	.	.	.	.	.	.	.	.	.	.	.	.	L	.	.	.	Refresh RF2,3,4
or	.	.	.	.	.	.	.	.	.	.	.	.	L	.	.	.	All other OR terms
SY-	.	.	.	.	.	.	.	.	.	.	.	.	L	L	.	.	Refresh RF3,4
or	.	.	.	.	.	.	.	.	.	.	.	.	L	L	.	.	All other OR terms
SX-	L	.	.	H	H	L	.	.	.	.	.	.	.	.	.	.	ROM read wait cutoff
or	.	L	.	H	H	L	.	.	.	.	.	.	.	.	.	.	ROM write wait cutoff
or	.	.	.	.	.	.	L	.	.	.	.	.	.	.	.	.	I/O read wait cutoff
or	.	.	.	.	.	.	.	L	.	.	.	.	.	.	.	.	I/O write wait cutoff
or	.	.	.	.	.	.	.	L	.	.	.	.	.	.	.	.	All other OR terms

L = Low signal.
H = High signal.

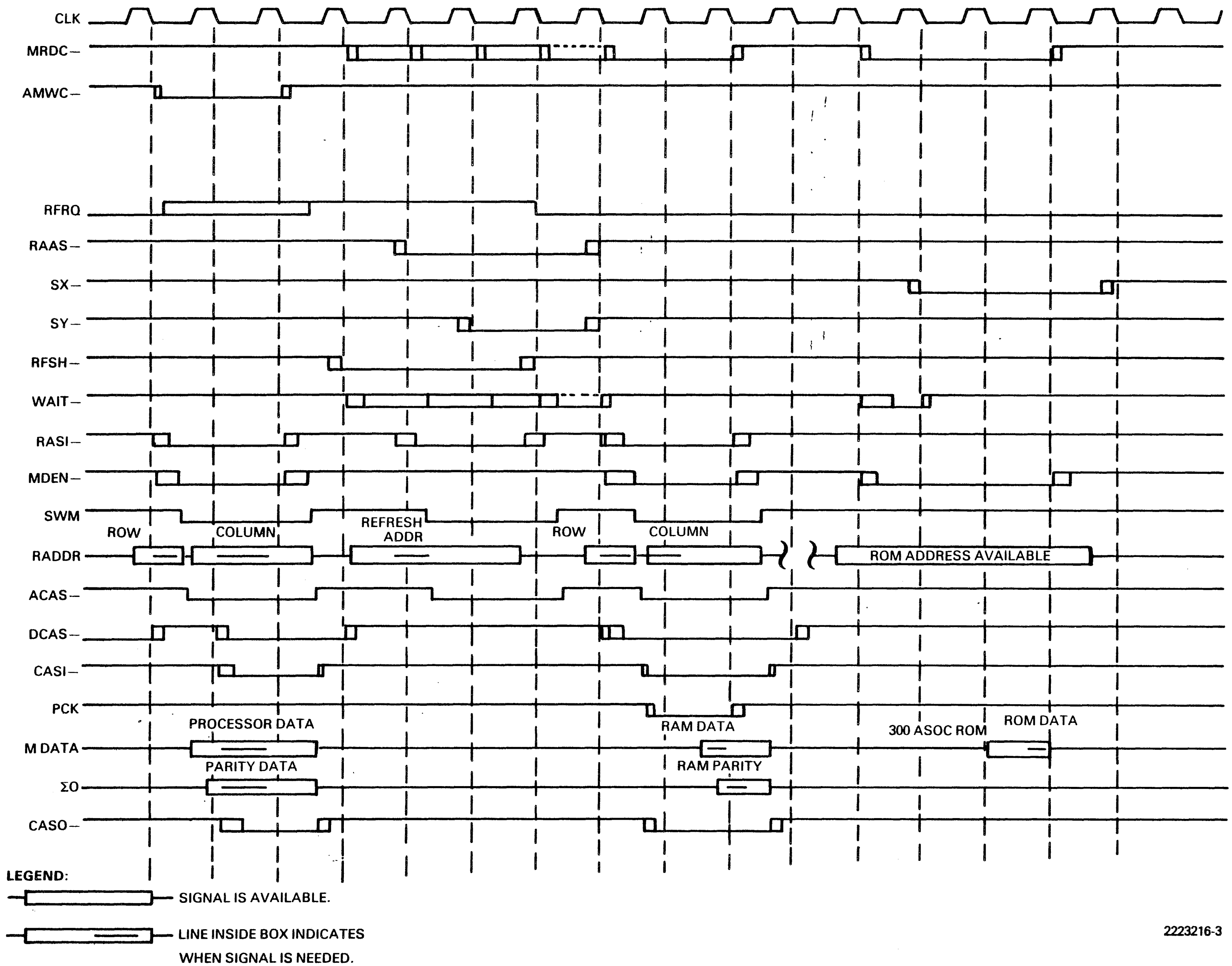


Figure 2-3 Memory System Timing Diagram

2.4.6 Floppy Disk Controller

The floppy disk controller (FDC) section contains a floppy disk controller IC (FD1793-02), a floppy disk support logic IC (WD1691), and a pulse delay IC (WD2143), all made by Western Digital. The FDC also has a voltage-controlled oscillator (VCO) and one-half of a 74LS221 one-shot. Two 2114 static RAMs, addressed by a CMOS 4040, act as a sector buffer, and a programmable array logic (PAL) IC decodes and controls operations. Miscellaneous logic handles signal timing and buffering.

The logic described in this section includes:

* Floppy disk controller IC
* Sector buffer
* Data write precompensation circuit
* Data separator
* Diskette drive interface

2.4.6.1 Floppy Disk Controller IC. The Western Digital FD1793-02 chip is the FDC IC. This IC does serial/parallel data conversion, locates sectors on the disk, seeks the diskette drive, and performs other high-level functions. A complete description of the FD1793-02 chip can be found in the literature available from Western Digital. The 1.0-MHz controller input clock provides the correct data rate for standard 5 1/4-in diskettes. Because U20 divides the clock down from 15.0 MHz, the duty cycle is 467 ns low, 533 ns high.

2.4.6.2 Sector Buffer. During read or write operations, data must be transmitted at a rate between 23 us per byte and 32 us per byte nominal (for double-density operation). A sector buffer, operating independently of the processor during a read or a write, ensures that the diskette drive performs properly. This buffer consists of:

* A 1K x 8 static RAM device
* A counter (to address the RAM sequentially)
* Control logic and a bus buffer (so that the CPU and the FDC can access the buffer)

Two bits (M0, M1) in latch U47 control the basic operating modes of the sector buffer. These four modes are as follows:

Latch U47 Bits M0	M1	Mode
1	1	FDC reads RAM and writes data to diskette.
0	1	FDC reads diskette and writes data to RAM.
0	0	CPU reads or writes RAM sequentially.
1	0	CPU reads or writes the FDC directly.

The counter that addresses the buffer increments automatically each time either the CPU or the FDC accesses the RAM. To set up a fixed starting address within the RAM, the CPU writes to the FDC sector register while the M0, M1 bits are set to 0, 0. This resets the address counter. The FDC is not affected because the CPU can access the FDC only in mode M0, M1.

The PAL provides the control logic for the sector buffer, aided by a flip-flop that provides a 1-us FDC clock-synchronized signal. The PAL uses this signal, derived from the FDC data request (DRQ) line, to generate the read or write command for the FDC when the sector buffer is in modes 1, 1 or 1, 0. The FDC activates the DRQ line when a sector write requires a byte or when a byte is ready in a sector read.

This control logic and the CPU generate other signals to control the RAM and the counter. These signals are given in Table 2-9. The timing diagram in Figure 2-4 defines the usage of these signals. When the logical AND of terms from one row is ORed with the AND of terms from another row, the output goes low when the result is true.

Table 2-9 Programming for the HAL10L8 Device

Output	IORQ-	XA1	XA0	M1	M0	DRQD	IORC-	AIOWC-	FLCS	DEN-	Comment
	Input										
YA0	.	.	L	L	H	.	.	.	L	.	CPU <---> FDC Mode 0,1
or	.	.	.	.	.	.	L	L	.	.	(Unused)
YA1	.	L	.	L	H	.	.	.	L	.	CPU <---> FDC Mode 0,1
or	.	.	.	.	.	.	L	L	.	.	(Unused)
FRD-	.	.	.	L	H	.	L	.	L	.	CPU <-- FDC Mode 0,1
or	.	.	.	H	L	H	.	.	.	.	FDC --> RAM Mode 1,0
FWR-	.	.	.	L	H	.	.	L	L	.	CPU --> FDC Mode 0,1
or	.	.	.	H	H	H	.	.	.	.	FDC <-- RAM Mode 1,1
RWE-	.	L	.	L	L	.	.	L	L	.	CPU --> RAM Mode 0,0
or	.	.	.	H	L	H	.	.	.	.	FDC --> RAM Mode 1,0
RCS-	L	L	.	L	L	.	.	.	L	.	CPU <--> RAM Mode 0,0
or	.	.	.	H	.	H	.	.	.	.	FDC <--> RAM Mode 1,X
RRST-	.	H	L	L	L	.	.	L	L	.	Reset counter Mode 0,0
or	.	.	.	.	.	.	L	L	.	.	(Unused)
FDEN-	L	L	.	L	L	.	.	.	L	.	CPU <--> RAM Mode 0,0
or	.	.	.	L	H	.	.	.	L	L	CPU <--> FDC Mode 0,1

Legend:
- L = Low signal.
- H = High signal.

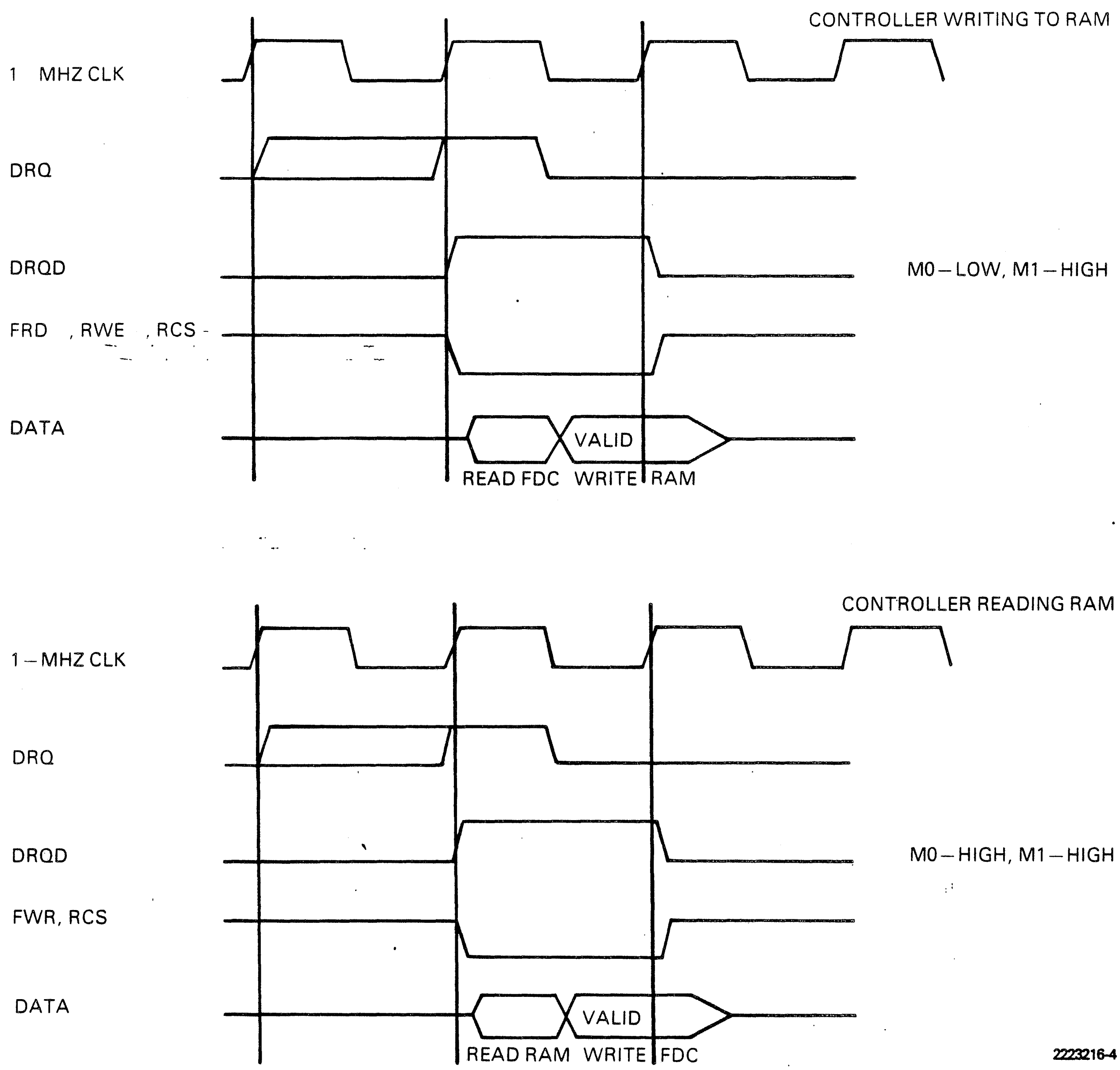


Figure 2-4 Floppy Disk Timing Diagrams

2.4.6.3 Write Precompensation Circuit. Using modified frequency modulation (MFM) to write certain double-density data patterns on magnetic media causes a "bit shift", requiring disk write precompensation. Compensating for the bit shift prevents the read data transitions from moving outside the detection range of the read circuitry. As track length shortens toward the center of the disk, data bits are stored closer together, so the bit shift problem gets worse. The ideal compensation gradually adjusts the write hardware as the track number increases. However, a compromise solution produces nearly the same results. The precompensation is turned off while the head is over the outer half of the disk, then turned on when the head is over the inner half of the disk. Disk drives can have either 40 or 80 tracks, so the software checks the type of drive installed, then determines the halfway point. For this reason, U47 (rather than the FDC) controls the TG43 signal. (Halfway point for an 8-in diskette = TG43 - track number greater than 43.)

The write precompensation and data separator circuits are controlled by U14, R17, R18, and R19 on the motherboard. When the RDDATA- line (pin 11 of U14) is high, it forces the PU and PD- outputs from the WD1691 to a tristate condition. R17 adjusts the PUMP line (pins 13/14 of U14) voltage to 1.4 Vdc. R18 generates a square wave of 2.0 MHz ± 5.0 percent from the VCO (pin 16 of U14). The pulsewidth (monitored from pin 5 of U14) should be 750 ns, giving a write pulse width of 187.5 ns. The waveform is visible only when the computer is writing data to a diskette.

R19 controls the write pulsewidth through U15 (the WD2143 IC), determining the amount of precompensating bit shift. The precompensation pulsewidth (monitored from pin 1 of U15 during a write operation) should be set to approximately 200 ns.

The FDC signals EARLY and LATE control the direction of bit shift. These signals cause WD1691 to select the appropriate tap along the WD2143 (adjustable delay line) for the bit pattern being written. If precompensation is not needed on outer tracks, the TG43 signal inhibits the precompensation process.

Because single-density frequency modulation (FM) encoded data does not require precompensation, the FD1691 also disables the precompensation when the double-density enable signal (DDEN-) is inactive (high).

2.4.6.4 Data Separator. The data separator is composed of two parts: clock recovery and separation of the data from the clock. The actual separation of data from clock signals takes place in the FD1793-02 FDC. The WD1691 contains the digital circuits necessary to implement a phase-locked loop (PLL), the VCO is a 74LS628 chip, and external components provide the loop filter. The one-shot U29 shortens and stabilizes the pulsewidth of the incoming read pulses so that the PLL and data recovery operations operate properly during the lockup interval.

The PLL provides a continuous clock locked in a specific phase relationship with transitions in the incoming data. For this system, the falling edge of the RDDATA- signal should be nearly centered on the high or low pulse of the RCLK signal.

When the PLL is adjusted correctly, it locks to an incoming pulse train in a frequency range from 217 kHz to 294 kHz (± 15 percent) within 150 us. The pulses should be low-going, 2 us maximum applied to the RDDATA- input (P9 pin 30), and the DDEN- line must be low.

Because of the analog nature of the PLL circuits, a linear regulator governs the power-supply voltage to the VCO and the loop filter. The regulator prevents digital noise on the 5-V supply from interfering with the PLL operation.

The data separator works with either single-density (FM) or double-density (MFM) data. The choice is controlled by the DDEN- line.

2.4.6.5 Diskette Drive Interface. The diskette drives communicate through a series of buffers and receivers. Low-impedance ribbon cables connect the controller to the drive. P9 connects the internal diskette drives, and P13 connects the external drives. All signals driven by the controller (except for the SID1- signal) have separate drivers for each connector. The receivers with their terminating pullup resistors are shared between the two connectors.

Connector P9 interfaces with a 34-conductor ribbon cable that has two 34-pin, card-edge connectors (one for each of the diskette drives that can be mounted inside the system unit chassis). There is always one diskette drive installed in the system unit, mounted on the left side (as viewed by a user). This drive should be strapped for SELECT on pin 10 (drive 0). When only one drive is installed, the select line and all common lines except pin 32 (side select) should be terminated at the drive.

If another drive is installed internally, it should be strapped for SELECT on pin 12 (drive 1) with only the select line terminated. With two drives installed, the terminating resistor must be installed on the right-hand drive (drive 1) only.

NOTE

> The floppy disk controller and individual diskette drive logic signals assign drives using the convention of: DRIVE 0, DRIVE 1, DRIVE 2, and DRIVE 3 (for a four-drive system). The diagnostics diskette uses the convention: DRIVE 1, DRIVE 2, DRIVE 3, and DRIVE 4 for a four-drive system. Operating systems may use yet another convention, such as DRIVE A, DRIVE B, DRIVE C, and DRIVE D. Be sure to use the correct drive designator.

Connector P13 interfaces with a 40-wire ribbon cable ending in a 37-pin, D-type connector. The user mounts the mate to this connector on the back panel of the system unit chassis. When external drives are installed, all lines used must terminate at the external drive.

All diskette drives must be of the same type. That is, all must be either 320K-byte drives (double-sided, 48 tracks per inch [tpi]) or all must be 640K-byte drives (double-sided, 96 tpi). A jumper from E1 to E2 selects 320K-byte drives; a jumper from E3 to E4 selects 640K-byte drives. The absence of a jumper selects 160K-byte drives. A jumper can be on either E1-E2 or E3-E4, but not both.

The diskette drives do not need head-load solenoids for proper operation. However, if the drives are equipped with head-load solenoids, they should be strapped for head load with the motor on.

The signals STEP, DIRC, WG, and WDOUT are buffered by the 74LS244 in order to drive the two standard 7416 loads. This buffer is necessary because the FD1793-02 and the WD1691 can drive only one TTL load. The input signals WRITEPROT-, INDEX-, TRK00-, and RDDATA- are buffered by the 74LS244, providing more static protection than the MOS-device inputs, and a small amount of hysteresis.

To install external diskette drives, a short cable assembly links the motherboard connector P13 with a 37-pin, d-type connector on the back of the system unit chassis. Section 5 contains the wiring assembly diagrams for this cable. (External diskette drives require an external power source.)

Table 2-10 gives the pin-outs for the internal diskette drive connector on the motherboard. Table 2-11 gives the pin-outs for the external diskette drive connector on the motherboard. D-type connector pin numbers are given in parentheses.

Table 2-10 Internal Diskette Drive Connector Pin-Out

Signal	Return	Signal Name	Source	Function
2	1	--	NC*	--
4	3	--	NC	--
6	5	--	NC	--
8	7	INDEX-	Drive	Indicates index hole
10	9	SELECT 1-	System	Drive select 1
12	11	SELECT 2-	System	Drive select 2
14	13	--	NC	--
16	15	MOTOR ON-	System	Drive motors ON
18	17	DIRECTION-	System	Step IN/OUT direction
20	19	STEP-	System	Step IN/OUT command
22	21	WRITE DATA-	System	Serial data to drive
24	23	WRITE GATE-	System	Enables writing to drive when signal is low
26	25	TRACK 00-	Drive	Indicates head is over track 00 when signal is low
28	27	WRITE PROT-	Drive	Indicates diskette is write-protected
30	29	READ DATA-	Drive	Serial data from drive
32	31	SIDE 1-	System	Side select (0,1 = high, low)
34	33	--	NC	--

* NC = Not connected.

Table 2-11 External Diskette Drive Connector Pin-Out

Signal	Return	Signal Name	Source	Function
2 (1)	1(20)	--	NC*	--
4 (2)	3(21)	--	NC	--
6 (3)	5(22)	--	NC	--
8 (4)	7(23)	--	NC	--
10 (5)	9(24)	--	NC	--
12 (6)	11(25)	INDEX-	Drive	Indicates index hole
14 (7)	13(26)	MOTOR 3-	System	Drive motor 3 enable
16 (8)	15(27)	SELECT 4-	System	Drive select 4
18 (9)	17(28)	SELECT 3-	System	Drive select 3
20(10)	19(29)	MOTOR 4-	System	Drive motor 4 enable
22(11)	21(30)	DIRECTION-	System	Step IN/OUT direction
24(12)	23(31)	STEP-	System	Step IN/OUT command
26(13)	25(32)	WRITE DATA-	System	Serial data to drive
28(14)	27(33)	WRITE GATE-	System	Enables write when low
30(15)	29(34)	TRACK 00-	Drive	Indicates head is over track 00 when low
32(16)	31(35)	WRITE PROT-	Drive	Indicates diskette is write-protected
34(17)	33(36)	READ DATA-	Drive	Serial data from drive
36(18)	35(37)	SIDE 1-	System	Side select (0 = high)
38(19)	37	--	NC	--
40	39	--	NC	--

* NC = not connected.

2.4.6.6 Diskette Drive. The Texas Instruments Professional Computer is equipped with one 5 1/4-in, double-sided, diskette drive. The self-contained unit consists of a spindle drive, a head positioner, and a read-write-erase system.

Plastic guides help to position the diskette inside the diskette slot. After you insert the diskette and close the access door, three things happen: the diskette clamps to the drive hub; a 500-ms delay begins, and the servo-controlled drive motor starts.

The head positioner is a 4-phase stepper-motor and band assembly with some related electronics. It moves the head (using one-step rotation to cause a one-track linear movement) to the proper track of the diskette.

The following sensor systems are built into the unit.

* The track 00 sensor. This switch determines that the head/carriage system is at track 00.

* The index sensor. When the phototransistor sees the LED light source through an index hole, it sends out a signal.

* The write-protect sensor. When this switch finds a write-protect tab applied to a diskette, it disables the write head.

The diskette drive reads and writes digital data using MFM. The write operation records a 0.33-mm (0.013 in) data track, which is later tunnel-erased to 0.30 mm (0.012 in). The track-to- track access time is 6 ms. The drive speed is 300 rpm.

Table 2-12 gives the specifications for the diskette drive.

Table 2-12 Diskette Drive Specifications

Physical Dimensions:

Height	85.85 mm (3.38 in)
Width	149.10 mm (5.87 in)
Depth	203.20 mm (8.00 in)
Weight	2.04 kg (4.50 lb)

Environmental Parameters:

	Operating	Storage
Temperature	10 °C to 40 °C (50 °F to 104 °F)	-40 °C to 65 °C (-40 °F to 149 °F)
Relative Humidity (@ 40 °F wet-bulb temperature, no condensation	20 % to 80 %	5 % to 95 %
Altitude	Mean sea level to 10 000 ft	Mean sea level to 45 000 ft

Power Requirements

Voltage	Current
+5 Vdc (+/- 0.25 V)	600 mA
+12 Vdc (+/- 0.6 V)	900 mA

2.4.7 CRT Controller Board

The CRT controller board drives either a monochrome analog or a color TTL display and makes the Texas Instruments Professional Computer a complete alphanumeric and raster graphics system.

As a stand-alone option, the controller board provides one page of high-resolution (80 columns x 25 lines) alphanumeric display. This board also supports the optional graphics video controller piggyback board, which is described in Section 3.

The system makes no physical distinction between color and monochrome; the board supports output in either eight-level gray scale or eight-color RGB (red, green, blue). Color is determined by the monitor used. Refer to Section 6, drawing 2223011, for logic diagrams.

Figure 2-5 is a block diagram of the alphanumeric CRT controller board.

Table 2-13 lists the video ac parameters.

Figure 2-6 shows the timing diagram for the Alphanumerics State Machine PAL.

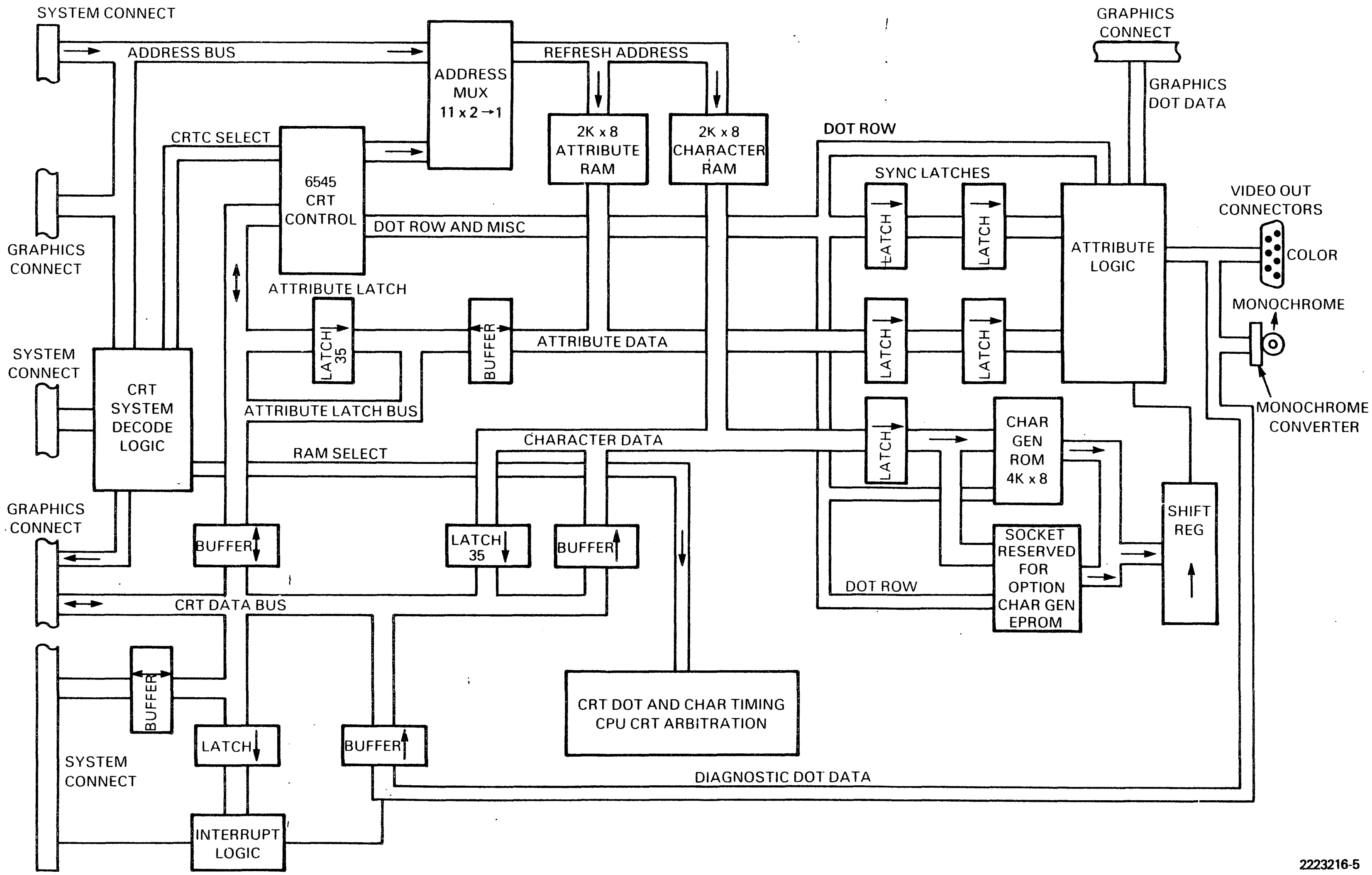


Figure 2-5 Alphanumeric CRT Controller Board Block Diagram

Table 2-13 Video AC Parameters

#*	Parameter	Value	Value**	Tolerance
A	Video dot frequency	18.000 MHz	--	1 %
B	Video dot pulsewidth	55.55 ns	--	1 %
C	Character block horizontal	9 dots	--	--
D	Character block vertical	12 dots	14 scan lines	--
E	Number of character lines	25 rows	--	--
F	Characters/character line	80 columns	--	--
G	Number of active scan lines	300	350	--
H	Total scan lines	320	385	--
J	Vertical sync width	0.156 ms	0.156 ms	1 %
K	Vsync front porch	0 ms	0 ms	1 %
L	Vsync back porch	0.884 ms	1.664 ms	1 %
M	Vertical blanking interval	1.040 ms	1.82 ms	1 %
N	Active vertical display time	15.60 ms	18.20 ms	1 %
	Total vertical time	16.63 ms	20.02 ms	1 %
Q	Vertical rate	60.10 Hz	49.95 Hz	2 Hz
	Hsync width	4.50 us	--	1 %
S	Hsync front porch	2.00 us	--	1 %
T	Hsync back porch	5.50 us	--	1 %
U	Horizontal blanking interval	12.00 us	--	1 %
V	Active horizontal display time	39.99 us	--	1 %
W	Total horizontal time	51.99 us	--	1 %
X	Horizontal rate	19231 Hz	--	100 Hz

* Letters refer to areas on the timing diagram in the next figure.

** These values reflect the vertical timing adjustments for 50-Hz refresh.

CAUTION

50-Hz operation can be used only in areas that run on 50-Hz line frequency. Using 50-Hz operation in any other area can damage your computer. To select 50-Hz operation, jumper pins E5-E6 on the motherboard.

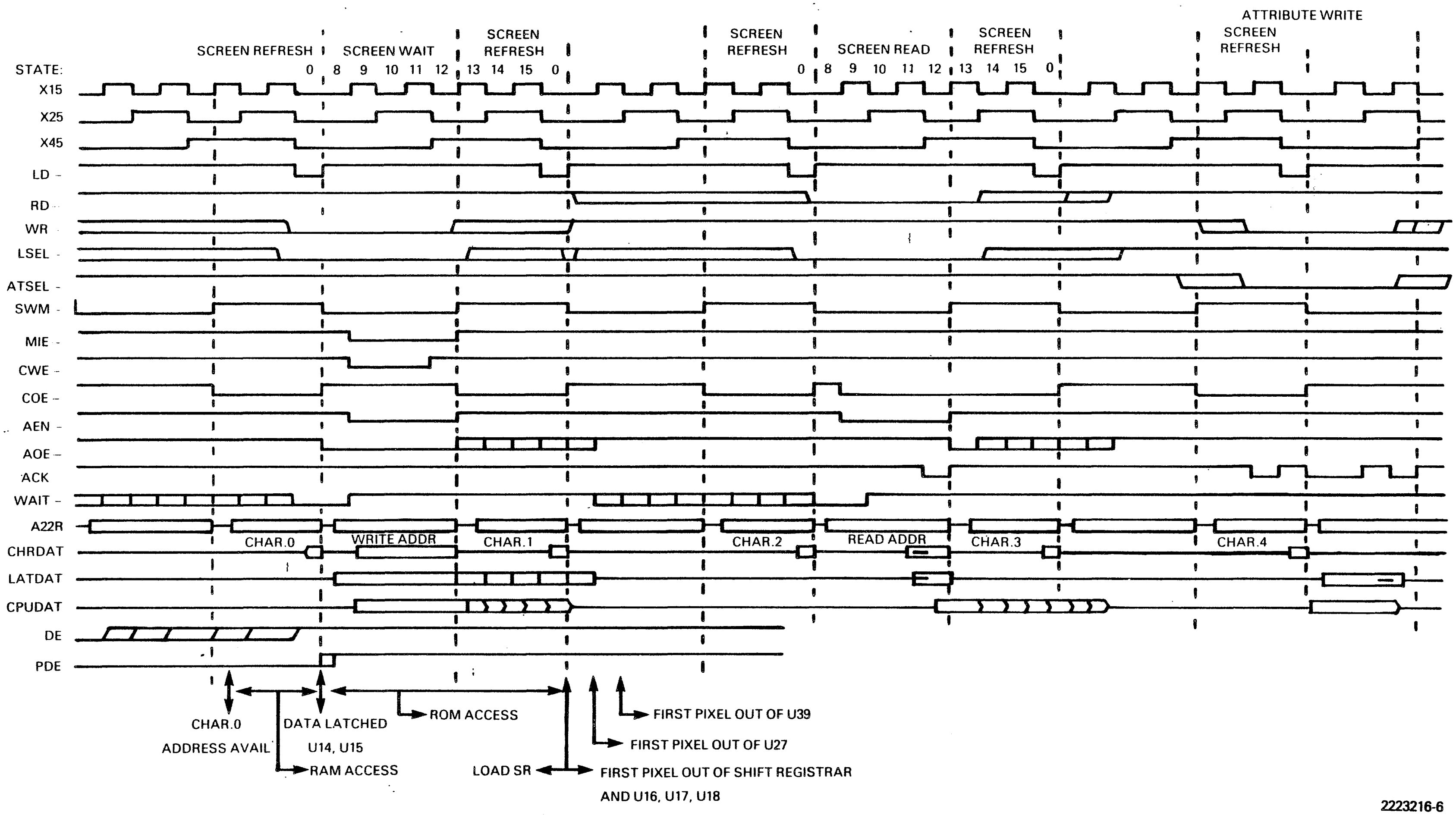


Figure 2-6 Alphanumerics State Machine Timing Diagram

The CRT controller board features described in the following paragraphs include:

* Display characteristics
* Character attributes
* Character sets
* Cursor
* Scrolling
* Video connector
* CRT controller IC
* CRT screen/CPU arbitration logic
* CRT address decode logic
* Character sets and attribute logic
* CRT interrupt logic
* Diagnostic loopback

2.4.7.1 Display Characteristics. The display characteristics are as follows:

* A 7 x 9 character in a 9 x 12 image cell
* Twenty-five lines of 80 characters
* A resolution of 720 pixels horizontally x 300 pixels vertically
* A horizontal scan rate of 19 200 lines per second
* A vertical scan rate of 60 (50 frames per second)
* A dot rate of 18.0000 MHz

NOTE

The horizontal scan rate is an important consideration. Many monitors available today have a horizontal scan rate of 15 750. Only a monitor having a horizontal scan rate of 19 200 lines per second can operate with the Texas Instruments Professional Computer.

2.4.7.2 Character Attributes. The controller's video memory is organized as 2K bytes x 16 bits. The first 8 bits convey character information. The second 8 bits select the following attributes on a character basis:

* Bit 0, intensity level 1 (blue)
* Bit 1, intensity level 2 (red)
* Bit 2, intensity level 4 (green)
* Bit 3, character enable
* Bit 4, reverse
* Bit 5, underline
* Bit 6, blink
* Bit 7, alternate character set

NOTE

> The three intensity bits (bit 0 through bit 2) determine the gray scale intensity level and the RGB outputs for color. Thus, hi/norm video in monochrome is handled by a one-of-eight intensity select instead of a high-intensity bit.

To access the attributes, the software writes the attribute values into an attribute latch. The attribute value is then assigned to the character each time that character is written to the screen (until a screen read is done).

When any character on the screen is read, its attributes are copied to the attribute latch. These values are then read by a subsequent latch read operation.

Handling the attributes by this method ensures that, in block moves (moving data from one screen area to another), the characters retain their attributes.

2.4.7.3 Character Sets. The video controller contains a 4K character generator ROM, which contributes 256 characters. Use the socket provided to add an optional 2K or 4K ROM/EPROM and expand the character set to the maximum 512 characters. Attribute bit 7 selects the expanded character set.

Refer to subparagrahph 2.4.8.4 for more information on the character ROM.

2.4.7.4 Cursor. Programming can change the cursor appearance. The possibilities include blinking, non-blinking, block, underline and reverse-video. Hardware handles the cursor display through a special set of registers in the controller. Using these registers, the software can position the cursor anywhere on the screen (or off the screen if no visible cursor is desired).

2.4.7.5 Scrolling. The hardware maintains a screen start register that supports character line scrolling in four directions. The software determines the need for a scroll, then changes the value of this register by one line. The screen appears to jump by one line. The scrolling operation always affects all of the screen. It is not possible to scroll one region without affecting another.

Because the controller contains only 2K bytes of screen memory, scrolling results in a "wrap"; the original top line of screen contents moves to the bottom of the screen. Therefore, the software must clear the top line of the screen (or bottom) before the scroll-up (or -down) operation. To simplify programming of the line clear operation, the 2K bytes of memory is phantomed over a 4K-byte address space.

Status lines must be implemented in software. That is, during scroll operations, the status line must be moved to its new memory position before writing. The screen start register changes the screen-to-memory correspondence.

2.4.7.6 Video Connector. The video connector located on the rear edge of the PWB is a standard, 9-pin, female, D-type connector. This connector is for a color display unit. The signals available on this connector are given in Table 2-14. All signals are at standard TTL levels.

Table 2-14 Color Video Connector Pin-Out

Pin	Function
1	Ground
2	Logic ground
3	Red video
4	Green video
5	Blue video
6	Logic ground
7	NC (no connection)
8	Horizontal drive (NEGATIVE TRUE)
9	Vertical drive (POSITIVE TRUE)

The other video connector, on the lower rear edge of the PWB, is a standard RCA phono jack. This connector is for a monochrome display. The signal available at this connector is a composite type, 1 V peak-to-peak, 75-ohm load.

2.4.8 CRT Controller IC

The CRTC IC (6545A-1) contains the logic for:

* Generating the horizontal and vertical synchronizing signals
* Blanking display during retrace
* Addressing screen memory during screen refresh
* Cursor coincidence
* Starting screen display registers for use in scrolling

The CRTC contains eighteen registers that must be appropriately set before board operation begins. To access these registers, the CPU first writes the address of the register to be accessed into the CRTC address register. Then information can be written to that register. When writing to or reading from (where appropriate) the data register, the information is accessed by the address latched in the address register.

Table 2-15 shows how to program these registers, using the signals chip select (CS), register select (RS), and read/write (R/W-). Assume the following conditions:

* A character rate (SWM-) of 2.0 MHz
* 12 lines per character block
* 25 rows on the display
* 24 character times of horizontal blanking (12.0 us)
* 20 line times of vertical blanking (1.04 ms)

For more detailed programming information, refer to The Synercom Data Book.

Table 2-15 CRTC Programming Values

Signal Name			Register Address	Register Name	Refresh Rate Value	
CS-	RS	R/W-			60 Hz	50 Hz
H	X	X	--	No register selected	--	--
L	L	L	--	Set address register	--	--
L	L	H	--	Set status register	--	--
L	H	L	0	Horizontal total characters minus one	103	103
L	H	L	1	Horizontal displayed characters	80	80
L	H	L	2	Horizontal sync position	84	84
L	H	L	3	VSYNC width, HSYNC width	39H	59H
L	H	L	4	Vertical total rows minus 1	24	31
L	H	L	5	Vertical adjust lines	20	00
L	H	L	6	Vertical displayed rows	25	25
L	H	L	7	Vertical sync position	25	28
L	H	L	8	Mode control	00H	00H
L	H	L	9	Scan lines per row minus 1	11	11
L	H	L	10	Cursor start line and BLINK	40H	40H
L	H	L	11	Cursor end line	11	11
L	H	L	12	Display start address high	00H	00H
L	H	L	13	Display start address low	00H	00H
L	H	X	14	Cursor position address high	00H	00H
L	H	X	15	Cursor position address low	00H	00H
L	H	H	16	Light pen position address high	--	--
L	H	H	17	Light pen position address low	--	--

Legend:
- H = High signal.
- L = Low signal.
- X = Don't care.

?.4.8.1 CRT Screen/CPU Arbitration Logic Subsystem. The CRT controller arbitration logic gives the programmer free access to the RT display. There is little overhead time caused by arbitration conflicts, because the refresh memory and its control logic allow two complete memory cycles between each character displayed on the screen. One cycle accesses the character for display; the CPU uses the other cycle for read or write operations. Therefore, the CPU waits less than two display-character times for memory access. Because a character time is 500.8 ns and the CPU clock is 200 ns, a synchronization delay can occur. The total time for a worst-case CPU access is 1.0 us. The usual access time is 600 ns (3 to 0 wait states).

The logic that generates this arbitration scheme includes a counter (which also counts the nine dots per character), a PAL (which has internal registers and gets feedback from the outputs), and a small alphanumerics state machine (which provides RAM buffer control, control outputs for the RAM, and the wait state control for the CPU). The counter uses inputs to the PAL to identify the state within the display cycle of the state machine. The internal PAL registers define other states used during the CPU read and write cycles. To define the CPU cycle type being executed, the PAL uses the inputs RD-, WR-, CSEL-(character select), and ATSEL-(attribute select).

The outputs from the PAL are:

* COE-, the RAM output enable
* CWE-, the RAM write enable
* AEN-, the attribute bus buffer enable
* AOE-, the attribute latch output enable
* ACK-, the attribute latch clock
* MIE-, the character bus input buffer enable
* SWM-, the signal that switches the RAM address multiplexer from the CRTC to the CPU
* WAIT-, the CPU wait control line

The counter (U24, a 74LS163) goes through states 8,9,10,11,12,13,14,15,0, and repeat.

Latch U10 is included because the window (when read data from the video RAM is available) is rather short. This latch captures and holds the data for the CPU until the end of the CPU read cycle. The ACK line, which clocks the attribute latch, clocks this latch when read data is available from the RAM. The output is enabled onto the local bus by a combination of CSEL- and RD-.

The CRT arbitration PAL programming is given in Table 2-16. In the "comment" column, the states generated by the AND of inputs are listed according to the counter state number. When the logical AND of terms from one row is ORed with the AND of terms from another row, the output goes low when the result is true.

Refer to Figure 2-6 for an illustration of the timing produced for typical cycles by the alphanumerics state machine.

Table 2-16 Alphanumerics State Machine PAL

Output	X1	X2	X4	LD-	RD-	WR-	CSEL-	ATSEL-	SWMUX	MIE-	CWE-	COE-	AEN-	ACK-	AOE-	WAIT-	Comment
	Input																
SWMUX	.	.	L	.	.	.	.	.	.	.	.	.	.	.	.	.	S8,9,10,11,12 X4 delayed
or	.	.	L	.	.	.	.	.	.	.	.	.	.	.	.	.	All other terms
MIE-	L	L	L	H	.	L	L	.	.	.	.	.	.	.	L	L	S9 RAM write begins
or	.	.	.	.	.	.	.	.	.	.	L	.	.	.	.	.	S10,11,12 RAM write continues
or	.	.	.	.	.	.	.	.	.	.	L	.	.	.	.	.	All other terms
CWE-	L	L	L	H	.	L	L	.	.	.	.	.	.	.	L	L	S9 RAM write begins
or	H	L	L	H	.	L	L	.	.	.	L	.	.	.	.	.	S10 RAM write continues
or	L	H	L	H	.	L	L	.	.	.	L	.	.	.	.	.	S11 RAM write continues
or	.	.	H	L	.	.	.	.	.	.	.	.	.	.	.	.	All other terms inactive
COE-	.	.	H	.	.	.	.	.	.	.	.	.	.	.	.	.	S13,14,15,0 screen refresh
or	.	L	L	H	L	.	L	.	.	.	.	.	.	H	.	L	S9,10 RAM read
or	.	.	.	.	L	.	L	.	.	.	.	.	L	.	.	.	S10,11,12 RAM read continues
or	.	.	H	L	.	.	.	.	.	.	.	.	.	.	.	.	All other terms inactive
AEN-	L	L	L	H	.	L	L	.	.	.	.	.	.	.	L	L	S9 RAM write begins
or	.	.	.	.	.	.	.	.	.	.	L	.	.	.	.	.	S10,11,12 RAM write continues
or	.	L	L	H	L	.	L	.	.	.	.	.	.	H	.	L	S9,10 RAM read
or	.	H	L	H	L	.	L	.	.	.	.	.	L	.	.	.	S11,12 RAM read
or	.	.	H	L	.	.	.	.	.	.	.	.	.	.	.	.	All other terms inactive
ACK-	H	H	L	H	L	.	L	.	.	.	.	.	L	.	.	.	S12 RAM read
or	L	.	.	.	.	L	.	L	.	.	.	.	.	.	.	.	Write attribute latch
or	.	.	H	L	.	.	.	.	.	.	.	.	.	.	.	.	All other terms inactive
AOE-	L	L	L	L	.	L	L	.	.	.	.	.	.	.	H	L	S8 RAM write
or	.	.	.	.	.	L	L	.	.	.	.	.	.	.	L	.	S9 till not write
or	.	.	.	.	L	.	.	L	.	.	.	.	.	.	.	.	Read attribute latch
or	.	.	.	.	L	.	L	.	.	.	.	.	.	L	.	.	S13 till not read
or	.	.	.	.	L	.	L	.	.	.	.	.	.	.	L	.	S13 till not read
WAIT-	.	.	.	.	.	L	L	.	.	.	.	.	.	.	H	.	RAM write before S9
or	.	.	.	.	L	.	L	.	.	.	.	.	H	H	.	.	RAM read before S9
or	.	.	H	L	.	.	.	.	.	.	.	.	.	.	.	.	All other terms inactive

Legend:
L = Low signal.
H = High signal.

2.4.8.2 CRT Address Decode Logic. The CRT controller board handles both alphanumeric- and graphics- address decode for the CRT subsystem. All of the screen data is mapped into the processor memory address space including the assorted latches and I/O ports.

The decoding is done with three ICs: a HAL10L8 PAL, one-half of a 74LS20, and a 74LS155 decoder. The PAL produces the following signals:

* ZBEN-, the master expansion bus buffer enable
* XBEN-, the secondary bus buffer enable
* RD-, a decoded and buffered read control
* WR-, a buffered and decoded write control
* GSEL-, the graphics screen memory select
* CSEL-, the alphanumerics screen memory select
* CR/AT-, selects one half of the 74LS155 (which decodes the CRTC and the attribute latch)
* XSEL-, selects the other half of the 74LS155 (which decodes the graphics latch and the miscellaneous input buffer)

The XBEN- signal develops an enable clock for the CRTC by inverting and delaying the signal that provides the required setup time (90 ns) for the 6545a-1 CRTC. The CRTE (CRT enable) signal has a pulsewidth greater than 266 ns, satisfying the requirement of the CRTC. The other setup and hold times are easily met.

The 74LS155 decodes the following signals:

* ATSEL-, the attribute latch select
* CRTSEL-, the CRTC chip select
* LAT-

LAT- combines with WR- and clocks the interrupt enable and screen enable latches. The other half of the 74LS155 decodes the three graphics board latches and the buffer enable for miscellaneous inputs. The address space that each of these devices occupies is given in Table 2-17.

Table 2-17 CRT System Memory Map

Address		Device
C0000-C7FFF		Graphics RAM Bank A
C8000-CFFFF		Graphics RAM Bank B
D0000-D7FFF		Graphics RAM Bank C
D8000-DDFFF		Unusable
DE000-DE7FF		Active character memory
DE800-DEFFF		Phantom character memory
DF000	bit 0	Misc input buffer, blue feedback, read only
DF000	bit 1	Misc input buffer, red feedback, read only
DF000	bit 2	Misc input buffer, green feedback, read only
DF000	bit 3	Misc input buffer, interrupt pending, read only
DF010		Graphics blue palette latch, write only
DF020		Graphics green palette latch, write only
DF030		Graphics red palette latch, write only
DF800		Attribute latch
DF810		CRTC address register, write only
DF811		CRTC status register, read only
DF812		CRTC registers write access, write only
DF813		CRTC registers read access, read only
DF820	bit 7	Miscellaneous output latch, interrupt enable
DF820	bit 6	Miscellaneous output latch, alphanumerics screen enable

PAL coding is given in Table 2-18. When the logical AND of terms from one row is ORed with the AND of terms from another row, the output goes low when the result is true.

Table 2-18 Alphanumeric Decoding PAL

Output:	MRDC-	AMWRC-	A1516-	A19	A18	A17	A14	A13	A12	A11	Comment
ZBEN-	L	.	.	H	H	L	.	.	.	.	CRT space read
or	.	L	.	H	H	L	.	.	.	.	CRT space write
XBEN-	L	.	L	H	H	L	H	H	H	H	CRTC/ATT read
or	.	L	L	H	H	L	H	H	H	H	CRTC/ATT write
RD-	L	.	.	H	H	L	.	.	.	.	CRT space read
or	L	L	.	.	.	.	.	.	.	.	(Inactive term)
WR-	.	L	.	H	H	L	.	.	.	.	CRT space write
or	L	L	.	.	.	.	.	.	.	.	(Inactive term)
GSEL-	.	.	H	H	H	L	.	.	.	.	Graphic access
or	L	L	.	.	.	.	.	.	.	.	(Inactive term)
CSEL-	.	.	L	H	H	L	H	H	L	.	Character access
or	L	L	.	.	.	.	.	.	.	.	(Inactive term)
CR/AT-	.	.	L	H	H	L	H	H	H	H	CRTC/ATT access
or	L	L	.	.	.	.	.	.	.	.	(Inactive term)
XSEL-	.	L	L	H	H	L	H	H	H	L	Extra I/O write
or	L	.	L	H	H	L	H	H	H	L	Extra I/O read

Legend:
L = Low signal.
H = High signal.

2.4.8.3 Character Set and Attribute Logic. Two 74LS374s (U14, U15) latch the RAM output (both character and attribute) at the end of each screen refresh access cycle. This allows a full character cycle time (500.8 ns) to access the character ROM and EPROM and set up the dot shift register. The required ROM access time is 452.8 ns. So that the character set can include the ability for block graphics, bit 7 out of the ROMs indicates that the leftmost and rightmost character dots are to be copied to the left and right character-cell border dots. The character ROMs should be programmed with active-low data; that is, when a dot is to appear, the ROM should be programmed with a zero.

Figure 2-7 shows some sample characters. The reverse video block and the cursor affect the entire 9 x 12 character cell; the underline appears on row 11. The descenders of lowercase letters should drop only one dot line below the level of the other characters so that the underline, cursor, and reverse video will appear in an acceptable form.

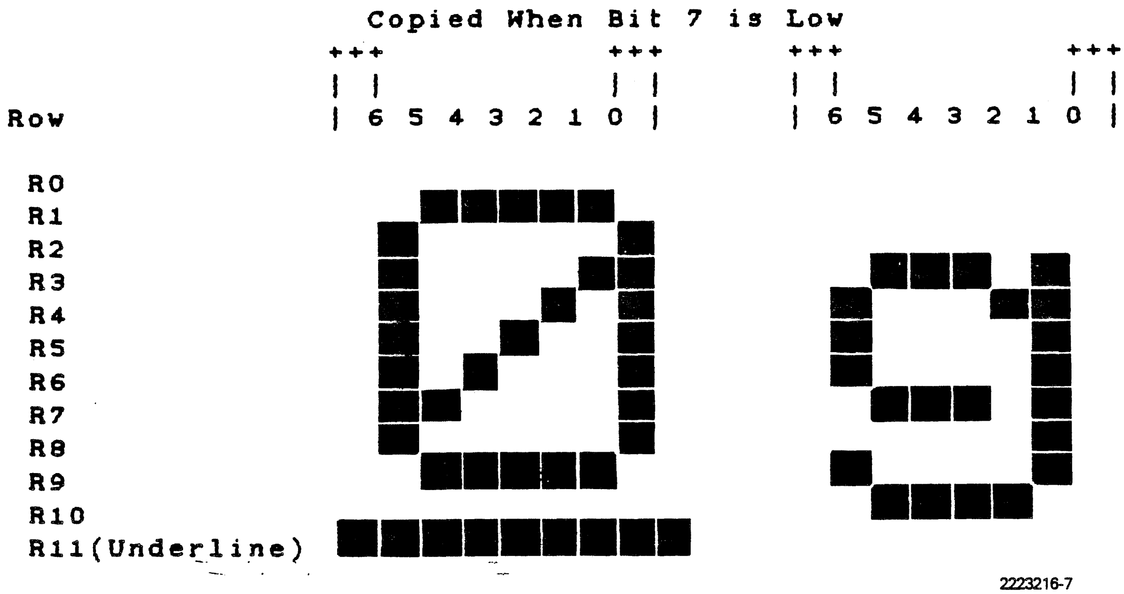


Figure 2-7 Sample Character Font Definition

2.4.8.4 Generating a Character ROM. To generate a character ROM (or EPROM), assemble and link the source code, then program the device.

The source file for a character ROM is organized into 16 bytes for each of the 256 characters (4096 bytes). When assembled and linked, this file fits into a 4K ROM. Each character can contain only 12 rows of dots, and the last 4 bytes of each character must be set to FFH.

Each character on the monitor fits within a 9-column by 12-row block. Each byte corresponds to the 9 columns within one row. For regular characters, the first row is blank (reserved for ascenders), the last two rows are blank (reserved for descenders), and the two outside columns are usually blank (for intercharacter spacing). Generally, then, a typical character fits within a 7-column by 9-row block.

For each character block, column 1 is at the right side and column 9 is at the left.

Each byte is encoded as follows:

* Bit 0 (the low bit) is at the right side of the character block and bit 7 (the high bit) is at the left.

* Setting a bit to 0 means to put a dot at that location.

* Setting a bit to 1 means do not put a dot at that location.

* Setting the high bit to 0 encodes column 1 the same as column 2 and encodes column 9 the same as column 8.

* Bit 0 encodes column 2; bit 1 encodes column 3; and so on.

Two encoding examples are shown in Figure 2-8

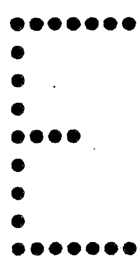
Example 1 is the letter "E." Example 2, a meaningless graphic character, illustrates some specific applications. Both hexadecimal and binary encoding are shown beside each character.

Example 1:

Example 1:

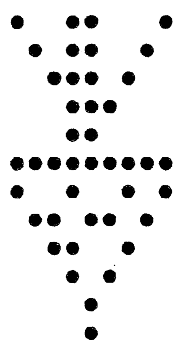
Dot Count 9 8 7 6 5 4 3 2 1

Bit Count 76543210



Hexadecimal	Binary
FFH	11111111
80H	10000000
BFH	10111111
BFH	10111111
BFH	10111111
87H	10000111
BFH	10111111
BFH	10111111
BFH	10111111
80H	10000000
FFH	11111111
FFH	11111111

Example 2:



Hexadecimal	Binary
67H	01100111
A6H	10100110
C5H	11000101
E3H	11100011
E7H	11100111
00H	00000000
6DH	01101101
92H	10010010
CDH	11001101
EBH	11101011
F7H	11110111
F7H	11110111

Notes:

1. Column 1 and column 9 must be the same.
2. Column 1 and column 2 must be the same if the high bit is 0.
3. Column 8 and column 9 must be the same if the high bit is 0.
4. No capability exists for a half-dot shift.
5. Each character must have sixteen bytes; otherwise, strange characters result.

2223216-8

Figure 2-8 Encoding Examples

2.4.8.5 Attribute Interaction. The attributes available for use with the character display can be used in any of the 128 possible combinations. The following paragraphs explain what happens when several attributes are active at once.

The attributes have a priority in their effects, and the highest priority attributes affect all attributes that have a lower priority. The order of priority is as follows.

Highest	Color attributes - red, blue, green
	Reverse video and cursor
	Character enable
	Blink
Lowest	Underline

For example, when the underline and blink attributes are set, both character and underline blink. When the character enable is set to disable, no character, underline, or blinking activity is present. When reverse video and blink are set, the character goes on and off, the background is lighted, and the foreground is dark and blinking. When the character enable is set to disable and reverse video is set, the entire cell is lighted (according to the color attributes).

The color attributes define the characteristics of the "light" portion of the character, that is, either the color (when a color monitor is used) or the intensity (when a monochrome monitor is used).

When the graphics board is used with the alphanumerics CRT controller board, the graphics screen "shows through" the "dark" portion of the alphanumeric character display.

2.4.8.6 Attribute Hardware. The attribute logic design is of the "pipeline" type because the activity of the attributes must occur with dot-timing precision (within 55 ns). To get data from a latch, through several levels of logic, and set up into the next latch, some SCHOTTKY logic is used. The attribute data from the RAM latches is latched again by two 74S175s (U16, U17). This latching allows for the one-character delay through the character ROM and provides tightly timed outputs to the logic. The cursor (CUR) and display enable (DE) lines are also delayed twice to keep them synchronous with the other information (U18).

Propagation delay through the logic can cause timing skews greater than a dot time, so the outputs of the first logic level are relatched one dot-time later. After going through the second logic level (MUX U20), the outputs are latched again for presentation to the video outputs (U39 74S174).

The red, blue, and green outputs are buffered by a 74LS244 before being sent to the 9-pin connector. The color outputs and composite sync are buffered by a 74S00, which has an isolated power supply. They are combined by a resistor network and buffered by a transistor to make up the composite video output. The mapping of colors to intensity in the composite video output is given in Table 2-19.

Table 2-19 Color Map

Code	Color	Composite Video Output (in Volts)
Composite sync		0.47
000	Black	0.78
001	Blue	0.88
010	Red	0.97
011	Magenta	1.07
100	Green	1.18
101	Cyan	1.28
110	Brown	1.37
111	White	1.47

To blank the alphanumerics display to black, set the CRT ENABLE bit in the miscellaneous output latch to low. The board enters this state on power-up.

2.4.8.7 CRT Interrupt Logic Subsystem. The CRT controller board contains a logic subsystem that allows the CRTC to generate an interrupt during the vertical interval. The processor uses this interrupt when doing scrolls with a status line or other operations that must be done during the vertical blanking interval. To enable this interrupt, set the interrupt enable bit in the miscellaneous latch to high. Vertical blanking causes the CPU nonmaskable interrupt, and the interrupt pending bit is set. This bit is read from the miscellaneous buffer. To reset the interrupt, set the interrupt enable bit to low.

2.4.8.8 Diagnostic Loopback. One diagnostic requires that the three color outputs be looped back to the miscellaneous input buffer so that the CPU can read them. Using a program with careful timing from the vertical interval, the CPU can check the action of the atribute bits and the graphics board palette circuits.

2.5 EXPANSION BUS

The other logical function area of the motherboard is the expansion bus. It provides space for the different option boards available for the Texas Instruments Professional Computer.

The expansion bus interface consists of five card-edge connectors, making it easy to add memory-mapped or I/O-mapped options to the system. The expansion bus supports devices that require interrupts for efficient operation. The system does not provide the special-purpose hardware required by direct memory access (DMA) devices.

The expansion bus pin-outs are given in Table 2-20.

Table 2-20 Expansion Bus Pin-Outs

Pin	Signal	Pin	Signal
A01	NMI-	B01	Ground
A02	DATA 7	B02	RESET
A03	DATA 6	B03	+5 V power
A04	DATA 5	B04	IR0 (interrupt 0)
A05	DATA 4	B05	No connection (bussed)
A06	DATA 3	B06	No connection (bussed)
A07	DATA 2	B07	-12 V power
A08	DATA 1	B08	Reserved
A09	DATA 0	B09	+12 V power
A10	WAIT-	B10	Ground
A11	Logic ground	B11	AMWC- (memory write)
A12	ADDRESS 19 (MSB)*	B12	MRDC- (memory read)
A13	ADDRESS 18	B13	AIOWC- (I/O write)
A14	ADDRESS 17	B14	IORC- (I/O read)
A15	ADDRESS 16	B15	No connection (bussed)
A16	ADDRESS 15	B16	No connection (bussed)
A17	ADDRESS 14	B17	No connection (bussed)
A18	ADDRESS 13	B18	No connection (bussed)
A19	ADDRESS 12	B19	No connection (bussed)
A20	ADDRESS 11	B20	PCLK (5-MHz clock)
A21	ADDRESS 10	B21	IR6 (interrupt 6)
A22	ADDRESS 9	B22	IR5 (interrupt 5)
A23	ADDRESS 8	B23	IR4 (interrupt 4)
A24	ADDRESS 7	B24	IR1 (interrupt 1)
A25	ADDRESS 6	B25	IR2 (interrupt 2)
A26	ADDRESS 5	B26	No connection (bussed)
A27	ADDRESS 4	B27	RFSH (refreshing)
A28	ADDRESS 3	B28	ALE (address latch)
A29	ADDRESS 2	B29	+5 V power
A30	ADDRESS 1	B30	OSC (15-MHz clock)
A31	ADDRESS 0 (LSB)*	B31	Ground

* MSB = Most significant bit; LSB = Least significant bit.

2.5.1 Expansion Bus Signal Descriptions

* NMI-. The nonmaskable interrupt signal can be driven by any of the expansion boards to interrupt the system processor. Typically, it is used to alert the processor to a parity error in memory devices residing in the I/O channel. An open collector device pulls this line low when it is being driven by an expansion board. Otherwise, it is held high by a pullup resistor.

* DATA 0-7. These lines form the 8-bit system data bus and can be driven by the processor, memory devices, I/O, or the expansion interface. These bidirectional lines are active high. D0 is the least-significant bit, (LSB) and D7 is the most-significant bit (MSB).

* WAIT-. This signal indicates when a device is holding the system processor, thereby extending the length of a memory refresh or I/O cycle. When a slow device is addressed on the expansion bus, the signal asserts this line low, which extends the cycle-completion time. This line should never be held low longer than 10 processor clock cycles. When driven by an expansion board, an open collector device pulls this line low. Otherwise, a pullup resistor holds it high.

* ADDRESS 0-19. These lines form a 20-bit system address bus, which can address up to 1 megabyte of memory. They are normally driven by the system processor to address memory and I/O devices within the system. (Only XA0 trough XA9 are used for I/O addressing.) These lines are active high.XA0 is the LSB and XA19 is the MSB.

* RESET. This line initializes or resets system logic at power-up or after a power failure. It is active high. A power-supply monitoring device generates RESET immediately when the 12-V line drops below 11.1 V. It returns low 3 ms after regulation resumes. No operator intervention is required.

* INTERRUPT 0-6. These lines signal the processor that an I/O device requires attention. When several devices require service at the same time, the device asserting the lowest-numbered line gets serviced first. These lines are active high. The interrupt request signal must be held high until the interrupt request has been acknowledged.

* AMWC- (or MWRITE-). The memory write signal is usually driven by the system ubdex(AMWC-) processor. It indicates that the information on the data bus should be written to the memory address given on the address bus. This signal is active low.

* MRDC- (or MREAD-). The memory read signal is driven by the

system processor. It indicates that the memory addressed by the address bus should be placed on the data bus. This signal is active low.

* AIOWC- or (IOWRITE-). The I/O write signal is driven by the system processor. It indicates that the I/O device addressed by the address bus should accept the data on the data bus. This signal is active low.

* IORC- or (IOREAD-). The I/O read line is driven by the system processor. It indicates that the I/O device addressed by the address bus should place its data on the data bus. This signal is active low.

* PCLK (processor clock). This is the system clock. It is a one-third division of the OSC clock and has a period of 200 ns (5.0 MHz). The clock has a duty cycle of 37.6 percent (± 3.0 percent).

* RFSH (refreshing). This line indicates that a memory refresh cycle is taking place. It is positive true. When this signal is asserted, all expansion bus activity is ignored. Do not use this line for any purpose.

* ALE (address latch). This line indicates that the processor is placing a valid address on the address bus. The address is valid on the falling edge of this signal.

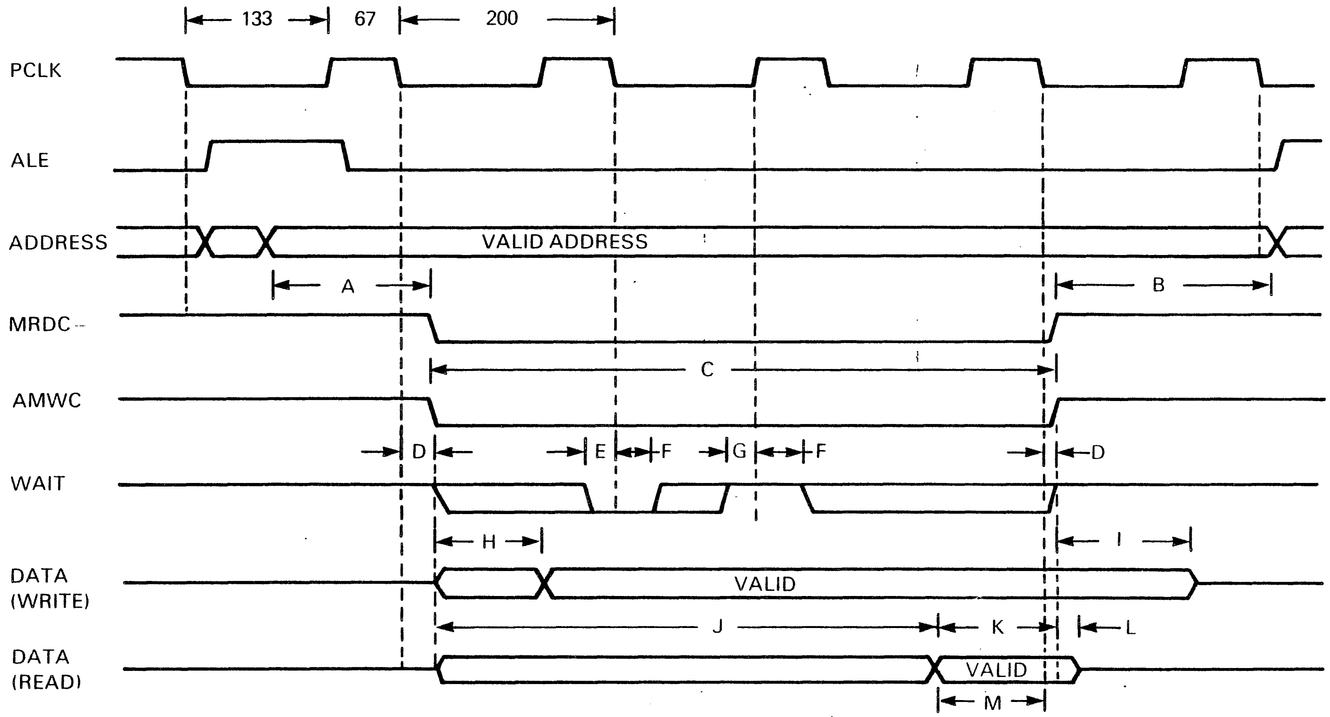
* OSC (clock). This signal describes a high-speed clock having a 66.7-ns period (15.0 MHz). It has a 50-percent duty cycle.

2.5.2 Loading and Driving Requirements

The expansion bus can drive five expansion boards. Each board can support the equivalent of two TTL input loads on any one line of the bus. Open collector outputs, which drive the bus, should be able to sink 16 milliamperes (mA) at 0.5 V. Data bus drivers should be able to sink 24 mA at 0.5 V and source 3 mA at 2.4 V and 15 mA at 2.0 V. Drivers for the interrupt lines IR0-IR6 should be able to source 1 mA at 3.5 V and sink 1 mA at 0.5 V.

2.5.3 Memory Timing

The memory bus cycles can be lengthened in integral multiples of the CLK cycle time (200 ns) using the WAIT- line. Figure 2-9 shows the timing relationships of the expansion bus memory interface.



LEGEND:

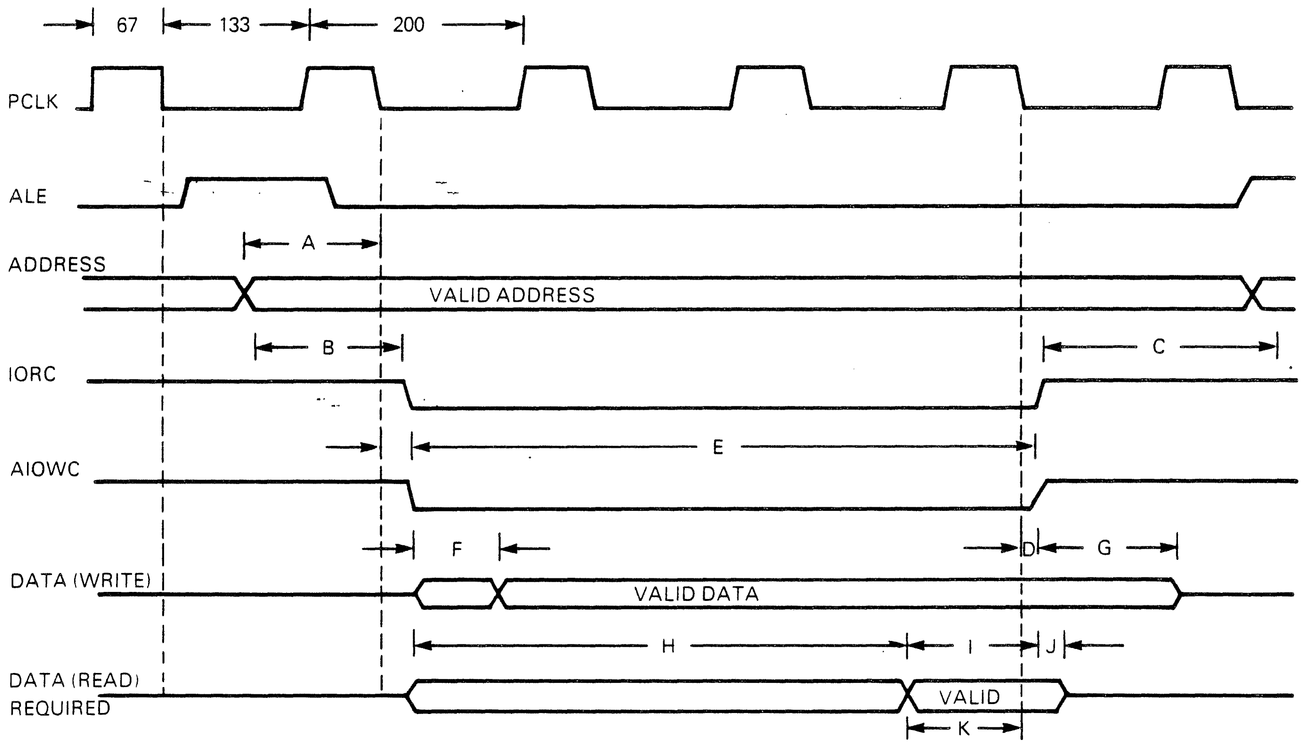
A	ADDRESS SETUP TO COMMAND	72 MIN		
B	ADDRESS HOLD FROM COMMAND	176 MIN		
C –	COMMAND ACTIVE	575 MIN	375 min (WITHOUT WAIT STATE)	
D –	COMMAND DELAY FROM PCLK	35 MAX	10 MIN	
E –	WAIT ACTIVE SETUP	40 MIN		
F	WAIT HOLD	0 MIN		
G	WAIT INACTIVE SETUP	50 MIN		
H –	DATA VALID AFTER AMWC ACTIVE	120 MAX		
I –	DATA HOLD AFTER AMWC INACTIVE	108 MIN		EXPANSION BUS
J –	REQUIRED ACCESS TIME FROM MRDC INACTIVE	515 MAX	315 MAX	
K	DATA SETUO TO MRDC INACTIVE	77 MIN		MEMORY INTERFACE
L	DATA HOLD FROM MRDC INACTIVE	4 MIN		
M –	DATA SETUP TO PCLK LOW	50 MIN		TIMING DIAGRAM

2223216-9

Figure 2-9 Expansion Bus Memory Interface Timing Diagram

2.5.4 I/O Timing

Figure 2-10 shows the expansion bus timing relationships for standard I/O cycles. This timing includes the single wait state that the motherboard always inserts in I/O cycles.



LEGEND:

A	ADDRESS SETUP TO PCLK LOW	62 MIN	
B	ADDRESS SETUP TO AIWOC - OR IORC	72 MIN	
C	ADDRESS HOLD AFTER AIOWC - OR IORC	176 MIN	
D	COMMAND DELAY FROM PCLK	35 MAX	10 MIN
E	ACTIVE IORC OR AIORC TIME	575 MIN	
F	DATA VALID FROM AIOWC LOW	120 MAX	
G	DATA HOLD AFTER AIOWC – HIGH	108 MIN	
H	REQUIRED ACCESS TIME FOR IORC –	515 MAX	
I	REQUIRED DATA SETUP TO RISING EDGE OF IORC –	85 MIN	
J	REQUIRED DATA HOLD AFTER RISING EDGE OR IORC --	-4 MIN	
K	REQUIRED DATA SETUP TO PCLK LOW	50 MAX	

2223216-10

Figure 2-10 Expansion Bus I/O Interface Timing Diagram

Section 3

HARDWARE OPTIONS

3.1 INTRODUCTION

This section describes the hardware options available for the Texas Instruments Professional Computer. Subsections describe the following options:

* Expansion Memory
* Synchronous-Asynchronous Communications Board
* Internal Modems
* Graphics Video Controller Board
* Winchester Disk Drive

The optional diskette drive is identical to the factory-installed diskette drive. Therefore, it is not described in this section. For information, refer to subparagraph 2.4.6.6.

3.2 EXPANSION MEMORY, 512/768 K BYTES

Section 2 describes the expansion memory boards that connect to the motherboard, increasing the memory to 256K bytes (K = 1024). Two additional expansion memory boards (each 256K bytes) are available for the Texas Instruments Professional Computer. One board plugs into the expansion bus, increasing the memory to 512K bytes. The second board mounts on the first (piggyback style so that they use only one of the expansion bus slots), increasing the memory capacity to 768K bytes. This additional memory operates at the same speed as the motherboard memory, so that there is no increase in execution time when the memory is increased.

NOTE

The 512/768 K byte expansion boards are added after the motherboard 192K-byte board is installed.

The first expansion memory card is the controller card. This card contains thirty-six 64K-bit dynamic RAM ICs. The card also holds:

* Decoding logic to establish the addresses

* Parity check logic for error detection

* Timing and refresh logic to operate the expansion memory system.

Connectors and logic for the addition of the second expansion card are also part of the controller card.

The second card also contains thirty-six 64K-bit dynamic RAM ICs. Because the controller card contains all the logic for both cards, this second card is smaller.

3.2.1 Addressing the Expansion Memory

The expansion memory operates at a fixed address in the computer's memory space. Addresses 040000H through 07FFFH are for the first 256K bytes; addresses 080000H through 0BFFFFH are for the second 256K bytes. If the second card is not installed, its assigned memory space can be used by other hardware products.

3.2.2 Expansion Memory Control Logic

The expansion bus contains a bidirectional buffer to separate the data bus from the expansion memory, thereby providing sufficient drive and margins to the data transfers. The hard array logic (HAL) chip HAL16R4 (U2) handles address decoding, buffer control, as well as timing and refresh. The refresh timer (U4) is a one-shot, and the delay line (U3) provides the multiplexer timing.

3.2.2.1 Expansion Memory Refresh Logic. The dynamic RAM refresh logic operates synchronously with the accesses to the RAM memory. Refresh cycles begin only when a RAM cycle is not in progress. This means that the RAM refresh can occur at the same time as accesses to other system memory (ROMs or the main system memory) or I/O space. Each time a refresh cycle begins, a refresh timer (U4) starts. When it times out, it provides the signal beginning another refresh cycle. This timer is set to 15 us maximum, which allows for the worst-case refresh request latency. To maintain the contents of the RAM under worst-case conditions, the refresh must occur at least 128 times within 2 ms. (The average refresh timing is once per 15.625 us.) The worst-case latency for a refresh request is about 600 ns.

Once a refresh cycle has begun, it must be completed (including the precharge) before the next cycle begins. If a RAM access cycle starts before the refresh cycle completes, the HAL state machine puts the CPU into a wait state until the refresh operation completes. In the worst case, this delay could extend the usual memory access time by three wait states or 600 ns.

Assuming a refresh timer value of 14 us, and an average 400-ns slowdown of the CPU, the average refresh overhead is about 2.9 percent. The worst case is about 4.3 percent.

3.2.2.2 CAS and Address MUX Switch Generation. A delay line from the Column Address Strobe X (CASX-) produces the address multiplexer control (MSEL). The delay line is set at 40 ns. U1 buffers the CASI- line, and the RAM buffers are taken from the delay line 60 ns after CASX-. This ensures the maintenance of an adequate row address hold, and enough column address setup time. The RAM still operates quickly enough to finish an access within the system cycle time.

The CASX- timing depends on whether the cycle is a read or a write. If the cycle is a read, the CASX- signal from the logic array is equivalent to the RASI- signal. This provides the maximum available time for the RAM chip to access it's data and present it to the expansion bus. The delay line guarantees the timing of MSEL and CASI- to the dynamic RAMs.

If the cycle is a write, then the CASX- signal follows the rising edge of the first system clock during the write cycle. This is about 130 ns after the occurrence of RASI-. This delay allows time for the data from the processor to propagate through the data buffers and U6, the parity generator chip (74LS280).

3.2.2.3 Expansion Memory Parity Generation and Checking. The parity generator/checker chip (74LS280) generates a 1 to the parity RAM bit whenever there is an even number of "1"s in the data byte being written. A separate data bus on the parity RAM chip uses a tristate driver to provide a high on the output whenever it is not driving the output line (as in the write cycle). The parity is then taken from the "odd sum" output of the parity generator and used to write to the dynamic RAMs. The WCAS- line from the logic array holds the parity error flip-flop (U5) clear. The timing on this line stays low until after the CASI- line clocks the flip-flop. This prevents the generation of a parity error during write.

When the RAM is read, all of the data bits and the parity bit are presented to the generator/checker, and the parity output is sampled at the end of the read cycle. If a parity error is discovered, flip-flop U5 is set to interrupt the CPU on the NMI- line. This NMI- line clears on the next read with correct parity, or on the first write to this board.

Using the "odd sum" method of parity checking does not cause a parity error, even when the system attempts to read from nonexistent RAM. (To determine the size of system memory, system software sometimes "feels" for memory not present.

3.2.2.4 Expansion Memory Control State Machine. A hard array logic device (HAL16R4), set up as a state machine (U2), drives the memory control. This device has four outputs equipped with clocked flip-flops and four outputs that are direct combinations of the inputs. Table 3-1 gives the logic for the memory control state machine. The logical AND of the terms on a line ORed with the AND of terms on other lines results in low-going outputs. This occurs either directly, on those outputs without registers, or after the clock on those outputs having registers.

Table 3-1 Expansion Memory Control State Machine Logic - HAL16R4

Output	MRD-	MWR-	RFRQ	XA19	XA18	LGND-	XXXX	B2IN-	RASI-	XWAIT-	CASX-	BUFE-	RFSH-	RRAS-	WCAS-	ZZZZ-	Comment
	Input																
RASI-	L	.	.	L	H	L	.	.	.	.	.	.	H	H	.	.	Memory read low bank
or	L	.	.	H	L	L	.	L	.	.	.	.	H	H	.	.	Memory read high bank
or	.	L	.	L	H	L	.	.	.	.	.	.	H	H	.	.	Memory write low bank
or	.	L	.	H	L	L	.	L	.	.	.	.	H	H	.	.	Memory write high bank
or	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	Refresh
XWAIT-	.	.	.	.	.	.	.	.	.	.	.	L	L	.	.	.	Refresh+memcycle 1, 2
or	.	.	.	.	.	.	.	.	.	.	.	L	.	L	.	.	Refresh+memcycle 2, 3
CASX-	.	L	.	.	.	.	.	.	.	.	.	.	.	.	H	.	Write, either bank
or	L	.	.	L	H	L	.	.	.	.	.	.	H	H	.	.	Read low bank
or	L	.	.	H	L	L	.	L	.	.	.	.	.	.	.	.	Read high bank
BUFE-	.	L	.	L	H	L	.	.	.	.	.	.	.	.	.	.	Write low bank
or	.	L	.	H	L	L	.	L	.	.	.	.	.	.	.	.	Write high bank
or	L	.	.	L	H	L	.	.	.	.	.	.	.	.	.	.	Read low bank
or	L	.	.	H	L	L	.	L	.	.	.	.	.	.	.	.	Read high bank
The following four outputs have flip-flops:																	
RFSH-	H	H	H	.	.	.	.	.	.	.	.	.	.	H	.	.	Refresh 1; no memcycle
or	.	.	H	L	L	.	.	.	.	.	.	.	.	H	.	.	Refresh 1; motherboard cycle
or	.	.	H	H	H	.	.	.	.	.	.	.	.	H	.	.	Refresh 1; graphic cycle
or	.	.	H	H	L	.	.	H	.	.	.	.	.	H	.	.	Refresh 1; high bank not in
or	.	.	.	.	.	H	.	.	.	.	.	.	.	H	.	.	Refresh 1; illegal cycle
or	.	.	.	.	.	.	.	.	.	.	.	.	L	H	.	.	Refresh RF2,3
or	L	L	.	.	.	.	.	.	.	.	.	.	.	H	.	.	Reset
RRAS-	.	.	.	.	.	.	.	.	.	.	.	.	L	.	.	.	Refresh RF2,3,4
or	L	L	.	.	.	.	.	.	.	.	.	.	.	.	.	.	Reset
WCAS-	.	L	.	L	H	L	.	.	.	.	.	.	H	H	.	.	Write low bank
or	.	L	.	H	L	L	.	L	.	.	.	.	H	H	.	.	Write high bank
or	L	L	.	.	.	.	.	.	.	.	.	.	.	.	.	.	Reset
ZZZZ-	L	L	.	.	.	.	.	.	.	.	.	.	.	.	.	.	Reset

Legend:

L = Low signal.

H = High signal.

Notes for Table 3-1

1. The signal RASI- activates RAS- from the RAM address multiplexer of the 2964.
2. The signal XWAIT- puts the processor into a wait state.
3. The signal BUFE- activates the expansion memory system data buffer.
4. The signal CASX- controls the CAS and MSEL generation.
5. The signal RFSH- instructs the 2964 address multiplexer to put out the refresh address.
6. The signal RRAS- combines with RFSH- to indicate that a refresh RAS is in progress.
7. The signal WCAS- delays CASX- during a write cycle.
8. The signal ZZZZ- is not used.

A timing diagram of the memory system, shown in Figure 3-1, indicates the major operations of the memory system.

