As you are now the owner of this document which should have come to you for free, please consider making a donation of £1 or more for the upkeep of the (Radar) website which holds this document. I give my time for free, but it costs me money to bring this document to you. You can donate here https://blunham.com/Misc/Texas

Many thanks.

Please do not upload this copyright pdf document to any other website. Breach of copyright may result in a criminal conviction.

This Acrobat document was generated by me, Colin Hinson, from a document held by me. I requested permission to publish this from Texas Instruments (twice) but received no reply. It is presented here (for free) and this pdf version of the document is my copyright in much the same way as a photograph would be. If you believe the document to be under other copyright, please contact me.

The document should have been downloaded from my website https://blunham.com/, or any mirror site named on that site. If you downloaded it from elsewhere, please let me know (particularly if you were charged for it). You can contact me via my Genuki email page: https://www.genuki.org.uk/big/eng/YKS/various?recipient=colin

You may not copy the file for onward transmission of the data nor attempt to make monetary gain by the use of these files. If you want someone else to have a copy of the file, point them at the website. (https://blunham.com/Misc/Texas). **Please do not point them at the file itself as it may move or the site may be updated.**

It should be noted that most of the pages are identifiable as having been processed by me.

I put a lot of time into producing these files which is why you are met with this page when you open the file.

If you find missing pages, pages in the wrong order, anything else wrong with the file or simply want to make a comment, please drop me a line (see above).

It is my hope that you find the file of use to you.

Colin Hinson
In the village of Blunham, Bedfordshire.

TEXAS INSTRUMENTS

TM 990

TM 990/303B Floppy-Disk Controller

MICROPROCESSOR SERIES™

User's Guide

TABLE OF CONTENTS

TABLE OF CONTENTS

TABLE OF CONTENTS

APPENDICES

LIST OF ILLUSTRATIONS

LIST OF ILLUSTRATIONS

LIST OF ILLUSTRATIONS

LIST OF TABLES

LIST OF TABLES

SECTION 1

INTRODUCTION

1.1 GENERAL

This manual covers the installation, operation, and theory of operation of the TM 990/303B floppy disk controller module, shown in Figure 1-1. Figure 1-2 is a block diagram of the module. This module provides a controlling interface between microcomputers such as the TM 990/101MA or the TM 990/102 and the disk drives listed in Table 1-1. It also features an I/O port addressable by the CPU module to provide an auxiliary interface to an RS-232-C device (not TTY).

TABLE 1-1. DISK DRIVES SUPPORTED BY THE TM 990/303B

Company and Model	TM 990/303B Supports:			
	No. of Sides	IBM Sgl Density	IBM Dbl Density	TI Dbl Density
Eight-Inch Disk Drives				
CDC 9404B	1	Y*	N	N
CDC 9406-4	2	Y*	Y*	Y*
Qume DT-8 (Qumetrack 842)	2	Y*	Y*	Y*
Shugart SA800	1	Y*	Y*	Y*
Shugart SA801	1	Y*	Y*	Y*
Shugart SA850/851	2	Y*	Y*	Y*
Siemens FDD100-8	1	Y	Y	Y
Siemens FDD200-8	2	Y	Y	Y
Tandon TM848-1	1	Y	Y	Y
Tandon TM848-2	2	Y	Y	Y
Five ($5\frac{1}{4}$) Inch Disk Drives				
BASF 6106	1	Y*	Y	N
CDC 9408	1	Y	Y	N
CDC 9409T	2	Y	Y*	N
Qume DT-5	2	Y*	Y	N
Shugart SA400	1	Y*	Y	N
Shugart SA450	2	Y*	Y*	N
Shugart SA410	1	Y	Y	N
Shugart SA460	2	Y	N	N
Siemens FDD100-5	1	Y*	Y	N
Siemens FDD200-5	2	Y	Y	N
Siemens FDD196-5	1	Y	N	N
Siemens FDD296-5	2	Y*	N	N
Tandon TM100-1	1	Y	Y	N
Tandon TM100-3	1	Y	Y	N
Tandon TM100-4	2	Y	Y*	N

Y = supported; Y* = supported for bootload also; N = not supported

The TM 990/303B can be used with microcomputer modules such as the TM 990/100MA, TM 990/101MA, or TM 990/102. However, because of buffering on the TM 990/100M and TM 990/100MA modules, the controller cannot do DMA (direct memory access) with the memory on these modules. To do DMA with a TM 990/100M or TM 990/100MA, an expansion memory module must be used.

If the TM 990/303B is to be used with the TM 990/101M module, the CPU module must be modified to the equivalent of the TM 990/101MA microcomputer module. All TM 990/101M modules returned to the factory for repair are automatically updated. The PCB board number will show if the module has been modified to the correct revision level. This board number is found on the lower left of the conductor side of the module. A board number of 994725-1 A B or 994725-1 Ⓑ (or later letter) indicates a PCB that has been modified to the proper revision level. A board number such as 994725-1 A indicates a PCB not modified to the proper revision level.

1.2 FEATURES

The TM 990/303B floppy disk controller has the following features:

- Formats supported:
  - IBM single density format,
  - IBM double density format,
  - IBM modified double density with 256 bytes/sector on all tracks,
  - TI Digital Systems Group (DSG) double density format (TILINE floppy controller).
- Disk sizes: Standard or mini (8 or 5¼ inch).
- Number of disk sides supported: One or two (see Table 1-1).
- Number of disk drives (daisy chained): Four maximum (note that some mini drives can accommodate a maximum of only three drives in a daisy chain).
- Recording methods:
  - Single density frequency modulation (FM),
  - Double density modified frequency modulation (MFM).
- Data format:
  - IBM 3740 compatible,
  - TI FS 990 compatible.
- System interfaces:
  - Controller initialization through the Communication Register Unit (CRU),
  - Data and commands through DMA transfer,
  - Interrupts.
- RS-232-C interface is addressable by the host CPU module (to provide an additional EIA interface for system).
- Three LEDs indicate controller status.
- Bootstrap load feature can be used to initialize system from diskette.
- Controller firmware is provided on two TMS 2532s (4 K words).

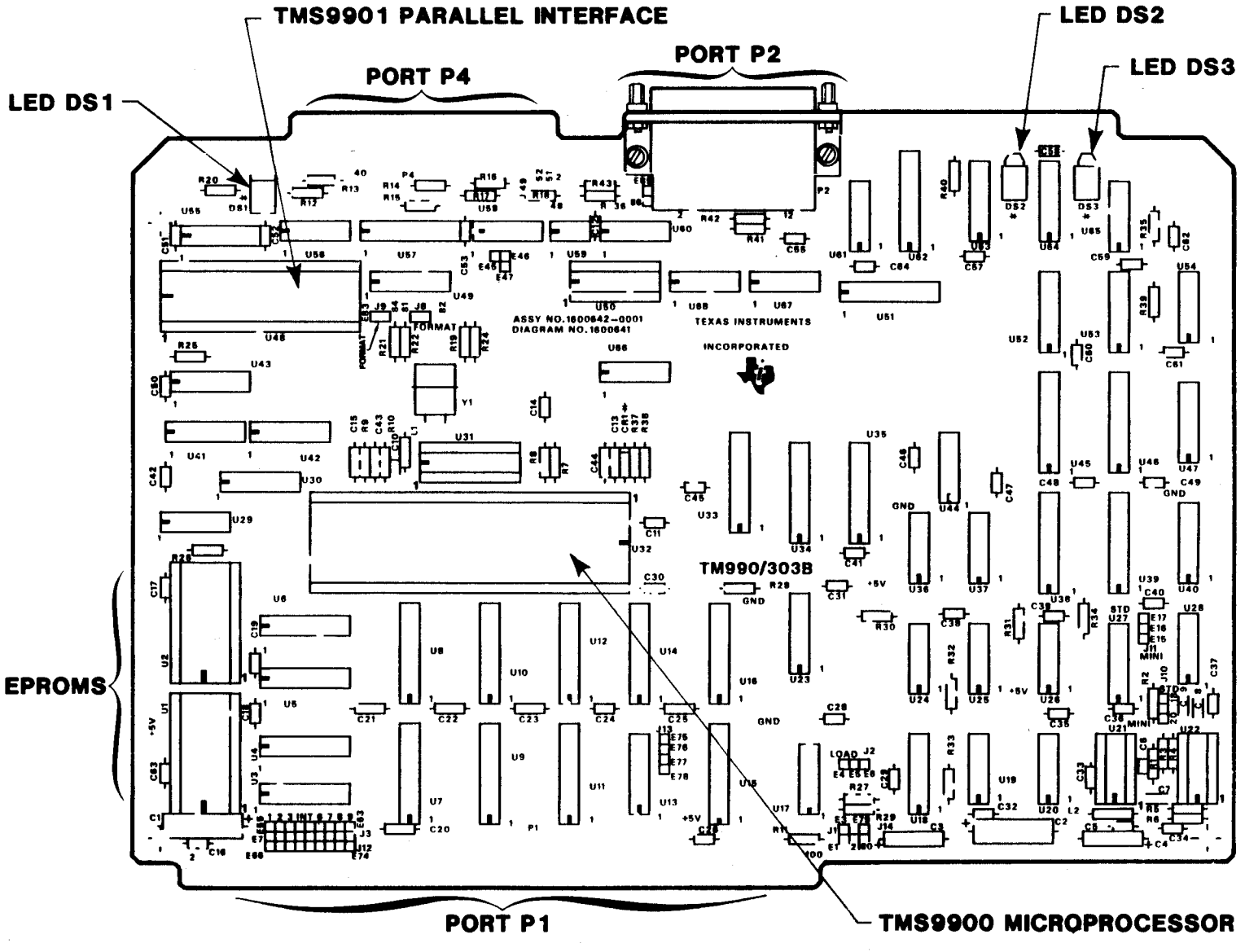


FIGURE 1-1. PRINCIPAL COMPONENTS OF THE TM 990/303B MODULE

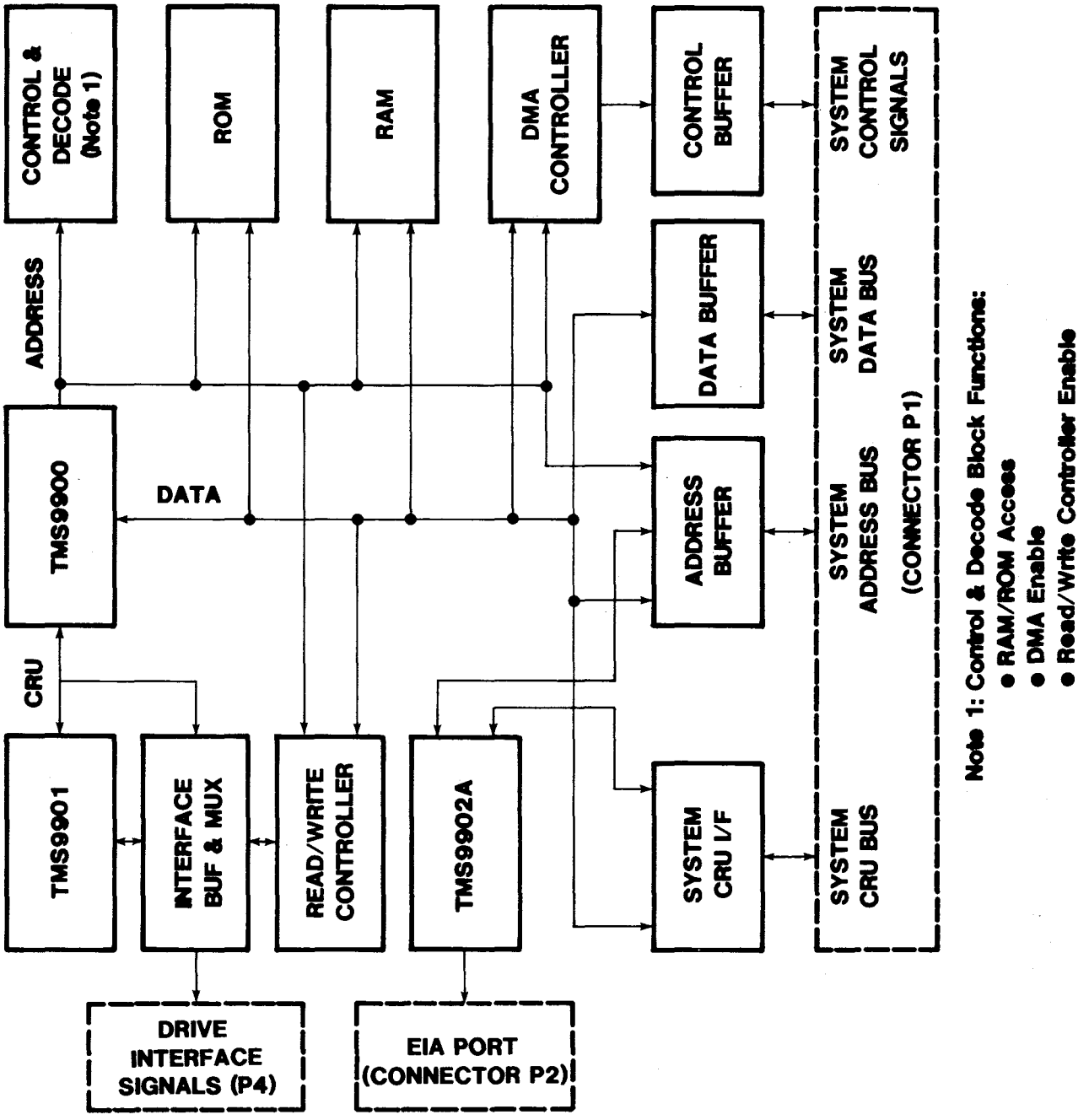


FIGURE 1-2. TM 990/303B BLOCK DIAGRAM

Software on the controller includes the following features:

- 19 commands including controller self test, read and write to/from diskette and host memory, read and write to/from controller and host RAM, bootstrap load from diskette software, format diskette, execute program in controller memory, read status of specified drive, store drive status, modify controller interface, read id, read deleted, write deleted, read unformatted, read drive interface line status,

- Command completion interrupt to host (interrupt level jumper selectable); completion status reported to host,

- Controller call through interrupt via CRU,

- Controller generated interrupt upon command completion.

1.3 MANUAL ORGANIZATION

This manual is organized as follows:

- Section 1: General information and specifications of the TM 990/303B.
- Section 2: How to install the TM 990/303B module including required peripheral equipment, disk drive requirements, system configuration, cabling, example program.
- Section 3: Bringing up the system, example software, bootload at power up, command explanation, data formats, and disk drive parameters.
- Section 4: Theory of operation including circuit descriptions, timing diagrams.
- Section 5: Programming TMS 9902A EIA port controller at port P2.
- Appendices containing auxiliary data including disk drive installation details (jumper positions, etc.), logic diagrams, parts list, cable pinouts, disk format tables, etc.

1.4 TYPICAL SYSTEM CONFIGURATION

Figure 1-3 shows a typical system configuration with a TM 990/102-3 module, a TM 990/303B. a terminal connected to the CPU module's EIA port, and a printer connected to the controller module's EIA port. The same software run on a comparable TM 990/101MA, TM 990/203, and TM 990/303A system can also be used in this system shown in Figure 1-3 if the port on the TM 990/303B has a CRU software base address of 0180_{16} (the same as the base address for the TM 990/101MA's auxilary port).

Figure 1-3 shows two standard-sized floppy disk drives connected to the floppy controller by a TM 990/527 cable; if a four-drive standard-size capability is desired, the user can provide his own cable using the data in Appendix F.

Note that the disk drives are "daisy-chained"; that is, they are connected by a single cable containing connectors for each disk unit.

In a typical system, the bootstrap load feature of the TM 990/303B would be used to load the routines to initialize the system into host RAM. An initial load routine would be used to bring in other system tasks from the diskette to host RAM. These routines could include a file manager, device service routines to drive peripherals, as well as other system software requirements. The disk controller could then cause execution of a task in host memory that would start dedicated system functions.

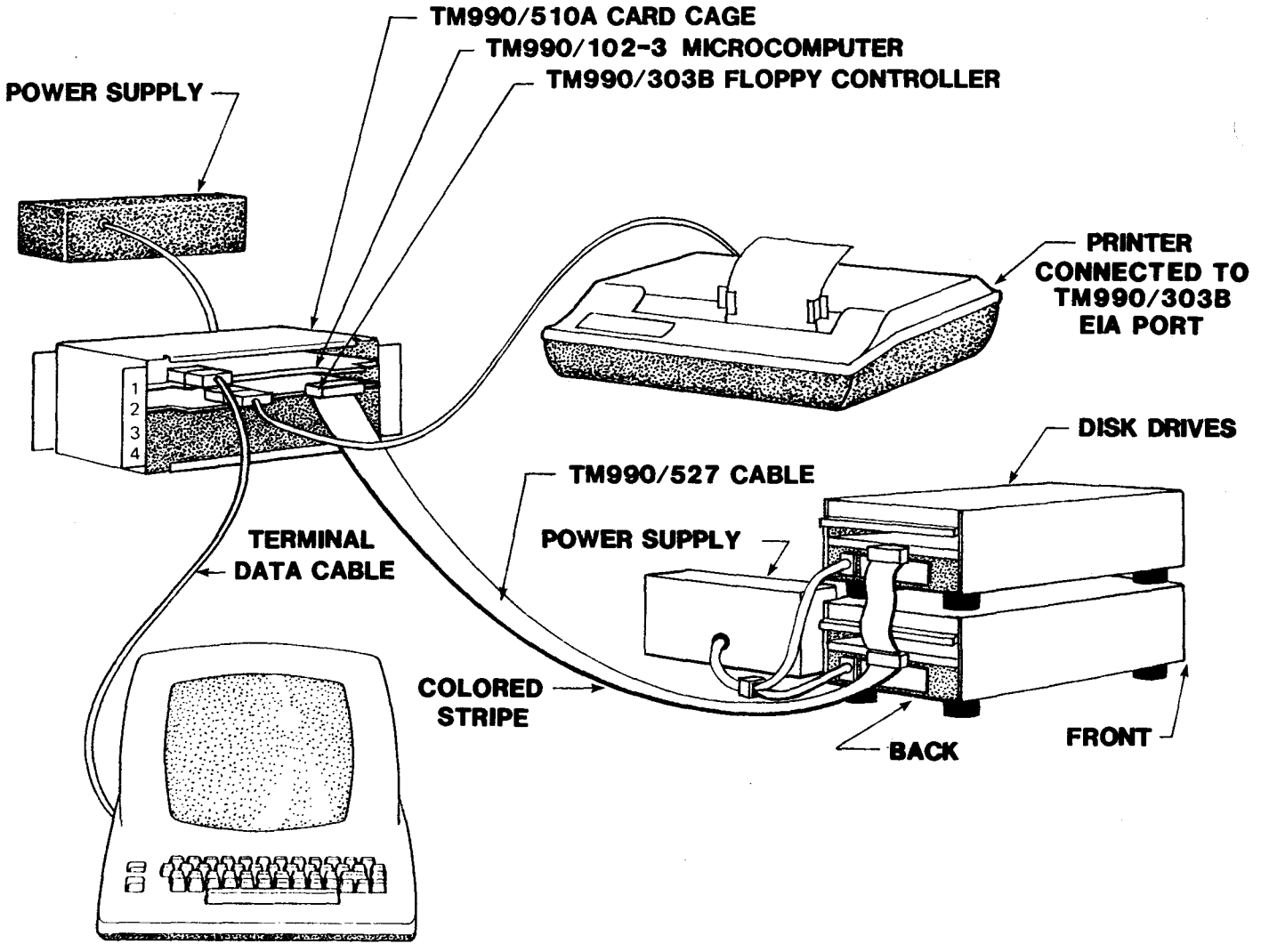


FIGURE 1-3. TYPICAL SYSTEM CONFIGURATION USING TWO MODEL 800 DISK DRIVES

CAUTION

Before applying power to the system carefully follow the installation procedures specified in Section 2 of this manual. When using a TM 990/510 or TM 990/520 card cage, the etch on the backplane between lines 95 and 96 must be open (cut) in each slot that the TM 990/303B module is installed. On TM 990/510A, /520A, and /530 chassis, remove the jumper corresponding to the slot containing the TM 990/303B module. Follow the procedures in section 2.6 and Figure 2-2. This applies to all TM 990/303B modules installed.

1.5 POWER

1.5.1 TM 990/303B Power Requirements

The following are dc power requirements for the TM 990/303B module:

	+5 Vdc		+12 Vdc		-12 Vdc		(voltage tolerance ±3%)	Unit
	Typ	Max	Typ	Max	Typ	Max		
TM 990/303B	2.1	3.0	0.1	0.2	0.04	0.2		Amps

During a powerup or a powerdown of either the disk controller or the disk drive (or both), data previously recorded on the diskette will not be destroyed due to controller action. Any operation in progress during a power sequence will not be completed.

1.5.2 Disk Drive DC Power Requirements

Individual disk drive power requirements are shown in Appendix A, section A.1.

Usually, mini disk drives function best when the peak-to-peak ripple for +5 Vdc is 50 mV or less, and the ripple for +12 Vdc is 100 mV or less.

If power is being supplied from separate power supplies, the system requires that -12V be turned on first and be turned off last. There is no required sequence in turning on the remaining voltages. This does not apply if the system uses only one power supply.

1.6 ENVIRONMENT

Ambient Temperature:

Operating: 0 to 70 degrees C (32 to 158 degrees F) at sea level
Storage: -40 to +100 degrees C (-40 to +212 degrees F)

Shock:

Shipping: 15 g applied to shipping container

Ambient Humidity:

Operating: 5 to 85 percent relative humidity without condensation
Storage: 5 to 95 percent without condensation

1.7 APPLICABLE DOCUMENTS

- TM 990/100MA Microcomputer User's Guide
- TM 990/101MA Microcomputer User's Guide
- TM 990/102 Microcomputer User's Guide
- TMS 9900 Microprocessor Data Manual
- TMS 9901 Programmable Systems Interface Data Manual
- TMS 9902A Asynchronous Communication Controller Data Manual
- TM 990/201/206 Expansion Memory Modules
- TM 990/202 EPROM/RAM Module
- TM 990/203 Dynamic RAM Memory Expansion Module
- TM 990/204 Memory Module with Battery Backup
- TM 990/425B Demonstration Software for TM 990/303B
- TM 990/527 Cable for Standard Disk Drives User's Guide
- TM 990/535B Cable for Mini Disk Drives User's Guide

SECTION 2

INSTALLATION AND OPERATION

2.1 GENERAL

This section covers the installation of the TM 990/303B Floppy Disk Controller and has some example software to illustrate controller operation.

CAUTION

1. Before applying power to your TM 990/303B module, properly set jumper plugs and connectors as described in the following:

   - jumpers at TM 990/303B module (section 2.4, Figure 2-1, Table 2-1),
   - jumpers on the disk drive (Appendix A or in drive user's guide),
   - cable attachment to standard drive (section 2.7, Figures 2-4, 2-5), or cable attachment to mini drive (section 2.7, Figures 2-6, 2-7),
   - cut backplane etch in chassis slot where controller module is installed (see section 2.6) or remove the jumpers between line 95 and 96 on the backplane of chassis.

2. When using a TM 990/510 or /520 card cage, the etch on the backplane between lines 95 and 96 must be open (cut) in each slot that the TM 990/303B module is installed. Follow the procedures in section 2.6 and Figure 2-2. This applies to all TM 990/303B modules installed. For TM 990/510A, TM 990/520A, and TM 990/530 card cages, remove the jumpers provided for this purpose.

3. Verify the system configuration using the check list in section 2.8 before applying power. Improper switch or jumper settings may harm equipment.

2.2 UNPACKING

Find the following:

- TM 990/303B Floppy Disk Controller Module
- TM 990/303B Floppy Disk Controller User's Guide
- Warranty Card
- Factory Repair and Exchange Policy

Remove the TM 990/303B module from its protective packing. Report any discrepancies to your supplier.

2.3 REQUIRED EQUIPMENT

This floppy disk controller module is recommended for use only with the disk drives listed in Table 1-1.

A power supply must be provided to meet the total power requirements of the system, including, for example:

- disk drive(s),
- TM 990/303B module as specified in section 1.5,
- microcomputer module,
- if installed, an expansion memory module (necessary for TM 990/100M or TM 990/100MA system) or other auxiliary module.

A TM 990/510A (four slots), TM 990/520A (eight slots), TM 990/530 (16 slots) or equivalent card cage should be used to provide signal and power busing to the microcomputer, disk controller, and (if used) expansion memory module. The microcomputer should be placed in the top of the cage with other modules below it.

A terminal (and proper terminal cabling), such as the Texas Instruments 743 is required for user interaction.

A connecting cable such as the TM 990/527 (two standard disk drives) or TM 990/535B (three mini drives) is needed between disk controller port P4 and the floppy disk drive(s). Cables to accomodate four drives (either 8 or 5¼ inch drives) are easily made and are usually available from a distributor. Appendix F lists disk drive pinouts.

An expansion RAM memory module can provide additional storage and workspace for the host system. Suggested expansion memory modules include the TM 990/202 EPROM/RAM module, the TM 990/203 dynamic memory module or the TM 990/206 static RAM module. Because memory on the TM 990/100M and the TM 990/100MA modules is buffered, they cannot be used for DMA.

2.4 JUMPERS ON TM 990/303B MODULE

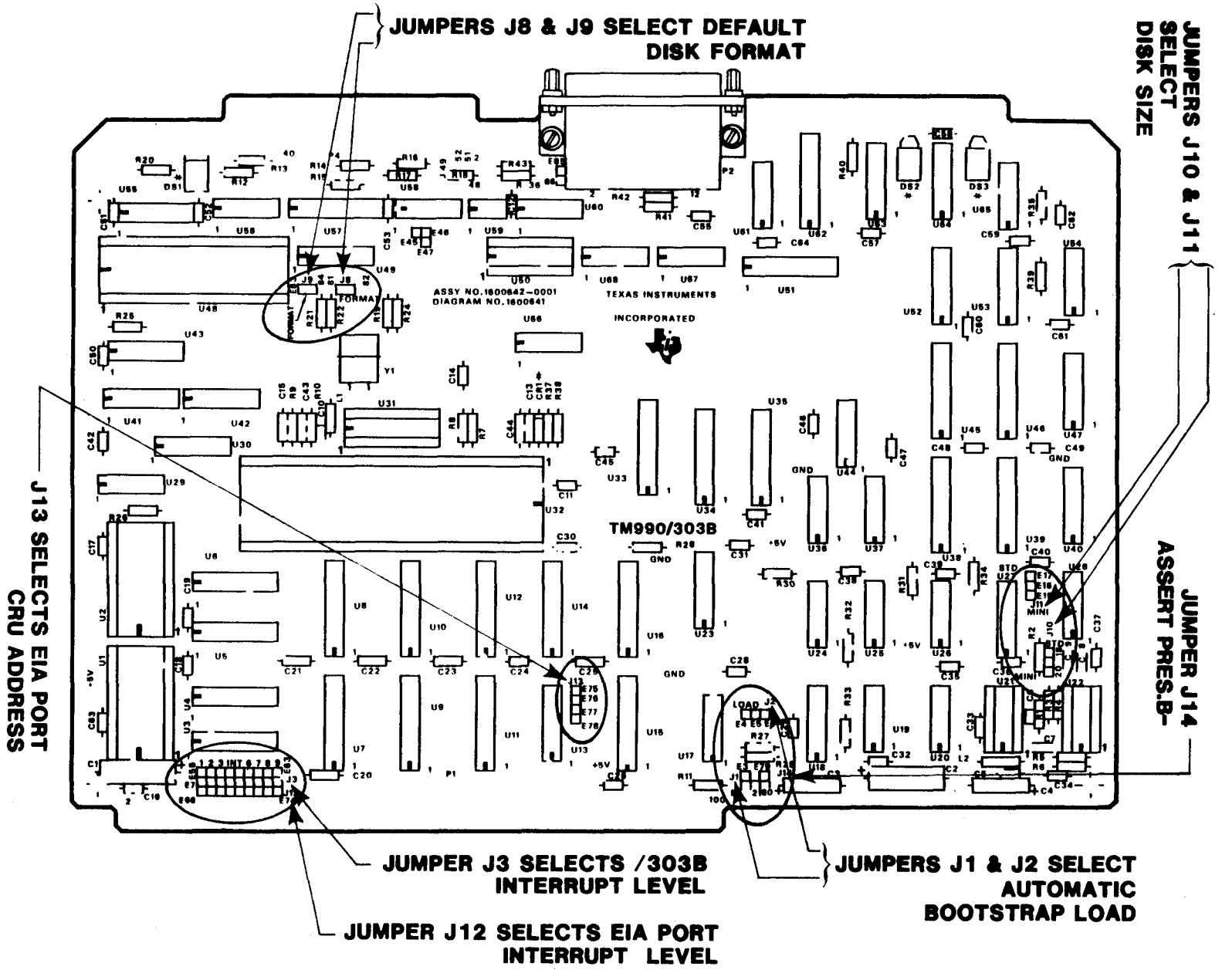
Module jumper locations are shown in Figure 2-1; module jumper settings are listed in Table 2-1. Note that jumpers J4, J5, J6, and J7 are not used.

NOTE

The Define Drive command (command 10_{16} in Table 3-2) must specify the same disk size (standard or mini) as specified by jumpers J10 and J11. Failure to match the software designation (via the Define Drive command) and the hardware settings at J10 and J11 will result in incorrect performance. This command does not have to conform to the settings of jumpers J8 and J9 which set the default value only at a powerup or bootload; this default can be changed by the Define Drive command.

2.5 JUMPERS ON DISK DRIVES

Follow the manufacturer's instructions for setting jumpers on the respective disk drive printed circuit boards. Suggested disk drive jumper settings are listed in Appendix A.



NOTE: There are no jumpers J4, J5, J6, and J7.

FIGURE 2-1. TM 990/303B JUMPER LOCATIONS

TABLE 2-1. TM 990/303B JUMPER SETTINGS (page 1 of 2)

Function	Jumper Setting * = As Shipped	Comments
1. Bootstrap Load Selection		
- Automatic Bootstrap Load	J1 E1-E2 J2 E4-E5	
- No Automatic Bootstrap Load	*J1 E1-E3 *J2 E5-E6	
2. Controller Interrupt Level to Host CPU		
- Interrupt Level 1	J3 E55-E7	
- Interrupt Level 2	*J3 E56-E8	
- Interrupt Level 3	J3 E57-E9	
- Interrupt Level 4	J3 E58-E10	
- Interrupt Level 5	J3 E59-E11	
- Interrupt Level 6	J3 E60-E53	
- Interrupt Level 7	J3 E61-E54	
- Interrupt Level 8	J3 E62-E64	
- Interrupt Level 9	J3 E63-E65	
3. EIA Port Interrupt Level to Host CPU		TMS 9902A can be programmed to issue interrupts for four different conditions. For no interrupts, remove the jumper.
- Interrupt Level 1	J12 E7-E66	
- Interrupt Level 2	J12 E8-E67	
- Interrupt Level 3	J12 E9-E68	
- Interrupt Level 4	J12 E10-E69	
- Interrupt Level 5	J12 E11-E70	
- Interrupt Level 6	*J12 E53-E71	See CAUTION below.
- Interrupt Level 7	J12 E54-E72	
- Interrupt Level 8	J12 E64-E73	
- Interrupt Level 9	J12 E65-E74	
4. Reset Host CPU at Powerup Bootload		J14 (E79-E80) must be jumpered when used with CPUs such as the TM 990/100 or /101 (but unjumpered when used with /102 which which asserts PRES- at powerup).
- Assert PRES- to host CPU at powerup bootload (J1 jumpered for automatic bootload also)	*J14 In	
- Do not assert PRES- to host CPU at powerup bootload	J14 Out	
5. EIA Port: CRU Software Base Address (R12 Contents) & Port Deselect		
- 0180_{16} Base Address	*J13 E78-E77	
- 0300_{16} Base Address	J13 E77-E76	
- Deselect EIA Port	J13 E76-E75	Port inoperable; See CAUTION.

CAUTION

If J13 is set to deselect the EIA port (P2) with E76-E75 jumpered, DO NOT select an EIA interrupt: leave J12 unjumpered. Follow this precaution to prevent unwanted interrupts if the TMS 9902A is not in the reset state.

TABLE 2-1. TM 990/303B JUMPER SETTINGS (page 2 of 2)

Function	Jumper Setting * = As Shipped	Comments	TWOSIDED-
6. Select Default Diskette Format for: Note: S = Single, D = Double			
a. Powerup Bootload (J1/J2 set for automatic bootload)			
IBM S Density, S Sided	*J8 In, J9 In	See Note 1	Not Active
IBM S Density, D Sided	J8 In, J9 In	See Note 1	Active
IBM D Density, S Sided	J8 In, J9 Out	See Note 1	Not Active
IBM D Density, D Sided	J8 In, J9 Out	See Note 1	Active
IBM Modified D Density, S Sided	J8 In, J9 Out	See Note 1	Not Active
IBM Modified D Density, D Sided	J8 In, J9 Out	See Note 1	Active
TI D Density, S Sided	J8 Out, J9 In	See Note 1	Not Active
TI D Density, D Sided	J8 Out, J9 In	See Note 1	Active
Mini S Density, S Sided	J8 Out, J9 Out	See Note 1	---
Mini D Density, D Sided	J8 Out, J9 Out	See Note 1	---
b. Powerup Reset only (J1/J2 not set for automatic bootload)			
IBM S Density, S Sided	*J8 In, J9 In	See Note 1	Not Active
IBM S Density, D Sided	J8 In, J9 In	See Note 1	Active
IBM D Density, S Sided	J8 In, J9 Out	See Note 1	Not Active
IBM D Density, D Sided	J8 In, J9 Out	See Note 1	Active
TI D Density, S Sided	J8 Out, J9 In	See Note 1	Not Active
TI D Density, D Sided	J8 Out, J9 In	See Note 1	Active
Mini S Density, S Sided	J8 Out, J9 Out	See Note 1	---
Note 1: J8 = E81-E82, J9 = E84-E83. A powerup bootload or powerup reset requires the correct sided diskette in the drive. The correct diskette will result in pin P4-10 being a "0" for two-sided diskettes and a "1" for one-sided diskettes correctly placed in the drive. This signal is inverted at U49 and is valid only when the drive is selected for access. P4-10 is not used for mini drives because mini drives do not have this interface line.			
7. Select Disk Size			
- Standard Size	*J10 E41-E42 *J11 E44-E45		
- Mini Size	J10 E40-E41 J11 E43-E49		

NOTE

Jumpers J4, J5, J6, and J7 are not used.