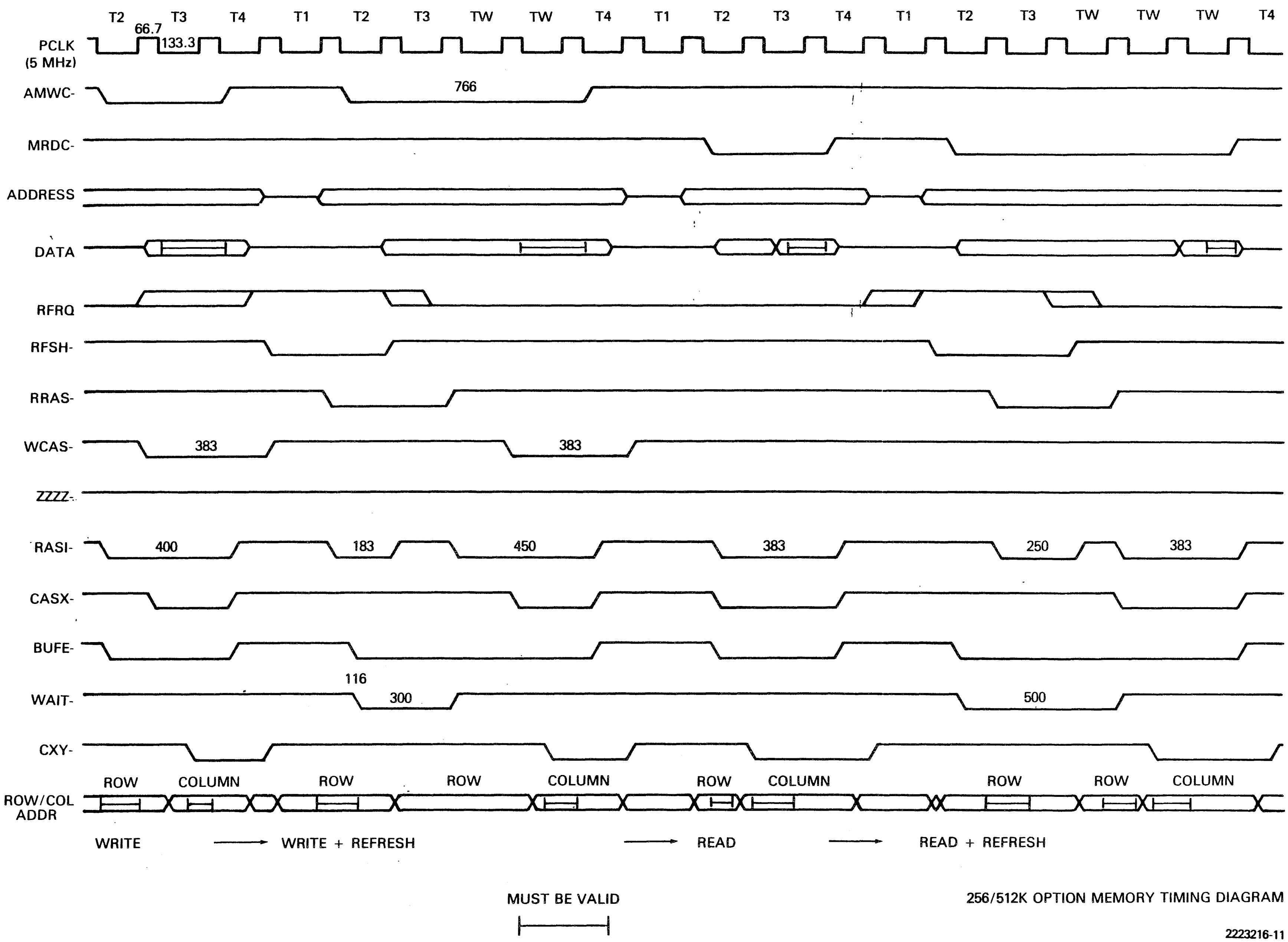


Figure 3-1 Expansion Memory Timing Diagram

3.3 SYNCHRONOUS-ASYNCHRONOUS COMMUNICATIONS BOARD

This subsection describes the theory of operation and the functions of the synchronous-asynchronous communications (sync-async comm) board. Figure 3-2 is a block diagram of the sync-async comm board. Refer to Section 6, drawing 2223096, for logic diagrams.

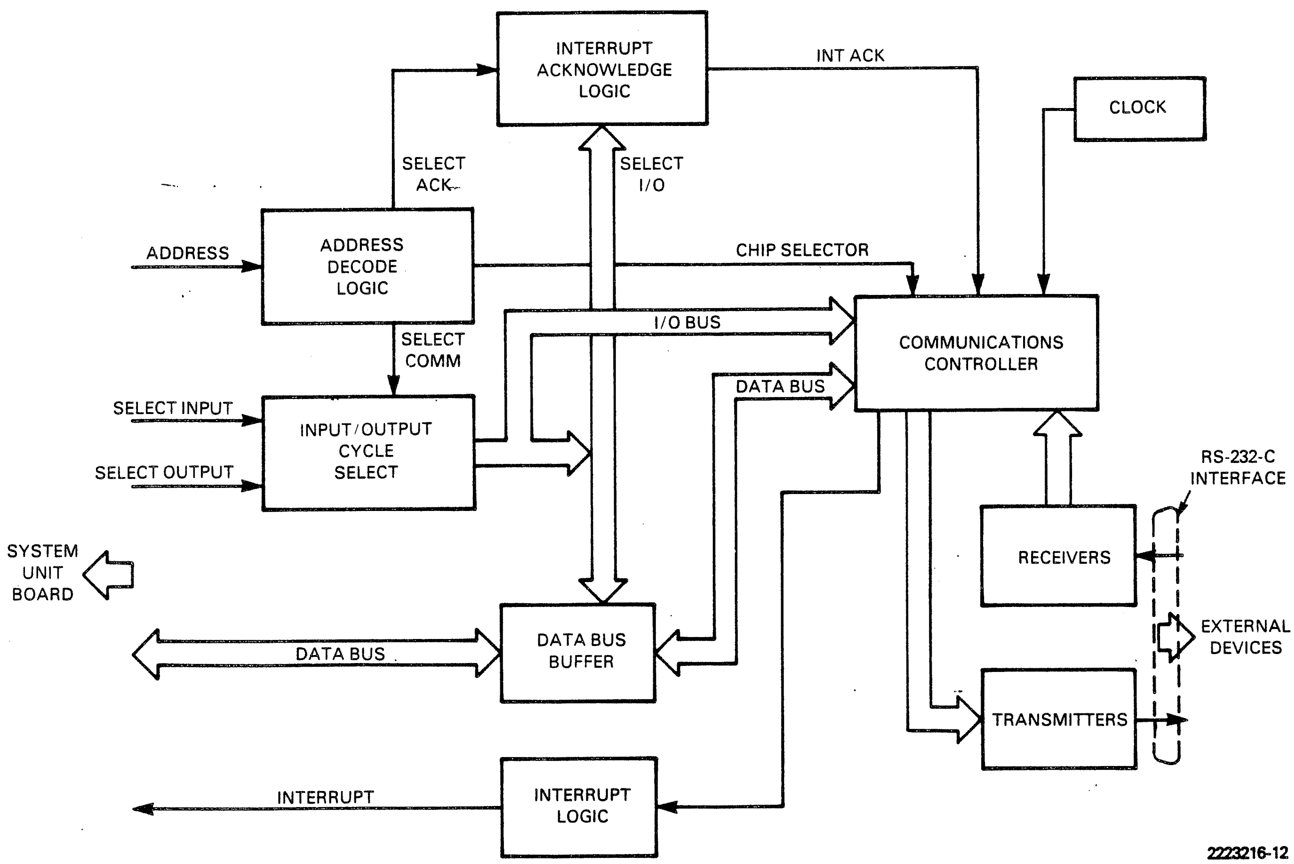


Figure 3-2 Sync-Async Comm Board Block Diagram

The sync-async comm board is based upon the Zilog Z8530 Serial Communications Controller (SCC). This device automatically handles asynchronous protocols. It also services most synchronous protocols, including data link control (SDLC) and high-level data link control (HDLC), (both bit-oriented.) Cyclic redundancy check (CRC) is an automatic function and can be included in any transmission.

NOTE

A sample program, showing general programming procedures and recommended use of the sync-async comm board, is included in Appendix E of this manual. For more detailed information, refer to the Zilog 8530 Technical Manual.

The functions of the sync-async comm board are:

* System interface
* Baud rate generation
* Port addresses

3.3.1 System Interface

Most of the components on the board are involved in handling the interface between the system bus and the Z8530. Of special note is the logic that generates the interrupt acknowledge (INTACK) signal that the Z8530 requires in response to an interrupt request. The INTACK- signal is software-generated. It is not part of the system interrupt acknowledge signal because of the setup time required and because the system expansion bus does not provide for expanding the number of interrupt levels.

To generate the INTACK- signal, the software does a AIOWC- (write) to the I/O address for interrupt acknowledge and then does a IORC- (read) from the same address. The data received on this read is the interrupt vector from the Z8530.

The AIOWC- signal clears U5B, activating the INTACK- signal to the Z8530. When the IORC- occurs, the vector from the Z8530 is gated onto the data bus. The rising edge of IORC- clocks U5B to the inactive state which releases the INTACK-.

Other logic on the system side of the board delays the read and write commands to the SCC so that the address and data setup times and the hold-time requirements of the part can be met. IORQ is connected to the input of a flip-flop 74LS74 (U5A). The clock input is connected to the system CLK line. The rising edge of the clock occurs 133 ns after the IORC- or AIOWC- signal occurs. The output of U5A, gated with IORC- and AIOWC-, delays the start of the SCCRD- and SCCWR- signals. The clear input to U5A is connected to BDCS, allowing the SCCRD- and SCCWR- signals to occur only when the board is selected.

Resetting the Z8530 requires that the SCCRD- and the SCCWR- lines be held active simultaneously. This results from the logical OR of U6C and U6D with the RESET signal from the bus and the SCCRD- and SCCWR- lines.

U4C inverts and buffers the interrupt output from the SCC. This signal then goes to a set of stake pins and is used to determine the interrupt level at which the board is operated.

3.3.2 Baud Rate Generation

The 4.9152-MHz crystal oscillator on the board, divided by 2, provides a clock for the SCCs (internal baud rate generators). To generate a specific baud rate, program the values given in Table 3-2.

Table 3-2 Sync-Async Comm Board Baud Rate

Baud Rate	Sync Value	Percentage of Error	Async Value	Percentage of Error
19 200	62	0.000	2	0.000
9 600	126	0.000	6	0.000
7 200	169	-0.196	9	-3.030
4 800	254	0.000	14	0.000
3 600	339	0.098	19	1.587
2 400	510	0.000	30	0.000
2 000	612	0.065	36	1.053
1 800	681	-0.049	41	-0.775
1 200	1022	0.000	62	0.000
600	2046	0.000	26	0.000
300	4094	0.000	54	0.000
200	6142	0.000	82	0.000
150	8190	0.000	10	0.000
134.5	9134	0.001	69	0.001
110	11169	-0.001	96	0.026
75	16382	0.000	1022	0.000
50	24574	0.000	1534	0.000

3.3.3 Addressing

A 74LS139 decoder (U3) and several gates (to qualify the address) comprise the address selection logic. The board design presents a choice of four address locations, permitting the addition of several communications boards to the system.

As with other I/O devices for this bus, only 10 of the address lines are decoded. U3 provides two decoded outputs: INTCS-, which activates the INTACK logic; and SCCCS-, which activates the Z8530. The logical OR of INTCS- and SCCCS- creates the board select signal (BDCS). The logical AND of IORC- and AIOWC- creates IORQ. BDCS and IORQ combined enable the bus buffer U7.

3.3.4 Programming

The sync-async comm board port number is programmed by placing jumpers on the board. Five I/O addresses and a distinct interrupt level control each port.

Table 3-3 gives the board addresses for the four possible ports. P60 is the board connector.

Table 3-3 Sync-Async Comm Board Port Addresses

Port 1 Interrupt

Jumper Locations	P60 Pin No.	Address	Function
E1-E2	8 (INT0)	00E0	Interrupt acknowledge
E7-E8		00E4	CHB command
		00E5	CHB data
		00E6	CHA command
		00E7	CHA data

Port 2 Interrupt

Jumper Locations	P60 Pin No.	Address	Function
E4-E5	50 (INT1)	00E8	Interrupt acknowledge
E10-E11		00EC	CHB command
		00ED	CHB data
		00EE	CHA command
		00EF	CHA data

Port 3 Interrupt

Jumper Locations	P60 Pin No.	Address	Function
E2-E3	48 (INT2)	00F0	Interrupt acknowledge
E8-E9		00F4	CHB command
		00F5	CHB data
		00F6	CHA command
		00F7	CHA data

Port 4 Interrupt

Jumper Locations	P60 Pin No.	Address	Function
E5-E6	46 (INT4)	00F8	Interrupt acknowledge
E11-E12		00FC	CHB command
		00FD	CHB data
		00FE	CHA command
		00FF	CHA data

Two channels (A and B) from each port control the Z8530 operations. Channel A, the main communications channel through which data transfer takes place, also monitors or controls some of the RS-232-C signals. Channel B does nothing but control or monitor signals. It is not used for data transfer.

Each channel can be accessed by two addresses: "command" and "data." The command address for either channel is used to access any of the 15 read or write registers that control the Z8530 operations. The data address for channel A is used to read received data and to write transmitted data. The data address for channel B is not used.

Because the Z8530 does not contain pin-outs for the DSR, SCF, and RI signals, unused pins from channel B are used for these signals. Table 3-4 lists the specific pin-out for these signals. Table 3-5 lists the Channel B pin-out for the Z8530 interrupt enables.

Table 3-4 Channel B Pin-Out for Z8530

Z8530 Signal	Channel B Pin-Out
DSR	DCD
SCA	DTR
SCF	SYNC/HUNT
RI	CTS

Table 3-5 Channel B Pin-Out for Z8530 Interrupt Enable

Z8539 Interrupt	Channel B Pin-Out
DSR	DCD
SCA	none
SCF	SYNC/HUNT
RI	CTS

Each port has an I/O address used to acknowledge the Z8530 interrupts. An I/O write followed by an I/O read done at this address acknowledges the interrupt. The data written during the I/O write is irrelevant. After the I/O read, the Z8530 returns the code for the interrupt that occurred. These codes are explained in the <u>Zilog 8530 Technical Manual</u>.

The external connector (J69) is an RS-232-C type. Table 3-6 identifies the signals at this connector.

Table 3-6 RS-232-C Connector Signals

Pin	Signal Name	Signal
1	Chassis ground	AA
2	Transmitted data	BA
3	Received data	BB
4	Request to send	RTS/CA
5	Clear to send	CTS/CB
6	Data set ready	DSR/CC
7	Signal ground	AB
8	Data carrier detect	DCD/CF
9	No connection	--
10	No connection	--
11	Secondary request to send	SCA/CH
12	Secondary clear to send	SCF/CI
13	No connection	--
14	No connection	--
15	Transmitter clock in	TXC/DB
16	No connection	--
17	Receiver clock in	RSC/DD
18	No connection	--
19	No connection	--
20	Data terminal ready	DTR/CD
21	No connection	--
22	Ring indicator	RI/CE
23	Same as pin 11	SCA/CH
24	External transmitter clock	DA
25	No connection	--

3.4 INTERNAL MODEMS

Texas Instruments offers two internal modems for the Professional Computer. One is a Bell 103-compatible type, which operates at 300 baud. The other is Bell 212-compatible and operates at 1200 baud. Both are full-duplex modems, and the Bell 212-compatible can operate in full-duplex, synchronous, 1200 baud. These are "smart" modems, and can handle a variety of commands for establishing communications. Both modems have automatic dialing capability using either pulse or tone dialing. The modem also provides status indications for monitoring the progress of the dialing procedure.

The following subsections describe the architecture and interface of the modems to the system for those users who want to write their own communication program, and who want to use an internal modem.

3.4.1 Architecture

The interface hardware for the modem board is identical to that created for the sync-async comm board. Therefore, it is easy to adapt software written for the sync-async comm board so that it can operate with either of the modems. Adding code to handle the modem dialing procedure is the major change required. The same port addresses and interrupt levels used by the sync-async comm board are used by the modem boards.

Figure 3-3 shows a block diagram of modem hardware. The serial controller (Zilog 8530) sends the modem commands during the modem initialization and dialing procedure. Then the Z8530 transfers data between the modem and the remote system.

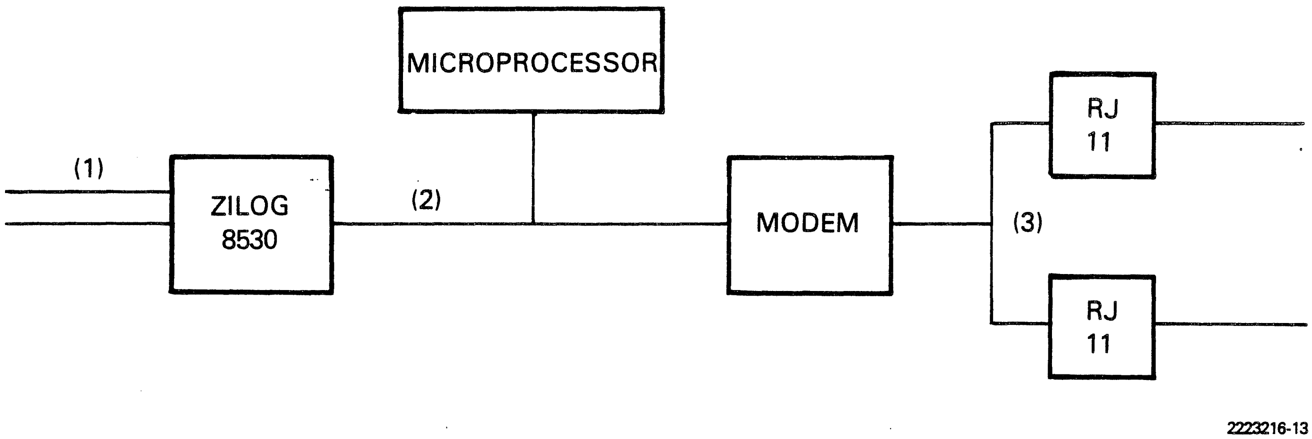
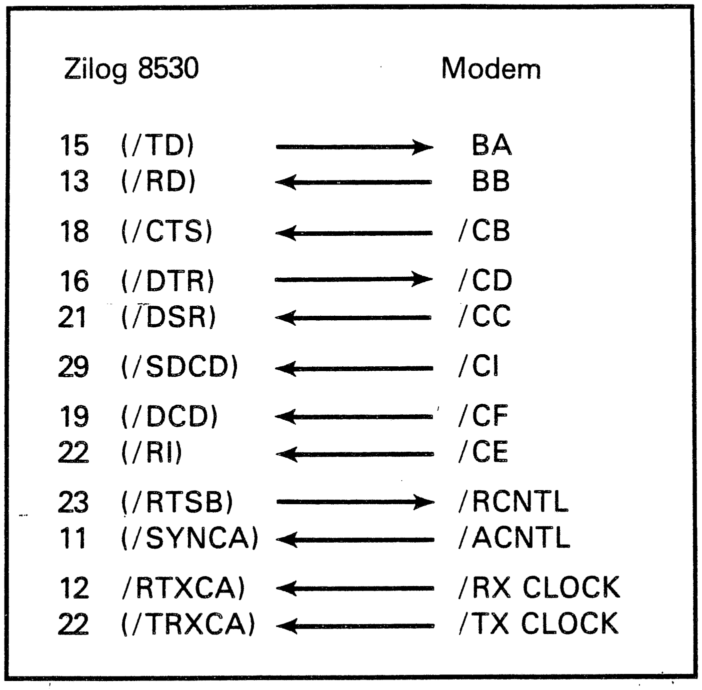


Figure 3-3 Modem Hardware Interface

3.4.2 Zilog 8530--Modem Signals

Two special control signals, /RNCTL (request control mode) and /ACNTL (acknowledge control mode), tell the modem how to handle information passed by the Z8530. /RNCTL information is processed as commands, while /ACNTL information is interpreted as data to be transmitted.

The signals that appear at the Zilog 8530--modem interface are shown in Figure 3-4.



2223216-14

Figure 3-4 Zilog 8530--Modem Interface Signals

The following paragraphs give brief descriptions of these signals.

NOTE

> In the following descriptions, "ON" refers to an active-low TTL voltage level.

<u>(/TD) -> BA</u> The Z8530 sends data to the modem on this line. The condition of /RCNTL determines the type of data (either transmitted data or command data).

<u>BB -> (/RD)</u> The modem sends data to the Z8530 on this line. The condition of /RCNTL determines the type of data (either transmitted

data or command data).

<u>/CB -> /CTS)</u> When this signal is on, the modem is ready to receive transmitted data from the Z8530. Even when this signal is off, the Z8530 can still send command data if /ACNTL is on and /CD (DTR) is off. No transmitted data is sent while this signal is off.

<u>(/DTR) -> /CD</u> When this signal is on, the terminal is ready to start the communication. This signal is turned on while the unit is in the command mode, but before giving the start-dial command. (If the start-dial command is given before /DTR is on, the modem returns a "command failed" status.)

<u>/CC -> (/DSR)</u> The modem completes dialing, then turns this signal on while waiting for the answer tone and the carrier. The modem indicates three things by turning this signal on: that it is electrically connected to the communication line; that it is off-hook; and that it is ready to start communication activity.

<u>/CI -> (/SDCD)</u> After answering a call, the modem generates this signal to indicate how fast data is being transmitted to the terminal. Turning the line on indicates that data is being transmitted at high speed. Turning the line off indicates that data is being transmitted at low speed. During the originate modes, this signal represents the selected rate of data transfer.

<u>/CF -> (/DCD)</u> When this signal is on, the modem is receiving the data signal from the communications line and communications can begin.

<u>/CE -> (/RI)</u> The modem generates the voltage levels on this line to indicate the ringing activity. When the signal is on, the line is ringing. Between rings, or when there is no ringing, the signal is off. The software detects the ringing activity through the Z8530, and asserts DTR if the call is to be answered.

<u>(/RTSB) -> /RCNTL</u> The software uses this signal to change the mode of data transfer. When this signal is on, it indicates that the terminal wants to enter into the command mode. In command mode, the modem does not transmit the data received on the line BA. Instead, it uses the data for command and status information exchange between the terminal and the modem. During initialization and dialing procedures, the modem uses the command mode to send modem dialing commands and to receive status information.

Once the data transfer mode is initiated, the command mode cannot be invoked again unless the line is disconnected.

<u>/ACNTL -> (SYNCA)</u> The modem generates this signal in response to the /RCNTL signal from software. The software does not send any command data on line BA until this signal is turned on. When the /RCNTL signal goes away and the modem enters the data transfer mode, this signal is turned off. The /ACNTL signal is usually pulled high on the RS-232 interface board. When both /RCNTL and /ACNTL are on, the

terminal can exchange commands and information with the modem.

The /ACNTL signal combined with the /RCNTL signal can differentiate between the modem board and a sync-async comm board. To check for an installed modem, the software first activates the RCNTL, then waits for the modem to return the /ACNTL signal. If no acknowledge signal returns, then a sync-async comm board is installed, rather than a modem board.

/RX CLOCK -> (/RTXCA) This is the receive data clock line for asynchronous communication.

/TX CLOCK -> (/TRXCA) This is the transmit data clock line for asynchronous communication.

3.4.3 Modem Initialization

At power-up, the RESET signal on the system bus initializes the modem, using the operating defaults. The user can reset the modem to these same defaults at any time with the software reset command.

The default parameters are listed in Table 3-7.

Table 3-7 Modem Default Parameters

Parameter	Default Setting
Dialing	Pulse dial
Line termination	On hook
Modem transmitter	Squelched
Modem mode	Originate
Data/command mode	Data mode
Communication	Asynchronous

3.4.4 Command Mode Operation

The modem has two modes of operation, data transfer mode and command (also called control) mode. The terminal system software communicates with the processor on the modem board, either for the data transfer or the command mode. All data and command transfer passes through the USART.

At power-up, the default setting is for the data transfer mode. For various reasons, such as a software request for diagnostic status information, it is necessary to place the unit in command mode. The terminal and the modem are in master-slave configuration, and the modem cannot initiate the command mode.

To prepare for command mode operation, the Z8530 must be set up for 300-baud operation, no parity, 8 bits per character, one stop bit, and one start bit. The Zilog 8530 Technical Manual contains details on setting the Z8530. Also, refer to subsection 3.3 of this manual. Appendix F contains "RCNTL", a sample subroutine that checks for an installed modem.

Once the appropriate signals are set, the modem and the terminal can enter into a command status transfer dialogue. The software asserts line /RCNTL, requesting the modem to enter the command mode. The modem responds by asserting the line /ACNTL. The software then waits until /ACNTL is turned on by the modem before sending any commands.

To find the status of the modem, the computer transmits the code "send diagnostic status" (44H). The modem returns a 2-byte response, the first byte indicating that the "status byte follows" and the second byte giving the status.

The commands and status codes are listed later in this section. Appendix F contains "DIAGST", a sample routine for starting a dialogue in the command mode.

After the modem completes a command from the computer, it sends a "command complete" (A=41H) code or a "command failed" (Z=5AH) code. After sending a command, the computer waits before sending another command, expecting either a direct response or a command complete/failed status.

The terminal software can insert a fail-safe time-out between issuing a command to the modem and receiving the command status to protect against possible modem malfunction.

After the software completes the command/status dialogue, it releases the /RCNTL line. The modem responds by releasing the /ACNTL line. The system is now in the data transfer mode.

The command mode cannot be reentered unless the communication is halted and the phone line is disconnected. The software turns off the DTR signal when the line is to be disconnected. The modem disconnects the line any time DTR is turned off, once the connection has been established.

3.4.5 Dialing Procedure

To begin a call, the terminal transmits the telephone number to be dialed (including any separator symbols such as ( ), -, +, or @) and instructions on the method of dialing (such as T or P). For example, in the telephone number T(713)-895-0001X, T requests tone dialing, and X is the telephone number terminator. The number can be a maximum of 23 digits long. The modem responds with the "command complete" status, then dials the number. Appendix F contains "Dialer", a sample routine for dialing a telephone number.

The ( ) and - separators are used for number-grouping purposes only. They have no meaning to the modem. The modem reads the + separator as tandem dialing. Each time the modem finds a +, it waits for another dial tone before continuing. The @ symbol represents blind dialing. When the modem finds the @ separator, it waits 2.0 ± 0.1 s after the command is received, then dials the number without waiting for a dial tone.

The dialing methods include tone dialing, pulse dialing, and automatic selection. The modem is able to alternate dialing methods during the dialing procedure. Simply insert the proper characters (T for tone dialing, P for pulse dialing) in the telephone number. For example, in the number

T8-50-33333344-P(713)-895-0001,

the modem dials all the digits to P using the tone mode; all digits after P are dialed using the pulse mode. The modem echoes the number back to the terminal (without separators) as it dials each digit, then sends status to the terminal for full call-progress monitoring. The status can be ringing, busy, no answer, or voice. The terminal screen displays the appropriate message.

When the connection attempt is successful, the modem does not return a status indicator. Instead, the computer monitors the signal /DCD. The modem asserts /DCD, indicating a successful connection.

The dialing procedure is aborted any time the DTR signal is dropped. The modem sees this as a command to stop dialing, and goes on hook.

The modem waits through 10 rings before reporting a no-answer condition. The default time to wait between retries is 11 s, the default number of retries is 0. Ten rings as a no-answer condition is a fixed number; however, the time to wait between retries and the number of retries can be programmed into the terminal software.

3.4.6 Time-Outs

Both the terminal and the modem can cause time-outs. The terminal time-outs are: loss of carrier, long space received, and no response. The two types of modem time-outs are: loss of carrier and abort timer.

Table 3-8 summarizes the time-outs.

Table 3-8 Types and Durations of Disconnects

Terminal		Modem	
Type	Duration	Type	Duration
Loss of carrier	200 ms	Abort timer	17 s
Long space received	1.5 s	Loss of carrier	50 ms
No response time-out	1 s		

The following paragraphs give brief descriptions of all time-out conditions.

3.4.6.1 Terminal or Software Time-Outs.

* Loss of Carrier. If the terminal is programmed for fail-safe disconnects when the carrier goes off, it waits 50 ms before disconnecting.

* Long Space Received. At start-up, the terminal sends a command to the modem, then waits for the modem to turn on the /ACNTL signal. If the modem fails to return the signal within 1.5 s, the terminal disconnects.

* No Response. The terminal sends a command to the modem, then waits for the modem response. After 1 s, the terminal disconnects.

3.4.6.2 Modem Time-Outs.

* Loss of Carrier. During a temporary loss of carrier, this timer holds the DCD line true. However, if the carrier stays off for 50 ms (the length of the timer), the modem turns off the DCD signal to the Z8530, causing the software to recognize the loss of the carrier.

* Abort Timer - Originate Mode. During the automatic dialing procedure, the modem goes off hook to listen for the dial tone. The modem waits 17 s, then sends the "command failed" status and goes on hook. The terminal responds by dropping DTR.

  The abort timer resets after the dialing procedure is complete. If the modem being used is a Bell 212A-compatible type, the abort timer is set for Bell 212 high-band carrier.

* Abort Timer - Answer Mode. During a manual dialing procedure, the answer-tone abort timer is used instead of the dial-tone abort timer. The originating modem looks for

an answer from the remote modem. The answer depends upon the type of modem installed in the remote system. If the remote is Bell 103-compatible, the modem looks for the carrier. If the remote is Bell 212-compatible, the modem looks for the scrambled mark or the unscrambled mark. The modem waits 17 s for the answer tone, then drops DSR.

3.4.7 Modem Software

The modem software is very simple. Some commands are only 1 byte long, such as the "Manual Disconnect" command. Field commands, such as "Telephone Number" (an op code followed by a field), are longer.

The terminal sends a command to the modem. The modem returns a direct response or a status byte (command complete or command failed). The terminal does not send additional commands until this handshake is completed.

Table 3-9 lists the software commands from the terminal to the modem.

Table 3-9 Commands from the Software to the Modem

ASCII Code	Command
A	Dial following telephone number, select dialing mode
B	Next byte contains number of retries (ASCII, 0-9)
C	Next 2 bytes contain time (in s) between retries (ASCII, 0-99 s)
D	Request diagnostic status
E	Disconnect on loss of carrier
F	Do not disconnect on loss of carrier
G	Manual answer
H	Select 1200- bps option
L	Select 300- bps option
M	What modem type?
O	Manual originate
P	Dial following telephone number using pulse dialing
R	Start RDLB test*
S	Synchronous communication mode
T	Dial following telephone number using tone dialing
U	Asynchronous communication mode
W	Software reset
X	Telephone number terminator
Y	Start ALB test**
+	Tandem dialing (wait for another dial tone)
@	Blind dial (wait 2.0 s, then dial)

* The RDLB (Remote Digital Loopback) test is for a Bell 212-compatible modem. It checks the condition of the communication lines. The originating modem makes the answering modem echo all received data back to the originating modem.

** The ALB (Analog Loopback) test causes the modem's internal logic to connect the transmitter to the receiver and loopback the data.

Table 3-10 lists the possible responses from the modem.

Table 3-10 Response from the Modem to the Software

ASCII Code	Command
A	Command completed
B	Busy tone
D	Diagnostic status follows
E	Phone number terminator
F	Phone number follows
H	Bell 212A option installed
L	Bell 103 option installed
N	No answer
O	Lost call
R	Ringing from ringback
V	Voice reception
Z	Command failed

One possible modem response is D, diagnostic status follows. Immediately after the modem sends this reply, it sends one of the diagnostic indicators from Table 3-11.

Table 3-11 Diagnostic Status Indicators

Byte Value	Meaning
00	Good check
01	ROM error
02	RAM error
04	Processor error
08	Timer error
10	Not used
20	Not used
40	Not used
80	Not used

3.5 GRAPHICS VIDEO CONTROLLER BOARD

The graphics video controller board operates with the CRT controller board. It is mounted (piggyback fashion) on the CRT controller board, and all its connections are to the CRT controller board. Figure 3-5 is a block diagram of the graphics video controller board. (Refer to Section 6 for logic diagrams.)

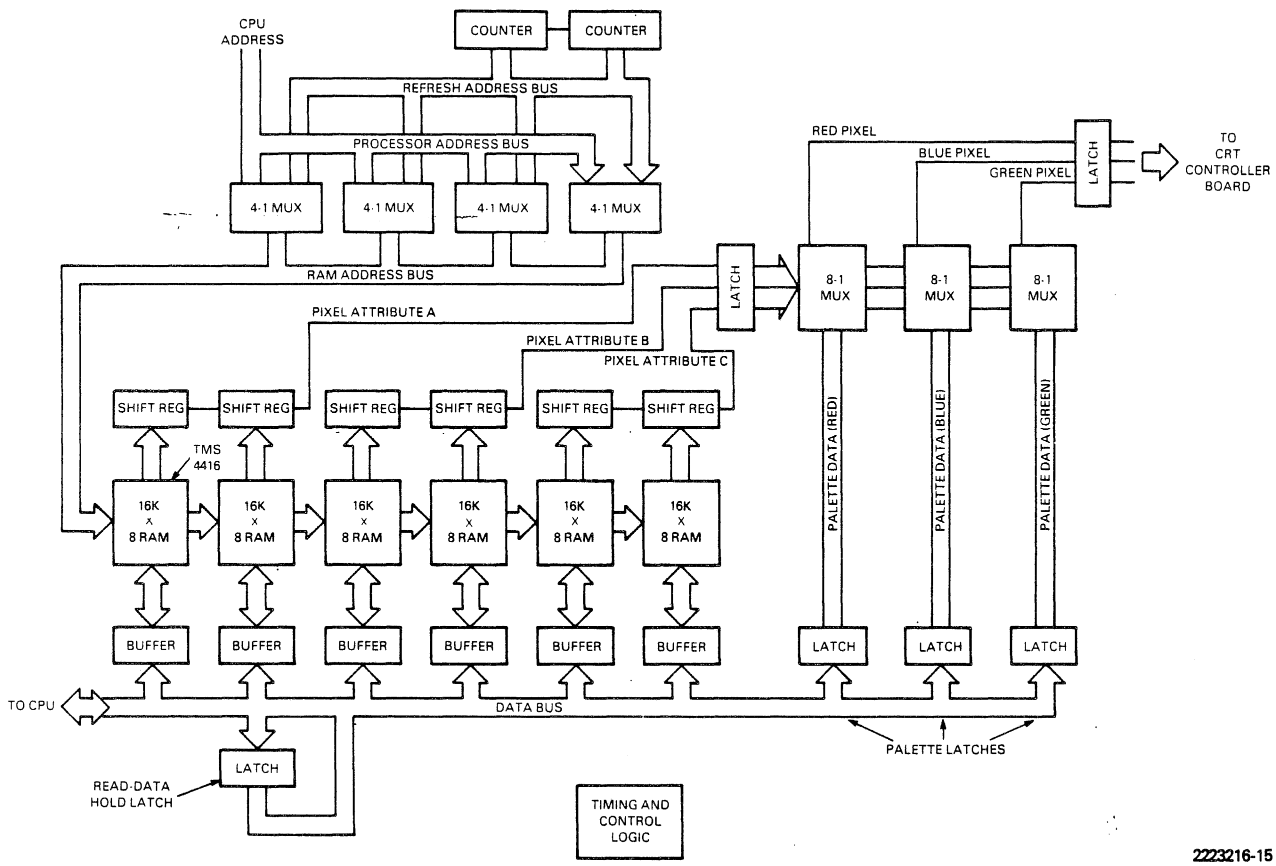


Figure 3-5 Graphics Video Controller Board Block Diagram

The graphics video controller board uses the same number of pixels (720 horizontal x 300 vertical) on the screen as does the alphanumerics board. Each pixel can contain a maximum of three attribute bits (labeled A, B, and C). These attribute bits are converted by a palette look-up table to three colors - red, blue, and green.

Aspects of the graphics video controller board described in this section include:

* Pixel addressing
* Color selection
* Timing and synchronization
* Graphics logic array program

3.5.1 Pixel Addressing

Each dot on the graphics screen is a pixel. Each pixel has a 3-bit value associated with it that selects one of eight palettes (0 - 7). Each palette is assigned one of eight colors, as determined by the contents of the latch. The latch is simply an array of eight 3-bit values. The palette number of each pixel is an index into that array. So, the color of a pixel is the color value of the latch entry that corresponds to the palette number of the pixel. Changing either the palette or the color assigned to the palette changes the color of that pixel. Changing the color assigned to a palette changes the color of every pixel with the same palette number.

A plane is a block of memory containing 1 bit for each pixel in the display. Each of the 3 bits assigned to a pixel is in a different plane. All three planes are formatted identically; only the segment address differs from plane to plane. The segment addresses of the three planes are C000, C800, and D000. For example, if a bit assigned to pixel (x, y) is the fifth bit of memory location C000:mmmm, then the other two bits assigned to that pixel are the fifth bits of locations C800:mmmm and D000:mmmm.

In the following explanation, memory addresses refer to offsets into the segment of any of the three graphics planes. The diagram below shows the organization of graphics screen memory into pixels. Pixels are numbered (x coordinate, y coordinate) and are zero relative.

```
Byte
Address   | Pixels Represented
            ---------------------------------------------------------------
0000-005B |(8,0) - (15,0)|(0,0) - (7,0)|(24,0) - (31,0)|(16,0) - (23,0)
005C-00B7 |(8,1) - (15,1)|(0,1) - (7,1)| . . . . .
            .                .                 .
            .                .                 .
            .                .                 .
```

Pixel (0,0) is the MSB of location 0001.
The LSB of location 0001 is pixel (7,0).
Pixel (8,0) is the MSB of location 0000.
The LSB of location 0000 is pixel (15,0).
Pixel (16,0) is the MSB of location 0003.

The bytes are flip-flopped in this way so that if a move instruction i executed from a word in the graphics plane to a word register, the register then contains 16 consecutive pixel bits in order from MSB to LSB. For example, if a MOV AX, ES:0000 is executed (where ES contains the segment address of the desired graphics plane), the MSB of AX is pixel (0,0) and the LSB is pixel (15,0). With this scheme, 45 words are necessary to represent the 720 pixels in each row of the display. There is one unused word at the end of each line, so a new row begins every 46 words, or 92 bytes. Line one (zero-relative) begins at byte address 92 decimal, 005CH. Therefore, pixel (0,1) is the MSB of location 005DH and pixel (8,1) is the MSB of location 005CH (because the bytes are flip-flopped).

Example:

To find the values of the rightmost 16 pixels on the bottom line of the display,

```
     299 (zero-relative number of last line on display)
X     92 (bytes per line)
+     88 (first word = 0, second word = 2, so 45th word = 88)
--------
= 27596  (6BCC hex)
```

So, MOV AX, ES:6BCC puts the values of the last 16 pixels on the display in AX, with the LSB of AX being the pixel in the lower right corner.

The three graphics planes are named A, B, and C. The segment addresses of the planes A, B, and C are C000, C800, and D000, respectively. In determining the palette number of a pixel, the bit from the C plane is the most significant, the bit from the A plane is the least significant, and the B plane bit is in the middle.

Example:

To find the color of the pixel in the lower right corner of the display, first find the palette number assigned to it.

The MSB of the palette number is the LSB of D000:6BCC;
the middle bit of the palette number is the LSB of C800:6BCC;
the LSB of the palette number is the LSB of C000:6BCC

Say, for example, that these three bits are 1, 0, and 1, respectively. Then the color of the lower right pixel is whatever color is assigned to palette 5. If the default color assignments are in effect, the color of the pixel is cyan.

3.5.2 Color Selection

Each of the eight entries in the latch has one bit for each of the three primary colors: green, red, and blue. The eight available colors are formed by combinations of those three colors, as listed in Table 3-12.

Table 3-12 Color Combinations

Green	Red	Blue	Color	Color
0	0	0	black	0
0	0	1	blue	1
0	1	0	red	2
0	1	1	magenta	3
1	0	0	green	4
1	0	1	cyan	5
1	1	0	yellow	6
1	1	1	white	7

To access the latch, you must write all eight bits of a particular primary color to the appropriate memory location for that color. You cannot change all three bits corresponding to one palette number in a single write. The latch consists of three memory locations, one for each of the primary colors. These locations are:

Blue latch	DF00:0010
Green latch	DF00:0020
Red latch	DF00:0030

You can write to these locations, but you cannot read from them. For this reason, it is necessary to maintain a memory image of the three color latches ifindividual palettes are to be changed. You are then able to change a single palette by setting the appropriate bits in the memory image to the desired value and updating all three color latches.

Each of the three color bits of a palette is in the same bit position in all three color latches. However, the scheme for determining which bit in the latch is addressed by a pixel is not the same as that for determining the palette number. In determining the latch bit addressed by the three-bit value assigned to a pixel, the B plane value is the most significant and the C plane value is in the middle. The A plane value is still the least significant. Bit 7 is the MSB and bit 0 is the LSB of the color latch byte. Table 3-13 displays the correspondence between the bits assigned to a pixel and the bit positions in any of the three color latches, and shows the comparison of these bit positions to the palette numbers.

Table 3-13 Bit Correlations

B Plane Bit	C Plane Bit	A Plane Bit	Latch Bit Addressed	Palette Number
0	0	0	0	0
0	0	1	1	1
0	1	0	2	4
0	1	1	3	5
1	0	0	4	2
1	0	1	5	3
1	1	0	6	6
1	1	1	7	7

Figure 3-6 shows this correspondence horizontally, so that the color latch byte appears as a byte register.

B plane bit	1	1	1	1	0	0	0	0
C plane bit	1	1	0	0	1	1	0	0
A plane bit	1	0	1	0	1	0	1	0
Latch bit addressed	7	6	5	4	3	2	1	0
Palette number	7	6	5	4	3	2	1	0

2223216-18

Figure 3-6 Color Latch Byte

Example

This example shows how to create a memory image of the default values of the three color latches.

Combining information from Table 3-12 (the Color Combinations table), with information from Table 3-13 (the Bit Correlations table), yields the information necessary to construct Table 3-14.

Table 3-14 Default Values of Color Latches

Latch Bit	Palette Number (= Color Number)	Green Bit	Red Bit	Blue Bit
7	7 (white)	1	1	1
6	6 (yellow)	1	1	0
5	3 (magenta)	0	1	1
4	2 (red)	0	1	0
3	5 (cyan)	1	0	1
2	4 (green)	1	0	0
1	1 (blue)	0	0	1
0	0 (black)	0	0	0

The default condition is palette number = color number; therefore, the color latches are set as follows:

Green latch = 11001100 binary = CC hexadecimal at DF00:0020
Red latch = 1111000 binary = F0 hexadecimal at DF00:0030
Blue latch = 10101010 binary = AA hexadecimal at DF00:0010

Example:

This example lists the steps necessary to change palette three to yellow from the default condition (magenta).

1. Find the desired palette number (three) in Table 3-14, then find the associated latch bit (five).

2. Find the desired color (yellow) in Table 3-14, then find the bit settings (red = 1, green = 1, blue = 0).

3. Set bit five in each of the color latches to the values determined in the previous step. This change creates the new values:

   Green latch = 11101100 binary = EC hexadecimal
   Red latch = 11110000 binary = F0 hexadecimal
   Blue latch = 10001010 binary = 8A hexadecimal.

4. Write the new values (from the previous step) to the three color latch addresses. (In this example, it is not necessary to change the red latch, because the value did not change.)

3.5.3 Timing and Synchronization

The same dot clock that generates internal timing for the CRT controller board clocks the graphics video controller board. Monitoring the display enable (DE) signal from the CRT controller board helps to synchronize the pixel outputs from the two boards. If the DE signal has been low for a long period, the graphics board assumes that the scan is in the vertical interval. When DE goes high again, the graphics board resets the graphic memory and scan counters to zero. When DE is low for a short period (horizontal retrace, for example), the scan counters are stopped. This places the last pixel on a line adjacent to the first pixel on the following line.

The graphics video controller board gives the CPU essentially free access to the screen memory. During a single screen display cycle, the hardware can access the refresh memory twice -- once to read the data for screen display, and once for the CPU to read or write data if needed. To provide enough time for this access, a display cycle accesses 16 adjacent pixels of 3 attribute bits each. These are read in parallel and loaded into three 16-bit shift registers for display. After the memory has been read for screen display, the CPU access cycle starts when a read or write cycle is requested. The accessed memory is broken up into one of six separate bytes by properly decoding the enabling of bus buffers and write enable signals to the memory.

Dynamic memory is used on the graphics video board because of the large amount of memory required. The memory chips are organized into 16k x 4 bits and are packaged in an 18-pin, dual inline package (DIP). The 8 address lines are multiplexed into 256 row addresses and 64 column addresses to get to the 16 K locations in the memory. The addresses to the RAM also need to be multiplexed between the CPU and the refresh counter. Performing this four-way multiplexing are four 74LS153 dual 4-to-1 multiplexers (U33 through U36).

Figure 3-7 is a timing diagram for the graphics video controller board. A 74LS163 4-bit counter (U39) and a HAL16R8A-1 logic array (U41) generate the timing. A 74LS163 counter connected as a one-shot (U40), a 75LS00 gate (U44), and a 74LS04 gate (U45) provide the stop, start, and reset logic for the refresh counter.

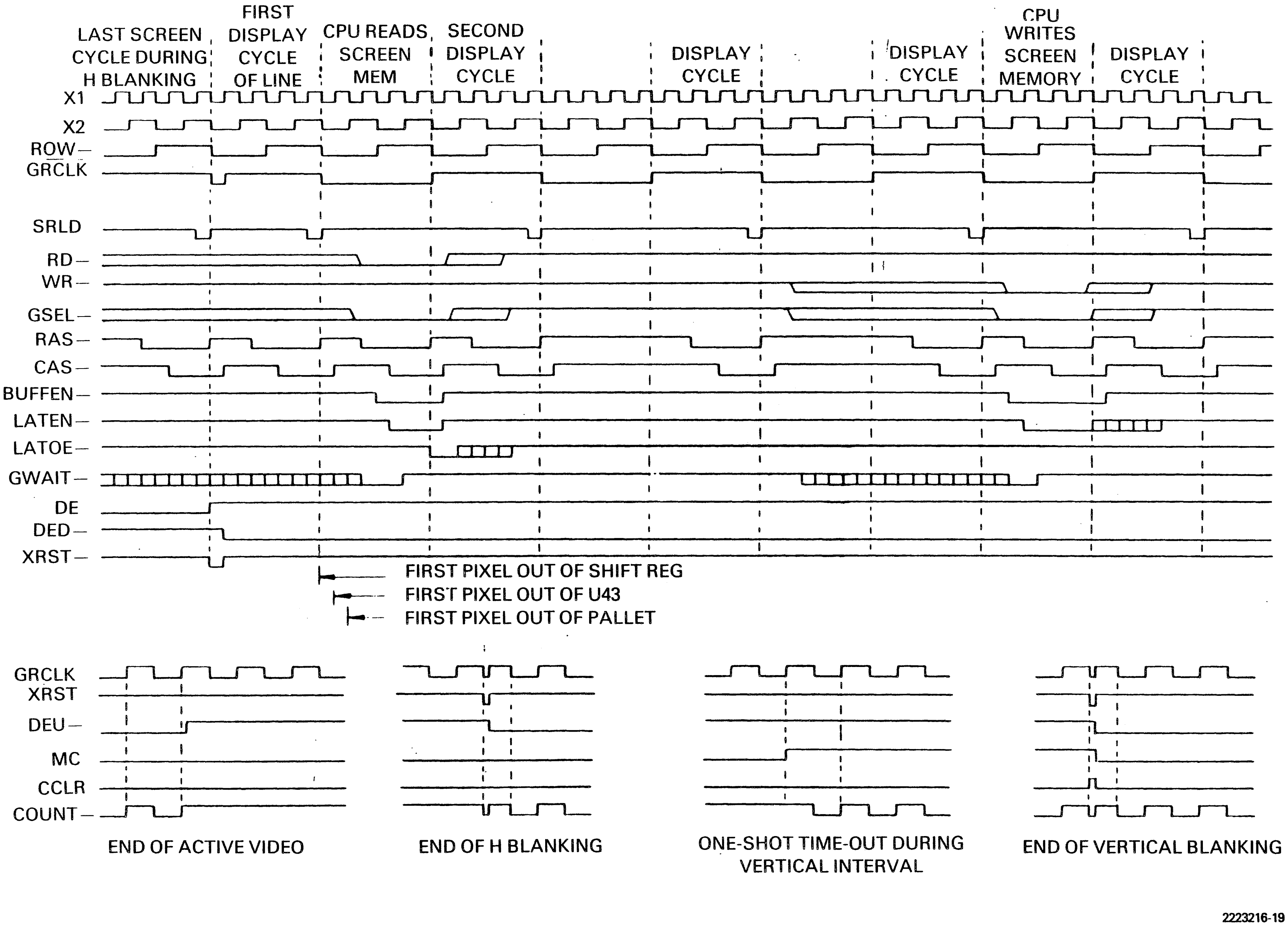


Figure 3-7 Graphics Video Controller Timing Diagram

3.5.4 Graphics Logic Array Program

Programming for the logic array is given in Table 3-15.

Table 3-15 Programming for the Graphics State Machine HAL

Output	RD-	WR-	GSEL-	DE	X1	X2	ROW-	GRCLK	LATEN-	LATOE-	RAS-	CAS-	BUFEN-	SRLD-	GWAIT-	DED-	Comment
LATEN-	L	.	L	.	.	.	H	L	.	.	L	.	.	.	.	.	Read S5,6,7,8
or	.	L	L	.	.	.	.	L	.	.	.	.	L	.	.	.	Write S3
or	.	L	L	.	.	.	.	.	L	.	.	.	.	.	.	.	Write S4 till not write
or	L	L	.	.	.	.	.	.	.	.	.	.	.	.	.	.	All other ORs inactive
LATOE-	L	.	L	.	H	H	H	L	.	.	L	.	.	.	.	.	Read S8
or	L	.	L	.	.	.	.	.	.	L	.	.	.	.	.	.	Read S9 till not read
or	L	L	.	.	.	.	.	.	.	.	.	.	.	.	.	.	All other ORs inactive
RAS-	.	.	.	.	L	H	L	H	.	.	.	.	.	.	.	.	Refresh screen S11
or	.	L	L	.	.	.	L	L	.	.	.	.	L	.	.	.	Write S3
or	L	.	L	.	L	H	L	L	.	H	.	.	.	.	.	.	Read S3
or	.	.	.	.	H	H	L	.	.	.	L	.	.	.	.	.	CPU S4, refresh S12
or	.	.	.	.	.	L	H	.	.	.	L	.	.	.	.	.	CPU S5,6, refresh S13,14
or	.	.	.	.	L	H	H	.	.	.	L	.	.	.	.	.	CPU S7, refresh S15
or	L	L	.	.	.	.	.	.	.	.	.	.	.	.	.	.	(Inactive term)
CAS-	.	.	.	.	.	.	H	.	.	.	L	.	.	.	.	.	S13,14,15,0,5,6,7,8
or	.	.	.	.	.	.	H	.	.	.	L	.	.	.	.	.	All other ORs.
BUFEN-	L	.	L	.	.	.	.	L	.	.	L	.	.	.	.	.	Read S4,5,6,7,8
or	.	L	L	.	H	L	L	L	H	.	.	.	.	.	.	.	Write S2
or	.	L	L	.	.	.	.	L	.	.	.	.	L	.	.	.	Write S3,4,5,6,7,8
or	L	L	.	.	.	.	.	.	.	.	.	.	.	.	.	.	All other ORs inactive
SRLD-	.	.	.	.	L	H	H	H	.	.	.	.	.	.	.	.	S15
or	L	L	.	.	.	.	.	.	.	.	.	.	.	.	.	.	All other ORs inactive
GWAIT-	L	.	L	.	.	.	.	.	H	H	.	.	.	.	.	.	Read
or	.	L	L	.	.	.	.	.	H	.	.	.	.	.	.	.	Write
or	L	L	.	.	.	.	.	.	.	.	.	.	.	.	.	.	All other ORs inactive
DED-	.	.	.	H	.	.	.	.	.	.	.	.	.	.	.	.	Delayed DE
or	.	.	.	H	.	.	.	.	.	.	.	.	.	.	.	.	All other ORs.

Legend:
L = Low signal.
H = High signal.

When the logical AND of terms from one row of Table 3-15 is ORed with the AND of terms from another row, the output goes low when the result is true.

3.6 WINCHESTER DISK DRIVE AND CONTROLLER OPTION

The Winchester disk drive and controller board option consists of a controller board, cable and hardware, and a 5- or 10-megabyte Winchester drive. Aspects of this option described in the following paragraphs include:

* Winchester hardware theory of operations
* Register assignments
* Bit definitions for registers and ports
* Controller status bit combinations
* Normal command sequence operation

3.6.1 Winchester Hardware Theory of Operation

The Winchester controller is addressed by the 8088 as a block of four I/O ports: 0030H through 0033H. I/O reads are indicated by the bus signal IORC, and I/O writes are indicated by the bus signal AIOWC-.

The controller can generate an interrupt to the host under one of the following conditions:

* When data is ready to be read from or written to the controller
* When the operation is completed, and the controller is requesting a status read (C/D- = 1, I/O = 1)

Both of the interrupt conditions can be individually disabled. When the interrupt is active, the computer's interrupt line 6 is held high until it is cleared by a read to the controller status register.

3.6.1.1 On-Board EPROM/ROM. A 4K x 8-bit EPROM/ROM contains the driver routines for the controller. Addressing this device causes the output to drive the data bus through a tristate buffer. The EPROM/ROM is at memory address 0F8000H. Access time to either the EPROM or the ROM is less than 350 ns.

3.6.1.2 Commands and Command Testing. The computer sends a 6-byte block to the controller to specify the operation. This block is the device control block (DCB). Table 3-16 gives the bit definition for the DCB.

Table 3-16 Device Control Block Bit Diagram

Byte	BIT NUMBER 7	6	5	4	3	2	1	0
0	COMMAND CLASS			OPCODE				
1	LOGICAL UNIT NUMBER			HIGH ADDRESS (See Note 1)				
2	MIDDLE ADDRESS (See Note 1)							
3	LOW ADDRESS (See Note 1)							
4	INTERLEAVE OR NUMBER OF BLOCKS (See Note 2)							
5	CONTROL FIELD							

Notes:
1. Refer to paragraph 3.6.1.6.
2. Interleave factor for FORMAT, CHECK TRACK, and READ ID commands.

3.6.1.3 Explanation of Bytes in the Device Control Block. The 6 bytes that comprise the device control block are defined as follows:

Byte	Definition
0	Bits 7, 6, and 5 identify the class of the command. Bits 4 through 0 contain the opcode of the command.
1	Bits 7, 6, and 5 identify the logical unit number (LUN). Bits 4 through 0 contain logical disk address 2.
2	Bits 7 through 0 contain logical disk address 1.
3	Bits 7 through 0 contain logical disk address 0.
4	Bits 7 through 0 specify the interleave or block count.
5	Bits 7 through 0 contain the control field.

3.6.1.4 Control Field Detailed Description. Byte 5, the control field of the DCB, allows the user to choose options for several different types and makes of disk drives. The following listing defines the bits of the control byte. The step options are encoded in control byte 5 of the command descriptor. The encoding is done with bits 0 through 3 as given in Table 3-17.

Table 3-17 Command Descriptor Byte

Description	Bit No. 3	2	1	0
Default 3-ms step rate	0	0	0	0
Seagate ST506 (MLC2)	0	0	0	1
Tandon fast-step	0	0	1	0
Texas Instruments fast-step	0	0	1	1
200-us buffered-step	0	1	0	0
70-us buffered-step	0	1	0	1
30-us buffered-step	0	1	1	0
15-us buffered-step	0	1	1	1
Olivetti 2 ms/step (561)	1	0	0	0
Olivetti (562) fast-step (1.1 ms typical)	1	0	0	1
Spare (for future use)	1	1	1	1

To configure a drive for fast-step or buffered-step, refer to the manufacturer's manual for instructions. If the drive is hardware-configured for fast-step, all commands requiring the seek option selection must use the fast-step option for that drive.

NOTE

> The step option bits (3 through 0) are mutually exclusive. Select only one option for any configuration.

Bits 4 and 5 are reserved for future use.

Set bit 6 to 0 for regular operation. When this bit is set to 1 during a read sector command, any failing sectors are not reread on the next revolution.

Set bit 7 to 0 for regular operation. Setting this bit to 1 disables the four retries by the controller on all disk-access commands. Set bit 7 to 1 only during the performance evaluation of a disk drive.

3.6.1.5 Command Completion Status Byte. At the end of a command, the controller returns a completion status byte to the computer. This byte indicates whether or not an error has occurred during command execution. (If the error bit is set, and you want to know what caused the error, you must send the REQUEST SENSE STATUS command.)
The format of the completion status byte is :

I/O Port Address 0030 (read)	7 (MSB)	6	5	4 (Bit Number)	3	2	1	0 (LSB)
	Don't care	Don't care	Drive No.	Don't care	Don't care	Don't care	Error bit	Don't care

3.6.1.6 Logical Address (HIGH, MIDDLE and LOW). The logical address of the drive is computed by using the following equation:

Logical Address = (CYADR x HDCYL + HDADR) x SETRK + SEADR

Where: CYADR = Cylinder address
HDCYL = Number of heads per cylinder
HDADR = Head address
SETRK = Number of sectors per track
SEADR = Sector address

3.6.1.7 Sector Interleaving. The disk controller supports variable sector interleaving. When a format command is issued, an interleave value can be passed in byte 4 of the device control block (DCB). The maximum interleave value is the number of sectors per track minus 1. When transferring multiple data sectors, the interleave factor can be adjusted to achieve maximum system performance.

The practice of interleaving involves mapping logical continuous sectors of data from a given track onto nonadjacent physical sectors. For example, an interleave factor of 5 means that every fifth physical sector is transferred as the next logical continuous data sector. It does not mean that five sectors of data are transferred on one revolution.

If the interleave factor is too low, the CPU cannot transfer the full sector of data during the sector-interleave time available. The controller has to wait one full revolution before reading the next logical sector from the disk. Increasing the interleave factor increases the system's operating speed.

The operating system should perform multiple-sector data transfers to take full advantage of the controller's interleaving feature. In single-sector transfers, the differences in speed between various interleave factors is probably not noticeable.

3.6.2 Register Assignments

The register assignments for the I/O ports of the Winchester controller are given in Table 3-18.

Table 3-18 Winchester Controller I/O Port Assignment

Address	Functions	
	In	Out
0030H	Data IN port	Data OUT port
0031H	Status register	RESET
0032H	Not used	Not used
0033H	Not used	Interrupt mask

An IN function gets data from the Winchester controller board and puts it on the computer's I/O expansion bus. Conversely, an OUT function sets data from the computer's I/O expansion bus onto the Winchester disk controller board.

For byte definitions of the registers, refer to the I/O memory map given in Table 2-1.

For pin-outs of the Winchester cable, refer to paragraph 3.6.20, Electrical Interface.

3.6.2.1 Data Input Port. Disk read data and controller sense bytes pass through this register to the computer. The data is held for each handshake cycle. The format is as follows:

I/O Port Address	(MSB) 7	6	5	4	3	2	1	(LSB) 0
0030	DATA 7	DATA 6	DATA 5	DATA 4	DATA 3	DATA 2	DATA 1	DATA 0

3.6.2.2 Data Output Port. Command bytes and disk data pass through this register to the controller. Data is latched until updated by the CPU. The bit arrangement is as follows:

I/O Port Address	MSB 7	6	5	4	3	2	1	LSB 0
0030 (write)	DATA 7	DATA 6	DATA 5	DATA 4	DATA 3	DATA 2	DATA 1	DATA 0

3.6.2.3 Controller Status Register. This register stores the controller status. It enables the CPU to read the controller status and to monitor the controller operation. The controller status byte is defined as follows:

I/O Port Address	MSB 7	6	5	4	3	2	1	LSB 0
0031 (read)	Don't care	Don't care	Don't care	Don't care	Don't care	COMMAND /DATA	INPUT/ OUTPUT	DATA REQUEST

3.6.2.4 Reset Port. This byte resets the controller. Any write to port 0031 causes a reset. Reset clears each error status, aborts all operations, and places the Winchester controller in the command receive mode. The byte definition follows:

I/O Port Address	MSB 7	6	5	4	3	2	1	LSB 0
0031 (write)	Don't care	Don't care	Don't care	Don't care	Don't care	Don't care	Don't care	Don't care

3.6.2.5 Interrupt Mask. This is a 2-bit field that determines which interrupts are to be serviced by the CPU. The interrupt mask byte definition follows:

I/O Port Address 0033	7 (MSB)	6	5	4	3	2	1	0 (LSB)
	Don't care	Don't care	Don't care	Don't care	Don't care	Don't care	DATA INTR. ENABLE	STATUS INTR. ENABLE

3.6.2.6 Error Status Byte. This special byte is available only after the completion of a command. The controller sets the I/O and C/D bits with DRQ to indicate that this byte is available. A definition of the error status byte follows:

I/O Port Address 0030 (read)	7 (MSB)	6	5	4	3	2	1	0 (LSB)
	Don't care	Don't care	Drive No.	Don't care	Don't care	Don't care	Error bit	Don't care

3.6.3 Bit Definitions for Registers and Ports

Table 3-19 gives the definitions of bits for the Winchester controller registers and ports.

Table 3-19 Bit Definitions for Controller Registers and Ports

	Logical State	
Data Bit	Data true ; data high ; logical one >= 2.4 V	Data false ; data low ; logical zero <= 0.7 V
DATA 0-7 READ or WRITE	Data bit = 1	Data bit = 0
DATA REQUEST	Commands, status, or data ready to be transferred to or from controller.	No command, status, or data transfers to or from controller.
INPUT/ OUTPUT-	The CPU reads data or status from the controller.	The CPU writes data or commands to the controller.
COMMAND/ DATA-	When INPUT/OUTPUT- is high, status is sent to the CPU.	When INPUT/OUTPUT- is high, data is sent to the CPU.
	When INPUT/OUTPUT- is low, commands are sent to the controller.	When INPUT/OUTPUT- is low, data is sent to the controller.
STATUS INTERRUPT ENABLE	Controller interrupts the CPU after the CPU completes the current command and is ready to return the status byte.	No status interrupt permitted.
DATA INTERRUPT ENABLE	Controller interrupts the CPU when data needs to be read from or written to the controller.	No data interrupt permitted.

3.6.4 Controller Status Bit Combinations

Table 3-20 gives all valid controller status bit combinations.

Table 3-20 Valid Bit Combinations for Controller Status

COMMAND/ DATA	INPUT/ OUTPUT	DATA REQUEST	Meaning of Pattern
0	0	0	Not valid
0	0	1	A data byte may be sent from the CPU to the Winchester controller. The controller waits for data to be written.
0	1	0	Not valid
0	1	1	A data byte may be sent to the CPU from the Winchester controller. The controller waits until data is read.
1	0	0	Not valid
1	0	1	Command bytes may be sent to the Winchester controller from the CPU.
1	1	0	Not valid
1	1	1	A status byte may be sent from the Winchester controller to the CPU.

3.6.5 Normal Command Sequence Operation

Figure 3-8 depicts the logical flow of the controller functions.

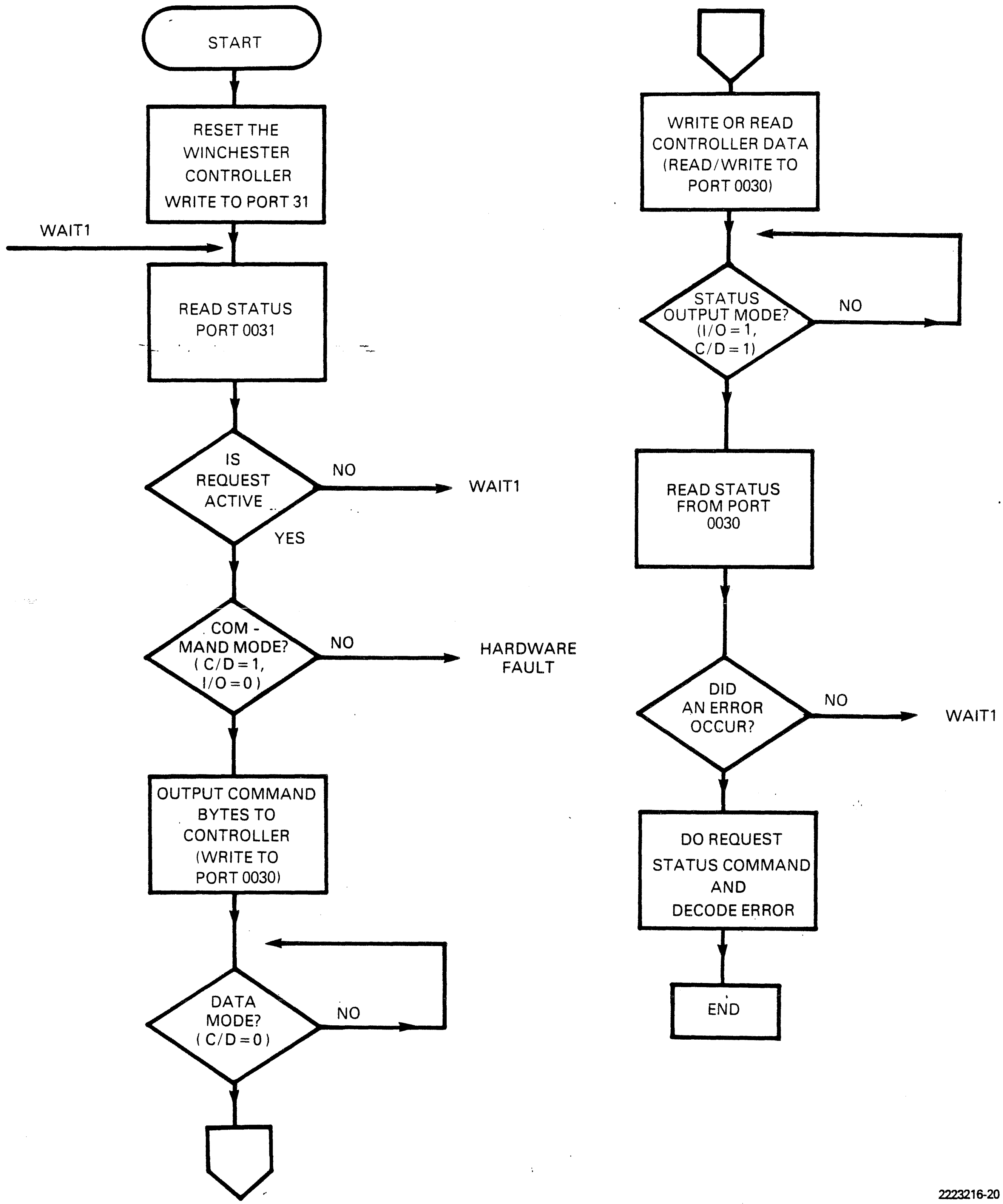


Figure 3-8 Controller Operational Flowchart

3.6.6 Detailed Description of Commands

The commands fall into eight classes -- 0 through 7; however, only classes 0 and 7 are used. Classes 1 through 6 are reserved. Class 0 commands are data, non-data transfer, and status commands. Class 7 commands perform diagnostics.

Each command is described in the following paragraphs. The command description includes class, opcode, and format. "Don't care" bits are shown as "unused."

3.6.6.1 TEST DRIVE READY Command. This command selects a particular drive and verifies that the drive is ready. The following diagram shows the format of the device control block for this command:

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

To determine that a drive has completed seeking before issuing the next command, use the TEST DRIVE READY command with overlapped seeks. (Refer to the paragraph entitled "SEEK Command" in this section.) If the drive is still seeking, the end-of-command status byte indicates an error, and the sense status indicates "drive still seeking." This is a type 0 error, code 8. Sequential TEST DRIVE READY commands determine when the drive is ready to accept another command.

3.6.6.2 RECALIBRATE DRIVE Command. This command places the read/write (R/W) arm at track 000. Bit definitions for this command are as follows:

Byte	7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
0	0	0	0	0	0	0	0	1
1	0	0	DRIVE	unused	unused	unused	unused	unused
2	unused	unused	unused	unused	unused	unused	unused	unused
3	unused	unused	unused	unused	unused	unused	unused	unused
4	unused	unused	unused	unused	unused	unused	unused	unused
5	RETRY?	0	0	0	STEP 3	STEP 2	STEP 1	STEP 0

3.6.6.3 REQUEST SENSE STATUS Command. The computer sends this command immediately after it detects an error. The controller then returns 4 bytes of drive and the controller status. The formats for these 4 bytes are shown after the DCB. Definitions of these bytes follow.

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

Bit 7, the address valid bit in the error code byte, is relevant only when the previous command required a logical block address. In this case, it is always returned as a 1; otherwise, it is set to 0. For instance, assume that a RECALIBRATE command is followed immediately by a REQUEST SENSE STATUS command. The address valid bit could be returned as 0 because the command does not require a logical block address to be passed in its DCB.

The format for the sense bytes returned is as follows:

Byte	7	6	5	4	3	2	1	0
0	ADDRESS?	0	ERROR TYPE		ERROR CODE			
1	0	0	DRIVE	HIGH ADDRESS (see Note)				
2	MIDDLE ADDRESS (see Note)							
3	LOW ADDRESS (see Note)							

NOTE: Refer to paragraph 3.6.1.6.

When an error occurs on a multiple-sector data transfer (read or write), the REQUEST SENSE STATUS command returns the logical address of the failing sector in bytes 1, 2, and 3. If the REQUEST SENSE STATUS command is issued after any of the format commands or the CHECK TRACK FORMAT command, and if no error exists, the logical address returned by the controller points to one sector beyond the last track formatted or checked. If an error does exist, the logical address returned points to the track in error. Table 3-21, Table 3-22, and Table 3-23 list the types 0, 1, 2, and 3 error codes. Table 3-24 summarizes the error codes returned by the REQUEST SENSE STATUS command.

Table 3-21 Type 0 Error Codes, Winchester Disk

Code	Definition
0H	The controller detected no error during the execution of the previous operation.
1H	The controller did not detect an index signal from the drive.
2H	The controller did not get a SEEK COMPLETE signal from the drive after seek operation.
3H	The controller detected a write fault from drive during last operation.
4H	After the controller selected the drive, the drive did not respond with READY signal.
5H	Not used.
6H	After stepping maximum number of cylinders, controller did not receive track 00 signal from the drive.

Table 3-22 Type 1 Error Codes, Controller Board

Hex Code	Message	Definition
0H	ID Read Error	The controller detected an ECC error in the target ID field on the disk.
1H	Data Error	The controller detected an uncorrectable ECC error in the target sector during a read operation.
2H	Address Mark	The controller did not detect the target address mark (AM) on the disk.
3H	Not used.	
4H	Sector Not Found	The controller found the correct cylinder and head, but not the target sector.
5H	Seek Error	The controller detected an incorrect cylinder or track, or both.
6H	Not used.	
7H	Not used.	
8H	Correctable Data Error	The controller detected a correctable ECC error in the target data field.
9H	Bad Track	The controller detected the bad track flag during the last operation.
AH	Format Error	During a CHECK TRACK FORMAT command, the controller detected one of the following: * Track not formatted * Wrong interleave * ID ECC error on at least one sector

Table 3-23 Types 2 and 3 Error Codes, Command and Miscellaneous

Code	Type	Message	Definition
0H	2	Invalid Command	The controller received an invalid command from the host.
1H	2	Illegal Disk Address	The controller detected an address beyond the maximum range.
0H	3	RAM Error	The controller detected a data error during the RAM sector buffer diagnostic.
1H	3	Program Memory Checksum Error	During its internal diagnostics, the controller detected a program memory checksum error.
2H	3	ECC Polynominal Error	During the controller's internal diagnostics, the hardware ECC generator failed its test.

Table 3-24 Error Code Summary

Error Code	Meaning
00H	No error detected (command completed OK).
01H	No index detected from disk drive.
02H	No seek complete from disk drive.
03H	Write fault from disk drive.
04H	Drive not ready after it was selected.
05H	Not used.
06H	Track 00 not found.
07H-0FH	Not used.
10H	ID field read error.
11H	Uncorrectable data error.
12H	Address mark not found.
13H	Not used.
14H	Target sector not found.
15H	Seek error.
16H-17H	Not used.
18H	Correctable data error.
19H	Bad track flag detected.
1AH	Format error.
1BH	Not used.
1CH	Illegal (direct) access to an alternate track.

Table 3-24 Error Code Summary (Concluded)

Error Code	Meaning
1DH	On a FORMAT ALTERNATE TRACK command, the track is already assigned or is flagged as a bad track.
1EH	When the controller attempted to access an alternate track from a spared track, the alternate track was not flagged as an alternate.
1FH	On a FORMAT ALTERNATE TRACK command, the bad track equaled the alternate track.
20H	Invalid command.
21H	Illegal disk address.
22H-2FH	Not used.
30H	Ram diagnostic failure.
31H	Program memory checksum error.
32H	ECC diagnostic failure.
33H-3FH	Not used.

Note: The Address Valid bit (bit 7) may or may not be set and is not included here.

3.6.6.4 FORMAT DRIVE Command. This command uses the selected interleave factor to format all sectors having ID and data fields, and writes 6CH into data fields. The controller formats from the starting address, which is passed in the command, to the end of the disk.

Setting bit 5 (from control byte 5 of the command block) with the FORMAT DRIVE command causes the sector buffer to be used as the data pattern written on the disk data fields.

To initialize the sector buffer, issue the WRITE SECTOR BUFFER command before the FORMAT DRIVE command. Byte definitions are as follows:

Byte	BIT NUMBER 7	6	5	4	3	2	1	0
0	0	0	0	0	0	1	0	0
1	0	0	DRIVE	HIGH ADDRESS (Note 1)				
2	MIDDLE ADDRESS (Note 1)							
3	LOW ADDRESS (Note 1)							
4	0	0	0	INTERLEAVE FACTOR (Note 2)				
5	RETRY?	0	0	0	STEP 3	STEP 2	STEP 1	STEP 0

Notes:

1. Refer to paragraph 3.6.1.6.
2. Factor is number of sectors per track minus one.

3.6.6.5 CHECK TRACK FORMAT Command. This command checks the format on the specified track for correct ID and interleave. The command does not read the data field. The byte configuration is as follows:

Byte	7	6	5	4	3	2	1	0
0	0	0	0	0	0	1	0	1
1	0	0	DRIVE	HIGH ADDRESS (See note 1)				
2	MIDDLE ADDRESS (See note 1)							
3	LOW ADDRESS (See note 1)							
4	0	0	0	INTERLEAVE FACTOR (See note 2)				
5	RETRY?	0	0	0	STEP 3	STEP 2	STEP 1	STEP 0

Notes:
1. Refer to paragraph 3.6.1.6.
2. Factor is number of sectors per track minus one.

3.6.6.6 FORMAT TRACK Command. The FORMAT TRACK command reformats the track, eliminating all references to bad and alternate tracks. Setting bit 5 from control byte 5 of the command block causes the sector buffer to be used as the data pattern in the data fields. Otherwise, the command writes 6CH in the data fields. The byte definitions are as follows:

Byte	BIT NUMBER 7	6	5	4	3	2	1	0
0	0	0	0	0	0	1	1	0
1	0	0	DRIVE	HIGH ADDRESS (See note 1)				
2	MIDDLE ADDRESS (See note 1)							
3	LOW ADDRESS (See note 1)							
4	0	0	0	INTERLEAVE FACTOR (See note 2)				
5	RETRY?	0	0	0	STEP 3	STEP 2	STEP 1	STEP 0

Notes:
1. Refer to paragraph 3.6.1.6.
2. Factor is number of sectors per track minus one.

3.6.6.7 FORMAT BAD TRACK Command. This command formats a specified track, setting the bad sector flag in the ID fields. No data fields are written. The byte definitions are as follows:

Byte	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
0	0	0	0	0	0	1	1	1
1	0	0	DRIVE	HIGH ADDRESS (See note 1)				
2	MIDDLE ADDRESS (See note 1)							
3	LOW ADDRESS (See note 1)							
4	0	0	0	INTERLEAVE FACTOR (See note 2)				
5	RETRY?	0	0	0	STEP 3	STEP 2	STEP 1	STEP 0

NOTES:

1. Refer to paragraph 3.6.1.6.
2. Factor is number of sectors per track minus one.

3.6.6.8 READ Command. Starting with the sector address given in this command, the controller reads a specified number of sectors. The byte definitions are as follows:

Byte	7	6	5	4	3	2	1	0
0	0	0	0	0	1	0	0	0
1	0	0	DRIVE	HIGH ADDRESS (See note 1)				
2	MIDDLE ADDRESS (See note 1)							
3	LOW ADDRESS (See note 1)							
4	BLOCK COUNT							
5	RETRY?	Note 2	0	0	STEP 3	STEP 2	STEP 1	STEP 0

Notes:

1. Refer to paragraph 3.6.1.6.
2. If this bit is set in the READ command and an ECC error is found, retry the command.

3.6.6.9 WRITE Command. This command writes the specified number of sectors, starting with the initial sector address contained in the DCB. Byte definitions are as follows:

Byte	7	6	5	4	3	2	1	0
0	0	0	0	0	1	0	1	0
1	0	0	DRIVE	HIGH ADDRESS (See note)				
2	MIDDLE ADDRESS (See note)							
3	LOW ADDRESS (See note)							
4	BLOCK COUNT							
5	RETRY?	0	0	0	STEP 3	STEP 2	STEP 1	STEP 0

Note: Refer to paragraph 3.6.1.6.

3.6.6.10 SEEK Command. This command initiates a seek to the track specified in the DCB. The drive must be formatted. The byte definitions are as follows:

Byte	7	6	5	4	3	2	1	0
0	0	0	0	0	1	0	1	1
1	0	0	DRIVE	HIGH ADDRESS (See note)				
2	MIDDLE ADDRESS (See note)							
3	LOW ADDRESS (See note)							
4	U N U S E D							
5	RETRY?	0	0	0	STEP 3	STEP 2	STEP 1	STEP 0

Note: Refer to paragraph 3.6.1.6.

For drives using buffered seeks, SEEK commands can be overlapped. After the controller issues a SEEK to the drive, it does not wait for the drive to complete the SEEK, but returns a completion status. If the return status shows no error, then the SEEK was issued correctly. If there is an error, then the SEEK was not issued. After transferring the status, another command can be issued to either drive. If a drive with an outstanding SEEK receives a new command, the controller waits (holding BUSY active) until the SEEK completes before executing the new command. (See the section entitled "TEST DRIVE READY Command" for a special case.) There is no time-out condition in the controller waiting for the buffered-step SEEK to complete.

3.6.6.11 INITIALIZE DRIVE CHARACTERISTICS Command. This command enables the controller to work with drives that have different capacities and characteristics. However, both Winchester drives must be of the same manufacturer and model number.

After the computer sends the command (DCB) to the controller, it sends an 8-byte block of data containing the drive parameters. Some of the parameters occupy 2 bytes; all 2-byte parameters are transferred with the most significant byte (MSB) first. The 8 bytes are:

C = Maximum number of cylinders (2 bytes)
E = Maximum ECC data burst length (1 byte)
H = Maximum number of heads (1 byte)
P = Starting write precompensation cylinder (2 bytes)
W = Starting reduced write current cylinder (2 bytes)

When the controller is powered up or reset, the following default values are set:

Maximum number of cylinders (C)= 153
Maximum ECC data burst length (E)= 11 bits
Maximum number of heads (H)= 4
Starting write precompensation cylinder (P)=64
Starting reduced write current cylinder (W)= 128

The parameter for the maximum ECC burst length defines the length of a burst error in the data field that the controller is to correct. The burst length is defined as the number of bits from the first error bit to the last error bit. For example, if the controller detects a 5-bit ECC error and the erroneous data appears (before correction) as C5 (1100 0101), it could appear as D4 (1101 0100) after the correction. However, if the CPU has set the maximum ECC burst length at 4 bits, the controller might flag this data as uncorrectable. This is a type 1, code 1 error.

Byte definitions for the INITIALIZE DRIVE CHARACTERISTICS command are as follows:

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

Byte definitions for the drive parameter bytes (passed to the controller after the INITIALIZE DRIVE CHARACTERISTICS command has been issued) are as follows:

Byte	7	6	5	4	3	2	1	0
0	MAXIMUM NUMBER OF CYLINDERS: MSB							
1	MAXIMUM NUMBER OF CYLINDERS: LSB							
2	0	0	0	0	MAXIMUM NUMBER OF HEADS			
3	STARTING REDUCED WRITE CURRENT CYLINDER: MSB							
4	STARTING REDUCED WRITE CURRENT CYLINDER: LSB							
5	STARTING WRITE PRECOMPENSATION CYLINDER: MSB							
6	STARTING WRITE PRECOMPENSATION CYLINDER: LSB							
7	0	0	0	0	MAXIMUM ECC DATA BURST LENGTH			

3.6.6.12 READ ECC BURST ERROR LENGTH Command. This command transfers 1 byte to the CPU. This byte contains the value of the ECC burst length that the controller detected during the last READ command. This byte is valid only after a correctable ECC data error, type 1, code 8. Byte definitions are as follows:

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

3.6.6.13 FORMAT ALTERNATE TRACK Command. The FORMAT ALTERNATE TRACK command formats the header fields of the "bad track" with the alternate track information (assigned by the CPU). The alternate track is formatted to identify it as an alternate. The command byte definitions for FORMAT ALTERNATE TRACK are as follows:

Byte	BIT NUMBER 7	6	5	4	3	2	1	0
0	0	0	0	0	1	1	1	0
1	0	0	DRIVE	HIGH ADDRESS (Note 1)				
2	MIDDLE ADDRESS (Note 1)							
3	LOW ADDRESS (Note 1)							
4	0	0	0	INTERLEAVE FACTOR (Note 2)				
5	RETRY?	0	(Note 3)	0	STEP 3	STEP 2	STEP 1	STEP 0

Notes:

1. Refer to paragraph 3.6.1.6.

2. Factor is number of sectors per track minus one.

3. If this bit is set, the data in the existing sector buffer is used to fill the data field. If this bit is cleared, the data field is written with 6CH.

The interleave byte (4) is programmed the same as in the FORMAT command, and is used on the alternate track. If bit 5 of the control byte (5) is set, the data in the existing sector buffer is written to the data field. If not, the data field is written with 6CH.

After issuing the command, the controller asks for the Assigned Alternate Address data block. These 3 bytes point to the CPU-assigned alternate logical address. Again the sector address is ignored.

The byte definitions for the Assigned Alternate Address Data Block are as follows:

Byte	7	6	5	4	3	2	1	0
0	0	0	0	HIGH ADDRESS (See note)				
1	MIDDLE ADDRESS (See note)							
2	LOW ADDRESS (See note)							

Note: Refer to paragraph 3.6.1.6.

3.6.7 Alternate Track Assignment

The computer both assigns alternate tracks and locks out bad tracks. Bad areas on the disk are labeled defective on a track basis by issuing a FORMAT BAD TRACK command (command code 07). One procedure for assignment and handling of alternate tracks is given below.

1. Give the FORMAT DISK command (command code 04). This formats the entire disk drive starting at logical track 000.

   a. If any errors occur, give the REQUEST SENSE STATUS command.

   b. If a format error is indicated, bytes 1, 2, and 3 of the returned status give the address of the bad track.

   c. Give a FORMAT BAD TRACK command (command code 07) to the track.

   d. Reissue the FORMAT DISK command.

   e. If any other errors occur during the subsequent formatting, reissue the REQUEST SENSE STATUS, FORMAT BAD TRACK, and FORMAT DISK commands until the entire disk is formatted.

2. Give the RECALIBRATE command (command code 01) to position the heads over track 000.

All sectors on the disk are read to see if any uncorrectable ECC errors occurred in the data. The FORMAT command places a 6CH pattern in the data fields of all sectors, and the computer program can verify this data pattern after the data is read into memory. However, verifying the data byte for byte is not usually necessary, because the error detection and correction circuitry flags all uncorrectable errors. If a large block of host memory is available, multiple sector reads can be issued to speed up the verify process.

When an uncorrectable error is found, issuing a FORMAT BAD TRACK command (command code 07) to the failing track writes a bad track flag into all identifier fields. Later accessing of this track results in an error, causing the sense status that follows to show an error code 19H.

NOTE

> Whenever a user program accesses the disk, be sure that the operating system does not allow the program to issue a READ or WRITE command to the alternate tracks.

The disk controller has no way of knowing when an alternate track is being read. The alternate tracks are sometimes assigned at the end of the disk (highest track numbers), but they can be assigned to any tracks so long as the track label is maintained by the computer. Given the error correction capability of the controller, four tracks reserved as alternates should be adequate for all disk drives currently available. However, the system programmer should consult the disk drive manual for the hard-defect specifications.

3.6.8 Alternate Address Protocol

After receiving the FORMAT ALTERNATE TRACK command and the assigned alternate, the controller performs the following steps:

1. Seeks to the "alternate assigned track" and verifies that it is not already an assigned alternate or a flagged bad track.

NOTE

> If the track has already been assigned as an alternate or is flagged "bad", then error code 1DH is given and the command is aborted. This usually implies that the computer is attempting to assign two bad tracks to the same alternate track.