2.6 MODULE INSTALLATION

Turn power off before installation of modules into the card cage. Install the microcomputer module at the top of the cage with other modules beneath, with any blank slots being at the bottom of the card cage. Where the TM 990/303B modules are installed in a TM 990/510 or /520 card cage, cut the connection on the card cage backplane between lines 95 and 96 as shown in Figure 2-2. This break is made at the slot containing the TM 990/303B module:

- on TM 990/510A, /520A, and 530 card cages, remove the jumper at the slot, or
- on TM 990/510 and 520 card cages, cut the etch as shown in Figure 2-2.

This is the same as for multicontroller systems as explained in section 2.10. Properly install the modules and connect cables as specified in section 2.7. All modules should be close together in the card cage with empty slots at the bottom. Next, conduct a system check and apply power as described in section 2.8.

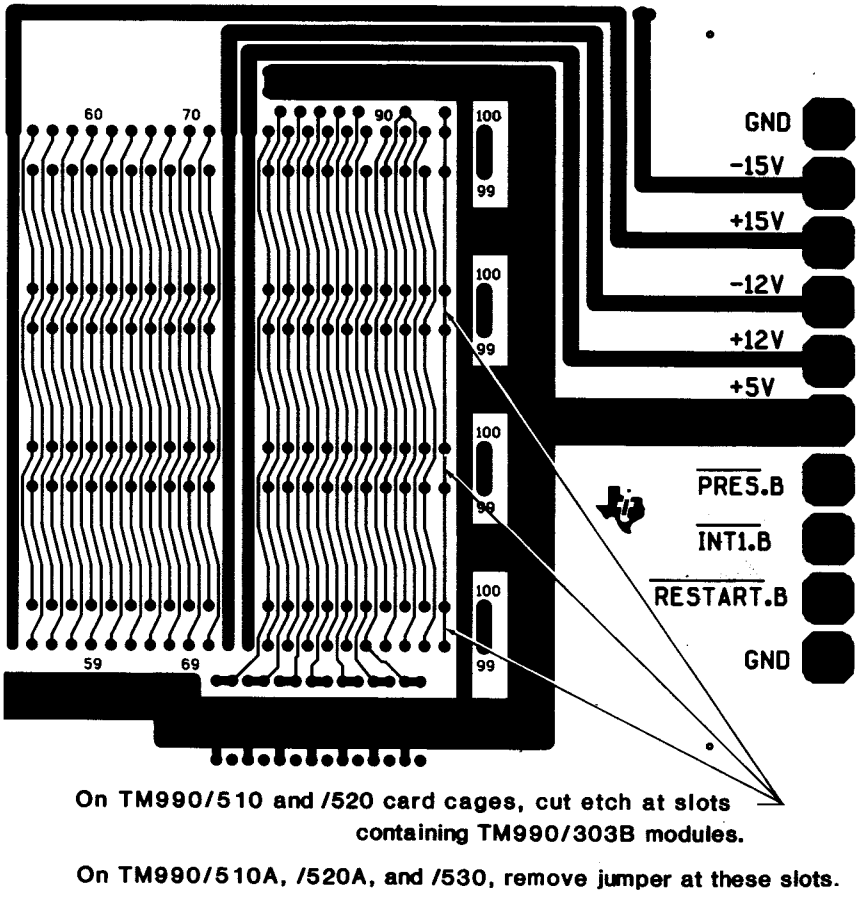


FIGURE 2-2. LOCATION OF SOLDER BRIDGE BETWEEN PINS 96 AND 95 OF MOTHERBOARD

2.7 CABLING

Figure 2-3 shows a typical system configuration using a TM 990/527 cable which connects one or two eight-inch drives to the controller module. Detail connections using this cable are shown in Figures 2-4 and 2-5. To connect an eight-inch disk drive system:

- Connect the EIA data cable from the terminal to connector P2 of either the microcomputer module or the controller module.

- The TM 990/527 cable has three connectors on it. The end with the two connectors closest together attaches to the two disk drives. The single connector at the other end goes to the controller module, and is attached with the colored stripe to the left of connector P4 as shown in Figure 2-4. As seen in the figure, pin 2 and the colored stripe are at the left of the connector when properly installed. The colored stripe encloses the wire to pin 1.

- Connect one or both of the disk drive cable connectors to the back of the disk drive(s). See the manufacturer's installation instructions for proper orientation of pin 1 on the disk drive connector. Pin 1 on the TM 990/527 cable is denoted by a diamond engraved in the connector on the side near the colored stripe. See Figure 2-5 for orientation. If pin 1 is on top of the edge connector at the disk drive, the diamond on the connector must also be on top, oriented with that pin.

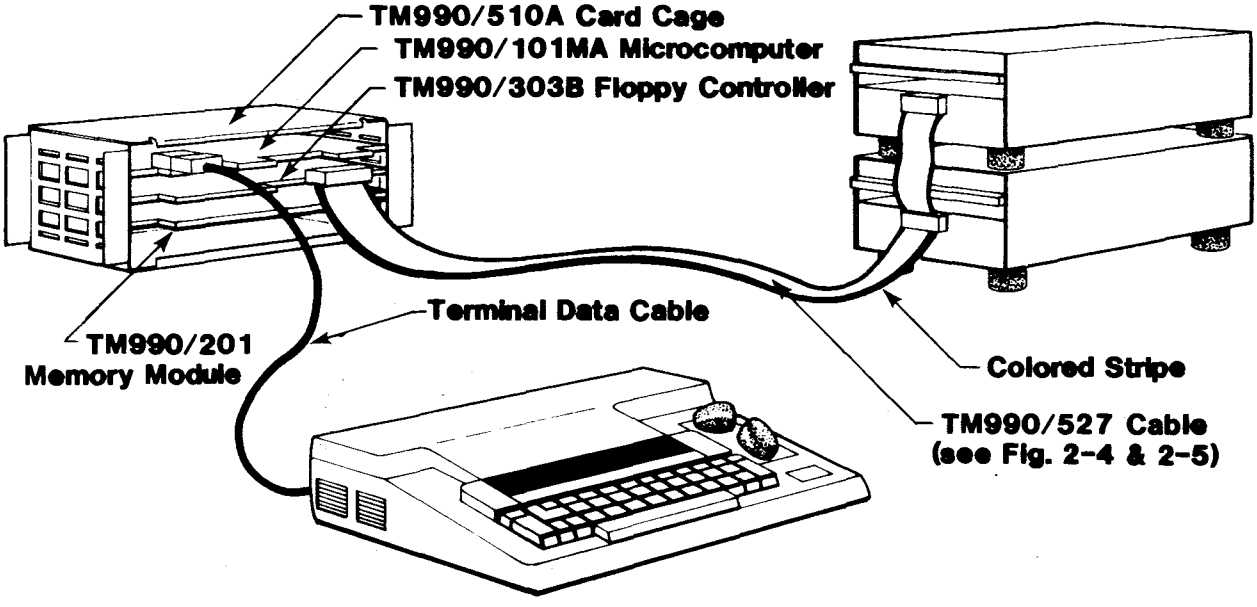


FIGURE 2-3. SYSTEM INTERCONNECTIONS USING TM 990/527 CABLE (STANDARD SIZE)

NOTE

If you want to make your own cable, be aware that the connector plugs of various vendors, including TI, do not necessarily use the numbering schemes on the module edge connector. ALWAYS refer to the board edge connector configuration when wiring a connector. Connector pinouts are listed in Appendix F.

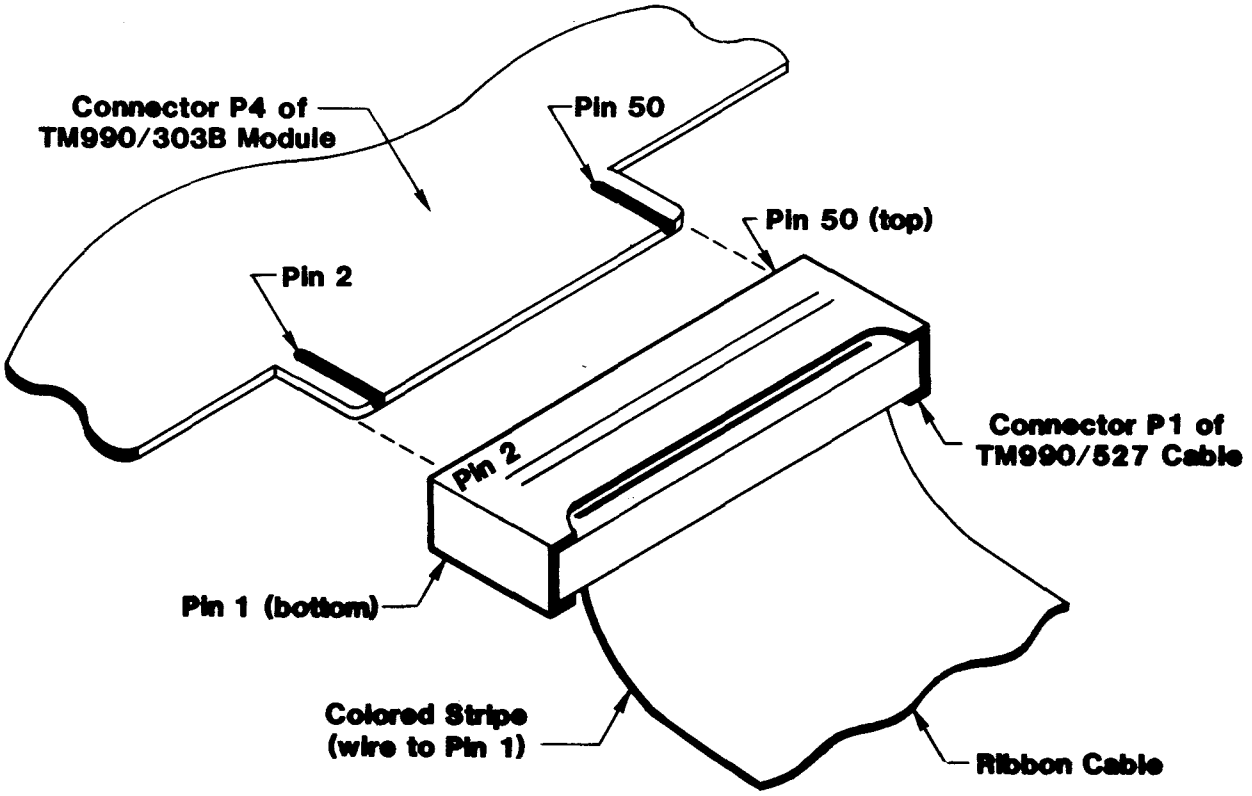


FIGURE 2-4. CONNECTING TM 990/527 DISK DRIVE CABLE TO P4 OF TM 990/303B MODULE

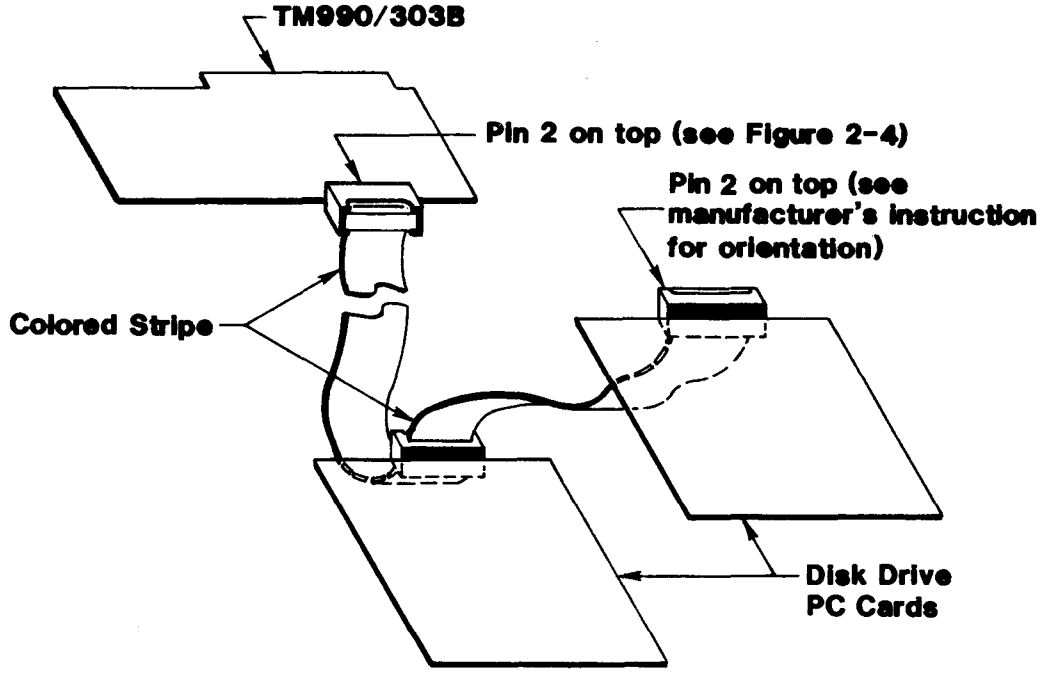


FIGURE 2-5. TM 990/527 CABLING BETWEEN CONTROLLER AND EIGHT-INCH DRIVES

Figures 2-6 and 2-7 show similar cabling for five-inch mini drives using the TM 990/535B cable and its interface card. The card acts as the interface between the 50-pin connector P4 on the TM 990/303B module and the 34-wire cable.

Appendix F contains pin designations of both cables.

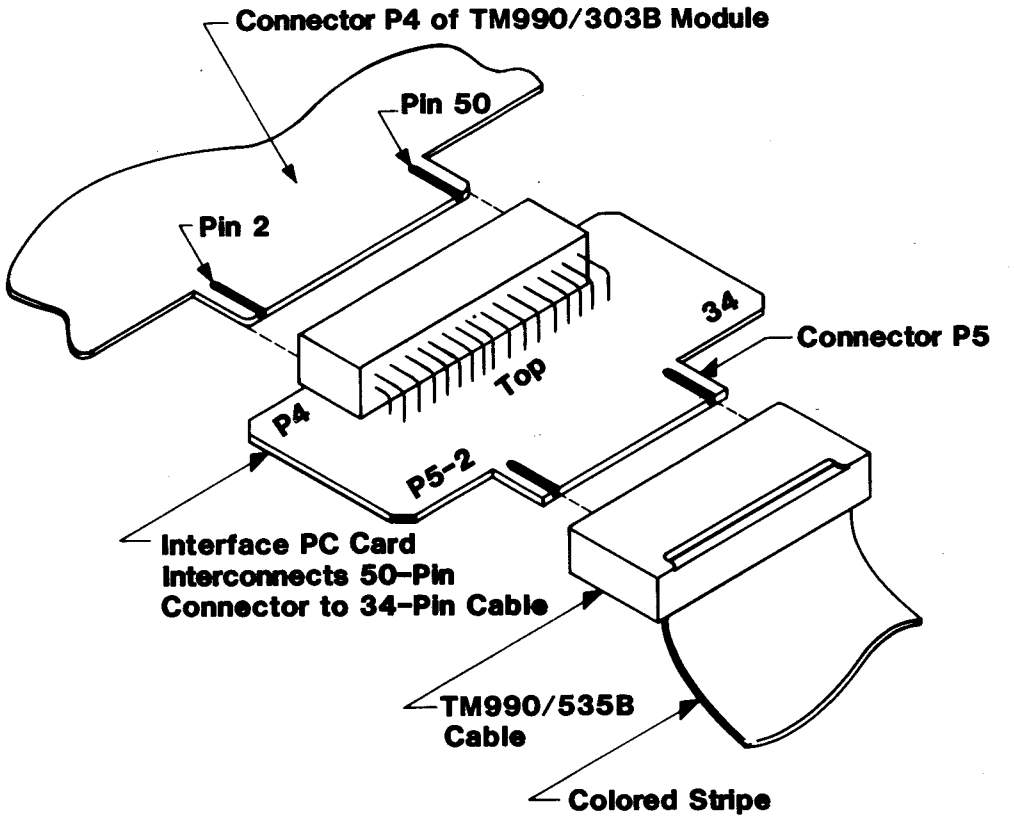


FIGURE 2-6. CONNECTING TM 990/535B DISK DRIVE CABLE TO P4 OF TM 990/303B MODULE

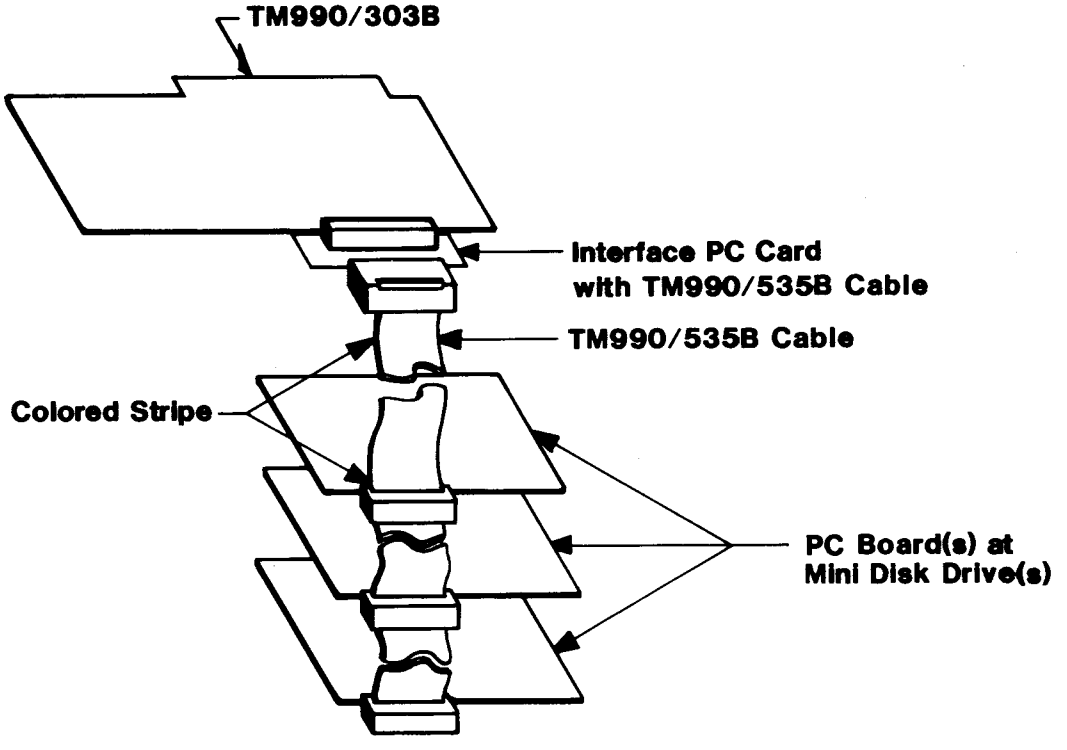


FIGURE 2-7. TM 990/535B CABLING BETWEEN CONTROLLER AND MINI DISK DRIVES

2.8 SYSTEM CHECK AND POWER APPLICATION

Do not apply power until the PC cards (TM 990/303B as well as the disk drive) are properly jumpered and installed and cables are connected as specified in section 2.7. Before applying power, use the following checklist to verify proper installation according to applicable sections, figures, and tables:

- ☐ Jumpers on TM 990/303B module (section 2.4, Figure 2-1, Table 2-1),
- ☐ Jumpers on disk drives (see applicable table in Appendix A),
- ☐ Backplane interconnection at card cage for pins 95/96 is open at TM 990/303B controller module slot (section 2.6, Figure 2-2),
- ☐ Cable attachment to standard drive (section 2.7, Figures 2-4, 2-5), or cable attachment to mini drive (section 2.7, Figures 2-6, 2-7),
- ☐ Applicable jumpers and switches set on other system modules.

Cable attachments should be as shown in Figures 2-3 to 2-7. When system installation has been checked and verified, apply power.

CAUTION

If power is being supplied from separate power supplies, the system requires that -12V be turned on first and be turned off last. There is no required sequence in turning on the remaining voltages. This does not apply if the system uses only one power supply.

With power applied, the controller will execute a self-test. LEDs DS2 and DS3 will go on. When the self-test is complete, DS2 and DS3 will go off and DS1 will go on and remain on, indicating no error. LED interpretation is shown in Table 2-2.

2.9 ONBOARD LED ERROR CHECK

Placement of the three LEDs on the TM 990/303B is shown in Figure 1-1 (DS3 to DS1, left to right as seen from the front of the card cage). These lights reflect module status as shown in Table 2-2.

2.10 TWO OR MORE TM 990/303B MODULES IN A SYSTEM

More than one TM 990/303B module may be installed in a card cage and share the bus on the motherboard. The following items must be considered:

- Access to the bus must be arbitrated
- Each TM 990/303B module must have a unique 32-bit CRU address space for communicating with the host microcomputer.

NOTE

During data transfer from the disk, the TM 990/303B requires bus access every 32 microseconds for double density formats and 64 microseconds for single density formats: if bus access is denied for longer than the specified period, then data loss will occur.

TABLE 2-2. ERROR INTERPRETATION ON LEDs

Condition/Examples	DS3	DS2	DS1
No error condition	OFF	OFF	ON
No power	OFF	OFF	OFF
Disk drive error (write protect, drive not ready, bad command, change error, seek incomplete, search error)	ON	OFF	OFF
Controller error (CRC, overrun, ID)	ON	ON	OFF
Self-test errors:			
*ROM checksum error	OFF	ON	OFF
*RAM test error	OFF	ON	ON
*TMS 9901 timer/interrupt error	ON	OFF	OFF
†Disk logic error	ON	OFF	ON
**DMA logic error	ON	ON	OFF
**Map check error	ON	ON	ON

* Self-test check performed at powerup, reset, and by self-test command.
** Self-test check performed at powerup bootload only.
† With a single mini disk drive and the MX jumper on the drive installed, a disk logic error will occur during a self-test. This is because the drive interface signals are always present on the interface with the MX jumper installed; whereas, normally they are present only when the drive is selected. The disk logic test checks that with no drive selected, the signals present on the interface are pulled high by pullup resistors. To ensure correct operation, DO NOT install the MX jumper with a single mini-drive configuration.

2.10.1 Bus Access Arbitration

Access through the common bus will be arbitrated by the GRANTIN.B-, GRANTOUT.B-, HOLD.B-, and HOLDA.B signals as shown on page 4 of the schematics in Appendix D. Through GRANTIN.B- (from P1-96) and GRANTOUT.B- (from pin P1-95), the bus is arbitrated so that the module in the highest position in the chassis (closest to the microcomputer module) has priority for the bus over the module(s) beneath it in the chassis. If a higher-priority module gains bus control when a lesser-priority module is using it, the lesser-priority module is placed in a hold state until the higher-priority module is finished: then the lower-priority module resumes control where it was interrupted.

The interconnection on the motherboard of the card cage has to be opened between the pins of 96 (top) and 95 (bottom) for the slots containing the TM 990/303B modules. This can be done by removing a jumper on the TM 990/510A, /520A, and /530 card cages, or by cutting the etch on the back of the TM 990/510 and /520 card cages. The location of these lines is shown in Figure 2-2. This allows the upper module to obtain bus control by presenting his GRANTOUT.B- signal (via pin 95 on the lower side of the TM 990/303B module) to the lower module's GRANTIN.B- line (pin 96 on the upper side of the module). Thus, the controller module with the higher priority (higher slot position) may suspend bus access to the lower-priority module.

2.10.2 Unique CRU Address Required for Each Module

Communication between the host CPU and the TM 990/303B is via the CRU. This is covered in detail in Section 3. As stated earlier, each module will require a unique CRU base address for communication with the host microcomputer. This CRU address is selected by bits programmed into the 74S287 PROM at socket U13. The 74S287 PROMs outputs are labeled D01 through D04. This socket must contain a PROM programmed with a pattern so that each module has a unique CRU address. For example (as shown in Figure 3-3 of the next section), the TM 990/303B module is shipped so that it occupies the 32 bits of CRU address space between software base address 0200_{16} and $023E_{16}$ (hardware base address 0100_{16} and $011F_{16}$). D01 is enabled for command data transfer (CUE, ACCEPT, COMMAND, BUSY, RESET, INT ISSUED, INT ENABLE) and D04 is enabled for an 8-bit data transfer (an 8-bit byte comprising part of the 20-bit address passed through the CRU). The 74S287 PROM at U13 must be programmed so that its D01 and D04 outputs have active low values on them when driven by address lines A3.B to A10.B with the correct select address present. When using multiple 303B modules, the address line values (as derived from the CRU) must be different for each module. As shipped, the U13 PROM outputs a low at D01 (command transfer through CRU) when address lines A3.B to A10.B are all zeroes except for A6.B (a one, setting a CRU base address of 200_{16}). The U13 PROM outputs a high at D04 (data transfer through CPU) when address lines A3.B to A10.B are all zeroes except for A6.B and A10.B (which are ones, setting a CRU base address of 220_{16}). Instructions to reprogram U13 are provided in Appendix E of this manual.

2.11 DEMONSTRATION PROGRAM

With the system properly connected and powered up, the program in Figures 2-8 and 2-9 (Figure 2-9 is the main program with Figure 2-8 substitute code for mini drives) can be entered into host memory using monitors such as TIBUG or MAPBUG. (This program cannot be run on the TM 990/100M/100MA microcomputers unless a memory module is in the system -- DMA cannot be accomplished to the RAM on these microcomputer modules.) This program has four command lists "chained" together (Define Drive, Format Track, Write to Disk, and Read from Disk commands). The program sequence is as follows:

- the disk controller reads the character string "Congratulations it works" from host memory and writes this message to the disk,
- the controller reads the message back from disk to another part of host RAM,
- the CPU writes the message received from disk to the system terminal.

This program is written assuming the following system characteristics:

- The system contains a TM 990/303B module, terminal, disk drive, cabling, and one of the following CPU modules:
  - TM 990/101MA or TM 990/102 CPU with system RAM from $FC00_{16}$ to $FFFF_{16}$; however, if expansion memory is used, it must not conflict with the CPU RAM address, or
  - TM 990/100MA CPU with RAM expansion memory module; in this case, be sure that expansion memory contains RAM from $FC00_{16}$ to $FFFF_{16}$ and is jumpered accordingly.

NOTE

Extended addressing is not used in this example software.

- The disk format is eight-inch IBM single-sided, single-density, and follows the format block as shown in Appendix A, Table A-1 for the Shugart SA800 disk drive. This format is called out in the define drive block of code at source lines 0054 to 0062 (line numbers are in the first column); note that the assembled values for this block match the values shown in the appendix. If another drive is used, the define format values for the other drives can be substituted using values listed in Appendix A; these values must be substituted into source lines 0054 to 0062. For convenience, Figure 2-8 contains mini disk drive substitute values for lines 0054 to 0062 for mini drives listed in the note to the figure. To use the program in Figure 2-9 with one of these mini drives, values in Figure 2-8 must be substituted for values at the corresponding line numbers in Figure 2-9.

- The user must jumper the controller module for the disk size and format desired; the TM 990/303B module comes from the factory jumpered for IBM single density, single sided, standard size.

- A TM 990 monitor is necessary for the program since monitor XOP 14 is used to print messages. The program can be inserted line-by-line into memory using the memory inspect/change command. Load the object shown in the third column into the address in the second column.

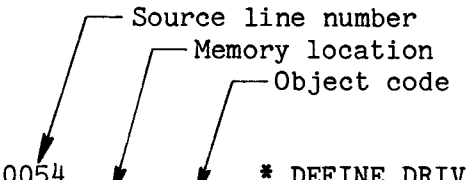
- As written, the program uses a bias of $FC00_{16}$ for relocatable code and uses the area from $FC00_{16}$ to $FDA2_{16}$ as assembled. If it is wished to relocate the program by hand to another memory location, the bias must be changed in those object codes (third column) where a tic mark (i.e., an apostrophe) is next to the object code. For example, if the object code is FC24' and you wish to locate the program (including reserved workspace) beginning at $A000_{16}$, then the $FC00_{16}$ bias must be changed to an $A000_{16}$ bias, and the resulting code would then be A024.

- The disk drive accessed is DS1 which should be jumpered DS1 at the disk drive PC board.

CAUTION

1. Data will be written to and read from the second track, first sector on the diskette (make sure that this area on the diskette can be written over).

2. Also, make sure that the controller module does not mistakely address memory on both an expansion module and on a CPU module at the same time (not a problem with the TM 990/100MA). This can happen using extended addressing with the TM 990/101MA CPU. For instance, if a TM 990/203 memory expansion module contains RAM at $FC000_{16}$ to $FFFFF_{16}$ (note, five hex digits) and the TM 990/101MA contains RAM at $F000_{16}$ to $FFFF_{16}$, then a DMA to extended addresses $FF000_{16}$ to $FFFFE_{16}$ will access memory on both modules because the CPU module does not decode the most significant four extended addressing bits. Thus, a DMA to the TM 990/203 address $FFC00_{16}$ will be decoded by the TM 990/101MA as a DMA to $FC00_{16}$.




```
0054                 * DEFINE DRIVE FORMAT BLOCK FOR IBM SINGLE DENSITY
0055 FC24 0001  FORMAT DATA >0001        WORD 0, MINI SIZE, ONE SURFACE
0056 FC26 0028         DATA 40           WORD 1, 40-TRACK DISKETTE
0057 FC28 09C4         DATA 2500         WORD 2, HEAD STEP TIME X 10 US
0058 FC2A 05DC         DATA 1500         WORD 3, STEP SETTLING TIME X 10 US
0059 FC2C 1388         DATA 5000         WORD 4, HEAD LOAD TIME X 10 US
0060 FC2E 03E8         DATA 1000         WORD 5, HEAD UNLOAD TIMEOUT (MS)
0061 FC30 0000         DATA 0            WORD 6, SYNC TYPE, INTERLACE FACTOR
0062 FC32 4080         DATA >4080        WORD 7, 16 SEC/TR, 128 BYTES/SEC
```

NOTE: Substitute this code in Figure 2-9 when using one of the following mini drives: Shugart SA450, Siemens FDD100-5, or Siemens FDD200-5. For other mini drives, use the parameters in Table A-9 in Appendix A. Change at the line numbers or memory addresses (column one and two numbers) in Figure 2-9.

FIGURE 2-8. SUBSTITUTE CODE TO USE A MINI DISK DRIVE WITH THE PROGRAM IN FIGURE 2-9

If assembled on a NON TM 990 assembler, the AORG directive is used to start the program beginning at $FC20_{16}$. The entry vector label on the END assembler directive is used to designate the program counter value for execution.

NOTE

Most TM 990 line-by-line assemblers do not use the entry vector label on the END directive.

The AORG and entry vector label on the END directive allow the program object code to be loaded from cassette or floppy with the monitor "Load" program command without a bias specified, and then executed immediately with the monitor execute command as shown below using the TIBUG monitor:

```
?L
FLOPDEMO
?E
CONGRATULATIONS, IT WORKS!

?
```

Otherwise, the program object code must be loaded directly through the monitor "M" command. Program entry is at memory address $FC82_{16}$: this value must be in the program counter before each execution of the program. The first instruction, RSET, is used by the TM 990/102 CPU to make sure it is not in the mapper-on mode.

Note that the program checks the Operation Complete bit of the Command List to determine completion of the Command List. It also enters a timeout loop in case of failure to complete the command and set the Operation Complete bit. The use of a timeout routine is necessary as a regular programming practice to avoid hangups in case the command cannot be completed and the Operation

Complete bit is not set. The timeout loop here decrements a register from $FFFF_{16}$ down to zero and does this one to ten times for a maximum timeout of approximately 5 seconds. If the command list had not been completed in this time, a message is issued telling of an error and the number of the command list being executed. After each decrement, the word 0 of the applicable command list is checked for a non-zero value, meaning that the operation had been completed and the program can move on to the next step (without completing the entire decrement loop).

```
FLOPDEMO   SDSMAC 3.3.0 79.312   13:36:06 MONDAY, MAR 15, 1982.
                                                          PAGE 000
0001                    IDT  'FLOPDEMO'
0003 FC20               AORG  FC20          PROGRAM LOAD ADDRESS
0004              *** *** *** *** *** *** *** *** *** *** *** *** *** ***
0005              ** THIS PROGRAM IS A DEMONSTRATION OF THE TM 990/303B FLOPPY
0006              ** DISK CONTROLLER WITH A TM 990/101MA OR TM 990/102 CPU
0007              ** WITHOUT ADDITIONAL MEMORY, OR A TM 990/100MA CPU WITH
0008              ** EXPANDED MEMORY. USE OF THIS EXAMPLE PROGRAM ASSUMES
0009              ** THE FOLLOWING BUT CAN BE CHANGED TO USE OTHER FORMATS:
0010              ** 1. THE CONTROLLER IS CABLED TO ONE OF THE STANDARD-SIZED
0011              **    DISK DRIVES SUITABLE FOR THIS CONTROLLER WITH THE DRIVE
0012              **    JUMPERED AS DS1.
0013              ** 2. NO INTERRUPTS TO THE HOST ARE GENERATED FOR COMMAND
0014              **    COMPLETION.
0015              ** 3. THIS PROGRAM CAN BE RE-EXECUTED CONTINUOUSLY WITH THE
0016              **    USER MAKING EXPERIMENTAL MODIFICATIONS AS DESIRED. IF
0017              **    OPERATION IS UNSUCCESSFUL -- CHECK JUMPERS ON CONTROL
0018              **    AND ON DISK DRIVE TO VERIFY FORMAT COMPATIBILITY.
0019              ** 4. DISK FORMAT IS IBM STANDARD SIZE, AND THE CONTROLLER
0020              **    MODULE IS JUMPERED THIS WAY AT J10 AND J11. SINGLE
0021              **    DENSITY IS SPECIFIED IN THE DISK DRIVE FORMAT BLOCK
0022              **    BEGINNING AT MEMORY ADDRESS >FC34 (2ND COLUMN OF LIST-
0023              **    ING). THE USER CAN CHANGE THE FORMAT PARAMETERS IN
0024              **    THIS BLOCK IF DESIRED (E.G., FOR MINI FORMAT), BUT
0025              **    JUMPER ACCORDINGLY.
0026              ** 5. IT IS SUGGESTED THAT A NEW DISKETTE BE USED FOR THIS
0027              **    DEMO PROGRAM. ONLY ONE TRACK (SECOND TRACK) WILL
0028              **    BE FORMATTED; THE FORMAT USED WILL BE IBM STANDARD.
0029              ** 6. A CPU MODULE IS USED WITH A MONITER IN LOWER MEMORY.
0030              **    RAM AT FC00-FDA2 (NO EXTENDED MEMORY) OR A MEMORY MODULE
0031              **    WITH RAM AT THESE ADDRESSES USED WITH A TM 990/100.
0032              **    IF A MEMORY MODULE IS USED WITH A TM 990/101 OR
0033              **    TM 990/102, ONBOARD RAM SHOULD BE DELETED FROM THE
0034              **    CPU MODULE.
0035              ** 7. A CHECK IS MADE FOR ERRORS IN COMMAND EXECUTION;
0036              **    HOWEVER, NO ERROR CORRECTION ROUTINES ARE PROVIDED.
0037              ** 8. DATA WILL BE WRITTEN TO THE FIRST SECTOR OF THE SECOND
0038              **    TRACK ON THE DISKETTE. TAKE CARE THAT DATA WILL NOT BE
0039              **    DESTROYED AT THIS ADDRESS.
0040              ** 03/20/82  J.J.WALSH
0041              ** 03/20/82  M.B.ALYN
0042              *** *** *** *** *** *** *** *** *** *** *** *** *** ***
0043              ** *  *  *  *  *  *  *  *  *  *  *  *  *  *  *  *  *  *
0044              ** EQUATES, COMMAND LISTS, AND COMMAND LIST DATA
0045              ** *  *  *  *  *  *  *  *  *  *  *  *  *  *  *  *  *  *
0046              * EQUATE MNEMONICS
0047       0008   COMND  EQU  8             DISPLACEMENT ON CRU FOR COMMAND BIT
0048       000A   CUE    EQU  10            DISPLACEMENT ON CRU FOR CUE BIT
0049       000B   ACCEPT EQU  11            DISPLACEMENT ON CRU FOR BUSY BIT
0050       000C   BUSY   EQU  12            DISPLACEMENT ON CRU FOR ACCEPT BIT
0051              * ADDRESS BYTES FOR FIRST COMMAND LIST
0052 FC20 0000   CMLST1 DATA >0000         EXTENDED ADDR. ALL ZEROES (NO EXT.)
0053 FC22 FC34'         DATA CLIST1        BYTES 2 AND 3 (1ST COMMAND LIST ADDR)
0054              * DEFINE DISK FORMAT BLOCK FOR IBM SINGLE DENSITY
0055 FC24 0101   FORMAT DATA >0101         DISK SIZE, NO. OF SURFACES
0056 FC26 004D          DATA 77            NUMBER OF TRACKS
```

FIGURE 2-9. DEMO PROGRAM TO READ TO/WRITE FROM DISK (SHEET 1 OF 5)

```
0057 FC28 03E8          DATA 1000          HEAD STEP TIME X 10 US
0058 FC2A 05DC          DATA 1500          STEP SETTLING TIME X 10 US
0059 FC2C 0DAC          DATA 3500          HEAD LOAD TIME X 10 US
0060 FC2E 03E8          DATA 1000          HEAD UNLOAD TIMEOUT (MS)
0061 FC30 0000          DATA 0             DENSITY, SYNC TYPE, INTERLACE FACTOR
0062 FC32 6880          DATA >6880         SECTORS/TRACK
0063                ** FOUR COMMAND LISTS,
0064                ** CHAINED (VIA WORDS 8 AND 9)
0065                ** TO EXECUTE IN SUCCESSION
0066                * FIRST COMMAND LIST -- DEFINE DISK FORMAT
0067 FC34 0000  CLIST1 DATA 0             WORD 0, CLEAR FLAGS
0068 FC36 0000          DATA 0             WORD 1, CLEAR FLAGS
0069 FC38 1000          DATA >1000         WORD 2, DEFINE DRIVE FORMAT COMMAND
0070 FC3A 0000          DATA 0             WORD 3. NOT NEEDED
0071 FC3C 0000          DATA 0             WORD 4, NOT NEEDED
0072 FC3E 0000          DATA 0             WORD 5, NOT NEEDED
0073 FC40 0000          DATA 0000          WORD 6, ADDRESS OF THE --
0074 FC42 FC24'         DATA FORMAT        WORD 7, FORMAT PARAMETERS ADDRESS
0075 FC44 8000          DATA >8000         WORD 8, CHAIN TO NEXT COMMAND LIST
0076 FC46 FC48'         DATA CLIST2        WORD 9, NEXT COMMAND LIST ADDRESS
0077                * SECOND COMMAND LIST -- FORMAT SECOND TRACK OF SIDE 0
0078 FC48 0000  CLIST2 DATA 0             WORD 0, CLEAR FLAGS
0079 FC4A 0000          DATA 0             WORD 1, CLEAR FLAGS
0080 FC4C 0900          DATA >0900         WORD 2, FORMAT TRACK COMMAND
0081 FC4E 8001          DATA >8001         WORD 3, FORMAT SECOND TRACK (01)
0082 FC50 0001          DATA >0001         WORD 4, SURFACE/SECTOR ADDRESS
0083 FC52 0000          DATA 0             WORD 5, NOT NEEDED
0084 FC54 0000          DATA 0000          WORD 6, FIRST BYTE OF THE 20-BIT ADDR
0085 FC56 FD80'         DATA PATERN        WORD 7, ADDRESS OF FORMAT PATTERN
0086 FC58 8000          DATA >8000         WORD 8, CHAIN TO NEXT COMMAND ADDRESS
0087 FC5A FC5C'         DATA CLIST3        WORD 9, NEXT COMMAND LIST ADDRESS
0088                * THIRD COMMAND LIST -- WRITE TO DISKETTE
0089                * PHYSICAL STORAGE MODE USED IN WORDS 3 AND 4
0090 FC5C 0000  CLIST3 DATA 0             WORD 0. CLEAR FLAGS
0091 FC5E 0000          DATA 0             WORD 1, CLEAR FLAGS
0092 FC60 0400          DATA >0400         WORD 2, WRITE TO DISK DS1 COMMAND
0093 FC62 8001          DATA >8001         WORD 3, (& 4) TRACK ADDRESS
0094 FC64 0001          DATA >0001         WORD 4, SURFACE/SECTOR ADDRESS
0095 FC66 0080          DATA >80           WORD 5, WRITE >80 BYTES (1 SECTOR)
0096 FC68 0000          DATA 0000          WORD 6, ADDRESS OF THE --
0097 FC6A FD5E'         DATA MESG          WORD 7, MESSAGE IN HOST
0098 FC6C 8000          DATA >8000         WORD 8, CHAIN TO NEXT COMMAND LIST
0099 FC6E FC70'         DATA CLIST4        WORD 9, NEXT COMMAND LIST ADDRESS
0100                * FOURTH COMMAND LIST -- READ FROM DISKETTE TO CPU MEMORY
0101                * MASS STORAGE MODE USED IN WORDS 3 AND 4
0102 FC70 0000  CLIST4 DATA 0             WORD 0, CLEAR FLAGS
0103 FC72 0000          DATA 0             WORD 1, CLEAR FLAGS
0104 FC74 0300          DATA >0300         WORD 2, READ FROM DISK DS1 COMMAND
0105 FC76 0000          DATA 0             WORD 3, (& 4) TRACK, SURFACE AND
0106 FC78 0D00          DATA >0D00         WORD 4, SECTOR ADDR - MASS STORAGE
0107 FC7A 0080          DATA >80           WORD 5, READ THE SECTOR
0108 FC7C 0000          DATA 0             WORD 6. MASS STORAGE ADDRESS OF
0109 FC7E FD82'         DATA FNLMSG        WORD 7, MESSAGE FROM DISK
0110 FC80 0000          DATA 0             WORD 8, NO CHAINING
```

FIGURE 2-9. DEMO PROGRAM TO READ TO/WRITE FROM DISK (SHEET 2 OF 5)

```
0112                    ** * * * * * * * * * * * * * * * * * * * * * * *
0113                    ** TASK AREA
0114                    ** * * * * * * * * * * * * * * * * * * * * * * *
0115                    ** ROUTINE TO SEND 3 COMMAND LIST ADDRESS BYTES
0116                    ** TO DISK CONTROLLER MODULE THROUGH CRU
0117                    **
0118 FC82 0360  START  RSET              MAPPER OFF (TM 990/102 ONLY)
0119 FC84 02E0         LWPI >FC00        DEFINE WORKSPACE POINTER
     FC86 FC00
0120 FC88 020C         LI   R12,>210     CRU SOFTWARE BASE ADDRESS
     FC8A 0210
0121                   * INITIALIZE CONTROLLER THROUGH CRU
0122 FC8C 1D0E         SBO  14           RESET CONTROLLER
0123 FC8E 1E0E         SBZ  14           RELEASE RESET, OPERATE
0124                   * TEST CRU CONDITIONS; IF ACCEPT OR BUSY NOT ZERO,
0125                   * USE RESET TO CLEAR THESE (IN ACTUAL PRACTICE,
0126                   * USE LOOP UNTIL BITS BECOME SET)
0127 FC90 1F0B  ACEP1  TB   ACCEPT       ACCEPT = ZERO?
0128 FC92 13FE         JEQ  ACEP1        NO, LOOP UNTIL ACCEPT = 0
0129 FC94 1F0C  BUSY1  TB   BUSY         YES, BUSY = ZERO?
0130 FC96 13FE         JEQ  BUSY1        NO, LOOP UNTIL BUSY = 0
0131                   * SET UP ADDRESS OF FIRST COMMAND LIST IN R2
0132 FC98 0202         LI   R2,CMLST1+1  MSB'S IN SECOND BYTE
     FC9A FC21'
0133                   * FOR FIRST ADDRESS BYTE, COMMAND BIT = 1
0134 FC9C 1D08         SBO  COMND        COMMAND BIT A ONE
0135 FC9E 3232         LDCR *R2+,8       ADDRESS BYTE TO CRU
0136 FCA0 1D0A         SBO  CUE          CAUSE INTERRUPT
0137 FCA2 1F0B  ACCEP1 TB   ACCEPT       CONTR RECV BYTE? (ACCEPT=1?)
0138 FCA4 16FE         JNE  ACCEP1       NO. LOOP UNTIL RECEIVED
0139 FCA6 1E0A         SBZ  CUE          YES, ACKNOWLEDGE THIS
0140 FCA8 1F0B  ACCEP2 TB   ACCEPT       ACCEPT = 0?
0141 FCAA 13FE         JEQ  ACCEP2       NO, LOOP UNTIL ACCEPT=0
0142 FCAC 1E08         SBZ  COMND        COMMAND=0 FOR BYTES 2 & 3
0143                   * SEND SECOND ADDRESS BYTE
0144 FCAE 3232         LDCR *R2+,8       ADDRESS BYTE TO CRU
0145 FCB0 1D0A         SBO  CUE          CAUSE INTERRUPT
0146 FCB2 1F0B  ACCEP3 TB   ACCEPT       CONTR RECV BYTE? (ACCEPT=1?)
0147 FCB4 16FE         JNE  ACCEP3       NO. LOOP UNTIL RECEIVED
0148 FCB6 1E0A         SBZ  CUE          YES, ACKNOWLEDGE THIS
0149 FCB8 1F0B  ACCEP4 TB   ACCEPT       ACCEPT = 0?
0150 FCBA 13FE         JEQ  ACCEP4       NO, LOOP UNTIL ACCEPT=0
0151                   * SEND THIRD ADDRESS BYTE
0152 FCBC 3212         LDCR *R2,8        ADDRESS BYTE TO CRU
0153 FCBE 1D0A         SBO  CUE          CAUSE INTERRUPT
0154 FCC0 1F0B  ACCEP5 TB   ACCEPT       CONTR RECV BYTE? (ACCEPT=1?)
0155 FCC2 16FE         JNE  ACCEP5       NO. LOOP UNTIL RECEIVED
0156 FCC4 1E0A         SBZ  CUE          YES, ACKNOWLEDGE THIS
0157 FCC6 1F0B  ACCEP6 TB   ACCEPT       ACCEPT = 0?
0158 FCC8 13FE         JEQ  ACCEP6       NO, LOOP UNTIL ACCEPT=0
0159                   ** THIRD BYTE OF FIRST ADDRESS SENT
0160                   ** SET UP COMMAND LIST 1 NUMBER IN ERROR MESSAGE
0161 FCCA 0205         LI   R5,>2031     ASCII SPACE AND 1 IN R5
     FCCC 2031
0162 FCCE C805         MOV  R5,@WORD     MOVE TO MESSAGE
     FCD0 FD58'
```

FIGURE 2-9. DEMO PROGRAM TO READ TO/WRITE FROM DISK (SHEET 3 OF 5)

```
0164                    ** CLEAR FIRST WORD OF EACH COMMAND LIST AND FINAL
0165                    ** MESSAGE LOCATION IN CASE THIS DEMO PROGRAM IS
0166                    ** EXECUTED AGAIN AND AGAIN.
0167 FCD2 04E0                 CLR  @CLIST1
     FCD4 FC34'
0168 FCD6 04E0                 CLR  @CLIST2
     FCD8 FC48'
0169 FCDA 04E0                 CLR  @CLIST3
     FCDC FC5C'
0170 FCDE 04E0                 CLR  @CLIST4
     FCE0 FC70'
0171 FCE2 04E0                 CLR  @FNLMSG      ZERO PREVENTS MESSAGE WRITE
     FCE4 FD82'
0172                    ** SET UP FOR FIRST WORD OF EACH COMMAND LIST TO BE
0173                    ** BE READ BY TIMEOUT AND OPERATION COMPLETE ROUTINE.
0174                    ** INCREMENT NUMBER OF COMMAND LIST IN ERROR
0175                    ** (BIT 0 A ONE = COMMAND COMPLETE)
0176                    ** NO ERROR CORRECTION ROUTINE PROVIDED IN THIS DEMO PROGRAM
0177 FCE6 0204                 LI   R4,CLIST1    SET UP 1ST LIST ADDR
     FCE8 FC34'
0178 FCEA 06A0                 BL   @TIMOUT      GO TIME OUT
     FCEC FD22'
0179 FCEE 05A0                 INC  @WORD        INCREMENT WORD IN ERROR MESSAGE
     FCF0 FD58'
0180 FCF2 0204                 LI   R4,CLIST2    SET UP 2ND LIST ADDR
     FCF4 FC48'
0181 FCF6 06A0                 BL   @TIMOUT      GO TIME OUT
     FCF8 FD22'
0182 FCFA 05A0                 INC  @WORD        INCREMENT WORD IN ERROR MESSAGE
     FCFC FD58'
0183 FCFE 0204                 LI   R4,CLIST3    SET UP 3RD LIST ADDR
     FD00 FC5C'
0184 FD02 06A0                 BL   @TIMOUT      GO TIME OUT
     FD04 FD22'
0185 FD06 05A0                 INC  @WORD        INCREMENT WORD IN ERROR MESSAGE
     FD08 FD58'
0186 FD0A 0204                 LI   R4,CLIST4    SET UP 4TH LIST
     FD0C FC70'
0187 FD0E 06A0                 BL   @TIMOUT      GO TIME OUT
     FD10 FD22'
0188                    ** MESSAGE WRITTEN TO DISKETTE
0189                    ** THEN READ FROM DISKETTE TO NEW HOST MEMORY LOCATION
0190                    ** NOW WRITE TO TERMINAL FROM NEW HOST ADDRESS
0191 FD12 2FA0  WRITE  XOP  @FNLMSG,14   WRITE MESSAGE USING TIBUG
     FD14 FD82'
0192                    * IF USING /102 MAPBUG MONITOR, MUST BRANCH AT >A0
0193 FD16 0460                 B    @>80         BRANCH TO TIBUG. MISSION COMPLETE
     FD18 0080
0194                    ** ERROR ROUTINES IF DEMONSTRATION FAILS, SUGGEST THAT
0195                    ** USER CHECK JUMPER SETTINGS TO SEE THAT THEY AGREE
0196                    ** WITH THE FORMAT SELECTED (DEFAULT IS IBM STANDARD,
0197                    ** SINGLE DENSITY. CHECK BOTH CONTROLLER AND DISK DRIVE.
0198 FD1A 2FA0  ERROR  XOP  @ERRMSG,14   SEND ERROR MESSAGE
     FD1C FD40'
0199 FD1E 0460                 B    @>80         BRANCH TO TIBUG FOR USER INPUTS
     FD20 0080
```