2. Formats the track as an assigned alternate track.

3. Seeks to the bad track and formats the header as a spare

track pointing to the assigned alternate.

4. Destroys data fields on both the bad track and alternate track.

The procedure for using the FORMAT ALTERNATE TRACK command is as follows:

1. Format the entire disk, including spare tracks.

2. Verify the disk.

3. Assign each media defect an alternate track.

4. Assign alternate tracks for drive manufacturer's defect list.

The controller automatically seeks to the assigned alternate track when an access is made to a flagged defective track. Consecutive accessing does not result in reseeking to the alternate track. The controller maintains position on the alternate track.

NOTE

> When using the FORMAT ALTERNATE TRACK command, be sure to include (in the controller initialization) cylinder and head ranges for the alternate tracks.

Generally, the actual disk space is greater than the amount fixed by the system software. This extra space can be used for alternate tracks as needed. The alternate tracks are invisible to the host.

The number of spare tracks depends on the drive size and the number of defects allowed by the drive manufacturer. Generally, one spare track is allotted for each 50 to 100 tracks.

Direct access (attempted data transfers or seeks) to an alternate track results in an error code 1CH, and no transfer takes place.

3.6.9 WRITE SECTOR BUFFER Command

This command is used to fill the sector buffer with a host-given data pattern. No data is transfered between the drive and the controller. The command accepts 512 bytes of data and stores them in the sector buffer. The byte definitions are as follows:

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

3.6.10 READ SECTOR BUFFER Command

This command sends 512 bytes of data from the sector buffer to the CPU. The byte definitions are as follows:

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

3.6.11 RAM DIAGNOSTICS Command

This command performs a data pattern test on the RAM buffer. The byte definitions are as follows:

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

3.6.12 DRIVE DIAGNOSTICS Command

This command tests both the drive and the drive-to-controller interface. The controller sends RECALIBRATE and SEEK commands to the selected drive and verifies sector 0 of all the tracks on the disk. The controller does not perform any write operations during the command; it assumes the disk has been previously formatted. The byte definitions for the command are as follows:

Byte	7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
0	1	1	1	0	0	0	1	1
1	unused	unused	unused	unused	unused	unused	unused	unused
2	unused	unused	unused	unused	unused	unused	unused	unused
3	unused	unused	unused	unused	unused	unused	unused	unused
4	unused	unused	unused	unused	unused	unused	unused	unused
5	RETRY?	0	0	0	STEP 3	STEP 2	STEP 1	STEP 0

3.6.13 CONTROLLER INTERNAL DIAGNOSTICS Command

This command causes the controller to perform a self-test. The controller checks its internal processor, data buffers, ECC circuitry, and the checksum of the program memory. The controller does not access the disk drive. The byte definitions are as follows:

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

3.6.14 READ LONG Command

This command transfers the target sector and 4 bytes of data ECC to the CPU. If an ECC error occurs during the read, the controller does not attempt to correct the data field. This command is useful for recovering data from a sector with an uncorrectable ECC error and for diagnostic operations. The byte definitions are as follows:

Byte	7	6	5	4	3	2	1	0
0	1	1	1	0	0	1	0	1
1	0	0	DRIVE	HIGH ADDRESS (See note)				
2	MIDDLE ADDRESS (See note)							
3	LOW ADDRESS (See note)							
4	BLOCK COUNT (See note)							
5	RETRY?	0	0	0	STEP 3	STEP 2	STEP 1	STEP 0

Note: Refer to paragraph 3.6.1.6.

3.6.15 WRITE LONG Command

This command transfers a sector of data and four appended ECC bytes to the disk drive. During this write operation, the computer supplies the 4 ECC bytes instead of using the hardware-generated ECC bytes. This command is useful only for diagnostic operations. The byte definitions are as follows:

Byte	7	6	5	4	3	2	1	0
0	1	1	1	0	0	1	1	0
1	0	0	DRIVE	HIGH ADDRESS (see Note)				
2	MIDDLE ADDRESS (see Note)							
3	LOW ADDRESS (see Note)							
4	BLOCK COUNT							
5	RETRY?	0	0	0	STEP 3	STEP 2	STEP 1	STEP 0

Note: Refer to paragraph 3.6.1.6.

3.6.16 Execution Order of Remaining Diagnostics

Not all of the diagnostics are executed by the computer on power-up. The remaining diagnostics should be called by the CPU in the following order.

1. CONTROLLER INTERNAL DIAGNOSTICS (command code E4). This command tests all the logical and decision-making capabilities of the controller, the program memory checksum, and the error detection and correction circuits (ECC). Executing this diagnostic ensures that the controller can communicate with the computer.

2. RAM DIAGNOSTICS (command code E0). This command verifies that the sector buffer is operational by writing, reading, and verifying various data patterns to and from all locations.

3. INITIALIZE DRIVE CHARACTERISTICS (command code 0C). This command sends the new drive configuration to the controller when the parameters of the connected drives differ from the defaults. The INITIALIZE DRIVE CHARACTERISTICS command must be issued before executing the DRIVE DIAGNOSTIC command.

4. TEST DRIVE READY (command code 00). This command, issued before the DRIVE DIAGNOSTIC is executed, finds out when the drive is ready to accept a command.

5. DRIVE DIAGNOSTIC (command code E3). This command issues a RECALIBRATE to the disk drive and then steps though all tracks, verifying the ECC on the identifier fields of the first sector of each track. If this diagnostic passes, it implies that the disk has been formatted and that the first ID field of each track is good.

3.6.17 Error Correction Philosophy

The typical error-correction time of the controller is approximately 50 ms, which is greater than the time for one revolution of the disk. The sector in error can be reread (if bit 6 is not set in byte 5 of the READ command DCB) on the next revolution during a READ command. In most cases, the error is soft and does not reappear on the reread. This initial reread of the failing sector is in addition to the retry count passed in the DCB (bit 7, byte 5).

The controller presets the error retry count to 4 each time a sector is read successfully. Sometimes, an error labeled uncorrectable is later found to be correctable. If this happens during a multiple-sector transfer, the controller resets the retry count to 4 before another sector is read.

3.6.18 Sector Field Description

Table 3-25 describes the sector information fields.

Table 3-25 Sector Field Format

Field	Number of Bytes	Field Description
AM	4	Address mark
GAP1	9	Zero byte gap
SYNC	1	ID sync byte
GAP2	2	ID zero byte gap
COM	1	ID compare byte
CYLH	1	Cylinder high (MSB)
CYLL	1	Cylinder low (LSB)
HEAD	1	Head number
SEC	1	Sector number
FLAG	1	Flag byte
ZER	1	Zero byte
ECC	4	ID ECC bytes
GAP3	16	Zero byte gap
SYNC2	1	Data field sync byte
GAP4	2	Data field zero byte gap
DATA	512	Data field
ECC2	4	Data field ECC bytes
GAP5	43	Inter-record zero gap

Notes:
1. Cylinder (track) numbering is 0-based.
2. Sector numbering is 1-based.
3. Disk surface numbering is 0-based

The track layout for the 512 bytes/sector, 17 sectors/track is given in Table 3-26.

Table 3-26 512-Bytes-Per-Sector Format

BYTE	7 (MSB)	6	5	4	3	2	1	0 (LSB)
1-4	ADDRESS MARK							
5-13	0	0	0	0	0	0	0	0
14	ID SYNC BYTE							
15-16	0	0	0	0	0	0	0	0
17	ID COMPARE BYTE							
18	CYLINDER NUMBER (MSB)							
19	CYLINDER NUMBER (LSB)							
20	HEAD NUMBER							
21	SECTOR NUMBER							
22	FLAG BYTE							
23	0	0	0	0	0	0	0	0
24-27	ID ERROR CORRECTION CODE BYTES							
28-43	0	0	0	0	0	0	0	0
44	DATA FIELD SYNC BYTE							
45-46	0	0	0	0	0	0	0	0
47-558	512 BYTES DATA							
559-562	DATA FIELD ERROR CORRECTION CODE BYTES							
563-605	0	0	0	0	0	0	0	0

605 bytes/sector including ID and overhead
Track Capacity = 10416

```
10285  =  17 sectors of 605 bytes/sector
 +131  =  Speed tolerance gap
------
10416
```

3.6.19 Specifications - Controller Board

Table 3-27 gives the Winchester controller board specifications.

Table 3-27 Winchester Controller Board Specifications

Environmental Parameters:

	Operating	Storage
Temperature	10 °C to 40 °C (32 °F to 131 °F)	-10 °C to 60 °C (-40 °F to 167 °F)
Relative Humidity (@ 40 °F wet-bulb temperature, no condensation)	10% to 90%	10% to 90%
Altitude	Mean sea level to 10 000 ft	Mean sea level to 45 000 ft

Power Requirements:

Voltage	Range	Current
+5.0 Vdc	4.75 to 5.25 Vdc	2.5 A maximum 2.0 A typical
-12.0 Vdc	-10.8 to -13.2 Vdc	66.0 mA maximum 48.0 mA typical

3.6.20 Electrical Interface

This paragraph specifies the electrical interface requirements for the 5 1/4-in Winchester disk drive.

All Winchester controller boards use header assemblies interchangeable with the AMP type 87215-7 for the 20-pin connectors (to J2/P2), and type 1-87215-7 for the 34-pin connector (to J1/P1). Section 5 contains assembly drawings showing the pin-outs for these connectors. The connector layout is shown in Figure 3-9.

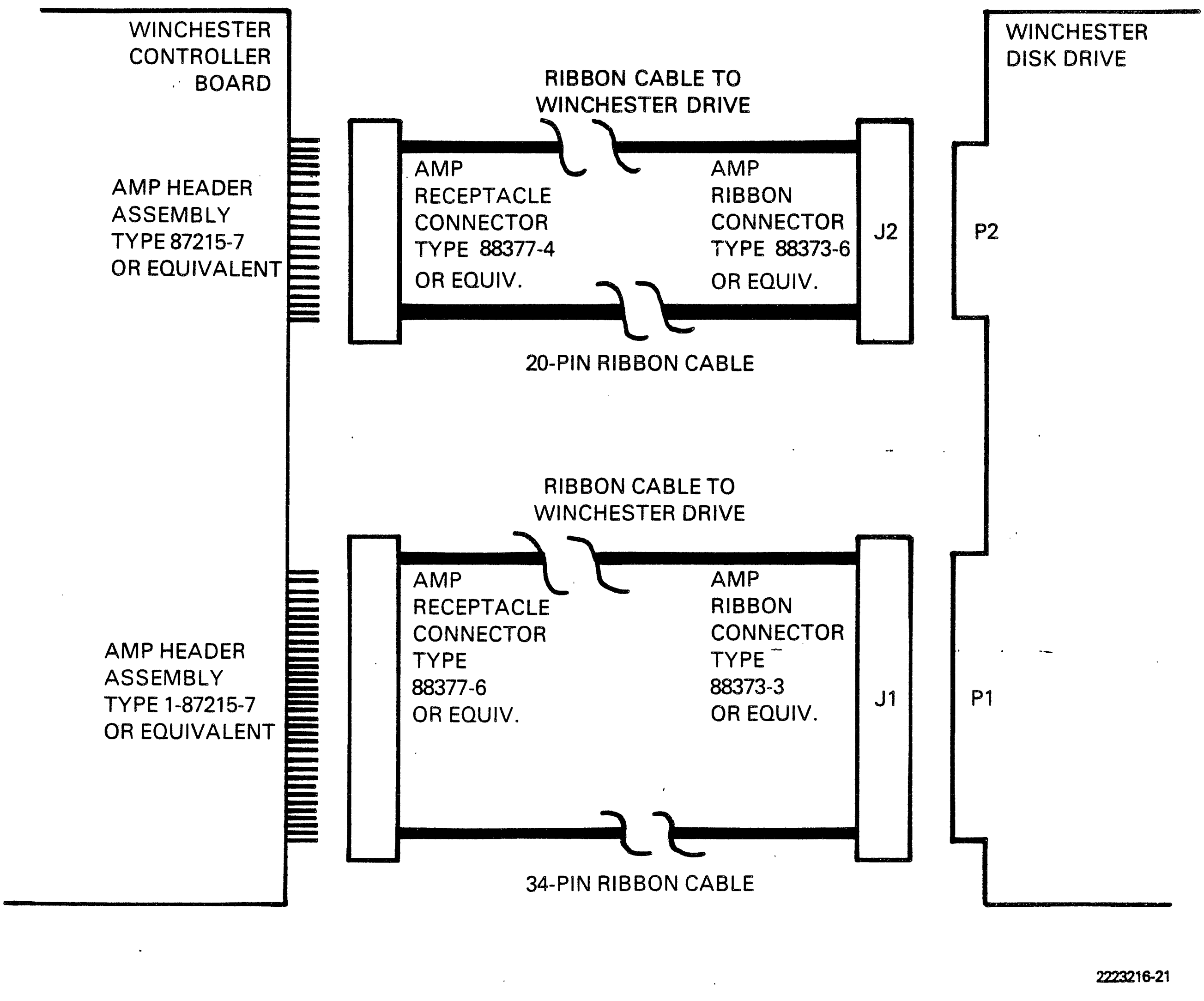


Figure 3-9 Control and Data Cabling for the Winchester Disk Drive

Section 4

DEVICE SERVICE ROUTINES

4.1 ROM INTERFACE INFORMATION

This section provides information on writing software for compatibility with future products and on interfacing with the hardware of the Texas Instruments Professional Computer. The interface information includes interrupt vectors, system memory maps, and ROM usage. The system ROM contains instructions for hardware device control of the standard I/O devices in the system unit.

The functions described are implemented with code in the system ROM, and thus are available to all users of the system regardless of which disk operating system (DOS) is installed. However, the user must be careful to avoid causing any conflicts with the operating system's use of these same functions.

Typically, these functions are accessed through the 8088 software interrupt mechanism. Each major device service routine (DSR), such as keyboard, display, and disk, has a unique vector. Individual functions of a DSR are accessed by placing an opcode in register AH and executing an INT (interrupt) instruction of the applicable type. To replace all or part of a DSR, just patch the interrupt vector to point to the user-written code.

For specific information on the architecture of the Intel 8088 microprocessor, read the IAPX 88 Book or the IAPX 86,88 User's Manual.

4.2 WRITING SOFTWARE FOR COMPATIBILITY WITH FUTURE PRODUCTS

The software you develop for this product undoubtedly represents a large investment of your time and money. Making changes and releasing new versions of software is usually difficult and expensive, and should be avoided. This guide will help you to create software that can be used with future Texas Instruments products.

4.2.1 Compatibility Levels

In order for the software to work on more than one hardware product, compatibility must exist at some level: either the operating system level, the system ROM interface level, or the hardware interface level.

4.2.1.1 Operating System. Software that interfaces at the operating system level is compatible with all products using the same operating system, including products of other manufacturers.

4.2.1.2 System ROM Interface. Software that interfaces with the Texas Instruments-supplied system ROMs through the interface vectors is compatible with other hardware products having the same functional characteristics. These products can differ in physical or electrical characteristics from the standard Texas Instruments product. Programs compatible at this level or at the DOS level are more likely to be compatible with future products.

4.2.1.3 Hardware Interface. Programs that use the hardware directly (for example, input or output to hardware addresses) are least likely to be usable in another computer system.

4.2.2 Areas of Hardware Compatibility

Texas Instruments recognizes that the system ROM interface is not sufficient for all applications. Products using the advanced capabilities of the hardware cannot be restricted to usage of this interface. The following paragraphs describe the hardware compatibility that can be expected in future subsystems or subsystems accessed from ROM only.

4.2.2.1 Alphanumeric CRT. The alphanumeric CRT is well-supported by the system ROM. Accessing the screen directly can speed processing, lets you use "windowing", and lets you use horizontal scrolling. You should restrict direct access to the alphanumeric CRT screen to the attribute latch and to address 0DE000H, the actual memory buffer for the screen. (The "H" represents hexadecimal.) Before using the screen directly, these programs should issue a Clear Screen function call to ensure that the hardware is set up for direct access. Refer to paragraph 2.4.7 for information about the CRT hardware.

Using the ROM functions to put data on the screen while accessing the screen directly can cause undesirable hardware actions. It is possible, for instance, that the screen can be hardware-scrolled, so that the logical upper left position is no longer the physical upper left position. All operations on the cursor should use the ROM interface calls. This will ensure that possible redesigning of the cursor logic does not prevent the program from running.

4.2.2.2 Graphics CRT. The graphics screen is not supported by the system ROM; therefore, all graphics screen functions must go directly to the hardware. The graphics screen size is 720 by 300.

To simplify modification, all routines that access the graphics hardware should be arranged in a modular fashion. Hardware-specific constants should be given symbolic names. Refer to subsection 3.5 for more information.

Texas Instruments will endeavor to keep future graphics hardware fully compatible with the current hardware.

4.2.2.3 Disk Subsystem. The disk subsystem is fully supported in the system ROM, with the exception of the ability to format diskettes. For normal operations, direct access to any of the disk hardware should not be necessary. Upon request, Texas Instruments will supply a format routine to qualified software vendors.

4.2.2.4 Keyboard System. The keyboard system is fully supported in the system ROM. Direct access to the keyboard interface is not necessary for normal operations.

4.2.2.5 Interrupt Controller. The interrupt controller system is used by the system ROM, but it is not supported in a fashion usable by software writers. In future products, Texas Instruments will attempt to keep the same interrupt levels, usage, and hardware addresses for accessing the device. However, the constants used to access this hardware should be symbolic to facilitate modification.

4.2.2.6 System Timers and Speaker. The system ROMs contain vectors that allow other software to intercept the 25-ms system timer interrupts. The extra timer is reserved for use by Texas Instruments software products.

The speaker (or bell) is well-supported by the system ROM. Direct access is not necessary.

4.2.2.7 Parallel Printer Port. The parallel printer port system is fully supported in the system ROM. Direct access is not necessary for normal operation.

4.2.2.8 Serial Communications. The serial communications hardware is not directly supported by the system ROM. To ensure future compatibility, Texas Instruments does not intend to change this hardware.

4.3 SYSTEM ROM INTERRUPT VECTOR USAGE

The system ROM uses interrupt vector locations in the first 1K bytes of memory. These vector locations are used for hardware interrupts, as interfaces to the ROM functions, and other uses as given in Table 4-1. The vectors marked with an asterisk are actually used by the ROM. The other vector locations cause a "wild" interrupt if vectored to, and the usual display is:

"** SYSTEM ERROR ** - 1042"

To patch in replacement routines for those in the ROM, any of these vectors can be changed by the disk operating system (DOS) or by applications software. Table 4-1 gives vector usage in terms of "interrupt type," which is the number used in an INT instruction. To calculate the absolute address of the vector, multiply the interrupt type by four. For example, the keyboard print screen interrupt vector (type 5EH) would be a double word at location 0:0178H (5E x 4 = 178H).

NOTE

The symbol "H" denotes a hexadecimal value.

Table 4-1 System Interrupt Vector Usage

Vector	Description	Reference
00	Divide-by-zero trap	IAPX 88 Book !
01	Single-step trap	IAPX 88 Book !
02*	Non-maskable interrupt	IAPX 88 Book !
03	Break (single-byte) software interrupt	IAPX 88 Book !
04	Overflow trap	IAPX 88 Book !
05-1F	(Reserved by Intel)	IAPX 88 Book !
20-3F	(Reserved for MS-DOS)	MS-DOS Operating System @
40	8259 interrupt 0	Component Data Catalog !
41	8259 interrupt 1	Component Data Catalog !
42	8259 interrupt 2	Component Data Catalog !
43*	8259 interrupt 3 (Timer 1)	Component Data Catalog !
44	8259 interrupt 4	Component Data Catalog !
45	8259 interrupt 5	Component Data Catalog !
46*	8259 interrupt 6 (Disk controller)	Component Data Catalog !
47*	8259 interrupt 7 (Keyboard UART)	Component Data Catalog !
48*	Speaker DSR interface	Section 3 !!
49*	CRT DSR interface	Section 3 !!
4A*	Keyboard DSR interface	Section 3 !!
4B*	Parallel port DSR interface	Section 3 !!
4C	(Reserved for future use)	**
4D*	Disk DSR interface	Section 3 !!
4E*	Time-of-day clock DSR interface	Section 3 !!
4F*	System configuration call	Section 3 !!
50*	Fatal software error trap	**
51*	Restart timing event	**
52*	Cancel timing event	**
53*	SVC interface subroutine	**
54*	Activate task subroutine	**
55-56	(Reserved for future use)	**
57*	CRT mapping vector	Section 3 !!
58*	System timing, 25 ms (time slicing)	Section 3 !!
59*	Common interrupt exit vector (ROM)	Section 3 !!
5A*	System timing, 100 ms (timing serv.)	Section 3 !!
5B*	Keyboard mapping vector	Section 3 !!
5C*	Keyboard program pause key vector	Section 3 !!
5D*	Keyboard program break key vector	Section 3 !!
5E*	Keyboard print screen vector	Section 3 !!
5F*	Keyboard queueing vector	Section 3 !!

Notes:
* Vector actually used by ROM.
** Texas Instruments use only - not to be changed.
@ Texas Instruments Incorporated publication
! Intel Incorporated publication
!! This manual

Table 4-1 System Interrupt Vector Usage (Concluded)

Vector	Description			Reference
60*	System ROM	DS pointer	(180H)	Section 3 !!
	(F400:A000)	DS size in bytes	(182H)	Section 3 !!
61*	Factory ROM	DS pointer	(184H)	Section 3 !!
	(F400:0000)	DS size in bytes	(186H)	Section 3 !!
62*	Option ROM	DS pointer	(188H)	Section 3 !!
	(F400:2000)	DS size in bytes	(18AH)	Section 3 !!
63*	Option ROM	DS pointer	(18CH)	Section 3 !!
	(F400:4000)	DS size in bytes	(18EH)	Section 3 !!
64*	Option ROM	DS pointer	(190H)	Section 3 !!
	(F400:6000)	DS size in bytes	(192H)	Section 3 !!
65*	Option ROM	DS pointer	(194H)	Section 3 !!
	(F400:8000)	DS size in bytes	(196H)	Section 3 !!
66*	Memory size in paragraphs		(198H)(word)	Section 3 !!
	Outstanding interrupt count		(19AH)(byte)	Section 3 !!
	Installed drive types		(19BH)(byte)	Section 3 !!
67*	Extra system configuration (word 1)		(19CH)	Section 3 !!
	Extra system configuration (word 2)		(19EH)	Section 3 !!
68-9F	Reserved for Texas Instruments			
A0-DF	User interrupt vectors			
E0-E3	Reserved for CP/M [tm]			CP/M 86 Programmer's Guide @
E4-FF	Reserved for Texas Instruments			

Notes:
* Vector actually used by ROM.
** Texas Instruments use only - not to be changed.
@ Texas Instruments Incorporated publication
! Intel Incoporated publication
!! This manual

4.3.1 Hardware Interrupt Service Routines

All standard interrupt service routines (ISR) have limited internal stacks. They provide four levels (8 bytes), which is the amount required by any application program or subroutine that runs with interrupts enabled. An ISR needs 8 bytes of the user's stack; 2 bytes to push the user's code segment (CS), 2 bytes for the instruction pointer (IP), 2 bytes for flags, and 2 bytes to push the data segment (DS). The ISR saves the user's stack segment and stack pointer in the RAM data area of the system ROM. The ISR then changes the stack segment and stack pointer so that they point to the internal stack of the interrupt routine. When the ISR is complete, it executes a long jump to the common interrupt exit vector.

4.3.2 Common Interrupt Exit Vector

All ISRs (in the ROM and in Texas Instruments applications programs) use a common interrupt exit vector. The ISR executes a long jump (LONG JMP) to the routine pointed out by the common interrupt exit vector. The common interrupt exit routine restores the stack and commonly used registers, decrements the outstanding interrupt counter (INTCTR), sends the end-of-interrupt (EOI) command to the interrupt controller, and returns to the interrupted code with a return-from-interrupt instruction (IRET).

A real-time operating system (OS), such as the OS kernel of TI communication programs, uses the INTCTR to keep track of the outstanding interrupts. Be sure to include the appropriate code when creating an ISR.

A sample interrupt service routine, with installation and removal instructions, is included in Appendix G.

The common interrupt exit routine is contained in ROM, but an OS can patch it so that all interrupt service routines exit through the operating system. Because the interrupt structure is complex (due to interaction between the shared interrupts and the requirement for a common exit point), the potential user should read the following paragraphs, carefully studying the examples given.

4.3.3 Timer Interrupts

The system timer ticks every 25 ms. The ISR for this timer is located in the ROM, and it processes events such as disk motor time-outs and date/time-keeping. Software interrupts are performed at two points during this interrupt service routine, allowing access to the timing services. One interrupt occurs every count (every 25 ms), and the other occurs every four counts (100-ms intervals). Usually, these interrupt vectors point to an IRET instruction in the ROM. The user can patch one or both of the vectors to point to his own routines. These routines are free to use the AX, BX, DI, and ES registers, but they must preserve any other registers used. The stack used is the internal stack of the timer interrupt service routine and it is limited in depth. If the user does not re-enable interrupts (the INT instruction disabled them), there are 8 levels (16 bytes) of stack available. If the interrupts are re-enabled, the user has only four levels (8 bytes) available. If more stack size is required, the user should switch to an internal stack of the required size (allotting 8 bytes for higher priority interrupts).

It is important to remember that the routines installed in this manner are executing at the interrupt level. Interrupts must not be disabled for any significant length of time, because any time spent in these routines directly affects system efficiency. The user must also understand how some other mechanism (such as a timing event in the handler routine of the OS) can patch the timing vectors and install its own routines. Instead of using the IRET instruction to end the routine, make a long jump to the original vector address (which was saved when the routine was installed.)

4.4 ROM STRUCTURE

The following paragraphs describe the use, format, and calling sequences for optional ROMs.

4.4.1 ROM Usage

Optional ROMs provide an interface between the hardware and the system software. With this interface installed, modification of the hardware requires changing only the ROM software, not all of the applications programs.

The system defines locations for six ROMs. One of these is the system ROM. Texas Instruments has reserved another (on the main board) for future use. The four remaining are the optional ROMs, which can be used by any of the available operating systems.

Table 4-2 shows the ROM addresses and suggestions for their use.

Table 4-2 ROM Addresses and Suggested Uses

Absolute Address	CS:Offset	Use	Comments
F4000H	F400:0000H	Miscellaneous I/O option	Reserved for Texas Instruments
F6000H	F400:2000H	Local area network	Reserved for Texas Instruments
F8000H	F400:4000H	Mass storage	Texas Instruments Winchester card
FA000H	F400:6000H	Open	Open
FC000H	F400:8000H	System ROM expansion	Reserved for Texas Instruments
FE000H	F400:A000H	System ROM	Reserved for Texas Instruments

4.4.2 ROM Format

The ROM format must be known to:

* Identify the ROM
* Use a standard calling sequence
* Use the diagnostics

ROMs can be one of the following sizes:

* 256 bytes
* 512 bytes
* 1024 bytes
* 2048 bytes
* 4096 bytes
* 8192 bytes

The ROM size, in binary, is stored in the first word in the ROM. The word value is stored low byte first, following the INTEL Corporation convention.

The second word in the option ROM is the power-up initialization address. The system ROM uses a NEAR call to this address during the power-up process. The user must ensure that the initialization address is calculated as an offset from the segment address F400.

The next location in the ROM stores a text string identifying the ROM. The first entry in this string is the length of the string (1 byte). This information determines how much material is displayed.

The rest of the string consists of a five-character version number, a space character, a six-character name, and any descriptive text (copyright, for example) that the vendor requires.

The option ROM code and fixed data (in a format determined by the vendor) follows the text string.

The last word in the ROM stores the cyclic redundancy check (CRC-16) remainder from all the previous bytes in the ROM. Both the power-up test and the advanced diagnostics test read this word to see if the ROM is working properly. The CRC-16 routine, available in the system ROM, calculates this remainder. When the CRC remainder is correctly placed, running the CRC-16 routine through the entire length of the ROM (including the CRC) results in a zero remainder. The CRC-16 routine available in the system ROM calculates the remainder.

4.4.3 Option ROM Interrupt Vector Usage

The system ROM uses interrupt vector locations in the first 1K bytes of RAM for hardware interrupts, interface to the ROM functions, and other ISRs. See paragraph 4.3.1 for more information.

Interrupt vectors access the option ROM entry points. The option software can use the vectors above 80H (vector address 200H).

NOTE

Conflicting vector assignments can cause data loss or data errors. Be extremely careful when making these assignments.

4.4.4 RAM Usage by Option ROM

Each ROM has a separate RAM data area assigned to it. These data areas float; therefore, the ROM does not require a dedicated area in RAM. Copying the data area and updating the pointer moves the data area. The ROM accesses these data areas using the pointers and sizes in the interrupt vector area, so that moving the data area does not affect the ROM. The ROM initializes the pointers and data areas at boot-up time, so the system ROM data area pointer is the only one used.

All option ROMs are addressed at absolute segment addresses F400H, with an offset from 0000 to A000H. The ROM code is linked so that its code segment is F400H. This code segment was chosen so that option ROMs can be addressed with the same code segment as the system ROM. This enables the option ROM to access the ROM powerup entry routines as NEAR instead of FAR. The first location of the system ROM, described in segment:offset notation, is F400:A000.

There is another advantage to linking the ROMs this way. The interrupt vector area at location 0000:0000 is now also accessable as F400:C000. This simplifies slightly the code sequence used to assign a local data area.

4.4.5 Initializing the Option ROM

The power-up sequence executed by the main ROM tests each option ROM address in sequence. Address 0F400:0000H is tested first and address 0F400:8000H (the main board option ROM) is tested last. When a ROM is found, the diagnostics performs a CRC-16 calculation. The system displays an error message if the ROM is bad. If the ROM is good, the system initializes the option ROM. The initialization code saves the BX, DX. SI, SP, CS, SS, and DS registers so that using a NEAR return instruction returns control to the system ROM.

4.5 BOOTING UP THE SYSTEM

Most system software is contained in some mass-storage system (diskette, Winchester disk, or local network server). The user must be able to find and load the system software from these devices. The Texas Instruments Professional Computer loads a single sector of program information from a known point on the specified device. The system then calls the code that was loaded, which "bootstraps" the rest of the programs.

The location loaded at power-up is the lowest logical sector available. For diskettes and Winchester disks, this location is cylinder (track) 0, surface (side) 0, and sector 1. (Sector numbers start at 1.)

4.5.1 Boot Sequence

The options installed in the system determine the boot sequence. The sequence starts at the highest-priority option address (0F400:0000H), proceeds to the lowest (0F400:8000H), then boots the diskette system. The boot sequence is:

1. Local Area Network (LAN)
2. Winchester disk subsystem
3. Diskette drive A
4. Diskette drive B
5. Diskette drive C
6. Diskette drive D

Pressing the ESC key during the power-up sequence (immediately after the "white flash" appears across the top of the screen) changes the boot priority. Each time the ESC key is pressed, the system lowers the boot sequence to the next available option. For example, if the system contains either an LAN or a Winchester disk, pressing the ESC key once lowers the boot sequence to the first diskette. If the system contains both an LAN and a Winchester, pressing the ESC key once moves from the LAN down to the Winchester, while pressing the ESC key twice moves to the first diskette.

4.5.2 Loading and Calling the Boot Code

The booting device loads the boot code at address 0000:C000H. The stack operates below this address. After the code is loaded, the system checks address 0000:C1FCH for the bytes 74H and 69H (ti). The presence of these bytes indicates a Texas Instruments system disk. If these bytes are absent, the system generates an error message. (Texas Instruments disks used only for data storage contain the characters "NO".) The system then runs the CRC-16 test over all 512 bytes of the lowest logical sector loaded at power-up. If the CRC-16 remainder is incorrect, the system generates an error message. If the system passes both these tests, it calls the boot sector code at address 0000:C000H (FAR). The logical drive number (0, 1, 2, 3) from which the system boots is placed in register BL.

Before loading the operating system, the boot code performs other required initializations such as setting up the type of floppy disk (single or double sided, 40 or 80 track), or setting up the type of Winchester drive. (The DSR must be able to recognize the disk format for further loading.)

The boot code then loads any system files needed by the OS and jumps to the OS code. If the OS requires RAM where the system ROMs are using it, the RAM data areas used by the ROM can be moved. The pointers to the RAM segments must be modified accordingly. If a ROM is not using a RAM data area, its pointer is 0000. This pointer must remain zero even if the area is moved. Table 4-3 gives the addresses of these pointers.

Table 4-3 Pointer addresses and Descriptions

Address	Pointer Description	ROM Address
0000:0180	System ROM data segment pointer	F400:A000
0000:0182	System ROM data length in bytes	
0000:0184	Option ROM data segment pointer	F400:0000
0000:0186	Option ROM data length in bytes	
0000:0188	Option ROM data segment pointer	F400:2000
0000:018A	Option ROM data length in bytes	
0000:018C	Option ROM data segment pointer	F400:4000
0000:018E	Option ROM data length in bytes	
0000:0190	Option ROM data segment pointer	F400:6000
0000:0192	Option ROM data length in bytes	
0000:0194	Option ROM data segment pointer	F400:8000
0000:0196	Option ROM data length in bytes	

If any errors occur during the loading and initializing of the OS, the boot code returns to the caller. The registers BX, ES, CS, and the stack must be preserved. The register DS must be preserved unless the ROM data areas are moved. If the data areas are moved, adjust the DS register by the amount of difference between the original position and the new position. A DSR error code returns to the caller displayed as a system error message. This code is presented in register AH.

Appendix H gives a sample source program that could be used in the boot sector.

4.5.3 Booting From an Option Device

When an option device is to be booted up, it must be the last one called in the power-up sequence. Otherwise, other options must be called and initialized during the boot sequence. Appendix G contains a sample assembly code showing the boot sequence.

If more than one bootable option is present in the system, each one must have the DX register set to OFFFFH. The bootable option then calls all lower priority ROMs in the system. Any ROM called in this manner performs all required initialization except for booting. Because the system ROM sets the DX register to 0000H when it calls the option ROMs, an option device will boot if called by the system ROM, but not if called by another ROM.

If booting from an option device fails, the ROM displays the appropriate error messages and returns to the caller with registers BX, DX, SI, and DS intact. The system ROM then calls the other options. If none of the options boot, the system ROM boots the Floppy Disk system.

This procedure can cause multiple initializations of the options. However, no harm results. Entering the warm boot sequence (CTRL/ALT/DEL) from the keyboard also causes multiple initializations.

4.6 SYSTEM CONFIGURATION FUNCTION CALLS

The following paragraphs describe the function calls for the two types of system configuration information, which are:

* Function calls that return the information in a register (System Configuration Function)

* Function calls that return the address of the information (Extra System Configuration Function)

The first type, System Configuration Function, returns most of the information required for application programs. Extra System Configuration Function, the second type, is intended for use at the system level. This method contains additional information usable for changing the configuration of devices set by software.

4.6.1 System Configuration Function

This function is used to determine the installation status of certain system options. It is invoked by executing an INT 4FH instruction.

Upon return, register BX contains the size of contiguous RAM (starting at 00000H) in paragraphs (16-byte blocks). A 128K-byte system, for example, would return 2000H in BX.

Register AX contains the system configuration word, which reflects the installation status of various system options. The bits of the word are defined in Table 4-4.

Table 4-4 System Configuration Word-Bit Definition

Bit	Definition
0*	Diskette drive 0 (internal) installed
1	Diskette drive 1 (internal) installed
2	Diskette drive 2 (external) installed
3	Diskette drive 3 (external) installed
4	E1-E2 jumper (0 indicates Drive A is double-sided)
5	E3-E4 jumper (0 indicates Drive A has 80 tracks)
6	E5-E6 jumper (0 indicates a 50-Hz system)
7	Winchester disk controller installed
8	Serial port 1 installed
9	Serial port 2 installed
10	Serial port 3 installed
11	Serial port 4 installed
12	Graphics RAM bank A installed
13	Graphics RAM bank B installed
14	Graphics RAM bank C installed
15	Reserved

* Bit 0 is the least-significant bit. Unless otherwise stated, a statement is true when its corresponding bit is a 1.

4.6.2 Extra System Configuration Function

This function determines the installation status of system options that are not covered in the standard system configuration call. Whereas the standard system configuration call returns a word containing the information necessary for most applications, the extra system configuration function is used primarily for systems programming purposes.

The extra system configuration function is invoked by placing a 0BH in register AH and executing an INTerrupt 48H. Upon return, register AL contains the drive-type byte (AH is undefined). BX contains extra system configuration word 1, and CX contains extra system configuration word 2. The bits of extra system configuration word 1 are defined in Table 4-5.

Table 4-5 Extra System Configuration Word 1 (BX)

Bit	Definition
0*	8087 numeric coprocessor is installed
1	\
2	
3	
4	> Reserved
5	
6	
7	/
8	300/1200 baud modem in port 1
9	300/1200 baud modem in port 2
10	300/1200 baud modem in port 3
11	300/1200 baud modem in port 4
12	300 baud modem in port 1
13	300 baud modem in port 2
14	300 baud modem in port 3
15	300 baud modem in port 4

* Bit 0 is the least-significant bit. Unless otherwise stated, a statement is true when its corresponding bit is a 1.

Word 2 of the Extra System Configuration function call is contained in CX. This word is currently undefined, and is being reserved for later expansion.

The drive-type byte defines the types of the installed diskette drives. This information, combined with the "installed drive" bits in the standard system configuration word, yields complete information about the drives in the system. At power-up, the drive A definition jumpers (E1 - E2 and E3 - E4) are read. The information is stored in memory as a byte of four identical, 2-bit fields. This byte is read during the extra configuration function call and returned in register AL. The drive byte (in AL) is the 2-bit configuration code for all four of the diskette drives, which is shown in Figure 4-1.

7	6	5	4	3	2	1	0
Drive D		Drive C		Drive B		Drive A	

Each 2-bit field is defined as:

MSB*	LSB		Definition	
0	0	=	Single-sided	40 track
0	1	=	Double-sided	40 track
1	0	=	Single-sided	80 track
1	1	=	Double-sided	80 track

*MSB = Most significant bit; LSB = Least significant bit.

2223216-22

Figure 4-1 Register AL Drive Byte

The operating system uses this drive byte to format, copy, and use diskette files. It is possible to mix drive types in one system (for example, one single-sided and one double-sided drive) by setting the drive-type byte with the pertinent information; but, this is <u>not</u> recommended. Mixed-drive type systems are confusing. Users frequently insert the wrong diskettes, thereby losing data.

4.6.3 Get Pointer to System Configuration

This function is invoked by placing a 09H in register AH and executing an interrupt 48H. On return, ES contains the segment, and BX contains the offset of the standard system configuration word (hereafter, the notation for this is ES:BX). This function is used by system software that has a need to change the configuration information. Although an application program can access the information in this manner, the configuration must not be changed.

4.6.4 Get Pointer to Extra System Configuration

This function is invoked by placing a 0AH in register AH and executing an INTerrupt 48H. On return, ES:BX points to the extra system configuration information, formatted as follows:

```
ES:[BX-3]=(word) Size of memory in 16-byte blocks
ES:[BX+0]=(byte) Drive-type byte
ES:[BX+1]=(word) Extra system configuration word 1
ES:[BX+3]=(word) Extra system configuration word 2
```

This function is used by system software that has a need to change the configuration information. Although an application program can access the information in this manner, the configuration must not be changed.

4.7 GENERAL-PURPOSE ROM FUNCTIONS

The following paragraphs describe some general-purpose functions, summarize the ROM interface interrupts, and explain how the RAM uses the ROM.

4.7.1 Delay

This function causes a delay, in milliseconds, of the value placed in register CX. To invoke the function, place the delay value in CX, 05H in AH, and execute an INT 48H. The delay is approximate, but can be used wherever an inexact software delay is acceptable. All registers except CX are preserved.

4.7.2 CRC Calculation

This function calculates the cyclic redundancy check (CRC-16) value for a specified block of memory. It is invoked by placing the address of the memory block in ES:BX, the size of the block in BP, and the value 06H in AH, then executing an INT 48H. On return, DX contains the CRC value; if DX=0000, the Z-flag is set. For memory blocks that follow the convention of the CRC being the last word in the block, this routine allows easy CRC checking. First, the CRC of the memory block is calculated, with the size of the block set to the actual size minus two. The CRC word is then written to the last word of the block. Subsequently, the CRC of this block can be checked by calling this function with the actual size of the memory block (including the previously calculated CRC). By definition, the CRC result of this block is zero (if the CRC matches the data) and the Z-flag is set; otherwise, the CRC fails and the Z-flag is reset. All registers are used except DI, SI, and DS. ES remains unchanged.

4.7.3 Print ROM Message

This function displays a ROM CS-relative message. It is invoked by placing the offset of the zero-terminated message in SI, 07H in AH, and executing an INT 48H. This function is used by the option ROMs, because all the ROMs share a common CS. It is not a general-purpose routine.

4.7.4 Display System Error Code

This function is used to display a system error in the standard format:

** System Error** - xxxx

It is invoked by placing the error code (the xxxx value in the displayed message above) in BX, placing the value 08H in AH, and executing an INT 48H.

4.8 SPEAKER DSR

The following paragraphs describe the speaker DSR and the functions it provides to the system or application programs that use it. The functions are:

* Sound the Speaker
* Get Speaker Status
* Set Speaker Frequency
* Speaker ON
* Speaker OFF

The speaker DSR functions are located in the system ROM and are accessed through the software interrupt mechanism of the 8088 microprocessor. The desired function is chosen by placing an opcode in register AH and executing an INT 48H instruction. All registers are preserved except AX.

4.8.1 Sound the Speaker - AH = 0

This function turns the speaker on (at the current frequency) for the length of time specified in register AL. Time is measured in 25-ms increments. For example, a value of 40 in AL causes the speaker to sound for 1 second. Timing is handled in the ROM with the result that the request turns on the speaker, starts the timer, and immediately returns to the user. The sound continues until timed out by the ROM code. Because this function call occurs asynchronously with the 25-ms system timer, the time can be "off" by as much as 25 ms. For example, specifying a single 25-ms unit of time can cause the speaker to sound for a period of 0 to 25 ms. If there is need to synchronize with the sound or simply to know when sound is turned off, use the Get Speaker Status (AH=1) function.

4.8.2 Get Speaker Status - AH = 1

This function returns the status of the speaker in the Z-flag. If the speaker is currently enabled (sound), the Z-flag is set at 0. If the speaker is currently disabled (no sound), the Z-flag is set at 1. This function can be used to find out when a sound requested with the Sound the Speaker (AH=0) function has been completed.

4.8.3 Set Speaker Frequency - AH = 2

This function sets the frequency of the speaker. Usually this function is called only when the speaker is disabled. The value in CX sets the frequency of the timer that drives the speaker. The input frequency of the timer is 1.25 MHz, and the value in CX becomes a divider for this frequency. For example, the system beep routine (800 Hz) uses a value of 1563 (1 250 000 Hz / 1563 = 800 Hz).

4.8.4 Speaker ON - AH = 3

This function enables the speaker (turns on the sound). The speaker remains on until it is turned off by either

(1) the Speaker OFF (AH=4) function or

(2) by the ROM timing routine, which results from either the Sound the Speaker (AH=0) function or a normal system beep.

4.8.5 Speaker OFF - AH = 4

This function performs the reverse of the Speaker ON (AH=3) function by disabling the speaker (turning off the sound).

4.9 TIME-OF-DAY CLOCK DSR

The following paragraphs describe the time-of-day clock DSR and the functions it provides to the system or application programs that use it. The functions are:

* Set the date
* Set the time
* Get the date and time

The clock DSR consists of routines to set and read the time of day and date information kept by the timing services of the system ROM. At power-up, the time is set to 00:00:00.00, and the date is set to 0000. These can be reset by system or user programs. Once set with a valid time, the clock keeps the correct time with a 1/10-s resolution. The time is kept in 24-hour format and the date is simply a cumulative count of days since the clock was started. As a matter of convenience (for MS-DOS), the date is specified as the number of days since January 1, 1980. For example, the date value for September 10, 1982, is 983.

The three clock functions are located in the system ROM and are accessed through the software interrupt mechanism of the 8088 microprocessor. The desired function is chosen by placing an opcode in register AH and executing an INT 4EH instruction. All registers are preserved except AX and any other registers in which information is returned.

4.9.1 Set the Date - AH = 0

This function sets the date to the value in the BX register. The date is simply a count of days since the clock was started. By convention, this is the number of days since 1-1-80. The count is incremented when the hour rolls over from 23 to 00.

4.9.2 Set the Time - AH = 1

To set the time, the registers must be initialized as follows:

CH = Hours (00 - 23)
CL = Minutes (00 - 59)
DH = Seconds (00 - 59)
DL = Hundredths of seconds (00 - 99)

It is the user's responsibility to make sure the values passed are within the ranges specified. These values are not checked for range and can be set to represent a meaningless time. The time, however, eventually counts into the normal sequence.

4.9.3 Get the Date and Time - AH = 2

This function returns the current date in register AX and the current time in registers CX/DX in the formats described previously.

4.10 CRT DSR

The following paragraphs describe the CRT DSR and the functions it provides to the system or application programs that use it. The major functions are (1) video mode control and (2) character handling.

For information about the CRT graphics hardware, refer to paragraph 2.4.7, and to subsection 3.5. The CRT DSR functions are located in the system ROM and are accessed through the use of the 8088 software interrupt mechanism (essentially an address-independent subroutine call). A typical user of this DSR is the OS-dependent system interface code (the BIOS), which resides on a particular OS disk and is loaded into RAM during disk boot up. The desired function is chosen by placing an opcode in register AH. The CRT opcodes and functions are given in Table 4-6. Various CRT functions require parameters to be passed in specific registers in addition to AH. After register AH and the parameter registers are set up, the user can execute an INT 49H and the specified function is performed. During this interrupt, all registers are preserved except AX, CX, and DX.

Table 4-6 CRT DSR Opcodes and Functions

Opcode	Function
00H	(Null function)
01H	Set cursor type
02H	Set cursor position
03H	Read cursor position
04H	(Null function)
05H	(Null function)
06H	Scroll text block
07H	Scroll text block
08H	Read character and attribute at current cursor position
09H	Write character and attribute at current cursor position
0AH	Write character only at current cursor position
0BH	(Null function)
0CH	(Null function)
0DH	(Null function)
0EH	Write ASCII teletype
0FH	(Null function)
10H	Write block of characters at current cursor with attribute
11H	Write block of characters only at current cursor
12H	Set entire screen to specified attribute(s)
13H	Clear text screen and home the cursor
14H	Clear graphics screen
15H	Set TTY status line beginning
16H	Set attribute latch to specified attribute(s)
17H	Read physical display begin pointer
18H	Print TTY string

4.10.1 Set Cursor Type - AH = 01H

This function allows an application to define the starting and ending scan line for the cursor and its characteristics (either blinking or no cursor). Required input for this function is described in Figure 4-2.

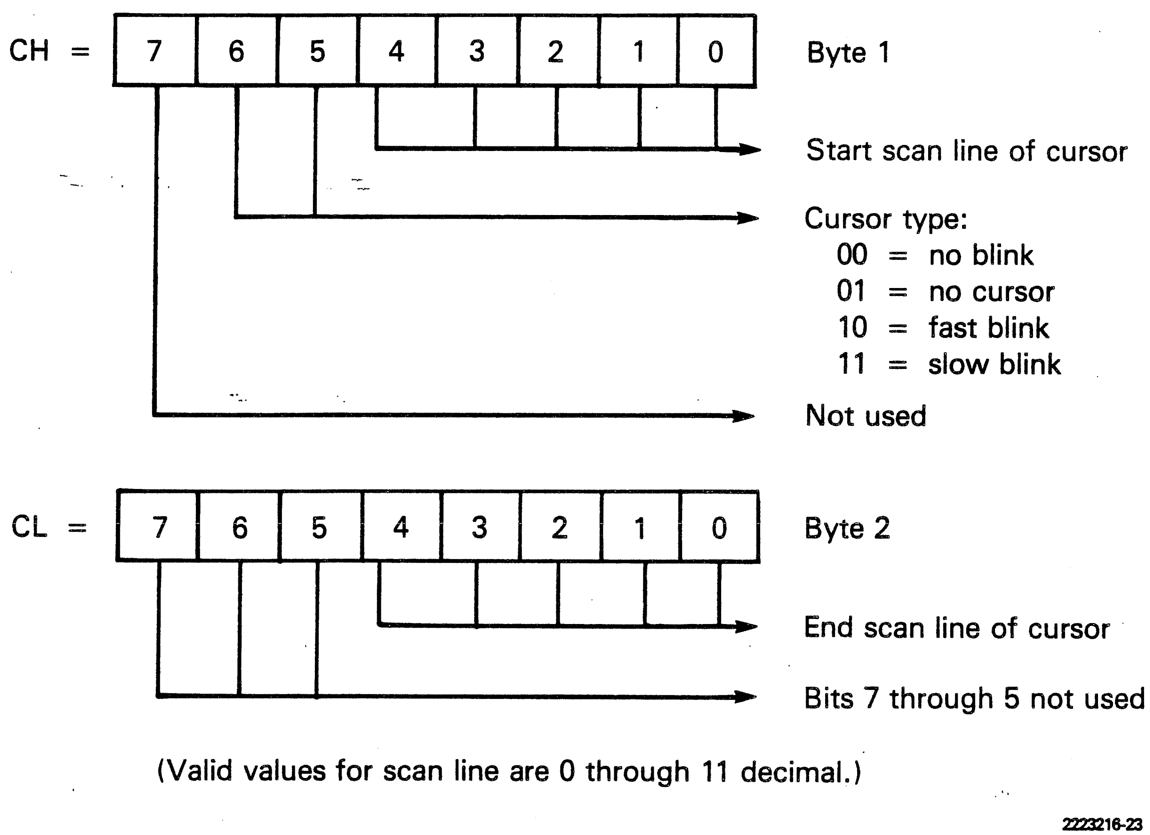


Figure 4-2 Byte Definition - Set Cursor Type

4.10.2 Set Cursor Position - AH = 02H

NOTE

> The user should be aware that screen coordinates use the 0,0 coordinate as the upper left-hand corner of the display. All routines that require a coordinate parameter use this convention. The screen should look to the user as though he were working with the absolute value of fourth-quadrant coordinates of a two-dimensional coordinate system.

This function causes the cursor (of the current type) to be set at the specified x,y (column/row) coordinate of the display.

Required input for this function is as follows:

DH = x Column coordinate
(Valid values are 0 through 79 decimal.)

DL = y Row coordinate
(Valid values are 0 through 24 decimal.)

4.10.3 Read Cursor Position - AH = 03H

This function returns the current position and type of the cursor. Output from the read cursor position routine is as follows:

DH, DL = x, y (column/row) location of the cursor

CH, CL = current cursor type

Refer to paragraph 4.10.1 for an explanation of the values for CH and CL.

The "phantom" position of the cursor in column 81 creates a special situation in reading the cursor postion. If a character is written in the last column of the screen by a TTY write, it can be read, even though it is not visible. This position, column 81 of the last line, becomes visible after another character is written and the screen scrolls. The position returns as column 0, row 25. This is invalid input to the Set Cursor Position (AH=02H) routine.

See paragraph 4.10.18 for additional information on the cursor.

4.10.4 Scroll Text Block - AH = 06H and 07H

The ROM contains only one general-purpose scroll routine, which handles both upward and downward scrolling. When the destination coordinates are less than the source coordinates, the scroll is up and to the left; when the destination coordinates are greater than the source coordinates, the scroll is down and to the right.

The scrolling functions allow an application program to specify a block of text, then move or copy that block to another location on the screen. Specifying a scroll with blanking causes the source text to be blanked as it is moved. During this process, the source character is read to a temporary register and its location is blanked. Then the character is rewritten to its destination location. This provides for a nondestructive move in the event that the source and destination locations are the same and blanking is specified. This method satisfies the requirement that, in scrolling, the data being moved or copied be preserved in its destination location.

Required input for this function is as follows:

```
          AL =  0 (Blank out source text.  This is  a move block.)
                                  or
          AL = >0 (Don't blank source text.  This is a copy block.)

     (DH,DL) = Source begin column/row location

     (BH,BL) = Destination begin column/row location

          CH = Column length of block
                (Valid values are 1 through 80 decimal.)

          CL = Line length of block
                (Valid values are 1 through 25 decimal.)
```

The source text block boundaries in (x,y) coordinates are as follows:

```
Upper left  = (DH,DL)
Upper right = (DH + CH , DL)
Lower left  = (DH , DL + CL)
Lower right = (DH + CH , DL + CL)
```

The following items further describe the scrolling routines and explain the sequence of operation.

* A sentence is considered the smallest logical block of text. Therefore, with this scrolling capability, the user can specify a block to be a sentence. This may (or may not) wrap to a new line and "unwrap" as it is moved (or copied) to its destination (that is, the column length parameter would bypass line boundaries and pick up characters from the next line). The user should note that this is quite effective when the line length is equal to one but might cause unwanted block movement if the line length is greater than one.

* Boundary checking for the scrolling routine is done on a character basis as the characters are being moved. When a scroll down is in progress, the scroll copies the last character in the source block to the last character position in the destination block. The processing is backward through the blocks while checking character positions for out-of-bound characters. This means that in the scroll-down action, no scroll takes place if any destination position lies beyond the end of the screen. Asymmetrically, when a scroll up is in progress, the scroll copies the first character in the source block to the first character position in the destination block. The scroll proceeds forward, through the blocks, while checking character positions for out-of-bound characters. In the scroll-up action, the scroll takes place until it reaches a source character position that lies beyond the end of the screen.

* When the user requests scrolling with blanking, the status of the attribute latch at entry is preserved. The character attributes follow the character as it is moved on the screen, and the blanked area is written with the default attributes (that is, high intensity for a monochrome monitor, and white for a color monitor).

* When the user requests scrolling without blanking, the attribute latch is set to the same status as the attribute of the last character that was scrolled (that is, the attribute of the first character of the source block when scrolling down, or the attribute of the last character of the source block when scrolling up).

4.10.5 Read Character/Attribute at Cursor Position - AH = 08H

This function returns a character and its associated attribute from the current cursor position on the screen as follows. See paragraph 4.10.15 for attribute values and a description of the attributes supported.

AH = Attribute value

AL = Character read

NOTE

The attribute latch remains set to the attribute that is returned.

4.10.6 Write Character/Attribute at Cursor Position - AH = 09H

This function enables the writing of a character with the given attribute at the current cursor position. (The attribute latch remains set to the attribute specified in register BL.) The user can specify a count and cause the character to be written a given number of times starting at the cursor's current position. This function does not increment the cursor automatically, and the cursor remains at its current position while the characters are written in succession from that location. If an application uses this method of writing characters, it is assumed that the application also handles the cursor positioning. Therefore, no cursor movement is implemented. Control characters (CR,LF, and so on) are not executed as such when using this function; their symbols are printed on the display. For more information, refer to paragraph 4.10.15.

The required input for this function is as follows:

AL = Character to write

BL = Attribute of character(s)

CX = Number of times to write the character

4.10.7 Write Character at Cursor Position - AH = 0AH

This function is similar to the preceding function. The difference is that the character being written takes on the attributes remaining in the attribute latch from the last CRT call. For more information, refer to paragraph 4.10.6.

The required input for this function is as follows:

AL = Character to write

CX = Number of times to write the character

4.10.8 Write ASCII Teletype - AH = 0EH

This function allows TTY output to the screen from application programs. Writing begins at the current cursor position, and the cursor is advanced automatically to its next position on the screen. For more information, refer to paragraph 4.10.18. The screen is scrolled automatically when needed (such as writing past the end of the screen). The control characters CR, LF, BS, and BEL are executed rather than written.

NOTE

> If a status region is currently in use, the scroll starts one line before the beginning of the status region, exactly as if that line were the end of the screen.

Because the contents of the attribute latch remain unchanged, each character written with this function assumes the attributes of the previously written character.

The required input for this function is as follows:

AL = Character to write

4.10.9 Write Block of Characters at Cursor With Attribute - AH=10H

This function writes a given block of data with a specified attribute to the screen, starting at the current cursor position. This function requires less screen I/O overhead if an application program has a "known" block of data to be written to the screen. "Known" means that the block is of a given length, and is in a given contiguous area of memory. As with the Write/Character Attribute at Cursor Position function, the cursor is not automatically incremented. For more information, see paragraph 4.10.15.

The required input for this function is as follows:

AL = Attribute(s) of characters *

DX = Segment location of character block

BX = Offset location of character block

CX = Block length **

4.10.10 Write Block of Characters Only at Cursor Position - AH=11H

This function is similar to the preceding function, with the difference that the attribute parameter is not specified. The characters assume the attribute(s) remaining in the attribute latch from the last CRT call.

The required input for this function is as follows:

AL = Don't care

DX = Segment location of character block

BX = Offset location of character block

CX = Block length **

* The attribute(s) specified is in effect for the entire block and the attribute latch remains set to the attribute specified in register AL.

** This routine "clips" any characters that do not fit on the screen. Characters are written to the end of the screen, then all other characters are lost/not written. To prevent losing characters, the user should place the cursor so that the number of character positions from the cursor to the end of the screen is greater than or equal to the block length.

4.10.11 Change Screen Attribute(s) - AH = 12H

This function specifies attribute(s) that affect all of the characters on the display. The attribute latch is set to the attribute specified in register AL on exit. This routine does not change the position of any characters on the screen. Two examples are blinking of the entire screen and reverse video of the entire screen. For more information, see paragraph 4.10.15.

The required input for this function is as follows:

AL = Attribute(s) to use

4.10.12 Clear Text Screen and Home the Cursor - AH = 13H

This routine clears the text screen and sends the cursor to the home position (0,0 coordinates).

NOTE

This function "erases" any data contained in the status region but leaves the status region implementation in effect.

The required input for this function is as follows:

AH = 13H (function number)

4.10.13 Clear Graphics Screen(s) - AH = 14H

This function clears the graphics screen.

Required input for this function is as follows:

AH = 14H (function number)

4.10.14 Set TTY Status Region Beginning - AH = 15H

This function specifies a beginning line on the screen. The text from this beginning line to the end of the screen is considered the status region. This fucnction can define a status region of one or more lines. This region remains in effect until it is reset. During TTY writes, this area remains intact and everything above this line scrolls as necessary. In order to write to this area, the user should:

1. Read and save the current cursor position.

2. Locate the cursor within the status region.

3. Use one of the write character functions (not the TTY write).

4. Restore the cursor to its original position.

Required input for this function is as follows:

CH = 0 (must always be zero)

CL = Start line of status region
(Valid values are 0 through 24.)

A value of zero (0) for the start line resets the status region implementation. The start line must be a line after the current cursor position, or no status region is implemented.

4.10.15 Set Attribute(s) - AH = 16H

This function provides an alternate method with which to control the following attribute(s).

* Intensity levels 1, 2, and 3 (blue, red, and green)

* Character enable/disable

* Reverse/normal video

* Underline

* Blink

* Alternate character set

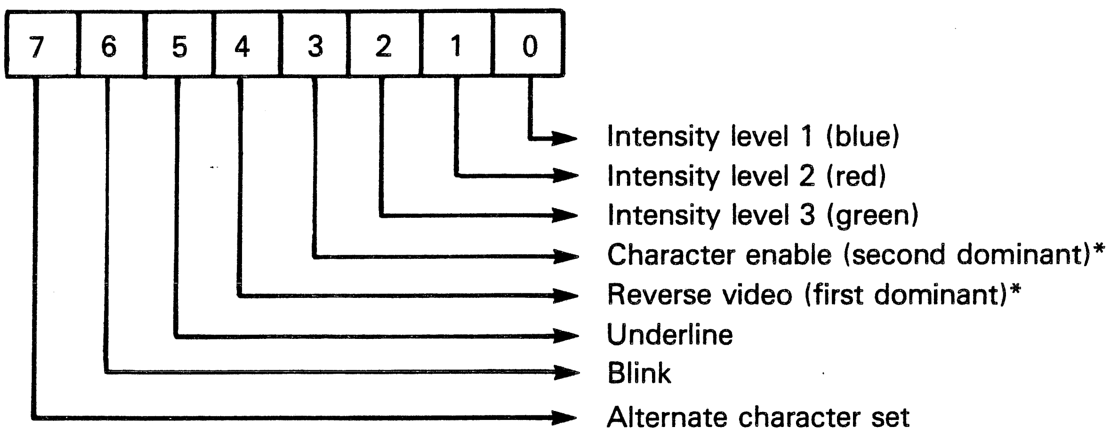
This function sets the specified attribute(s) into the attribute latch, and subsequent characters written to the screen assume the attribute(s). Combining this function with a Write Character (either block or single) at Cursor Position (AH=0AH) function has the same effect as the Write Character/Attribute (either block or single) at Cursor Position (AH=09H) function. The attribute latch remains set to the attribute specified in register BL.

Although more than one attribute can be used, certain combinations do not make sense. For instance, if the character enable attribute is set to a zero, then the character will not appear nor will any of the other attributes except for reverse video.

The required input for this function is shown in Figure 4-3.

BL = Attribute(s) to set

(BL is used to distinguish this function from the change screen attributes function).



* The user can specify more than one attribute. For instance, it is possible to have reverse video with an underlined, blinking, red character. The user can mix the intensity (color) bits for different intensities or colors for a given character.

2223216-24

Figure 4-3 Byte Definition - Set Attribute(s)

4.10.16 Get Physical Display-Begin Pointer - AH = 17H

This function is used to return the physical display-begin pointer to an application. Logically, the display-begin pointer is always at 0,0, but there is a physical address (offset) associated with the beginning of the display that changes from time to time as the screen is scrolled, cleared, or otherwise changed. This routine returns that offset address relative to the CRT memory area whose segment address is DE00H. The screen memory is a 2K-byte contiguous block of RAM. Once the starting location of this block is known to the application, any character on the screen can be accessed. For example, the last character on the screen is located at (DE00H: display-begin +2000) and the eightieth character on the screen (top line, last character on the line)is located at (DE00H:display-begin + 80). This returns the display-begin pointer as follows:

DX = 16-bit display-begin pointer (offset)

Example: DX = 0 implies that the first character on the display resides in memory location DE00:0000H

DX = 150H implies that the first character on the display resides in memory location DE00:0150H

4.10.17 Print TTY String - AH = 18H

With this function, the user can have a contiguous string of characters, of a given length, located in a code segment to be printed (starting at the current cursor position) in a TTY fashion. As with the Write TTY function, this routine executes the control characters CR, LF, BS, and BEL and scrolls the screen if necessary.

Required input for this function is as follows:

BX = Address (offset) of the string*

Where: (BX) byte 0 = length of the string
(BX) byte 1 = first character of the string

* The user's code segment address is obtained from the stack and therefore does not need to be passed as a parameter.

4.10.18 CRT TTY Mode Behavior

The following is a brief description of the behavior of the CRT when used in the TTY mode as well as its behavior when being used in "mixed" modes. The user should read this information carefully, especially if the user mixes non-TTY functions with TTY functions.

Internally, the CRT DSR implements a "phantom" column 81 on each line, which is actually column 1 of the following line. This "phantom" column occurs when a TTY write puts a character in the eightieth column of the current line. If a carriage return (<CR>) command is issued at this point, the cursor moves from the column 81 of the current line back to column 1 of the current line. However, if the cursor is in column 81, reading the cursor position returns (current line + 1, column 0), instead of (current line, column 81). The user must be aware of this before attempting to restore a cursor position which logically came from column 81, because the Set Cursor Position function has no concept of a column 81. This concept disturbs the TTY mode and it restores the cursor to a new logical position, that is, to column 1 of the next line. Although the column 1 position has only one physical location, it can be interpreted as two different logical locations, depending on the current CRT action (mode).

4.10.19 Custom Encoding of the CRT

It is possible for the user to custom encode the characters displayed on the CRT, using the CRT "mapping" function. This mapping allows the applications first to intercept characters (and CRT actions if necessary) then to encode them.

Upon entry to the CRT DSR, a software interrupt is executed, which points to an IRET instruction. An application program can reprogram the IRET to intercept calls to the CRT DSR. The program can thereby "take over" the CRT. This is the typical method used to remap characters to the screen. For instance, this feature can be used to scan through a table, converting English characters to characters in some other language. Another use is intercepting "function calls" (such as scroll or attribute handling) so that the application program can custom encode CRT functions. The user must be careful when performing this operation, however, because it is possible to disturb the data structures of the CRT DSR.

NOTE

> After finishing with this function, the user <u>must</u> restore the vector to its original value. Otherwise, the system could "go away."

After the user enters his mapping routine, he can use all registers except ES, DS, and BP. To use these registers, he must save them, then restore them upon exit. Before using this mapping feature, the user must look at the opcode in register AH to determine if it is a write character request. If so, he must also preserve register AH and any registers associated with the write function contained therein. For example, to map all dollar sign symbols ($) to the percent sign (%), the routine monitors register AH on each call to the CRT DSR. If AH contains a write character opcode, the routine then looks at register AL. If register AL contains 24H (the ASCII code for "$"), the user changes that register to 25H (the ASCII code for "%"), then executes an IRET instruction, returning to the screen with the new character. (The currency symbol returned depends on the internation keyboard being used.) All registers are preserved, but register AL has been changed.

4.11 DISK DSR

Table 4-7 describes the disk device service routines (disk DSR) supported by the Texas Instruments Professional Computer. To access a function, place the proper opcode in register AH, then execute an INT 4DH. On return, all registers are preserved except where stated.

Table 4-7 Disk DSR Opcodes and Functions

A1b

Code	Description
00H	Reset disk system
01H	Return status code (for last operation)
02H	Read sectors
03H	Write sectors
04H	Verify sector CRCs
05H	Null operation
06H*	Verify data
07H*	Return retry status
08H*	Set standard disk interface table (DIT) for unit
09H*	Set DIT address for unit
0AH*	Return DIT address for unit
0BH*	Turn off diskette drive motors

* These functions are primarily for the use of system-level software and utilities.

4.11.1 Reset Disk System - 00H

Input: AH = 00H

Output: AH = 00H

This function causes the disk system to restore itself to a known state. The actions performed for each supported device varies with the requirements of the device and the device-dependent software. In general, the function causes the disk controller(s) to reinitialize before their next use.

4.11.2 Return Status Code - 01H

Input: AH = 01H

Output: AH = 00H
AL = Status code for last disk I/O operation
CF = 0 (No change)

Not all disk DSR functions are I/O operations (this one, for instance). A status is returned in AH for each function, but the status of the last I/O request is always retained for later access (via this function), if desired.

4.11.3 Read Sectors - 02H

Input: AH = 02H
AL = Number of sectors to transfer
CH = Cylinder number
CL = Sector number
DH = Track (surface or side) number
DL = Drive number
ES:BX = Segment:offset of buffer

Output: AH = I/O status code
(For more information, refer to paragraph 4.11.13.)
AL = Number of unprocessed sectors
ES:BX = Segment:offset of the last sector processed*

This function reads data from the disk. Any number of sectors can be transferred subject to memory boundary limitations (The segment's 64K boundary and disk boundaries cannot be crossed.)

* "Last sector processed" means exactly that. Even if the read was in error, the data is transferred to memory.

4.11.4 Write Sectors -03H

Input: AH = 03H
AL = Number of sectors to transfer
CH = Cylinder number
CL = Sector number
DH = Track (surface or side) number
DL = Drive number
ES:BX = Segment:offset of buffer

Output: AH = I/O status code
(For more information, refer to paragraph 4.11.13.)
AL = Number of unprocessed sectors
ES:BX = segment:offset of the last sector processed*

This function writes data to the disk. Any number of sectors can be transfered subject to memory boundary limitations. (The segment's 64K boundary and disk boundaries cannot be crossed.)

* "Last sector processed" means exactly that. If the write is in error, ES:BX points to the data which the DSR is attempting to transfer.

4.11.5 Verify Sector CRCs - 04H

Input: AH = 04H
AL = Number of sectors to transfer
CH = Cylinder number
CL = Sector number
DH = Track (surface or side) number
DL = Drive number
ES:BX = Segment:offset of buffer

Output: AH = I/O status code
(For more information, see paragraph 4.11.13.)
AL = Number of unprocessed sectors
ES:BX = Segment:offset of the last sector processed*

This function verifies the CRCs of the specified sectors. Because this function is handled like an I/O function, ES:BX must be set as though a transfer is to take place although no data is actually transferred. Any number of sectors can be processed subject to memory boundary limitations. (The segment's 64K boundary and disk boundaries cannot be crossed.)

* "Last sector processed" has little meaning in this case because this function does not actually transfer data.

4.11.6 Null Operation - 05H

This function is not currently supported.

4.11.7 Verify Data - 06H

Input: AH = 06H
AL = Number of sectors to process
CH = Cylinder number
CL = Sector number
DH = Track (surface or side) number
DL = Drive number
ES:BX = Segment:offset of buffer

Output: AH = I/O status code
(For more information, see paragraph 4.11.13.)
AL = Number of unprocessed sectors
ES:BX = On error, segment:offset of WORD in error

This function verifies disk data against data in memory. Any number of sectors can be processed subject to memory boundary limitations. (The segment's 64K boundary and the disk boundaries cannot be crossed.)

4.11.8 Return Retry Status - 07H

Input: AH = 07H

Output: AH = 00H
AL = Soft error status of last I/O operation

This function is similar to the Return Status Code function. It returns the "soft" error status of the last operation. Soft error refers to an error that did not recur when the last operation was retried.

4.11.9 Set Standard Disk Interface Table - 08H

Input: AH = 08H
AL = Standard DIT number
(Valid values are 0 through 3.)
DL = Diskette drive number
(Valid values are 0 through 3.)

Output: AH = Error status
(For more information, see paragraph 4.11.13.)

(Note: This function is used by the operating system software.)

Disk interface tables (DITs) are data structures containing information that the device-dependent part of the DSR uses to interface with the device-dependent code for a specific disk device.

With this function, the user can set a diskette drive to one of four standard configurations by setting the drives's DIT. The standard DIT numbers are defined as follows:

Number	Description
0	Single sided, 48 tpi, 8 sectors/track, 512-byte sectors
1	Double sided, 48 tpi, 8 sectors/track, 512-byte sectors
2	Single sided, 96 tpi, 8 sectors/track, 512-byte sectors
3	Double sided, 96 tpi, 8 sectors/track, 512-byte sectors

4.11.10 Set DIT Address for Drive - 09H

Input: AH = 09H
DL = Disk drive number
(Valid value is 0 through 7.)
ES:BX = Segment:offset of DIT for drive

Output: AH = Error status
(For more information, see paragraph 4.11.13.)

(Note: This function is used by the operating system software.)

Disk interface tables (DITs) are data structures containing information that the device-dependent part of the DSR uses to interface with the device-dependent code for a specific disk device.

With this function, the user can set any disk to a nonstandard configuration. The disk drives are dynamically linked to the system by this mechanism.

4.11.11 Return DIT Address for Drive - 0AH

Input: AH = 0AH
DL = Disk drive number
(Valid value is 0 through 7.)

Output: AH = Error status
(For more information, see paragraph 4.11.13.)
ES:BX = Segment:offset of DIT for drive

(Note: This function is used by the operating system software.)

Disk interface tables (DITs) are data structures containing information that the device-independent part of the DSR uses to interface with the device-dependent code for a specific disk device.

With this function, the user can access a drive's DIT for information and verification purposes.

4.11.12 Turn Off All Diskette Drives - 0BH

Input: AH = 0BH

Output: AH = 0
ES:BX = not preserved

(Note: This function is used by the operating system software.)

During regular operation, the diskette drive motors are left ON for a short period following a read or write operation, thereby saving the time the motor would use to come up to speed. Some applications, notably diagnostics, require assurance that the motors are not running.

4.11.13 Status Codes

All functions return a status code in register AH and an error flag in CF. If the carry condition is set (CF = 1), then an error has occurred and AH contains the error code. If the no-carry condition is set (CF = 0), no error has occurred and AH contains a zero. The error codes are given in Table 4-8.

Table 4-8 Error Codes

Value	Description
00H	No error
80H	Time-out - drive not ready or hardware failed
40H	Seek failed - track not found
20H	Controller hardware failed
10H	CRC error
08H	Data request error - controller failure
04H	Record (sector) not found
02H	No data - bad disk format
01H	Command error - bad opcode or parameter
03H	Disk write protected
05H	Data did not verify
09H	I/O transfer crosses 64K byte boundary

4.11.14 Disk Interface Tables (DITs)

The Disk Interface Table (DIT) structure interfaces device-specific code with the generalized disk driver code.

Because DITs contain read-only data exclusively, they can be placed in ROM.

The structure of a DIT is shown in Figure 4-4.

← 16 bits →

Offset	Field		Description
00H	DITDIR		Long pointer to disk interface routine
02H			
04H	DITSEC		Sector size in bytes
06H	DITTRK	DITCYL	Track size in sectors; cylinder size in tracks
08H	DITDSK	DITERR	Disk size in cylinders; error retry limit

All other fields depend on the code requirements of the specific device.

A. General DIT Structure

← 16 bits →

Offset	Field		Description
00H	FLPDIR		Long pointer to diskette interface routine
02H			
04H	DITSEC		Sector size in bytes
06H	DITTRK	DITCYL	Track size in sectors; cylinder size in tracks
08H	DITDSK	DITERR	Disk size in cylinders; error retry limit
0AH	PRCOMP		Threshold track number for changing write precompensation

B. Diskette Drive DIT Structure

2223216-25

Figure 4-4 DIT Structure

The following procedure shows how to set up the disk DSR in order to access a flexible disk (floppy) with a "nonstandard" format. ("Nonstandard" is a format that usually is not supported by the Texas Instruments Professional Computer.)

```
MOV     AH, 0AH                      ; Set "return DIT address" opcode
MOV     DL,<unit number>             ; Any floppy disk unit (0 - 3)
INT     4DH                          ; Call disk DSR
LES     BX, ES:(DWORD PTR [BX]       ; ES:BX := address of floppy code
MOV     <your DIT>+0,BX              ; Put address of floppy-specific
MOV     <your DIT>+2,ES              ;   code in your own DIT

     .
     .
     .
  <initialize your DIT>      ; Do whatever else you need to your DIT
     .
     .
     .
MOV ES,SEG<your DIT>         ; EX:BX = address of your DIT
MOV BX, OFFSET<your DIT>
MOV AH,9                     ; Set "SET DIT ADDRESS" opcode
MOV DL,<unit number>         ; Unit number
INT 4DH                      ; Call disk DSR
     .
     .
     .
     .
```

NOTE

The floppy-specific code comprehends only double-density (MFM) recording format. It does not know how to access single-density (FM) recording format diskettes.

4.12 KEYBOARD DSR

This subsection describes the keyboard DSR and the functions it provides to the system or application programs that use it. It also shows the various codes returned by the DSR for the standard configuration of the keyboard.

The keyboard DSR functions are located in the system ROM and are accessed through the 8088 software interrupt mechanism (essentially an address-independent subroutine call). The typical user of the keyboard DSR is the system interface code (the BIOS). Each operating-system-dependent BIOS resides on a particular operating system diskette and is loaded into RAM during disk boot.

The functions described in this subsection access a buffer that is controlled by the keyboard interrupt service routine. All encoding and any special handling (described in subsequent paragraphs) occurs in the interrupt service routine. All discussions of keyboard mapping vectors refer to actions occurring during the servicing of the keyboard hardware (not software) interrupt.

Placing an opcode in register AH and executing an INT 4AH chooses the desired function. All registers except AX are preserved. The functions of the keyboard DSR are described in the following paragraphs.

4.12.1 Initialization Logic

The code for this function is automatically executed during power-up or reboot and is not directly available to the user. It performs diagnostics on the keyboard hardware, sends to it the required initialization sequences, and initializes the DSR internal data areas.

4.12.2 Read Keyboard Input - AH = 0

This function reads and removes the current character (if any) from the keyboard buffer. The character value is returned in register AX. If no character is ready, the DSR waits until one is received before it returns to the caller. This character has already been fully encoded (Table 4-10 lists the ASCII codes.) Typically, the encoded ASCII character is returned in register AL, and register AH contains 00. If AL = 0, then the coded value in AH corresponds to one of the various function keys. (Table 4-11 lists the non-ASCII codes for the function keys.)

4.12.3 Read Keyboard Status - AH = 1

This function determines that a character is ready at the keyboard but does not actually read it. If no character is waiting, it returns with the Z-flag set (ZF = 1).. If the Z-flag is reset (ZF = 0), a character is available to be read. The character value is returned in AX, but is not removed from the keyboard buffer.

4.12.4 Read Keyboard Mode - AH = 2

This function determines the current mode of the keyboard. The mode value is returned in register AL in the format shown in Figure 4-5. The definition of the byte is as follows.

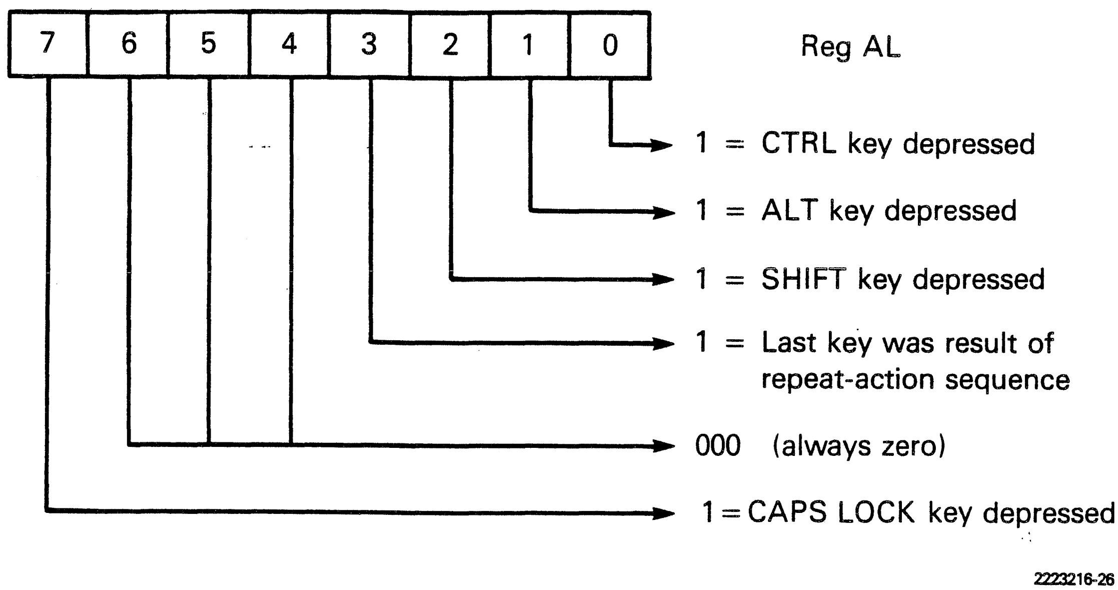


Figure 4-5 Byte Definition - Keyboard Modes

Because the "mode" applies to the last character typed and not necessarily to the one at the front of the queue, this function return valid information only if the keyboard buffer contains one or less characters. In order to use this function, read the key normally, then make a status check to ensure that the buffer is empty. When the buffer is empty, the mode reading will be valid.

Use this function only if it is necessary to know the state of the mode when the last character was typed. See the section entitled "Custom Encoding of the CRT" in Section 4 for an explanation of remapping the keyboard.

4.12.5 Flush Keyboard Buffer - AH = 3

This function is used to "flush" (empty) the keyboard type-ahead buffer. It simply resets the queue pointers, which effectively empties the buffer.

4.12.6 Keyboard Output - AH = 4

This function sends the keyboard command in AL directly to the keyboard, with appropriate handshaking. On return, the Z-flag has the status of the operation. If the Z-flag is set (ZF=1), the command was performed correctly; otherwise (ZF=0), an error was made. The keyboard commands sent by the CPU are given in Table 4-9.

Table 4-9 Keyboard Commands

Register AL	Function Performed
00	Performs a power-up reset and installs default parameters
01*	Turns repeat-action feature ON
02	Turns repeat-action feature OFF
03	Locks the keyboard
04*	Unlocks the keyboard
05	Turns keyclick ON**
06*	Turns keyclick OFF**
07	Resets
08	Returns keyboard ROM version

* Indicates the default value.
** Keyclick requires a hardware modification. (It is not presently supported.)

These commands are intended for "one-shot" use, to set the keyboard mode at power-up. Although they may be sent at any other time, the overhead of receiving several commands can cause the keyboard to miss fast keystrokes. There are other ways to implement these commands. A CRT emulator program may be required to turn repeat-action on and off in response to escape sequences from a host. For example, if an application needs to set/reset the repeat-action mode, or to lock/unlock the keyboard in real time, these functions can be programmed into a keyboard mapping routine. Refer to paragraph 4.10.19.

4.12.7 Put Character Into Keyboard Buffer - AH = 5

This function places the 16-bit value in BX directly into the keyboard buffer. On return, if the Z-flag is reset (ZF=0), the character was placed in the buffer (this is the usual case). If the Z-flag is set (ZF=1), it means that the buffer was full and the character was not placed in the buffer. (The character remains in BX.) Assuming that the buffer was empty at the start, and that no keys on the keyboard have been pressed, a Read Keyboard Input (AH=0) function call retrieves this character. Any 16-bit value can be placed into the buffer, but unless the user has some explicit application that understands "strange" characters from the keyboard, it is recommended that only standard characters generated by the keyboard be used. The format for the characters is the same as that given in the Read Keyboard Input function.

To place a normal ASCII character into the buffer, make the function call with the character value in BL and zero in BH. To place function keys into the buffer, make the function call with the extended function value in BH, and zero in BL. (See Table 4-10 and Table 4-11.)

This function is useful when a program needs characters to appear as though they had been typed. Two examples follow.

* An application can disable the operating system printer "echo" feature by inserting the appropriate "echo off" character (CTRL N for MS-DOS) into the buffer during initialization. The operating system sees this as just another key and turns off the echo.

* Many operating systems lack a chaining feature, and this function can provide one. Immediately before a program terminates, flush the keyboard buffer, then place characters simulating a typed command into the buffer. When the program terminates, the operating system takes over, reads the keyboard buffer, and performs that command (which could invoke a second program, thereby "chaining" programs).

4.12.8 General Keyboard Layout

The outline of the keyboard and the key-position numbers associated with each of the keys are shown in Figure 4-6. The numbers in the upper right-hand corner of the keys are the scan codes sent from the keyboard. These codes are used internally by the keyboard DSR to encode a key when pressed. The mode keys (marked ***) do not generate a scan code.

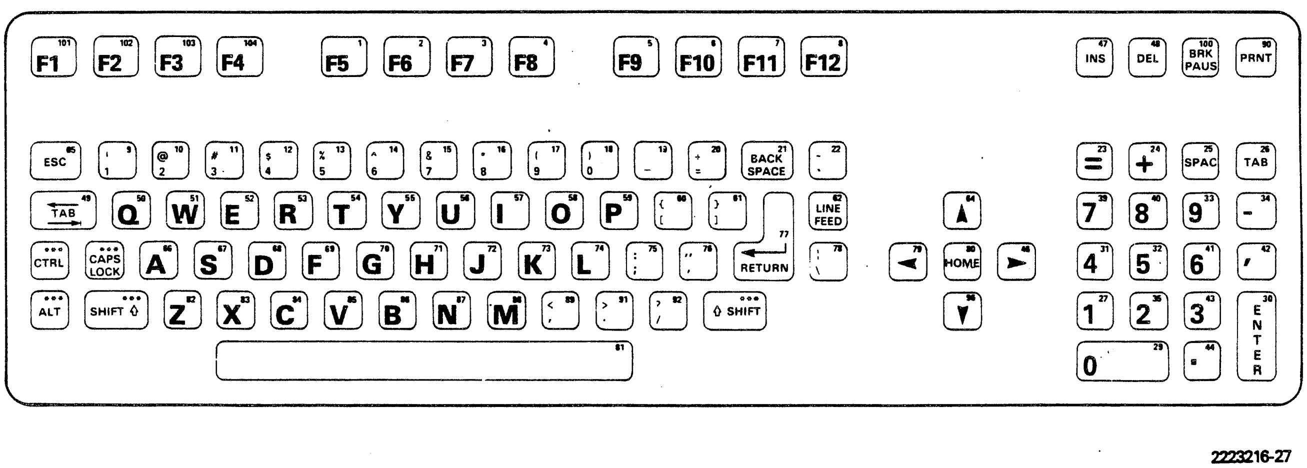


Figure 4-6 General Keyboard Layout Showing Scan Codes

4.12.9 Character Codes

Table 4-10 lists the character and extended function codes returned by the keyboard DSR. The modes are handled internally by the keyboard DSR, and the returned code reflects the mapping shown in this table.

Table 4-10 Standard Keyboard Character Codes

Key #	Normal		SHIFT		CTRL		ALT		Comments
01	f5	3F*	sf5	58*	cf5	62*	af5	6C*	F5
02	f6	40*	sf6	59*	cf6	63*	af6	6D*	F6
03	f7	41*	sf7	5A*	cf7	64*	af7	6E*	F7
04	f8	42*	sf8	5B*	cf8	65*	af8	6F*	F8
05	f9	43*	sf9	5C*	cf9	66*	af9	70*	F9
06	f10	44*	sf10	5D*	cf10	67*	af10	71*	F10
07	f11	45*	sf11	08*	cf11	0A*	af11	0C*	F11
08	f12	46*	sf12	09*	cf12	0B*	af12	0D*	F12
09	1	31	!	21	---	--	alt1	78*	
10	2	32	@	40	Fnul	03*	alt2	79*	
11	3	33	#	23	---	--	alt3	7A*	
12	4	34	$	24	---	--	alt4	7B*	
13	5	35	%	25	---	--	alt5	7C*	
14	6	36	^	5E	RS	1E	alt6	7D*	
15	7	37	&	26	---	--	alt7	7E*	
16	8	38	*	2A	---	--	alt8	7F*	
17	9	39	(	28	---	--	alt9	80*	
18	0	30	)	29	---	--	alt0	81*	
19	-	2D	_	5F	US	1F	alt-	82*	
20	=	3D	+	2B	---	--	alt=	83*	
21	BS	08	BS	08	DEL	7F	---	--	Back space
22	'	60	~	7E	---	--	---	--	
23	=	3D	=	3D	=	3D	pf1	8C*	Numeric =
24	+	2B	+	2B	+	2B	pf2	8D*	Numeric +
25	SP	20	SP	20	SP	20	pf3	8E*	Numeric SPACE
26	HT	09	Bktab	0F*	HT	09	pf4	8F*	Numeric TAB
27	1	31	1	31	1	31	---	--	Numeric 1
28	---	--	---	--	---	--	---	--	(unused)
29	0	30	0	30	0	30	---	--	Numeric 0
30	CR	0D	CR	0D	CR	0D	---	--	Numeric ENTER
31	4	34	4	34	4	34	---	--	Numeric 4
32	5	35	5	35	5	35	---	--	Numeric 5
33	9	39	9	39	9	39	---	--	Numeric 9
34	-	2D	-	2D	-	2D	---	--	Numeric -
35	2	32	2	32	2	32	---	--	Numeric 2

Table 4-10. Standard Keyboard Character Codes (Continued)

Key #	Normal		SHIFT		CTRL		ALT		Comments
36	---	--	---	--	---	--	---	--	(Unused)
37	---	--	---	--	---	--	---	--	(Unused)
38	---	--	---	--	---	--	---	--	(Unused)
39	7	37	7	37	7	37	---	--	Numeric 7
40	8	38	8	38	8	38	---	--	Numeric 8
41	6	36	6	36	6	36	---	--	Numeric 6
42	,	2C	,	2C	,	2C	---	--	Numeric ,
43	3	33	3	33	3	33	---	--	Numeric 3
44	.	2E	.	2E	.	2E	---	--	Numeric .
45	---	--	---	--	---	--	---	--	(Unused)
46	C-rt	4D*	sC-rt	8A*	cC-rt	74*	aC-rt	4E*	Right Arrow
47	Ins	52*	sIns	28*	cIns	29*	aIns	2A*	INS
48	Del	53*	sDel	38*	cDel	39*	aDel	3A*	DEL
49	HT	09	Bktab	0F*	HT	09	---	--	TAB
50	q	71	Q	51	DC1	11	altQ	10*	
51	w	77	W	57	ETB	17	altW	11*	
52	e	65	E	45	ENQ	05	altE	12*	
53	r	72	R	52	DC2	12	altR	13*	
54	t	74	T	54	DC4	14	altT	14*	
55	y	79	Y	59	EM	19	altY	15*	
56	u	75	U	55	NAK	15	altU	16*	
57	i	69	I	49	HT	09	altI	17*	
58	o	6F	O	4F	SI	0F	altO	18*	
59	p	70	P	50	DLE	10	altP	19*	
60	[	5B	{	7B	ESC	1B	---	--	
61	]	5D	}	7D	GS	1D	---	--	
62	LF	0A	LF	0A	cLF	75*	aLF	4F*	Line Feed
63	---	--	br	--	---	--	---	--	(Unused)
64	C-up	48*	sC-up	88*	cC-up	84*	aC-up	49*	Up Arrow
65	ESC	1B	ESC	1B	ESC	1B	---	--	ESC
66	a	61	A	41	SOH	01	altA	1E*	
67	s	73	S	53	DC3	13	altS	1F*	
68	d	64	D	44	EOT	04	altD	20*	
69	f	66	F	46	ACK	06	altF	21*	
70	g	67	G	47	BEL	07	altG	22*	

Table 4-10. Standard Keyboard Character Codes (Concluded)

Key #	Normal	SHIFT	CTRL	ALT	Comments
71	h 68	H 48	BS 08	altH 23*	
72	j 6A	J 4A	LF 0A	altJ 24*	
73	k 6B	K 4B	VT 0B	altK 25*	
74	l 6C	L 4C	FF 0C	altL 26*	
75	; 3B	: 3A	--- --	--- --	
76	' 27	" 22	--- --	--- --	
77	CR 0D	CR 0D	CR 0D	--- --	Return
78	\ 5C	| 7C	FS 1C	--- --	
79	C-lf 4B*	sC-lf 8B*	cC-lf 73*	aC-lf 4C*	Left Arrow
80	Home 47*	sHome 86*	cHome 77*	aHome 85*	HOME
81	SP 20	SP 20	SP 20	SP 20	Space bar
82	z 7A	Z 5A	SUB 1A	altZ 2C*	
83	x 78	X 58	CAN 18	altX 2D*	
84	c 63	C 43	ETX 03	altC 2E*	
85	v 76	V 56	SYN 16	altV 2F*	
86	b 62	B 42	STX 02	altB 30*	
87	n 6E	N 4E	SO 0E	altN 31*	
88	m 6D	M 4D	CR 0D	altM 32*	
89	, 2C	< 3C	--- --	--- --	
90	Ptogl 72*	*** **	--- --	--- --	PRINT
91	. 2E	> 3E	--- --	--- --	
92	/ 2F	? 3F	--- --	--- --	
93	--- --	--- --	--- --	--- --	(Unused)
94	--- --	--- --	--- --	--- --	(Unused)
95	--- --	--- --	--- --	--- --	(Unused)
96	C-dn 50*	sC-dn 89*	cC-dn 76*	aC-dn 51*	Down Arrow
97	---- --	--- --	--- --	--- --	(Unused)
98	--- --	--- --	--- --	--- --	(Unused)
99	--- --	--- --	--- --	--- --	(Unused)
100	Ppau **	Pbrk **	--- --	--- --	BRK/PAUS
101	f1 3B*	sf1 54*	cf1 5E*	af1 68*	F1
102	f2 3C*	sf2 55*	cf2 5F*	af2 69*	F2
103	f3 3D*	sf3 56*	cf3 60*	af3 6A*	F3
104	f4 3E*	sf4 57*	cf4 61*	af4 6B*	F4

Notes to Table 4-10:

1. Key # is shown in Figure 4-6.

2. In the "Normal","SHIFT", "CTRL", and "ALT" columns, both the "graphic" and the hexadecimal values of the character are given in the form: GGG HH. Mnemonics are used for the "graphic" descriptions of the function keys. These are generally self-explanatory: a leading a, s, or c indicates ALT, SHIFT, or CTRL, respectively. For example, f1 is the F1 function key; af1 is the F1 key pressed while

holding down the ALT key. C-rt means cursor right (right arrow), and cLF = CTRL linefeed.

3. Entries consisting of "--- --" indicate that the combination is suppressed within the keyboard DSR.

4. Entries consisting of "xxx **" indicate special handling in the form of direct action by the keyboard DSR. (For details, see paragraph 4.12.14.)

5. Normal (ASCII) characters are returned in register AL with the scan code key number in AH.

6. Entries consisting of "xxx yy" are returned with AL=0 and the indicated value (yy) in AH.

7. An asterisk after a number means extended codes, listed in Table 4-11.

4.12.10 Extended Codes

The "extended" codes are non-ASCII codes. They represent special function keys on the keyboard. To distinguish these codes, register AL contains 00 upon returning from a Read Keyboard (AH=1 or AH=2) function call, and the extended code is in register AH. The code range (00H through FFH) includes normal ASCII codes. The extended codes are given in Table 4-11. Use the mnemonics to cross-reference with Table 4-10.

Table 4-11 Extended Function Codes

MSD* / LSD	0	1	2	3	4	5	6	7	8
0	Pbrk	altQ	altD	altB	f6	C-dn	cf3	af9	alt9
1	Ppau	altW	altF	altN	f7	aC-dn	cf4	af10	alt0
2		altE	altG	altM	f8	Ins	cf5	Ptogl	alt-
3	Fnul	altR	altH		f9	Del	cf6	cC-lf	alt=
4		altT	altJ		f10	sf1	cf7	cC-rt	cC-up
5		altY	altK		f11	sf2	cf8	cLF	aHome
6		altU	altL		f12	sf3	cf9	cC-dn	sHome
7		altI			Home	sf4	cf10	cHome	
8	sf11	altO	sIns	sDel	C-up	sf5	af1	alt1	sC-up
9	sf12	altP	cIns	cDel	aC-up	sf6	af2	alt2	sC-dn
A	cf11		aIns	aDel		sf7	af3	alt3	sC-rt
B	cf12			f1	C-lf	sf8	af4	alt4	sC-lf
C	af11		altZ	f2	aC-lf	sf9	af5	alt5	pf1
D	af12		altX	f3	C-rt	sf10	af6	alt6	pf2
E		altA	altC	f4	aC-rt	cf1	af7	alt7	pf3
F	Bktab	altS	altV	f5	aLF	cf2	af8	alt8	pf4

* MSD = most significant digit; LSD = least significant digit

4.12.11 Keyboard Modes

In the standard keyboard, the mode keys have the effect shown in Table 4-11. The latching (push-push) CAPS LOCK key affects the alphabetic keys (50-59, 66-74, and 82-88 on the standard keyboard) by forcing the SHIFT mode. Normally the alphabetic keys produce lowercase characters, and the SHIFT key temporarily causes them to be uppercase. When the CAPS LOCK mode is invoked (the CAPS LOCK key is latched down and the LED in the CAPS LOCK key lights), the alphabetic keys produce uppercase and the SHIFT key has no further effect (on the alphabetic keys).

In the standard encoding, the only valid combination of mode keys is CTRL/ALT/DEL, which is used for system reset. Simultaneously pressing the CTRL, ALT, and DEL keys results in the keyboard DSR initiating the equivalent of a system power-up reboot. The action is handled internally by the DSR and does not return a code. This function is "hardwired" and cannot be disabled. In any other case, when two or more mode keys are pressed simultaneously, only one is recognized. The order of precedence, beginning with the highest, is as follows:

ALT, CTRL, SHIFT, and CAPS LOCK

The ALT key has a special use, letting the user enter any character code (00H-0FFH) from the keyboard. When the ALT key is held down and the decimal value of the desired character is typed on the numeric keypad with three keystrokes, the value is returned to the application as a normal character directly through the Read Keyboard Input (AH=0) function. If fewer than three digits are typed, the next non-ALT key struck sends the currently accumulated ALT/NUM value (from the first one or two keystrokes). If the first one or two keystrokes were the zero key, the next key pressed sends its normal character, because the zero is simply a "place keeper" and adds nothing to the ALT/NUM value. Pressing more than three keys sends the accumulated value and starts a new three-keystroke sequence.

Example:

ALT 003 places the value for an ETX in the keyboard buffer.

ALT 3, followed by any non-ALT key performs the same function.

4.12.12 Type-Ahead Buffer

The DSR implements a circular type-ahead queue, which can buffer up to 15 keystrokes. (Each keystroke is 2 bytes.) If the queue is filled, entering further characters from the keyboard sounds the system beeper. The Flush Keyboard Buffer (AH=3) function resets the queue pointers, effectively emptying the buffer.

4.12.13 Repeat-Action Feature

If the repeat-action feature (the default) is enabled, there is a half-second delay and all keys become repeat-action at a 15-cps rate. Repeat-action characters are ignored when the queue currently contains more than one pending character. This means that the application does not have to worry about the repeat-action "coasting" problem. That is, if the application does not or cannot read the keyboard input faster than the repeat-action rate, the undesired repeat-action characters are not queued and the keyboard does not get ahead of the application.

4.12.14 Special Handling

These paragraphs describe functions handled by the keyboard DSR. Several of these require immediate reaction (for example, pausing the output routine so a fast-scrolling screen can be read). Most of the keyboard DSR functions are implemented with the software interrupt facility of the 8088 microprocessor.

Each of the defined interrupt vectors points to some default piece of code that either does nothing (for example, a single IRET instruction) or performs some system function. An application program can change these interrupt vectors in order to gain direct access to a function. However, the application must preserve the original contents of the vector and restore it before terminating and returning to the system. If the application routine is used, it must end with an IRET or the equivalent (FAR) RET 2, which allows flags to be passed.

The stack used is the internal stack of the keyboard interrupt service routine and only 10 levels (20 bytes) of stack are available to the user's routine. Interrupts are disabled when the user routine is entered (by the INT instruction). Interrupts should be re-enabled immediately unless it is necessary for them to remain disabled. Registers AX, BX, CX, DI, and ES can be used (information is passed in AX); any others must be preserved. When the available stack is too small, the routine must switch to an internal stack of sufficient size (including 8 bytes for possible interrupts). Also, the routine is executed as a part of the keyboard interrupt service routine, which means that no other keystrokes are accepted until the user routine finishes and returns. The normal way to communicate with the outside world (outside the service routine) is to set a flag and watch for it in the application. This, for example, is how the BREAK function is implemented in MS-DOS. Control should not be retained by the user's routine unless a complete system initialization is to be performed.

4.12.15 User-Available Interrupts

The following is a summary of the software interrupts (performed by the keyboard DSR) that can be used by application programs. The interrupts are presented in their order of execution. The number in parentheses, the "interrupt type," is used in an interrupt instruction. The absolute address of the corresponding vector is the interrupt type times 4. As an example, the address of the keyboard mapping vector is 5BH x 4 = 16CH. Any of the special key interrupt functions can be bypassed by re-encoding the key code. For more information on the key code, refer to paragraph 4.10.19.

The keyboard DSR interrupts and their mapping vectors are:

- Keyboard mapping (5BH)
- Program pause (5CH) *
- Program break (5DH) *
- Print screen (5EH) *
- Keyboard queueing (5FH)

* These interrupts occur after internal encoding.

4.12.15.1 Keyboard Mapping. This interrupt is performed each time a key is pressed but before it is encoded, allowing the user to encode the key. When the user encodes the key, the DSR places the key code in the queue and performs the keyboard queuing (5FH) interrupt. Otherwise, the DSR encodes the key, checks for the special keys, and then queues the key code, causing the keyboard queuing interrupt. For more information on using this interrupt to remap the keyboard, refer to paragraph 4.10.19.

4.12.15.2 Program Pause. Pressing the (unshifted) <u>BRK/PAUS</u> key causes a software interrupt and allows the user to perform an action or return a key code. It returns an extended code (refer to Table 4-11) to the caller if desired. At system power-up, the vector is set so that the <u>PAUS</u> key sequence causes a screen hold, which stops a fast-scrolling screen. An application program can change the interrupt vector in order to support a pause function of its own, but the program is responsible for remembering the original vector and restoring it before terminating.

The carry flag determines the action of the keyboard DSR on return from the software interrupt. If the carry flag is set, the DSR does nothing else and simply exits. If the carry flag is reset, then the character value in AX is placed into the queue. Before the software interrupt is executed, the carry flag is reset and the extended code for the program pause function is placed in AX. Therefore, if an IRET instruction is used to return instead of the default ROM pause routine, the DSR returns the program pause function code to the application. Because the carry flag is used to pass information, the IRET instruction must be simulated with a (FAR) RET 2 if the user needs to return with the carry flag set. (The IRET instruction restores flags to their pre-interrupt state.)

4.12.15.3 Program Break. Pressing the (shifted) BRK/PAUS key causes a software interrupt and allows the user to perform an action or return a key code. It can be set to return an extended code (see Table 4-11) to the caller, if desired. During power-up initialization, this interrupt vector is set to point to an IRET instruction so that the BRK key sequence is ignored other than returning the break code. An application program can change the interrupt vector in order to support a break function of its own. However, the program is responsible for preserving the original contents of the vector and restoring it before terminating. For more information on the encoding/software-interrupt technique, see paragraph 4.12.15.

4.12.15.4 Print Screen. Pressing the SHIFT and PRNT keys causes another software interrupt. The user can perform an action or return a key code. This interrupt normally vectors to an IRET instruction within the ROM. The DSR checks the carry flag upon return, as described in paragraph 4.12.15.

The carry flag is set before the interrupt is executed, so that when the routine consists only of an IRET, the key is effectively ignored. This can be (and is, by the MS-DOS BIOS) patched so that it vectors to an actual print screen routine. This routine executes as a part of the keyboard interrupt service routine and, therefore, cannot be interrupted by another keystroke. The preferred way to handle the Print Screen function is to use this interrupt to start the Print Routine (in the background) then return immediately, thereby reenabling the keyboard.

4.12.15.5 Keyboard Queueing. This software interrupt occurs every time a character, whether encoded by the DSR or by the user, is placed in the type-ahead buffer. This interrupt lets the real-time OS know when there is a character to read. The user can choose to ignore the key (not queueing the keycode). Refer to paragraph 4.12.15 for keyboard queuing interrupt conditions.

4.12.16 Custom Encoding

An application program can encode the keyboard using this function. Each time a key is pressed on the keyboard, the keyboard sends one or two key codes to the DSR. The mode keys are handled internally. (For more information, refer to paragraph 4.12.17.) The DSR performs a software interrupt each time it receives a key code (not including the mode keys). Normally the interrupt vector points to an IRET instruction. An application program can reprogram the vector to intercept these key codes. Because everything comes through this vector, the application can take control of everything but the system reset combination (CTRL/ALT/DEL). The routine that intercepts the key codes typically scans through some tables to encode its special keys, then executes a (FAR) RET 2 instruction.

NOTE

> It is essential that the application restore the vector to its original value after completion. Otherwise, the system will crash when the special encoding routine is later written over.

When the software interrupt is performed (from the keyboard ISR) the keyboard scan code (including the repeat-action bit, if set) is in AL, the mode byte is in AH (the mode byte is shown in figure) and the carry flag is set (CF=1). If the carry flag is reset (CF=0) when returned from the interrupt, then the standard encoding is bypassed. Instead, the values in AL and AH are placed directly into the type-ahead buffer. This is one way to change the standard encoding of the keyboard.

If the carry flag is set, and the value of AL is returned as 0FFH, the keystroke is ignored entirely, and nothing is placed in the buffer. This can be used when the special handling routine performs some function directly and does not need to send a character. The repeat-action bit is included in the scan code as the high bit of AL and in the mode byte as bit 3 of AH. The user can choose which of the two is more accessible to his particular routine.

If the scan code is used in a table look-up or a direct comparison, the user must strip off the (possible) repeat-action bit (the instruction is AND AL, 7FH). Because this is a software interrupt, the IRET instruction must be simulated with a (FAR) RET 2 in order to pass flags back.

4.12.17 Keyboard Interface Protocol

Pressing a key on the keyboard sends a byte representing the key position to the keyboard DSR. If the state of the mode keys (SHIFT, ALT, CAPS LOCK, and CTRL) has changed since the last keystroke, the key-position byte is preceded by a byte showing the current status of the mode keys. The mode byte is never sent alone. It will always be followed by the key-position byte.

The mode byte is never sent during a repeat-action transmission, because it is sent only if the mode has changed since the last transmission. The mode cannot change during the repeat-action function.

The second byte (key position) contains a repeat-action key bit (bit 7). This bit is set to 1 during a repeat-action key transmission, and reset to 0 during a non-repeat-action transmission. If the key is still pressed after a half-second delay, the code is sent again, this time with bit 7 set to 1. The keyboard remapping routine uses this bit to suppress the repeat-action key function when necessary.

All communication with the keyboard is:

* Asynchronous
* Serial
* 8 data bit
* 1 stop bit
* Even parity.

The keyboard transmits its data at 2440 bps and receives its commands at 305 bps.

Both bytes have similar formats, as shown in Figure 4-7. However, bits 3 through 6 of the mode key status byte are all set to 1.

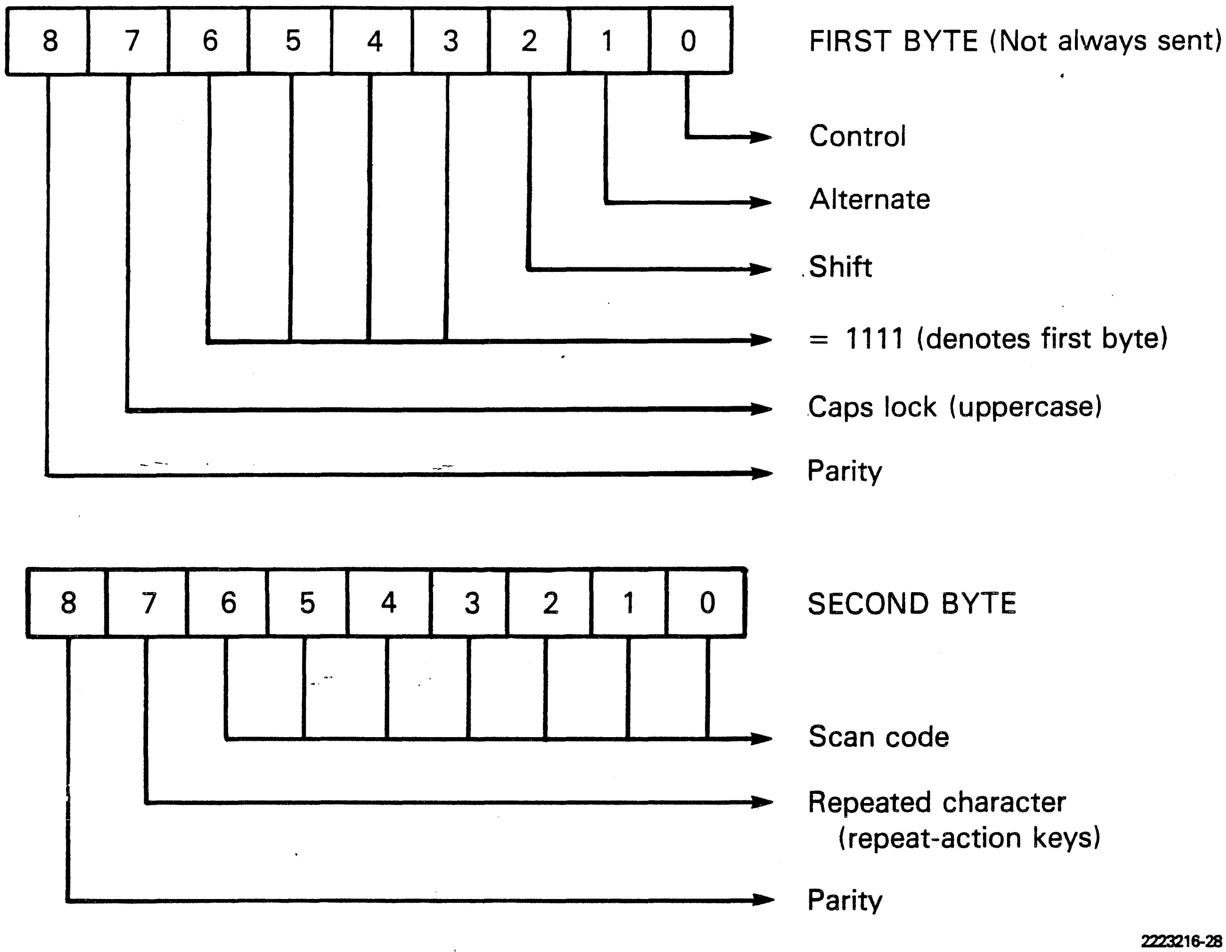


Figure 4-7 Byte Definition - Keycode

The keyboard understands several commands, as explained in the Keyboard Output (AH=4) function, and the keyboard generally acknowledges each command.

The codes sent by the keyboard (refer to Tables 4-10 and 4-11) range from scan code 01 through scan code 104 (01H through 68H). The spare scan codes (from 69H through 6FH) will possibly be assigned in the future. If so, the size of the standard encoding tables will also be increased. Codes 70H through 73H are status codes returned by the keyboard in response to commands. Codes 74H through 77H are unused but reserved, and codes 78H through 7FH are for encoding the mode key status byte. For more specific information, refer to the paragraph entitled "Receiving and responding to commands from the system unit" in Section 2.

4.13 PARALLEL PRINTER PORT DSR

The following paragraphs describe the functions that the parallel printer port DSR provides to the system or application programs that use it.

The printer DSR provides routines to implement a Centronics-compatible parallel port interface. The user is able to output characters, get printer status, and initialize the printer.

The printer DSR functions, located in the system ROM, are accessed through the software interrupt mechanism of the 8088 microprocessor. To choose a function, place the opcode in register AH, place zeros in register DL, and execute an INT 4BH instruction. (For an explanation of register DL, see paragraph 4.13.4.) All registers are preserved except AH, which always returns with the printer status. (See paragraph 4.13.3.)

The functions available are:

Output Character to Printer (AH=0, DL=0)
Initialize Printer (AH=1, DL=0)
Return Printer Status (AH=2, DL=0)

4.13.1 Output Character to Printer - AH = 0, DL = 0

This function sends the character in AL to the printer port. The BUSY signal from the printer is checked before sending the character. If the printer is still busy after approximately 0.33 s, the DSR sets the time-out bit in the status byte (in AH) and returns. If the printer is not busy, the DSR returns with the time-out bit reset. Any unusual conditions on the status signals from the printer cause the printer to go BUSY. Time-out also occurs if the printer sets FAULT, PAPER OUT, or NOT SELECT. The printer can also set BUSY, causing a time-out.

It generally is not advisable to rely on the time-out of the printer output routine during regular use, especially if one is using the DSR from the printer task of a real-time OS. This time-out is a software loop and causes the application to "hang" during the time-out period. The preferred method has the application watching the BUSY signal through the printer status call so that the application can implement and control a time-out.

The standard sequence used to print a character is:

```
REPEAT
    Interrupt 4BH with AH = 2 and DL = 0 (see paragraph 4.13.3,
    "Return Printer Status."
UNTIL
    STATUS = NOT BUSY
END

INTerrupt 4BH with AH = 0, DL = 0 and AL = <character>
IF STATUS = (time-out)
THEN
    <handle the error>  (FAULT or PAPER OUT or (NOT SELECTED))
END
```

Note:
Refer to Figure 4-8 for byte definition of the Return Printer Status function.

4.13.2 Initialize Printer - AH = 1, DL = 0

This function activates the INIT signal on the interface causing the printer to perform the equivalent of a power-up reset. The specific action taken is printer-dependent (refer to the appropriate printer manual). The system software activates this signal only once, at actual system power-up (not on system reset CTRL/ALT/DEL).

4.13.3 Return Printer Status - AH = 2, DL = 0

This function reads the printer status port and returns the information in register AH. This is the same information as that returned after the Output Character to Printer (AH=0, DL=0) function, and the Initialize Printer (AH=1, DL=0) function.

The bits of AH are encoded as shown in Figure 4-8.

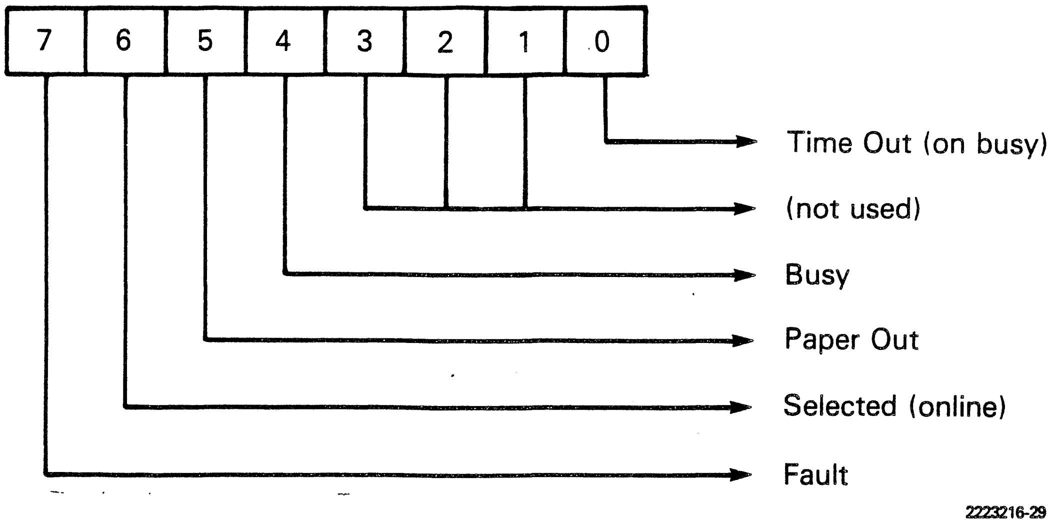


Figure 4-8 Byte Definition - Return Printer Status

4.13.4 Use Under an Operating System

When the software interrupt technique interfaces with ROM routines, a DSR can be enhanced or replaced by patching its interface interrupt vector. Under MS-DOS, for example, the serial printer support emulates the parallel printer functions of the ROM.

The printer interface is implemented by patching a small routine in front of the printer interrupt vector. This routine looks at register DL to determine the desired printer. If DL=0, a jump to the ROM routine is made, and the user is unaware of the patch. If DL=1, AH is decoded to perform the appropriate function on the serial printer. If DL = FFH, then the desired function is performed on the default (currently configured) printer.

Because the serial support emulates the status returned by the parallel routines of the ROM, the user knows of the operation only because he set register DL. Some operating systems do not require that register DL be set. In the case of MS-DOS, however, the DSR is extended in a manner that requires the setting of DL. Refer to the documentation appropriate for the operating system in use.

4.14 WINCHESTER ROM

The Winchester ROM, on the Winchester controller board, interfaces with the system ROM software, specifically the system disk DSR. The Winchester ROM is addressed by the system processor. Its address, as determined by the hardware, is 0F8000H. The convention locates the ROM at the address (as seen by the software) of 0F400:4000H.

In addition to the disk DSR software, the Winchester ROM contains the software necessary to drive the Winchester controller, to boot up the system from the Winchester disk, to format the disk, and to run diagnostics (both power-up and advanced) on the controller and disk.

After initialization, all regular operations of the Winchester ROM (read, write, verify, and so on) are done through the disk DSR. (See subsection 4.11.)

4.14.1 Limitations

The DSR and other utilities provided by the system ROM limit the types of Winchester drives that can be used by the system. The limits are as follows:

* X x Y cylinders per drive where $1 < X < 256$ and $1 < Y < 15$
* 16 surfaces per drive
* 17 sectors per track
* 512 bytes per sector
* 255 error retries
* 11-bit error-burst length

Most of the routines within the ROM are driven by data structures that describe the type of drive. The system is powered up assuming the following drive parameters:

153 cylinders
4 surfaces
125 first track of reduced write current
64 first track of write precompensation
1 error retry
11-bit error-burst length
3-ms step option

If the default parameters are not correct for the type of drive in use, an Initialize Winchester Disk System option call must be made to install the correct parameters. The system can boot the first sector with the default parameters.

4.14.2 System Interface

The Winchester controller board ROM is initialized to the system when it is called by the system ROM following the power-up self-test. The system ROM tests the Winchester disk controller ROM to make certain the controller is functioning properly before calling it. To allow the system ROM to test and call it, the Winchester disk controller ROM contains a header defining the ROM size, the entry point of the ROM, a version number for the ROM, and an identification message preceded by the message length.

The entry point called by the system ROM is required to do any device- dependent initialization and, optionally, to boot the system from the device that the called ROM serves. For the Winchester disk, the operations are as follows:

* Set the RAM area of the ROM in the system. Set the device-installed bit in the system configuration word. This second step permits the system unit to "sense" that the controller is installed, and, under the diagnostics diskette Display System Configuration test, to display all options installed in the system unit.

* If the caller has passed the "do not boot flag" (0FFFFH in register DX), return control to the caller. Otherwise (with 0 in register DX), the initialization sequence continues.

* If the user has pressed the ESC key, control returns to the system ROM and the system boots from the diskette.

* Otherwise, display the Winchester disk controller ROM sign-on message and execute the controller's power-up tests.

* Test all ROMs that have a lower priority than the Winchester disk controller ROM and then call them. The "do not boot" flag (DX = 0FFFFH) must be set so that the ROM can do any required initialization of associated hardware.

* Read in the boot sector from the disk, check it for usability, and jump to the code in the boot sector.

* If any errors occur in the above area, control is returned to the system ROM.

4.14.3 System RAM Usage

The Winchester disk ROM uses 30 bytes of RAM in the system RAM area. This RAM is allocated as a contiguous block of memory only after previously called ROMs have been allocated their RAM space. This RAM block is pointed to by a word in the system vector area. The data structure of this vector area is given in Table 4-12.

Table 4-12 RAM Segment Pointers

Address	User	Value	Address
0000:0180	System ROM U63	RAM segment address for ROM	F400:A000
0000:0182	System ROM U63	Length of RAM segment in bytes	
0000:0184	F400:0000 ROM	RAM segment address for ROM	F400:0000
0000:0186	F400:0000 ROM	Length of RAM segment in bytes	
0000:0188	F400:2000 ROM	RAM segment address for ROM	F400:2000
0000:018A	F400:2000 ROM	Length of RAM segment in bytes	
0000:018C	Windisk ROM	RAM segment address for ROM	F400:4000
0000:018E	Windisk ROM	Length of RAM segment in bytes	(30H)
0000:0184	F400:6000 ROM	RAM segment address for ROM	F400:6000
0000:0186	F400:6000 ROM	Length of RAM segment in bytes	
0000:0184	Option ROM U62	RAM segment address for ROM	F400:8000
0000:0186	Option ROM U62	Length of RAM segment in bytes	

All accesses to the Winchester disk controller RAM area are through the segment pointer at 0000:018CH. Because the Winchester disk controller ROM is located at segment 0F400H, the segment pointer location can also be reached from the code segment at address 0F400:C18CH.

The segment pointer allows the Winchester disk controller RAM area to be located anywhere, but care must be taken if the area is moved after the system is initialized. If this is done, the Winchester disk system must be reinitialized with the Winchester disk option call "0" (Initialize System) after the RAM area is moved and the vectors are set to the new values. To do this, pass the new segment address in DS and 000CH as the pointer to the initialization data. (See paragraph 4.14.18.1.)

4.14.4 Power-up Testing

To determine that the Winchester disk controller is working properly, it is tested by its own internal diagnostics and the RAM diagnostics. Failures are reported as system errors 11xx, where xx indicates the error received. If an error occurs, control is returned to the system ROM.

4.14.5 Booting from the Winchester

After the power-up testing of the controller completes, the Winchester goes through the boot sequence. Only drive 4 (E: for MS-DOS) can be booted. If drive 5 is connected to the controller, it can be used for data only.

First, the boot procedure polls the drive for the ready condition. If the drive is not ready (as would be true after the power is turned on), the ROM routines wait approximately 30 seconds for the ready condition. If the user presses the ESC key at any time during this wait, control is returned to the system ROM, and the diskette drive conducts the initialization boot.

4.14.6 Error Recovery

The error recovery procedures depend on the error. For hardware controller errors (time-outs), the controller is reset, and no retries are attempted. A hardware error code is returned from the disk DSR.

For disk drive errors (seek incomplete, write fault, and so on), no retries are reported, and the disk DSR returns the hardware error code.

Read Data operations have two types of errors: correctable and uncorrectable. If the data is correctable, it is corrected, and no error is reported directly. A DSR Read Soft Retry Status reports this error.

For uncorrectable errors, a "restore" is done before each retry. If the retry does not succeed, the data buffer is filled; with CCH when the data cannot be read at all, or with the uncorrected data if the data can be read but contains an ECC error.

For other operation errors, a "restore" is placed before each retry.

4.14.7 Error Reporting

The disk DSR is capable of reporting only a few errors. The power-up boot can report more but not all. Table 4-13 is a listing of errors reported by the disk controller and the codes reported by the DSR.

Table 4-13 Winchester DSR Error Codes

Reported Error		Controller Error	
20H	Hardware failure	01H	No index detected
20H	Hardware failure	02H	No seek complete
20H	Hardware failure	03H	Write fault
20H	Hardware failure	04H	DRIVE NOT READY during operation
20H	Hardware failure	06H	Track 00 not found
10H	CRC error	10H	ID field read error
10H	CRC error	11H	Uncorrectable data error
02H	Disk format error	12H	Address mark not found
04H	Record not found	14H	Record not found
40H	Seek error	15H	Seek error
00H	No error (on RETURN)	18H	Correctable data error
10H	CRC error (soft stat)	18H	Correctable data error
01H	Command error	19H	Bad track flag detected
02H	Disk format error	1AH	Format error
01H	Command error	1CH	Illegal access to alternate track
01H	Command error*	1DH	Illegal alternate track for format
02H	Disk format error	1EH	Expected alternate track, isn't
01H	Command error*	1FH	Alternate track = bad track
01H	Command error*	20H	Invalid command
01H	Command error*	21H	Illegal disk address
20H	Hardware failure*	30H	RAM diagnostic failure
20H	Hardware failure*	31H	Program memory checksum error
20H	Hardware failure*	32H	ECC diagnostic failure

* This error should never be encountered by the DSR.

The errors that can be reported during boot are the controller errors given in Table 4-13 and Table 4-14.

Table 4-14 Displayed Error Codes

All errors have the following message displayed:

** SYSTEM ERROR - 11xx **

Where xx = the extended error

Extended Error	Explanation
33H	Status error on REQUEST SENSE STATUS command
40H	Time-out while waiting for WRITE DATA mode
41H	READ MODE while waiting for WRITE DATA mode
42H	COMMAND MODE while waiting for WRITE DATA mode
43H	STATUS MODE while waiting for WRITE DATA mode
44H	WRITE MODE while waiting for READ DATA mode
45H	Time-out while waiting for READ DATA mode
46H	COMMAND MODE while waiting for READ DATA mode
47H	STATUS MODE while waiting for READ DATA mode
48H	WRITE MODE while waiting for COMMAND mode
49H	READ MODE while waiting for COMMAND mode
4AH	Time-out while waiting for COMMAND mode
4BH	STATUS MODE while waiting for COMMAND mode
4CH	WRITE MODE while waiting for STATUS mode
4DH	READ MODE while waiting for STATUS mode
4EH	COMMAND MODE while waiting for STATUS mode
4FH	Time-out while waiting for STATUS mode
51H	Disk not ready
52H	CRC error
53H	Seek error
54H	Sector-not-found error
55H	Disk (unknown) error (controller failure)
56H	Not a TI-system disk
57H	Disk format error
58H	Bad boot sector CRC or bad controller
59H	System ROM version doesn't support Winchester

4.14.8 Hardware Interface Routines

This interface to the Winchester disk system implements additional functions in a straightforward way. The calls provide a method of interfacing with the hardware that is almost hardware-independent.

To use this interface, do a long call through the first doubleword in the RAM area of the Winchester disk controller ROM. Place the opcode for the operation in register AH. Other register usages are explained with each operation.

For more information, refer to paragraph 4.4.4 and to the table in paragraph 4.5.2.

The programming steps required to do the long call are given below.

```
  WINROM DD    00000000                       ;LOCAL PLACE TO STORE VECTOR
                                              ;TO ROM.
        ...

  ;  The next steps get the entry vector for the Winchester ROM
  ;  code from the ROM data area and put it into local storage


        PUSH ES                               ;SAVE ES
        XOR  AX,AX                            ;SET ES TO 0000H
        MOV  ES,AX
        MOV  ES,ES:WORD PTR 18CH              ;GET WINCH RAM SEGMENT INTO ES
        LES  AX,ES:DWORD PTR 0000             ;GET VECTOR FOR WINCH ROM
        MOV  WORD PTR WINROM+2,ES             ;SAVE IN OUR DATA AREA
        MOV  WORD PTR WINROM,AX               ;
        POP  ES                               ;RESTORE ES

        .....

  ;  The following steps access the Winchester ROM functions
  ;  after the above initialization is completed

        MOV  AH,OPCODE                        ;SET OPCODE INTO AH
        CALL WINROM                           ;GO DO THE OPERATION
```

The following paragraphs explain the operations available from this entry point.

4.14.8.1 Initialize Winchester Disk System.

Opcode: AH = 00H
Entry: DS:SI = POINTER TO DATA BLOCK

Offset	Value/Use
00H	(Word) Sector size in bytes
02H	(Byte) Track size in sectors
03H	(Byte) Number of surfaces
04H	(Byte) Number of cylinders on disk
05H	(Byte) Number of error retries
06H	(Word) Reduced write current cylinder
08H	(Word) Write precomp start cylinder
0AH	(Byte) Step option
0BH	(Byte) Error-burst corrected length

Exit: AL = Error code
Used: AX, BX

This operation tells the disk subsystem the type of Winchester drive being used. It sets the hardware and software data structures so that a user can simply call the DSR to use the drive.

4.14.8.2 Check Winchester ROM Version.

Opcode: AH = 01H
Entry: None
Exit: AX = BCD ROM version number
Used: AX

Example: If ROM is V1.23, then AX returns 0123H

This operation returns the Winchester ROM version number. This is often useful for software-compatibility checks.

4.14.8.3 Request Controller Error Sense.

Opcode:	AH = 02H
Entry:	DS:SI = Address of 6-byte data block
Exit:	AL = Error code Z = Set if no error Data block contains what controller returned.
Used:	AX,CX,SI,DI

This operation gets error information from the controller and returns an error code. If the controller hardware is broken, appropriate error codes are returned.

4.14.8.4 Send Winchester Controller Command.

Opcode:	AH = 03H
Entry:	DS:SI = Address of 6-byte data block containing command and other data (see hardware spec)
Exit:	AL = Error code if Carry flag is set Z = Set, C = Reset if no error Z = Set, C = Set if time-out Z = Reset, C = Set if improper controller mode
Used:	AX,CX,SI

This operation sends a command to the controller. It does not wait for a response.

4.14.8.5 Get Data From the Winchester Controller.

Opcode:	AH = 04H
Entry:	ES:DI = Address of buffer to receive data CX = Number of bytes of data to get
Exit:	AL = Error code if Carry flag is set Z = Set, C = Reset if no error Z = Set, C = Set if time-out Z = Reset, C = Set if improper controller mode
Used:	AX,CX,DI

This operation waits for the controller to provide data and then puts it into the user's buffer. The operation waits about 1 second before returning a time-out error. If the controller is in the command state or the status state, an appropriate error code is returned.

4.14.8.6 Write Data to the Winchester Controller.

Opcode:	AH = 05H
Entry:	ES:DI = Address of data buffer to transmit
	CX = Number of bytes of data to put
Exit:	AL = Error code if Carry flag is set
	Z = Set, C = Reset if no error
	Z = Set, C = Set if time-out
	Z = Reset, C = Set if improper controller mode
Used:	AX,CX,DI

This operation waits for the controller to ask for data and then writes from the user's buffer to the controller. The operation waits about 1 second before returning a time-out error. If the controller is in the command state or the status state, an appropriate error code is returned.

4.14.8.7 Get Status From Winchester Controller.

Opcode:	AH = 06H
Entry:	None
Exit:	AL = Error code if Carry flag is set
	Z = Set, C = Reset if no error
	Z = Set, C = Set if time-out
	Z = Reset, C = Set if controller mode is not status
	Z = Reset, C = Reset if status indicates controller has an error
Used:	AX,CX

This operation waits for the status return from the controller. The operation waits about 1 second before returning a time-out error. If the controller is in the command state or the data-transfer state, an appropriate error code is returned.

4.14.8.8 Get and Compare Data From the Winchester Controller.

Opcode:	AH = 07H
Entry:	ES:DI = Address of buffer to receive data
	CX = Number of bytes of data to get
Exit:	AL = Error code if C flag is set
	Z = Set, C = Reset if no error
	Z = Set, C = Set if time-out
	Z = Reset, C = Set if improper controller mode
	Z = Reset, C = Reset if data does not compare; if no compare, DI to the miscompared data
Used:	AX,CX,DI

This operation waits for the controller to provide data and then compares it with the data in the user's buffer. If the data does not compare, the data pointer (DS:DI) is set to point at the data address that does not compare. After a wait of about 1 s, the controller returns a time-out error. If the controller is in the command state or the status state, an appropriate error code is returned.

4.14.8.9 Enable Data and Status Interrupt From Controller.

Opcode:	AH = 08H
Entry:	None
Exit:	None
Used:	AX

This operation enables the Winchester controller interrupts to the system bus. However, this operation does not enable the system interrupts from the interrupt controller or from the processor interrupt.

4.14.8.10 Enable Status Interrupt From Controller.

Opcode:	AH = 09H
Entry:	None
Exity:	None
Used:	AX

This operation enables the Winchester controller interrupts to the system bus. However, this operation does not enable the system interrupts from the interrupt controller or from the processor interrupt.

4.14.8.11 Disable Data and Status Interrupt From Controller.

```
Opcode:   AH = 0AH
Entry:    None
Exity:    None
Used:     AX
```

This operation disables the Winchester controller interrupts to the system bus. However, this operation does not disable the system interrupts from the interrupt controller or from the processor interrupt.

4.14.8.12 Poll for Controller Request.

```
Opcode:   AH = 0BH
Entry:    None
Exit:     Z = Set if request is not active
          Z = Reset if request is active
Used:     AX
```

This operation determines when the controller is ready for command, status, data in, or data out.

4.14.8.13 Format a Track.

```
Opcode:   AH = 0CH
Entry:    DL = Drive number (4,5)
          DH = Interleave factor
          CX = Logical track number to format
     The drive parameters must have been set using operation 0.
Exit:     AL = Error code, 0 if OK
          CX = Track number of error, if there is an error
Used:     AX,BX,CX,DX,SI,DI
```

This operation formats a track on the Winchester disk. The drive parameters must be set up by a call to operation 0. Multiplying the cylinder number by the number of surfaces, then adding in the surface number yields the logical track number. The interleave factor is typically 12 or 13 for optimum use of the DSR in reading sequential sectors. The error code returned is the controller error code with extentions for such conditions as time-outs. This operation always does a RESTORE operation before the track format, so it is slow to format a disk.

4.14.8.14 Format an Alternate Track.

```
Opcode:    AH = 0DH
Entry:     DL = Drive number (4,5)
           DH = Interleave factor
           CX = Logical track number to format
           BX = Logical track number of alternate
     The drive parameters must have been set using operation 0.
Exit:      AL = Error code, 0 if OK
           CX = Track number of error, if there is an error
Used:      AX,BX,CX,DX,SI,DI
```

Formatting routines use this operation to map a bad track to an alternate track. The drive parameters must be set up by a call to operation 0. Multiplying the cylinder number by the number of surfaces, then adding the surface number yields the logical track number. The interleave factor is typically 12 or 13 for optimum use of the DSR in reading sequential sectors. The error code returned is the controller error code with extensions for such conditions as time-outs.

4.14.8.15 Format a Track as Bad.

```
Opcode:    AH = 0EH
Entry:     DL = Drive number (4,5)
           DH = Interleave factor
           CX = Logical track number to format
     The drive parameters must have been set using operation 0.
Exit:      AL = Error code, 0 if OK
           CX = Track number of error, if there is an error
Used:      AX,BX,CX,DX,SI,DI
```

This operation formats a defective track so that read operations do not miss the defect. The drive parameters must be set up by a call to operation 0. Multiplying the cylinder number by the number of surfaces, then adding the surface number yields the logical track number. The factor is typically 12 or 13 for optimum use of the DSR in reading sequential sectors. The error code returned is the controller error code with extentions for such conditions as time-outs. This operation always does a RESTORE operation before the track format.

4.14.8.16 Check the Track Format.

```
Opcode:    AH = 0FH
Entry:     DL = Drive number (4,5)
           DH = Interleave factor
           CX = Logical track number to check
     The drive parameters must have been set using operation 0.
Exit:      AL = Error code, 0 if OK
           CX = Track number of error, if there is an error
Used:      AX,BX,CX,DX,SI,DI
```

This operation checks a track for proper format. This routine does not report errors for tracks that have been formatted as bad tracks or alternate tracks unless the ID fields are incorrect. The drive parameters must be set up by a call to operation 0. Multiplying the cylinder number by the number of surfaces, then adding the surface number, yields the logical track number. The interleave factor is typically 12 or 13 for optimum use of the DSR in reading sequential sectors. The error code returned is the controller error code with extentions for such conditions as time-outs.

4.14.8.17 Format a Winchester Drive.

```
Opcode:    AH = 10H
Entry:     DL = Drive number (4,5)
           DH = Interleave factor
           CX = Logical track number to begin format
     The drive parameters must have been set using operation 0.
Exit:      AL = Error code, 0 if OK
           CX = Track number of error, if there is an error
Used:      AX,BX,CX,DX,SI,DI
```

This operation formats a Winchester drive. The drive parameters must be set by a call to operation 0. Multiplying the cylinder number by the number of surfaces, then adding the surface number, yields the logical track number. The interleave factor is typically 12 or 13 for optimum use of the DSR in reading sequential sectors. The error code returned is the controller error code with extentions for such conditions as time-outs. If an error occurs during the drive formatting operation, register CX returns the track in error. If the formatting operation must be completed, increment the track number and call the routine again. This could be necessary, for instance, if a drive defect falls directly on an address mark or ID field.

Appendix A

SYSTEM I/O MAP

Table A-1 System I/O Map

Address	Device	Bit/Use
Motherboard:		
00000	U47 Latch	0 Speaker timer enable
		1 Timer 1 interrupt enable
		2 Timer 2 interrupt enable
		3 Single-density (FM) enable
		4 Track greater than 1/2 (TG43)
		5 Diskette side one enable (FSID-)
		6 Diskette mode control (M1)
		7 Diskette mode control (M0)
00001	U48 Input buffer	0 Option jumper E1-E2
		1 Option jumper E3-E4
		2 Option jumper E5-E6
		3 Parity interrupt pending
		4 Printer port BUSY
		5 Printer port paper out
		6 Printer port printer selected
		7 Printer port NO fault
00002	U49 Latch	0-7 Printer port data outputs
00003	U50 Latch	0 LED 1 OFF
		1 LED 2 OFF
		2 LED 3 OFF
		4 Parity interrupt enable
		5 Printer port not auto feed
		6 Printer port not strobe
		7 Printer port not initialized

Table A-1 System I/O Map (Continued)

Address	Device	Bit/Use
Motherboard(Continued):		
00004	U51 Latch	0 Diskette Drive SELECT 1 1 Diskette Drive SELECT 2 2 Diskette Drive SELECT 3 3 Diskette Drive SELECT 4 4 Diskette Drive MOTOR 1 5 Diskette Drive MOTOR 2 6 Diskette Drive MOTOR 3 7 Diskette Drive MOTOR 4
00005-0000F	Reserved	
00010	U44 8251 USART	Data Register
00011	U44 8251 USART	Control Register
00012-00013	Reserved	
00014	U45 8253 Timer	Counter 0
00015	U45 8253 Timer	Counter 1
00016	U45 8253 Timer	Counter 2
00017	U45 8253 Timer	Control register
00018	U46 8259A Interrupt controller	
00019	U46 8259A Interrupt controller	
00020	FDC Command register or RAM	
00021	FDC Track register	
00022	FDC Sector register or RAM reset	
00023	FDC Data register	
00024-0002F		Reserved
Winchester Controller Board:		
00030	Winchester I/O port	Input: 0-7 Don't care. Data is held for each handshake cycle. Output: 0-7 Don't care. Data is latched til updated.

Table A-1 System I/O Map (Continued)

Address	Device	Bit/Use
00031	Winchester reset register	
		Read:
		0 Data request
		1 Input/Output
		2 Command/Data
		3 Interrupt pending (Level 6)
		Write:
		0-7 Don't care (Any write will do a RESET)
Winchester Controller Board (Continued):		
00032		Not used
00033		Interrupt Mask
		0 Status interrupt enable
		1 Data interrupt disable
Future Options:		
00034-0003B		Reserved
0003C-0003F		Local Area Net I/O
00040-000BF		Reserved

Table A-1 System I/O Map (Continued)

Address	Device	Bit/Use
Clock and Analog Interface:		
000C0	Clock/Analog Interface	
		0 End of conversion (EOC)(Active HIGH)
		1 Not used (tied LOW)
		2 Lightpen interrupt latch ON
		3 Battery low
		4 Switch 4
		5 Switch 3
		6 Switch 2
		7 Switch 1
000C1		Do not allow light pen interrupt (tri-state signal)
000C2		Allow light pen interrupt (Pass interrupt signal)
000C8		Joystick port X1 (Current sense)
000C9		Joystick port Y1 (Current sense)
000CA		Joystick port X2 (Current sense)
000CB		Joystick port Y2 (Current sense)
000CC		Analog input 4 (SW4) (Voltage sense)
000CD		Analog input 3 (SW3) (Voltage sense)
000CE		Analog input 2 (SW2) (Voltage sense)
000CF		Analog input 1 (SW1) (Voltage sense)
000D0		Clock Control
		0 Address Bit 0 MSM5832 clock
		1 Address Bit 1 MSM5832 clock
		2 Address Bit 2 MSM5832 clock
		3 Address Bit 3 MSM5832 clock
		4 HOLD
		5 WRITE
		6 READ
		7 + or - 30 sec adjust
000D1-000D7		Reserved
000D8		Clock data (low nibble only)
000D9-000DF		Reserved

Table A-1 System I/O Map (Concluded)

Address	Device	Bit/Use
Sync-Async Comm Board:		
000E0-000E3	COMM Port 1 IR1	Interrupt Acknowledge
000E4		CHB command
000E5		CHB data
000E6		CHA command
000E7		CHA data
000E8-000EB	COMM Port 2 IR2	Interrupt Acknowledge
000EC		CHB command
000ED		CHB data
000EE		CHA command
000EF		CHA data
000F0-000F3	COMM Port 3 IR3	Interrupt Acknowledge
000F4		CHB command
000F5		CHB data
000F6		CHA command
000F7		CHA data
000F8-000FB	COMM Port 4 IR4	Interrupt Acknowledge
000FC		CHB Command
000FD		CHB Data
000FE		CHA Command
000FF		CHA Data
00100-003FF		Available for future products

Appendix B

SYSTEM MEMORY MAP

Table B-1 System Memory Map

Address	Devices
Dynamic RAM:	
00000-0FFFF	64-kbytes motherboard RAM
10000-1FFFF	64-kbytes expansion RAM board Bank 1
20000-2FFFF	64-kbytes expansion RAM board Bank 2
30000-3FFFF	64-kbytes expansion RAM board Bank 3
40000-BFFFF	Expansion bus memory
CRT Controller:	
C0000-C7FFF	Graphics RAM Bank A
C8000-CFFFF	Graphics RAM Bank B
D0000-D7FFF	Graphics RAM Bank C
D8000-DDFFF	Reserved
DE000-DE7FF	Active character memory
DE800-DEFFF	Phantom character memory
DF000	
	Bit 0 Miscellaneous input buffer, BLUE feedback, read only
	Bit 1 Miscellaneous input buffer, RED feedback, read only
	Bit 2 Miscellaneous input buffer, GREEN feedback, read only
	Bit 3 Miscellaneous input buffer, interrupt pending, read only

Table B-1. System Memory Map, Concluded

Address	Devices
DF001-DF00F	Miscellaneous input buffer
DF010-DF01F	Graphics RED palette latch, write only
DF020-DF02F	Graphics GRN palette latch, write only
DF030-DF03F	Graphics BLU palette latch, write only
DF040-DF7FF	Reserved
DF800-DF80F	Attribute latch
DF810	CRT controller address register, write only
DF811	CRT Controller status register, read only
DF812	CRT Controller address register, write only
DF813	CRT Controller address register, write only
DF814-DF81F	Reserved
DF820	
	Bit 7 Miscellaneous output latch, interrupt enable
	Bit 6 Miscellaneous output latch, alphanumerics screen enable
Other Peripherals:	
DF821-DFFFF	Reserved
E0000-E7FFF	Reserved for speech storage RAM
E8000-F3FFF	Reserved
ROM Usage:	
F4000-F5FFF	8K ROM space(Clock/Analog Interface)
F6000-F7FFF	8K ROM space(Local Area Net Option Board)
F8000-F9FFF	8K ROM space(Winchester Controller)
FA000-FBFFF	8K ROM space(Reserved)
FC000-FDFFF	8K ROM space, 1 wait state (XU62) (motherboard)
FE000-FFFFF	8K system ROM, 1 wait state (U63) (motherboard)

Appendix C

CHARACTER SET

Table C-1 ASCII Control Characters

From USA Standards Institute Publication X3.4-1968

ACK	acknowledge	FF	form feed
BEL	bell	FS	file separator
BS	backspace	GS	group separator
CAN	cancel	HT	horizontal tabulation
CR	carriage return	LF	line feed
DC1	device control 1	NAK	negative acknowledge
DC2	device control 2	NUL	null
DC3	device control 3	RS	record separator
DC4	device control 4	SI	shift in
*DEL	delete	SO	shift out
DLE	data link escape	SOH	start of heading
EM	end of medium	STX	start of text
ENQ	enquiry	SUB	substitute
EOT	end of transmission	SYN	synchronous idle
ESC	escape	US	unit separator
ETB	end of transmission block	VT	vertical tabulation
ETX	end of text		

* Not strictly a control character

Table C-2. Numeric Cross Reference for Character Sets

Decimal	Hexadecimal	Keystroke(s)	ASCII Character	Displayed Character	Comments
0	00	CTRL 2	NUL		
1	01	CTRL A	SOH	☺	
2	02	CTRL B	STX	☻	
3	03	CTRL C	ETX	♥	
4	04	CTRL D	EOT	♦	
5	05	CTRL E	ENQ	♣	
6	06	CTRL F	ACK	♠	
7	07	CTRL G	BEL	•	
8	08	CTRL H, BACKSPACE, SHIFT, BACKSPACE	BS	◘	
9	09	CTRL I	HT	◇	
10	0A	CTRL RETURN, CTRL J, LINE FEED	LF	◙	
11	0B	CTRL K	VT	♂	
12	0C	CTRL L	FF	♀	
13	0D	CTRL M, RETURN, SHIFT RETURN	CR	♪	
14	OE	CTRL N	SO	♫	
15	OF	CTRL O	SI	☼	
16	10	CTRL P	DLE	►	
17	11	CTRL Q	DC1	◄	
18	12	CTRL R	DC2	↕	
19	13	CTRL S	DC3	‼	
20	14	CTRL T	DC4	¶	
21	15	CTRL U	NAK	§	

Table C-2. Numeric Cross-Reference for Character Sets (Continued)

Decimal	Hexadecimal	Keystroke(s)	ASCII Character	Displayed Character	Comments
22	16	CTRL V	SYN	▬	
23	17	CTRL W	ETB	↨	
24	18	CTRL X	CAN	↑	
25	19	CTRL Y	EM	↓	
26	1A	CTRL Z	SUB	→	
27	1B	CTRL [, ESC, SHIFT ESC, CTRL ESC	ESC	←	
28	1C	CTRL \	FS	∟	
29	1D	CTRL]	GS	↔	
30	1E	CTRL 6	RS	▲	
31	1F	CTRL –	US	▼	
32	20	CTRL SPACE, SPACE BAR, ALT SPACE, SHIFT SPACE	SP		Blank space
33	21	!	!	!	Exclamation point
34	22	"	"	"	Quotation marks
35	23	#	#	#	Number, Pound
36	24	$	$	$	Dollar sign
37	25	%	%	%	Percent sign
38	26	&	&	&	Ampersand
39	27	'	'	'	Apostrophe
40	28	(	(	(	Open parenthesis
41	29	)	)	)	Close parenthesis
42	2A	*	*	*	Asterisk
43	2B	+	+	+	Plus

Table C-2. Numeric Cross-Reference for Character Sets (Continued)

Decimal	Hexadecimal	Keystroke(s)	ASCII Character	Displayed Character	Comments
44	2C	,	,	,	Comma
45	2D	-	-	-	Minus, Hyphen
46	2E	.	.	.	Period, Decimal point
47	2F	/	/	/	Slash, Virgule
48	30	0	0	0	Zero
49	31	1	1	1	One
50	32	2	2	2	Two
51	33	3	3	3	Three
52	34	4	4	4	Four
53	35	5	5	5	Five
54	36	6	6	6	Six
55	37	7	7	7	Seven
56	38	8	8	8	Eight
57	39	9	9	9	Nine
58	3A	:	:	:	Colon
59	3B	;	;	;	Semicolon
60	3C	<	<	<	Less than
61	3D	=	=	=	Equals sign
62	3E	>	>	>	Greater than
63	3F	?	?	?	Question mark
64	40	@	@	@	Commercial "at"
65	41	A	A	A	A (uppercase)
66	42	B	B	B	B (uppercase)
67	43	C	C	C	C (uppercase)
68	44	D	D	D	D (uppercase)

Table C-2. Numeric Cross-Reference for Character Sets (Continued)

Decimal	Hexadecimal	Keystroke(s)	ASCII Character	Displayed Character	Comments
69	45	E	E	E	E (uppercase)
70	46	F	F	F	F (uppercase)
71	47	G	G	G	G (uppercase)
72	48	H	H	H	H (uppercase)
73	49	I	I	I	I (uppercase)
74	4A	J	J	J	J (uppercase)
75	4B	K	K	K	K (uppercase)
76	4C	L	L	L	L (uppercase)
77	4D	M	M	M	M (uppercase)
78	4E	N	N	N	N (uppercase)
79	4F	O	O	O	O (uppercase)
80	50	P	P	P	P (uppercase)
81	51	Q	Q	Q	Q (uppercase)
82	52	R	R	R	R (uppercase)
83	53	S	S	S	S (uppercase)
84	54	T	T	T	T (uppercase)
85	55	U	U	U	U (uppercase)
86	56	V	V	V	V (uppercase)
87	57	W	W	W	W (uppercase)
88	58	X	X	X	X (uppercase)
89	59	Y	Y	Y	Y (uppercase)
90	5A	Z	Z	Z	Z (uppercase)
91	5B	[	[	[	Open bracket
92	5C	\	\	\	Left slash
93	5D	]	]	]	Close bracket

Table C-2. Numeric Cross-Reference for Character Sets (Continued)

Decimal	Hexadecimal	Keystroke(s)	ASCII Character	Displayed Character	Comments
94	5E	^	^	^	Circumflex
95	5F	_	_	_	Underline
96	60	`	`	`	Graves accent
97	61	a	a	a	a (lowercase)
98	62	b	b	b	b (lowercase)
99	63	c	c	c	c (lowercase)
100	64	d	d	d	d (lowercase)
101	65	e	e	e	e (lowercase)
102	66	f	f	f	f (lowercase)
103	67	g	g	g	g (lowercase)
104	68	h	h	h	h (lowercase)
105	69	i	i	i	i (lowercase)
106	6A	j	j	j	j (lowercase)
107	6B	k	k	k	k (lowercase)
108	6C	l	l	l	l (lowercase)
109	6D	m	m	m	m (lowercase)
110	6E	n	n	n	n (lowercase)
111	6F	o	o	o	o (lowercase)
112	70	p	p	p	p (lowercase)
113	71	q	q	q	q (lowercase)
114	72	r	r	r	r (lowercase)
115	73	s	s	s	s (lowercase)
116	74	t	t	t	t (lowercase)
117	75	u	u	u	u (lowercase)
118	76	v	v	v	v (lowercase)

Table C-2. Numeric Cross Reference for Character Sets (Continued)

Decimal	Hexadecimal	Keystroke(s)	ASCII Character	Displayed Character	Comments
119	77	w	w	w	w (lowercase)
120	78	x	x	x	x (lowercase)
121	79	y	y	y	y (lowercase)
122	7A	z	z	z	z (lowercase)
123	7B	{	{	{	Open brace
124	7C	¦	¦	¦	Vertical rule, Bar
125	7D	}	}	}	Close brace
126	7E	~	~	~	Tilde
127	7F	CTRL—BACKSPACE	DEL		ASCII DEL
128	80	ALT 128		Ç	
129	81	ALT 129		ü	
130	82	ALT 130		é	
131	83	ALT 131		â	
132	84	ALT 132		ä	
133	85	ALT 133		à	
134	86	ALT 134		å	
135	87	ALT 135		ç	
136	88	ALT 136		ê	
137	89	ALT 137		ë	
138	8A	ALT 138		è	
139	8B	ALT 139		ï	
140	8C	ALT 140		î	
141	8D	ALT 141		ì	
142	8E	ALT 142		Ä	

Table C-2. Numeric Cross-Reference for Character Sets (Continued)

Decimal	Hexadecimal	Keystroke(s)	ASCII Character	Displayed Character	Comments
143	8F	ALT 143		Å	
144	90	ALT 144		É	
145	91	ALT 145		æ	
146	92	ALT 146		Æ	
147	93	ALT 147		ô	
148	94	ALT 148		ö	
149	95	ALT 149		ò	
150	96	ALT 150		û	
151	97	ALT 151		ù	
152	98	ALT 152		ÿ	
153	99	ALT 153		Ö	
154	9A	ALT 154		Ü	
155	9B	ALT 155		¢	
156	9C	ALT 156		£	
157	9D	ALT 157		¥	
158	9E	ALT 158		Pt	
159	9F	ALT 159		ƒ	
160	A0	ALT 160		á	
161	A1	ALT 161		í	
162	A2	ALT 162		ó	
163	A3	ALT 163		ú	
164	A4	ALT 164		ñ	
165	A5	ALT 165		Ñ	
166	A6	ALT 166		ª	
167	A7	ALT 167		º	

Table C-2. Numeric Cross-Reference for Character Sets (Continued)

Decimal	Hexadecimal	Keystroke(s)	ASCII Character	Displayed Character	Comments
168	A8	ALT 168		ξ	
169	A9	ALT 169		┌	
170	AA	ALT 170		┐	
171	AB	ALT 171		½	
172	AC	ALT 172		¼	
173	AD	ALT 173		;	
174	AE	ALT 174		≪	
175	AF	ALT 175		≫	
176	B0	ALT 176			
177	B1	ALT 177			
178	B2	ALT 178			
179	B3	ALT 179		I	
180	B4	ALT 180			
181	B5	ALT 181			
182	B6	ALT 182			
183	B7	ALT 183			
184	B8	ALT 184			
185	B9	ALT 185			
186	BA	ALT 186		II	
187	BB	ALT 187			
188	BC	ALT 188			
189	BD	ALT 189			
190	BE	ALT 190			
191	BF	ALT 191			
192	C0	ALT 192		L	

Table C-2. Numeric Cross-Reference for Character Sets (Continued)

Decimal	Hexadecimal	Keystroke(s)	ASCII Character	Displayed Character	Comments
193	C1	ALT 193		⊥	
194	C2	ALT 194		T	
195	C3	ALT 195			
196	C4	ALT 196			
197	C5	ALT 197			
198	C6	ALT 198			
199	C7	ALT 199			
200	C8	ALT 200			
201	C9	ALT 201			
202	CA	ALT 202			
203	CB	ALT 203			
204	CC	ALT 204			
205	CD	ALT 205			
206	CE	ALT 206			
207	CF	ALT 207			
208	D0	ALT 208			
209	D1	ALT 209			
210	D2	ALT 210		π	
211	D3	ALT 211			
212	D4	ALT 212			
213	D5	ALT 213			
214	D6	ALT 214			
215	D7	ALT 215			
216	D8	ALT 216			
217	D9	ALT 217			

Table C-2. Numeric Cross-Reference for Character Sets (Continued)

Decimal	Hexadecimal	Keystroke(s)	ASCII Character	Displayed Character	Comments
218	DA	ALT 218			
219	DB	ALT 219			
220	DC	ALT 220			
221	DD	ALT 221			
222	DE	ALT 222			
223	DF	ALT 223			
224	E0	ALT 224		α	
225	E1	ALT 225		β	
226	E2	ALT 226		Γ	
227	E3	ALT 227		π	
228	E4	ALT 228		Σ	
229	E5	ALT 229		σ	
230	E6	ALT 230		μ	
231	E7	ALT 231		τ	
232	E8	ALT 232		Φ	
233	E9	ALT 233		θ	
234	EA	ALT 234		Ω	
235	EB	ALT 235		δ	
236	EC	ALT 236		∞	
237	ED	ALT 237		$\emptyset$	
238	EE	ALT 238		ϵ	
239	EF	ALT 239		$\cap$	
240	F0	ALT 240		$\equiv$	
241	F1	ALT 241		$\pm$	
242	F2	ALT 242		$\geqslant$	

Table C-2. Numeric Cross-Reference for Character Set (Concluded)

Decimal	Hexadecimal	Keystroke(s)	ASCII Character	Displayed Character	Comments
243	F3	ALT 243		≤	
244	F4	ALT 244		⌠	
245	F5	ALT 245		⌡	
246	F6	ALT 246		÷	
247	F7	ALT 247		≈	
248	F8	ALT 248		°	
249	F9	ALT 249		●	
250	FA	ALT 250		•	
251	FB	ALT 251		√	
252	FC	ALT 252		η	
253	FD	ALT 253		z	
254	FE	ALT 254		■	
255	FF	ALT 255			

Appendix D

CURRENT REQUIREMENTS

This appendix contains information on the current allocations for the Texas Instruments Professional Computer. Current requirements for the options and the printed wiring boards are listed below.

Total current available:

* 5 Volt line 10.0 A
* 12 Volt line 4.5 A
* -5 Volt line 0.5 A

Table D-1 Current Allocations

Device Name	5 Volt Line	12 Volt Line	-12 Volt Line
Motherboard	1.9	0.1	0.0
CRT Controller	1.3	0.0	0.0
RAM Expansion	0.2	0.0	0.0
Graphics	0.8	0.0	0.0
Diskette Drive	1.0	1.2	0.0
Winchester Drive	1.1	1.8	0.0
Winchester Controller	1.7	0.0	0.0
Communications	0.2	0.1	0.1
Modem	0.0	0.1	0.1
Speech	1.9	0.1	0.1

Appendix E

ASYNCHRONOUS COMMUNICATIONS SAMPLE PROGRAM

Control and Status signals

Listed below are the RS232-C control and status signals, with the corresponding 8530 functions used to control and monitor them. This table is a summary of information available from the sync-async comm board schematic.