FIGURE 2-9. DEMO PROGRAM TO READ TO/WRITE FROM DISK (SHEET 4 OF 5)

```
0200                    ** COMMAND TIMEOUT ROUTINE READS FIRST WORD OF COMMAND
0201                    ** LIST FOR COMPLETION AND ALSO RUNS TIMER IN CASE
0202                    ** COMPLETION DOES NOT OCCUR WITHIN A SPECIFIED TIME;
0203                    ** IF TIMEOUT OCCURS OR ERROR FOUND, ERROR MESSAGE
0204                    ** WRITTEN OUT.
0205 FD22 0205  TIMOUT LI   R5,10        SET UP MASTER COUNTER
     FD24 000A
0206 FD26 0706  TIMOU1 SETO R6           SET UP INNER COUNTER
0207 FD28 0606  TIMOU2 DEC  R6           DECREMENT INNER COUNTER
0208 FD2A 16FE         JNE  TIMOU2       IF NOT 0, KEEP LOOPING
0209 FD2C C0D4         MOV  *R4,R3       LOOK AT WORD 0 FOR COMPLETION
0210 FD2E 1605         JNE  TIMOU4       IF COMPLETE GO CHECK ERROR
0211 FD30 0605         DEC  R5           NOT COMPLETE, DECREMENT COUNTER
0212 FD32 16F9         JNE  TIMOU1       IF NOT 0, GO TO TOP OF OTHER LOOP
0213 FD34 020B  TIMOU3 LI   R11,ERROR    SET UP ERROR RETURN
     FD36 FD1A'
0214 FD38 045B         RT                ERROR RETURN (*R11)
0215 FD3A 0A13  TIMOU4 SLA  R3,1         DO WE HAVE ERRORS ?
0216 FD3C 16FB         JNE  TIMOU3       IF YES, GO TO ERROR EXIT
0217 FD3E 045B         RT                NO-ERROR EXIT (*R11)
0218                    ** *  *  *  *  *  *  *  *  *  *  *  *  *  *  *  *  *  *  *  *
0219                    ** DEMO MESSAGES
0220                    ** *  *  *  *  *  *  *  *  *  *  *  *  *  *  *  *  *  *  *  *
0221                    ****** ERROR MESSAGE ******
0222 FD40   0A  ERRMSG BYTE >0A,>0D      LINE FEED CARRIAGE RETURN
     FD41   0D
0223 FD42   45         TEXT 'ERROR IN COMMAND LIST '
     FD43   52
     FD44   52
     FD45   4F
     FD46   52
     FD47   20
     FD48   49
     FD49   4E
     FD4A   20
     FD4B   43
     FD4C   4F
     FD4D   4D
     FD4E   4D
     FD4F   41
     FD50   4E
     FD51   44
     FD52   20
     FD53   4C
     FD54   49
     FD55   53
     FD56   54
     FD57   20
0224 FD58 0000  WORD   DATA 0            WORD NUMBER PLACED HERE
0225 FD5A   0A         BYTE >0A,>0D,>07  LF, CR, BELL
     FD5B   0D
     FD5C   07
0226 FD5D   00         BYTE 0            END-OF-MESSAGE DELIMITER
```

FIGURE 2-9. DEMO PROGRAM TO READ TO/WRITE FROM DISK (SHEET 5 OF 5)

SECTION 3

COMMUNICATING WITH THE TM 990/303B DISK CONTROLLER

3.1 GENERAL

This section describes the methods for communicating between a host microcomputer and the TM 990/303B disk controller. This section is designed to help the user construct the device service routine (DSR) for handling the storage and retrieval of data to and from the TM 990/303B. Included are the following means of communication:

1. Initial contact through Communication Register Unit (CRU) (section 3.3)

2. Communication via Command List in host memory (section 3.4)

3. Interrupts to both controller (section 3.2) and host (section 3.5)

4. Bootstrap load at powerup (section 3.6)

Initial method of communication would be through the CRU (1 above) or bootstrap load at powerup (4 above). Using the CRU, explained in section 3.3, a 19-bit address is passed to the disk controller; this is the address in host memory of a ten-word block that defines a command for the controller to execute. This ten-word block is called the Command List and is used to transfer command data to the controller and return status and error data to the host. Command data to the controller includes the number of bytes to transfer, data addresses in host memory and on diskette, and the chaining address of the next Command List to be executed. The controller accesses the Command List via direct memory access (DMA). The controller executes the command in the Command List and reports back completion status (successful completion, error occurrence, etc.) via the same Command List in host memory. Communication through the Command List in memory is explained in detail in section 3.4.

The disk controller can also indicate command completion (successful or otherwise) via a dedicated interrupt. The specified interrupt level must be jumpered at the disk controller, and the host is responsible for enabling the interrupt at the CRU interface, in the Command List, at the host TMS 9901 interface, and at the host microprocessor interrupt mask. Interrupts to the host are covered in section 3.5.

Some transactions through the Command List require the use of data placed in other host memory blocks. A series of Command Lists can be "chained" by giving the memory address of the next Command List in the last two words of the present Command List. Thus, one beginning Command List address entered through the CRU can be used to start execution of a series of commands defined in a series of Command Lists, each list located in its own memory location. If one command in the chain is terminated prematurely, execution of the present command and future commands in the chain is terminated.

If jumpered, a bootstrap load can be initiated at powerup. In this case, the bootload program is brought into memory from a sector on a formatted diskette (the first sector of the first track). This bootload program usually is used to bring in a program from diskette that will initialize the system. Different bootload formats are explained in Table 3-2, Command 05.

3.2 CONSIDERATIONS

1. If an error occurs during a command to the disk controller, the executing command is terminated. If the command was part of a "chain" of commands, termination of this command also terminates the execution of the other commands in the chain. ("Chaining" is where the address of the next Command List is obtained from the previous Command List, etc.) Thus, it is recommended that an interrupt be issued at the end of every command to allow an interrupt service routine to determine corrective action in case operation is terminated other than successfully.

2. Interrupt service routines at the host, as a result of command completion, should consider the following:

   - Maintenance by the host microcomputer of a Command List address table pointer showing the address of the just-completed Command List so that errors can be monitored by the interrupt service routine.

   - Error handling routines.

   - Reenabling of interrupts at the host CRU, TMS 9901, and microprocessor as well as specifying interrupts in the Command List.

   Interrupts and interrupt service routines are covered in more detail in section 3.5.

3. If it is desired to reset the host microcomputer upon a powerup bootload, jumper J14 must be installed correctly on the controller module. Jumper J14 is required with CPUs such as the TM 990/100MA or TM 990/101MA, but not with the TM 990/102 which asserts PRES.B- to its microprocessor at powerup.

4. The controller will prevent any writing to certain parts of its RAM (e.g. writing to it via the Write Controller Memory command) because parts of controller memory are reserved for functions such as memory mapping. The only area of controller RAM that can be written to is from address $FC06_{16}$ to $FFFE_{16}$. See Figure 3-1.

5. The host should initialize the error status bits to zeros before executing a command because the disk controller writes to the Command List, setting the appropriate error and status bits.

6. Should the controller become "locked up," recovery would be through writing a one, then a zero to the RESET bit on the CRU: issuing a firmware Reset Command would be ineffective since the controller is locked up and cannot accept the firmware command. Lock up situations are discussed in section 3.3.1.5.

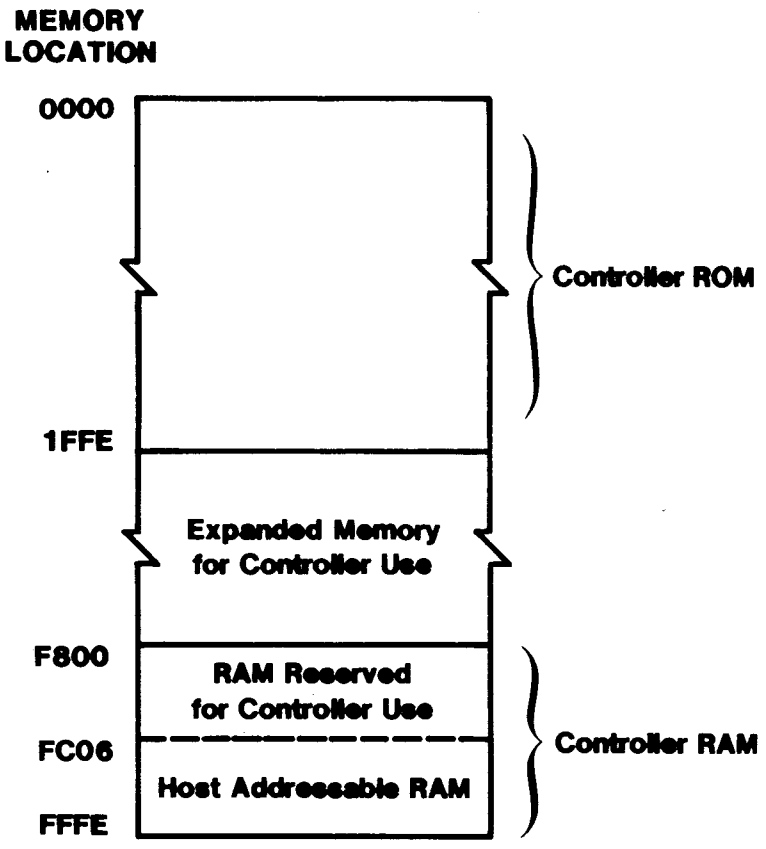


FIGURE 3-1. DISK CONTROLLER MEMORY MAP

7. Completion of a command by the controller can be determined by the host by checking the Operation Complete bit of word 0 in the Command List. However, it is possible that the controller can inadvertently "lock up" during operation and never set the Operation Complete bit. Because of this, a timeout routine should also be used that would transfer host control to a recovery routine should the controller not complete its command in a proper time period. Such a timeout routine is shown in the demonstration program in Section 2 (Figure 2-9, starting at source line 0200). In this example routine, a continuous check is made of word 0 during each count of a timing loop (word 0 had been initialized to all zeroes). If this zero state does not change (command completion will set one or more bits in the word) during the specified loop time period, host control is transferred so that a recovery can be made (instead of lockup). See the next consideration below.

8. As the controller has the ability to execute program code passed to its memory (firmware command 08), the code passed to the disk controller for controller execution must be thoroughly tested beforehand. The controller could become inadvertently "locked up" by having the passed program accessing some sensitive TM 990/303B CRU and memory locations. The host could check for a "locked up" situation via a timeout operation. (Such a timeout routine, also discussed above, is shown in the fifth page of the listing in Figure 2-9, beginning at source line 0200: recovery could be through the RESET bit on the CRU interface as discussed in consideration number 6.)

9. Care must be taken when passing position-dependent code to the disk controller memory and then commanding the controller to execute the code (Write Controller Memory and Execute Controller Memory commands). The controller cannot rebias any code passed to the TM 990/303B memory.

10. Do not attempt to read or write across 64 K memory boundaries (this consideration applies to systems using extended addressing). Instead, use multiple writes to write across such boundaries. For example, to write from diskette to host memory space $FC00_{16}$ to $101FE_{16}$, execute two writes -- one to $FC00_{16}$ to $FFFE_{16}$ and a second to 10000_{16} to $101FE_{16}$.

11. The controller may be used with different disk formats, and the user must be aware of changes in hardware as well as software when changing controller use from one format to another. For example, when changing from a standard-size diskette drive to a mini diskette drive, the user can specify the new format in software using the Define Drive command. However, he also must change jumper settings on the TM 990/303B module (i.e., change jumpers J10 and J11 from the STD setting to MINI setting). Make sure power is off when switching from one model drive to another.

12. It is not recommended that a system configuration contain drives of different make or model. Although each make of drive uses one of two diskette sizes (standard or mini), there are significant differences between standard and mini formats.

13. The onboard memory of the model TM 990/100M or TM 990/100MA microcomputer cannot be accessed via DMA and must be outside the memory map of the expansion memory. See consideration 14 below. To use these CPUs with the TM 990/303B, an expansion memory module is needed. Make sure the expansion module is mapped outside the CPU's onboard memory.

14. When using an expansion memory module, be sure that the CPU memory is not at the same location as expansion memory. This is especially true when using 19-bit (extended) addressing. For example, accessing $1FF00_{16}$ in a system can cause an access to that address on a 19-bit extended addressing memory module as well as attempt to access $FF00_{16}$ on a CPU module that does not decode extended addressing (e.g. TM 990/101MA CPU).

15. If the TM 990/303B is to be used with the TM 990/101M module, the /101 board must be at revision B or later. Board revision level is shown on the non-component side following the PCB number "BD 994725-1." For example, PCB number BD 994725-1A (B) or BD 994725-1 B are at the correct revision level; however, BD 994725-1A is not at the correct revision level. See section 1.1 for detailed information.

16. When using IBM double density format, do not attempt a multiple-track read or write originating from Side 0, Track 0. Track 0 is single density (it holds half as much data as a double density track) and therefore, has a hole from mass storage address $D00_{16}$ to $19FE_{16}$.

3.3 COMMUNICATION THROUGH THE CRU (Software Base Address 210_{16})

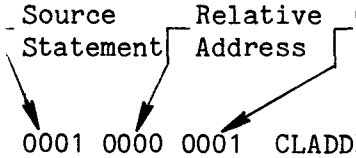
Initial communication between the host and disk controller is through the CRU. Using the CRU, the disk controller is told the 19-bit address of the Command List to be executed. Initially, the transfer of this address is via three data transfers in which one byte of the three-byte address is sent to the controller in each transfer (there is a command that can set up a default address). Three bytes contain the Command List address (in the order sent using the CRU):

1. Four zeroes followed by the most significant four bits of the 19-bit address. If a memory module with extended addressing (19-bit address) capability is used, set these bits to the value as mapped. If extended addressing is not decoded, these bits should be zeroes.

2. The most-significant eight bits of the remaining 15-bits.

3. The least-significant seven bits of the remaining 15 bits and an additional least-significant zero to make eight bits.

For example, suppose the address of the command list is $1FE80_{16}$, then the following code will transfer the address to the controller via the CRU once the proper handshaking is made:



```
Source     Relative  Object
Statement  Address   Code

0001 0000 0001  CLADDR  DATA 0001      CMD LIST ADDR: MOST SIG 4 BITS
0002 0002 FE80         DATA >FE80     LEAST SIGNIFICANT 15 BITS AND AN LS BIT (0)
                       .
                       .          (handshaking to set up communication via CRU)
                       .
0003 0004 0021         LI R1,CLADDR+1   POINT TO FIRST ADDRESS BYTE (HEX 01)
     0006 0001'
0004 0008 3231         LDCR *R1+,8        TRANSFER FIRST BYTE (HEX 01)
                       .
                       .         (handshaking with controller)
                       .
0005 000A 3231         LDCR *R1+,8        TRANSFER SECOND BYTE (HEX FE)
                       .
                       .         (handshaking with controller)
                       .
0006 000C 3211         LDCR *R1,8         TRANSFER THIRD BYTE (HEX 80)
0007 000E
                       .
                       .         (handshaking with controller, etc.)
                       .
```

This initial communication is via 16 CRU bits starting at CRU software base address 0210_{16} (or at the base address specified in the PROM at socket U13). The CRU software base address is contained in R12. The user can program his own PROM for an alternate address and insert it in this socket (this is described in Appendix E).

Figure 3-2 shows the signals, signal timing, and data bits that are output to and input from the disk controller over the CRU.

NOTES

1. In Figure 3-2, a software base address of 210_{16} is used (hardware base address 108_{16} as shown in the 32-bit CRU block in Figure 3-3). The middle 16 bits of the 32-bit CRU address block are used; the first and last 8 bits are reserved.

2. Communicating through the CRU can be made more efficient by using command code 12_{16}, which sets up a default address for all CRU communications. By using this command, a non-busy controller can be interrupted to execute a Command List by setting CUE high. When ACCEPT goes high, the controller is retrieving the Command List from the default address.

TO DISK CONTROLLER	CRU BIT ADDRESSED	FROM DISK CONTROLLER
	HDW BASE + 0	0
LSB ↑	1	0
COMMAND	2	0
LIST	3	0
ADDRESS	4	0
↓	5	0
	6	0
MSB	7	0
COMMAND	8	0
0	9	0
CUE	10	0
0	11	ACCEPT
0	12	BUSY
INTERRUPT ENABLE	13	0
RESET	14	0
0	HDW BASE + 15	INTERRUPT ISSUED

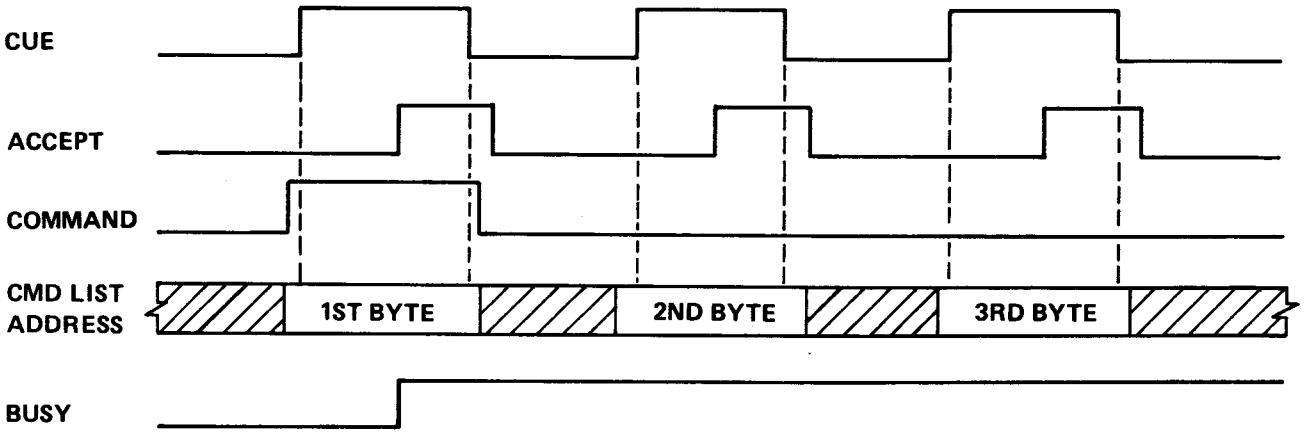


FIGURE 3-2. CRU INTERFACE AND TIMING

The entire CRU addressing scheme for each TM 990/303B takes up 32 bit blocks of CRU address space. However, only the middle 16 bits of the 32 bit CRU address are used; the first and last 8 bits are reserved. Figure 3-3 shows these 16 CRU bits in relation to the 32-bit block being addressed starting at CRU software base address 200_{16}.