Table E-1 RS232-C Control and Status signals

RS232-C Signal	Pin Number	8530 Function	Accessed through
Data Terminal Ready (DTR)	20	DTRA	Channel A, WR5, Bit 7
Request-to-Send (RTS)	4	RTSA	Channel A, WR5, Bit 1
Data Set Ready (DSR)	6	DCDB	Channel B, RR0, Bit 3
Data Carrier Detect (DCD)	8	DCDA	Channel A, RR0, Bit 3
Clear-To-Send (CTS)	5	CTSA	Channel A, RR0, Bit 5
Ring Indicator (RI)	22	CTSB	Channel B, RR0, Bit 5
Speed Selector (CH)	11, 23	DTRB	Channel B, WR5, Bit 7
Speed Indicator (CI)	12	SYNCB	Channel B, RR0, Bit 4

```
$ERRORPRINT
$XREF
;****************************************************************************
;
; TITLE     - COMMEX - Example of Async communications
;
; COMPUTER - 8088 ASSEMBLY LANGUAGE
;
; ABSTRACT - This a sample program showing typical initialization
;            of the TIPC communications board in asynchronous, polled mode.
;
;****************************************************************************
 NAME COMMEX
$TITLE(COMMEX - ASYNC COMMUNICATIONS EXAMPLE)
$EJECT
;****************************************************************************
;   PUBLIC DEFINITIONS
;****************************************************************************
 PUBLIC COMMEX
$EJECT
;****************************************************************************
;   LOCAL CONSTANTS
;****************************************************************************
P1CMDA  EQU 0E6H  ; PORT 1, CHANNEL A COMMAND ADDRESS.
P1CMDB  EQU 0E4H  ; PORT 1, CHANNEL B COMMAND ADDRESS.
$EJECT
BIOCODE SEGMENT BYTE PUBLIC
 ASSUME CS:BIOCODE,DS:BIOCODE
;****************************************************************************
; 8530 Initialization Routine
;
;       This routine initializes Port 1 according to a table of initialization
;       parameters stored in PARMST.  PARMST contains an image of the contents
;       of the various 8530 registers.  The contents of each register is pre-
;       ceeded by the number of the register itself.  This number is used to
;       select the appropriate register on the 8530.
;
; This initialization programs the port for asynchronous, polled
;       operations where all interrupts from channel A (i.e., receive,
;       transmit and external status interrupts) and channel B (i.e., external
;       status interrupts) are disabled.  The software is to poll read
;       register RR0 in channel A to determine when data has been received
;       and whether transmission of data has completed.
;
;****************************************************************************
COMMEX PROC NEAR
;
;       First, the 8530 channel A is initialized.
;
 MOV SI,OFFSET PARMTA ;  SI=Address of Chn A parm table.
 MOV DX,0E6H    ;  DX=Port 1,Channel A Command address.
 MOV CX,PARMAS  ;  CX=Parameter table size.
INIA: LODS DS:BYTE PTR[SI]  ;  Get byte from parameter table.
```

```
 OUT DX,AL    ;  Write it to 8530 until
 LOOP INIA    ;  all registers are programmed.
;
;       Now to initialize channel B.
;
 MOV SI,OFFSET PARMTB ;  SI=Address of Chn B parm table.
 MOV CX,PARMBS  ;  CX=Parameter table size.
 MOV DX,0E4H          ;  DX=Port 1,Channel B Command address.
INIB: LODS DS:BYTE PTR[SI]  ;  Get byte from parameter table.
 OUT DX,AL    ;  Write it until all registers
 LOOP INIB    ;  are programmed.
 RET
COMMEX ENDP
$EJECT
```

```
;***************************************************************************
;   This area contains the initialization parameters for channels A and B
;   of port 1.
;****************************************************************************
;
; Initialization parameters for channel A.
;
PARMTA  LABEL NEAR
        DB 09   ; Select WR9 code.
 DB 11000000B  ; Reset 8530.
 DB 11   ; Select WR11 code.
 DB 01010010B  ; Rcv clock=Baud rate generator.
     ; Xmt clock=Baud rate generator.
  DB 14   ; Select WR14.
 DB 00000011B  ; Enable baud rate generator.
 DB 12   ; Select WR12.
 DB 6   ; Baud rate (low byte)= 9600 baud.
 DB 13   ; Select  WR13.
 DB 0   ; Baud rate (high byte)= 9600 baud.
 DB 15   ; Select WR15.
 DB 0   ; Disable external status interrupts.
 DB 1   ; Select WR1.
 DB 0   ; Disable all other interrupts.
 DB 3   ; Select WR3.
 DB 01000001B  ; Rcv=7 bits of data + parity bit.
 DB 4   ; Select WR4.
 DB 01000110B  ; x16 clock input,1 stop bit,
     ; even parity enabled.
 DB 5   ; Select WR5.
 DB 10101010B  ; Turn on DTR and RTS,
     ; Transmit enable,
     ; Xmt=7 bits of data + parity bit.
PARMAS EQU $-PARMTA
;
; Initialization parameters for channel B.
;
PARMTB LABEL NEAR
 DB 15   ; Select WR15.
 DB 00   ; Disable external status interrupts.
 DB 01   ; Select WR1.
 DB 00   ; Disable all other interrupts.
PARMBS EQU $-PARMTB
;
$EJECT
```

```
;*************************************************************************
;  8530 Receive Character Routine
;
;  This routine is called to read a single received character from the
;  8530 receive receive fifo.  If no character is available in the fifo,
;  this routine waits until a character is received before returning to
;  the caller.
;
;*************************************************************************
READCH PROC NEAR
 MOV DX,0E6H    ; DX=Port 1, Chn A, command address.
TRYRAG: IN AL,DX    ; Read RR0 contents.
 AND AL,00000001B  ;Q: Any characters in rcv fifo ?
 JZ TRYRAG    ;  No, try again.
 MOV DX,0E7H    ;  Yes, DX=data port address.
 IN AL,DX    ; AL=character received.
 RET
READCH ENDP
$EJECT
```

```
;***************************************************************************
;  8530 Transmit Character Routine
;
;  This routine is called to write a single character (in AL register) to
;  the 8530 for transmission.  If a character is currently being transmitted
;  this routine waits until transmission of that character completes before
;  attempting to transmit the next character.
;
;***************************************************************************
WRITEC PROC NEAR
 MOV DX,0E6H    ; DX=Port 1, Chn A, command address.
TRYXAG: IN AL,DX    ; Read RR0 contents.
 AND AL,00000100B  ;Q: Character being transmitted ?
 JZ TRYXAG    ;  Yes, try again.
 MOV DX,0E7H    ;  No, DX=data port address.
 OUT DX,AL    ; AL=character received.
 RET
WRITEC ENDP
BIOCODE ENDS
```

Appendix F

MODEM SAMPLE ROUTINES

RCNTL

```
;*************************************************************
;  RCNTL -  This subroutine determines whether a modem is
;           installed in port 1 and if so, activates the
;           RCNTL signal to initiate the modem Control Mode.
;*************************************************************
RCNTL PROC   NEAR
      MOV    DX,00E4H         ; DX = PORT 1 CHANNEL B ADDRESS.
      MOV    AL,05            ; WR5 SELECT.
      OUT    DX,AL            ; SELECT REGISTER 5.
      MOV    AL,02            ; TURN ON RCNTL (RTS IN CHANNEL B).
      OUT    DX,AL            ;
;
;     NOW TO DETERMINE IF MODEM IS INSTALLED.
;
LOOP: MOV    DX,00E6H         ; DX = PORT 1 CHANNEL A ADDRESS.
      MOV    AL,10H           ; RESET EXTERNAL STATUS INTERRUPTS.
      OUT    DX,AL            ;
      IN     AL,DX            ; READ RR0.
      TEST   AL,00010000B     ;Q: IS ACNTL (SYNCA) ACTIVE ?
      JZ    LOOP              ; NO, CONTINUE TO LOOK FOR ACNTL.
      RET                     ; YES, RETURN TO CALLER IN CONTROL MODE.
RCNTL ENDP
```

DIAGST

```
;*************************************************************
; DIAGST - This routine requests the diagnostics status from
;          the modem and returns the result in register AL.
;          It is assumed that the Zilog 8530 has been previously
;          initialized and that the modem has been placed in
;          Control Mode.
;*************************************************************
DIAGST  PROC    NEAR
        MOV     DX,00E7H          ; DX = PORT 1, CHANNEL A DATA PORT ADDRESS.
        MOV     AL,'D'            ; AL = DIAGNOSTIC COMMAND CODE.
        OUT     DX,AL             ; REQUEST MODEM DIAGNOSTICS STATUS.
LOOP:   MOV     DX,00E6H          ; DX =PORT 1, CHANNEL A COMMAND ADDRESS.
        IN      AL,DX             ; READ CHANNEL A'S RR0.
        TEST    AL,00000001B      ;Q: HAS A CHARACTER ARRIVED FROM MODEM ?
        JZ      LOOP              ; NO, WAIT FOR COMMAND RESPONSE.
        MOV     DX,00E7H          ; YES, DX = PORT 1 DATA PORT ADDRESS.
        IN      AL,DX             ; READ DATA FROM RCV FIFO.
        RET                       ; RETURN WITH STATUS IN AL.
DIAGST  ENDP
```

DIALER

```
;*********************************************************
; DIALER - This routine dials a typical phone number. It
;          does not monitor the progress of the call and
;          it assumes the Zilog 8530 has been previously
;          initialized and that the modem has been placed
;          in Control Mode.
;
;          The phone number to be dialed is contained in
;          a buffer (phonum) and is terminated by a null.
;*********************************************************
DIALER  PROC    NEAR
        MOV     DX,00E7H         ; DX = PORT 1, CHANNEL A DATA PORT ADDRESS.
        MOV     DI,OFFSET PHONUM ; DI=ADDRESS OF PHONE NUMBER BUFFER.
        MOV     AL,'T'           ;Use T for touch tone
        OUT     DX,AL            ;Transmit - command to modem

;       Nest send the strip of telephone numbers


LOOP:   MOV     AL,[DI]          ; GET PHONE NUMBER DIGIT.
        XOR     AL,AL            ;Q: END OF PHONE NUMBER ?
        JE      SENDPT           ; YES, SEND PHONE NUMBER TERMINATOR.
        OUT     DX,AL            ; NO, SEND DIGIT TO MODEM.
        INC     DI               ; POINT TO NEXT DIGIT.
        JMP     LOOP             ; CONTINUE IN LOOP.
;
SENDPT: MOV     AL,'X'           ; AL = PHONE NUMBER TERMINATOR COMMAND.
        OUT     DX,AL            ; SEND TO MODEM.
;
;       NOW TO WAIT FOR THE DIAL COMMAND COMPLETION.
;       AL RETURNS THE STATUS OF THE DIAL COMMAND.
LOOP1:  MOV     DX,00E6H         ; DX =PORT 1, CHANNEL A COMMAND ADDRESS.
        IN      AL,DX            ; READ CHANNEL A'S RR0.
        TEST    AL,00000001B     ;Q: HAS A CHARACTER ARRIVED FROM MODEM ?
        JZ      LOOP1            ; NO, WAIT FOR COMMAND RESPONSE.
        MOV     DX,00E7H         ; YES, DX = PORT 1 DATA PORT ADDRESS.
        IN      AL,DX            ; READ DATA FROM RCV FIFO.
        RET
DIALER  ENDP
```

Appendix G

BOOT ROUTINE AND SAMPLE ASSEMBLY CODE

Appendix G gives a sample source program that could be in the boot sector. This example is excerpted from the MS-DOS V1.10 boot sector.

```
;**********************************************************************
;
;  TITLE     - BOOT -   SAMPLE BOOT ROUTINE FOR THE TI PROFESSIONAL COMPUTER
;
;  ABSTRACT - This routine is responsible for loading the system files
;       from the disk.  This routine resides in the 'boot' sector
;       (track 0 sector 1) of the disk which is loaded at absolute
;       location 0C000H boot code in the system ROM and then executed.
;**********************************************************************
        NAME    BOOT
        TITLE   TIPC BOOT LOGIC)
        PAGE
;**********************************************************************
;                           LOCAL CONSTANTS
;**********************************************************************
;
VERS    EQU     0                 ; Current version of BOOT logic
REV     EQU     0                 ; Revision level
;
CR      EQU     0DH
LF      EQU     0AH
;----------------------------------------------------------------------
; WINCHESTER disk DIT (Disk Interface Table) equates
;
DITSTRC STRUC
DITDIR  DD      0                 ; Disk Interface Routine vector (dword)
DITSEC  DW      512               ; Sector size in bytes (word)
DITTRK  DB      17                ; Track size in sectors (byte)
DITCYL  DB      4                 ; Cylinder size in tracks (byte)
DITDSK  DB      153               ; Disk size in cylinders (BYTE)
DITERR  DB      1                 ; Maximum number of error retries
DITWRC  DW      64                ; reduced write current
DITPRC  DW      64                ; Write pre-comp threshold cylinder
DITSTP  DB      10000000B         ; Step option
DITBUR  DB      11                ; Error burst length
        DW      0000              ; reserved for expansion
DITSTRC ENDS
        PAGE
;----------------------------------------------------------------------


;   ROM BIOS interface vectors:
;
BELINT  EQU     48H               ; System beeper I/O and general ROM interface
CRTINT  EQU     49H               ; Screen I/O
KEYINT  EQU     4AH               ; Keyboard I/O
PRTINT  EQU     4BH               ; Parallel port I/O
GAMINT  EQU     4CH               ; Analog Input/Clock I/O
DSKINT  EQU     4DH               ; Floppy disk I/O
CLKINT  EQU     4EH               ; Time-of-day clock I/O
CONINT  EQU     4FH               ; System configuration
```

```
;
;-----------------------------------------------------------------------------
;   FIXED ROM DATA AREA - (absolute offsets from absolute 0)
;   These equations define the ROM communications area, containing data
;   that must be accessed by both the ROM and user/application programs.
;   This data is accessed from the 'user' program by setting DS = 0.
;
DSADDR  EQU     4*60H           ; (WORD) pointer to DS for System ROM (ROMDAT)
DSSIZR  EQU     4*60H+2         ; (WORD) size of DS for System ROM (ROMDAT)
DSADD0  EQU     4*61H           ; (WORD) pointer to DS for ROM at ROMCOD:0000
DSSIZ0  EQU     4*61H+2         ; (WORD) size of DS for ROM at ROMCOD:0000
DSADD2  EQU     4*62H           ; (WORD) pointer to DS for ROM at ROMCOD:2000
DSSIZ2  EQU     4*62H+2         ; (WORD) size of DS for ROM at ROMCOD:2000
DSADD4  EQU     4*63H           ; (WORD) pointer to DS for ROM at ROMCOD:4000
DSSIZ4  EQU     4*63H+2         ; (WORD) size of DS for ROM at ROMCOD:4000
DSADD6  EQU     4*64H           ; (WORD) pointer to DS for ROM at ROMCOD:6000
DSSIZ6  EQU     4*64H+2         ; (WORD) size of DS for ROM at ROMCOD:6000
DSADD8  EQU     4*65H           ; (WORD) pointer to DS for ROM at ROMCOD:8000
DSSIZ8  EQU     4*65H+2         ; (WORD) size of DS for ROM at ROMCOD:8000
MEMSIZ  EQU     4*66H           ; (WORD) memory size (number of 16-byte blocks)
;-----------------------------------------------------------------------------
; DISK DSR OPERATION CODES
;
DKRSET  EQU     0               ; Reset disk system, drive parms must be preset
DKSTAT  EQU     1               ; Get disk status in (al)
DKREAD  EQU     2               ; Read sectors into memory
DKWRIT  EQU     3               ; Write memory to disk sectors
DKVERF  EQU     4               ; Verify crc on disk sectors
DKVRFY  EQU     6               ; Verify memory against disk sectors
DKSSTA  EQU     7               ; Get disk status for pre-retry (if any)
DKFSET  EQU     8               ; Set UNIT & standard DIT for a drive
DKXSET  EQU     9               ; Set UNIT & DIT address for a drive
DKRDIT  EQU     10              ; Return DIT address for drive
DKKMOT  EQU     11              ; Turn off Floppy Disk Motors
DKBADC  EQU     12              ; Old >= this is a bad command
;*****************************************************************************
;
;   IO segment - defines load address and entry point for BIOS
;
IO      SEGMENT AT 40H          ; absolute location 400H
        ASSUME  CS:IO
;
IOSYS   PROC    FAR             ; IO.SYS loaded here (40:0000)
IOSYS   ENDP
;
IO      ENDS
        PAGE
CODE    SEGMENT AT 0000         ; Absolute location 0000H
        ASSUME  CS:CODE, DS:CODE, SS:CODE, ES:CODE
```

```
;***********************************************************************
;       Data area for ROM definitions
;***********************************************************************
        ORG     400H
OLDRMD  LABEL   WORD                ; Initial location of ROM data area
;
IOSIZ   EQU     7                   ; Number of sectors in IO.SYS
        ORG     1200H               ; (OLDRMD+7*512)
ROMDAT  LABEL   BYTE                ; Location of rom data area
        ORG     1200H+13CH
;
MOVDST  EQU     (ROMDAT-OLDRMD)/16
;***********************************************************************
;                       MODULE ENTRY POINT
;***********************************************************************
;
        ORG     0C000H
BOOT    PROC    FAR                 ; Entry point for boot logic
        JMP     BOOTST
;
;-----------------------------------------------------------------------
;                       HEADER DATA AREA
;-----------------------------------------------------------------------
        ORG     0C003H              ; Always start here
;
;   File access table - Shows the loader where to find IO.SYS
;
IOSEC   DB      8                   ; 1 side 40 track load sector
IOTRK   DB      0                   ; 1 side 40 track load track
IOHEAD  DB      0                   ; 1 side 40 track load head
;
;       THE FOLLOWING BYTE MUST BE SET UP BY THE FORMAT COMMAND TO
;       INDICATE THE DRIVE TYPE FOR WHICH THE DISK IS FORMATTED.
;       THE PERMISSIBLE VALUES ARE 0-3 WHICH CORRESPOND  WITH THE
;       FLOPPY DISK TYPE.
;
DSKTYP  DB      00                  ; Disk formatted type
;
BOODRV  DB      00                  ; Storage for boot drive number
;
;       WINCHESTER DIT
;
WINDIT  DITSTRC <>                  ; 18 BYTES LONG
;
SIGNON  DB      CR,LF,'BOOT V'
        DB      VERS/10+'0','.',VERS MOD 10+'0',REV+'0'
        DB      ' (c) 1983 Texas Instruments, Inc.',0
;
        PAGE
;***********************************************************************
```

```
;
;   DISK BOOT LOGIC - ROMDAT is moved from its initial location (absolute
;       address 400H) to its working location under MSDOS (absolute address
;       1200H).  This code is called (FAR) by the ROM.  If an error is
;       found, it can perform a RETurn to let the ROM handle it in the
;       same manner as the other boot-time errors.
;
;   INPUT:  BL = Floppy drive from which to attempt the boot
;               stack is set up below this code by the ROM
;
BOOTST:
        MOV     SI,OFFSET SIGNON; Signon the boot sector
        CALL    MSG
;
;   First, move the ROM data area out of the way.
;
        PUSH    ES              ; Save the ROM's ES
        MOV     AX,CS           ; Note that CS = CODE = 0000H)
        MOV     DS,AX           ; DS = CS = CODE = 0000H
        MOV     ES,AX           ; ES = CS = CODE = 0000H
        MOV     BOODRV,BL       ; Save boot drive
        MOV     BX,DSADD8       ; Point to last possible rom data area pointer
BOO2:   MOV     AX,[BX]         ; Get data pointer
        OR      AX,AX           ;Q:  Data area in use?
        JNZ     BOO4            ; Y: Jump and calc data length
        SUB     BX,4            ; N: Point to next data area pointer
        JMP     BOO2            ;    And check it
;
BOO4:   MOV     CL,4              ; Convert dsaddX pointer to absolute address
        SHL     AX,CL             ;
        ADD     AX,[BX+2]         ; And add in the last data area length
        SUB     AX,offset OLDRMD; Subtract the original location
        MOV     CX,AX             ; Results in total length to move
        MOV     SI,offset OLDRMD; DS:SI = source for the move
        MOV     DI,offset ROMDAT; ES:DI = destination for the move
        MOV     BP,CX             ; Get length of move into BP
        ADD     BP,DI             ;   + ROMDAT = lowest available memory
;
        MOV     DX,DS:word ptr DSADDR ; Pick up the ROMDAT pointer
        CMP     DX,(offset ROMDAT/16) ; Q: Has the move already taken place ?
                                      ;   (True if ROM is retrying the boot)
        JE      BOO7                  ;   Y: Then skip the move this time
;                                     ;   N: Then do the move
        ADD     SI,CX                 ; Do the move in reverse in case ROMDAT
        ADD     DI,CX                 ;  area is larger than move length.
        DEC     SI                    ;  0 relative
        DEC     DI                    ;  0 relative
        STD                           ;
        CLI                           ; Protect the move
        REP     MOVSB                 ;;; Do the move
        CLD                           ;;; RESET STRING DIRECTION
        MOV     BX,DSADDR             ;;; Set up the rest - start with DSADDR
BOO5:
        CMP     word ptr [BX],0000    ;;; Q: ROM's DSADDx = zero ?
```

```
        JZ      BOO6                        ;;;   N: Not installed, go to next one
        ADD     word ptr [BX],MOVDST        ;;;   Y: Then adjust it
BOO6:                                       ;;;
        CMP     BX,DSADD8                   ;;; Q: Are we all done ?
        JE      BOO7                        ;;;   Y: Continue with boot
        ADD     BX,4                        ;;;   N: Point to the next ROM's DSADDx
        JMP     SHORT BOO5                  ;;; and loop for next one
BOO7:
        STI                                 ;;; Shields down
;
;     HERE ROMDAT HAS BEEN MOVED AND BP CONTAINS LOWEST AVAILABLE MEM ADDRESS
;
;       TELL DSR ABOUT THIS DISK TYPE
;
        MOV     DL,BOODRV           ; Drive number
        MOV     AL,DSKTYP           ; Get the disk formatted type
        MOV     AH,DKFSET           ; Set the floppy DIT opcode
        INT     DSKINT              ; Go do it.
;
;       Set up the WINCHESTER if it is installed.
;
        MOV     AX,DS:WORD PTR DSADD4  ;check for winchester
        OR      AX,AX                  ;Q;  winchester installed?
        JZ      BOOT20                 ; N; jump
        PUSH    ES                     ; Y; save DS
        MOV     ES,AX                  ;get winchester ROM ES
        MOV     SI,OFFSET WINDIT+4     ;get pointer to new DIT
        MOV     AH,0                   ;copy and set new winchester DIT
        CALL    ES:DWORD PTR 0000      ;call the winchester ROM
        POP     ES                     ;retrieve ES
BOOT20:
;
; Load IO.SYS first - 7 sectors (3.5K) loaded, have to miss the ROM data area
;
; If a disk error occurs,  it
; returns to the caller for error handling (the caller is assumed to be
; the routine DKBOOT in the System ROM).
;
        MOV     BX,offset OLDRMD          ; Transfer offset (ES already set)
        MOV     CX,word ptr IOSEC         ; Starting at proper track and sector
        MOV     DH,IOHEAD                 ;  ... and head
        MOV     DL,BOODRV                 ; From boot disk.
                                          ; 7 sectors
        MOV     AX,DKREAD*256+IOSIZ       ; Select disk read function
        INT     DSKINT                    ; Disk DSR
        JB      NOBOOT                    ; If error, die
        MOV     BL,DL                     ; Tell BIOS init about the boot drive
        MOV     AX,BP                     ; And the lowest available address
        JMP     IOSYS                     ; Else, go to BIOS init code
;
;    register AH contains an error code to be reported by the ROM
;
NOBOOT:
        POP     ES                  ; Restore original ES before ROM gets at it
```

```
        MOV     DS,DS:word ptr DSADDR   ; And point DS at the new ROMDAT
        RET                     ; FAR Return to DKBOOT
BOOT    ENDP
        PAGE
;*****************************************************************************
;   MSG - Output string of characters in the current CS to the CRT.
;       The string should be terminated with a zero byte.
;   INPUT:  SI = offset of string in current CS
;   OUTPUT: (screen)
;   USED:   AX,SI
;   STACK:
;-----------------------------------------------------------------------------
MSG     PROC    NEAR
MSG0:
        LODS    CS:byte ptr [SI]; Get the char
        OR      AL,AL           ; Q:  Last char?
        JNZ     MSG1            ;  N: Jump and print it.
        RET                     ;  Y: *** RETURN ***
MSG1:   MOV     AH,CRTWTY
        INT     CRTINT          ; Else print it
        JMP     MSG0            ; And loop
MSG     ENDP
;
CODLEN  EQU     $-BOOT          ; LENGTH OF THE CODE
CODFIL  EQU     512-CODLEN-4    ; TOTAL SPACE AVAIL FOR CODE
        DB      CODFIL DUP (0)  ; SPACE FILLER
;
        DB      'ti'            ; Disk identifier
        DW      0000H           ; Boot sector CRC (Calculated by a utility)
;
CODE    ENDS
        END
```

The following pages show a sample assembly code. This code will set up a 2048 byte ROM at address F400:6000.

```
;
DATASIZ EQU     30H             ;length of required data area
                                ;can be zero but must be multiple of 16
;ROMDS   EQU     184H           ;for ROM at F400:0000
;ROMDS   EQU     188H           ;for ROM at F400:2000
;ROMDS   EQU     18CH           ;for ROM at F400:4000
ROMDS   EQU     192H            ;for ROM at F400:6000
;ROMDS   EQU     196H           ;for ROM at F400:8000 (main board)
;
CODE    SEGMENT AT 0F400H
        ASSUME CS:CODE
        ORG     6000H
;
; ROM HEADER
;
        DW      2048                    ;ROM size
        DW      ENTRY                   ;entry point address
        DB      MSGLEN                  ;message length
MSGBEG  DB      0DH,0AH                 ;carriage return, line feed
        DB      'V1.23 XMPROM,          ;version, 6-character name,
                example ROM             ;    message
        DB      0DH,0AH                 ;carriage return, line feed
MSGLEN  EQU     $-MSGBEG
;
;  ENTRY POINT FOR POWERUP CODE
;
ENTRY   PUSH    BX                      ;save important registers
        PUSH    DX                      ;
        PUSH    SI                      ;
        PUSH    DS                      ;
;
; ALLOCATE OPTION ROM DATA AREA IN RAM
;
        XOR     AX,AX                   ;setup segment to point
        MOV     DS,AX                   ;  to vector area
        MOV     BX,180H                 ;check for RAM in use starting
                                        ;  with system area
        MOV     AX,[BX]                 ;get segment address for ROM
        MOV     CL,4                    ;convert to absolute address
        SHL     AX,CL                   ;
ENT00:  ADD     AX,[BX+2]               ;add in length of segment
        ADD     BX,4                    ;point to next ROMS RAM seg pointer
        CMP     BX,ROMDS                ;Q:  is this the pointer for my ROM?
        JNZ     ENT00                   ; N: continue adding up RAM usage
        SHR     AX,CL                   ; Y: convert address to segment
        MOV     [BX],AX                 ;    store my segment address
        MOV     [BX+2],DATASIZ          ;    and the segment length
;
; SET UP MY DS AS REQUIRED TO MY DATA AREA.  THIS CAN BE DONE
; EACH TIME THIS ROM IS CALLED.
;
```

```
        MOV DS,CS:(WORD PTR ROMDS+0C000H)

    . . .                                 ;additional init coed as required

        POP     DS                        ;retrieve the calling ROMS regs
        POP     SI                        ;
        POP     DX                        ;
        POP     BX                        ;
;
        ORG     6000H+2048-2              ;address for the ROM CRC
        DW      ROMCRC                    ;
        ENDS
        END
```

Appendix H

SAMPLE INTERRUPT SERVICE ROUTINE

An ISR example, with the appropriate routines to install and remove it, follows. The source of the common interrupt exit routine and the code to count the number of outstanding interrupts (INTCTR) are also given. Using this code is not mandatory, but is recommended to maintain future compatibility.

Example 1

```
;------------------------------------------------------------------
;  SAMPLE  INTERRUPT SERVICE ROUTINE - (RAM-based; for ROM-based
;    code, the local data area would have to be in a separate DS)
;------------------------------------------------------------------
;
INTSEG  SEGMENT BYTE PUBLIC       ; Segment declaration
        ASSUME  CS:INTSEG
;
;   Local constants:
;
INTNUM  EQU     46H               ; Interrupt number (example)
STKSIZ  EQU     30                ; Size of local stack, including
                                  ;   space required to save any
                                  ;   registers used.
DSADDR  EQU     180H              ; Offset of pointer to System
                                  ;   ROM's DS (Segment = 0000)
INTCTR  EQU     19AH              ; Offset of outstanding interrupt
                                  ;   counter (Segment = 0000)
;
;   Local data storage:
;
DATSEG  DW      0000              ; If it is necessary for the
                                  ;   interrupt service routine
                                  ;   to access a certain DS, it
                                  ;   should be set up in this
                                  ;   variable when the interrupt
                                  ;   vector is initially installed
;
VECSAV  DW      0000              ; Save area for original vector
        DW      0000              ;   (It must be restored when the
                                  ;   user application finishes)
;
STKSAV  DW      0000              ; Location to save stack pointer
        DW      0000
;
        DW      STKSIZ+4 DUP (?); The local stack.  STKSIZ should
                                  ;   be the size (in words) of the
                                  ;   service routine's stack.  The
                                  ;   '+4' is to allow stack space
                                  ;   for the service routine to be
                                  ;   interrupted by a higher-
                                  ;   priority interrupt.
INTSTK  LABEL   WORD              ; Top of stack
```

Example 2

```
;------------------------------------------------------------------
;   The actual ISR - the appropriate hardware interrupt vector is
;   set up by the installation routine to point to this routine.
;------------------------------------------------------------------
INTSRV  PROC    FAR
;
;   In the case of shared interrupt levels, any determination of
;   which device caused the interrupt should be placed here,
;   before the local stack swap.  Note that this routine can't
;   use any registers or stack (if it does, it must be moved
;   to some point after the stack swap).
;       .
;       .
;       .
;
        PUSH    DS                  ; Save DS on 'interrupted' stack
        MOV     CS:STKSAV+2,SS      ; Save current SS & SP
        MOV     CS:STKSAV,SP
        MOV     SP,CS               ; Stack segment = local Code Seg
        MOV     SS,SP               ;
        MOV     SP,offset INTSTK; SS:SP = Local interrupt stack
        PUSH    AX                  ; Save AX
        XOR     AX,AX               ; AX = 0000
        MOV     DS,AX               ; Point to vector area
        INC     DS:(byte ptr INTCTR) ; Increment interrupt counter
;
;   Insert code to reset the cause of the interrupt
;       .
;       .
;       .
;
        STI                         ; Safe to re-enable interrupts now
;
        PUSH    BX                  ; Save rest of the environment
        PUSH    CX                  ; (The user need save only those
        PUSH    DX                  ;   registers actually used in
        PUSH    DI                  ;   his particular ISR)
        PUSH    SI
        PUSH    BP
        PUSH    ES
        MOV     DS,CS:DATSEG        ; Set up local DS from value
                                    ;   saved during installation.
;
;   Insert specific interrupt processing logic here
;       .
;       .
;       .
;
;
```

```
;    Exit logic
;
         POP     ES
         POP     BP                   ; Restore environment
         POP     SI
         POP     DI
         POP     DX
         POP     CX
         POP     BX
;
         CLI                          ; Disable interrupts
         MOV     AL,20H               ; Reset 8259 interrupt controller
         OUT     18H,AL
         XOR     AX,AX                ; AX = 0000
         MOV     DS,AX                ; Point to vector area
         DEC     DS:(byte ptr INTCTR) ; Decrement interrupt counter
         POP     AX
         MOV     SS,CS:STKSAV+2       ; Restore original SS:SP
         MOV     SP,CS:STKSAV         ;
         POP     DS                   ; Restore original DS
         IRET                         ; ** INTERRUPT RETURN **
;
INTSRV   ENDP
```

Example 3

```
;-----------------------------------------------------------------
;   Interrupt service routine INStallation routine
;-----------------------------------------------------------------
;
INTINS  PROC    NEAR
        PUSH    AX
        PUSH    BX
        PUSH    DS
        MOV     CS:DATSEG,DS        ; Set up CS-relative pointer to
                                    ;   the local DS - This is
                                    ;   necessary because the only
                                    ;   reference the ISR has when
                                    ;   it is invoked is the CS.
;
;   Patch the interrupt vector to point to the Interrupt Service
;   Routine, saving the original vector.  This illustrates the
;   'brute force' method of setting and getting vectors.  Most
;   8088 Operating Systems (e.g. MS-DOS) have system calls to
;   accomplish this feat.  Their use is preferable, because some
;   Operating Systems attempt to arbitrate vector usage.
;
        XOR     AX,AX               ; Clear AX
        MOV     DS,AX               ; DS <-- 0000
        CLI                         ; Protect the vector operation
;
;   Pick up original vector
;
        MOV     AX,DS:(word ptr (INTNUM*4))
        MOV     BX,DS:(word ptr (INTNUM*4+2))
;
;   Save original vector in local save area
;
        MOV     CS:VECSAV,AX
        MOV     CS:VECSAV+2,BX
;
;   Install vector to Interrupt Service Routine
;
        MOV     DS:(word ptr (INTNUM*4)),offset INTSRV
        MOV     DS:(word ptr (INTNUM*4+2)),CS
        STI                         ; Interrupts OK again
        POP     DS
        POP     BX
        POP     AX
        RET                         ; *** RETURN ***
INTINS  ENDP
```

Example 4

```
;-----------------------------------------------------------------
;   Interrupt Service Routine REMoval routine
;-----------------------------------------------------------------
INTREM  PROC    NEAR
        PUSH    AX
        PUSH    BX
        PUSH    DS
        XOR     AX,AX               ; Clear AX
        MOV     DS,AX               ; DS <-- 0000
        CLI                         ; Protect the vector operation
;
;   Get original vector from local save area
;
        MOV     AX,CS:VECSAV
        MOV     BX,CS:VECSAV+2
;
;   Restore original vector
;
        MOV     DS:(word ptr (INTNUM*4)),AX
        MOV     DS:(word ptr (INTNUM*4+2)),BX
        STI                         ; Interrupts OK again
        POP     DS
        POP     BX
        POP     AX
        RET                         ; *** RETURN ***
INTREM  ENDP

INTSEG  ENDS
        END
```

This is the source for the common interrupt exit routine as it exists in ROM. Any other common exit routine installed here will perform an identical function. The user should use this exit if the installed interrupt service routine will be running concurrently with a real-time Operating System (for instance, during the execution of any of the TI communication packages).

Example 5

```
;------------------------------------------------------------------
;    Common Interrupt Exit logic
;
;    INPUT:   ES:BX = SS:SP of the interrupted code
;             Interrupt stack contains saved ES,BX,AX (ES at top
;               of stack)
;             Stack of interrupted code contains saved DS
;------------------------------------------------------------------
ROMDAT  SEGMENT BYTE PUBLIC
        EXTRN   IXSSSV:WORD      ; Temporary stack pointer save
        EXTRN   IXSPSV:WORD
ROMDAT  ENDS

ROMCOD  SEGMENT BYTE PUBLIC
        ASSUME  CS:ROMCOD, DS:ROMDAT
;
INTXIT  PROC    FAR
        CLI                      ; Disable interrupts
        MOV     AL,20H           ; Reset 8259 interrupt controller
        OUT     18H,AL
        DEC     CS:(byte ptr INTCTR+0C000H) ; Decrement interrupt
                                 ;   counter (remember, this is in
                                 ;   ROM, so access to the vector
                                 ;   area is CS-relative)
        MOV     DS,CS:(word ptr DSADDR+0C000H) ; Get ROM's DS
        MOV     IXSSSV,ES        ; Save SS,SP of original code
        MOV     IXSPSV,BX
        POP     ES               ; Restore commonly used registers
        POP     BX               ;   from interrupt stack
        POP     AX
        MOV     SS,IXSSSV        ; Restore original SS,SP
        MOV     SP,IXSPSV
        POP     DS               ; Restore DS from original stack
        IRET                     ; *** INTERRUPT RETURN ***
INTXIT  ENDP
;
ROMCOD  ENDS
        END
```

Section 4

ASSEMBLY DRAWINGS AND LISTS OF MATERIALS

This section contains assembly drawings and lists of materials applicable to the Texas Instruments Professional Computer.

Drawings not available in time for printing:

Color Display Unit	2223219
Winchester Disk Controller	2223220
Parallel Test Plug Assembly	2223276
PWB,Parallel Test Plug	2223277
Configuration,Diskette Drive	2223279
Power Cord AC	0996289
Communications Loopback Plug	2207985

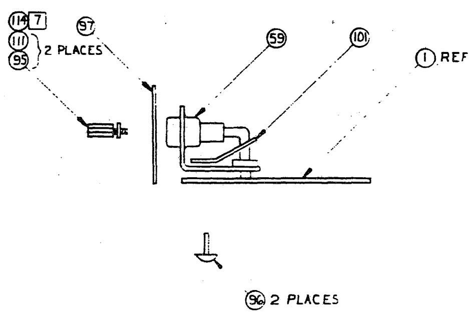
DWG NO 2223003 SH 1

NOTES UNLESS OTHERWISE SPECIFIED

1. MAXIMUM COMPONENT HEIGHT ABOVE THE BOARD SHALL BE 12.7 IN AREA INDICATED, 15.7 ALL OTHER AREAS
2. MAXIMUM LEAD LENGTH BELOW THE BOARD SHALL BE 3.9C
3. MARK APPROPRIATE DASH NUMBER AND REVISION LETTER IN SPACE INDICATED PER PROCESS 3
4. MARK SITE/DATE CODE AND SERIAL NUMBER IN SPACE INDICATED PER ITEM 100 PARAGRAPH 4.C AND PROCESS 3
5. CAUTION THIS ASSEMBLY AND CERTAIN COMPONENTS ARE SUSCEPTIBLE TO DAMAGE FROM ELECTROSTATIC DISCHARGE. OPERATOR AND EQUIPMENT GROUND AND PACKAGING IS REQUIRED. STATIC SENSITIVE COMPONENTS ARE: U1 THRU U67
6. INSTALL ITEM 98 (QTY 4) TO E2, E4, E6 & E20 (SINGLE PIN ONLY)
7. APPLY ITEM 114 (LOCKTITE SEALANT) AFTER INSTALLATION OF ITEMS 95 AND 111 ON SCREW END

REVISIONS			
REV / DESCRIPTION		DATE	APPROVED
A	CN 493819 (D)	5-28-82	M Bull
	(1) LM UPDATE		
B	CN493821 (E)	6-29-82	M Bull
	(1) IT 60 WAS QTY 3 (2) IT 60A WAS CR1, CR2, CR3 (3) IT 46 WAS PN 2211109-2 (4) IT 12 WAS PN 2220400-2 (5) ADD IT 103 & 104		
C	CN 477847 (D) Black	8-3-82	M Bull
	(1) UPDATE LM (2) CREATE -5001 LM (3) ADD DASH NUMBER CHART		
D	CN 502163 (D)	8-17-82	M Bull
	(1) LM UPDATE		
E	CN476090 (B) LAMB	10-4-82	M Bull
	(1) CHNG PER ENG REDESIGN		
F	CN 502198 (D)	11-1-82	M Bull
	(1) LM UPDATE (2) CHNG PER ENG SPEC		

29



VIEW A
SH 2 (D-5)

2223003-5001	MOTHERBOARD-PEGASUS, AUTO INSERT
2223003-0001	MOTHERBOARD-PEGASUS
PART NUMBER	DESCRIPTION

CONVERSION CHART	
mm	INCHES
0.25	.010
0.5	.02
1.5	.06
3.90	.154
12.7	.50
15.7	.62
86.36	3.40
184.15	7.25

SEQ NO	IDENT	F-SPEC	NO	ADDITIONAL	NOTES
3	MARK	902-01	01	HGT 1.5 CLR WHT, (CAT 4)	3 4
2	SLDR	127-01	00	HAND SOLDER	
1	SLDR	124-02	00	WAVE SOLDER	
	PROCESS		CLASSIFICATION		

PROCESSES — FOR CORRELATION TO GOVT/IND SPECIFICATIONS SEE TI DRAWING 72946

TEST PROC	2223270-0001					*	*
ASSEMBLY	2223003-0001	A	B	C	D	E	F
PWB	2223004-0001	B	B	B	C	C	E
DIAGRAM	2223005-0001	*	*	*	A	A	A

REV STATUS OF SHEETS			
REV	F	F
SH	1	2

-3 QTY	-2 QTY	-1 QTY	ITEM NO	PART OR IDENTIFYING NUMBER	NOMENCLATURE OR DESCRIPTION	NOTES

PARTS LIST

NEXT ASSY	USED ON
2223038	8755

APPLICATION

UNLESS OTHERWISE SPECIFIED
- DIMENSIONS ARE IN MILLIMETERS
- TOLERANCES: 2 PLACE DECIMALS ±0.25; 1 PLACE DECIMALS ±0.5; ANGLES ±1°
- INTERPRET DRAWING PER DOD-D-1000
- REMOVE ALL BURRS AND SHARP EDGES
- CONCENTRICITY MACHINED DIAMETERS 0.25 FIM
- DIMENSIONAL LIMITS APPLY BEFORE PROCESSES
- PARENTHETICAL INFO FOR REF ONLY

HOLE TOLERANCE	
0.33 THRU 3.18	+0.10/-0.03
3.20 THRU 6.35	+0.13/-0.03
6.38 THRU 12.70	+0.15/-0.03
12.73 THRU 19.05	+0.20/-0.03
19.08 THRU 25.40	+0.25/-0.03
25.43 THRU 50.80	+0.30/-0.03

THIRD ANGLE PROJECTION

DWN 3/15/82
CHK 4/30/82
ENGR 10-4-82
APVD ENGR
QA
APVD MFG
RLSE 5/19/82

TEXAS INSTRUMENTS INCORPORATED

SI-METRIC

MOTHERBOARD-PEGASUS

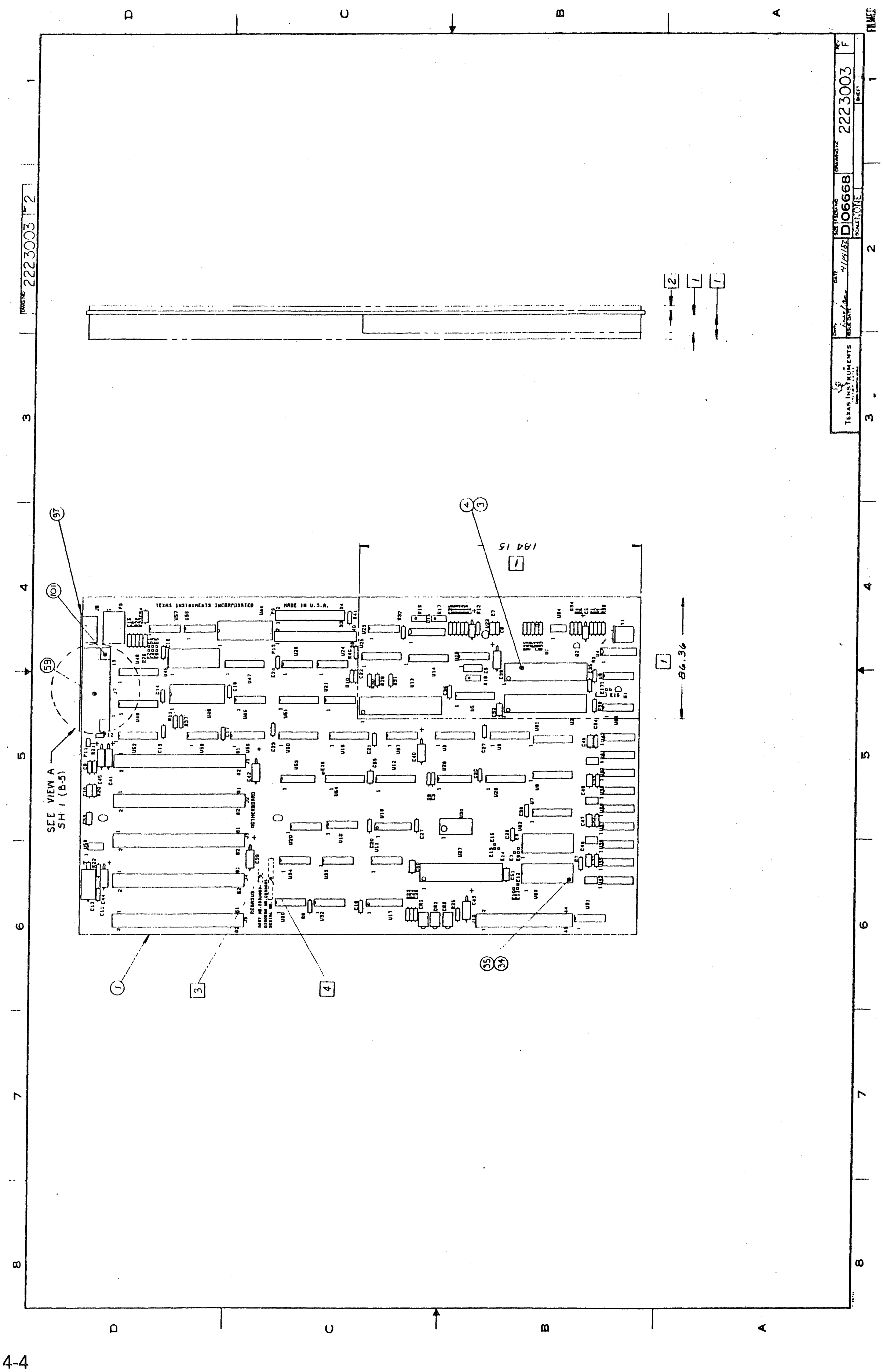
SIZE D | FSCM NO 06668 | DRAWING NO 2223003

SCALE NONE | SHEET 1 OF 2

320

LM

FILMED



TEXAS INSTRUMENTS
2223003
D 06668
SEE VIEW A
SH 1 (B-5)
86.36

LIST OF MATERIALS

11/24/82

PART NUMBER REV DESCRIPTION..........................
2223003-0001 F MOTHERBOARD - PEGASUS

ITEM.	QUANTITY.	COMPONENT..	DESCRIPTION..........................	UM
0002	00001.000	2223005-0001	DIAGRAM,LOGIC, MOTHERBOARD	EA
0004	00002.000	2210188-0018	SOCKET,DIP,40-PINS,LOW PROFILE SEE T -I DRAWING	EA
0004A			XU1,XU2 SEE T -I DRAWING	
0009	00001.000	2210835-0010	CRYSTAL,15.00 MHZ,HC-18/U MOD CASE SEE TI- DRAWING	EA
0009A			Y1 SEE TI- DRAWING	
0013	00005.000	2211342-0016	CONN,CARD-EDGE,31 DUAL POS,NO EARS SEE TI- DRAWING	EA
0013A			J1,J2,J3,J4,J5 SEE TI- DRAWING	
0014	00001.000	0996166-0005	HEADER,SOCKET,SHORT SOLDER T 6 CIRCUITS AMP - 350827-1	EA
0014A			P6 AMP - 350827-1	
0024	00001.000	2210704-0001	IC,LS280,9-BIT ODD/EVEN PARITY GEN/CHK V-LIST-LS280 BURN-IN	EA
0024A			U31 V-LIST-LS280 BURN-IN	
0026	00001.000	2210293-0003	DELAY MODULE, TAPPED, 3NS RISE TIME MAX SEE TI- DRAWING	EA
0026A			U30 SEE TI- DRAWING	
0032	00001.000	2211342-0015	CONN,CARD-EDGE,22 DUAL POS,NO EARS SEE TI- DRAWING	EA
0032A			J10 SEE TI- DRAWING	
0035	00002.000	2210188-0016	SOCKET,DIP,24-PIN,LOW PROFILE SEE T -I DRAWING	EA
0035A			XU62,XU63 SEE T -I DRAWING	
0041	00001.000	0996151-0005	HEADER,17 PINS PER ROW,STRAIGHT,DBL ROW 5935-0900-000	EA
0041A			P9, 5935-0900-000	
0042	00001.000	0996151-0002	HEADER,20 PINS,STRAIGHT,DOUBLE ROW 22526--65611-140	EA
0042A			P13 22526--65611-140	
0043	00001.000	0996151-0008	HEADER,PIN,3 PINS, STR. DOUBLE ROW 022526-65611-106	EA
0043A			E1-E6 022526-65611-106	
0044	00004.000	2211348-0002	HEADER,1-ROW 2-POS,100 CENTER GOLD SEE TI- DRAWING	EA
0044A			E17-E18,E19-E20,J11-J12 SEE TI- DRAWING	
0054	00001.000	2211079-0006	IC, +5 VOLT REGULATOR,BURN-IN SEE TI- DRAWING	EA
0054A			U22 SEE TI- DRAWING	
0058	00001.000	2220495-0001	CONN,PCB-MTG,5 FEMALE CONTACTS,RT ANGLE SEE TI- DRAWING	EA
0058A			J8 SEE TI- DRAWING	
0059	00001.000	2220488-0003	CONNECTOR,RECEPTACLE,PCB,25-PINS SEE TI- DRAWING	EA
0059A			J7 SEE TI- DRAWING	
0060	00001.000	0972537-0003	DIODE,LED RED RT ANGLE 072619-550-0406	EA

LIST OF MATERIALS

11/24/82

PART NUMBER REV DESCRIPTION..........................
2223003-0001 F MOTHERBOARD - PEGASUS

ITEM.	QUANTITY.	COMPONENT..	DESCRIPTION..........................	UM
0060A			CR1 072619-550-0406	
0078	00001.000	0972227-0014	RESISTOR, 100K VARIABLE-CERMET ELEMENT 032997-3292W-1-104	EA
0078A			R17 032997-3292W-1-104	
0079	00001.000	0972227-0013	RESISTOR, 50000 OHM, 22-TURN TRIMMER SEE TI- DRAWING	EA
0079A			R18 SEE TI- DRAWING	
0080	00001.000	0972227-0009	RES,VAR, 5000 OHMS,1/2 WATT, CERMET 032997-3292W-1-502	EA
0080A			R19 032997-3292W-1-502	
0083	00001.000	0972927-0025	CAPACITOR,82PF 500V 5% FIX,MICA DIELECTR MIL -CMR05E820-JOD	EA
0083A			C5 MIL -CMR05E820-JOD	
0089	00001.000	0972763-0021	CAP.,FIXED,AXIAL LEAD,.047 UF,+80%,-20% 1632-0000-000	EA
0089A			C11 1632-0000-000	
0090	00001.000	2211700-0002	CAP, 220UF, 6.3V, 20% SEE TI- DRAWING	EA
0090A			C12 SEE TI- DRAWING	
0093	00002.000	2211878-0002	TRANS,MPS6602,NPN,COMPLEMENTRY DRIVER SEE TI- DRAWING	EA
0093A			Q1,Q2 SEE TI- DRAWING	
0095	00002.000	0532348-0401	STUD, EXTENSION-CRES	EA
0096	00002.000	0972446-0013	RIVET,.116 DIA 5/16 LG DOME HD ALUM - - -000	EA
0097	00001.000	2223036-0001	PLATE,KEYBOARD PLUG 1678-3036-009	EA
0098	00004.000	0972487-0001	JUMPER PLUG,CONNECTOR BLACK 5935-0900-000	EA
0100	00001.000	0994396-0001	PROC., SITE/DATE CODE AND SERIALIZATION	EA
0101	00001.000	0235728-0125	TAB TIN PL BRS STUD DIA.130X.032 THK AMP -42822-2	EA
0103	00001.000	0972537-0004	LED,YELLOW,RT ANG PCB MTG,2.3V,5.0VR SEE TI- DRAWING	EA
0103A			CR2 SEE TI- DRAWING	
0104	00001.000	0972537-0002	DIODE,LED GREEN RT ANGLE 072619-550-0206	EA
0104A			CR3 072619-550-0206	
0110	REF	2223270-0001	SPECIFICATION,UNIT TEST-MOTHERBOARD	EA
0111	00001.000	0411100-0070	LOCKWASHER #4 INTERNAL TOOTH CRES QPL - MS35333-70	EA
0113	AR	0411435-0408	TAPE,INSULATION,ELECT.1/4 IN MMM - 56-1/4	RL
0114	AR	0415804-0005	SEALING COMPOUND,ANAEROBIC-BLUE GRADE C	OT
0999	00001.000	2223003-5001	MOTHERBOARD - PEGASUS - AUTO INSERT 1254-3004-005	EA
9999	00001.000	0239999-9999	COST, SHRINKAGE	EA

LIST OF MATERIALS

11/24/82

PART NUMBER	REV	DESCRIPTION............................
2223003-5001	F	MOTHERBOARD - PEGASUS - AUTO INSERT

ITEM.	QUANTITY.	COMPONENT..	DESCRIPTION............................	UM
0001	00001.000	2223004-0001	PWB MOTHERBOARD 1669- -000	EA
0003	00001.000	2220419-0001	IC,MICROPROCESSOR,CPU SEE TI- DRAWING	EA
0003A			U1 SEE TI- DRAWING	
0005	00001.000	2220424-0001	IC,MICROPROCESSOR BUS CONTROLLER SEE TI- DRAWING	EA
0005A			U3 SEE TI- DRAWING	
0006	00001.000	2220414-0001	IC,TTL,CLOCK GENERATOR AND DRIVER SEE TI- DRAWING	EA
0006A			U4 SEE TI- DRAWING	
0007	00003.000	2210720-0001	IC,LS373,OCTAL D-TYPE LATCHES V-LIST-LS373 BURN-IN	EA
0007A			U5,U6,U7 V-LIST-LS373 BURN-IN	
0008	00004.000	2210702-0001	IC,LS273,OCTAL,D-FLIP-FLOP W/COM CLOCK V-LIST-LS273 BURN-IN	EA
0008A			U47,U49,U50,U51 V-LIST-LS273 BURN-IN	
0010	00001.000	2220435-0001	IC,PROGRAMMABLE INTERRUPT CONTROLLER SEE TI- DRAWING	EA
0010A			U46 SEE TI- DRAWING	
0011	00001.000	2220412-0001	IC,USART,PROG.COMMUNICATION INTERFERENCE SEE TI- DRAWING	EA
0011A			U44 SEE TI- DRAWING	
0012	00001.000	2220626-0001	IC,MOS,16-BIT PRGMBL INTERVAL TIMER SEE TI- DRAWING	EA
0012A			U45 SEE TI- DRAWING	
0015	00001.000	2210653-0001	IC,LS138,3-TO-8 LINE DECODER V-LIST-LS138 BURN-IN	EA
0015A			U55 V-LIST-LS138 BURN-IN	
0016	00001.000	2210654-0001	IC,LS139,DUAL 2-TO-4 LINE DECODER V-LIST-LS139 BURN-IN	EA
0016A			U53 V-LIST-LS139 BURN-IN	
0017	00001.000	2223052-0002	ROM,SYSTEM DECODE HAL12L6	EA
0017A			U54	
0018	00000.000	2211984-0007	IC,DMPAL12L6NC SEE TI- DRAWING	EA
0018A			*U54,ALTERNATE FOR ITEM 17 SEE TI- DRAWING	
0019	00001.000	2211102-0001	IC,F4071BPCQR,QUAD,2-INPUT,4071-BURN-IN SEE TI-DRAWING	EA
0019A			U58 SEE TI-DRAWING	
0020	00002.000	0972141-0057	NETWORK,RES. 4.7 K OHM 2 % 14 PIN DIP BEC - 899-1-R4.7K	EA
0020A			U60,U66 BEC - 899-1-R4.7K	
0021	00001.000	2220445-0001	IC,DYNAMIC MEMORY CONTROLLER SEE TI- DRAWING	EA
0021A			U27 SEE TI- DRAWING	
0022	00001.000	2223053-0001	ROM,MEMORY CONTROL, HAL16R4A 1669- -000	EA

LIST OF MATERIALS

11/24/82

PART NUMBER   REV   DESCRIPTION...........................
2223003-5001   F   MOTHERBOARD - PEGASUS - AUTO INSERT

ITEM.	QUANTITY.	COMPONENT..	DESCRIPTION............................	UM
0022A			U28	
			1669- -000	
0023	00000.000	2211984-0011	IC,DMPAL16R4NC	EA
			SEE TI- DRAWING	
0023A			*U28,ALTERNATE FOR ITEM 22	
			SEE TI- DRAWING	
0025	00001.000	2210689-0001	IC,LS221,DUAL ONE-SHOT	EA
			V-LIST-LS221 BURN-IN	
0025A			U29	
			V-LIST-LS221 BURN-IN	
0027	00001.000	2210608-0001	IC,LS10,TRIPLE,3-INPUT NAND	EA
			V-LIST-LS10 BURN-IN	
0027A			U9	
			V-LIST-LS10 BURN-IN	
0028	00001.000	2210614-0001	IC,LS20,DUAL,4-INPUT NAND	EA
			V-LIST-LS20 BURN-IN	
0028A			U32	
			V-LIST-LS20 BURN-IN	
0029	00001.000	2210621-0001	IC,LS32,QUAD,2-INPUT OR	EA
			V-LIST-LS32 BURN-IN	
0029A			U34	
			V-LIST-LS32 BURN-IN	
0030	00003.000	2210631-0001	IC,LS74,DUAL D FLIP-FLOP W/PSET & CLR	EA
			V-LIST-LS74 BURN-IN	
0030A			U21,U33,U65	
			V-LIST-LS74 BURN-IN	
0031	00009.000	2211118-0005	IC,64K X 1-BIT RAM,350 NSEC,READ CY TIME	EA
0031A			U35,U36,U37,U38,U39	
0031B			U40,U41,U42,U43	
0033	00004.000	2210695-0001	IC,LS245,OCTAL BUS,XCIVER,3ST.OUTPUT	EA
			V-LIST-LS245 BURN-IN	
0033A			U8,U12,U52,U61	
			V-LIST-LS245 BURN-IN	
0034	00001.000	2223064-0001	ROM,SYSTEMS	EA
			SEE TI- DRAWING	
0034A			U63	
			SEE TI- DRAWING	
0036	00001.000	2220415-0001	IC,FLOPPY DISK CONTROLLER,PLASTIC	EA
			SEE TI- DRAWING	
0036A			U13	
			SEE TI- DRAWING	
0037	00001.000	2220421-0001	IC,FLOPPY DISK SUPPORT LOGIC	EA
			- - -000	
0037A			U14	
			- - -000	
0038	00001.000	2220418-0001	IC,FOUR PHASE CLOCK GENERATOR	EA
			SEE TI- DRAWING	
0038A			U15	
			SEE TI- DRAWING	
0039	00001.000	2223054-0002	ROM FLOPPY SYSTEM CONTROL	EA
0039A			U19	
0040	00000.000	2211984-0006	IC,BLANK PROGRAMMABLE ARRAY OF GATES	EA
			SEE TI- DRAWING	
0040A			*U19,ALTER FOR ITEM 39	
			SEE TI- DRAWING	
0045	00001.000	2211771-0001	IC,SN74LS628N,EXTERNAL, TEMPERATURE COMP	EA
			SEE TI- DRAWING	
0045A			U16	
			SEE TI- DRAWING	

LIST OF MATERIALS

11/24/82

PART NUMBER REV DESCRIPTION..........................
2223003-5001 F MOTHERBOARD - PEGASUS - AUTO INSERT

ITEM.	QUANTITY.	COMPONENT..	DESCRIPTION..........................	UM
0046	00002.000	2211126-0001	IC, 1K X 4 BIT STATIC RAM - - -000	EA
0046A			U17,U18 - - -000	
0047	00001.000	0972999-4040	NETWORK 4040 -SEE TI DRAWING	EA
0047A			U11 -SEE TI DRAWING	
0048	00001.000	2210667-0001	IC,LS163,SYNC 4-BIT BINARY CNT,SYNC CLR V-LIST-LS163 BURN-IN	EA
0048A			U20 V-LIST-LS163 BURN-IN	
0049	00002.000	0222222-7416	NETWORK SN7416N -SN7416N	EA
0049A			U23,U24 -SN7416N	
0050	00001.000	2211059-0001	IC,7407N3,HEX/BUF/DVP,BURN-IN SEE TI- DRAWING	EA
0050A			U26 SEE TI- DRAWING	
0051	00002.000	2210694-0001	IC,LS244,OCTAL BUF/LINE DRIVER/RECEIVER V-LIST-LS244 BURN-IN	EA
0051A			U25,U48 V-LIST-LS244 BURN-IN	
0052	00001.000	2210604-0001	IC,LS04,HEX INVERTERS V-LIST-LS04 BURN-IN	EA
0052A			U10 V-LIST-LS04 BURN-IN	
0053	00001.000	2210727-0001	IC,LS393,DUAL,4-BIT BINARY COUNTER V-LIST-LS393 BURN-IN	EA
0053A			U56 V-LIST-LS393 BURN-IN	
0055	00001.000	0222225-2311	NETWORK LM311N,SN72311P SEE - TI DRAWING	EA
0055A			U64 SEE - TI DRAWING	
0056	00001.000	2211349-0001	IC,SN75189AN3, QUAD LINE RECEIVERS SEE TI- DRAWING	EA
0056A			U57 SEE TI- DRAWING	
0057	00001.000	0996304-0001	IC,LM386,AMPL,PWR,AUDIO	EA
0057A			U59	
0061	00004.000	0972946-0089	RES FIX 10K OHM 5% .25 W CARBON FILM 1640-2132-000	EA
0061A			R10,R11,R34,R6 1640-2132-000	
0062	00005.000	0972946-0081	RES FIX 4.7K OHM 5 % .25 W CARBON FILM ROH - R-25	EA
0062A			R7,R8,R40,R41,R37 ROH - R-25	
0063	00003.000	0972946-0085	RES FIX 6.8K OHM 5 % .25 W CARBON FILM ROH - R-25	EA
0063A			R26,R27,R28 ROH - R-25	
0064	00003.000	0972946-0057	RES FIX 470 OHM 5 % .25 W CARBON FILM ROH - R-25	EA
0064A			R23,R24,R25 ROH - R-25	
0065	00004.000	0972946-0065	RES FIX 1.0K OHM 5% .25 W CARBON FILM ROH - R-25	EA
0065A			R4,R21,R38,R45 ROH - R-25	

11/24/82

PART NUMBER   REV   DESCRIPTION............................
2223003-5001   F   MOTHERBOARD - PEGASUS - AUTO INSERT

ITEM.	QUANTITY.	COMPONENT..	DESCRIPTION............................	UM
0067	00001.000	0972946-0017	RES FIX 10.0 OHM 5 % .25 W.CARBON FILM ROH - R-25	EA
0067A			R22 ROH - R-25	
0068	00003.000	0972946-0105	RES FIX 47 K OHM 5 % .25 W CARBON FILM ROH - R-25	EA
0068A			R13,R14,R20 ROH - R-25	
0069	00001.000	0972946-0037	RES FIX 68.0 OHM 5 % .25 W.CARBON FILM ROH - R-25	EA
0069A			R15 ROH - R-25	
0070	00003.000	0972946-0072	RES FIX 2.0K OHM 5 % .25 W CARBON FILM ROH - R-25	EA
0070A			R5,R3,R39 ROH - R-25	
0071	00001.000	0539370-0364	RES FIX FILM 604 OHM 1% .25 WATT COR - NA55	EA
0071A			R12 COR - NA55	
0072	00008.000	0972946-0045	RES FIX 150 OHM 5 % .25 W CARBON FILM SEE TI- DRAWING	EA
0072A			R29,R30,R31,R32 SEE TI- DRAWING	
0072B			R43,R44,R46,R47 SEE TI- DRAWING	
0073	00002.000	0972946-0058	RES FIX 510 OHM 5 % .25 W CARBON FILM ROH - R-25	EA
0073A			R1,R2 ROH - R-25	
0074	00001.000	0972946-0049	RES FIX 220 OHM 5 % .25 W CARBON FILM ROH - R-25	EA
0074A			R33 ROH - R-25	
0075	00001.000	0972934-0010	DIODE,1N755A 7.5 V 5% SIL VOLT REG QPL - 1N755A	EA
0075A			CR4 QPL - 1N755A	
0076	00001.000	0539370-0465	RES FIX FILM 6.81K OHM 1% .25 WATT COR - NA55	EA
0076A			R35 COR - NA55	
0077	00001.000	0539370-0441	RES FIX FILM 3.83K OHM 1% .25 WATT COR - NA55	EA
0077A			R36 COR - NA55	
0081	00001.000	0972757-0019	CAP,FIXED CER 3300PF 10% 50V	EA
0081A			C3	
0082	00001.000	0418356-2353	CAP FIX 0.68 MF 50V 10% TANTALUM SOLID QPL -M39003/1-2353	EA
0082A			C4 QPL -M39003/1-2353	
0084	00001.000	0972757-0009	CAP FIX CER 470PF 10% 50V	EA
0084A			C8	
0085	00001.000	0972924-0071	CAP FIX TANT SOLID 1.0 MFD 10 % 50 VOLT QPL -M39003/1-2356	EA
0085A			C2 QPL -M39003/1-2356	
0086	00001.000	0972757-0043	CAPACITOR,15PF,10%,50WVDC,CERAMIC SEE TI- DRAWING	EA

LIST OF MATERIALS

11/24/82

PART NUMBER	REV	DESCRIPTION..........................
2223003-5001	F	MOTHERBOARD - PEGASUS - AUTO INSERT

ITEM.	QUANTITY.	COMPONENT..	DESCRIPTION...........................	UM
0086A			C1 SEE TI- DRAWING	
0087	00001.000	0972763-0013	CAP,FIXED .010UF 50 VOLTS 004222-MC105E103Z	EA
0087A			C9 004222-MC105E103Z	
0088	00012.000	0972763-0025	CAPACITOR,.10UF 50V FX,CERAMIC DIEL COR CA-C03Z5U104Z050A	EA
0088A			C7,C10,C46,C47,C48,C49 COR CA-C03Z5U104Z050A	
0088B			C50,C51,C52,C53,C54,C55 COR CA-C03Z5U104Z050A	
0091	00027.000	0972763-0001	CAPACITOR,.001UF 50V FX CERAMIC DIEL COR CA-C02Z5U102Z100A	EA
0091A			C6,C13,C14,C15,C16,C17,C18 COR CA-C02Z5U102Z100A	
0091B			C19,C20,C21,C22,C23,C24,C25 COR CA-C02Z5U102Z100A	
0091C			C26,C27,C28,C29,C30,C31,C32 COR CA-C02Z5U102Z100A	
0091D			C33,C34,C35,C36,C37,C57 COR CA-C02Z5U102Z100A	
0092	00008.000	0972924-0018	CAP FIX TANT SOLID 6.8 MFD 10 % 35 VOLT QPL -M39003/1-2304	EA
0092A			C38,C39,C40,C41,C42, QPL -M39003/1-2304	
0092B			C43,C44,C45 QPL -M39003/1-2304	
0102	00001.000	2210600-0001	IC,LS00,QUAD,2-INPUT NAND V-LIST-LS00 BURN-IN	EA
0102A			U67 V-LIST-LS00 BURN-IN	
0105	00001.000	0972946-0035	RES FIX 56.0 OHM 5 % .25 W.CARBON FILM ROH - R-25	EA
0105A			R42 ROH - R-25	
0106	00002.000	0972757-0001	CAP,FIXED CERAMIC 100 PF 10% 50V UC -C51C101K	EA
0106A			C56,C58 UC -C51C101K	
0107	00001.000	0972946-0083	RES FIX 5.6K OHM 5 % .25 W CARBON FILM ROH - R-25	EA
0107A			R48 ROH - R-25	
0108	00001.000	0972946-0047	RES FIX 180 OHM 5 % .25 W CARBON FILM ROH - R-25	EA
0108A			R16 ROH - R-25	
0109	00001.000	0972934-0011	DIODE,1N756A 8.2 V 5% SIL VOLT REG QPL - 1N756A	EA
0109A			CR5 QPL - 1N756A	
0112	00001.000	0972946-0093	RES FIX 15K OHM 5% .25 W CARBON FILM ROH - R-25	EA
0112A			R9 ROH - R-25	

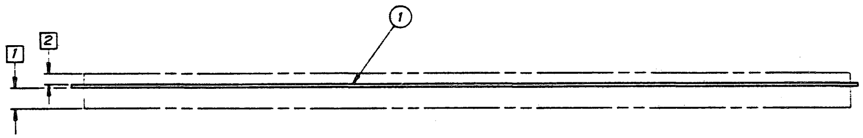
DWG NO 2223009 SH 1

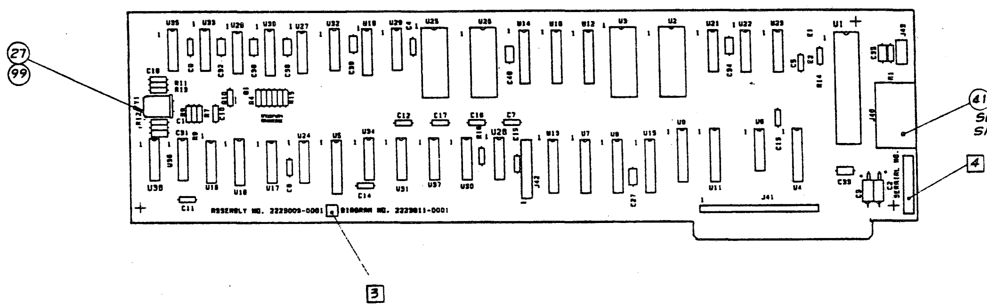
NOTES UNLESS OTHERWISE SPECIFIED

1. MAXIMUM COMPONENT HIEGHT ABOVE THE BOARD SHALL BE 9.6
2. MAXIMUM LEAD LENGTH BELOW THE BOARD SHALL BE 2.3
3. MARK APPROPRIATE REVISION LETTER IN SPACE INDICATED PER PROCESS 3
4. MARK SITE DATE CODE AND SERIAL NUMBER IN SPACE INDICATED PER ITEM 44 PARAGRAPH 4.0 AND PROCESS 3
5. CAUTION THIS ASSEMBLY AND CERTAIN COMPONENTS ARE SUSCEPTIBLE TO DAMAGE FROM ELECTROSTATIC DISCHARGE. OPERATOR AND EQUIPMENT GROUND AND PACKAGE IS REQUIRED. STATIC SENSITIVE COMPONETS ARE: U1 THRU U39, Y1 AND Q1
6. WIRE FROM VIDEO JACK (ITEM 56) MUST ATTACH TO J43 AS SHOWN PER PROCESS 2
7. AS FOLLOWS: REWORK REQUIRED FOR PWB REV "*" AND "A" AND "B"
   - (7A) ON COMPONENT SIDE, CUT ETCH CONNECTED TO U18 PIN 1
   - (7B) ON CONDUCTOR SIDE, CUT ETCH CONNECTED TO U27 PIN 1
   - (7C) USING (ITEM 95) JUMPER U17 PIN 1 TO THE FEED THRU BELOW U27 AS SHOWN (COMP SIDE)
   - (7D) USING (ITEM 95) JUMPER FROM U27 PIN 1 TO U39 PIN 1 (COMP SIDE)
   - (7E) PLACE MYLAR TAPE (ITEM 96) ON BOTTOM OF CONNECTOR (ITEM 41) AS INDICATED
   - (7F) MARK NUMBER "2" IN SPACE INDICATED PER PROCESS 3 (COND SIDE)
   - (7G) ADD JUMPER ITEM 95 FROM C2 TO J41 PINS 19 AND 20 AS INDICATED (COND SIDE)
   - (7H) MARK PWB WITH (C) ON CONDUCTOR SIDE NEXT TO BOARD NUMBER
8. BEND TAB (ITEM 56) 45° FROM OPTION PLATE (ITEM 53) AS INDICATED
9. REMOVE AND DISCARD 2 PANHEAD SCREWS INSTALLED IN (ITEM 41) PRIOR TO ASSEMBLY
10. APPLY ITEM 103 (LOCTITE C SEALANT) TO TO ENDS OF ITEM 100 (STUDDED STANDOFF) AFTER INSTALLATION

REVISIONS

REV	DESCRIPTION	DATE	APPROVED
E	CN 502186 (B) E. DeLeon	10-14-82	M B...
	(1) REDRAWN PER ENGINEERING REDESIGN		
F	CN 502178 (D) ...	11-4-82	M B...
	(1) LM UPDATE (2) SH 2 Z B-2 DELE IT'S 54 & 55 (3) FID ADD NOTE 7H		





PART NUMBER	DESCRIPTION
2223009-5001	ALPHA CRT CONTROLLER, AUTO INSERT
2223009-0001	ALPHA CRT CONTROLLER

-3 QTY	-2 QTY	-1 QTY	ITEM NO	PART OR IDENTIFYING NUMBER	NOMENCLATURE OR DESCRIPTION	NOTES

PARTS LIST

SEQ NO	IDENT	F-SPEC	NO	ADDITIONAL	NOTES
3	MARK	902-01	01	HGT 1.5 CLR WHT (CAT 4)	3 4
2	SLDR	127-01	00	HAND SOLDER	6
1	SLDR	124-02	00	WAVE SOLDER	
	PROCESS		CLASSIFICATION		

PROCESSES — FOR CORRELATION TO GOVT/IND SPECIFICATIONS, SEE TI DRAWING 729487

REV LEVEL BLOCK				
TEST SPEC	2223271-0001	*	*	
ASSEMBLY	2223009-0001	E	F	
PWB	2223010-0001	B	B	
DIAGRAM	2223011-0001	A	A	

REV STATUS OF SHEETS

REV	F	F
SH	1	2

NEXT ASSY	USED ON
2223051	8755
2223050	8755

APPLICATION

UNLESS OTHERWISE SPECIFIED
- DIMENSIONS ARE IN MILLIMETERS
- TOLERANCES 2 PLACE DECIMALS ±0.25, 1 PLACE DECIMALS ±0.5, ANGLES ±1°
- INTERPRET DRAWING PER DOD-D-1000
- REMOVE ALL BURRS AND SHARP EDGES
- CONCENTRICITY MACHINED DIAMETERS 0.25 FIM
- DIMENSIONAL LIMITS APPLY BEFORE PROCESSES
- PARENTHETICAL INFO FOR REF ONLY

HOLE TOLERANCE	
0.33 THRU 3.18	+0.10/-0.03
3.20 THRU 6.35	+0.13/-0.03
6.38 THRU 12.70	+0.18/-0.03
12.73 THRU 19.05	+0.20/-0.03
19.08 THRU 25.40	+0.25/-0.03
25.43 THRU 50.80	+0.30/-0.03

THIRD ANGLE PROJECTION

DWN Emilio ... 9-24-82
CHK MIKE BIEHL 6-8-82
ENGR GRAY McCORD 6-10-82
APVD ENGR GRAY McCORD 6-10-82
QA RALPH J. CAPAN 6-10-82
APVD MFG SUSAN BRIDGEN 6-10-82
RLSE MIKE R. WOLF 6-11-82

TEXAS INSTRUMENTS INCORPORATED

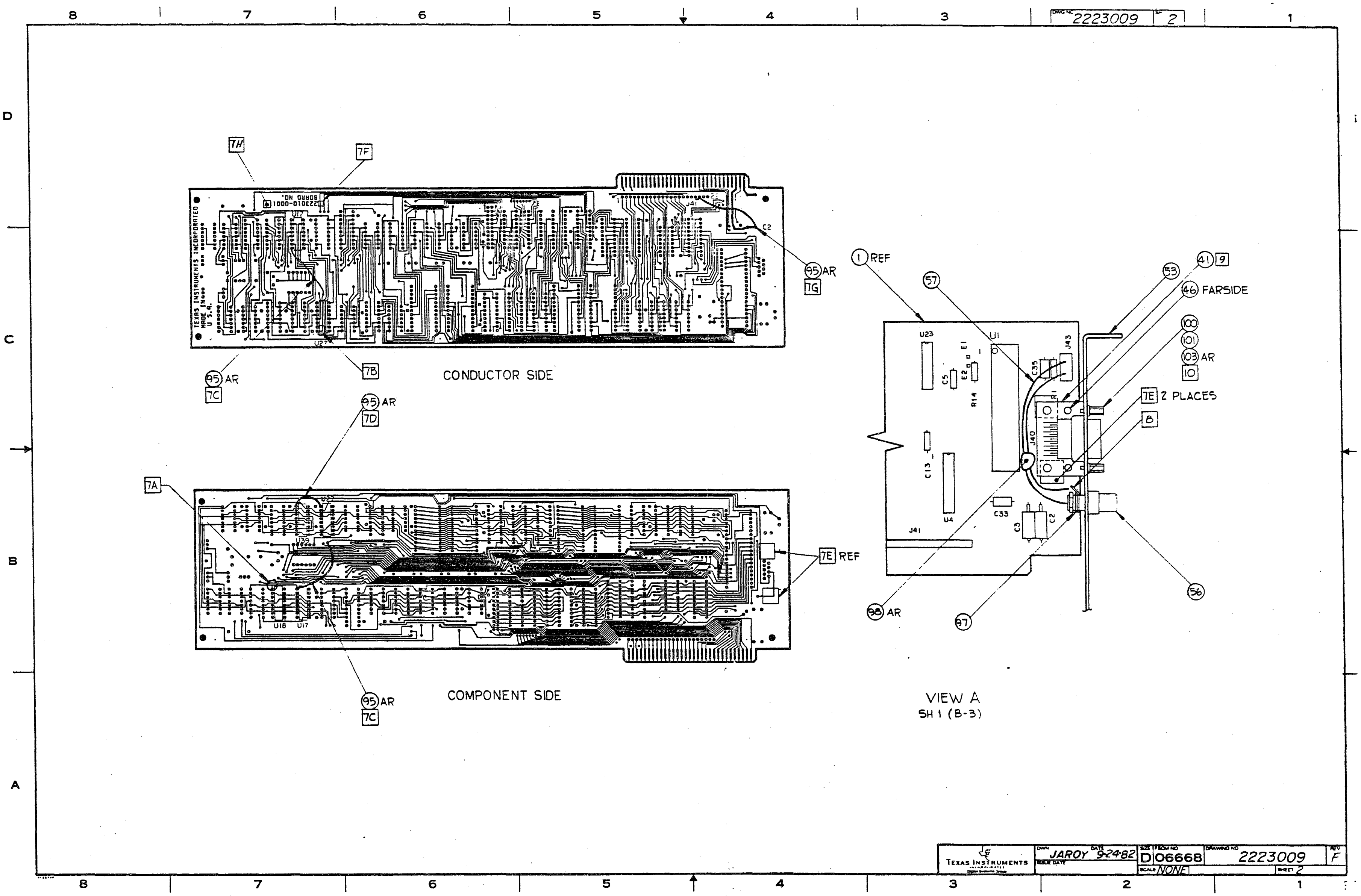
SI-METRIC

ALPHA CRT CONTROLLER

SIZE D FSCM NO 06668 DRAWING NO 2223009

SCALE NONE SHEET 1 OF 2

LM / CC



2223009
2
7H
7F
95 AR
7G
95 AR
7C
7B
CONDUCTOR SIDE
95 AR
7D
7A
7E REF
95 AR
7C
COMPONENT SIDE
1 REF
57
53
41 9
46 FARSIDE
100
101
103 AR
10
7E 2 PLACES
8
56
98 AR
97
VIEW A
SH 1 (B-3)
TEXAS INSTRUMENTS
JAROY 9-24-82
D 06668
2223009
F
NONE
2

LIST OF MATERIALS

11/24/82

PART NUMBER	REV	DESCRIPTION..........................
2223009-0001	F	ALPHA CRT CONTROLLER

ITEM.	QUANTITY.	COMPONENT..	DESCRIPTION..........................	UM
0002	REF	2223011-0001	LOGIC,DIAGRAM,ALPHA CRT CONTROLLER	EA
0025	00001.000	2210727-0001	IC,LS393,DUAL,4-BIT BINARY COUNTER V-LIST-LS393 BURN-IN	EA
0025A			U38 V-LIST-LS393 BURN-IN	
0027	00001.000	2210835-0004	CRYSTAL,18 MHZ,HC-18/U WITH GND LEAD SEE TI- DRAWING	EA
0027A			Y1 SEE TI- DRAWING	
0028	00001.000	2211878-0002	TRANS,MPS6602,NPN,COMPLEMENTRY DRIVER SEE TI- DRAWING	EA
0028A			Q1 SEE TI- DRAWING	
0038	00002.000	0418356-2305	CAPACITOR,TANTALUM,6.8UF,20?%,35V SEE TI- DRAWING	EA
0038A			C2,C3 SEE TI- DRAWING	
0041	00001.000	2220488-0001	CONNECTOR,RECEPTACLE,PCB,9-PINS SEE TI- DRAWING	EA
0041A			J40 SEE TI- DRAWING	
0043	00001.000	2210970-0005	CONN. 22-POS.,PC BD, SINGLE ROW,.100 CNT SEE TI- DRAWING	EA
0043A			J41 SEE TI- DRAWING	
0044	REF	0994396-9901	PROCEDURE,SITE & DATE CODE SERIALIZATION	EA
0045	00001.000	2211047-0002	CONNECTOR,RECEPTACLE,2-ROW,11-POSITION SEE TI- DRAWING	EA
0045A			J42 SEE TI- DRAWING	
0046	00002.000	0972446-0012	RIVET,.116 DIA 3/16 LG DOME HD ALUM -75021-0406	EA
0053	00001.000	2223033-0003	PLATE,OPTION BOARD,9-POSITION 1678-3333-007	EA
0056	00001.000	2220629-0001	AUDIO JACK,PANEL MNTNG,ROUND BASE,.185" 1254- -000	EA
0057	AR	0935172-3488	WIRE,UL 1430/3317,22AWG,GRA/YEL 1650-0000-000	FT
0095	AR	0996563-0001	WIRE,30AWG SOLID,KYNAR,INSULATED,BROWN 071124-BR212/1-30-	FT
0096	AR	0411435-0408	TAPE,INSULATION,ELECT.1/4 IN MMM - 56-1/4	RL
0097	00001.000	0411100-0074	LOCKWASHER 1/4 INTERNAL TOOTH CRES QPL - MS35333-74	EA
0098	AR	0996069-0003	ADH,SOLID,THRMPLSTC 25# BAG ANAEROBIC PERSAL-917-6302	EA
0099	00001.000	2211540-0001	FOAM, .35X.50X.05,POLY, ADHESIVE BACKED SEE TI- DRAWING	EA
0100	00002.000	0532348-0401	STUD, EXTENSION-CRES	EA
0101	00002.000	0411100-0070	LOCKWASHER #4 INTERNAL TOOTH CRES QPL - MS35333-70	EA
0102	REF	2223271-0001	SPECIFICATION,UNIT TEST-ALPHA CRT	EA
0103	AR	0415804-0005	SEALING COMPOUND,ANAEROBIC-BLUE GRADE C	QT
0999	00001.000	2223009-5001	ALPHA CRT CONTROLLER - AUTO INSERT 1254-3010-005	EA
9999	00000.750	0239999-9999	COST, SHRINKAGE	EA

LIST OF MATERIALS

11/24/82

PART NUMBER	REV	DESCRIPTION............................
2223009-5001	F	ALPHA CRT CONTROLLER - AUTO INSERT

ITEM.	QUANTITY.	COMPONENT..	DESCRIPTION............................	UM
0001	00001.000	2223010-0001	PWB,ALPHA CRT CONTROLLER 1669- -000	EA
0003	00001.000	2220443-0002	IC,CRT CONTROLLER,2 MHZ CLOCK RATE SEE TI- DRAWING	EA
0003A			U1 SEE TI- DRAWING	
0004	00002.000	0996952-0005	IC,2K X 8-BIT STATIC RAM,150NS,PLASTIC SEE TI- DRAWING	EA
0004A			U2,U3 SEE TI- DRAWING	
0005	00001.000	2223060-0001	LOGIC ARRAY,HAL10L8 1669- -000	EA
0005A			U4 1669- -000	
0006	00001.000	2223058-0001	LOGIC ARRAY,HAL16R8 1669- -000	EA
0006A			U5 1669- -000	
0007	00001.000	2210660-0001	IC,LS155,DUAL 2-LINE TO 4-LINE DECODER V-LIST-LS155 BURN-IN	EA
0007A			U6 V-LIST-LS155 BURN-IN	
0008	00003.000	2210695-0001	IC,LS245,OCTAL BUS,XCIVER,3ST.OUTPUT V-LIST-LS245 BURN-IN	EA
0008A			U7,U8,U9 V-LIST-LS245 BURN-IN	
0009	00004.000	2210721-0001	IC,LS374,OCTAL D-TYPE FLIP-FLOP V-LIST-LS374 BURN-IN	EA
0009A			U10,U11,U14,U15 V-LIST-LS374 BURN-IN	
0010	00003.000	2210764-0001	IC,S175,QUAD,F/F,DOUBLE RAIL OUTPUT V-LIST-S175 BURN-IN	EA
0010A			U16,U17,U27 V-LIST-S175 BURN-IN	
0011	00002.000	2210694-0001	IC,LS244,OCTAL BUF/LINE DRIVER/RECEIVER V-LIST-LS244 BURN-IN	EA
0011A			U12,U13 V-LIST-LS244 BURN-IN	
0012	00001.000	2210669-0001	IC,LS166,8-BIT PARALLEL/SERIAL INPUT V-LIST-LS166 BURN-IN	EA
0012A			U19 V-LIST-LS166 BURN-IN	
0013	00003.000	2210662-0001	IC,LS157,QUAD 2-LINE TO 1-LINE DATA SELE V-LIST-LS157 BURN-IN	EA
0013A			U21,U22,U23 V-LIST-LS157 BURN-IN	
0014	00001.000	2210761-0001	IC,S163,SYNCHRONOUS 4-BIT COUNTER V-LIST-S163 BURN-IN	EA
0014A			U24 V-LIST-S163 BURN-IN	
0015	00001.000	2223065-0001	ROM,CHARACTER GENERATOR - - -000	EA
0015A			U25 - - -000	
0016	00002.000	2210631-0001	IC,LS74,DUAL D FLIP-FLOP W/PSET & CLR V-LIST-LS74 BURN-IN	EA
0016A			U28,U29 V-LIST-LS74 BURN-IN	
0017	00001.000	2210649-0001	IC,LS125,QUAD BUS BUFFER W/3-STATE OUTPU V-LIST-LS125 BURN-IN	EA
0017A			U30 V-LIST-LS125 BURN-IN	
0018	00001.000	2210614-0001	IC,LS20,DUAL,4-INPUT NAND V-LIST-LS20 BURN-IN	EA

LIST OF MATERIALS

11/24/82

PART NUMBER REV DESCRIPTION.............................
2223009-5001 F ALPHA CRT CONTROLLER - AUTO INSERT

ITEM.	QUANTITY.	COMPONENT..	DESCRIPTION..............................	UM
0018A			U31 V-LIST-LS20 BURN-IN	
0019	00001.000	2210749-0001	IC,S86,QUAD,2-INPUT EXCLUSIVE OR V-LIST-S86 BURN-IN	EA
0019A			U32 V-LIST-S86 BURN-IN	
0020	00001.000	2210740-0001	IC,S10,TRIPLE,3-INPUT POSITIVE AND V-LIST-S10 BURN-IN	EA
0020A			U33 V-LIST-S10 BURN-IN	
0021	00001.000	2210621-0001	IC,LS32,QUAD,2-INPUT OR V-LIST-LS32 BURN-IN	EA
0021A			U34 V-LIST-LS32 BURN-IN	
0022	00001.000	2210735-0001	IC,S00,QUAD,2-INPUT NAND V-LIST-S00 BURN-IN	EA
0022A			U35 V-LIST-S00 BURN-IN	
0023	00001.000	2210738-0001	IC,S04,HEX INVERTERS V-LIST-S04 BURN-IN	EA
0023A			U36 V-LIST-S04 BURN-IN	
0024	00001.000	2210604-0001	IC,LS04,HEX INVERTERS V-LIST-LS04 BURN-IN	EA
0024A			U37 V-LIST-LS04 BURN-IN	
0029	00001.000	0972946-0041	RES FIX 100 OHM 5 % .25 W CARBON FILM ROH - R-25	EA
0029A			R1 ROH - R-25	
0030	00001.000	0972946-0074	RES FIX 2.4K OHM 5 % .25 W CARBON FILM ROH - R-25	EA
0030A			R2 ROH - R-25	
0031	00002.000	0972946-0066	RES FIX 1.1K OHM 5% .25 W CARBON FILM ROH - R-25	EA
0031A			R3,R10 ROH - R-25	
0032	00001.000	0972946-0091	RES FIX 12 K OHM 5% .25 W CARBON FILM ROH - R-25	EA
0032A			R4 ROH - R-25	
0033	00001.000	0972946-0076	RES FIX 3.0K OHM 5 % .25 W CARBON FILM ROH - R-25	EA
0033A			R5 ROH - R-25	
0034	00001.000	0972946-0084	RES FIX 6.2K OHM 5 % .25 W CARBON FILM ROH - R-25	EA
0034A			R6 ROH - R-25	
0035	00006.000	0972946-0081	RES FIX 4.7K OHM 5 % .25 W CARBON FILM ROH - R-25	EA
0035A			R7,R8,R9,R11,R14,R16 ROH - R-25	
0036	00002.000	0972946-0057	RES FIX 470 OHM 5 % .25 W CARBON FILM ROH - R-25	EA
0036A			R12,R13 ROH - R-25	
0037	00001.000	0972757-0009	CAP FIX CER 470PF 10% 50V	EA
0037A			C1	
0039	00014.000	0972763-0013	CAP,FIXED .010UF 50 VOLTS 004222-MC105E103Z	EA

LIST OF MATERIALS

11/24/82

PART NUMBER	REV	DESCRIPTION............................
2223009-5001	F	ALPHA CRT CONTROLLER - AUTO INSERT

ITEM.	QUANTITY.	COMPONENT..	DESCRIPTION............................	UM
0039A			C4,C5,C7,C8,C9,C10,C11,C12 004222-MC105E103Z	
0039B			C13,C14,C15,C16,C17,C18 004222-MC105E103Z	
0040	00010.000	0972763-0025	CAPACITOR,.10UF 50V FX,CERAMIC DIEL COR CA-C03Z5U104Z050A	EA
0040A			C27,C30,C31,C32,C33,C34 COR CA-C03Z5U104Z050A	
0040B			C35,C38,C39,C40 COR CA-C03Z5U104Z050A	
0047	00002.000	2210763-0001	IC,S174,HEX,FLIP-FLOP,SINGLE RAIL OUTPUT V-LIST-S174 BURN-IN	EA
0047A			U18,U39 V-LIST-S174 BURN-IN	
0049	00001.000	2210759-0001	IC,S157,QUAD,2/1 LINE SELECT/MULTIPLEXER V-LIST-S157 BURN-IN	EA
0049A			U20 V-LIST-S157 BURN-IN	
0050	00001.000	0972946-0079	RES FIX 3.9K OHM 5 % .25 W CARBON FILM ROH - R-25	EA
0050A			R15 ROH - R-25	

NOTES: UNLESS OTHERWISE SPECIFIED

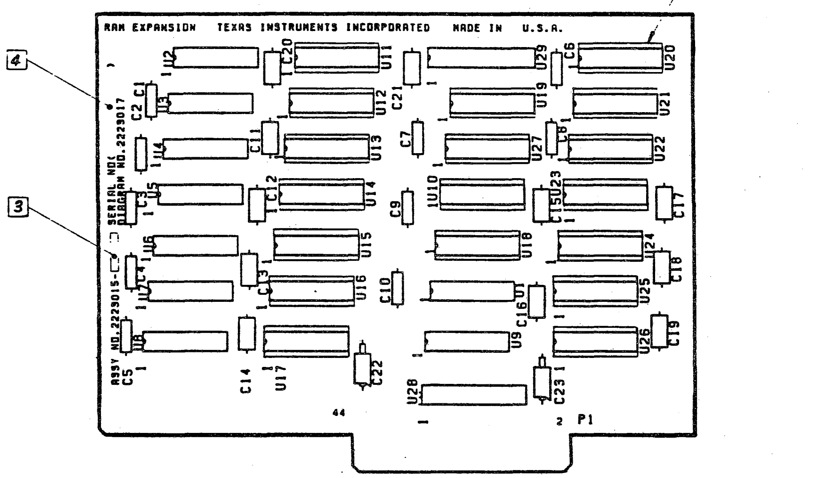
1. MAXIMUM COMPONENT HEIGHT ABOVE THE BOARD SHALL BE 19.00
2. MAXIMUM LEAD LENGTH BELOW THE BOARD SHALL BE 1.78
3. MARK APPROPRIATE DASH NUMBER AND REVISION LETTER IN SPACE INDICATED PER PROCESS 3
4. MARK SITE DATE CODE AND SERIAL NUMBER IN SPACE INDICATED PER ITEM 9 PARAGRAPH 4 AND PROCESS 3
5. CAUTION THIS ASSEMBLY AND CERTAIN COMPONENTS ARE SUSCEPTIBLE TO DAMAGE FROM ELECTROSTATIC DISCHARGE. OPERATOR AND EQUIPMENT GROUND AND PACKAGE IS REQUIRED. STATIC SENSITIVE COMPONENTS ARE : U1 THRU U29
6. ITEM 5 (2211118-0004) INSTALLED IN XU10 THRU XU18 ON DASH 2 ONLY AND XU10 THRU XU27 ON DASH 3 ONLY. (SOCKETS NOT POPULATED ON DASH 1)

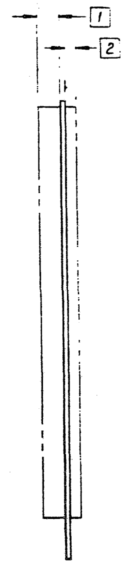
DWG NO 2223015

REVISIONS

REV	DESCRIPTION	DATE	APPROVED
A	LM UPDATE	4-30-82	
B	LM UPDATE	4-30-82	
C	CN 493816 (D) (1) LM IT 4A DES WAS U10 THRU U27	5-27-82	M Bull
D	CN 497899 (D) (1) LM UPDATE	8-3-82	M Bull
E	CN 500532 (D) (1) LM UPDATE (2) ADD NOTE 6	8-11-82	M Bull

(4) 18 PLACES (U10 - U27)





CONVERSION CHART	
mm	INCHES
0.25	.010
0.5	.02
1.5	.06
1.78	.070
19.00	.748

PART NO.	TITLE
2223015 - 5003	EXPANSION RAM (192K) AUTO INSERT
2223015 - 5002	EXPANSION RAM (128K) AUTO INSERT
2223015 - 5001	EXPANSION RAM AUTO INSERT
2223015 - 0003	EXPANSION RAM (192K)
2223015 - 0002	EXPANSION RAM (128K)
2223015 - 0001	EXPANSION RAM

-3 QTY	-2 QTY	-1 QTY	ITEM NO	PART OR IDENTIFYING NUMBER	NOMENCLATURE OR DESCRIPTION	NOTES

PARTS LIST

SEQ NO	IDENT	F-SPEC	NO	ADDITIONAL	NOTES
3	MARK	902-01	01	HGT 1.5 CLR WHT (CAT 4)	3 4
2	SLDR	127-01	00	HAND SOLDER	
1	SLDR	124-02	00	WAVE SOLDER	
	PROCESS		CLASSIFICATION		

PROCESSES — FOR CORRELATION TO GOVT AND SPECIFICATIONS SEE TI DRAWING 7294E

REV STATUS OF SHEETS

		A	B	C	D	E
ASSEMBLY	2223015-0001	A	B	C	D	E
PWB	2223016-0001	*	*	*	*	A
DIAGRAM	2223017-0001	*	*	*	*	*

NEXT ASSY	USED ON
2223051	8755
2223050	8755

APPLICATION

UNLESS OTHERWISE SPECIFIED
- DIMENSIONS ARE IN MILLIMETERS
- TOLERANCES 2 PLACE DECIMALS ±0.25, 1 PLACE DECIMALS ±0.5, ANGLES ±1°
- INTERPRET DRAWING PER DOD-D-1000
- REMOVE ALL BURRS AND SHARP EDGES
- CONCENTRICITY MACHINED DIAMETERS 0.25 FIM
- DIMENSIONAL LIMITS APPLY BEFORE PROCESSES
- PARENTHETICAL INFO FOR REF ONLY

HOLE TOLERANCE	
0.33 THRU 3.18	+0.10/-0.03
3.20 THRU 6.36	+0.13/-0.03
6.38 THRU 12.70	+0.15/-0.03
12.73 THRU 19.05	+0.20/-0.03
19.08 THRU 25.40	+0.25/-0.03
25.43 THRU 50.80	+0.30/-0.03

THIRD ANGLE PROJECTION

DWN 4-19-82
CHK 4-30-82
ENGR 5-4-82
APPROVED 5-4-82
5-4-82
5-4-82
RLSE 5/4/82

TEXAS INSTRUMENTS INCORPORATED

SI-METRIC

RAM EXPANSION

SIZE	FSCM NO	DRAWING NO
D	06668	2223015

SCALE NONE SHEET

LM

LIST OF MATERIALS

11/24/82

PART NUMBER	REV	DESCRIPTION............................
2223015-0001	F	EXPANSION RAM

ITEM.	QUANTITY.	COMPONENT..	DESCRIPTION............................	UM
0002	REF	2223017-0001	SCHEMATIC,EXPANSION RAM	EA
0004	00018.000	2210188-0012	SOCKET,DIP,16-PINS,LOW PROFILE	EA
			SEE T -I DRAWING	
0004A			XU10,XU11,XU12,XU13,XU14	
			SEE T -I DRAWING	
0004B			XU15,XU16,XU17,XU18,XU19	
			SEE T -I DRAWING	
0004C			XU20,XU21,XU22,XU23,XU24	
			SEE T -I DRAWING	
0004D			XU25,XU26,XU27	
			SEE T -I DRAWING	
0006	00001.000	0972763-0001	CAPACITOR,.001UF 50V FX CERAMIC DIEL	EA
			COR CA-C02Z5U102Z100A	
0006A			C5	
			COR CA-C02Z5U102Z100A	
0007	00001.000	0972763-0025	CAPACITOR,.10UF 50V FX,CERAMIC DIEL	EA
			COR CA-C03Z5U104Z050A	
0007A			C19	
			COR CA-C03Z5U104Z050A	
0008	00002.000	0972924-0018	CAP FIX TANT SOLID 6.8 MFD 10 % 35 VOLT	EA
			QPL -M39003/1-2304	
0008A			C22,C23	
			QPL -M39003/1-2304	
0009	REF	0994396-0001	PROC., SITE/DATE CODE AND SERIALIZATION	EA
0101	00001.000	2223015-5001	EXPANSION RAM -AUTO INSERT	EA
			1254-3016-006	
9999	00000.500	0239999-9999	COST, SHRINKAGE	EA

11/24/82

PART NUMBER	REV	DESCRIPTION............................
2223015-0002	E	EXPANSION RAM (128K)

ITEM.	QUANTITY.	COMPONENT..	DESCRIPTION............................	UM
0002	REF	2223017-0001	SCHEMATIC,EXPANSION RAM	EA
0003	00009.000	2211118-0004	IC,64K-BIT DYNAMIC RAM,150NS TA/ROW	EA
			TMS416-4-15NL	
0003A			U10,U11,U12,U13,U14,U15,U16	
			TMS416-4-15NL	
0003B			U17,U18	
			TMS416-4-15NL	
0004	00018.000	2210188-0012	SOCKET,DIP,16-PINS,LOW PROFILE	EA
			SEE T -I DRAWING	
0004A			XU10,XU11,XU12,XU13,XU14	
			SEE T -I DRAWING	
0004B			XU15,XU16,XU17,XU18,XU19	
			SEE T -I DRAWING	
0004C			XU20,XU21,XU22,XU23,XU24	
			SEE T -I DRAWING	
0004D			XU25,XU26,XU27	
			SEE T -I DRAWING	
0006	00001.000	0972763-0001	CAPACITOR,.001UF 50V FX CERAMIC DIEL	EA
			COR CA-C02Z5U102Z100A	
0006A			C5	
			COR CA-C02Z5U102Z100A	
0007	00001.000	0972763-0025	CAPACITOR,.10UF 50V FX,CERAMIC DIEL	EA
			COR CA-C03Z5U104Z050A	

LIST OF MATERIALS

11/24/82

PART NUMBER	REV	DESCRIPTION..........................
2223015-0002	E	EXPANSION RAM (128K)

ITEM.	QUANTITY.	COMPONENT..	DESCRIPTION..........................	UM
0007A			C19 COR CA-C03Z5U104Z050A	
0008	00002.000	0972924-0018	CAP FIX TANT SOLID 6.8 MFD 10 % 35 VOLT QPL -M39003/1-2304	EA
0008A			C22,C23 QPL -M39003/1-2304	
0009	REF	0994396-0001	PROC., SITE/DATE CODE AND SERIALIZATION	EA
0101	00001.000	2223015-5002	EXPANSION RAM (128K)-AUTO INSERT 1254-3018-001	EA
9999	00000.500	0239999-9999	COST, SHRINKAGE	EA

11/24/82

PART NUMBER	REV	DESCRIPTION..........................
2223015-0003	E	EXPANSION RAM (192K)

ITEM.	QUANTITY.	COMPONENT..	DESCRIPTION..........................	UM
0002	REF	2223017-0001	SCHEMATIC,EXPANSION RAM	EA
0003	00018.000	2211118-0004	IC,64K-BIT DYNAMIC RAM,150NS TA/ROW TMS416-4-15NL	EA
0003A			U10,U11,U12,U13,U14,U15,U16 TMS416-4-15NL	
0003B			U17,U18,U19,U20,U21,U22,U23 TMS416-4-15NL	
0003C			U24,U25,U26,U27 TMS416-4-15NL	
0004	00018.000	2210188-0012	SOCKET,DIP,16-PINS,LOW PROFILE SEE T -I DRAWING	EA
0004A			XU10,XU11,XU12,XU13,XU14 SEE T -I DRAWING	
0004B			XU15,XU16,XU17,XU18,XU19 SEE T -I DRAWING	
0004C			XU20,XU21,XU22,XU23,XU24 SEE T -I DRAWING	
0004D			XU25,XU26,XU27 SEE T -I DRAWING	
0006	00001.000	0972763-0001	CAPACITOR,.001UF 50V FX CERAMIC DIEL COR CA-C02Z5U102Z100A	EA
0006A			C5 COR CA-C02Z5U102Z100A	
0007	00001.000	0972763-0025	CAPACITOR,.10UF 50V FX,CERAMIC DIEL COR CA-C03Z5U104Z050A	EA
0007A			C19 COR CA-C03Z5U104Z050A	
0008	00002.000	0972924-0018	CAP FIX TANT SOLID 6.8 MFD 10 % 35 VOLT QPL -M39003/1-2304	EA
0008A			C22,C23 QPL -M39003/1-2304	
0009	REF	0994396-0001	PROC., SITE/DATE CODE AND SERIALIZATION	EA
0101	00001.000	2223015-5003	EXPANSION RAM (192K)-AUTO INSERT 1254-3020-001	EA
9999	00000.500	0239999-9999	COST, SHRINKAGE	EA

LIST OF MATERIALS

11/24/82

PART NUMBER	REV	DESCRIPTION
2223015-5001	E	EXPANSION RAM -AUTO INSERT

ITEM.	QUANTITY.	COMPONENT..	DESCRIPTION	UM
0001	00001.000	2223016-0001	PWB,EXPANSION RAM 1669- -000	EA
0003	00009.000	2211118-0004	IC,64K-BIT DYNAMIC RAM,150NS TA/ROW TMS416-4-15NL	EA
0003A			U1,U2,U3,U4,U5,U6,U7,U8,U9 TMS416-4-15NL	
0005	00002.000	2220360-0002	IC,OCTAL DRAM DRIVER, 3-STATE OUTPUTS SEE TI- DRAWING	EA
0005A			U28,U29 SEE TI- DRAWING	
0006	00009.000	0972763-0001	CAPACITOR,.001UF 50V FX CERAMIC DIEL COR CA-C02Z5U102Z100A	EA
0006A			C1,C2,C3,C4,C6,C7,C8,C9,C10 COR CA-C02Z5U102Z100A	
0007	00011.000	0972763-0025	CAPACITOR,.10UF 50V FX,CERAMIC DIEL COR CA-C03Z5U104Z050A	EA
0007A			C11,C12,C13,C14,C15,C16,C17 COR CA-C03Z5U104Z050A	
0007B			C18,C20,C21 COR CA-C03Z5U104Z050A	

11/24/82

PART NUMBER	REV	DESCRIPTION
2223015-5002	E	EXPANSION RAM (128K)-AUTO INSERT

ITEM.	QUANTITY.	COMPONENT..	DESCRIPTION	UM
0001	00001.000	2223016-0001	PWB,EXPANSION RAM 1669- -000	EA
0003	00009.000	2211118-0004	IC,64K-BIT DYNAMIC RAM,150NS TA/ROW TMS416-4-15NL	EA
0003A			U1,U2,U3,U4,U5,U6,U7,U8,U9 TMS416-4-15NL	
0005	00002.000	2220360-0002	IC,OCTAL DRAM DRIVER, 3-STATE OUTPUTS SEE TI- DRAWING	EA
0005A			U28,U29 SEE TI- DRAWING	
0006	00009.000	0972763-0001	CAPACITOR,.001UF 50V FX CERAMIC DIEL COR CA-C02Z5U102Z100A	EA
0006A			C1,C2,C3,C4,C6,C7,C8,C9,C10 COR CA-C02Z5U102Z100A	
0007	00010.000	0972763-0025	CAPACITOR,.10UF 50V FX,CERAMIC DIEL COR CA-C03Z5U104Z050A	EA
0007A			C11,C12,C13,C14,C15,C16,C17 COR CA-C03Z5U104Z050A	
0007B			C18,C20,C21 COR CA-C03Z5U104Z050A	

LIST OF MATERIALS

11/24/82

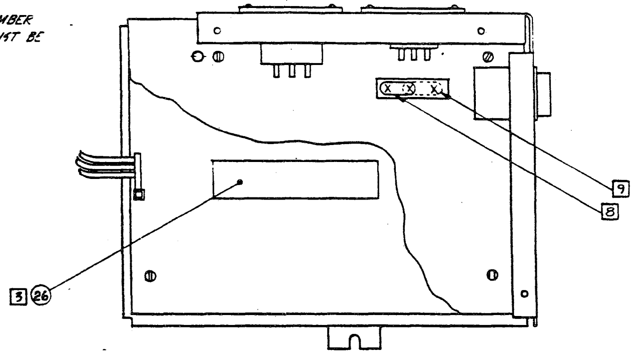
PART NUMBER	REV	DESCRIPTION
2223015-5003	E	EXPANSION RAM (192K)-AUTO INSERT

ITEM.	QUANTITY.	COMPONENT..	DESCRIPTION	UM
0001	00001.000	2223016-0001	PWB,EXPANSION RAM 1669- -000	EA
0003	00009.000	2211118-0004	IC,64K-BIT DYNAMIC RAM,150NS TA/ROW TMS416-4-15NL	EA
0003A			U1,U2,U3,U4,U5,U6,U7,U8,U9 TMS416-4-15NL	
0005	00002.000	2220360-0002	IC,OCTAL DRAM DRIVER, 3-STATE OUTPUTS SEE TI- DRAWING	EA
0005A			U28,U29 SEE TI- DRAWING	
0006	00009.000	0972763-0001	CAPACITOR,.001UF 50V FX CERAMIC DIEL COR CA-C02Z5U102Z100A	EA
0006A			C1,C2,C3,C4,C6,C7,C8,C9,C10 COR CA-C02Z5U102Z100A	
0007	00010.000	0972763-0025	CAPACITOR,.10UF 50V FX,CERAMIC DIEL COR CA-C03Z5U104Z050A	EA
0007A			C11,C12,C13,C14,C15,C16,C17 COR CA-C03Z5U104Z050A	
0007B			C18,C20,C21 COR CA-C03Z5U104Z050A	

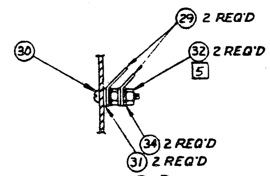
REV	DESCRIPTION	DATE	APPROVED
F	CN 501607 (D) Black (1) CHNG PER ENG SPEC	10-8-82	M Bills
G	CN485049 (D) PRUNYON (1) ON ALL LM'S ITEM 12 PIN WAS 0972969-0005, ALSO ITEM 12 QTY WAS 1.000		

NOTES: UNLESS OTHERWISE SPECIFIED

1. WIRES ARE TWISTED PAIR
2. HAND SOLDER PER PROCESS 1
3. LOCATE LABEL ITEM 26 APPROXIMATELY IN MIDDLE OF COVER ITEM 3
4. MARK APPROPRIATE T.I. PART NUMBER WHERE INDICATED. MARKING MUST BE PERMANENT AND LEGIBLE
5. TORQUE TO 1.7 ± 0.2 NM
6. TORQUE TO 0.7 ± 0.1 NM
7. TORQUE TO 0.9 ± 0.1 NM
8. VOLTAGE SELECTOR JUMPER: LEFT HAND POSITION IS 115V
9. VOLTAGE SELECTOR JUMPER: RIGHT HAND POSITION IS 220V

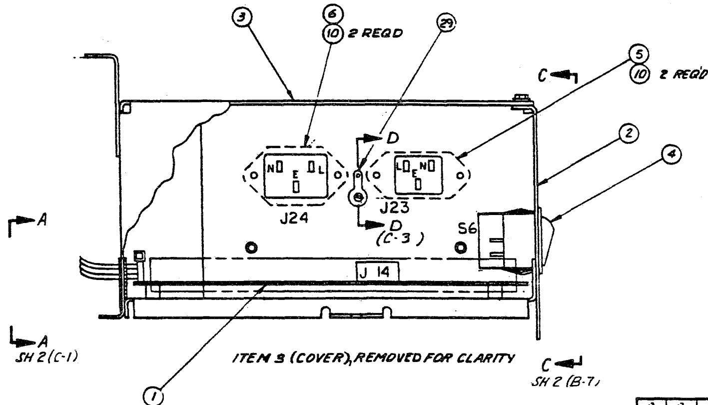


TOP VIEW WITH COVER, ITEM 3, REMOVED FOR CLARITY



SECTION D-D (B-5) SCALE NONE

CONVERSION CHART	
mm	INCHES
0.7 ± 0.1 NM	6 ± 1 INLB
0.9 ± 0.1 NM	8 ± 1.0 INLB
1.7 ± 0.2 NM	15 ± 2 INLB



ITEM 3 (COVER), REMOVED FOR CLARITY

PART NO	DESCRIPTION
~~2223037-0003~~	~~POWER SUPPLY ASSY, VDE~~ (INACTIVE)
2223037-0002	POWER SUPPLY ASSY, INTERNATIONAL
2223037-0001	POWER SUPPLY ASSY, 115V

SEQ NO	IDENT	F-SPEC	NO	ADDITIONAL CLASSIFICATION	NOTES
1	SLDR	127-01	00	HAND SOLDER	2

PROCESSES — FOR CORRELATION TO GOVT AND SPECIFICATIONS SEE TI DRAWING 72946

REV STATUS OF SHEETS: REV G F; SH 1 2

PARTS LIST

NEXT ASSY	USED ON
2223038	8755

APPLICATION

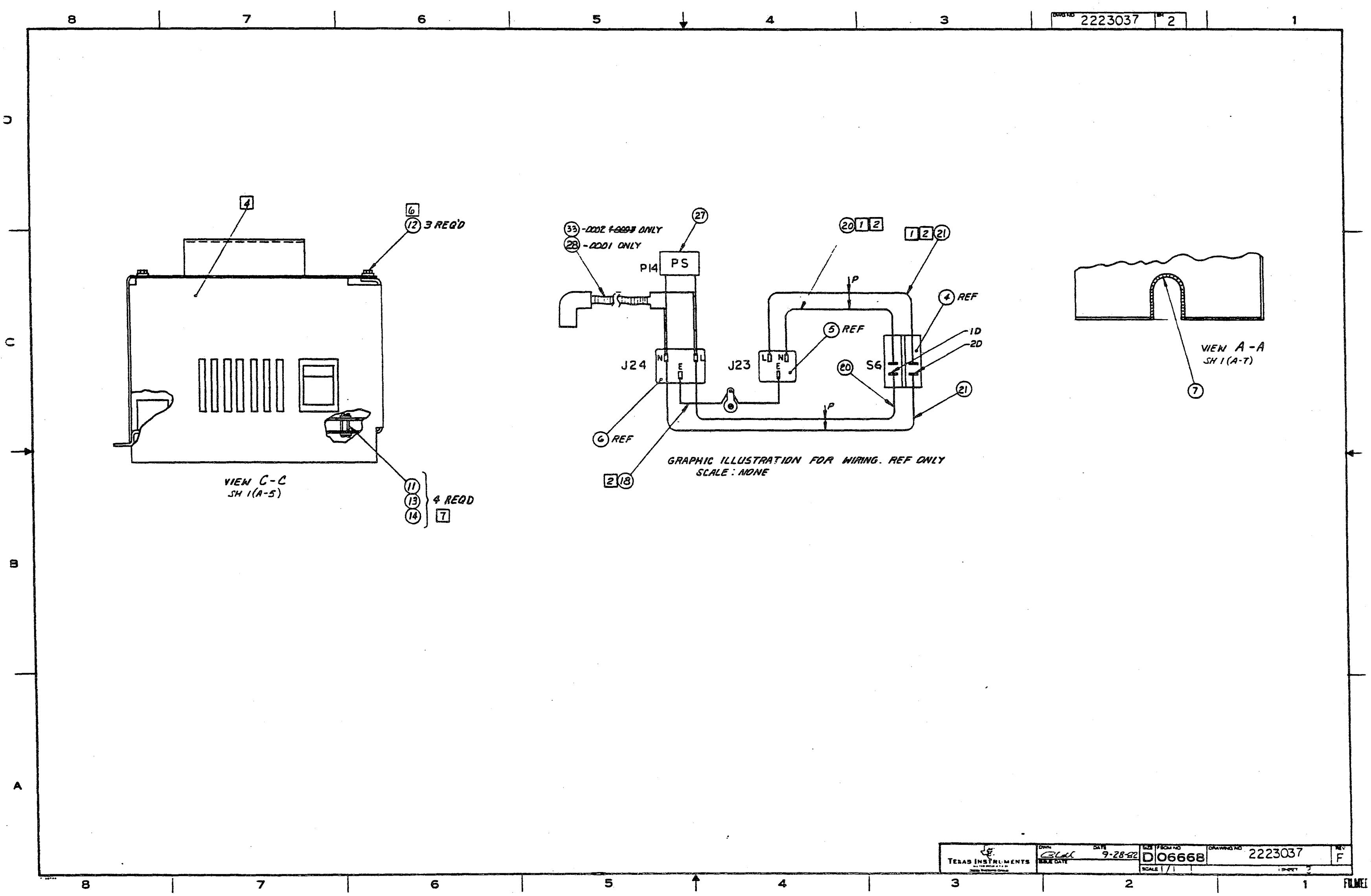
TEXAS INSTRUMENTS INCORPORATED — SI-METRIC

POWER SUPPLY ASSY

DWN 2-1-82; 4-22-82; 6-22-82; 6-22-82; 6-22-82; 6-22-82; 4/29/82

SIZE D FSCM NO 06668 DWG NO 2223037 SHEET 1 OF 2

300 LM


2223037
2
4
6
12 3 REQ'D
VIEW C-C
SH 1(A-5)
11
13
14
4 REQD
7
33 -0002 ONLY
28 -0001 ONLY
27
P14
PS
20 1 2
1 2 21
P
4 REF
5 REF
1D
2D
J24
J23
N
L
E
S6
20
21
6 REF
2 18
GRAPHIC ILLUSTRATION FOR WIRING. REF ONLY
SCALE: NONE
VIEW A-A
SH 1(A-7)
TEXAS INSTRUMENTS
9-28-82
D
06668
2223037
F

LIST OF MATERIALS

11/24/82

PART NUMBER	REV	DESCRIPTION
2223037-0001	F	POWER SUPPY ASSY-115V DOMESTIC

ITEM.	QUANTITY.	COMPONENT..	DESCRIPTION	UM
0001	00001.000	2223091-0001	POWER SUPPLY, PEGASUS SEE TI- DRAWING	EA
0002	00001.000	2223025-0001	CHASSIS,POWER SUPPLY 1678-3025-006	EA
0003	00001.000	2223026-0001	COVER,POWER SUPPLY 1678-3026-006	EA
0004	00001.000	2211949-0001	SWITCH,ROCKER,DPST,10A,250V SEE TI- DRAWING	EA
0005	00001.000	0996260-0001	RECEPTACLE,3-PIN AC PWR SCT -EAC-301	EA
0006	00001.000	2220485-0001	RECEPTACLE,AC POWER,FEMALE,3 PIN - - -C00	EA
0007	00000.500	0418082-0001	GROMMET, PLASTIC, EDGING	EA
0010	00004.000	0972831-0004	RIVET,1/8X.275,TUBULAR,STEEL,BLIND 019738-1821-0410	EA
0011	00004.000	0972988-0041	SCREW 8-32 X .250 PAN HEAD CRES	EA
0012	00003.000	0972969-0005	SCREW #6-20 X 3/8 LG THD PL HEX WASHER	EA
0013	00004.000	0411101-0059	LOCKWASHER # 8 EXTERNAL TOOTH CRES QPL - MS35335-59	EA
0014	00004.000	0416622-0024	WASHER #8 FLAT QPL - AN960C8L	EA
0018	00000.500	0996286-4455	WIRE,19-STRAND #20 GRN/YELLOW UL-1430 SEE TI- DRAWING	FT
0020	00001.400	0935172-5488	WIRE,UL 1430/3317,18AWG,GRA/YEL 1650-0000-000	FT
0021	00001.400	0935172-5088	WIRE,UL 1430/3317,18AWG,GRA/BLK 1650-0000-000	FT
0025	00000.000	2223000-0001	POWER SUPPLY, 115V 1254-1000-000	EA
0025A			*MAY BE USED AS AN 1254-1000-000	
0025B			*ALTERNATE TO ITEM 1 1254-1000-000	
0026	00001.000	2207869-0001	LABEL,WARNING HIGH VOLTAGE 1234-1869-000	EA
0027	00001.000	2223088-0001	CABLE ASSY,POWER RCPT TO PWR SUPPLY BD ----------000	EA
0028	00001.000	2220641-0001	FAN CORD ASSEMBLY,5 INCH,PVC SEE TI- DRAWING	EA
0029	00001.000	2210066-0006	LUG,RING TONGUE,TAPE MTD,#6,RED,22-16 000779-2-31879-2	EA
0030	00001.000	0972988-0030	SCREW 6-32 X .500 PAN HEAD CRES	EA
0031	00002.000	0411027-0806	WASHER,#6 FLAT,CRES,.156 X .375 X .049 QPL - MS15795-806	EA
0032	00002.000	0411115-0064	NUT,PLAIN 6-32 UNC-2B HEX CRES QPL - MS35649-264	EA
0034	00002.000	0411101-0058	LOCKWASHER #6 EXTERNAL TOOTH CRES QPL - MS35335-58	EA

LIST OF MATERIALS

11/24/82

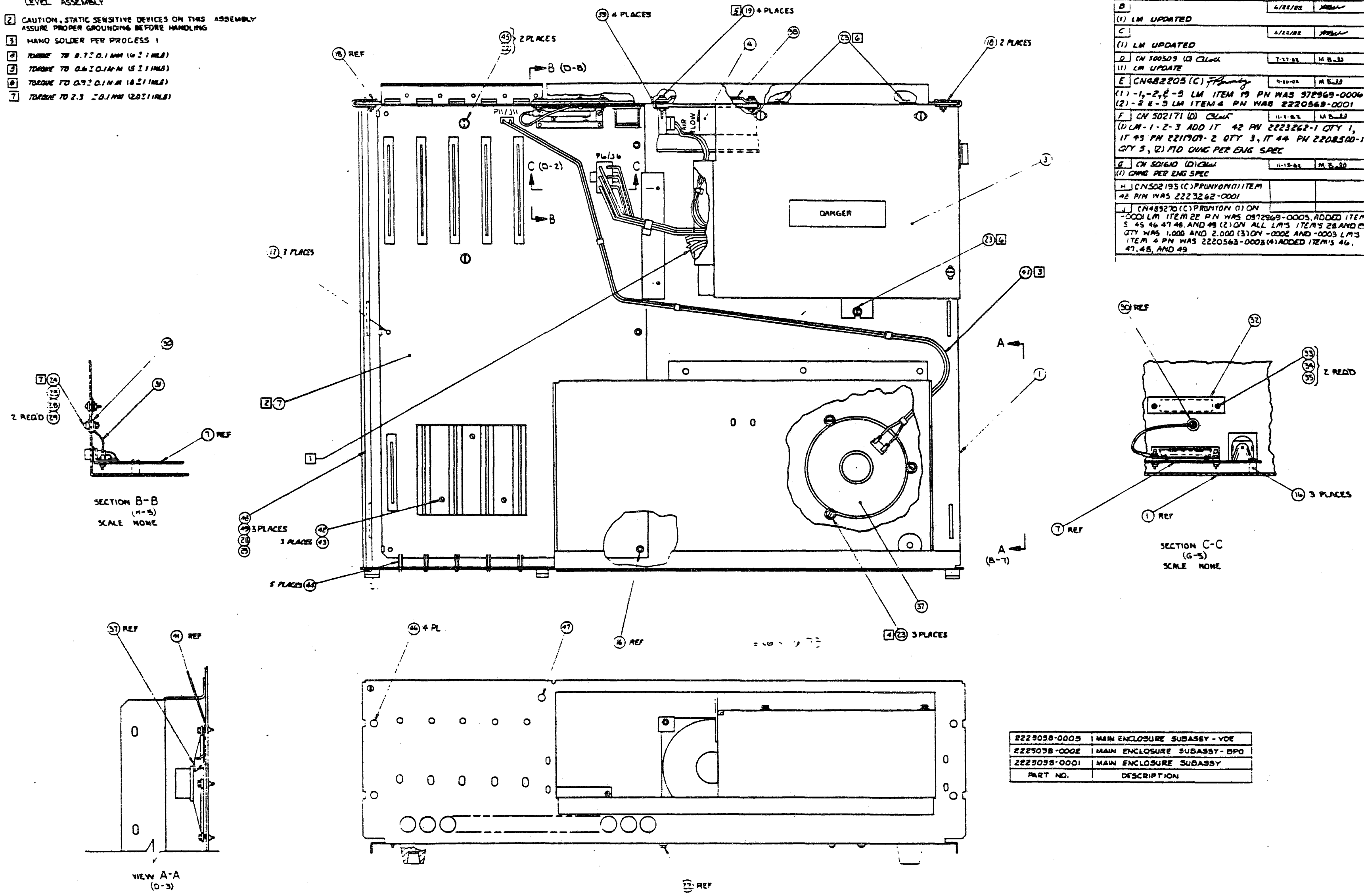
PART NUMBER	REV	DESCRIPTION...........................
2223037-0002	F	POWER SUPPLY ASSY,INTERNATIONAL

ITEM.	QUANTITY.	COMPONENT..	DESCRIPTION............................	UM
0001	00001.000	2223091-0001	POWER SUPPLY, PEGASUS SEE TI- DRAWING	EA
0002	00001.000	2223025-0001	CHASSIS,POWER SUPPLY 1678-3025-006	EA
0003	00001.000	2223026-0001	COVER,POWER SUPPLY 1678-3026-006	EA
0004	00001.000	2220637-0001	ROCKER SWITCH FOR EUROPEAN ASSEMBLIES SEE TI- DRAWING	EA
0005	00001.000	0996260-0001	RECEPTACLE,3-PIN AC PWR SCT -EAC-301	EA
0006	00001.000	2220485-0001	RECEPTACLE,AC POWER,FEMALE,3 PIN - - -000	EA
0007	AR	0418082-0001	GROMMET, PLASTIC, EDGING	EA
0010	00004.000	0972831-0004	RIVET,1/8X.275,TUBULAR,STEEL,BLIND 019738-1821-0410	EA
0011	00004.000	0972988-0041	SCREW 8-32 X .250 PAN HEAD CRES	EA
0012	00003.000	0972969-0005	SCREW #6-20 X 3/8 LG THD PL HEX WASHER	EA
0013	00004.000	0411101-0059	LOCKWASHER # 8 EXTERNAL TOOTH CRES QPL - MS35335-59	EA
0014	00004.000	0416622-0024	WASHER #8 FLAT QPL - AN960C8L	EA
0018	00000.500	0996286-4455	WIRE,19-STRAND #20 GRN/YELLOW UL-1430 SEE TI- DRAWING	FT
0020	00001.400	0935172-5488	WIRE,UL 1430/3317,18AWG,GRA/YEL 1650-0000-000	FT
0021	00001.400	0935172-5088	WIRE,UL 1430/3317,18AWG,GRA/BLK 1650-0000-000	FT
0025	00000.000	2223000-0002	POWER SUPPLY-BPO 1254-2000-000	EA
0025A			*MAY BE USED AS AN 1254-2000-000	
0025B			*ALTERNATE TO ITEM 1 1254-2000-000	
0026	00001.000	2207869-0001	LABEL,WARNING HIGH VOLTAGE 1234-1869-000	EA
0027	00001.000	2223088-0001	CABLE ASSY,POWER RCPT TO PWR SUPPLY BD ----------000	EA
0029	00001.000	2210066-0006	LUG,RING TONGUE,TAPE MTD,#6,RED,22-16 000779-2-31879-2	EA
0030	00001.000	0972988-0030	SCREW 6-32 X .500 PAN HEAD CRES	EA
0031	00002.000	0411027-0806	WASHER,#6 FLAT,CRES,.156 X .375 X .049 QPL - MS15795-806	EA
0032	00002.000	0411115-0064	NUT,PLAIN 6-32 UNC-2B HEX CRES QPL - MS35649-264	EA
0033	00001.000	2223048-0001	CABLE ASSY,INT'L FAN CORD ----------000	EA
0034	00002.000	0411101-0058	LOCKWASHER #6 EXTERNAL TOOTH CRES QPL - MS35335-58	EA

NOTES: UNLESS OTHERWISE SPECIFIED

1. CABLES TO BE CONNECTED AT NEXT LEVEL ASSEMBLY
2. CAUTION, STATIC SENSITIVE DEVICES ON THIS ASSEMBLY ASSURE PROPER GROUNDING BEFORE HANDLING
3. HAND SOLDER PER PROCESS 1
4. TORQUE TO 0.7 ± 0.1 N·M (6 ± 1 IN-LB)
5. TORQUE TO 0.6 ± 0.1 N·M (5 ± 1 IN-LB)
6. TORQUE TO 0.9 ± 0.1 N·M (8 ± 1 IN-LB)
7. TORQUE TO 2.3 ± 0.1 N·M (20 ± 1 IN-LB)

REVISIONS			
REV	DESCRIPTION	DATE	APPROVED
A	(1) LM UPDATED	4/22/82	
B	(1) LM UPDATED	6/22/82	
C	(1) LM UPDATED	6/22/82	
D	CN 500509 (D) (1) LM UPDATE	7-27-82	M B
E	CN482205 (C) (1) -1, -2, & -3 LM ITEM 19 PN WAS 972969-0006 (2) -2 & -3 LM ITEM 4 PN WAS 2220569-0001	9-10-82	M B
F	CN 502171 (D) (1) LM-1-2-3 ADD IT 42 PN 2223262-1 QTY 1, IT 43 PN 2211909-2 QTY 3, IT 44 PN 2208500-1 QTY 5, (2) FID CHNG PER ENG SPEC	11-1-82	M B
G	CN 501610 (D) (1) CHNG PER ENG SPEC	11-18-82	M B
H	CN502193 (C) PRUNYON (1) ITEM 42 P/N WAS 2223262-0001		
J	CN489270 (C) PRUNTON (1) ON -0001 LM ITEM 22 PN WAS 0972969-0005, ADDED ITEMS 45 46 47 48, AND 49 (2) ON ALL LM'S ITEM'S 28 AND 29 QTY WAS 1.000 AND 2.000 (3) ON -0002 AND -0003 LM'S ITEM 4 PN WAS 2220563-0003 (4) ADDED ITEM'S 46, 47, 48, AND 49		



PART NO.	DESCRIPTION
2223038-0003	MAIN ENCLOSURE SUBASSY - VDE
2223038-0002	MAIN ENCLOSURE SUBASSY - BPO
2223038-0001	MAIN ENCLOSURE SUBASSY

FLOOR	121-01	00	HAND SOLDER
SEE	IDENT	SPEC	NO.

PROCESS — CLASSIFICATION

PARTS LIST

NEXT ASSY	USED ON
2223051	8755
2223050	8755

APPLICATION

TEXAS INSTRUMENTS

MAIN ENCLOSURE SUBASSY

E 96214 2223038

LIST OF MATERIALS

11/24/82

PART NUMBER	REV	DESCRIPTION
2223038-0001	G	MAIN ENCLOSURE,SUBASSY

ITEM.	QUANTITY.	COMPONENT..	DESCRIPTION	UM
0001	00001.000	2223024-0001	CHASSIS,TERMINAL 1678-3024-008	EA
0003	00001.000	2223037-0001	POWER SUPPY ASSY-115V DOMESTIC 1669-1037-000	EA
0004	00001.000	2220632-0001	FAN,115 VAC,29 CFM,13 W,TUBEAXIAL SEE TI- DRAWING	EA
0007	00001.000	2223003-0001	MOTHERBOARD - PEGASUS 1254-3003-005	EA
0016	00003.000	2211907-0005	SPACER,PCB,.31"BODY,NYLON,HOLE/#6 SCREW SEE TI- DRAWING	EA
0017	00003.000	2220484-0001	SUPPORT,PC BOARD,SELF-MOUNT - - -000	EA
0018	00002.000	2220487-0001	SPEEDNUT,J-TYPE,WITH T-NUT - - -000	EA
0019	00004.000	0972684-0012	SCREW 6-32 X 1/2 THD SLOT HEX WASHER HD	EA
0022	00002.000	0972969-0005	SCREW #6-20 X 3/8 LG THD PL HEX WASHER	EA
0023	00006.000	0972684-0018	SCREW 8-32 X 3/8 THD FRM,SLOT HX WSR HD	EA
0024	00001.000	0972988-0045	SCREW 8-32 X .500 PAN HEAD CRES	EA
0025	00001.000	0411115-0084	NUT,PLAIN 8-32 UNC-2B HEX CRES QPL - MS35649-284	EA
0028	00001.000	0411104-0137	WASHER, LOCK-SPRING, HELICAL, #8 QPL - MS35338-137	EA
0029	00002.000	0411027-0807	WASHER,#8 FLAT,CRES,.188 X .375 X .049 QPL - MS15795-807	EA
0030	00001.000	0411100-0072	LOCKWASHER #8,INTERNAL TOOTH CRES QPL - MS35333-72	EA
0031	00001.000	2223079-0001	CABLE ASSEMBLY,GROUNDING ----------000	EA
0032	00001.000	2223080-0001	PLATE,BLANK,EXTERNAL FLOPPY 1678-3080-005	EA
0033	00002.000	0972988-0013	SCREW 4-40 X .250 PAN HEAD CRES	EA
0034	00002.000	0411104-0135	WASHER, LOCK-SPRING, HELICAL, #4 QPL - MS35338-135	EA
0035	00002.000	0411027-0803	WASHER .125 X .250 X .022 FLAT CRES QPL - MS15795-803	EA
0037	00001.000	2220556-0001	SPEAKERS,8 OHM 2 WATT SEE TI- DRAWING	EA
0038	00001.000	0972373-0001	GUARD FAN RTN --476143	EA
0039	00004.000	0972802-0014	FASTENER,SPEED NUT,STL,6-32,.41 L 078553-C10132-632	EA
0041	00001.000	2223108-0001	CABLE ASSY,SPEAKER ----------000	EA
0042	00001.000	2223262-0001	CARD GUIDE LOWER ----------000	EA
0043	00003.000	2211909-0002	PCB SPACER,NYLON,.37"BODY SEE TI- DRAWING	EA
0044	00005.000	2220850-0001	GUIDE,NYLON,2.50' LONG,GROOVE MOUNTING SEE TI- DRAWING	EA

LIST OF MATERIALS

11/24/82

PART NUMBER	REV	DESCRIPTION...........................
2223038-0002	G	MAIN ENCLOSURE SUBASSY-BPO

ITEM.	QUANTITY.	COMPONENT..	DESCRIPTION............................	UM
0001	00001.000	2223024-0001	CHASSIS,TERMINAL 1678-3024-008	EA
0003	00001.000	2223037-0002	POWER SUPPLY ASSY,INTERNATIONAL 1669-2037-000	EA
0004	00001.000	2220563-0003	FAN,230VAC,50/60 HZ,9W,32CFM,3.12X1.5" SEE TI- DRAWING	EA
0007	00001.000	2223003-0001	MOTHERBOARD - PEGASUS 1254-3003-005	EA
0016	00003.000	2211907-0005	SPACER,PCB,.31"BODY,NYLON,HOLE/#6 SCREW SEE TI- DRAWING	EA
0017	00003.000	2220484-0001	SUPPORT,PC BOARD,SELF-MOUNT - - -000	EA
0018	00002.000	2220487-0001	SPEEDNUT,J-TYPE,WITH T-NUT - - -000	EA
0019	00004.000	0972684-0012	SCREW 6-32 X 1/2 THD SLOT HEX WASHER HD	EA
0023	00006.000	0972684-0018	SCREW 8-32 X 3/8 THD FRM,SLOT HX WSR HD	EA
0024	00001.000	0972988-0045	SCREW 8-32 X .500 PAN HEAD CRES	EA
0025	00001.000	0411115-0084	NUT,PLAIN 8-32 UNC-2B HEX CRES QPL - MS35649-284	EA
0028	00001.000	0411104-0137	WASHER, LOCK-SPRING, HELICAL, #8 QPL - MS35338-137	EA
0029	00002.000	0411027-0807	WASHER,#8 FLAT,CRES,.188 X .375 X .049 QPL - MS15795-807	EA
0030	00001.000	0411100-0072	LOCKWASHER #8,INTERNAL TOOTH CRES QPL - MS35333-72	EA
0031	00001.000	2223079-0001	CABLE ASSEMBLY,GROUNDING ----------000	EA
0032	00001.000	2223080-0001	PLATE,BLANK,EXTERNAL FLOPPY 1678-3080-005	EA
0033	00002.000	0972988-0013	SCREW 4-40 X .250 PAN HEAD CRES	EA
0034	00002.000	0411104-0135	WASHER, LOCK-SPRING, HELICAL, #4 QPL - MS35338-135	EA
0035	00002.000	0411027-0803	WASHER .125 X .250 X .022 FLAT CRES QPL - MS15795-803	EA
0037	00001.000	2220556-0001	SPEAKERS,8 OHM 2 WATT SEE TI- DRAWING	EA
0038	00001.000	0972373-0001	GUARD FAN PTN --476143	EA
0039	00004.000	0972802-0014	FASTENER,SPEED NUT,STL,6-32,.41 L 078553-C10132-632	EA
0041	00001.000	2223108-0001	CABLE ASSY,SPEAKER ----------000	EA
0042	00001.000	2223262-0001	CARD GUIDE LOWER ----------000	EA
0043	00003.000	2211909-0002	PCB SPACER,NYLON,.37"BODY SEE TI- DRAWING	EA
0044	00005.000	2220850-0001	GUIDE,NYLON,2.50' LONG,GROOVE MOUNTING SEE TI- DRAWING	EA

LIST OF MATERIALS

11/24/82

PART NUMBER	REV	DESCRIPTION..........................
2223038-0003	G	MAIN ENCLOSURE SUB ASSY-VDE

ITEM.	QUANTITY.	COMPONENT..	DESCRIPTION..........................	UM
0001	00001.000	2223024-0001	CHASSIS,TERMINAL 1678-3024-008	EA
0003	00001.000	2223037-0002	POWER SUPPLY ASSY,INTERNATIONAL 1669-2037-000	EA
0004	00001.000	2220563-0003	FAN,230VAC,50/60 HZ,9W,32CFM,3.12X1.5" SEE TI- DRAWING	EA
0007	00001.000	2223003-0001	MOTHERBOARD - PEGASUS 1254-3003-005	EA
0016	00003.000	2211907-0005	SPACER,PCB,.31"BODY,NYLON,HOLE/#6 SCREW SEE TI- DRAWING	EA
0017	00003.000	2220484-0001	SUPPORT,PC BOARD,SELF-MOUNT - - -000	EA
0018	00002.000	2220487-0001	SPEEDNUT,J-TYPE,WITH T-NUT - - -000	EA
0019	00004.000	0972684-0012	SCREW 6-32 X 1/2 THD SLOT HEX WASHER HD	EA
0023	00006.000	0972684-0018	SCREW 8-32 X 3/8 THD FRM,SLOT HX WSR HD	EA
0024	00001.000	0972988-0045	SCREW 8-32 X .500 PAN HEAD CRES	EA
0025	00001.000	0411115-0084	NUT,PLAIN 8-32 UNC-2B HEX CRES QPL - MS35649-284	EA
0028	00001.000	0411104-0137	WASHER, LOCK-SPRING, HELICAL, #8 QPL - MS35338-137	EA
0029	00002.000	0411027-0807	WASHER,#8 FLAT,CRES,.188 X .375 X .049 QPL - MS15795-807	EA
0030	00001.000	0411100-0072	LOCKWASHER #8,INTERNAL TOOTH CRES QPL - MS35333-72	EA
0031	00001.000	2223079-0001	CABLE ASSEMBLY,GROUNDING ----------000	EA
0032	00001.000	2223080-0001	PLATE,BLANK,EXTERNAL FLOPPY 1678-3080-005	EA
0033	00002.000	0972988-0013	SCREW 4-40 X .250 PAN HEAD CRES	EA
0034	00002.000	0411104-0135	WASHER, LOCK-SPRING, HELICAL, #4 QPL - MS35338-135	EA
0035	00002.000	0411027-0803	WASHER .125 X .250 X .022 FLAT CRES QPL - MS15795-803	EA
0037	00001.000	2220556-0001	SPEAKERS,8 OHM 2 WATT SEE TI- DRAWING	EA
0038	00001.000	0972373-0001	GUARD FAN RTN --476143	EA
0039	00004.000	0972802-0014	FASTENER,SPEED NUT,STL,6-32,.41 L 078553-C10132-632	EA
0041	00001.000	2223108-0001	CABLE ASSY,SPEAKER ----------000	EA
0042	00001.000	2223262-0001	CARD GUIDE LOWER ----------000	EA
0043	00003.000	2211909-0002	PCB SPACER,NYLON,.37"BODY SEE TI- DRAWING	EA
0044	00005.000	2220850-0001	GUIDE,NYLON,2.50' LONG,GROOVE MOUNTING SEE TI- DRAWING	EA

DWG NO 2223050 SH 1

NOTES: UNLESS OTHERWISE SPECIFIED

[1] FACE PLATE OF ITEM 17 MUST BUTT AGAINST OPENING OF 'FRONT PANEL' BEFORE TIGHTENING ITEM 22

[2] FACE OF INSERT PLATE (ITEM 4) MUST BUTT AGAINST OPENING OF 'FRONT PANEL' BEFORE TIGHTENING ITEM 14

[3] CAUTION STATIC SENSITIVE DEVICES ON THIS ASSEMBLY, PROPER GROUNDING REQUIRED

[4] LOCATION OF LABELS (ITEMS 9, 37, 38 AND 39) IS NOT CRITICAL

[5] POWER CORD, ITEM 6 OR 7 IS TO BE PACKED LOOSE WITH UNIT

[6] ITEM 10 SELF DESTRUCT LABEL PLACED APPROXIMATELY IN THIS LOCATION

7. ~~ITEMS 42,43,44, & 45 TO BE PACKED LOOSE W/ UNIT (DASH ONE ONLY)~~

8. ITEM 46 & 48 TO BE PACKED LOOSE WITH UNIT (DASH ONE AND DASH TWO.

[9] SECURE ITEM 41, WITH CLIP PROVIDED, WHEN NOT IN USE

[10] IF CONNECTORS P21 AND P22 ARE NOT USED SECURE WITH ITEM 29 AS SHOWN

[11] SET MOTHERBOARD OPTION JUMPER PLUGS AS FOLLOWS: -0001, -0002 CONNECT 1 EACH TO E2, E4, E6 AND E20, -0003 CONNECT 1 EACH TO E4 AND E6, CONNECT E1 TO E2 AND E9 TO E20

[12] TORQUE TO 0.5 ± 0.1 NM

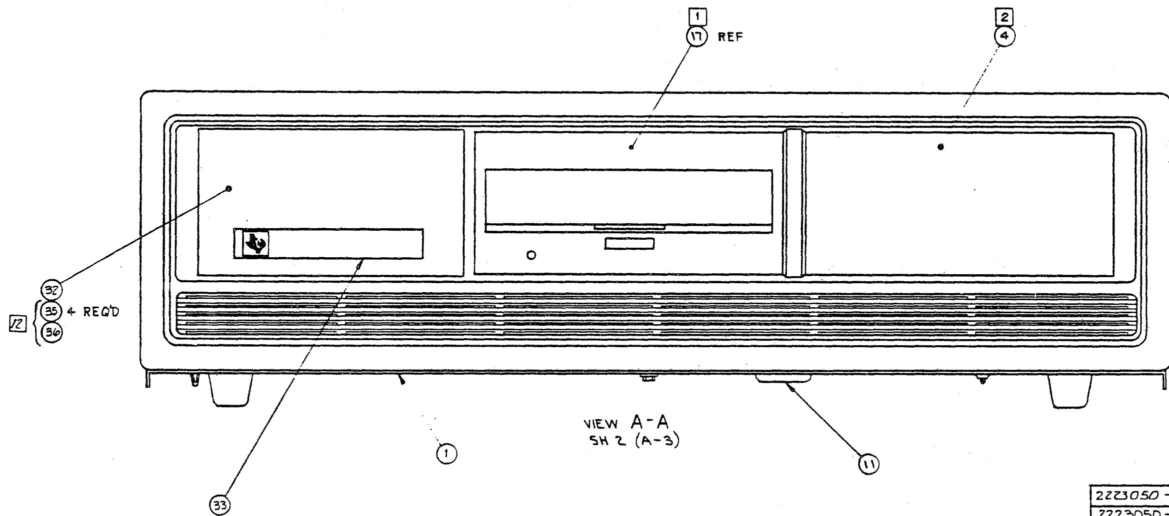
[13] TORQUE TO 0.7 ± 0.1 NM

[14] TORQUE TO 1.1 ± 0.1 NM

[15] TORQUE TO 0.9 ± 0.1 NM

16 REFER TO ITEM 50 FOR FLOPPY DRIVE CONFIGURATION INSTRUCTIONS WHICH MUST BE PERFORMED (-0001 AND -0003 ONLY)

REVISIONS			
REV	DESCRIPTION	DATE	APPROVED
A		6/22/82	[signature]
	(1) UPDATED LM		
B	CN 501136 (D) [signature]	7-13-82	M Buis
	(1) LM-2 DELE IT 22 PN996739-14 QTY 3 (2) LM-1 & -2 ADD IT 29, ADD IT 46, DELE IT 40 PN2220230-1 QTY AR		
C	CN 502136 (C) Black	9-9-82	M Buis
	(1) LM UPDATE (2) CREATE -3 LM		
D	CN 502200 (B) LAMB	9-30-82	M Buis
	(1) ADD NOTE 11 (2) SH 2 Z 2B ADD VIEW C		
E	CN 501609 (D) Black	10-19-82	M Buis
	(1) CHNG PER ENG SPEC		



PART NO.	DESCRIPTION
2223050-0003	SYSTEM ASSY, DOMESTIC - 320 K FLOPPY
2223050-0002	SYSTEM ASSY, DOMESTIC - BASIC
2223050-0001	SYSTEM ASSY, DOMESTIC - STANDARD

-3 QTY	-2 QTY	-1 QTY	ITEM NO	PART OR IDENTIFYING NUMBER	NOMENCLATURE OR DESCRIPTION	NOTES

PARTS LIST

UNLESS OTHERWISE SPECIFIED
- DIMENSIONS ARE IN MILLIMETERS
- TOLERANCES 2 PLACE DECIMALS ±0.25, 1 PLACE DECIMALS ±0.5, ANGLES ±1°
- INTERPRET DRAWING PER DOD-D-1000
- REMOVE ALL BURRS AND SHARP EDGES
- CONCENTRICITY MACHINED DIAMETERS 0.25 FIM
- DIMENSIONAL LIMITS APPLY BEFORE PROCESSES
- PARENTHETICAL INFO FOR REF ONLY

HOLE TOLERANCE	
0.33 THRU 3.18	+0.10/-0.03
3.20 THRU 8.36	+0.13/-0.03
8.38 THRU 12.70	+0.15/-0.03
12.73 THRU 19.06	+0.20/-0.03
19.08 THRU 25.40	+0.25/-0.03
25.43 THRU 50.80	+0.30/-0.03

THIRD ANGLE PROJECTION

NEXT ASSY	USED ON
2223040	8755

APPLICATION

DWN [signature] DATE 9-1-82
CHK M. Buis 6-22-82
ENGR [signature] 6-22-82
APVD ENGR [signature] 6-22-82
QA [signature]
APVD MFG Susan Bridges 6-22-82
RLSE [signature] 6/23/82

TEXAS INSTRUMENTS INCORPORATED
Digital Systems Group

SI-METRIC

SYSTEM ASSY, DOMESTIC

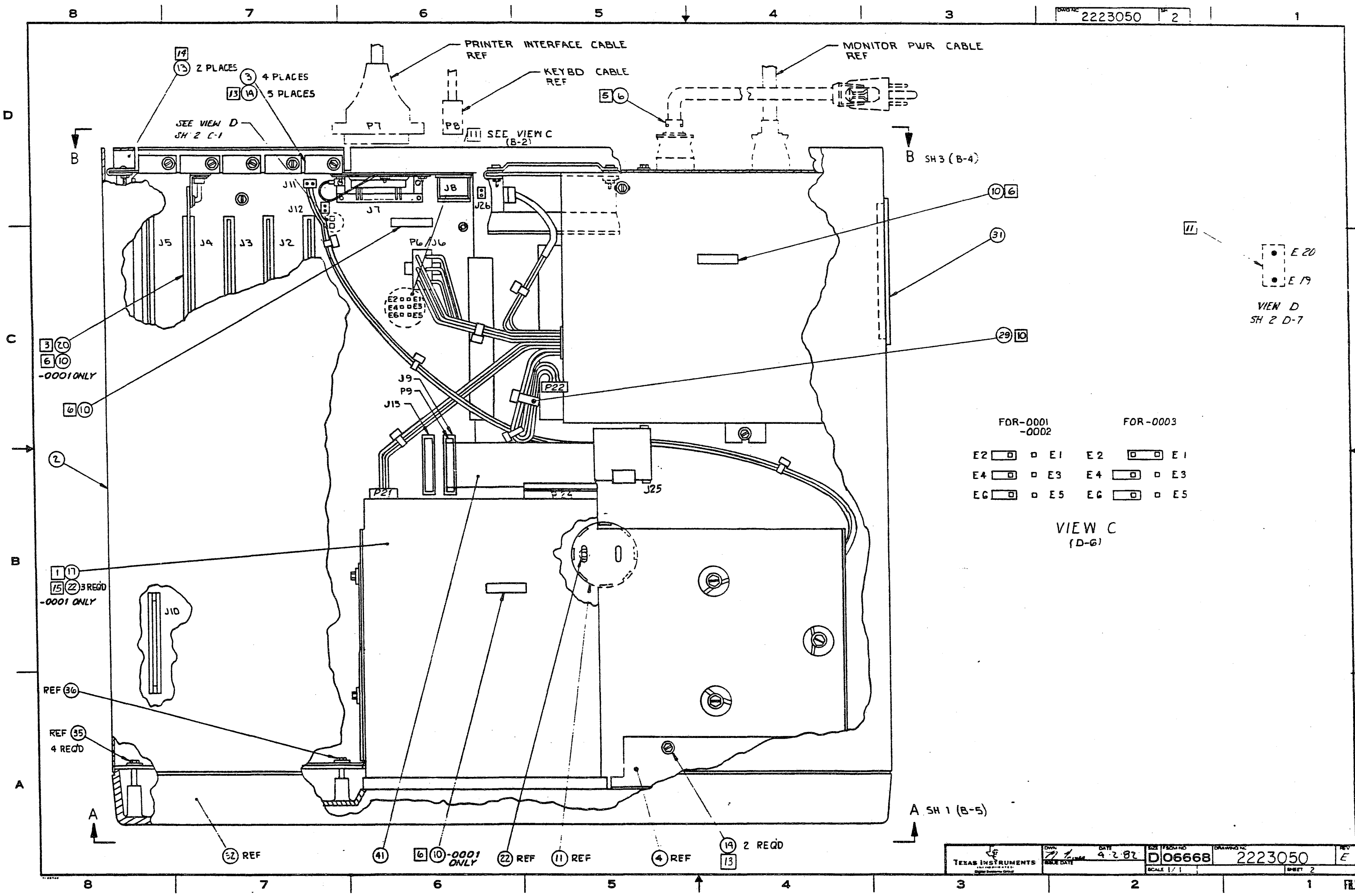
SIZE D FSCM NO 06668 DRAWING NO 2223050

SCALE 1/1 SHEET 1 OF 3

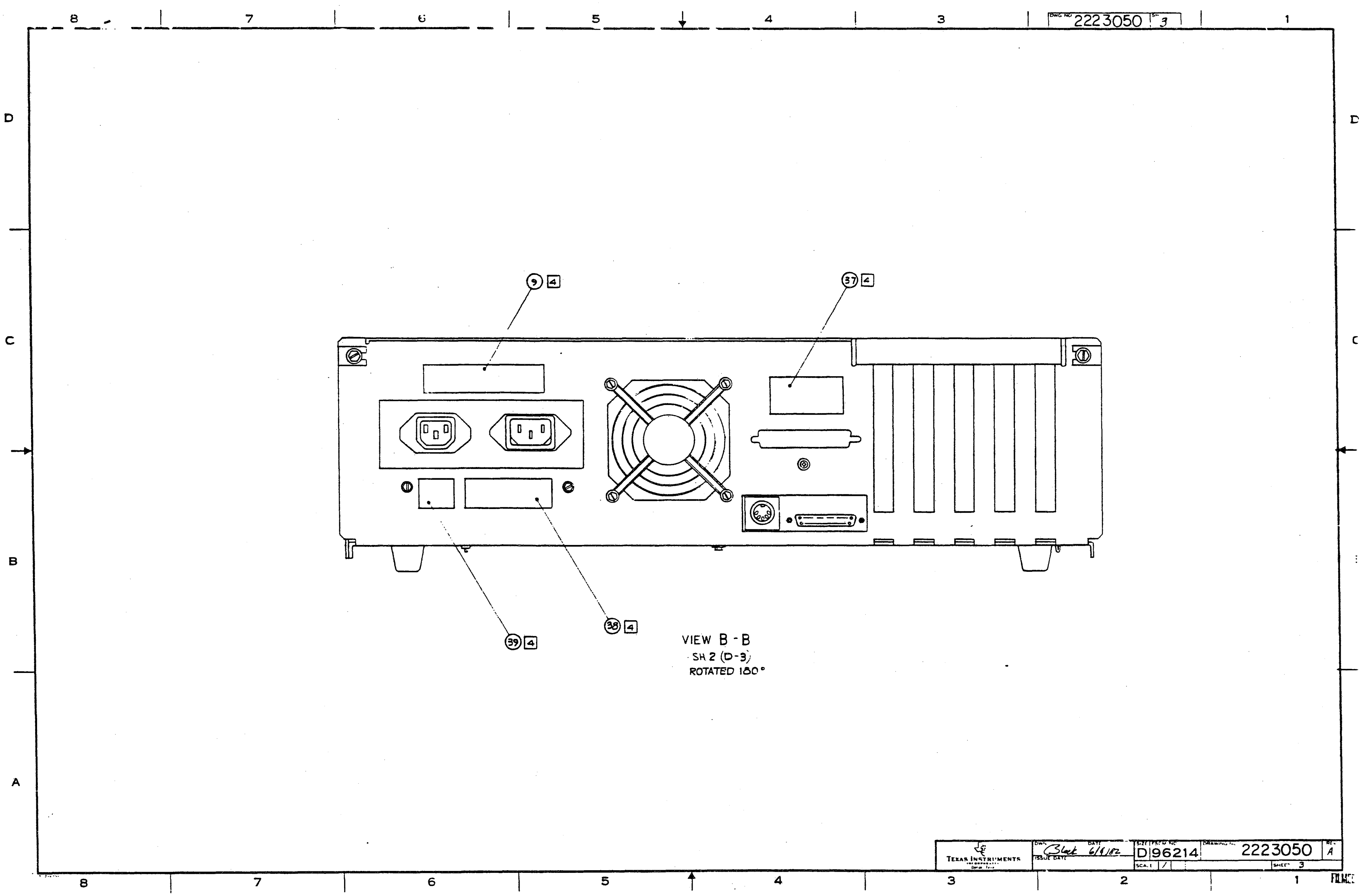
SEQ NO	IDENT	F-SPEC	NO	ADDITIONAL	NOTES	REV STATUS OF SHEETS	REV	E	E	A
	PROCESS			CLASSIFICATION			SH	1	2	3

PROCESSES — FOR CORRELATION TO GOVT AND SPECIFICATIONS SEE TI DRAWING 72945

300 CC LM



PRINTER INTERFACE CABLE REF
KEYBD CABLE REF
MONITOR PWR CABLE REF
2 PLACES
4 PLACES
5 PLACES
SEE VIEW D SH 2 C-1
SEE VIEW C (B-2)
B
B SH 3 (B-4)
P7
P8
J11
J12
J7
J8
J26
J5
J4
J3
J2
P6/J6
E2 E1
E4 E3
E6 E5
-0001 ONLY
J9
P9
J15
P22
P21
P24
J25
J10
3 REQD
-0001 ONLY
REF
REF
4 REQD
A
A SH 1 (B-5)
REF
-0001 ONLY
REF
REF
REF
2 REQD
E 20
E 19
VIEW D
SH 2 D-7
FOR-0001
-0002
FOR-0003
E2 E1
E4 E3
E6 E5
E2 E1
E4 E3
E6 E5
VIEW C
(D-6)
2223050
2
TEXAS INSTRUMENTS
9-2-82
D
06668
2223050
E


2223050 3
VIEW B - B
SH 2 (D-3)
ROTATED 180°
9 4
37 4
38 4
39 4
TEXAS INSTRUMENTS
D 96214
2223050
A

LIST OF MATERIALS

11/24/82

PART NUMBER	REV	DESCRIPTION............................
2223050-0001	F	SYSTEM ASSY-STANDARD

ITEM.	QUANTITY.	COMPONENT..	DESCRIPTION............................	UM
0001	00001.000	2223038-0001	MAIN ENCLOSURE,SUBASSY 1669-1038-000	EA
0002	00001.000	2223029-0001	COVER,TERMINAL 1678-3029-006	EA
0003	00004.000	2223033-0001	PLATE OPTION BOARD 1678-3133-009	EA
0004	00001.000	2223034-0001	INSERT PLATE,FLOPPY 1678-3134-008	EA
0006	00001.000	0996289-0001	CORD SET,3-PIN PWR-DOMESTIC BLACK 080126-0-7889-008-GY	EA
0007	00000.000	0996289-0002	CORD SET,3-PIN PWR-DOMESTIC GRAY W/CLIP 080126-0-7919-008-GY	EA
0007A			*MAY BE USED AS AN 080126-0-7919-008-GY	
0007B			*ALTERNATE TO ITEM 6. 080126-0-7919-008-GY	
0009	00001.000	2223075-0001	LABEL,SERIAL-950 TERM,STANDARD DOM 1669-1075-000	EA
0010	AR	0996943-0001	LABEL, SELF-DESTRUCT, .656 X .25 1652-1274-000	EA
0011	00001.000	2211919-0002	PLUG,HOLE-1.563 DIA SEE TI- DRAWING	EA
0013	00002.000	0972988-0043	SCREW 8-32 X .375 PAN HEAD CRES	EA
0014	00007.000	0972684-0011	SCREW, THREAD FORMING, #6-32 1282-5256-000	EA
0017	00001.000	2220446-0001	DISK DRIVE ASSY,FLOPPY,5.25 INCH 1254- -000	EA
0020	00001.000	2223009-0001	ALPHA CRT CONTROLLER 1254-3009-005	EA
0022	00003.000	2210071-0009	SCREW, 6-32 X 3/8, HEX HEAD SEE TI- DRAWING	EA
0025	REF	2223082-0001	INTERCONNECT DIAGRAM	EA
0029	00001.000	0972632-0001	STRAP,TIE DOWN,CABLE-NON-STD,0-1-1/4 D.	EA
0031	00001.000	2223076-0001	INSERT SWITCH OPENING 1255-3519-002	EA
0032	00001.000	2223020-0001	PANEL,FRONT 1255-3521-002	EA
0033	00001.000	2223090-0001	NAMEPLATE,PROFESSIONAL COMPUTER ----------000	EA
0035	00004.000	0972969-0009	SCREW,6-20 X 7/8 HEX WASHER HEAD SEE TI- DRAWING	EA
0036	00001.000	0972969-0008	SCREW,6-20 X 3/4 HEX WASHER HEAD	EA
0037	00001.000	2211184-0001	LABEL,.334H, FCC CLASS A EQUIPMENT SEE TI- DRAWING	EA
0038	00001.000	2269942-0001	LABEL,UL	EA
0039	00001.000	2269943-0001	LABEL,CSA	EA
0041	00001.000	2223097-0001	CABLE ASSY,MOTHERBOARD TO FLOPPY ----------000	EA
0046	00001.000	0999456-9701	MANUAL,INFORMATION REQUEST FORM 1225-9456-000	EA
0048	00001.000	2223203-0001	MANUAL-GETTING STARTED 1261-3203-000	EA
0050	REF	2223279-0001	CONFIGURATION,FLOPPY DISK DRIVES	EA

LIST OF MATERIALS

11/24/82

PART NUMBER REV DESCRIPTION...........................
2223050-0002 E SYSTEM ASSY-BASIC

ITEM.	QUANTITY.	COMPONENT..	DESCRIPTION............................	UM
0001	00001.000	2223038-0001	MAIN ENCLOSURE,SUBASSY 1669-1038-000	EA
0002	00001.000	2223029-0001	COVER,TERMINAL 1678-3029-006	EA
0003	00005.000	2223033-0001	PLATE OPTION BOARD 1678-3133-009	EA
0004	00002.000	2223034-0001	INSERT PLATE,FLOPPY 1678-3134-008	EA
0006	00001.000	0996289-0001	CORD SET,3-PIN PWR-DOMESTIC BLACK 080126-0-7889-008-GY	EA
0007	00000.000	0996289-0002	CORD SET,3-PIN PWR-DOMESTIC GRAY W/CLIP 080126-0-7919-008-GY	EA
0007A			*MAY BE USED AS AN 080126-0-7919-008-GY	
0007B			*ALTERNATE TO ITEM 6. 080126-0-7919-008-GY	
0009	00001.000	2223075-0002	LABEL,SERIAL-950 TERMINAL,BASIC DOMESTIC 1669-2075-000	EA
0010	AR	0996943-0001	LABEL, SELF-DESTRUCT, .656 X .25 1652-1274-000	EA
0011	00001.000	2211919-0002	PLUG,HOLE-1.563 DIA SEE TI- DRAWING	EA
0013	00002.000	0972988-0043	SCREW 8-32 X .375 PAN HEAD CRES	EA
0014	00007.000	0972684-0011	SCREW, THREAD FORMING, #6-32 1282-5256-000	EA
0025	REF	2223082-0001	INTERCONNECT DIAGRAM	EA
0029	00001.000	0972632-0001	STRAP,TIE DOWN,CABLE-NON-STD,0-1-1/4 D.	EA
0031	00001.000	2223076-0001	INSERT SWITCH OPENING 1255-3519-002	EA
0032	00001.000	2223020-0001	PANEL,FRONT 1255-3521-002	EA
0033	00001.000	2223090-0001	NAMEPLATE,PROFESSIONAL COMPUTER ----------000	EA
0035	00004.000	0972969-0009	SCREW,6-20 X 7/8 HEX WASHER HEAD SEE TI- DRAWING	EA
0036	00001.000	0972969-0008	SCREW,6-20 X 3/4 HEX WASHER HEAD	EA
0037	00001.000	2211184-0001	LABEL,.334H, FCC CLASS A EQUIPMENT SEE TI- DRAWING	EA
0038	00001.000	2269942-0001	LABEL,UL	EA
0039	00001.000	2269943-0001	LABEL,CSA	EA
0041	00001.000	2223097-0001	CABLE ASSY,MOTHERBOARD TO FLOPPY ----------000	EA
0046	00001.000	0999456-9701	MANUAL,INFORMATION REQUEST FORM 1225-9456-000	EA
0048	00001.000	2223203-0001	MANUAL-GETTING STARTED 1261-3203-000	EA

LIST OF MATERIALS

11/24/82

PART NUMBER REV DESCRIPTION...........................
2223050-0003 F SYSTEMS ASSY STANDARD-320K

ITEM.	QUANTITY.	COMPONENT..	DESCRIPTION...........................	UM
0001	00001.000	2223038-0001	MAIN ENCLOSURE,SUBASSY 1669-1038-000	EA
0002	00001.000	2223029-0001	COVER,TERMINAL 1678-3029-006	EA
0003	00004.000	2223033-0001	PLATE OPTION BOARD 1678-3133-009	EA
0004	00001.000	2223034-0001	INSERT PLATE,FLOPPY 1678-3134-008	EA
0006	00001.000	0996289-0001	CORD SET,3-PIN PWR-DOMESTIC BLACK 080126-0-7889-008-GY	EA
0007	00000.000	0996289-0002	CORD SET,3-PIN PWR-DOMESTIC GRAY W/CLIP 080126-0-7919-008-GY	EA
0007A			*MAY BE USED AS AN 080126-0-7919-008-GY	
0007B			*ALTERNATE TO ITEM 6. 080126-0-7919-008-GY	
0009	00001.000	2223075-0001	LABEL,SERIAL-950 TERM,STANDARD DOM 1669-1075-000	EA
0010	AR	0996943-0001	LABEL, SELF-DESTRUCT, .656 X .25 1652-1274-000	EA
0011	00001.000	2211919-0002	PLUG,HOLE-1.563 DIA SEE TI- DRAWING	EA
0013	00002.000	0972988-0043	SCREW 8-32 X .375 PAN HEAD CRES	EA
0014	00007.000	0972684-0011	SCREW, THREAD FORMING, #6-32 1282-5256-000	EA
0017	00001.000	2220446-0002	DISK DRIVE ASSY,FLOPPY,5.25INCH-DUAL HD SEE TI- DRAWING	EA
0020	00001.000	2223009-0001	ALPHA CRT CONTROLLER 1254-3009-005	EA
0022	00003.000	2210071-0009	SCREW, 6-32 X 3/8, HEX HEAD SEE TI- DRAWING	EA
0025	REF	2223082-0001	INTERCONNECT DIAGRAM	EA
0029	00001.000	0972632-0001	STRAP,TIE DOWN,CABLE-NON-STD,0-1-1/4 D.	EA
0031	00001.000	2223076-0001	INSERT SWITCH OPENING 1255-3519-002	EA
0032	00001.000	2223020-0001	PANEL,FRONT 1255-3521-002	EA
0033	00001.000	2223090-0001	NAMEPLATE,PROFESSIONAL COMPUTER ----------000	EA
0035	00004.000	0972969-0009	SCREW,6-20 X 7/8 HEX WASHER HEAD SEE TI- DRAWING	EA
0036	00001.000	0972969-0008	SCREW,6-20 X 3/4 HEX WASHER HEAD	EA
0037	00001.000	2211184-0001	LABEL,.334H, FCC CLASS A EQUIPMENT SEE TI- DRAWING	EA
0038	00001.000	2269942-0001	LABEL,UL	EA
0039	00001.000	2269943-0001	LABEL,CSA	EA
0041	00001.000	2223097-0001	CABLE ASSY,MOTHERBOARD TO FLOPPY ----------000	EA
0046	00001.000	0999456-9701	MANUAL,INFORMATION REQUEST FORM 1225-9456-000	EA
0048	00001.000	2223203-0001	MANUAL-GETTING STARTED 1261-3203-000	EA
0050	REF	2223279-0001	CONFIGURATION,FLOPPY DISK DRIVES	EA

DWG NO 2223051 SH 1

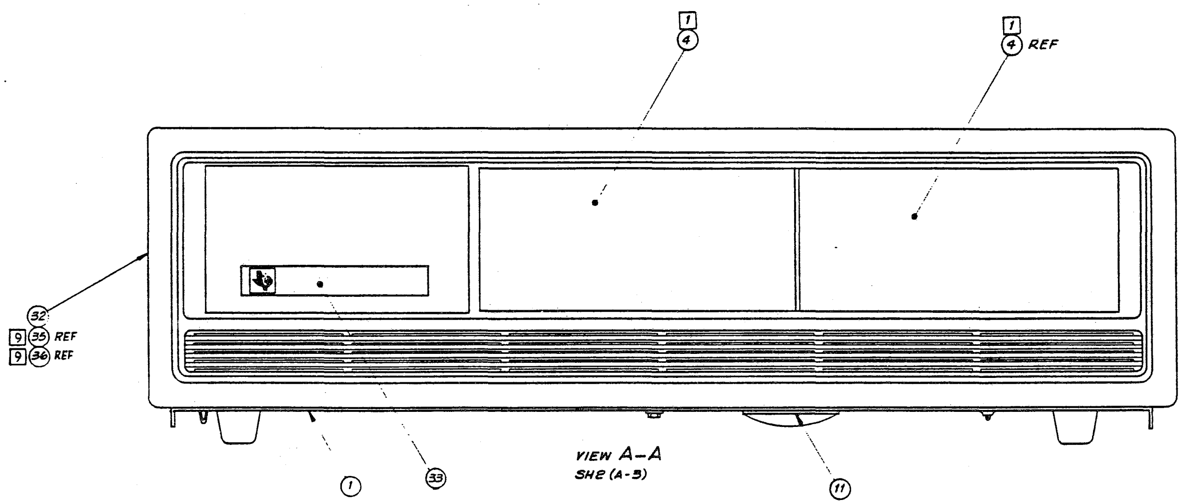
NOTES: UNLESS OTHERWISE SPECIFIED:

1. FACE OF INSERT PLATE (ITEM 4) MUST BE BUTT AGAINST OPENING OF FRONT PANEL BEFORE TIGHTENING (ITEM 14)
2. IF CONNECTORS P21 AND P22 ARE NOT USED, SECURE WITH ITEM 29 AS SHOWN
3. LOCATION OF LABELS (ITEM 9 AND 46) IS IS NOT CRITICAL
4. POWER CORD (ITEM 6) IS TO BE PACKED LOOSE WITH UNIT
5. SELF DESTRUCT LABEL (ITEM 10) TO BE PLACED APPROXIMATELY THIS LOCATION
6. MANUAL (ITEMS 46 & 48) TO BE PACKED LOOSE WITH UNIT
7. CAUTION STATIC SENSITIVE DEVICE ON THIS ASSEMBLY, PROPER GROUNDING REQUIRED
8. SECURE ITEM 41 WITH CLIP PROVIDED, WHEN NOT IN USE
9. TORQUE TO 0.5 ± 0.1 NM
10. TORQUE TO 0.7 ± 0.1 NM
11. TORQUE TO 1.1 ± 0.1 NM
12. SET MOTHERBOARD OPTION JUMPER PLUGS AS FOLLOWS: CONNECT 1 EACH TO E2, E4 CONNECT E5 TO E6, AND E19 TO E20

REVISIONS

REV	DESCRIPTION	DATE	APPROVED
A	(1) UPDATED LM	6/22/82	[signature]
B	CN 501137 (D)	7-13-82	M Bull
	(1) ALL LM ADD IT 29 & IT 46		
C	CN497471 (D) LAMB	9-29-82	M Bull
	(1) DELE IT 5 QTY 1 (2) ADD IT 48 (3) NOTE 6 WAS ...(ITEMS 5 & 46)...		
D	CN 501608(D)	10-20-82	M Bull
	(1) CHANGE PER ENG REDESIGN		

29



VIEW A-A
SHE (A-3)

PART NO.	DESCRIPTION
2223051-0003	SYSTEM ASSY, INTERNATIONAL - 115V
2223051-0002	SYSTEM ASSY, INTERNATIONAL - VDE
2223051-0001	SYSTEM ASSY, INTERNATIONAL - BPO

-3 QTY	-2 QTY	-1 QTY	ITEM NO	PART OR IDENTIFYING NUMBER	NOMENCLATURE OR DESCRIPTION	NOTES

PARTS LIST

NEXT ASSY	USED ON
2223040	8755

APPLICATION

UNLESS OTHERWISE SPECIFIED
- DIMENSIONS ARE IN MILLIMETERS
- TOLERANCES: 2 PLACE DECIMALS ±0.25, 1 PLACE DECIMALS ±0.5, ANGLES ±1°
- INTERPRET DRAWING PER DOD-D-1000
- REMOVE ALL BURRS AND SHARP EDGES
- CONCENTRICITY MACHINED DIAMETERS 0.25 FIM
- DIMENSIONAL LIMITS APPLY BEFORE PROCESSES
- PARENTHETICAL INFO FOR REF ONLY

HOLE TOLERANCE	
0.33 THRU 3.18	+0.10/-0.03
3.20 THRU 6.38	+0.13/-0.03
6.38 THRU 12.70	+0.18/-0.03
12.73 THRU 19.05	+0.20/-0.03
19.08 THRU 25.40	+0.25/-0.03
25.43 THRU 50.80	+0.30/-0.03

THIRD ANGLE PROJECTION

DWN [signature] 6-16-82
CHK M Bull 6-22-82
ENGR [signature] 6-22-82
APVD ENGR [signature] 6-22-82
QA [signature]
APVD [signature] 6-22-82
REL [signature] 6/23/82

TEXAS INSTRUMENTS INCORPORATED

SI-METRIC

SYSTEM ASSY, INTERNATIONAL

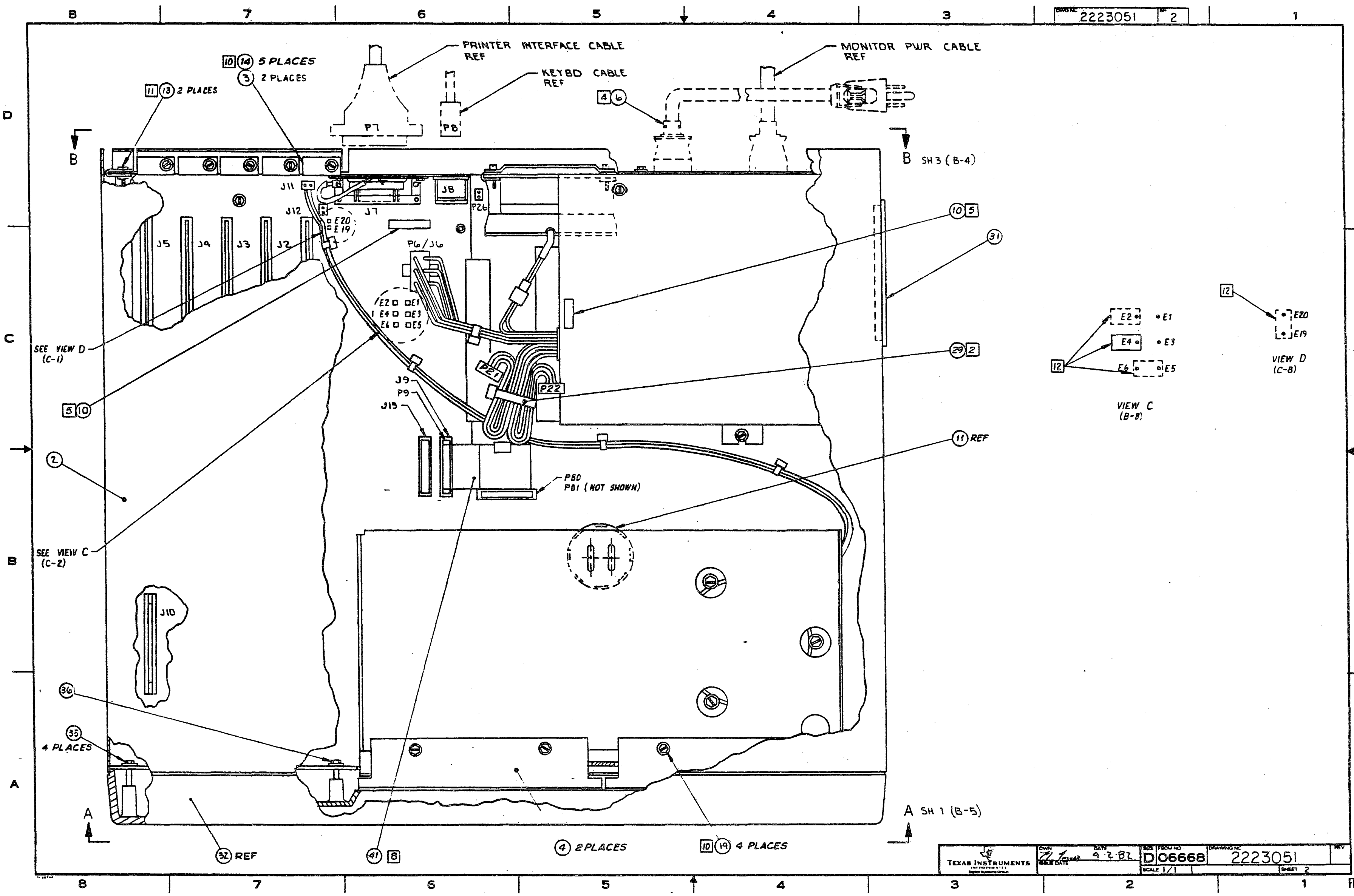
SIZE D | FSCM NO 06668 | DRAWING NO 2223051

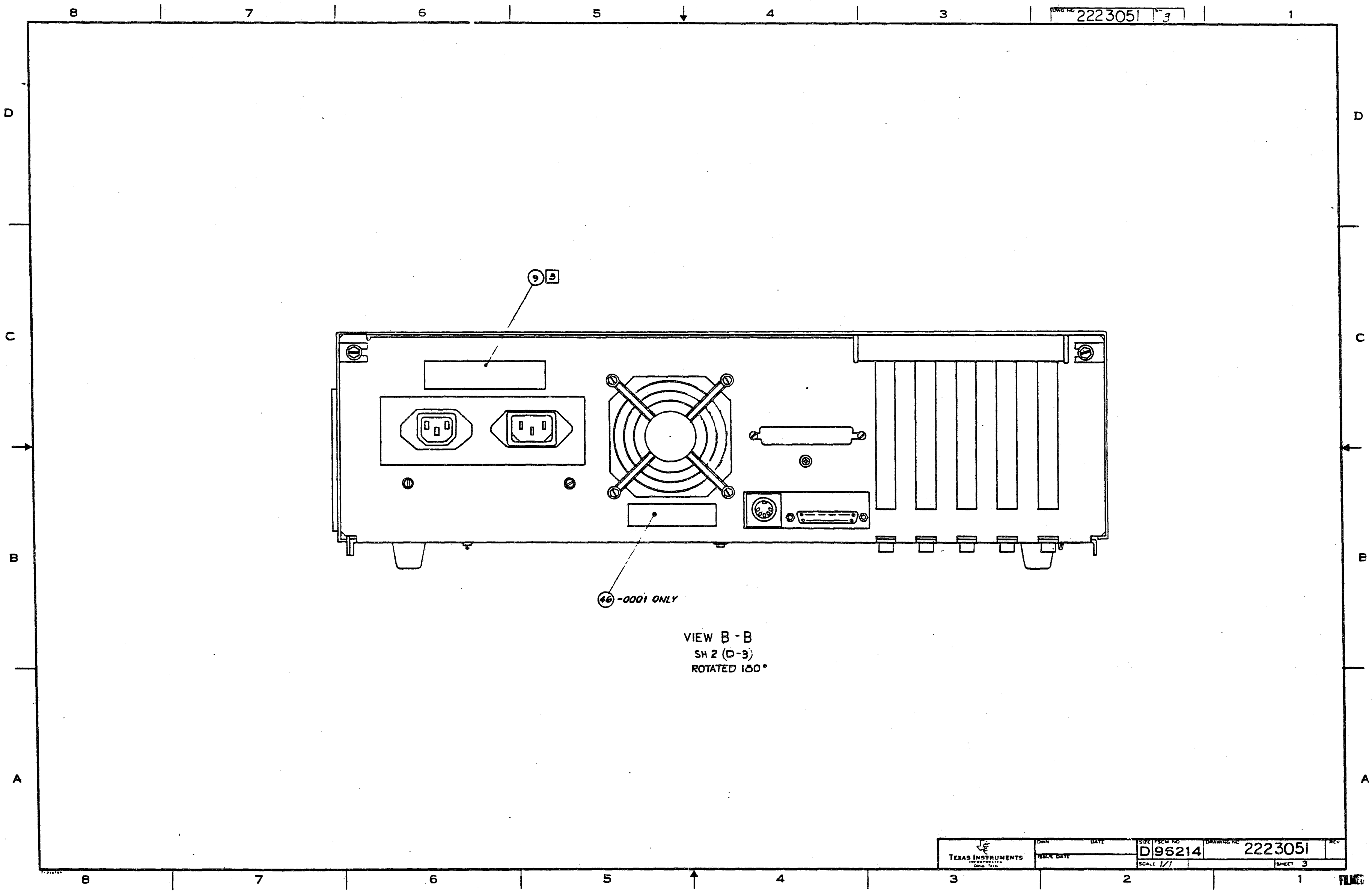
SCALE 1/1 | SHEET 1 OF 3

SEQ NO	IDENT	F-SPEC	NO	ADDITIONAL	NOTES	REV STATUS OF SHEETS	REV	D	D	D
	PROCESS		CLASSIFICATION				SH	1	2	3

PROCESSES — FOR CORRELATION TO GOVT AND SPECIFICATIONS SEE TI DRAWING 729467

4-300 CC LM


2223051
2
PRINTER INTERFACE CABLE REF
KEYBD CABLE REF
MONITOR PWR CABLE REF
10 14 5 PLACES
3 2 PLACES
11 13 2 PLACES
4 6
P7
P8
B
B SH 3 (B-4)
J11
J8
J12
J7
P26
E20
E19
J5
J4
J3
J2
P6/J6
10 5
31
E2 E1
E4 E3
E6 E5
SEE VIEW D (C-1)
5 10
29 2
P21
P22
J9
P9
J15
11 REF
P80
P81 (NOT SHOWN)
2
SEE VIEW C (C-2)
J10
36
35
4 PLACES
A
A SH 1 (B-5)
32 REF
41 8
4 2 PLACES
10 19 4 PLACES
12
E20
E19
VIEW D (C-8)
E2
E1
E4
E3
E6
E5
VIEW C (B-8)
TEXAS INSTRUMENTS
4-2-82
D 06668
2223051
SCALE 1/1
SHEET 2