	To Disk Controller	CRU Bit (Software Base Addresses in parentheses)	From Disk Controller
	LSB	(200)	0
	↑		0
	Command		0
	List		0
	Address		0
			0
			0
	MSB	(20E)	0
CRU Bits Used	LSB	Hdwr Base + 0 (210)	0
	↑	1 (212)	0
	Command	2 (214)	0
	List	3 (216)	0
	Address	4 (218)	0
		5 (21A)	0
		6 (21C)	0
	MSB	7 (21E)	0
	COMMAND	8 (220)	0
	0	9 (222)	0
	CUE	10 (224)	0
	0	11 (226)	ACCEPT
	0	12 (228)	BUSY
	INTERRUPT ENABLE	13 (22A)	0
	RESET	14 (22C)	0
	0	Hdwr Base + 15 (22E)	INTERRUPT ISSUED
	COMMAND	(230)	0
	0		0
	CUE		0
	0		ACCEPT
	0		BUSY
	INTERRUPT ENABLE		0
	RESET		0
	0	(23E)	INTERRUPT ISSUED

FIGURE 3-3. 32-BIT CRU INTERFACE BLOCK AS SHIPPED FROM FACTORY

Figure 3-4 (a) shows the "handshaking" between the disk controller and host to effect the transfer of the Command List Address. Figure 3-4 (b) shows the "handshaking" used in communicating with the TM 990/303B through the default command list address method. Figure 3-5 is example of code to make the transfer.

HOST	DISK CONTROLLER
1. Initial Setup: a. Set CRU software base address in R12. b. Set counter of address bytes to 3*. c. Wait until BUSY & ACCEPT are zeroes.	
2. Set COMMAND bit to one (next data byte is first of three bytes of address).	
3. Load data byte (address byte) onto CRU.	
4. Set CUE bit to 1 to cause interrupt to disk controller (INT1-).	
	5. Is COMMAND bit a one? (1st byte is being sent?)
	6. If yes, set interrupt mask, set BUSY bit to a one, and set byte counter to 3*.
	7. Store 1st byte.
	8. Set ACCEPT bit to one.
9. Is ACCEPT bit a one? If no, wait.	
10. If yes, set CUE to zero. Set COMMAND bit to zero.	
	11. Is CUE a zero?
	12. If yes, set ACCEPT to zero.
	13. Decrement counter (to 2).
14. Is ACCEPT a zero? If no, wait.	
15. If yes, decrement counter (to 2).	
16. Load 2d byte on CRU; set CUE to 1.	
	17. Is CUE a one?
	18. If yes, check COMMAND; if COMMAND is a one, return to step 6.
	19. If COMMAND a zero, store 2d byte.
	20. Set ACCEPT to one.
21. Is ACCEPT a one? If no, wait.	
22. If yes, set CUE to zero.	
	23. Is CUE a zero?
	24. If yes, set ACCEPT to zero.
	25. Decrement counter (to 1).
26. Is ACCEPT a zero? If no, wait.	
27. If yes, decrement counter (to 1).	
28. Load 3d byte on CRU, set CUE=1.	
	29. Is CUE a one?
	30. If yes, check COMMAND; if COMMAND is a one, return to step 6.
	31. If COMMAND a zero, store 3d byte.
	32. Set ACCEPT to one.
33. Is ACCEPT a one? If no, wait.	
34. If yes, set CUE to zero.	
	35. Is CUE a zero?
	36. If yes, set ACCEPT to zero
	37. Decrement counter (to 0).
	38. Counter = 0, go execute command.
39. Is ACCEPT a zero? If no, wait.	
40. If yes, decrement counter (to 0); exit.	

*NOTE: When counter reaches zero, routine is exited.

NOT USING DEFAULT ADDRESS

FIGURE 3-4 (A). COMMUNICATION BETWEEN THE HOST AND DISK CONTROLLER TO STORE COMMAND LIST ADDRESS THROUGH THE CRU

HOST	DISK CONTROLLER
1. Initial Setup: a. Default Address Command has been executed; thus, a default address will be used to locate the command list. b. Set CRU software base address in R12. c. Wait until BUSY & ACCEPT are zeroes.	
2. Set COMMAND bit to one.	
3. Set CUE bit to 1 to cause interrupt to disk controller (INT1-).	
	4. Is CUE high? If not, wait until it is high.
	5. Is COMMAND high? If not, go to 4.
	6. Set BUSY bit to a one.
	7. Set ACCEPT bit to one.
8. Is ACCEPT bit a one? If not, wait.	
9. If yes, set CUE and COMMAND to zero.	
	10. Is CUE a zero? If not, wait.
	11. If yes, is COMMAND a zero? If not, go to 10.
	12. Set ACCEPT to zero.
13. Is ACCEPT a zero? If not, wait.	

NOTE:

Default Address Command (12_{16}) must have been executed to set up default address before using this sequence.

USING DEFAULT ADDRESS FOR COMMAND LIST ADDRESS

FIGURE 3-4 (B). COMMUNICATION BETWEEN THE HOST AND DISK CONTROLLER TO STORE COMMAND LIST ADDRESS THROUGH THE CRU

```
* EQUATE MNEMONICS
COMND   EQU  8       DISPLACEMENT ON CRU FOR COMMAND BIT
CUE     EQU  10      DISPLACEMENT ON CRU FOR CUE BIT
ACCEPT  EQU  11      DISPLACEMENT ON CRU FOR ACCEPT BIT
BUSY    EQU  12      DISPLACEMENT ON CRU FOR BUSY BIT
* BYTE STORAGE FOR COMMAND LIST ADDRESS
ADDR    BYTE >00,>FE,>00  3 ADDRESS BYTES (COMMAND LIST ADDRESS)
* LOAD CRU SOFTWARE BASE ADDRESS IN REGISTER 12
        LI   R12,>210
* SET UP COUNTER TO COUNT THREE BYTES
        LI   R1,3
* BUSY & ACCEPT MUST BE ZEROES BEFORE CONTINUING
ACCEP1  TB   ACCEPT  ACCEPT = ZERO?
        JEQ  ACCEP1  NO, LOOP UNTIL ZERO
BUSY1   TB   BUSY    BUSY = ZERO?
        JEQ  BUSY1   NO, LOOP UNTIL ZERO
* LOAD ADDRESS OF THREE BYTES OF COMMAND LIST ADDRESS
        LI   R2,ADDR
**
** ROUTINE TO SEND THREE ADDRESS BYTES
** OF COMMAND LIST THROUGH CRU
**
* FOR FIRST BYTE, SET COMMAND BIT TO 1 (MEANS 1ST BYTE BEING SENT)
        SBO  COMND
LDBYTE  LDCR *R2+,8  ADDR BYTE TO CRU
        SBO  CUE     CAUSE INTERRUPT TO DISK CONTROLLER
ACCEP2  TB   ACCEPT  DISK ACKNOWLEDGES BYTE RECEIVED????
        JNE  ACCEP2  NO, LOOP UNTIL CONTROLLER SETS ACCEPT TO ONE
        SBZ  CUE     YES, ACKNOWLEDGE CONTROLLER SETTING ACCEPT BIT
        SBZ  COMND   FIRST BYTE SENT, COMMAND = 0 LAST 2 BYTES
ACCEP3  TB   ACCEPT  CONTROLLER RETURNS ACCEPT BIT TO ZERO???
        JEQ  ACCEP3  NO, LOOP UNTIL CONTROLLER SETS IT TO ZERO
        DEC  R1      YES, THIRD BYTE SENT????  (R1 EQUALS ZERO?)
        JNE  LDBYTE  NO, LOOP, LOAD ANOTHER BYTE ON CRU
        .            YES, CONTINUE
        .
        .
```

FIGURE 3-5. PROGRAM TO PASS A COMMAND LIST ADDRESS OF >0FE00

3.3.1 Output to Disk Controller Over CRU (See Figure 3-2)

3.3.1.1 Command List Address Byte (Bits 0-7)

This is one of the bytes of the address in host memory of the Command List. The Command List must be located in host memory on an even byte boundary (LSB of the address a 0).

3.3.1.2 COMMAND Bit (Bit 8)

When the Command List address is to be transferred, set the COMMAND bit to a logical one for transfer of the first byte of the Command List address, and set it to a logical zero for the second and third bytes of the Command List address. The COMMAND bit is valid only when the CUE bit (bit 10) is a logical one. COMMAND is also used in handshaking where a default Command List address is used.

3.3.1.3 CUE Bit (Bit 10)

Causes an interrupt to INT1- of the disk controller TMS 9900 microprocessor to initialize CRU data transfer. Checked as CRU bit during transfer of bytes two and three of Command List address over CRU. CUE is only used as an interrupt for the first Command List address byte transfer. When a default Command List address is used, CUE is used to perform the handshaking necessary to start command execution.

3.3.1.4 INTERRUPT ENABLE Bit (Bit 13)

Set this bit to a logical one to permit the disk controller hardware to issue interrupts to the host as directed by the Command List. Set this bit to a logical zero to clear the INTERRUPT ISSUED bit (bit 15). During interrupt driven operation, this bit is normally set to a one to enable interrupts, then set to a zero to clear the INTERRUPT ISSUED bit (bit 15), then back to a one to re-enable interrupts.

Considerations:

- interrupts must be enabled at both the host interrupt controller (e.g, TMS 9901) and microprocessor,
- the Command List must specify (word 2) that an interrupt is wanted at command completion, and
- the controller must be jumpered at J3 for the correct interrupt level.

3.3.1.5 RESET Disk Controller Bit (Bit 14)

Set this bit to a logical one to cause an unconditional reset of the disk controller (same as a powerup reset but not the result of the RESET switch toggled). This could be used to recover from a software "lockup" of the controller. After resetting the controller, set (toggle) RESET to a zero to allow normal operation to resume. RESET causes execution of the following:

1. Disables interrupts.
2. Turns off the Write gate.
3. Unloads head from disk surface.
4. Sets BUSY bit on CRU interface (bit 12) to logical one.
5. Sets up to receive CUE interrupt at CRU.
6. Initializes workspace registers for timer routine.

7. Clears status flags.
8. Indicates track position unknown.
9. Performs a self-test which includes tests of: controller ROM (checksum), controller RAM, TMS 9901 timer and interrupt handler, disk drive interface, and CRC logic.
10. Turns on LEDs to show self-test results. If the controller fails the test, it halts the self-test and displays the error code on the LEDs (LED interpretation is shown in Table 2-2).
11. Sets up CRU interface to receive commands.
12. Enables interrupts on controller.
13. Sets BUSY bit on CRU interface (bit 12) to logical zero (i.e., ready to receive command via CRU).
14. Enters an idle loop, waits for an interrupt from the CRU "CUE" bit.

CAUTION

This CRU bit (RESET) is tied to the controller TMS 9900 RESET- line which is interrupt level 0. Toggling this bit halts any operation in progress. This bit should only be used if the controller becomes "locked up" (i.e., the controller does not complete the command issued in a specified amount of time.) The controller cannot mask out this interrupt.

3.3.2 Input From Disk Controller Over CRU (see Figure 3-2)

3.3.2.1 ACCEPT Bit (Bit 11)

A logical one sensed at this bit indicates that the disk controller has recognized the enabled CUE bit and has read the COMMAND bit and address bits. The ACCEPT bit can be set to a one only if the CUE bit is a one, and can be set to a zero only if the CUE bit is a zero. This means that the ACCEPT bit can change state only if the CUE bit has already been changed to the same logical state.

3.3.2.2 BUSY Bit (Bit 12)

A logical one sensed at this bit indicates that the disk controller is currently executing a command and is unable to accept a new command. This bit will be a zero when the disk controller is not executing a command and is awaiting further command input. The BUSY bit should not be tested to determine if the disk controller has completed a command (because the BUSY bit remains on for all commands, or because another module may be using the controller after current command completion). Instead, check the OPERATION COMPLETE bit in the first word of the command's Command List for this status. The BUSY bit should be a zero before addressing the CRU to transfer the Command List address.

3.3.2.3 INTERRUPT ISSUED Bit (Bit 15)

A logical one sensed at this bit indicates that the disk controller has issued an interrupt. This bit is cleared by writing a zero to the INTERRUPT ENABLE bit (bit 13). Clearing this bit and re-enabling the interrupt should be part of the interrupt service routine (see section 3.3.1.4).

3.4 COMMUNICATION THROUGH MEMORY (COMMAND LIST)

The Command List is another means of communication between the disk controller and the host microcomputer. This list is a ten-word block, shown in Figure 3-6, of system memory that is accessed by the host directly and by the disk controller via direct memory access. The address of this block is given to the disk controller:

- via the CRU as explained in section 3.3, or
- via the last two words of the presently executing Command List ("chaining"), or
- the controller may already have the Command List address from the Default Command List Address command (Command 12_{16}).

CAUTION

Do not place the Command List in ROM. It is important that the disk controller write back to the Command List showing errors, command completion, etc. The Command List must be in RAM or the user will not be able to monitor command completion.

A summary of the Command List (Figure 3-6) is as follows:

Word 0: Disk Controller Primary Status. This contains three data bits designating that the disk controller has completed its operation (OC, bit 0), or that at least one of several errors occurred (ER, bit 1), or that the controller issued an interrupt upon completion (IO, bit 2). There are also four bits explaining status of the errors incurred by the disk unit and a bit indicating the error is defined in word 1. (Details of Word 0 are covered in section 3.4.1.)

Word 1: Disk Controller Secondary Status. This contains 14 bits indicating disk status and disk-type data (e.g., number of sides, diskette size, diskette format) from the disk drive as well as errors incurred by the disk unit. When an error is reported in this word, the unit error bit (bit 15) in Word 0 is set along with the ER bit. (Details of Word 1 are covered in section 3.4.2.)

NOTE

Word 1 drive status bits (8 to 15) do not represent the format specified by the Define Format command (command 10_{16}). Instead, these bit values are the values read via hardware on connector P4 (from the disk drive) and at jumpers J8 and J9.

Word 2: Command, Flags, and Drive No.: This word contains an eight-bit code for a command to the disk controller, two bits identifying which disk drive is to answer the command, and a flag field specifying additional command data to the disk controller. (Section 3.4.3.)

Words 3 & 4: These two words contain the storage address of the diskette data addressed or the controller RAM address, depending on the command. (Section 3.4.4.)

Word 5: This word contains the number of bytes to be transferred. This will be an even number with bit 15 forced to 0. (Section 3.4.5)

Words 6 & 7: These two words contain the 19-bit memory address of (1) the data to be transferrred to disk or (2) the location of a list which is used by some commands. (Section 3.4.6.)

Words 8 & 9: These two words contain the 19-bit memory address of the next Command Chain address. (Section 3.4.7.)

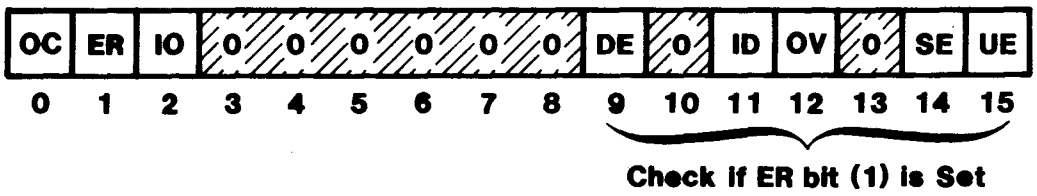
NOTE

Because the disk controller writes into the Command List to indicate the status of command completion, all bits of the Command List should be initialized by the host to a proper value (i.e., set Word 0 and 1 to zeroes).

CAUTION

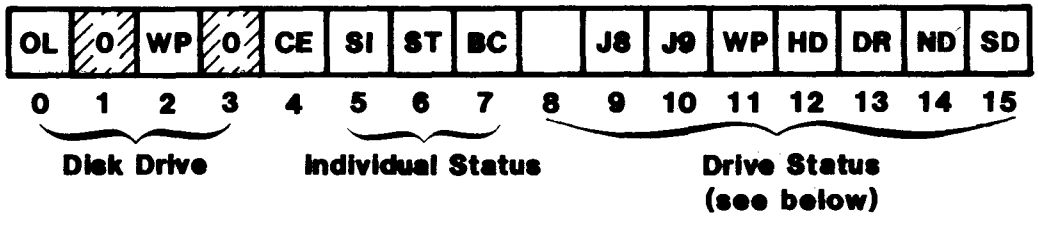
All the words of the Command List must be initialized to a proper value (depending on drive format), even though some commands do not use all words of the Command List. For example, when using IBM format and physical storage mode, the allowable sector range is from 1 to 26 (i.e., do not specify a sector 0). If an invalid value is detected in the Command List (even if not needed by the command issued), a Bad Command error is returned.