9 5
46 -0001 ONLY
VIEW B - B
SH 2 (D-3)
ROTATED 180°
DWG NO 2223051 SH 3
TEXAS INSTRUMENTS
SIZE D
FSCM NO 96214
DRAWING NO 2223051
SCALE 1/1
SHEET 3

LIST OF MATERIALS

11/24/82

PART NUMBER	REV	DESCRIPTION
2223051-0001	D	SYSTEM ASSY-INT'L BPO

ITEM.	QUANTITY.	COMPONENT..	DESCRIPTION	UM
0001	00001.000	2223038-0002	MAIN ENCLOSURE SUBASSY-BPO 1669-2038-000	EA
0002	00001.000	2223029-0001	COVER,TERMINAL 1678-3029-006	EA
0003	00005.000	2223033-0001	PLATE OPTION BOARD 1678-3133-009	EA
0004	00002.000	2223034-0001	INSERT PLATE,FLOPPY 1678-3134-008	EA
0006	00001.000	0996695-0001	CABLE,POWER W/O PLUG (INTL.) 080126-107-2-093	EA
0009	00001.000	2223075-0003	LABEL,SERIAL-950 TERM,BASIC BPO 1669-3075-000	EA
0010	AR	0996943-0001	LABEL, SELF-DESTRUCT, .656 X .25 1652-1274-000	EA
0011	00001.000	2211919-0002	PLUG,HOLE-1.563 DIA SEE TI- DRAWING	EA
0013	00002.000	0972988-0043	SCREW 8-32 X .375 PAN HEAD CRES	EA
0014	00009.000	0996741-0006	6-20 X 3/8 SEMS SCREW TYPE B SEE TI- DRAWING	EA
0025	REF	2223082-0001	INTERCONNECT DIAGRAM	EA
0029	00001.000	0972632-0001	STRAP,TIE DOWN,CABLE-NON-STD,0-1-1/4 D.	EA
0031	00001.000	2223076-0001	INSERT SWITCH OPENING 1255-3519-002	EA
0032	00001.000	2223020-0001	PANEL,FRONT 1255-3521-002	EA
0033	00001.000	2223090-0001	NAMEPLATE,PROFESSIONAL COMPUTER ----------000	EA
0035	00004.000	0972969-0009	SCREW,6-20 X 7/8 HEX WASHER HEAD SEE TI- DRAWING	EA
0036	00001.000	0972969-0008	SCREW,6-20 X 3/4 HEX WASHER HEAD	EA
0041	00001.000	2223097-0001	CABLE ASSY,MOTHERBOARD TO FLOPPY ----------000	EA
0046	00001.000	0999456-9701	MANUAL,INFORMATION REQUEST FORM 1225-9456-000	EA
0047	00001.000	2222574-0001	LABEL,CAUTION (BPO)	EA
0048	00001.000	2223203-0001	MANUAL-GETTING STARTED 1261-3203-000	EA

LIST OF MATERIALS

11/24/82

PART NUMBER	REV	DESCRIPTION..............................
2223051-0002	D	SYSTEM,ASSY-INT'L VDE

ITEM.	QUANTITY.	COMPONENT..	DESCRIPTION..............................	UM
0001	00001.000	2223038-0003	MAIN ENCLOSURE SUB ASSY-VDE 1669-3038-000	EA
0002	00001.000	2223029-0001	COVER,TERMINAL 1678-3029-006	EA
0003	00005.000	2223033-0001	PLATE OPTION BOARD 1678-3133-009	EA
0004	00002.000	2223034-0001	INSERT PLATE,FLOPPY 1678-3134-008	EA
0006	00001.000	0996290-0001	CORDSET,POWR-WEST EURO-RT ANGLE PLUG	EA
0009	00001.000	2223075-0004	LABEL,SERIAL-950 TERM,BASIC VDE 1669-4075-000	EA
0010	AR	0996943-0001	LABEL, SELF-DESTRUCT, .656 X .25 1652-1274-000	EA
0011	00001.000	2211919-0002	PLUG,HOLE-1.563 DIA SEE TI- DRAWING	EA
0013	00002.000	0972988-0043	SCREW 8-32 X .375 PAN HEAD CRES	EA
0014	00009.000	0996741-0006	6-20 X 3/8 SEMS SCREW TYPE B SEE TI- DRAWING	EA
0025	REF	2223082-0001	INTERCONNECT DIAGRAM	EA
0029	00001.000	0972632-0001	STRAP,TIE DOWN,CABLE-NON-STD,0-1-1/4 D.	EA
0031	00001.000	2223076-0001	INSERT SWITCH OPENING 1255-3519-002	EA
0032	00001.000	2223020-0001	PANEL,FRONT 1255-3521-002	EA
0033	00001.000	2223090-0001	NAMEPLATE,PROFESSIONAL COMPUTER ----------000	EA
0035	00004.000	0972969-0009	SCREW,6-20 X 7/8 HEX WASHER HEAD SEE TI- DRAWING	EA
0036	00001.000	0972969-0008	SCREW,6-20 X 3/4 HEX WASHER HEAD	EA
0041	00001.000	2223097-0001	CABLE ASSY,MOTHERBOARD TO FLOPPY ----------000	EA
0046	00001.000	0999456-9701	MANUAL,INFORMATION REQUEST FORM 1225-9456-000	EA
0048	00001.000	2223203-0001	MANUAL-GETTING STARTED 1261-3203-000	EA

LIST OF MATERIALS

11/24/82

PART NUMBER	REV	DESCRIPTION...........................
2223051-0003	D	SYSTEM ASSY-INT'L 115V

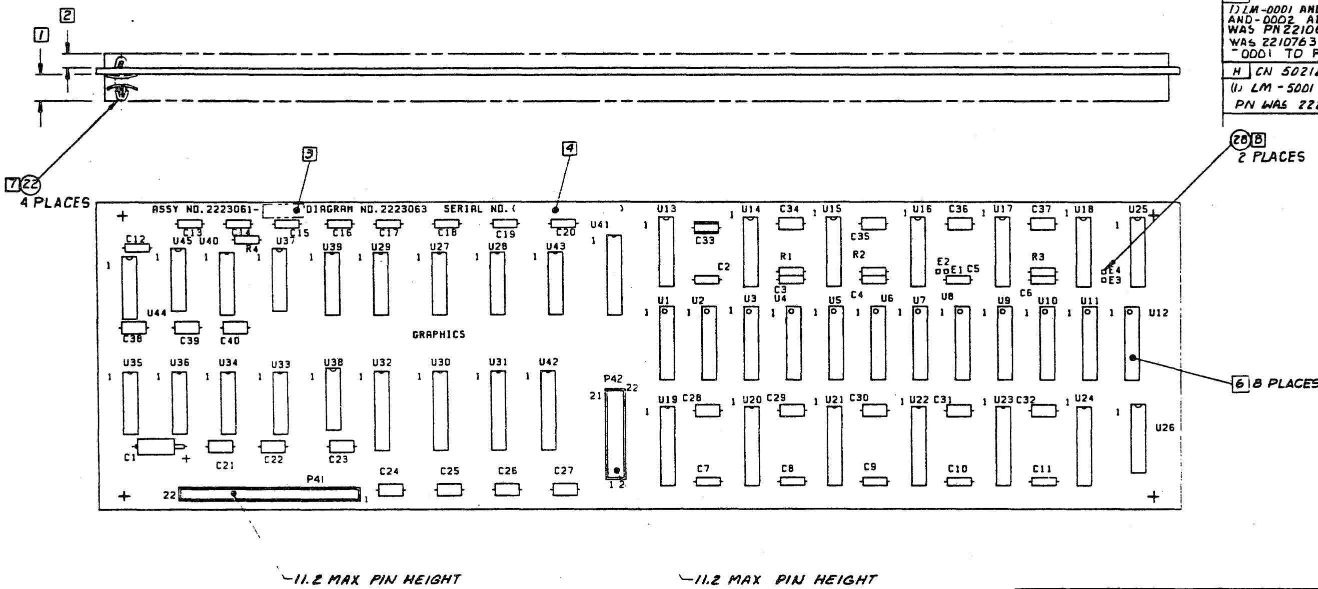
ITEM.	QUANTITY.	COMPONENT..	DESCRIPTION...........................	UM
0001	00001.000	2223038-0001	MAIN ENCLOSURE,SUBASSY 1669-1038-000	EA
0002	00001.000	2223029-0001	COVER,TERMINAL 1678-3029-006	EA
0003	00005.000	2223033-0001	PLATE OPTION BOARD 1678-3133-009	EA
0004	00002.000	2223034-0001	INSERT PLATE,FLOPPY 1678-3134-008	EA
0006	00001.000	0996289-0001	CORD SET,3-PIN PWR-DOMESTIC BLACK 080126-0-7889-008-GY	EA
0007	00000.000	0996289-0002	CORD SET,3-PIN PWR-DOMESTIC GRAY W/CLIP 080126-0-7919-008-GY	EA
0007A			*MAY BE USED AS AN 080126-0-7919-008-GY	
0007B			*ALTERNATE TO ITEM 6. 080126-0-7919-008-GY	
0009	00001.000	2223075-0005	LABEL,SERIAL-950 TERMINAL,BASIC 1669-5075-000	EA
0010	AR	0996943-0001	LABEL, SELF-DESTRUCT, .656 X .25 1652-1274-000	EA
0011	00001.000	2211919-0002	PLUG,HOLE-1.563 DIA SEE TI- DRAWING	EA
0013	00002.000	0972988-0043	SCREW 8-32 X .375 PAN HEAD CRES	EA
0014	00009.000	0996741-0006	6-20 X 3/8 SEMS SCREW TYPE B SEE TI- DRAWING	EA
0025	REF	2223082-0001	INTERCONNECT DIAGRAM	EA
0029	00001.000	0972632-0001	STRAP,TIE DOWN,CABLE-NON-STD,0-1-1/4 D.	EA
0031	00001.000	2223076-0001	INSERT SWITCH OPENING 1255-3519-002	EA
0032	00001.000	2223020-0001	PANEL,FRONT 1255-3521-002	EA
0033	00001.000	2223090-0001	NAMEPLATE,PROFESSIONAL COMPUTER ----------000	EA
0035	00004.000	0972969-0009	SCREW,6-20 X 7/8 HEX WASHER HEAD SEE TI- DRAWING	EA
0036	00001.000	0972969-0008	SCREW,6-20 X 3/4 HEX WASHER HEAD	EA
0041	00001.000	2223097-0001	CABLE ASSY,MOTHERBOARD TO FLOPPY ----------000	EA
0046	00001.000	0999456-9701	MANUAL,INFORMATION REQUEST FORM 1225-9456-000	EA
0048	00001.000	2223203-0001	MANUAL-GETTING STARTED 1261-3203-000	EA

NOTES: UNLESS OTHERWISE SPECIFIED

1. MAXIMUM COMPONENT HEIGHT ABOVE THE BOARD SHALL BE 5.6
2. MAXIMUM LEAD LENGTH BELOW BOARD SHALL BE 2.3
3. MARK APPROPRIATE DASH NUMBER AND REVISION LETTER IN SPACE INDICATED PER PROCESS 3
4. MARK SITE DATE CODE AND SERIAL NUMBER IN SPACE INDICATED PER ITEM 24 PARAGRAPH 4.0 AND PROCESS 3
5. CAUTION THIS ASSEMBLY AND CERTAIN COMPONENTS ARE SUSCEPTIBLE TO DAMAGE FROM ELECTROSTATIC DISCHARGE. OPERATOR AND EQUIPMENT GROUND AND PACKAGE IS REQUIRED. STATIC SENSITIVE COMPONENTS ARE: U1 THRU U45
6. COMPONENTS U5, U6, U7, U8, U9, U10, U11, U12 ARE USED ON -0002 ONLY
7. INSTALL ITEM 22 AFTER UNIT TEST
8. INSTALL ITEM 28 BETWEEN E2-E1 AND E4-E3 ON -0001 ASSEMBLY ONLY

DWG NO 2223061

REVISIONS			
REV	DESCRIPTION	DATE	APPROVED
A	LM UPDATE	6-7-82	
B	LM UPDATE	6-7-82	
C	CN 502108 (D) Block (1) LM UPDATE	8-3-82	M Biall
D	CN 500533 (D) RTF (1) LM-18-2 DELE IT1 P/N 2223061 QTY 1 (2) LM-5001 & -5002 IT 18 WAS 418356	8-10-82	M Biall
E	CN 497440 (C) RT (1) LM-5001 & -5002 IT 3 WAS 2220517-2 (2) LM-5001 & -5002 ADD IT 26	8-31-82	M Biall
F	CN 502187 (B) LAMB (1) UPDATE REV LEVEL	9-30-82	M Biall
G	CN502197 (D) S DAVIS 1) LM-0001 AND-0002 IT 999 WAS IT 101 AND 102 (2) LM-0001 AND-0002 ADDED IT 27 AND 28 (3) LM 5001 AND 5002 IT 4 WAS PN 2210669-1, ITEM 8 WAS PN 2210658-1, IT 13 PN WAS 2210763-1 (4) ADDED NOTE 7 AND 8 (5) ADD PN 2223273-0001 TO REV LEVEL BLOCK	10-27-82	M Biall
H	CN 502182 (B) Blac (1) LM-5001 & 5002 IT 8 PN WAS 2223265-1 (2) IT 13 PN WAS 2223264-1 (3) IT 4 PN WAS 2223268-1	11-4-82	M Biall



2223061-5002	GRAPHICS CRT CONTROLLER - 3 PLANE AUTO INSER
2223061-5001	GRAPHICS CRT CONTROLLER AUTO INSERTER
2223061-0002	GRAPHICS CRT CONTROLLER - 3 PLANE
2223061-0001	GRAPHICS CRT CONTROLLER
PART NUMBER	DESCRIPTION

-3 QTY	-2 QTY	-1 QTY	ITEM NO	PART OR IDENTIFYING NUMBER	NOMENCLATURE OR DESCRIPTION	NOTES

PARTS LIST

SEQ NO	IDENT	F-SPEC	NO	ADDITIONAL CLASSIFICATION	NOTES
3	MARK	902-01	01	HGT 1.5 CLR WHT (CAT 4)	3 4
2	SLDR	127-01	00	HAND SOLDER	
1	SLDR	124-02	00	WAVE SOLDER	

PROCESS — CLASSIFICATION
PROCESSES — FOR CORRELATION TO GOVT AND SPECIFICATIONS SEE TI DRAWING 72948

REVISION LEVEL BLOCK						
UNIT TEST	2223273-0001	G	G	G	G
ASSEMBLY	2223061-0001	E	F	G	H
PWB	2223062-0001	B	B	B	B
DIAGRAM	2223063-0001	A	A	A	A

REV STATUS OF SHEETS: REV / SH

NEXT ASSY	USED ON
2223040	8755

APPLICATION

UNLESS OTHERWISE SPECIFIED
- DIMENSIONS ARE IN MILLIMETERS
- TOLERANCES 2 PLACE DECIMALS ±0.25; 1 PLACE DECIMALS ±0.5, ANGLES ±1°
- INTERPRET DRAWING PER DOD-D-1000
- REMOVE ALL BURRS AND SHARP EDGES
- CONCENTRICITY MACHINED DIAMETERS 0.25 FIM
- DIMENSIONAL LIMITS APPLY BEFORE PROCESSES
- PARENTHETICAL INFO FOR REF ONLY

HOLE TOLERANCE	
0.33 THRU 3.18	+0.10/-0.03
3.20 THRU 6.36	+0.13/-0.03
6.38 THRU 12.70	+0.15/-0.03
12.73 THRU 19.05	+0.20/-0.03
19.08 THRU 25.40	+0.25/-0.03
25.43 THRU 50.80	+0.30/-0.03

THIRD ANGLE PROJECTION

DWN E. DeVoie 5-3-82
CHK M Biall 6-4-82
ENGR 6-4-82
APVD ENGR 6-4-82

TEXAS INSTRUMENTS INCORPORATED — SI-METRIC

GRAPHICS CRT CONTROLLER

SIZE D FSCM NO 06668 DRAWING NO 2223061
SCALE NONE SHEET

320

LM / CC 1

LIST OF MATERIALS

11/24/82

PART NUMBER	REV	DESCRIPTION...........................
2223061-0001	H	GRAPHICS,CRT CONTROLLER

ITEM.	QUANTITY.	COMPONENT..	DESCRIPTION...........................	UM
0002	REF	2223063-0001	DIAGRAM,LOGIC,GRAPHICS CRT CONTROLLER	EA
0007	00001.000	2210653-0001	IC,LS138,3-TO-8 LINE DECODER V-LIST-LS138 BURN-IN	EA
0007A			U26 V-LIST-LS138 BURN-IN	
0021	00001.000	2210288-0022	HEADER,1-ROW,22 CONTACTS, .100" CENTERS SEE TI- DRAWING	EA
0021A			P41 SEE TI- DRAWING	
0022	00004.000	0996341-0003	SPACER,PC BOARD,ZYTEL,NATURAL COLOR	EA
0023	00001.000	2210057-0011	HEADER, STR. PIN, 22 POS 007791--87215-7	EA
0023A			P42 007791--87215-7	
0024	REF	0994396-9901	PROCEDURE,SITE & DATE CODE SERIALIZATION	EA
0027	REF	2223273-0001	SPECIFICATION,UNIT TEST-GRAPHICS CRT	EA
0028	AR	0411400-0024	WIRE, 24AWG ELECTRO TIN PLATED COPPER	FT
0999	00001.000	2223061-5001	GRAPHICS,CRT CONTROLLER-AUTO INSERT 1254-3061-002	EA
9999	00001.000	0239999-9999	COST, SHRINKAGE	EA

11/24/82

PART NUMBER	REV	DESCRIPTION...........................
2223061-0002	H	GRAPHICS,CRT CONTROLLER 3 PLANE

ITEM.	QUANTITY.	COMPONENT..	DESCRIPTION...........................	UM
0002	REF	2223063-0001	DIAGRAM,LOGIC,GRAPHICS CRT CONTROLLER	EA
0007	00001.000	2210653-0001	IC,LS138,3-TO-8 LINE DECODER V-LIST-LS138 BURN-IN	EA
0007A			U26 V-LIST-LS138 BURN-IN	
0021	00001.000	2210288-0022	HEADER,1-ROW,22 CONTACTS, .100" CENTERS SEE TI- DRAWING	EA
0021A			P41 SEE TI- DRAWING	
0022	00004.000	0996341-0003	SPACER,PC BOARD,ZYTEL,NATURAL COLOR	EA
0023	00001.000	2210057-0011	HEADER, STR. PIN, 22 POS 007791--87215-7	EA
0023A			P42 007791--87215-7	
0024	REF	0994396-9901	PROCEDURE,SITE & DATE CODE SERIALIZATION	EA
0027	REF	2223273-0001	SPECIFICATION,UNIT TEST-GRAPHICS CRT	EA
0999	00001.000	2223061-5002	GRAPHICS,CRT CONTROLLER 3 PLANE-AUTO INS 1254-3063-004	EA
9999	00001.000	0239999-9999	COST, SHRINKAGE	EA

LIST OF MATERIALS

11/24/82

PART NUMBER REV DESCRIPTION..............................
2223061-5001 H GRAPHICS,CRT CONTROLLER-AUTO INSERT

ITEM.	QUANTITY.	COMPONENT..	DESCRIPTION..............................	UM
0001	00001.000	2223062-0001	PWB,GRAPHICS CRT CONTROLLER SEE TI- DRAWING	EA
0003	00004.000	2220517-0002	IC,16KX4BIT RAM,330NSEC READ CYCLE TIME SEE TI- DRAWING	EA
0003A			U1,U2,U3,U4 SEE TI- DRAWING	
0004	00006.000	2210669-0001	IC,LS166,8-BIT PARALLEL/SERIAL INPUT V-LIST-LS166 BURN-IN	EA
0004A			U13,U14,U15,U16,U17,U18 V-LIST-LS166 BURN-IN	
0005	00001.000	2223084-0001	GRAPHICS LOGIC ARRAY 1254- -000	EA
0005A			U41 1254- -000	
0006	00006.000	2210695-0001	IC,LS245,OCTAL BUS,XCIVER,3ST.OUTPUT V-LIST-LS245 BURN-IN	EA
0006A			U19,U20,U21,U22,U23,U24 V-LIST-LS245 BURN-IN	
0007	00001.000	2210653-0001	IC,LS138,3-TO-8 LINE DECODER V-LIST-LS138 BURN-IN	EA
0007A			U25 V-LIST-LS138 BURN-IN	
0008	00003.000	2210658-0001	IC,LS151,1-OF-8 DATA SELECTOR/MULTIPLEXE V-LIST-LS151 BURN-IN	EA
0008A			U27,U28,U29 V-LIST-LS151 BURN-IN	
0009	00003.000	2210702-0001	IC,LS273,OCTAL,D-FLIP-FLOP W/COM CLOCK V-LIST-LS273 BURN-IN	EA
0009A			U30,U31,U32 V-LIST-LS273 BURN-IN	
0010	00004.000	2210659-0001	IC,LS153,DUAL 4-LINE TO 1-L DATA SEL/MPX V-LIST-LS153 BURN-IN	EA
0010A			U33,U34,U35,U36 V-LIST-LS153 BURN-IN	
0011	00002.000	2210727-0001	IC,LS393,DUAL,4-BIT BINARY COUNTER V-LIST-LS393 BURN-IN	EA
0011A			U37,U38 V-LIST-LS393 BURN-IN	
0012	00001.000	2210720-0001	IC,LS373,OCTAL D-TYPE LATCHES V-LIST-LS373 BURN-IN	EA
0012A			U42 V-LIST-LS373 BURN-IN	
0013	00001.000	2210763-0001	IC,S174,HEX,FLIP-FLOP,SINGLE RAIL OUTPUT V-LIST-S174 BURN-IN	EA
0013A			U43 V-LIST-S174 BURN-IN	
0014	00002.000	2210667-0001	IC,LS163,SYNC 4-BIT BINARY CNT,SYNC CLR V-LIST-LS163 BURN-IN	EA
0014A			U39,U40 V-LIST-LS163 BURN-IN	
0015	00001.000	2210735-0001	IC,S00,QUAD,2-INPUT NAND V-LIST-S00 BURN-IN	EA
0015A			U44 V-LIST-S00 BURN-IN	
0016	00001.000	2210604-0001	IC,LS04,HEX INVERTERS V-LIST-LS04 BURN-IN	EA
0016A			U45 V-LIST-LS04 BURN-IN	
0017	00004.000	0972946-0081	RES FIX 4.7K OHM 5 % .25 W CARBON FILM ROH - R-25	EA
0017A			R1,R2,R3,R4 ROH - R-25	
0018	00001.000	0972924-0018	CAP FIX TANT SOLID 6.8 MFD 10 % 35 VOLT QPL -M39003/1-2304	EA

LIST OF MATERIALS

11/24/82

PART NUMBER	REV	DESCRIPTION
2223061-5001	H	GRAPHICS,CRT CONTROLLER-AUTO INSERT

ITEM.	QUANTITY.	COMPONENT..	DESCRIPTION	UM
0018A			C1 QPL -M39003/1-2304	
0019	00019.000	0972763-0013	CAP,FIXED .010UF 50 VOLTS 004222-MC105E103Z	EA
0019A			C2,C3,C4,C5,C6,C7,C8,C9,C10 004222-MC105E103Z	
0019B			C11,C12,C13,C14,C15,C16,C17 004222-MC105E103Z	
0019C			C18,C19,C20 004222-MC105E103Z	
0020	00020.000	0972763-0025	CAPACITOR,.10UF 50V FX,CERAMIC DIEL COR CA-C03Z5U104Z050A	EA
0020A			C21,C22,C23,C24,C25,C26,C27 COR CA-C03Z5U104Z050A	
0020B			C28,C29,C30,C31,C32,C33,C34 COR CA-C03Z5U104Z050A	
0020C			C35,C36,C37,C38,C39,C40 COR CA-C03Z5U104Z050A	
0025	00001.000	0972946-0027	RES FIX 27.0 OHM 5 % .25 W.CARBON FILM ROH - R-25	EA
0025A			R5 ROH - R-25	
0026	00000.000	2220517-0001	IC,16K X 4-BIT,RAM,260NSEC READ CYCLE T SEE TI- DRAWING	EA
0026A			*THIS ITEM MAY BE USED AS SEE TI- DRAWING	
0026B			*AN ALTERNATE TO ITEM 3. SEE TI- DRAWING	

11/24/82

PART NUMBER	REV	DESCRIPTION
2223061-5002	H	GRAPHICS,CRT CONTROLLER 3 PLANE-AUTO INS

ITEM.	QUANTITY.	COMPONENT..	DESCRIPTION	UM
0001	00001.000	2223062-0001	PWB,GRAPHICS CRT CONTROLLER SEE TI- DRAWING	EA
0003	00012.000	2220517-0002	IC,16KX4BIT RAM,330NSEC READ CYCLE TIME SEE TI- DRAWING	EA
0003A			U1,U2,U3,U4,U5,U6,U7,U8,U9, SEE TI- DRAWING	
0003B			U10,U11,U12 SEE TI- DRAWING	
0004	00006.000	2210669-0001	IC,LS166,8-BIT PARALLEL/SERIAL INPUT V-LIST-LS166 BURN-IN	EA
0004A			U13,U14,U15,U16,U17,U18 V-LIST-LS166 BURN-IN	
0005	00001.000	2223084-0001	GRAPHICS LOGIC ARRAY 1254- -000	EA
0005A			U41 1254- -000	
0006	00006.000	2210695-0001	IC,LS245,OCTAL BUS,XCIVER,3ST.OUTPUT V-LIST-LS245 BURN-IN	EA
0006A			U19,U20,U21,U22,U23,U24 V-LIST-LS245 BURN-IN	
0007	00001.000	2210653-0001	IC,LS138,3-TO-8 LINE DECODER V-LIST-LS138 BURN-IN	EA
0007A			U25 V-LIST-LS138 BURN-IN	
0008	00003.000	2210658-0001	IC,LS151,1-OF-8 DATA SELECTOR/MULTIPLEXE V-LIST-LS151 BURN-IN	EA
0008A			U27,U28,U29 V-LIST-LS151 BURN-IN	

LIST OF MATERIALS

11/24/82

PART NUMBER	REV	DESCRIPTION...........................
2223061-5002	H	GRAPHICS,CRT CONTROLLER 3 PLANE-AUTO INS

ITEM.	QUANTITY.	COMPONENT..	DESCRIPTION............................	UM
0009	00003.000	2210702-0001	IC,LS273,OCTAL,D-FLIP-FLOP W/COM CLOCK V-LIST-LS273 BURN-IN	EA
0009A			U30,U31,U32 V-LIST-LS273 BURN-IN	
0010	00004.000	2210659-0001	IC,LS153,DUAL 4-LINE TO 1-L DATA SEL/MPX V-LIST-LS153 BURN-IN	EA
0010A			U33,U34,U35,U36 V-LIST-LS153 BURN-IN	
0011	00002.000	2210727-0001	IC,LS393,DUAL,4-BIT BINARY COUNTER V-LIST-LS393 BURN-IN	EA
0011A			U37,U38 V-LIST-LS393 BURN-IN	
0012	00001.000	2210720-0001	IC,LS373,OCTAL D-TYPE LATCHES V-LIST-LS373 BURN-IN	EA
0012A			U42 V-LIST-LS373 BURN-IN	
0013	00001.000	2210763-0001	IC,S174,HEX,FLIP-FLOP,SINGLE RAIL OUTPUT V-LIST-S174 BURN-IN	EA
0013A			U43 V-LIST-S174 BURN-IN	
0014	00002.000	2210667-0001	IC,LS163,SYNC 4-BIT BINARY CNT,SYNC CLR V-LIST-LS163 BURN-IN	EA
0014A			U39,U40 V-LIST-LS163 BURN-IN	
0015	00001.000	2210735-0001	IC,S00,QUAD,2-INPUT NAND V-LIST-S00 BURN-IN	EA
0015A			U44 V-LIST-S00 BURN-IN	
0016	00001.000	2210604-0001	IC,LS04,HEX INVERTERS V-LIST-LS04 BURN-IN	EA
0016A			U45 V-LIST-LS04 BURN-IN	
0017	00004.000	0972946-0081	RES FIX 4.7K OHM 5 % .25 W CARBON FILM ROH - R-25	EA
0017A			R1,R2,R3,R4 ROH - R-25	
0018	00001.000	0972924-0018	CAP FIX TANT SOLID 6.8 MFD 10 % 35 VOLT QPL -M39003/1-2304	EA
0018A			C1 QPL -M39003/1-2304	
0019	00019.000	0972763-0013	CAP,FIXED .010UF 50 VOLTS 004222-MC105E103Z	EA
0019A			C2,C3,C4,C5,C6,C7,C8,C9,C10 004222-MC105E103Z	
0019B			C11,C12,C13,C14,C15,C16,C17 004222-MC105E103Z	
0019C			C18,C19,C20 004222-MC105E103Z	
0020	00020.000	0972763-0025	CAPACITOR,.10UF 50V FX,CERAMIC DIEL COR CA-C03Z5U104Z050A	EA
0020A			C21,C22,C23,C24,C25,C26,C27 COR CA-C03Z5U104Z050A	
0020B			C28,C29,C30,C31,C32,C33,C34 COR CA-C03Z5U104Z050A	
0020C			C35,C36,C37,C38,C39,C40 COR CA-C03Z5U104Z050A	
0025	00001.000	0972946-0027	RES FIX 27.0 OHM 5 % .25 W.CARBON FILM ROH - R-25	EA
0025A			R5 ROH - R-25	
0026	00000.000	2220517-0001	IC,16K X 4-BIT,RAM,260NSEC READ CYCLE T SEE TI- DRAWING	EA
0026A			*THIS ITEM MAY BE USED AS SEE TI- DRAWING	
0026B			*AN ALTERNATE TO ITEM 3. SEE TI- DRAWING	

NOTES: UNLESS OTHERWISE SPECIFIED.

REV	DESCRIPTION	DATE	APPROVED
A	CN 501646 (D) Glow (1) CHNG PER ENG REDESIGN	10-27-82	M Bile

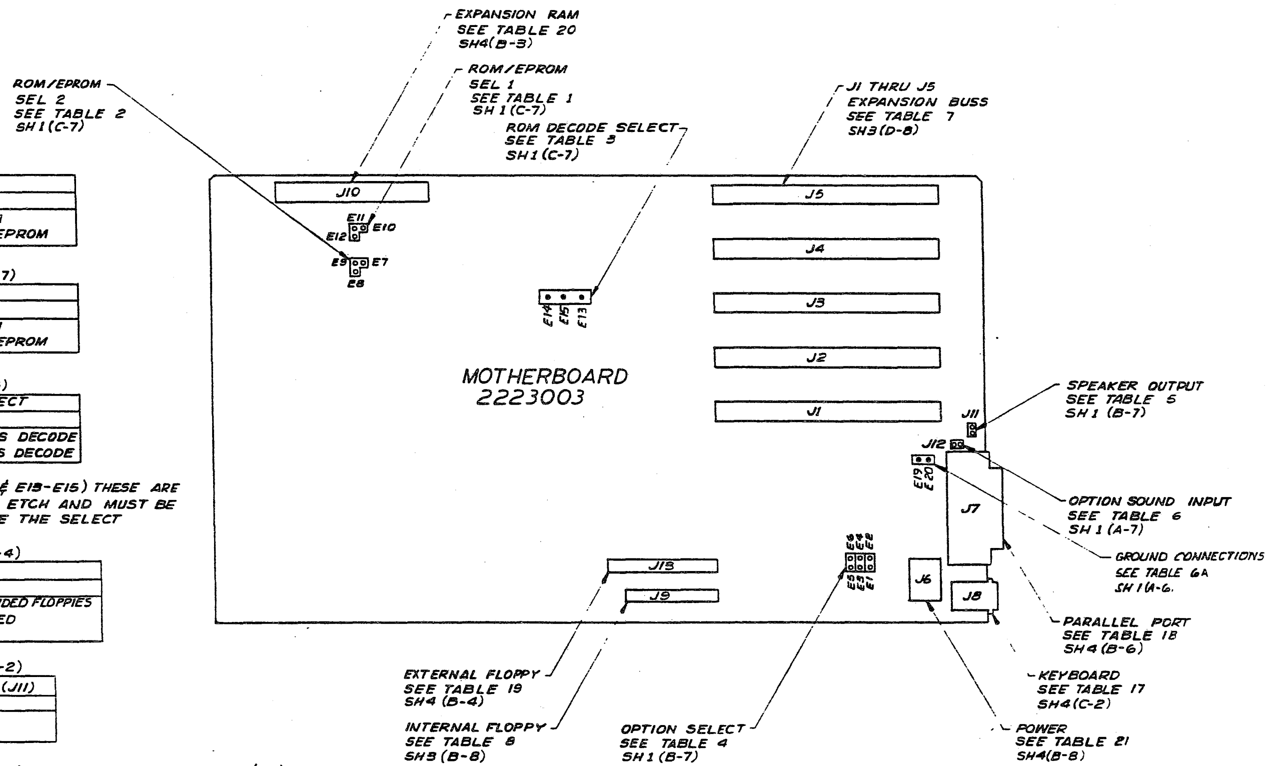


TABLE 1 SH1 (D-5)

ROM/EPROM SEL 1	
JUMPER	OPTIONS
E11-E10	U63 = ROM
E11-E12	U63 = 4K EPROM

TABLE 2 SH1 (D-7)

ROM/EPROM SEL 2	
JUMPER	OPTIONS
E9-E7	U62=ROM
E8-E9	U62 = 4K EPROM

TABLE 3 SH1 (D-5)

ROM DECODE SELECT	
JUMPER	OPTIONS
E13-E14	4K ROMS DECODE
E13-E15	8K ROMS DECODE

NOTE: (E10-E11, E9-E8, & E13-E15) THESE ARE CONNECTED BY ETCH AND MUST BE CUT TO CHANGE THE SELECT

TABLE 4 SH1 (A-4)

OPTION SELECT	
JUMPER	OPTIONS
E1-E2	DOUBLE SIDED FLOPPIES
E3-E4	RESERVED
E5-E6	50 HZ

TABLE 5 SH1 (C-2)

SPEAKER OUTPUT (J11)	
PIN NO.	SIGNAL
1	SPEAKER
2	GROUND

TABLE 6 SH1 (B-2)

OPTION SOUND INPUT (J12)	
PIN NO.	SIGNAL
1	SOUND IN
2	GROUND

TABLE 6A SH1 (B-3)

GROUND CONNECTIONS	
PIN NO.	SIGNAL
E19	LOGIC GROUND
E20	CHASSIS GROUND

PARTS LIST

QTY	QTY	QTY	ITEM NO	PART OR IDENTIFYING NUMBER	NOMENCLATURE OR DESCRIPTION	NOTES

NEXT ASSY	USED ON
2223051	8755

APPLICATION

TEXAS INSTRUMENTS INCORPORATED

SI-METRIC

ELECTRICAL PIN CONFIGURATION

SIZE D FSCM NO 06668 DWG NO 2223082

SCALE NONE SHEET 1 OF 4

SEQ NO	IDENT	F-SPEC	NO	ADDITIONAL
	PROCESS			CLASSIFICATION

PROCESSES — FOR CORRELATION TO GOVT AND SPECIFICATIONS SEE TI DRAWING 72965

REV STATUS OF SHEETS	REV	A	A		A
	SH	1	2	3	4

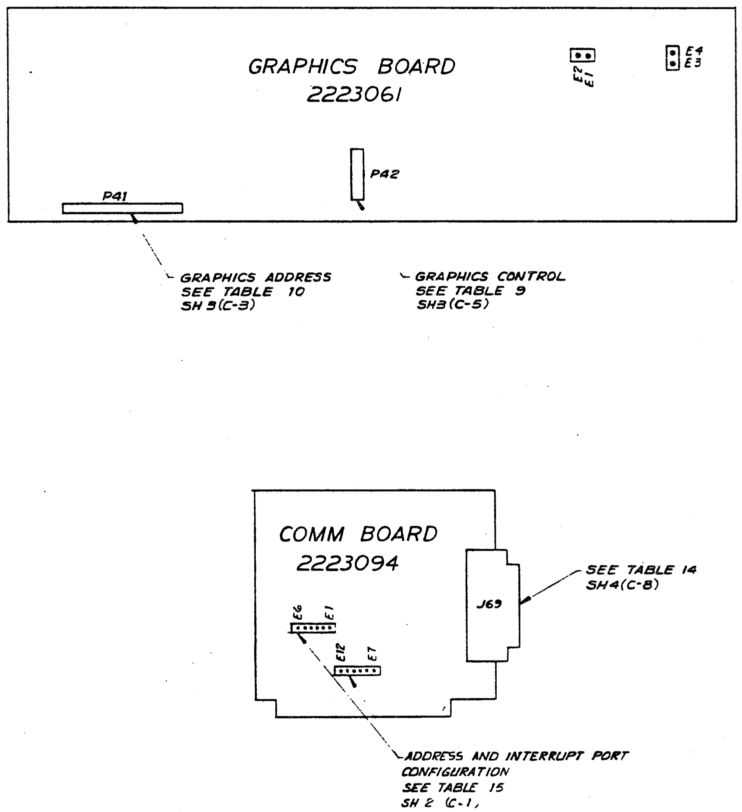


TABLE 15 SH 2 (A-4)

COMM BOARD PORT CONFIG			
PORT #	I/O ADDRESS	JUMPER OPTIONS	INTERRUPT LEVEL
1	ØEØ - ØE7	E1-E2, E7-E8	Ø
2	ØE8 - ØEF	E2-E3, E8-E9	1
3	ØFØ - ØF7	E4-E5, E10-E11	2
4	ØF8 - ØFF	E5-E6, E11-E12	4

TABLE 22 SH D-3 & D-4

BANK SELECT	
JUMPER	OPTION
E1-E2	BLANK GRAPHICS BANK 2
E3-E4	BLANK GRAPHICS BANK 3

TEXAS INSTRUMENTS | DATE 8/13/82 | SIZE D | FSCM NO 96214 | DRAWING NO 2223082 | REV A | SCALE NONE | SHEET 2

DWG. 2223082 SH 3

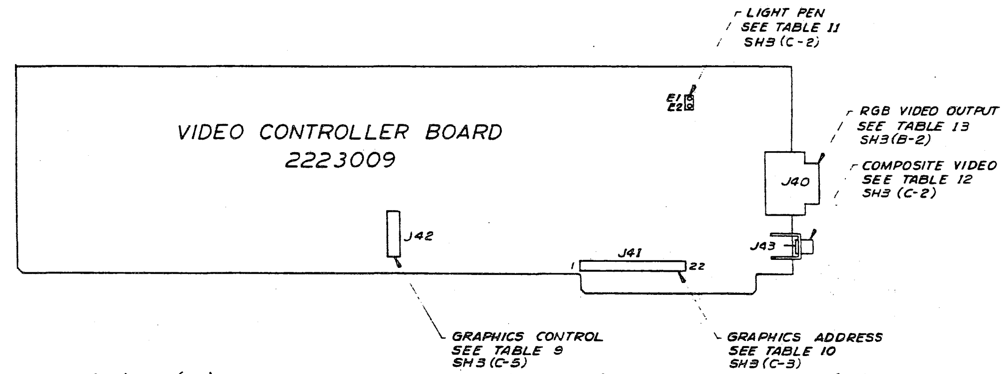


TABLE 7 SH1 (D-3)

EXPANSION BUSS CONNECTORS

PIN NO.	SIGNAL	PIN NO	SIGNAL
1/A01	NMI-	32/B16	
2/B01	GND	33/A17	XA14
3/A02	XD7	34/B17	
4/B02	RESET	35/A18	XA13
5/A03	XD6	36/B18	
6/B03	+5V	37/A19	XA12
7/A04	XD5	38/B19	
8/B04	IR0	39/A20	XA11
9/A05	XD4	40/B20	CLOCK
10/B05		41/A21	XA10
11/A06	XD3	42/B21	IR6
12/B06		43/A22	XA9
13/A07	XD2	44/B22	IR5
14/B07	-12V	45/A23	XA8
15/A08	XD1	46/B23	IR4
16/B08	DMA-	47/A24	XA7
17/A09	XD0	48/B24	IR1
18/B09	+12V	49/A25	XA6
19/A10	WAIT-	50/B25	IR2
20/B10	GND	51/A26	XA5
21/A11	RFSH	52/B26	
22/B11	AMWC-	53/A27	XA4
23/A12	XA19	54/B27	
24/B12	MRDC-	55/A28	XA3
25/A13	XA18	56/B28	ALE
26/B13	AIOWC-	57/A29	XA2
27/A14	XA17	58/B29	+5V
28/B14	IORC-	59/A30	XA1
29/A15	XA16	60/B30	OSC
30/B15		61/A31	XA0
31/A16	XA15	62/B31	GND

TABLE 8 SH1 (A-5)

INTERNAL FLOPPY CONNECTOR (J9)

PIN NO.	SIGNAL	FLOPPY "A" PIN NO.	FLOPPY "B" PIN NO.
1	GND	1	1
2		2	2
3	GND	3	3
4		4	4
5	GND	5	5
6		6	6
7	GND	7	7
8	INDEX-	8	8
9	GND	9	9
10	SEL1-	10	10
11	GND	11	11
12	SEL2-	12	12
13	GND	13	13
14		14	14
15	GND	15	15
16	MOT-	16	16
17	GND	17	17
18	DIRECT	18	18
19	GND	19	19
20	STEP-	20	20
21	GND	21	21
22	WDATA 1-	22	22
23	GND	23	23
24	WGATE -	24	24
25	GND	25	25
26	TRAK Ø -	26	26
27	GND	27	27
28	WPROT-	28	28
29	GND	29	29
30	RDATA -	30	30
31	GND	31	31
32	SIDI-	32	32
33	GND	33	33
34		34	34

TABLE 9 SH3(C-3), SH2(C-4)

GRAPHICS CONTROL (P42)

PIN NO.	SIGNAL	P42 PIN NO.	SIGNAL
1	N.C.	2	N.C.
2	GR2-	1	GGNL-
3	GR1-	4	GRDL-
4	GR3-	3	GBLL-
5	XD7	6	D7
6	GRED -	5	GRED-
7	XD6	8	D6
8	GBLU-	7	GBLU-
9	XD5	10	D5
10	GGRN-	9	GGRN-
11	XD4	12	D4
12	DCLK	11	DCLK
13	XD3	14	D3
14	RD-	13	RD-
15	XD2	16	D2
16	WR-	15	WR-
17	XD1	18	D1
18	GSEL	17	GSEL-
19	XD0	20	D0
20	DDE	19	DE
21	GWAIT-	22	GWAIT-
22	RESET-	21	RST-

TABLE 10 SH3(C-2) SH2(C-5)

GRAPHICS ADDRESS (P41)

PIN NO.	SIGNAL	P41 PIN NO.	SIGNAL
1	A0	1	A0
2	A1	2	A1
3	A2	3	A2
4	A3	4	A3
5	A4	5	A4
6	A5	6	A5
7	A6	7	A6
8	A7	8	A7
9	A8	9	A8
10	A9	10	A9
11	A10	11	A10
12	A11	12	A11
13	A12	13	A12
14	A13	14	A13
15	A14	15	A14
16	A15	16	A15
17	A16	17	A16
18	N.C.	18	N.C.
19	+5V	19	+5V
20	+5V	20	+5V
21	GND	21	GND
22	GND	22	GND

TABLE 11 SH3(D-1)

LIGHT PIN

PIN NO.	SIGNAL
1	INPUT
2	GND

TABLE 12 SH3(D-1)

COMPOSITE VIDEO

PIN	SIGNAL
BODY	GND
CENTER	OUTPUT

TABLE 13 SH3(D-1)

RGB VIDEO OUTPUT

PIN NO.	SIGNAL
1	GND
2	GND
3	VRED
4	VGRN
5	VBLU
6	GND
7	N.C.
8	HDRIVE
9	VDRIVE

TEXAS INSTRUMENTS | DATE 5/13/82 | SIZE D | FSCM NO. 96214 | DWG NO. 2223082 | SCALE NONE | SHEET 3

D C B A

TABLE 14 SH2(B-4)

J69	
PIN NO.	SIGNAL
1	CHASSIS GND
2	TXD
3	RXD
4	RTS
5	CTS
6	DSR
7	SIGNAL GND
8	DCD
9	
10	
11	SCA/CH
12	SCF/CI
13	
14	
15	TXC
16	
17	RXC
18	
19	
20	DTR
21	
22	RI
23	SCA/CH
24	BAUDOUT/DA

~~TABLE 15 SH2(B-6)~~

(COMMUNICATIONS BOARD) ADDRESS CONFIG *	
I/O ADDRESS	JUMPER OPTIONS
FFØX	E15-E18, E16-E19, E17-E20
FF2X	E3-E6, E15-E18, E16-E19
FF4X	E2-E5, E15-E18, E17-E20
FF6X	E2-E5, E3-E6, E15-E18
FF8X	E1-E4, E16-E19, E17-E20
FFAX	E1-E4, E16-E19, E3-E6
FFCX	E1-E4, E2-E5, E17-E20
FFEX	E1-E4, E2-E5, E3-E6

~~* THESE ARE THE ONLY ALLOWABLE JUMPER CONFIGURATIONS~~

~~TABLE 16 SH2(A-4)~~

INTERRUPT LEVEL CONFIGURATIONS *	
LEVEL	JUMPER OPTIONS *
1	E7-E8
2	E9-E10
4	E11-E12
5	E13-E14

~~* ONLY ONE JUMPER ALLOWED~~

TABLE 17 SH 1 (B-2)

KEYBOARD (J8)			
PIN NO.	SIGNAL	KEYBOARD PWB CONN	COLOR
1	PWR (+12V)	E2	GRN
2	RCV DATA	E3	RED
3	XMIT DATA	E4	BLK
4	LOOPBACK DATA	E5	BLU
5	GND	E1	YEL

TABLE 18 SH1 (B-2)

PARALLEL PORT (J7)	
PIN NO.	SIGNAL
1	PTSTR-
2	PD1
3	PD2
4	PD3
5	PD4
6	PD5
7	PD6
8	PD7
9	PD8
10	PACK-
11	PBUSY
12	PAPOUT
13	PSELECT
14	PTAF-
15	PFAULT
16	PTINIT-
17	GND
18	↑
19	│
20	│
21	│
22	│
23	│
24	↓
25	GND

TABLE 19 SH1 (B-5)

EXTERNAL FLOPPY CONNECTOR (J13)					
PIN NO.	SIGNAL	37 PIN D CONN	PIN NO.	SIGNAL	37 PIN D CONN
1	GND	20	21	GND	30
2		1	22	DIRECTZ-	11
3	GND	21	23	GND	31
4		2	24	STEPZ-	12
5	GND	22	25	GND	32
6		3	26	WDATA2-	13
7	GND	23	27	GND	33
8		4	28	WGATE2-	14
9	GND	24	29	GND	34
10		5	30	TRAKØ-	15
11	GND	25	31	GND	35
12	INDEX-	6	32	WPORT-	16
13	GND	26	33	GND	36
14	MOT3-	7	34	RDATA-	17
15	GND	27	35	GND	37
16	SEL4-	8	36	SID1-	18
17	GND	28	37	GND	
18	SEL3-	9	38		19
19	GND	29	39	GND	
20	MOT4-	10	40		

TABLE 20 SH1 (D-5)

EXPANSION RAM CONECTOR (J10)			
PIN NO.	SIGNAL	PIN NO.	SIGNAL
1		23	
2		24	MA4
3		25	
4	MD7	26	PØ
5	GND	27	
6	GND	28	MA6
7		29	
8	AMWC-	30	PE
9	+5V	31	CAS-
10	+5V	32	MDØ
11		33	MD2
12	MA1	34	MD1
13		35	MD4
14	MA5	36	MD3
15		37	
16	MA7	38	MD5
17		39	
18	MAØ	40	MD6
19		41	+5V
20	MA2	42	+5V
21		43	GND
22	MA3	44	GND

TABLE 21 SH1 (A-3)

POWER CONN (J6)	
PIN NO.	SIGNAL
1	+12V
2	+5V
3	GND
4	-12V
5	+5V
6	GND

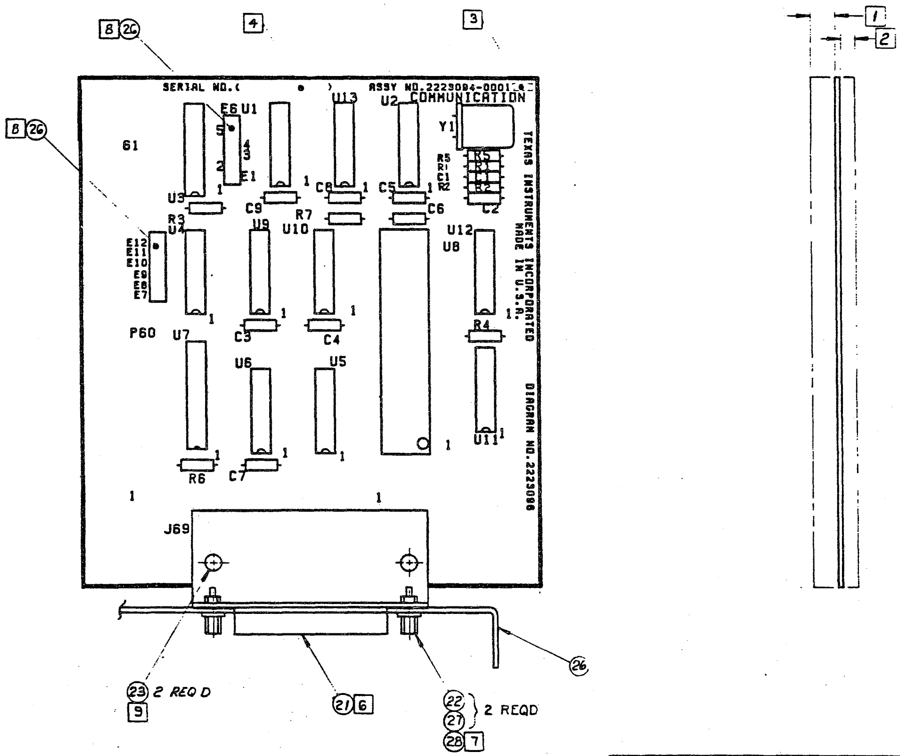
TEXAS INSTRUMENTS INCORPORATED	DWN [signature] DATE 5/13/82	SIZE D	FSCM NO. 96214	DRAWING NO. 2223082	REV. A
	ISSUE DATE	SCALE NONE		SHEET 4	

FILMED

NOTES: UNLESS OTHERWISE SPECIFIED.

1. MAXIMUM COMPONENT HEIGHT ABOVE THE BOARD SHALL BE 19.0
2. MAXIMUM LEAD LENGTH BELOW THE BOARD SHALL BE 2.3
3. MARK APPROPRIATE ~~DASH NUMBER AND~~ REVISION LETTER IN SPACE INDICATED PER PROCESS 3
4. MARK SITE DATE CODE AND SERIAL NUMBER IN SPACE INDICATED PER ITEM 25 PARAGRAPH 4.0 AND PROCESS 3
5. CAUTION THIS ASSEMBLY AND CERTAIN COMPONENTS ARE SUSCEPTIBLE TO DAMAGE FROM ELECTROSTATIC DISCHARGE. OPERATOR AND EQUIPMENT GROUND AND PACKAGE IS REQUIRED. STATIC SENSITIVE COMPONENT ARE: U1 THRU U11, Y1
6. REMOVE 2 PANHEAD SCREWS FROM ITEM 21 PRIOR TO ASSY AND DISCARD
7. APPLY ITEM 28 TO ITEM 22 AFTER ASSY
8. INSTALL ITEM 26 (QTY 2) 1 EACH TO E7-E8 AND E1-E2
9. INSTALL ITEM 23 FROM CONDUCTOR SIDE

REVISIONS			
REV	DESCRIPTION	DATE	APPROVED
A	LM UPDATE		
	FORMAL RELEASE		
B	CN 476726 (E) R. FLORES (1) LM UPDATE	6/25/82	M. BIEHL
C	CN 497454 (E) BLOCK (1) LM UPDATE	8/24/82	M. BIEHL
D	CN 502113 (D) LAMB (1) LM UPDATE (2) ADD NOTES 6 THRU 9 (3) ADD UNIT TEST TO REV LEVEL BLOCK	10-8-82	M. Biehl
E	CN 502400 (C) Block (1) LM UPDATE	11-4-82	M. Biehl



CONVERSION CHART	
mm	INCHES
0.25	.010
0.5	.02
1.5	.06
2.3	.09
19.0	.75

PART NO	DESCRIPTION
2223094-5001	COMMUNICATIONS CARD-AUTO INSERT
2223094-0001	COMMUNICATIONS CARD

-3 QTY	-2 QTY	-1 QTY	ITEM NO	PART OR IDENTIFYING NUMBER	NOMENCLATURE OR DESCRIPTION	NOTES

PARTS LIST

REV. LEVEL BLOCK						
UNIT TEST	2223274-0001				*	*
ASSEMBLY	2223094-0001	B	C	D	E	
PWB	2223095-0001	B	B	B	B	
DIAGRAM	2223096-0001	A	A	B	B	

SEQ NO	PROCESS	IDENT	F-SPEC	NO	ADDITIONAL	NOTES
3	MARK	902-01		01	HGT 1.5 CLR WHT (CAT 4)	3 4
2	SLDR	127-01		00	HAND SOLDER	
1	SLDR	124-02		00	WAVE SOLDER	

CLASSIFICATION

PROCESSES — FOR CORRELATION TO GOVT AND SPECIFICATIONS SEE TI DRAWING 729467

REV STATUS OF SHEETS: REV / SH

NEXT ASSY	USED ON
2223040	8755

APPLICATION

UNLESS OTHERWISE SPECIFIED
- DIMENSIONS ARE IN MILLIMETERS
- TOLERANCES: 2 PLACE DECIMALS ±0.25, 1 PLACE DECIMALS ±0.5, ANGLES ±1°
- INTERPRET DRAWING PER DOD-D-1000
- REMOVE ALL BURRS AND SHARP EDGES
- CONCENTRICITY MACHINED DIAMETERS 0.25 FIM
- DIMENSIONAL LIMITS APPLY BEFORE PROCESSES
- PARENTHETICAL INFO FOR REF ONLY

HOLE TOLERANCE — THIRD ANGLE PROJECTION

DRN: [signature] 4-7-82
CHK: MIKE BIEHL 5/25/82
ENGR: GRAY McCORD 5/26/82
APVD ENGR: GRAY McCORD 5/26/82
QA: RALPH CAPAN 5/26/82
APVD MFG: SUSAN BRIDGEN 5/26/82
REL: JIM MOON 5/26/82

TEXAS INSTRUMENTS INCORPORATED

SI-METRIC

COMMUNICATIONS CARD

SIZE D | FSCM NO 06668 | DWG NO 2223094

SCALE NONE | SHEET

LM

LIST OF MATERIALS

11/24/82

PART NUMBER REV DESCRIPTION.............................
2223094-0001 F COMMUNICATION CARD ASSEMBLY

ITEM.	QUANTITY.	COMPONENT..	DESCRIPTION.............................	UM
0002	REF	2223096-0001	DIA,LOGIC,DETAILED,COMMUNICATIONS CARD	EA
0010	00001.000	2220519-0001	IC,USART,SERIAL COMMUNICATIONS CONT 1254- -000	EA
0010A			U8 1254- -000	
0019	00001.000	2210835-0003	CRYSTAL QUARTZ, 4.9152 MHZ, HC181U SEE TI- DWG	EA
0019A			Y1 SEE TI- DWG	
0020	00002.000	2210288-0006	HEADER, 6-PINS .600 L,SNG ROW,STRT-POST SEE TI- DRAWING	EA
0020A			E1-E6,E7-E12 SEE TI- DRAWING	
0021	00001.000	2220488-0003	CONNECTOR,RECEPTACLE,PCB,25-PINS SEE TI- DRAWING	EA
0021A			J69 SEE TI- DRAWING	
0022	00002.000	0532348-0401	STUD, EXTENSION-CRES	EA
0023	00002.000	0972446-0013	RIVET,.116 DIA 5/16 LG DOME HD ALUM - - -000	EA
0024	REF	0994396-9901	PROCEDURE,SITE & DATE CODE SERIALIZATION	EA
0025	00001.000	2223033-0002	PLATE,OPTION BOARD,WITH CUT OUT 1678-3223-007	EA
0026	00002.000	0972487-0001	JUMPER PLUG,CONNECTOR BLACK 5935-0900-000	EA
0027	00002.000	0411100-0070	LOCKWASHER #4 INTERNAL TOOTH CRES QPL - MS35333-70	EA
0028	AR	0415804-0005	SEALING COMPOUND,ANAEROBIC-BLUE GRADE C	QT
0030	REF	2223274-0001	SPECIFICATION,UNIT TEST-COMMUNICATIONS	EA
0031	AR	0411435-0408	TAPE,INSULATION,ELECT.1/4 IN MMM - 56-1/4	RL
0999	00001.000	2223094-5001	AUTO-INSERTED PARTS LIST FOR -0001 1254-3095-003	EA
9999	00001.000	0239999-9999	COST, SHRINKAGE	EA

LIST OF MATERIALS

11/24/82

PART NUMBER	REV	DESCRIPTION..............................
2223094-5001	F	AUTO-INSERTED PARTS LIST FOR -0001

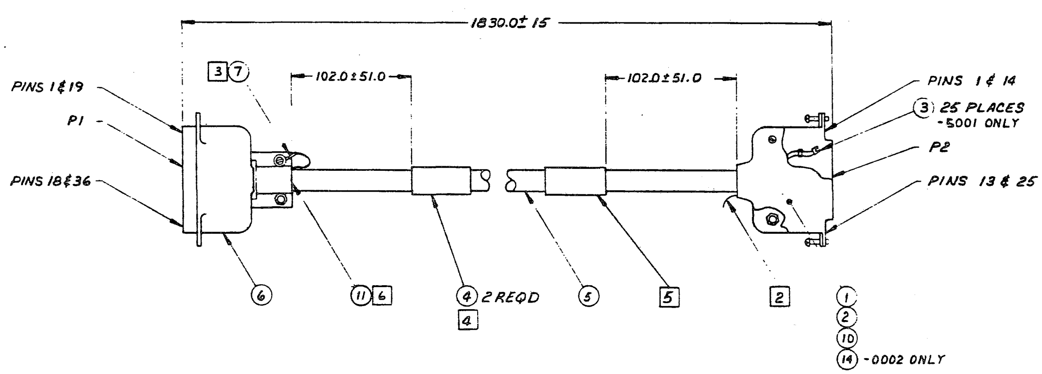
ITEM.	QUANTITY.	COMPONENT..	DESCRIPTION..............................	UM
0001	00001.000	2223095-0001	PWB COMMUNICATIONS CARD 1254- -000	EA
0003	00001.000	2210621-0001	IC,LS32,QUAD,2-INPUT OR V-LIST-LS32 BURN-IN	EA
0003A			U1 V-LIST-LS32 BURN-IN	
0004	00001.000	2210608-0001	IC,LS10,TRIPLE,3-INPUT NAND V-LIST-LS10 BURN-IN	EA
0004A			U2 V-LIST-LS10 BURN-IN	
0005	00001.000	2210654-0001	IC,LS139,DUAL 2-TO-4 LINE DECODER V-LIST-LS139 BURN-IN	EA
0005A			U3 V-LIST-LS139 BURN-IN	
0006	00001.000	2210600-0001	IC,LS00,QUAD,2-INPUT NAND V-LIST-LS00 BURN-IN	EA
0006A			U4 V-LIST-LS00 BURN-IN	
0007	00002.000	2210631-0001	IC,LS74,DUAL D FLIP-FLOP W/PSET & CLR V-LIST-LS74 BURN-IN	EA
0007A			U5,U13 V-LIST-LS74 BURN-IN	
0008	00001.000	2210602-0001	IC,LS02,QUAD,2-INPUT NOR V-LIST-LS02 BURN-IN	EA
0008A			U6 V-LIST-LS02 BURN-IN	
0009	00001.000	2210695-0001	IC,LS245,OCTAL BUS,XCIVER,3ST.OUTPUT V-LIST-LS245 BURN-IN	EA
0009A			U7 V-LIST-LS245 BURN-IN	
0011	00002.000	2211189-0001	IC,SN75188NP3,BURN-IN,QUADRUPLE LINE DRI SEE TI- DRAWING	EA
0011A			U9,U10 SEE TI- DRAWING	
0012	00002.000	2211349-0001	IC,SN75189AN3, QUAD LINE RECEIVERS SEE TI- DRAWING	EA
0012A			U11,U12 SEE TI- DRAWING	
0014	00001.000	0972946-0085	RES FIX 6.8K OHM 5 % .25 W CARBON FILM ROH - R-25	EA
0014A			R3 ROH - R-25	
0015	00006.000	0972946-0065	RES FIX 1.0K OHM 5% .25 W CARBON FILM ROH - R-25	EA
0015A			R1,R2,R4,R5,R6,R7 ROH - R-25	
0016	00001.000	2211247-0029	CAP,1000 PF,10%,50VDC,CERAMIC SEE TI- DRAWING	EA
0016A			C1 SEE TI- DRAWING	
0017	00001.000	2211247-0010	CAP,12.0 PF, 5%,50VDC,CERAMIC SEE TI- DRAWING	EA
0017A			C2 SEE TI- DRAWING	
0018	00007.000	0972763-0013	CAP,FIXED .010UF 50 VOLTS 004222-MC105E103Z	EA
0018A			C3,C4,C5,C6,C7,C8,C9 004222-MC105E103Z	

DWG NO 2223106

REV	DESCRIPTION	DATE	APPROVED
A	CN 502107 (D)	6-28-82	M Buis
	(1) LM-1 DELE IT3 539430-3 QTY 25 (2) LM-1 DELE IT5 2210505-7 QTY 6.5 (3) LM-1 ADD IT 101 (4) GENERATE LM-5001		
B	CN 503708 (D) R	9-21-82	M Buis
	(1) GENERATE LM-2 & -5002 (2) UPDATE LM-1 & -5001 (3) ADD NOTE 6 (4) NOTE 4 WAS MK "PRL PRINT" PER PROCS 1 (5) NOTE 5 WAS MK P/N & REV LTR PER PROCS 1 (6) DELE PRCS 1		

NOTES: UNLESS OTHERWISE SPECIFIED

1. CABLE CLAMP SCREWS AND RETAINER SCREWS INCLUDED WITH ITEM 2
2. P2 DRAIN WIRE FOLDED UNDER CLAM SHELLS OF CABLE CLAMP ASSEMBLY.
3. P1 DRAIN WIRE TO BE TERMINATED TO CABLE CLAMP SCREW BY SOLDER LUG ITEM 7
4. MARK PER ITEM 9; LINE 4 TEXT: PARALLEL PRINTER
5. MARK PER ITEM 9; LINE 4 TEXT: P2
6. ITEM 11 TO BE WRAPPED AROUND CABLE AND SECURED UNDER CABLE CLAMP



CONVERSION CHART	
mm	INCHES
0.25	0.010
0.5	0.02
3.18	0.125
102 ± 51.0	4.0 ± 2.0
1830 ± 15.0	72.0 ± 6.0

PART NO	DESCRIPTION
2223106-5002	CABLE ASSY PARALLEL, PRINTER, BULK/850
2223106-0002	CABLE ASSEMBLY, PARALLEL, PRINTER/850
2223106-5001	CABLE ASSY PARALLEL, PRINTER, BULK
2223106-0001	CABLE ASSEMBLY, PARALLEL, PRINTER

-3 QTY	-2 QTY	-1 QTY	ITEM NO	PART OR IDENTIFYING NUMBER	NOMENCLATURE OR DESCRIPTION	NOTES

PARTS LIST

SEQ NO	IDENT	F-SPEC	NO	ADDITIONAL	NOTES
	PROCESS			CLASSIFICATION	

PROCESSES — FOR CORRELATION TO GOVT/IND SPECIFICATIONS SEE TI DRAWING 72945

REV STATUS OF SHEETS — REV / SH

NEXT ASSY	USED ON
	8752
2223040	8755

APPLICATION

UNLESS OTHERWISE SPECIFIED
- DIMENSIONS ARE IN MILLIMETERS
- TOLERANCES 2 PLACE DECIMALS ± 0.25, 1 PLACE DECIMALS ± 0.5, ANGLES ± 1°
- INTERPRET DRAWING PER DOD-D-1000
- REMOVE ALL BURRS AND SHARP EDGES
- CONCENTRICITY MACHINED DIAMETERS 0.25 FIM
- DIMENSIONAL LIMITS APPLY BEFORE PROCESSES
- PARENTHETICAL INFO FOR REF ONLY

HOLE TOLERANCE	
0.33 THRU 3.18	+0.10/-0.03
3.20 THRU 6.35	+0.13/-0.03
6.38 THRU 12.70	+0.15/-0.03
12.73 THRU 19.05	+0.20/-0.03
19.08 THRU 25.40	+0.25/-0.03
25.43 THRU 50.80	+0.30/-0.03

THIRD ANGLE PROJECTION

DWN 5/14/82 · CHK · ENGR · APVD ENGR · QA · APVD MFG · RLSE

TEXAS INSTRUMENTS INCORPORATED — Digital Systems Group

SI-METRIC

CABLE ASSEMBLY, PARALLEL, PRINTER

SIZE	FSCM NO	DRAWING NO
D	06668	2223106

SCALE NONE · SHEET

LM

FILMED

LIST OF MATERIALS

11/24/82

PART NUMBER	REV	DESCRIPTION..............................
2223106-0001	B	CABLE ASSEMBLY,PARALLEL,PRINTER

ITEM.	QUANTITY.	COMPONENT..	DESCRIPTION..............................	UM
0001	00001.000	2220401-0003	CONNECTOR,PLUG,25X#20 AWG,SPRING	EA
0002	00001.000	2220380-0008	CABLE CLAMP ASSY,.400 IN. DIA. CABLE ACC SEE TI- DRAWING	EA
0006	00001.000	0414127-0001	CONNECTOR,PLUG-36 CONTACTS	EA
0007	00001.000	2220955-0001	SOLDER LUG, #4 SCREW 1254- -000	EA
0008	00001.000	2223107-0001	WIRE LIST PT TO PT PRL PTR CABLE ASSY	EA
0009	REF	2265070-0001	SPEC, PRE-PRINTED CABLE MARKER	EA
0010	00001.000	2220797-0001	FERRULE,CABLE CLAMP,SPLIT RING ALUMINUM SEE TI- DRAWING	EA
0101	00001.000	2223106-5001	BULK CABLE ASSY,PARALLEL,PRINTER 1650- -000	EA

11/24/82

PART NUMBER	REV	DESCRIPTION..............................
2223106-0002	B	CABLE ASSY,PARALLEL,PRINTER/850

ITEM.	QUANTITY.	COMPONENT..	DESCRIPTION..............................	UM
0001	00001.000	2220767-0002	CONNECTOR,PLUG,25 CONTACTS,2-ROW,22-26AG SEE TI- DRAWING	EA
0001A			P2 SEE TI- DRAWING	
0002	00001.000	2220380-0008	CABLE CLAMP ASSY,.400 IN. DIA. CABLE ACC SEE TI- DRAWING	EA
0006	00001.000	2220674-0001	CONNECTOR,RND CA TO PANEL,PLUG,STL SHELL SEE TI- DWG	EA
0006A			P1 SEE TI- DWG	
0008	REF	2223107-0001	WIRE LIST PT TO PT PRL PTR CABLE ASSY	EA
0009	REF	2265070-0001	SPEC, PRE-PRINTED CABLE MARKER	EA
0010	00001.000	2220797-0001	FERRULE,CABLE CLAMP,SPLIT RING ALUMINUM SEE TI- DRAWING	EA
0013	00000.000	0414127-0001	CONNECTOR,PLUG-36 CONTACTS	EA
0014	00001.000	2220827-0003	CONNECTOR,COVER,CAP,OR HOOD SEE TI- DRAWING	EA
0101	00001.000	2223106-5002	BULK CABLE ASSY PARALLEL 1620-8106-001	EA

11/24/82

PART NUMBER	REV	DESCRIPTION..............................
2223106-5001	B	BULK CABLE ASSY,PARALLEL,PRINTER

ITEM.	QUANTITY.	COMPONENT..	DESCRIPTION..............................	UM
0003	00025.000	0539430-0003	CONTACT,PIN 24-20AWG .068 INSUL DIA AMP -205202-2 ST	EA
0004	00002.000	2210317-0001	LABEL,BLANK,CABLE MARKER 085480-SLPF-19319-4	EA
0005	00006.500	2210505-0007	CABLE,SHIELDED,25 CONDUCTORS SEE TI- DRAWING	FT
0011	00000.130	0972361-0003	TAPE,FOAM,VINYL,SELF-ADH.25THK .50WIDE 012624-V548	RL

LIST OF MATERIALS

11/24/82

PART NUMBER	REV	DESCRIPTION
2223106-5002	B	BULK CABLE ASSY PARALLEL

ITEM.	QUANTITY.	COMPONENT..	DESCRIPTION	UM
0004	00002.000	2210317-0001	LABEL,BLANK,CABLE MARKER 085480-SLPF-19319-4	EA
0005	00007.000	2210505-0007	CABLE,SHIELDED,25 CONDUCTORS SEE TI- DRAWING	FT
0007	00001.000	2211389-0001	LUG,RING TONGUE,20-16AWG SEE TI- DRAWING	EA
0011	00000.130	0972361-0003	TAPE,FOAM,VINYL,SELF-ADH.25THK .50WIDE 012624-V548	RL

DWG NO 2223105 SH 1

APPLICATION	
NEXT ASSY	USED ON
2223046	8755

REVISIONS

REV	DESCRIPTION	DATE	APPROVED

1.0 SCOPE:

THIS SPECIFICATION COVERS THE REQUIREMENTS FOR A MONITOR CABLE.

2.0 APPLICABLE DOCUMENTS:

WHERE THIS SPECIFICATION REFERS TO ANOTHER DOCUMENT, THAT DOCUMENT IS OF THE ISSUE IN EFFECT ON THE DATE OF INVITATION TO BID OR REQUEST FOR PROPOSAL. REFERENCED DOCUMENT APPLY TO THE EXTENT SPECIFIED HEREIN. THIS SPECIFICATION GOVERNS WHEN A REFERENCED DOCUMENT CONFLICTS.

CONVERSION CHART	
mm	INCHES
0.25	.010
0.5	.02
3.81	.150
1219.2	48.00

SPECIFICATION CONTROL DRAWING

REV STATUS OF SHEETS				
REV				
SHEET	1	2	3	4

UNLESS OTHERWISE SPECIFIED
- DIMENSIONS ARE IN MILLIMETERS
- TOLERANCES: ANGLE ±1°, 1 PLACE DECIMALS ±0.5, 2 PLACE DECIMALS ±0.25
- PARENTHETICAL INFO FOR REF ONLY

THIRD ANGLE PROJECTION

		DATE
DWN	DUNHAM	5/12/82
CHK	M Bi[illegible]	6-10-82
ENGR	[illegible]	
APVD-ENGR	[illegible]	
QA		
APVD-MFG	[illegible]	
RLSE	[illegible]	6/11/82

TEXAS INSTRUMENTS INCORPORATED Digital Systems Group

SI-METRIC

INTERFACE CABLE, MONOCHROME MONITOR

SIZE A | FSCM NO 06668 | DRAWING NO 2223105

SCALE NONE | SHEET 1 OF 4

TI-25739

100

3.0 REQIREMENTS:

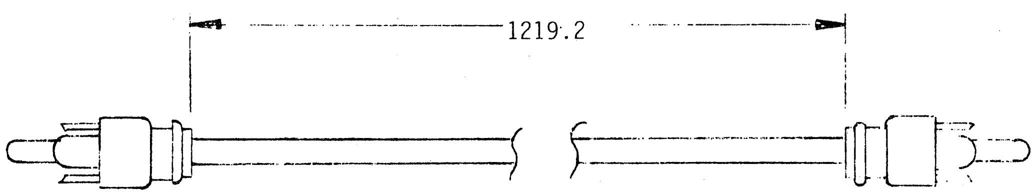
3.1 PHYSICAL: SEE FIGURE 1

3.1.1 CABLE MATERIAL:
ONE CONDUCTOR #27 AWG CONSISTING OF 7 STRANDS OF #56 AWG BARE COPPER WIRE OR 7 STRANDS OF #35 BARE COPPER COVERED STEEL WIRE. SHELD CONSISTS OF 4 ENDS OF #36 AWG TINNED COPPER SPIRAL WRAPPED OR BRAIDED COPPER WIRE. INTERNAL INSULATION OF POLYETHYLENE WITH OUTER JACKET AND CONNECTOR MOLDING TO BE LIGHT TAN IN COLOR MATCHING TI COLOR NUMBER 972939-2101. CABLE ASSEMBLY TO MEET THE REQUIREMENTS OF UL AND CSA.

3.1.2 MARKINGS:
PARTS OR WRAPPER SHALL BE MARKED WITH TEXAS INSTRUMENTS PART NUMBER .

3.1.3 IMPEDANCE:
CABLE IMPEDANCE SHALL BE 75 Ω NOMIMAL.

3.1.4 CONNECTORS:
BOTH ENDS OF THE SHIELDED CABLE SHALL BE TERMINATED EITHER WITH VICTOR PC-103 PHONO PLUGS OR BELDEN STYLE PHG761 SHORT STRAIGHT HANDLE PHONO PLUGS.



CABLE DIA. 3.81 NOM

FIGURE 1

	TEXAS INSTRUMENTS INCORPORATED DIGITAL SYSTEMS DIVISION HOUSTON, TEXAS	A	2223105 SHEET 2	REV

TI—4259-E

4.0 QUALITY ASSURANCE PROVISIONS:

4.1 RESPONSIBILITY FOR INSPECTION:

UNLESS OTHERWISE SPECIFIED IN THE CONTRACT OR PURCHASE ORDER, THE SUPPLIER SHALL BE RESPONSIBLE FOR PERFORMING INSPECTIONS THAT ARE SUFFICIENT TO ASSURE THAT THE PARTS SUPPLIED MEET THE REQUIREMENTS SPECIFIED HEREIN.

5.0 PREPARATION FOR DELIVERY:

5.1 PACKAGING:

PACKING AND WRAPPING SHALL BE SUFFICIENT TO PROTECT AGAINST DAMAGE OR LOSS DURING SHIPMENT FROM THE SUPPLIER TO THE DESTINATION SPECIFIED IN THE PURCHASE ORDER.

5.2 MARKING:

THE SHIPPING CONTAINER SHALL BE MARKED WITH THE TI PART NUMBER (SEE PART NUMBER BLOCK) AND THE COUNT CONTAINED. ADDITIONAL MARKING ARE PERMITTED.

SUGGESTED SOURCE(S) OF SUPPLY:

1. BELDEN CORPORATION
   P.O. BOX 1980
   RICHMOND, INDIANA 47374

2. VICTOR ELECTRIC WIRE & CABLE CO.
   618 MAIN ST.
   WEST WARWICK,R.I. 02893

TEXAS INSTRUMENTS PART NUMBER	MANUFACTURER'S PART NUMBERS		
	SOURCE 1	SOURCE 2	SOURCE 3
2223105-0001	IF-4310	TBD	

TEXAS INSTRUMENTS INCORPORATED
DIGITAL SYSTEMS DIVISION
HOUSTON, TEXAS

A	2223105	REV
	SHEET 4	

TI-4259-E

LIST OF MATERIALS

11/24/82

PART NUMBER	REV	DESCRIPTION
2207985-0001	C	TEST PLUG, EIA, COMMUNICATIONS

ITEM.	QUANTITY.	COMPONENT..	DESCRIPTION	UM
0001	00001.000	0539903-0001	HOOD,CONN 25 PIN WITH RETAINERS AMP - 206478-3	EA
0002	00001.000	0539409-0005	CONNECTOR,PLUG 25 PINS AMP -205208-1	EA
0002A			P1	
0003	00012.000	0539430-0003	CONTACT,PIN 24-20AWG .068 INSUL DIA AMP -205202-2 ST	EA
0004	00001.750	2210012-1999	WIRE,ELECT,WHT,26 AWG,19 X 38,U/L 1429 090484-SEE TI DWG	FT

LIST OF MATERIALS

11/24/82

PART NUMBER 2223099-0001 REV B

DESCRIPTION............................. PART NUMBER NOT AN ASSEMBLY

ITEM.	QUANTITY.	COMPONENT..	DESCRIPTION.............................	UM
0001	00009.000	2211118-0004	IC,64K-BIT DYNAMIC RAM,150NS TA/ROW TMS416-4-15NL	EA
0002	00001.000	2211752-0001	PLASTIC BAG, ANTI-STATIC SEE TI- DRAWING	EA
0003	AR	0970950-0003	URETHANE,SHEET SEE TI- DRAWING	EA
0004	00001.000	2223269-0001	CAUTION INSERT,RAM CHIP KIT	EA
0005	REF	0936660-0001	PEGASUS PACKAGING ASSY INDEX	EA

LIST OF MATERIALS

11/24/82

PART NUMBER REV DESCRIPTION...........................
2230528-0001 B KEYBOARD,TILTING,LOW PROFILE

ITEM.	QUANTITY.	COMPONENT..	DESCRIPTION...........................	UM
0001	00001.000	2230529-0001	BASE,KEYBOARD 1255-7500-001	EA
0002	00001.000	2230536-0001	HOUSING,SHAFT,RIGHT 1255-7504-001	EA
0003	00001.000	2230534-0001	HOUSING,SHAFT,LEFT 1255-7503-001	EA
0004	00002.000	2230532-0001	SHAFT,CLUTCH SPRING 1255-7502-001	EA
0005	00001.000	2230546-0001	SPRING,CLUTCH,RIGHT	EA
0006	00001.000	2230546-0002	SPRING,CLUTCH,LEFT	EA
0007	00001.000	2230547-0002	SPRING,RETURN,RIGHT	EA
0008	00001.000	2230547-0001	SPRING,RETURN,LEFT	EA
0009	00001.000	2230540-0001	FOOT,TILT ADJUSTMENT 1255-7506-001	EA
0010	00001.000	2230527-0001	KEYBOARD,LOW PROFILE	EA
0011	00001.000	2230530-0001	COVER,KEYBOARD 1255-7501-001	EA
0012	00002.000	2230538-0001	BUTTON,RELEASE 1255-7505-001	EA
0013	00002.000	2230554-0001	BRACKET,SPRING,BUTTON ----------000	EA
0014	00002.000	2230552-0001	CLIP,CLUTCH ----------000	EA
0015	00001.000	2230549-0001	CABLE ASSY,KEYBOARD	EA
0016	00001.000	2230553-0001	LABEL,SERIAL NO 1665-1553-000	EA
0017	00002.000	0972679-0029	SCREW	EA
0018	00012.000	0972679-0012	SCREW # 6-19 X 3/8 SLOTTED HEX	EA
0019	00002.000	0972679-0015	SCREW #6-19 X 3/4 THD SLOTTED HEX	EA
0020	00002.000	2230555-0007	RING,RETAINING	EA
0021	00001.000	0996943-0001	LABEL, SELF-DESTRUCT, .656 X .25 1652-1274-000	EA
0022	00002.000	2230556-0001	PAD,NONSKID,P/T	EA

Section 5

SCHEMATICS AND LOGIC DRAWINGS

This section contains schematic and logic drawings applicable to the Texas Instruments Professional Computer.

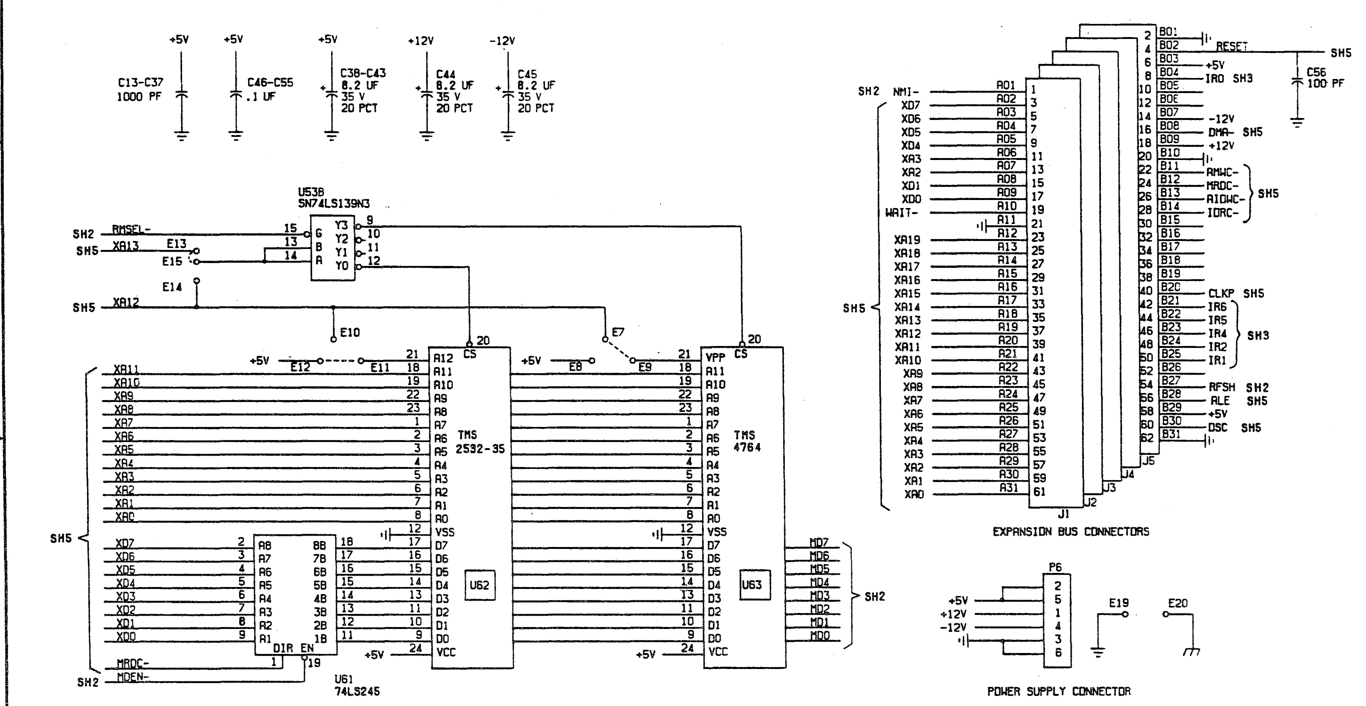
Drawings not available in time for printing:

Logic, Joystick	2223087*
Logic,Parallel Test Plug	2223278*

NOTES: UNLESS OTHERWISE SPECIFIED

1. RESISTORS ARE .25 W,5 PCT
2. CAPACITORS ARE 50 V, P80M20 PCT
3. 'NC' MEANS NOT CONNECTED

REV	DESCRIPTION	DATE	APPROVED
A	CN500540 (D) J Krivensky 25 AUG 82	9-17-82	



MOTHERBOARD ROMS & EXPANSION BUS

HIGHEST REFERENCE DESIGNATOR USED				
U67	C58	CR4	E20	J13
R48	Q2	Y1		

REFERENCE DESIGNATORS NOT USED				
E16				

TEXAS INSTRUMENTS INCORPORATED Dallas, Texas

DIAGRAM, LOGIC, DETAILED-
MOTHER BOARD

J. Krivensky 8/25/82; Pete Miller 4/30/82; Gary McCord 4/30/82; Gary McCord 4/30/82; Ralph Capan 4/30/82; Susan Bridgen 5/4/82; Mike Wolf 5/4/82

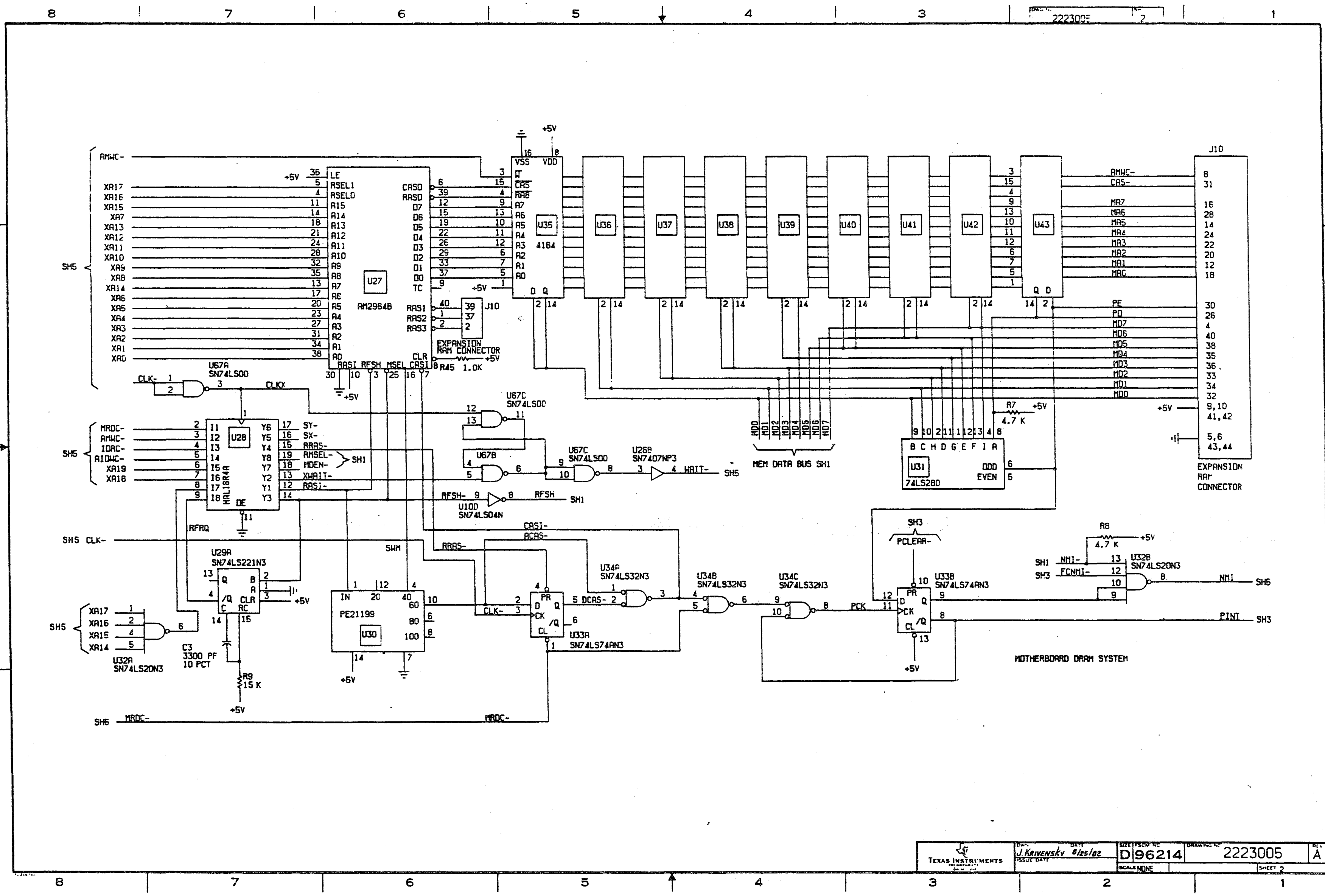
SIZE D | FSCM NO 96214 | DRAWING NO 2223005 | SCALE NONE | SHEET 1 OF 5

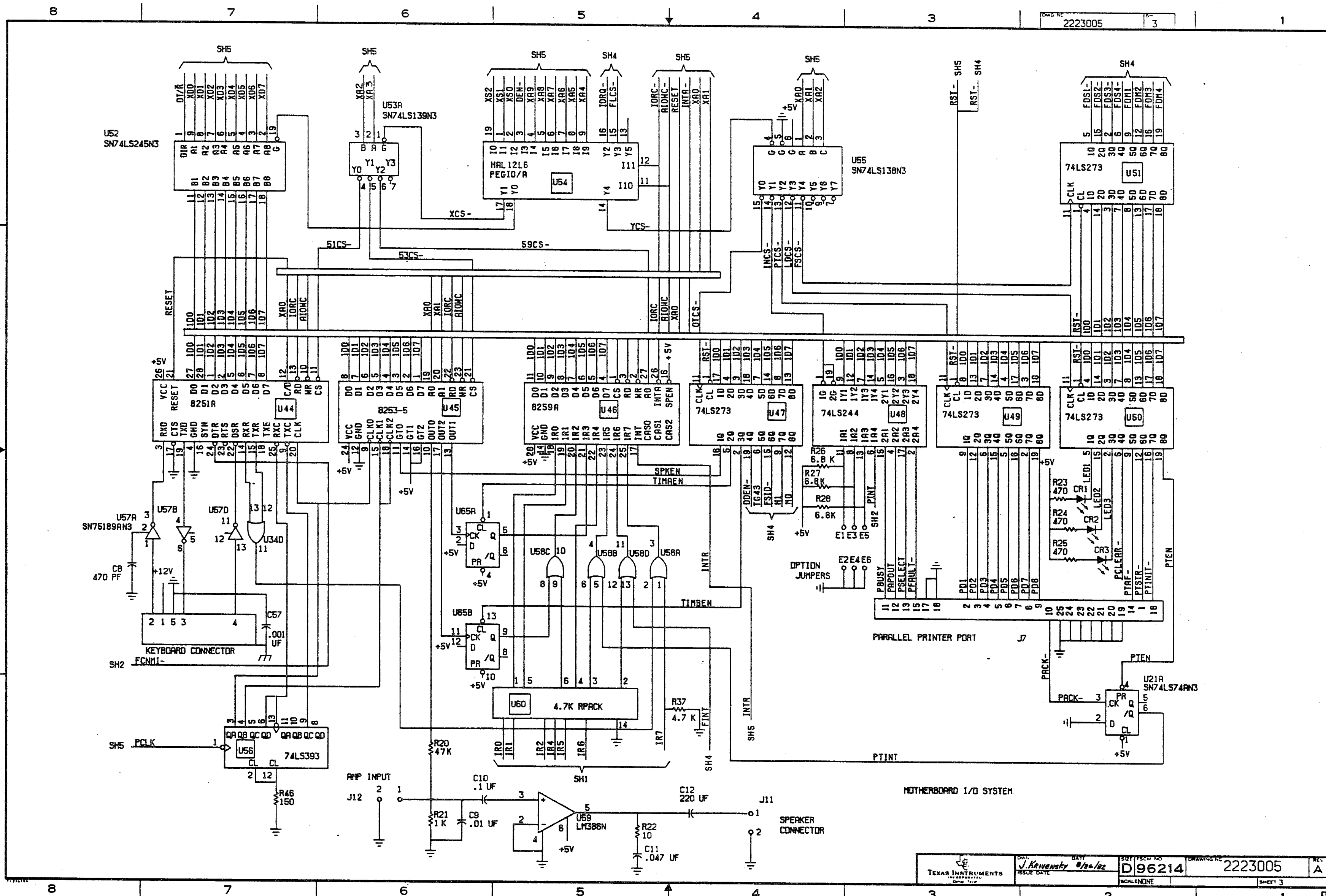
NEXT ASSY 2223003 | USED ON 8755 | APPLICATION

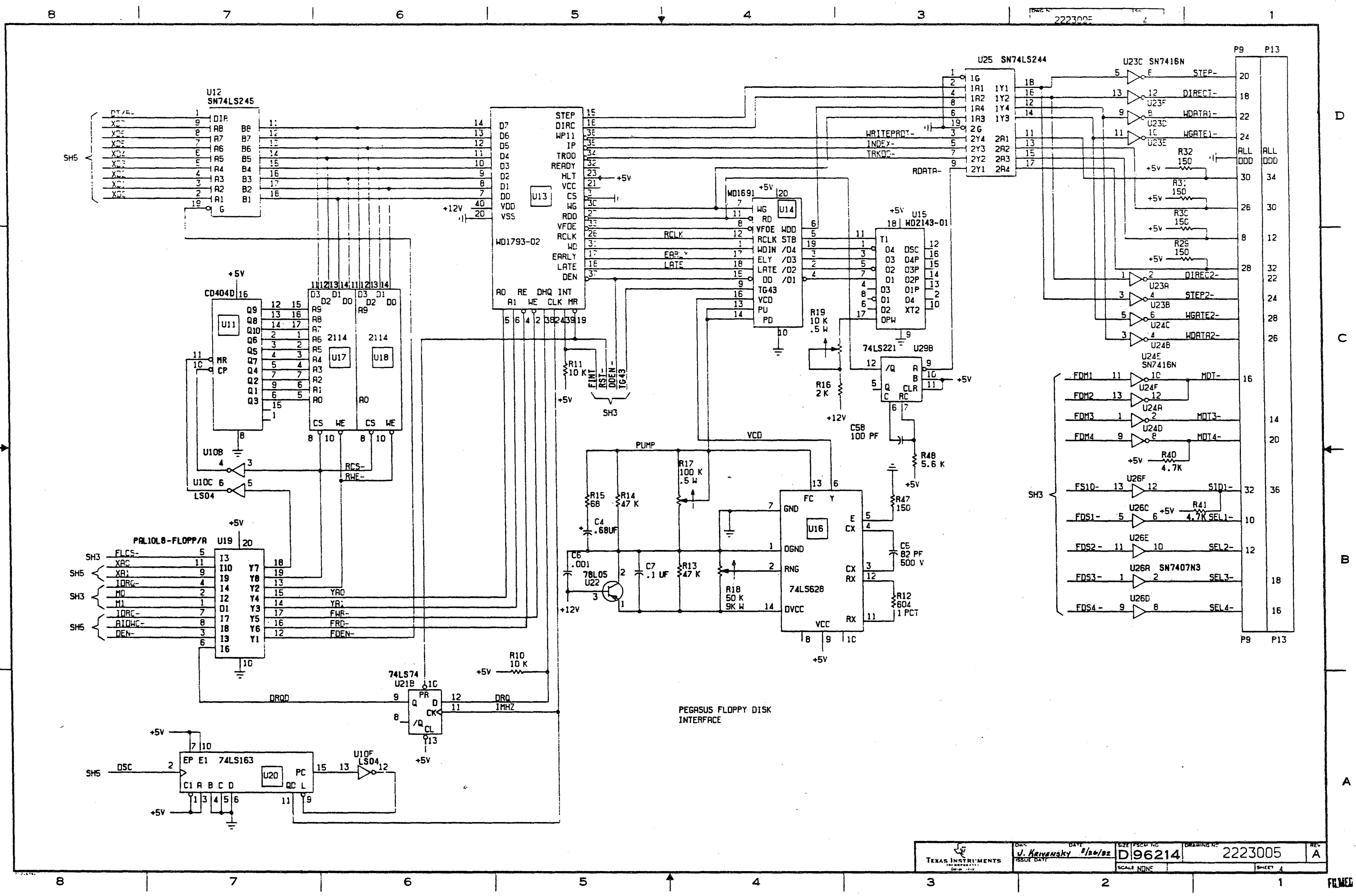
REV STATUS OF SHEETS: REV A A A A A; SH 1 2 3 4 5

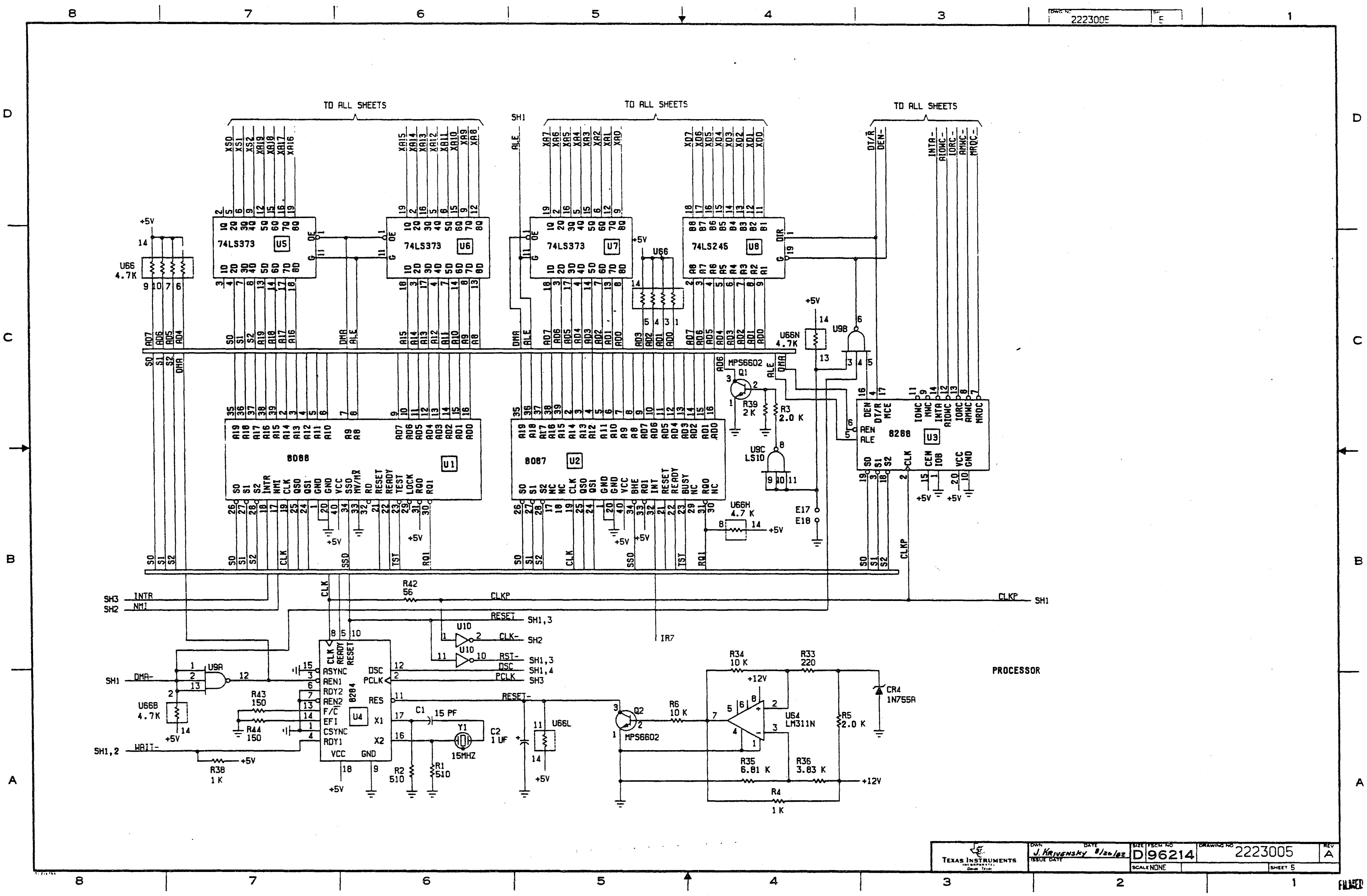
PROCESSES — FOR CORRELATION TO GOVT/IND SPECIFICATIONS, SEE TI DRAWING 72946

430


MOTHERBOARD DRAM SYSTEM
AM2964B
U27
U28
HAL16R4A
U29A
SN74LS221N3
U30
PE21199
U31
74LS280
U32A
SN74LS20N3
U32B
SN74LS20N3
U33A
SN74LS74AN3
U33B
SN74LS74AN3
U34A
SN74LS32N3
U34B
SN74LS32N3
U34C
SN74LS32N3
U67A
SN74LS00
U67B
U67C
SN74LS00
U26B
SN7407NP3
U10D
SN74LS04N
U35
4164
U36
U37
U38
U39
U40
U41
U42
U43
J10
EXPANSION RAM CONNECTOR
MEM DATA BUS SH1
SH5
SH1
SH3
PCLEAR-
AMWC-
CAS-
MA7
MA6
MA5
MA4
MA3
MA2
MA1
MA0
PE
PO
MD7
MD6
MD5
MD4
MD3
MD2
MD1
MD0
XA17
XA16
XA15
XA7
XA13
XA12
XA11
XA10
XA9
XA8
XA14
XA6
XA5
XA4
XA3
XA2
XA1
XA0
MRDC-
AMWC-
IORC-
AIOWC-
XA19
XA18
CLK-
CLKX
RFRQ
RFSH-
RFSH
WAIT-
XWAIT-
RAS1-
RMSEL-
MDEN-
RRAS-
CAS1-
RCAS-
SWM
DCAS-
PCK
NMI-
FCNMI-
NMI
PINT
R7
4.7 K
R8
4.7 K
R9
15 K
R45
1.0K
C3
3300 PF
10 PCT
+5V
TEXAS INSTRUMENTS
J. Krivensky 8/25/82
D
96214
2223005
A
SCALE NONE
SHEET 2


U52
SN74LS245N3
U53A
SN74LS139N3
HAL 12L6
PEGIO/A
U54
U55
SN74LS138N3
74LS273
U51
8251A
U44
8253-5
U45
8259A
U46
74LS273
U47
74LS244
U48
74LS273
U49
74LS273
U50
U57A
SN75189AN3
U57B
U57D
U34D
C8
470 PF
KEYBOARD CONNECTOR
C57
.001
UF
FCNMI-
U56
74LS393
PCLK
R46
150
U65A
U65B
U58C
U58B
U58D
U58A
U60
4.7K RPACK
R37
4.7 K
FINT
INTR
TIMBEN
SPKEN
TIMAEN
PTINT
R20
47K
AMP INPUT
J12
R21
1 K
C9
.01 UF
C10
.1 UF
U59
LM386N
R22
10
C11
.047 UF
C12
220 UF
J11
SPEAKER
CONNECTOR
R26
6.8 K
R27
6.8K
R28
6.8K
OPTION
JUMPERS
PARALLEL PRINTER PORT
J7
R23
470
R24
470
R25
470
CR1
CR2
CR3
LED1
LED2
LED3
U21A
SN74LS74AN3
PACK-
PTEN
MOTHERBOARD I/O SYSTEM
2223005
TEXAS INSTRUMENTS
J. Krivansky
D 96214
2223005
A
SHEET 3
FILMED

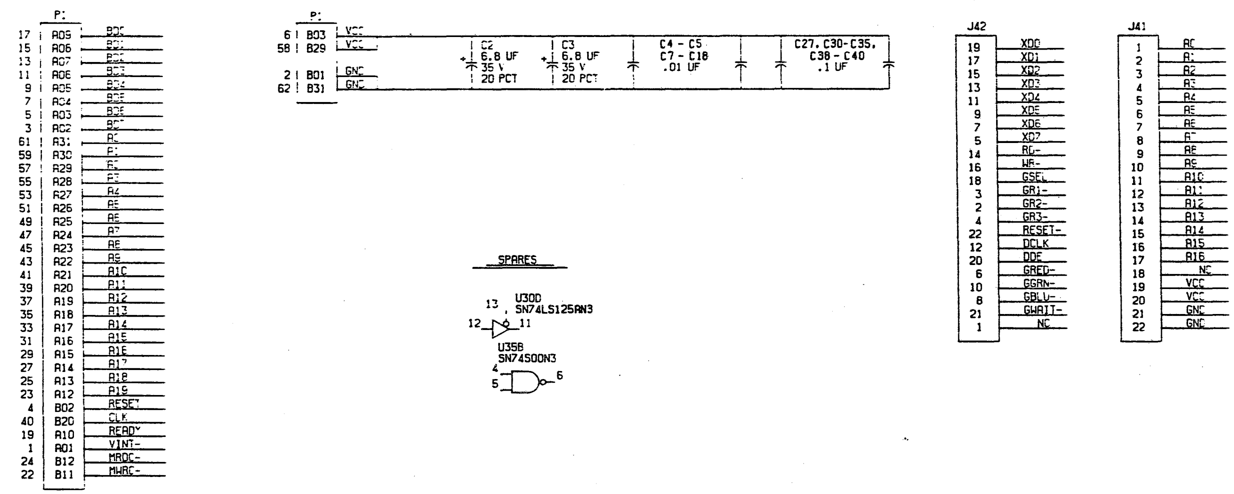

PEGASUS FLOPPY DISK INTERFACE
TEXAS INSTRUMENTS
J. Keivensky
D 96214
2223005
A
SCALE NONE
SHEET 4


TO ALL SHEETS
8088
8087
8288
8284
74LS373
74LS245
LM311N
MPS6602
PROCESSOR
TEXAS INSTRUMENTS
J. Krivensky 8/20/82
D 96214
2223005
SCALE NONE
SHEET 5
A

NOTES UNLESS OTHERWISE SPECIFIED

1. RESISTORS ARE .25 W,5 PCT
2. CAPACITORS ARE 50 V,P80M20 PCT
3. "NC" MEANS NOT CONNECTED
4. U26 NOT USED ON -0001

REV	DESCRIPTION	DATE	APPROVED
A	CN502137 (D) REDRAWN	11-2-82	



HIGHEST REFERENCE DESIGNATORS USED				
C40	J43	E2	R16	Q1
Y1	U39	P1		

REFERENCE DESIGNATORS NOT USED	
J1-39	C6,C21-C26,C28,C29,C36,C37

QTY	ITEM NO	PART OR IDENTIFYING NUMBER	NOMENCLATURE OR DESCRIPTION	PROCUREMENT SPECIFICATION	NOTES

PARTS LIST

NEXT ASSY	USED ON
2223009	8755

APPLICATION

UNLESS OTHERWISE SPECIFIED: DIMENSIONS ARE IN INCHES; TOLERANCES: ANGLES ±1°, 3 PLACE DECIMALS ±.010, 2 PLACE DECIMALS ±.03; INTERPRET DRAWING PER MIL-D-1000; REMOVE ALL BURRS AND SHARP EDGES; CONCENTRICITY MACHINED DIAMETERS .010 FIM; DIMENSIONAL LIMITS APPLY BEFORE PROCESSES; PARENTHETICAL INFO FOR REF ONLY

J. KRIVENSKY 7 MAY 82
PETE MILLER 5/25/82
GRAY MC CORD 5-26-82
GRAY MC CORD 5-26-82
RALPH CAPAN 5-26-82
SUSAN BRIDGEN 5-25-82
J.R. MOON 5/26/82

TEXAS INSTRUMENTS INCORPORATED Dallas, Texas

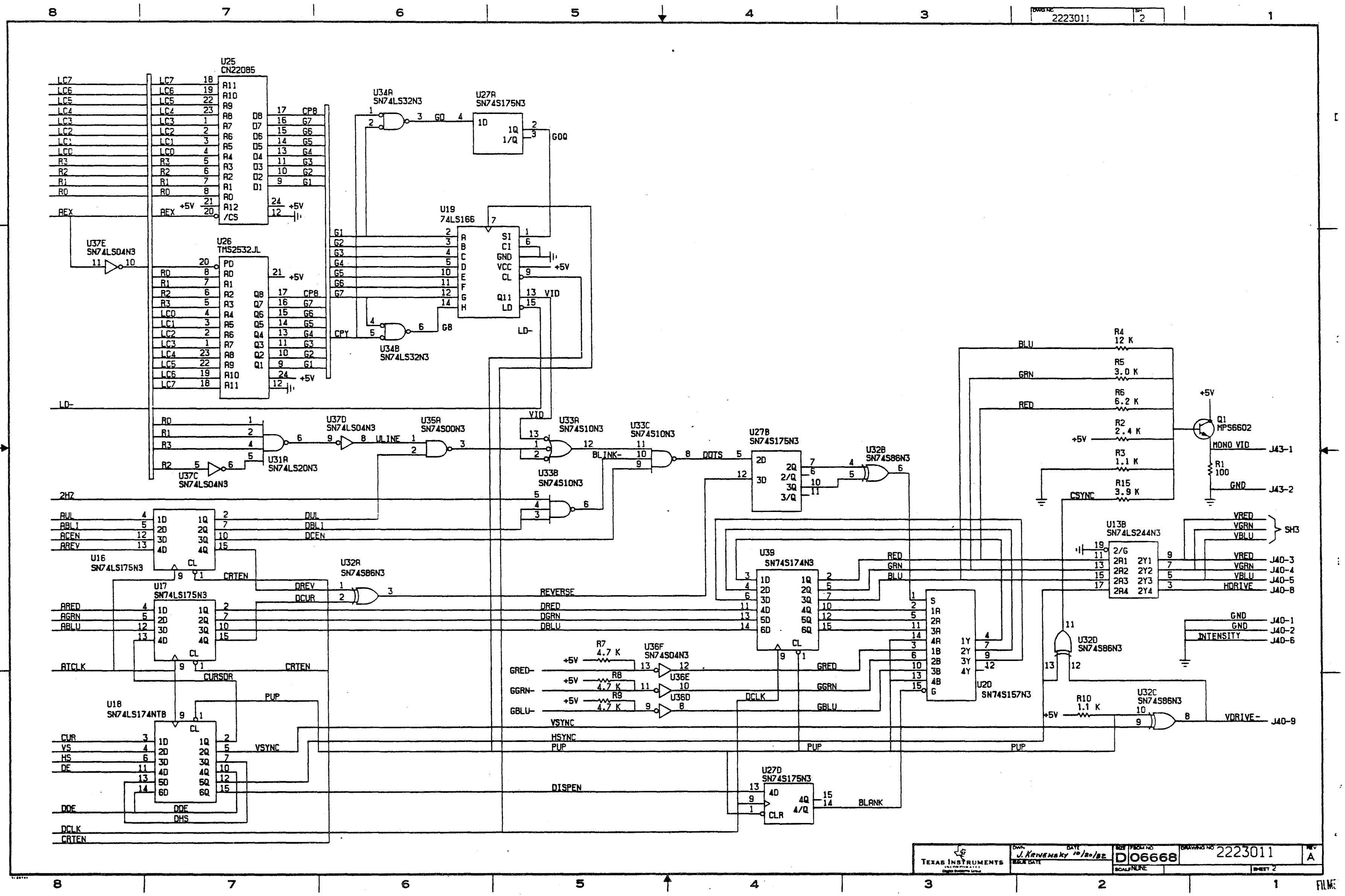
DIAGRAM LOGIC ALPHA CRT CONTROLLER

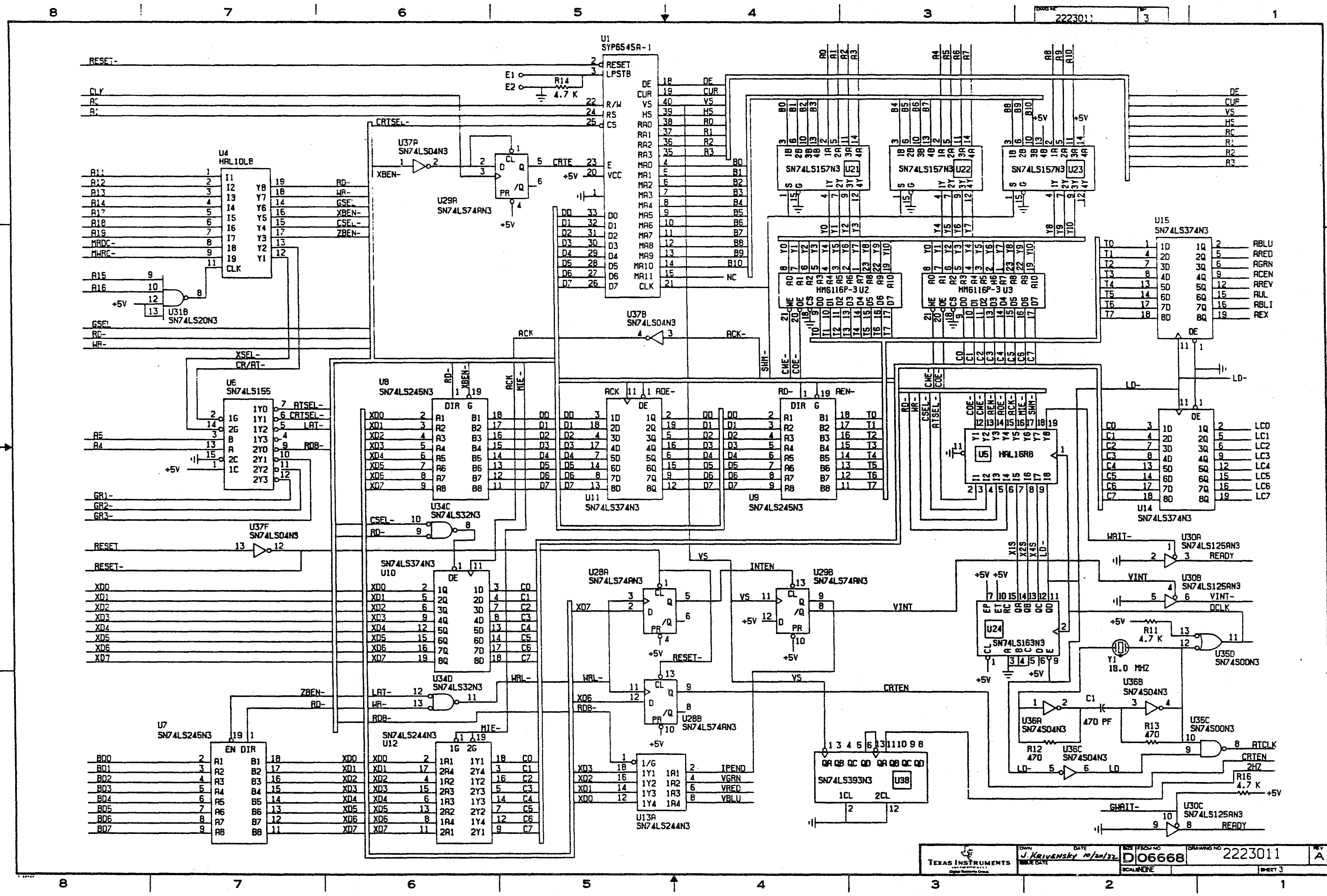
SIZE D FSCM NO 96214 DRAWING NO 2223011

SCALE NONE SHEET 1 OF 3

SEQ NO	IDENT	F-SPEC	NO	ADDITIONAL	NOTES	REV STATUS OF SHEETS	REV	A	A	A
	PROCESS			CLASSIFICATION			SH	1	2	3

PROCESSES — FOR CORRELATION TO GOVT/IND SPECIFICATIONS SEE TI DRAWING 729467

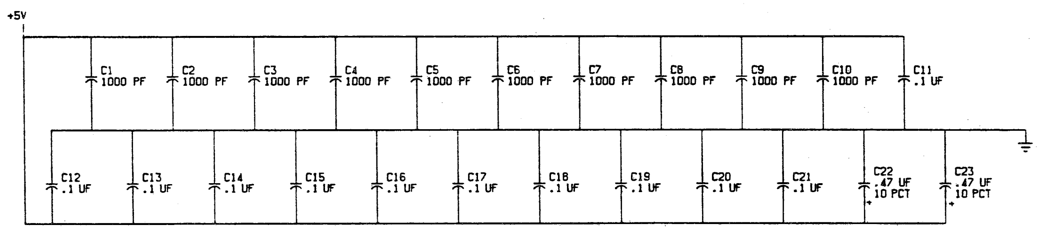

2223011
2
U25
CN22085
U26
TMS2532JL
U34A
SN74LS32N3
U27A
SN74S175N3
U19
74LS166
U34B
SN74LS32N3
U37E
SN74LS04N3
U37D
SN74LS04N3
U35A
SN74S00N3
U33A
SN74S10N3
U33C
SN74S10N3
U33B
SN74S10N3
U31A
SN74LS20N3
U37C
SN74LS04N3
U27B
SN74S175N3
U32B
SN74S86N3
U16
SN74LS175N3
U17
SN74LS175N3
U18
SN74LS174NTB
U32A
SN74S86N3
U36F
SN74S04N3
U36E
U36D
U39
SN74S174N3
U20
SN74S157N3
U27D
SN74S175N3
U13B
SN74LS244N3
U32D
SN74S86N3
U32C
SN74S86N3
Q1
MPS6602
MONO VID
J43-1
J43-2
J40-3
J40-4
J40-5
J40-8
J40-1
J40-2
J40-6
J40-9
SH3
TEXAS INSTRUMENTS
J. KRNENSKY 10/20/82
D 06668
2223011
A
NONE
SHEET 2



TEXAS INSTRUMENTS
J. Krivensky 10/20/82
D 06668
2223011
A
SHEET 3
SCALE NONE
U1 SYP6545A-1
U2 HM6116P-3
U3 HM6116P-3
U4 HAL10L8
U5 HAL16R8
U6 SN74LS155
U7 SN74LS245N3
U8 SN74LS245N3
U9 SN74LS245N3
U10 SN74LS374N3
U11 SN74LS374N3
U12 SN74LS244N3
U13A SN74LS244N3
U14 SN74LS374N3
U15 SN74LS374N3
U21 SN74LS157N3
U22 SN74LS157N3
U23 SN74LS157N3
U24 SN74LS163N3
U28A SN74LS74AN3
U28B SN74LS74AN3
U29A SN74LS74AN3
U29B SN74LS74AN3
U30A SN74LS125AN3
U30B SN74LS125AN3
U30C SN74LS125AN3
U31B SN74LS20N3
U34C SN74LS32N3
U34D SN74LS32N3
U35C SN74S00N3
U35D SN74S00N3
U36A SN74S04N3
U36B SN74S04N3
U36C SN74S04N3
U37A SN74LS04N3
U37B SN74LS04N3
U37F SN74LS04N3
U38 SN74LS393N3
Y1 18.0 MHZ
R11 4.7 K
R12 470
R13 470
R14 4.7 K
R16 4.7 K
C1 470 PF

NOTES UNLESS OTHERWISE SPECIFIED

1. CAPACITORS ARE .50 V,P80M20 PCT



-1 QTY	ITEM NO	PART OR IDENTIFYING NUMBER	NOMENCLATURE OR DESCRIPTION	PROCUREMENT SPECIFICATION	NOTES

PARTS LIST

UNLESS OTHERWISE SPECIFIED
- DIMENSIONS ARE IN INCHES
- TOLERANCES ANGLES ±1°; 3 PLACE DECIMALS ± .010; 2 PLACE DECIMALS ± .02
- INTERPRET DRAWING PER MIL-D-1000
- REMOVE ALL BURRS AND SHARP EDGES
- CONCENTRICITY MACHINED DIAMETERS .010 FIM
- DIMENSIONAL LIMITS APPLY BEFORE PROCESSES
- PARENTHETICAL INFO FOR REF ONLY

HOLE TOLERANCE

.013 THRU .125 + .004 − .001	.126 THRU .250 + .005 − .001	.251 THRU .500 + .006 − .001
.501 THRU .750 + .008 − .001	.751 THRU 1.000 + .010 − .001	1.001 THRU 2.000 + .012 − .001

NEXT ASSY	USED ON
2223015	8755

APPLICATION

DRAWN CDalekile 15 Feb 82

TEXAS INSTRUMENTS INCORPORATED Dallas Texas

DIAGRAM, LOGIC, DETAILED-RAM EXPANSION

SIZE D FSCM NO 96214 DRAWING NO 2223017

SCALE NONE SHEET 1 OF 3

SEQ NO	IDENT	F-SPEC	NO	ADDITIONAL	NOTES
	PROCESS		CLASSIFICATION		

PROCESSES — FOR CORRELATION TO GOVT/IND SPECIFICATIONS SEE TI DRAWING 22940

REV STATUS OF SHEETS	REV			
	SH	1	2	3

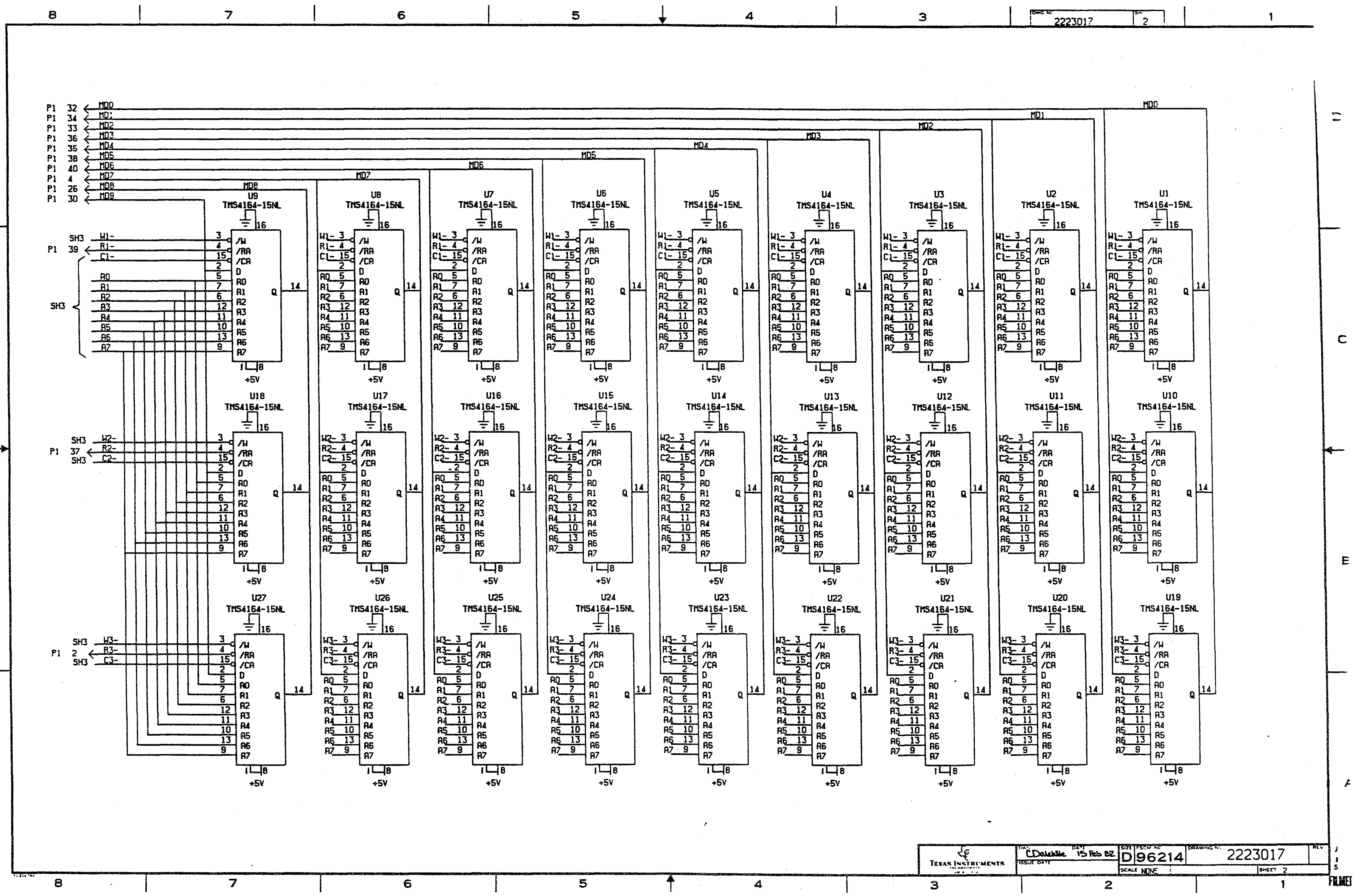
REVISIONS

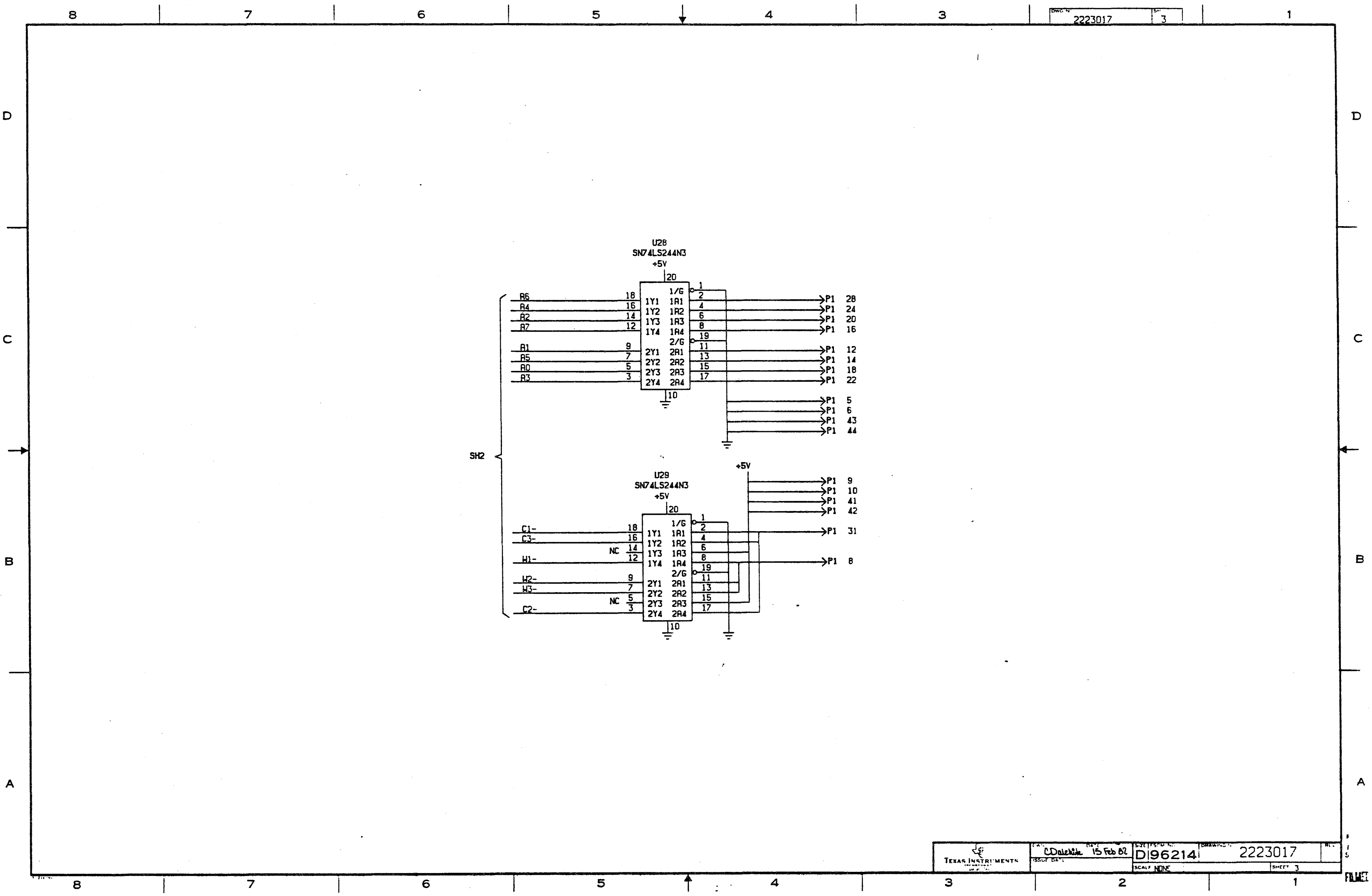
REV	DESCRIPTION	DATE	APPROVED

DWG NO 2223017

430

FILMED


2223017
SH 2
P1 32 MD0
P1 34 MD1
P1 33 MD2
P1 36 MD3
P1 35 MD4
P1 38 MD5
P1 40 MD6
P1 4 MD7
P1 26 MD8
P1 30 MD9
SH3 W1-
P1 39 R1-
C1-
SH3
A0
A1
A2
A3
A4
A5
A6
A7
SH3 W2-
P1 37 R2-
SH3 C2-
SH3 W3-
P1 2 R3-
SH3 C3-
U1 U2 U3 U4 U5 U6 U7 U8 U9
U10 U11 U12 U13 U14 U15 U16 U17 U18
U19 U20 U21 U22 U23 U24 U25 U26 U27
TMS4164-15NL
/W
/RA
/CA
D
Q
+5V
TEXAS INSTRUMENTS
15 Feb 82
D 96214
2223017
SCALE NONE
SHEET 2



U28
SN74LS244N3
U29
SN74LS244N3
+5V
SH2
P1
NC
2223017
TEXAS INSTRUMENTS
D 96214

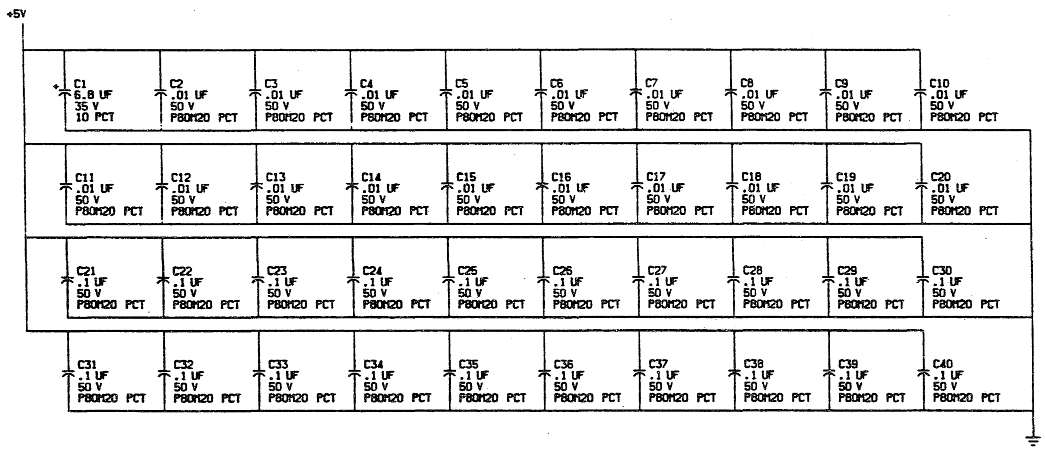
DWG NO	2223063	SH 1

REVISIONS

REV	DESCRIPTION	DATE	APPROVED
A	CN500543(D) CDalchite 21 Oct 82	11-6-82	[illegible]

NOTES: UNLESS OTHERWISE SPECIFIED

1. U5 THRU U12 USED ON -0002 ONLY



HIGHEST REFERENCE DESIGNATORS USED				
P2	U45	C40	R5	E4

REFERENCE DESIGNATORS NOT USED				

QTY	ITEM NO	PART OR IDENTIFYING NUMBER	NOMENCLATURE OR DESCRIPTION	PROCUREMENT SPECIFICATION	NOTES

PARTS LIST

NEXT ASSY	USED ON
2223061	8755

APPLICATION

UNLESS OTHERWISE SPECIFIED
- DIMENSIONS ARE IN INCHES
- TOLERANCES ANGLES ±2° 3 PLACE DECIMALS ±.010 2 PLACE DECIMALS ±.03
- INTERPRET DRAWING PER MIL D 1000
- REMOVE ALL BURRS AND SHARP EDGES
- CONCENTRICITY MACHINED DIAMETERS .010 FIM
- DIMENSIONAL LIMITS APPLY BEFORE PROCESSES
- PARENTHETICAL INFO FOR REF ONLY

HOLE TOLERANCE

CDalchite	21 Oct 82
Pete Miller	5 May 82
Gray McCord	5 May 82
Gray McCord	5 May 82
Ralph Capan	5 May 82
Susan Bridgen	5 May 82
Jim Moon	5 May 82

TEXAS INSTRUMENTS INCORPORATED Dallas, Texas

DIAGRAM, LOGIC GRAPHICS

SIZE FSCM NO	DRAWING NO
D 96214	2223063

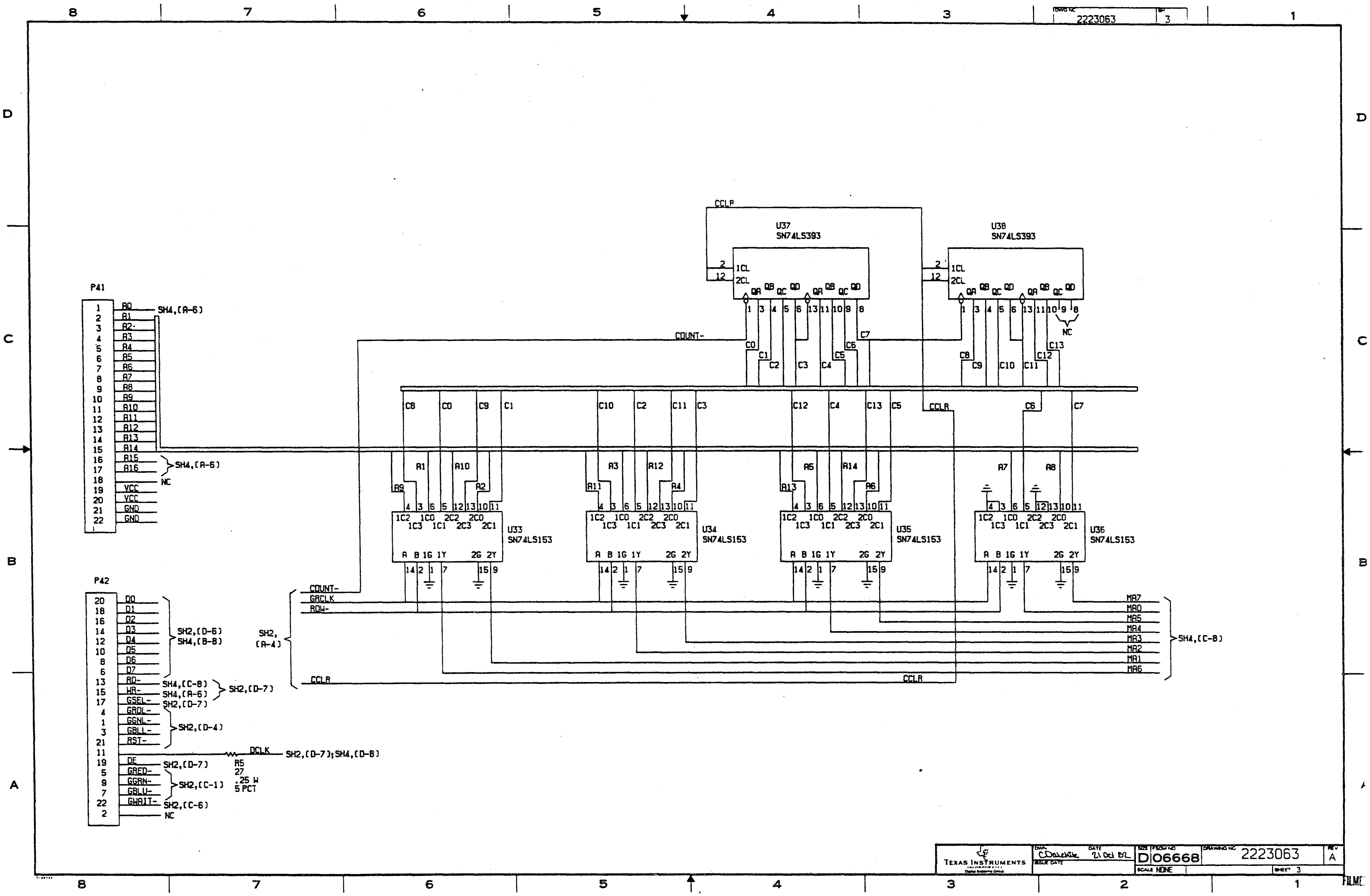
SCALE NONE SHEET 1 OF 4

SEQ NO	IDENT	F.SPEC	NO	ADDITIONAL	NOTES	REV STATUS OF SHEETS	REV	A	A	A	A
	PROCESS		CLASSIFICATION				SH	1	2	3	4

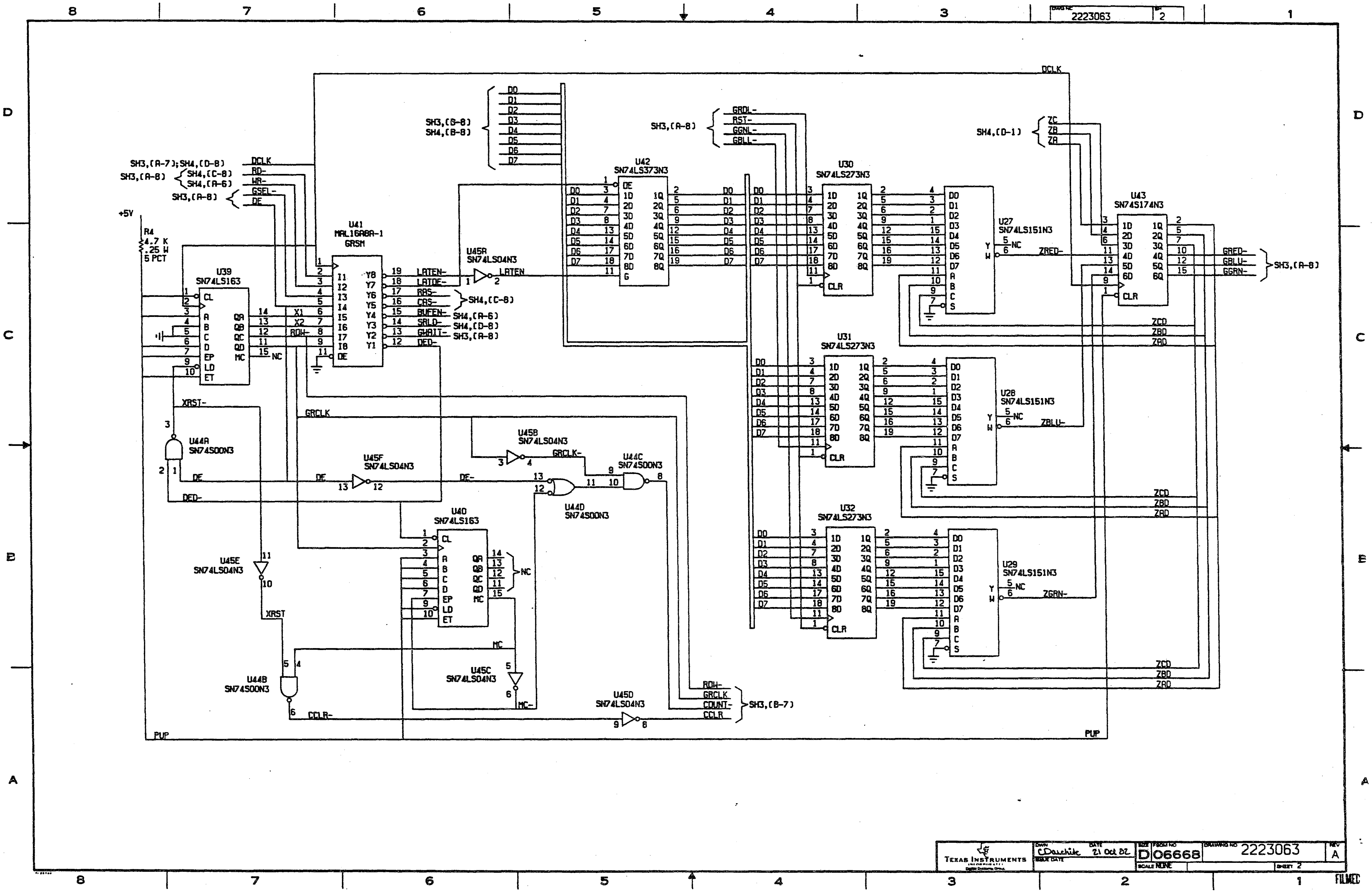
PROCESSES — FOR CORRELATION TO GOVT AND SPECIFICATIONS SEE TI DRAWING 72946?

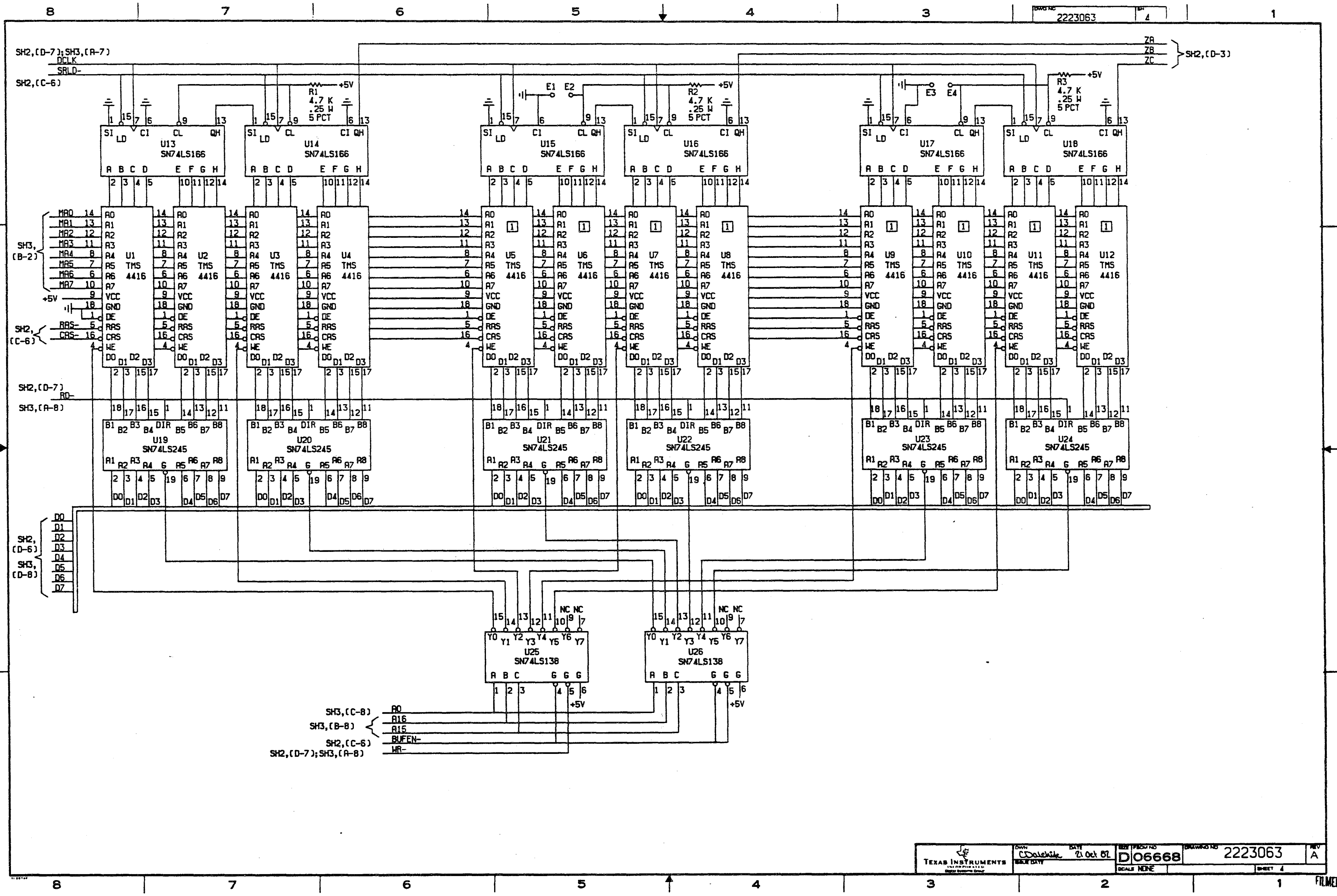
430 (H-MT)

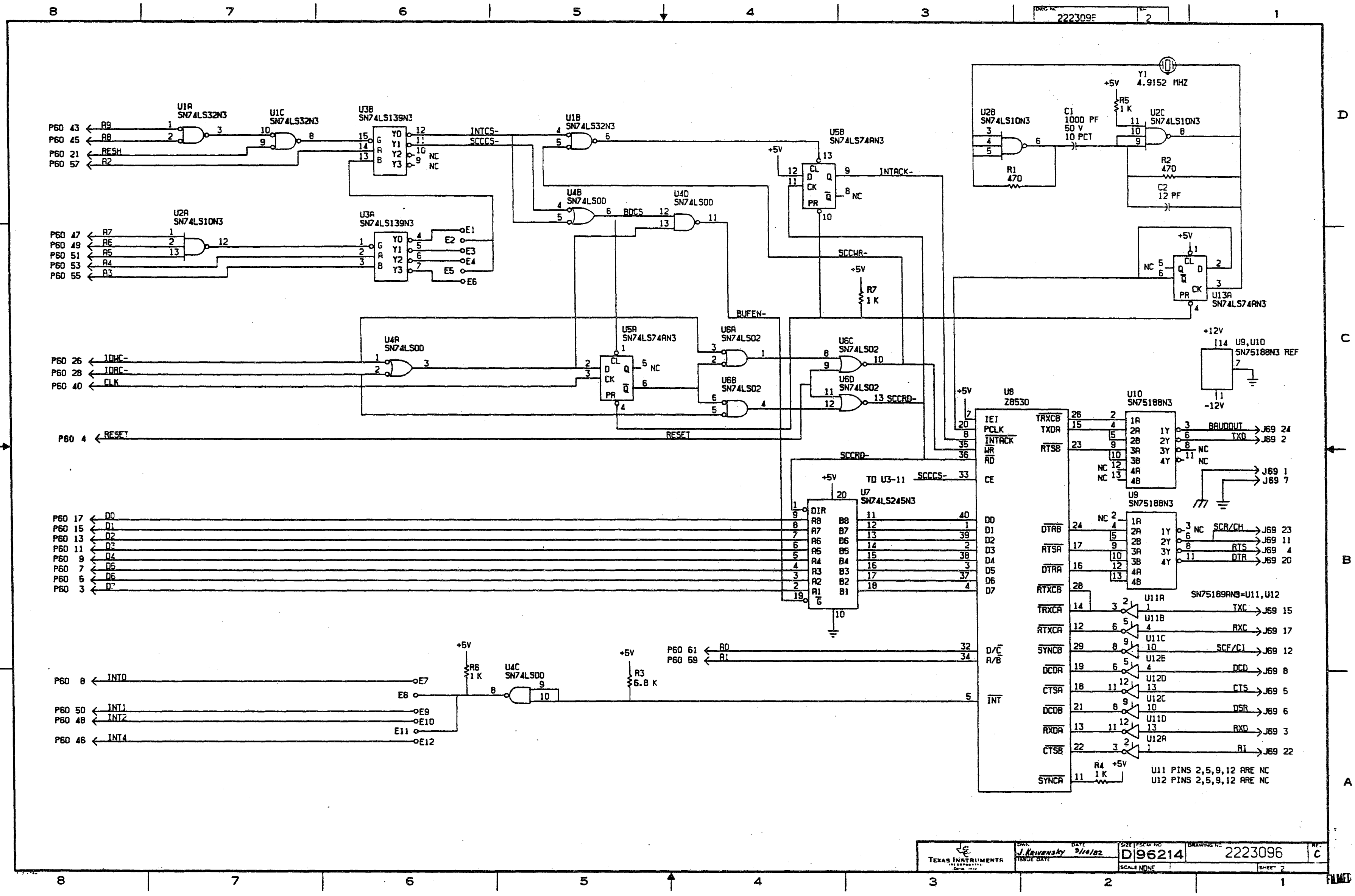
FILMED



U37
SN74LS393
U38
SN74LS393
U33
SN74LS153
U34
SN74LS153
U35
SN74LS153
U36
SN74LS153
P41
P42
CCLR
COUNT-
GRCLK
ROW-
SH4,(A-6)
SH4,(C-8)
SH2,(D-6)
SH4,(B-8)
SH2,(D-7)
SH2,(D-4)
SH2,(C-1)
SH2,(C-6)
SH2,(A-4)
DCLK
SH2,(D-7);SH4,(D-8)
R5
27
.25 W
5 PCT
NC
TEXAS INSTRUMENTS
21 Oct 82
D
06668
2223063
SCALE NONE
SHEET 3


U41
MAL16A8A-1
GRSM
U42
SN74LS373N3
U30
SN74LS273N3
U31
SN74LS273N3
U32
SN74LS273N3
U27
SN74LS151N3
U28
SN74LS151N3
U29
SN74LS151N3
U43
SN74S174N3
U39
SN74LS163
U40
SN74LS163
U44A
SN74S00N3
U44B
SN74S00N3
U44C
SN74S00N3
U44D
SN74S00N3
U45A
SN74LS04N3
U45B
SN74LS04N3
U45C
SN74LS04N3
U45D
SN74LS04N3
U45E
SN74LS04N3
U45F
SN74LS04N3
R4
4.7 K
.25 W
5 PCT
+5V
SH3,(A-7);SH4,(D-8)
SH3,(A-8)
SH4,(C-8)
SH4,(A-6)
SH3,(B-8)
SH4,(B-8)
SH4,(D-1)
SH3,(B-7)
DCLK
GRCLK
LATEN
XRST
CCLR-
PUP
TEXAS INSTRUMENTS
DWG NO 2223063
SH 2
SIZE D
FSCM NO 06668
DRAWING NO 2223063
REV A
SCALE NONE
SHEET 2
21 Oct 82


2223063
SH2,(D-7);SH3,(A-7)
DCLK
SRLD-
SH2,(C-6)
ZA
ZB
ZC
SH2,(D-3)
R1 4.7 K .25 W 5 PCT
R2 4.7 K .25 W 5 PCT
R3 4.7 K .25 W 5 PCT
+5V
E1 E2 E3 E4
U13 SN74LS166
U14 SN74LS166
U15 SN74LS166
U16 SN74LS166
U17 SN74LS166
U18 SN74LS166
SH3, (B-2)
MA0 MA1 MA2 MA3 MA4 MA5 MA6 MA7
SH2, (C-6)
RAS-
CAS-
U1 TMS 4416
U2 TMS 4416
U3 TMS 4416
U4 TMS 4416
U5 TMS 4416
U6 TMS 4416
U7 TMS 4416
U8 TMS 4416
U9 TMS 4416
U10 TMS 4416
U11 TMS 4416
U12 TMS 4416
SH2,(D-7)
RD-
SH3,(A-8)
U19 SN74LS245
U20 SN74LS245
U21 SN74LS245
U22 SN74LS245
U23 SN74LS245
U24 SN74LS245
SH2, (D-6)
SH3, (D-8)
D0 D1 D2 D3 D4 D5 D6 D7
U25 SN74LS138
U26 SN74LS138
NC NC
SH3,(C-8)
A0
SH3,(B-8)
A16
A15
SH2,(C-6)
BUFEN-
SH2,(D-7);SH3,(A-8)
WR-
TEXAS INSTRUMENTS
C Dalwhik 21 Oct 82
D 06668
2223063
A
SCALE NONE
SHEET 4
FILMED

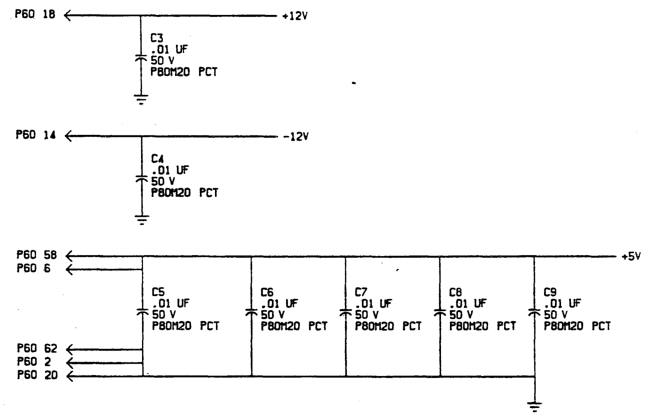

U1A SN74LS32N3
U1C SN74LS32N3
U3B SN74LS139N3
U1B SN74LS32N3
U5B SN74LS74AN3
U2B SN74LS10N3
C1 1000 PF 50 V 10 PCT
U2C SN74LS10N3
Y1 4.9152 MHZ
R5 1 K
R1 470
R2 470
C2 12 PF
U2A SN74LS10N3
U3A SN74LS139N3
U4B SN74LS00
U4D SN74LS00
U13A SN74LS74AN3
R7 1 K
U4A SN74LS00
U5A SN74LS74AN3
U6A SN74LS02
U6B SN74LS02
U6C SN74LS02
U6D SN74LS02
U9,U10 SN75188N3 REF
U8 Z8530
U10 SN75188N3
U9 SN75188N3
U7 SN74LS245N3
TO U3-11
SN75189AN3=U11,U12
U4C SN74LS00
R6 1 K
R3 6.8 K
R4 1 K
INTCS-
SCCCS-
BDCS
INTACK-
SCCWR-
BUFEN-
SCCRD-
RESET
IOWC-
IORC-
CLK
INT0
INT1
INT2
INT4
BAUDOUT
TXD
SCR/CH
RTS
DTR
TXC
RXC
SCF/CI
DCD
CTS
DSR
RXD
RI
U11 PINS 2,5,9,12 ARE NC
U12 PINS 2,5,9,12 ARE NC
TEXAS INSTRUMENTS
J. Krivansky 3/16/82
D 96214
2223096
C
SCALE NONE
SHEET 2
FILMED

DWG NO 2223096 SH 1

REV	DESCRIPTION	DATE	APPROVED
A	CN476727 (E)	6-17-82	
B	CN500549 (D)	9-16-82	
C	CN501644 (E)	11-2-82	

NOTES: UNLESS OTHERWISE SPECIFIED

1. RESISTANCE VALUES ARE IN OHMS



HIGHEST REFERENCE DESIGNATORS USED				
C9	E12	U13	J69	R7
P60	Y1			

REFERENCE DESIGNATORS NOT USED				
J1-68	P1-59			

PARTS LIST

QTY	ITEM NO	PART OR IDENTIFYING NUMBER	NOMENCLATURE OR DESCRIPTION	PROCUREMENT SPECIFICATION	NOTES

NEXT ASSY	USED ON
2223094	8755

APPLICATION

DRN J. KRIVENSKY 06 APR 82

TEXAS INSTRUMENTS INCORPORATED Dallas, Texas

DIAGRAM, LOGIC, DETAILED-COMMUNICATION, SYNC/ASYNC

SIZE D FSCM NO 96214 DRAWING NO 2223096

SCALE NONE SHEET 1 OF 2

REV STATUS OF SHEETS: REV C C; SH 1 2

PROCESSES — FOR CORRELATION TO GOVT/IND SPECIFICATIONS, SEE TI DRAWING 729467

430 H-MT FILMED

GLOSSARY

---

abort -- To end a program and return control to the operating system, usually when a mistake or malfunction occurs.

acknowledge character (ACK) -- A transmission control character sent by a receiver as an affirmative response to a sender.

address -- A number that represents a register, a memory location, or some other data source or destination.

analog -- An object (or variable) that is represented by a physical quantity, such as a continuously varying voltage. The physical quantity that represents the variable behaves as some function of the variable. (Contrast with digital).

AND -- A binary function which is "on" if and only if all of its inputs are "on".

arithmetic and logic unit -- The part of a computer that does arithmetic, logic, and similar operations.

array -- An arrangement of elements (such as numbers) usually related in some fashion.

ASCII -- (American Standard Code for Information Interchange), an eight-level (7 bits + parity) code consisting of control and graphic characters.

asynchronous transmission-- Transmission in which information characters arrive at irregular intervals of time (usually bracketed by start elements and stop elements). (Contrast with synchronous transmission).

audio frequencies -- Frequencies which can be heard by the human ear (usually between 15 cycles and 20 000 cycles per second).

auto-call -- A feature that allows a terminal to initiate a call automatically over a switched (telephone) line.

backup copy -- A copy of a file that is kept for reference in case the original file is destroyed.

BASIC (Beginner's All-Purpose Symbolic Instruction Code) -- a higher-level language, similar in structure to FORTRAN but somewhat easier to learn because of a smaller command repetoire and simpler syntax. BASIC was invented at Dartmouth College in 1963 and is probably the most popular language for personal computers.

batch processing -- a technique of data processing in which jobs are collected and grouped before processing. Data thus are normally processed in a deferred mode.

baud, baud rate -- a measure of data transfer rate, equal to the number of discrete conditions or signal events per second. (See bits per second).

binary digit (bit) -- the smallest unit of information in the binary system of notation.

bit -- the abbreviation for binary digit. In the binary notation, a bit is either of the characters 0 or 1.

bit transfer rate __ the number of bits transferred per unit time, usually expressed in bits per second (bps).

bootstrap (to "boot") -- to get a system running from a coldstart in a manner like "pulling oneself off the ground by tugging on ones bootstraps".

branch -- in programming, to make a selection from among alternative choices of instructions.

break -- a long space on an asynchronous communications line that is intended to alert the receiving CPU. Minimum duration is one character time.

buffer -- a device or area of memory which is used to hold something temporarily. For example, the screen buffer contains graphic information to be displayed on the video screen.

buffering -- (Disk Control) Storing data between transfer operations. Data read from disk is buffered before transfer to system memory and data to be written is buffered after transfer from system memory.

byte -- a binary element string of 8 bits, usually operated upon as a unit.

carrier -- a continuous frequency capable of being modulated or impressed with a signal.

CCITT -- (Comite Consultatif Internationale de Telegraphie et Telephonie), an international consultative committee which sets communications standards. The CCITT V24 interface standard is similar to the EIA RS-232-C standard.

COBOL -- (COmmon Business-Oriented Language)- a programming language designed for business data applications

code -- a system of symbols (bits) for representing data (characters).

compile -- to translate a computer program expressed in a human-oriented language into a computer-oriented language.

control character -- (1) A charcter whose occurrence in a particular context controls the handling of data. (2) In the ASCII code, any of the 32 characters in the first two columns of the standard code table.

CPS -- characters per second.

CPU (Central Processing Unit) -- unit of a computer that includes circuits controlling the interpretation and execution of instructions.

crosstalk -- the undesired transfer of energy from one circuit to another.

cursor -- a movable spot of light on the screen of a display device, usually indicating where the next character will be entered.

cyclic redundancy check (CRC) -- a method of error detection which matches CRC characters generated by transmitting and receiving devices based on the content of the message at that location.
(Disk Control) - Comparison of the checksum derived from data as it was originally written into disk storage with the checksum derived from the same data as it is being read out of storage. The first checksum is appended to the data as it is written to the disk. After reading this data, the controller computes a new checksum from it and compares the two. If the checksums match, the data is correct. A checksum error may indicate a damaged area on the disk, data that has changed since it was written, or erroneous reading of correct data where a retry may work.

cylinder -- in a disk pack, the set of all tracks with the same nominal distance from the axis about which the disk pack rotates. These tracks can be accessed without repositioning the access mechanism.

data -- a general term for any type of information.

data communications -- the movement of computer-encoded information by means of communications transmission systems.

debug -- to find and delete mistakes in computer programs or in other software.

default value -- the value chosen automatically by the computer when no explicit choice is made by the user.

delimiter -- a character that separates and organizes elements of data.

diagnostic -- pertaining to the detection of a malfunction.

digital -- the representation of numerical quantities by means of discrete integer numbers. It is possible to express in digital form all information stored, transferred or processed by a dual-state condition; e.g., ON/OFF, OPEN/CLOSED, or TRUE/FALSE. (Contrast with analog).

direct memory access (DMA) -- direct data transfer between an I/O peripheral and memory, without computer intervention. (Disk Control) - The technique generally used to transfer blocks of data between a peripheral and random-access memory. It is called direct because the host does not handle the data during the transfer operation.

directory -- a logically organized data structure which holds pointers to access data sets by sequential number or name.

display -- a visual presentation of information.

double-precision -- using two computer words instead of one to represent a number.

downtime -- the time interval during which a computer is inoperable due to a fault.

EIA (Electronic Industries Association) -- The EIA Standard RS-232-C defines interconnection interfaces for terminals.

emulate -- to imitate one system with another such that the imitating system accepts the same data and achieves the same results as the imitated system.

EOF(end-of-file mark) -- a code which signifies that the last record of a file has been read.

equalization -- compensation for the loss of signal in a line.

FCC -- Federal Communications Commission -- a board of commissioners having the power to regulate all interstate and foreign electrical communication systems originating in the United States.

field -- an area in a record (see record) treated as a unit.

FIFO -- First-In First-Out memory buffer.

file -- a group of related records handled as a unit.

firmware -- memory chips with software programs already built in.

flag -- a character that signals the occurrence of some condition, such as the end of a word.

foreground processing -- high-priority processing, usually resulting from real-time entries, given precedence by means of interrupts, over lower priority "background" processing.

formatting: (Disk Control) The division of tracks into sectors to make it easier to retrieve and update data. In each sector, the block of data is preceded by an identifying header. Gaps are inserted between sectors and between the header and data blocks within each sector to allow time for control logic functions and speed fluctuations in the disk drive assembly.

FSK(frequency-shift keying) -- a means of transmitting data in which a "1" is represented as one frequency and a "0" as another frequency.

G -- giga; when referring to computer memory it represents 1 073 741 824. Otherwise it is 1,000,000,000.

global -- in programming, it is something that is defined in one section of a program and used in at least one other section.

graphics -- symbols normally produced by handwriting, drawing, or printing. Synonymous with graphic symbol.

graphic character -- a character, other than a control character, that is normally represented by a graphic.

half duplex channel -- a communications line capable of transmitting in both directions, but not at the same time.

hardware -- physical equipment, as opposed to a computer program or method of use, e.g., mechanical, electrical, magnetic, or electronic devices.

hertz -- a unit of frequency equal to one cycle per second. Abbreviated Hz.

hexadecimal -- pertaining to a selection, choice, or condition that has sixteen possible values or states. These values or states usually contain 10 digits and 6 letters A through F. Hexadecimal digits are equivalent to a power of 16.

host computer (Also just "host") -- the primary or controlling computer to which the terminal is connected by cable for communications.

identification characters -- characters sent by a station on a switched line to identify the station.

input/output (I/O) -- something that can be in an input or output process, either simultaneously or seperately.

instruction -- in a programming language, a meaningful expression that tells the computer to execute a specific task.

instruction set -- the set of the instruction of a computer or language.

integrated circuit -- a combination of interconnected circuit elements inseperably associated on or within a continuous substrate.

integrated modem -- a modem that is an integral part of the device with which it operates.

intelligent terminal -- a synonym for a terminal that is programmable and can do some processing operations.

interface -- interconnection between two pieces of equipment having different functions.

interpreter -- a computer program that interprets programming languages. Synonymous with interpretive program.

interrupt -- the temporary stopping of some phase of computer operation caused by an event external to the operation.

job -- a task submitted for a computer to do, it usually contains all necessary instructions, files, and data to complete the task.

joystick -- a stick that is hand-held by the user and usually is used to position something on the screen.

K -- an abbreviation for the prefix kilo, i.e., 1000 in decimal notation. In storage capacity, K frequently means two to the tenth power which is 1024 in decimal notation.

Kb -- Kilobyte.

KHz -- Kilohertz. a unit of frequency equal to 1000 hertz.

LED (Light Emitting Diode) -- a small solid-state device which emits light when a current is applied.

library -- a group of related files.

light pen -- in computer graphics, a pen-like device that can sense light. When it is held up to a CRT it can be used to identify display elements.

line, communications -- describes cables, telephone lines, etc., over which data is transmitted to, and received from, the terminal. Also referred to as the "line").

list -- to print or display data.

listing -- a printout, usually of a program.

load -- to enter data into memory or into registers.

machine language -- a language that is used as is by a machine.

magnetic disk -- a flat circular plate with a magnetizable surface layer on which data can be stored by magnetic recording. The disk may be rigid or flexible.

mass storage -- storage having a very large storage capacity.

message -- in data communications, an amount of information that contains a predefined beginning and end.

modem -- (contraction of modulator/demodulator). a device which modulates and demodulates signals transmitted over communicationbs facilities. The modulator is included for transmission and the demodulator for reception. A modem is used to permit digital signals to be sent over analog lines. Also called a data set.

modulation -- the process by which some characteristic of one wave is varied in accordance with another wave or signal. This technique is used in modems to make computer signals compatible with communications facilities.

mnemonic -- symbol or symbols used instead of terminology more difficult to remember. Usually a mnemonic has two or three letters.

multiplexing -- using a transmission line to carry several different signals at one time.

NAND -- a logic operator. The NAND of any two statements P and Q is false if and only if both P and Q are true.

nanosecond -- one-thousand-millionth of a second.

noise -- undesirable disturbances in a communications system. Noise can generate errors in transmission.

non-impact printers -- a printer in which printing is not the result of mechanical impacts; e.g. thermal printers.

object code -- output from a compiler or assembler which is itself executable machine code or is suitable for processing to produce executable machine code.

offline (local) -- describes the state when equipment or devices are not connected to the communications line.

online -- describes the state when equipment or devices are connected to the communications lines under control of a processor either directly or through a communication system. The physical connection can be accomplished by either multiwire cable or a communications line.

open -- to prepare a file for processing, e.g. editing.

operating system -- software that controls the execution of computer programs and that may provide scheduling, debugging, input and output control, accounting, storage assignment, data management, and related service. Sometimes called Supervisor, Executive, Monitor, Master Control Program depending on the computer manufacturer.

parallel transmission -- method of data transfer in which all bits of a character or byte are transmitted simultaneously either over separate communications lines or on different carrier frequencies on the same communication line.

parameter -- a variable that is given a constant value for a specific purpose or process.

parity check -- addition of non-information bits to data, making the number of ones in each grouping of bits either always odd for odd parity or always even for even parity. A transmission error can then be detected by checking each group of bits received for correct parity.

password -- a word or string of characters that is recognizable by automatic means and that permits a user access to protected storage, files, or input or output devices.

program -- a series of instructions written to solve a problem. Also, to design, write, and test computer programs.

protocol -- a formal set of conventions or rules governing the format, timing, and error control to facilitate message exchange between two communicating processes.

protected field -- a field into which the operator cannot enter data.

queue -- a line formed by items in a system waiting to be processed.

RAM -- random-access memory.

read -- to get data from a storage device.

record -- a collection of fields; the information relating to one area of activity in a data processing activity, e.g., all information on one inventory item. Sometimes called item.

relational character -- a character that expresses a relationship between two operands. Common relational operators are > (greater than), < (less than), and = (equal to).

retry -- (Disk Control) Repetition of search or read/write operations to recover from "soft" (correctable) errors.

ROM -- Read-only memory.

run -- to process a task, e.g. a program, through a computer.

scratch file -- a file where temporary calculations and work is done.

scrolling -- the continuous vertical or horizontal movement of data across the screen face.

search -- (Disk Control) Reading headers on the track passing under a read/write head so as to locate the desired sector. The controller compares each identification (ID) read from the track with the ID of the desired sector.

sector -- part of a track or band on a magnetic disk.

seek -- (Disk Control) Moving a set of read/write heads so that one of them is over the desired track.

serial transmission -- a method of transmission in which each bit of information is sent sequentially on a single channel rather than simultaneously as in parallel transmission.

simplex circuit -- synonym for one-way circuit.

slave station -- a data station that is under the control of a master station.

software -- a set of computer programs, procedures, rules and associated documentation concerned with the operation of network computers, e.g., compilers, monitors, editors, utility programs. (Compare: hardware).

space -- usually equivalent to a binary zero condition.

switched network -- a communications system where the physical path of the messages may be different with each use, such as the public telephone network.

synchronous transmission -- transmission in which the data characters and bits are transmitted at a fixed rate with the transmitter and receiver synchronized.

syntax -- the format, or rules, in which instructions must be presented to the data processing equipment.

terminal -- a device or computer which may be connected to a local or remote host system, and for which the host system provides computational and data access services.

text -- a sequence of characters forming part of a transmission which is sent from the data source to the data sink, and contains the information to be conveyed.

track -- that portion of a moving data medium which is accessible to a given reading head position.

trap -- a jump to a specific location caused by a hardware condition.

turnaround time -- in communications the time required for a device to switch from receiving to sending on a two-way alternate circuit. Time is required by line propogation effects, modem timing and computer reaction.

TWX -- teletypewriter exchange service.

video -- computer data shown or displayed on a cathode ray tube monitor or display.

write -- to record data on some storage device.

INDEX

A

B

C