Word 0: Primary Status



OC = Operation Complete
ER = Error Occurred
IO = Interrupt Occurred
DE = Data Error
ID = Disk ID Error
OV = Overrun Error
SE = Data Field Search Error
UE = Unit Error

Word 1: Secondary Status



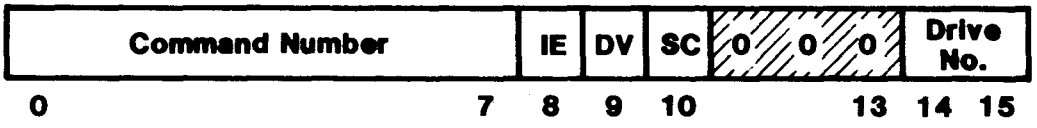
OL = Off Line
WP = Write Protect
CE = Diskette Status Change Error
SI = Seek Incomplete
ST = Self Test Error
BC = Bad Command

Drive Status Bits (reflect J8, J9 settings and inputs from disk drive via connector P4):

Bit 8: Spare (P4-24)
Bit 9: DF=Diskette Format; 0=TI Double, 1=IBM (J8)
Bit 10: DS=Diskette Size; 0=Mini, 1=Standard (J9)
Bit 11: WP=Diskette Write Protected?; 0=No, 1=Yes (P4-44)
Bit 12: HD=Head at Track 00?; 0=No, 1=Yes (P4-42)
*Bit 13: DR=Drive Ready?; 0=No, 1=Yes (P4-22)
*Bit 14: ND=New Diskette Inserted?; 0=No, 1=Yes (P4-12)
Bit 15: SD=Sides on Diskette; 0=1 side, 1=2 sides (P4-10)

* For mini disk drives, bits 13 (DRIVE READY), 14 (NEW DISKETTE), and 15 (TWOSIDED DISKETTE) are not used; these signals are not provided via connector P4 for mini drives (only standard drives).

Word 2: Command, Flags, and Drive Number



IE = Interrupt Enable Flag
DV = Data Verify Flag
SC = Sense Disk Change

FIGURE 3-6. TEN-WORD COMMAND LIST (Page 1 of 2)

Word 3: Storage Address (Most Significant Word or Track Number)

Bit 0: 0 = Mass storage mode, 1 = Physical storage mode

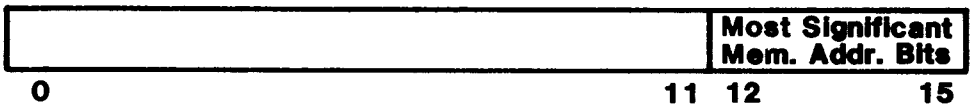
Word 4: Storage Address (Least Significant Word)

Word 5: Byte Count (MUST be an even number)

Word 6: Memory Address (Most Significant Word)

Word 7: Memory Address (Least Significant Word)

Word 8: Next Command Chain Address (Most Significant Word)



Bit 0: 1 = Chaining, 0 = No chaining

Word 9: Next Command Chain Address (Least Significant Word)

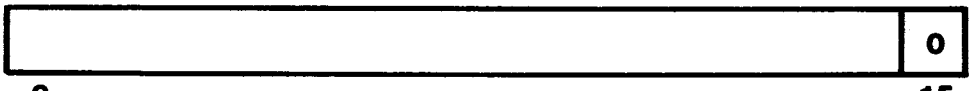
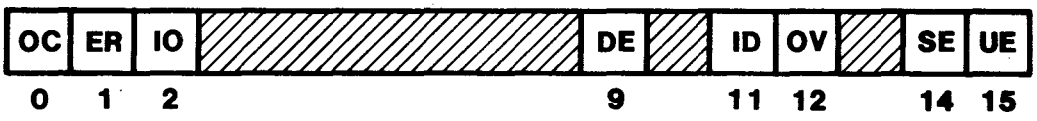


FIGURE 3-6. TEN-WORD COMMAND LIST (Page 2 of 2)

3.4.1 Word 0, Primary Status and Error Indicator Word



OC = Operation Complete
ER = Error Occurred
IO = Interrupt Occurred
DE = Data Error
ID = Disk ID Error
OV = Overrun Error
SE = Data Field Search Error
UE = Unit Error

= SET TO ZEROES, RESERVED

3.4.1.1 Word 0, Bit 0, Operation Complete (OC)

This bit is set when the command (in word 2) has been completed successfully or has been terminated as the result of an error (error causes are decoded by bits in words 0 and 1). Initialize this bit before the command is sent, and then monitor this bit to determine command completion (rather than the BUSY bit which remains on for all commands).

3.4.1.2 Word 0, Bit 1, Error Occurred (ER)

This bit is set when a command is terminated because of an error. Cause of the error is indicated by the error bits in words 0 (bits 9, 11, 12, and 14) and 1 (bits 4, 5, 6, or 7). Since error indicators are in both words 0 and 1, monitor bit 15 of word 0 (UE); if a one, the enabled error indicator is in word 1; if a zero, the enabled indicator is in word 0.

3.4.1.3 Word 0, Bit 2, Interrupt Occurred (IO)

When set, the disk controller had issued an interrupt to the host upon command completion. The interrupt enable bit in the CRU (CRU bit 13) must be set to a one and the Interrupt Enable (IE) flag in List Word 2 must be set to enable interrupts. After an interrupt occurs, interrupts must be cleared and re-enabled by setting the CRU Interrupt Enable bit to one, then zero, and then to one. The interrupt level is jumper selectable at J3 as explained in Table 2-1.

NOTE

The following bits in word 0 (bits 9, 11, 12, and 14) and bits 0, 2, 4, 5, 6 and 7 in word 1 explain errors. The ER bit (bit 1) will be set and the controller operation will terminate if one of these error indicators is set. Bit 15 (UE bit) of word 0 indicates if the set error bit is in word 0 (bit 15 a 0) or word 1 (bit 15 a 1).

3.4.1.4 Word 0, Bit 9, Data Error (DE)

This bit is set when an error occurs during the reading of the data field when executing a Read Data command (command 03) or when executing a Read Deleted Data or Format Track command (command 0A or 09) or during the reading of the ID field when executing a Read ID command (command 0C). A data error occurs when the calculated cyclic redundancy check (CRC) word does not match the precalculated CRC value written on the disk for the respective field. Before a command is terminated because of this error, four attempts to correctly read the data are made. This error also sets the ER bit (word 0, bit 1) and terminates the command. This bit and bit 11 (ID error) will be set if a deleted address mark is read during a Read Data command or if a normal address mark is read during a Read Deleted Data command.

3.4.1.5 Word 0, Bit 11, Disk ID Error (ID)

This bit is set when an unsuccessful search is made by the disk controller for the sector ID in the header area of each sector. Five tracks will be searched for this header, and four tries will be made at each track. This error sets the ER bit (word 0, bit 1) and terminates the command. This bit and the Data Error bit (bit 9) will be set if a deleted data address mark is read during a Read Data command or a normal address mark is read during a Read Deleted Data command.

3.4.1.6 Word 0, Bit 12, Overrun Error (OV)

This bit will be set if the DMA interface cannot transfer data at the rate required by the disk controller. This error will occur during execution of the Read Data or Write Data commands (specified in Word 2). This error also sets the ER bit (word 0, bit 1) and terminates the command.

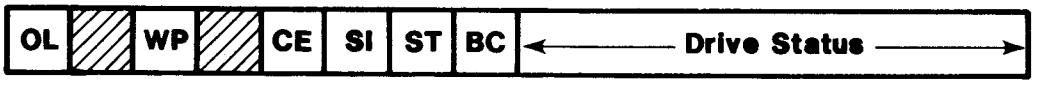
3.4.1.7 Word 0, Bit 14, Search Error (SE)

This bit is set when the disk controller fails to read a data field in 3 to 5 milliseconds after the track and sector ID field has been read (could be bad track format --- check jumper settings on controller and disk drive: they should match the desired drive format). Up to four retries will be made; if the fourth retry fails, the ER bit (word 0, bit 1) is also set and the command is terminated. This error can occur upon an unsuccessful Format Track command attempt.

3.4.1.8 Word 0, Bit 15, Unit Error (UE)

This bit is a logical OR of the error indicators in word 1 (i.e., if any of the errors indicated in word 1 are active, this indicator is set). This error indicator bit is set along with the ER bit (word 0, bit 1), which allows the host to more quickly determine the error type. If the ER bit is set, a check of the UE bit will indicate whether to scan word 1 for the error (UE bit set) or to scan word 0 for the error (UE bit reset).

3.4.2 Word 1, Secondary Status and Error Indicator Word



OL = Off Line Error
WP = Write Protect Error
CE = Diskette Status Change Error
SI = Seek Incomplete
ST = Self Test Error
BC = Bad Command

= SET TO ZEROES, RESERVED

3.4.2.1 Word 1, Bit 0, Unit Off Line Status (OL)

This bit indicates that the DRIVEREADY- signal is not active (drive not ready during an attempted disk operation). This signal becomes active (drive is ready) by the diskette being placed in a drive and both of the following occurring:

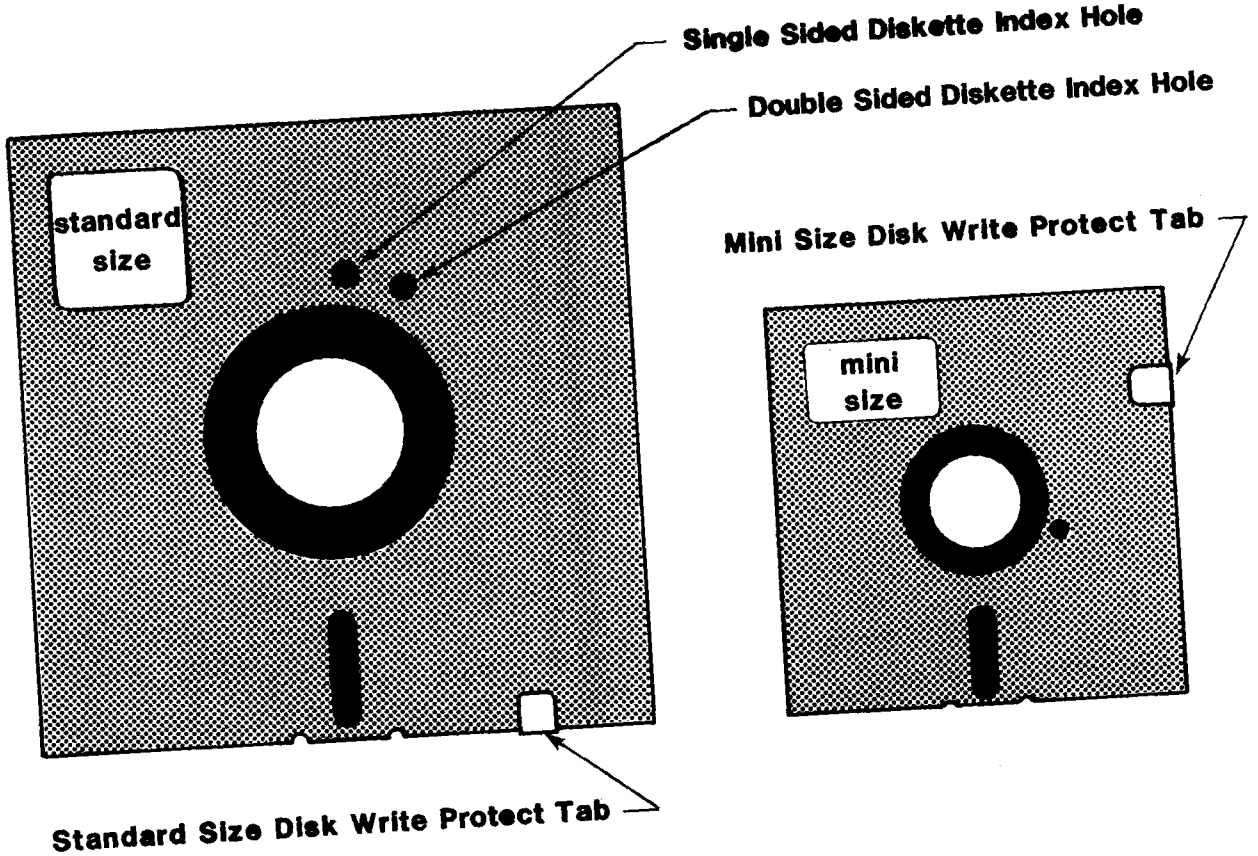
- power is applied to the drive, and
- passage of two index holes has been sensed by the drive.

If a double-sided diskette is inserted in a single-sided drive, because of the index hole placement on the diskette, the drive will never see two index holes and DRIVEREADY- will never be activated. The DRIVEREADY- line is disabled whenever the power is cycled, the door is opened, or the diskette removed, and consequently the OL bit is set. This error also sets two error bits (ER bit in word 0, bit 1, and the UE error in word 0, bit 15) and terminates the command. This error will only occur during disk operation commands.

3.4.2.2 Word 1, Bit 2, Write Protect Status (WP)

This status bit will be set when an attempt is made to write to a diskette that has been write-protected. Data cannot be written to a write-protected diskette. Figure 3-7 shows the write-protect tab locations. A diskette is write-protected when:

- the write-protect tab is removed from a standard-size diskette;
- the write-protect tab is installed on a mini-size diskette.



NOTES

1. Only 1 index hole will be on each diskette, but both types are shown above.
2. For standard drives, remove tab for write protect.
   For mini drives, attach tab for write protect.

FIGURE 3-7. WRITE PROTECT TAB ON DISKETTE

3.4.2.3 Word 1, Bit 4, Diskette Status Change Error (CE)

This error can be checked only on 8-inch drives that issue a DISKCHANGE- (diskette has been changed in drive since last disk access) signal to the controller via pin P4-12. This error condition can be checked by setting bit 10 of Word 2. DISKCHANGE- is enabled and latched when the disk is removed from the selected drive. The signal is reset to the inactive state by the rising edge of the next DRIVESELECT- signal after the disk change event occurred. A check is made for an error (1) between accesses to tracks in a multiple-track read or write and (2) right after drive select is enabled. This check takes place only (1) on 8-inch drives, (2) if the Sense Disk Change Flag (Word 2, bit 10) is set, and (3) if DRIVEREADY- is active, and (4) before executing a command that entails a disk access.

3.4.2.4 Word 1, Bit 5, Seek Incomplete Error (SI)

This bit will be set after failure to sense a signal stating that the disk access arm has been positioned over track 00 after completion of a Restore command. To see if this error can be overcome, issue another Restore command; if this second seek is successful, the SI bit will be reset. This error also sets the ER bit in word 0, bit 1, and the UE error bit in word 0, bit 15, and terminates the command.

3.4.2.5 Word 1, Bit 6, Self-Test Error (ST)

This bit is set when an error occurs during the running of the controller self test, a diagnostic executed by the Controller Self-Test command (command code 01). This error also sets two error bits (ER bit in word 0, bit 1, and the UE error in word 0, bit 15) and terminates the command. Self-test errors are indicated by the LEDs; LED interpretation is summarized in Table 2-2. During a powerup reset or a CRU reset, a Self-Test Error bit is not available to indicate an error condition. Use the Self-Test command to determine status after a Self-Test by checking LED error interpretation.

3.4.2.6 Word 1, Bit 7, Bad Command Error (BC)

This bit is set when an attempt is made to execute an invalid command (code not recognized) or execute a command with an invalid disk storage address. Examples are:

- An illegal command number is specified in Word 2 of the Command List.
- Diskette mass storage address is not on a sector boundary (mass storage address is explained in section 3.4.4.1).
- The mass storage address (on the diskette) is too large.
- An illegal disk is defined in the parameter table such as single density TI.
- An attempt is made to access the second side of a single-sided diskette.
- An attempt is made to access a side three. Floppy diskettes can have at most two sides per drive.
- A read or write is attempted with a byte count of 0.

- An attempt is made to write to controller RAM with an address of less than $FC06_{16}$.

- The physical storage address specifies sector 0 when using IBM format, which has its sector numbers starting at 1.

- An attempt is made to access a nonexistent track (during such commands as Format, Read, Write, Seek, Read ID, Read Deleted, Write Deleted, Read Unformatted, etc).

The disk storage address is contained in words 3 and 4 of the Command List and is computed as explained in section 3.4.4. This error also sets two error bits (ER bit in word 0, bit 1, and the UE error in word 0, bit 15) and terminates the command.

3.4.2.7 Word 1, Bits 8 to 15, Drive Status

Bits 8 to 15 indicate specifications and status of the disk drive and diskette as seen by the disk controller. Some reflect the position of jumper-selectable options on the disk controller module. These bits indicate the following:

Bit 8: Spare Input (shows logic state of pin P4-24 from disk drive).

Bit 9: Jumper J8 Status (Diskette Format): 0 = jumper out, 1 = jumper in

Bit 10: Jumper J9 Status (Diskette Format): 0 = jumper out, 1 = jumper in
(Bits 9 & 10 specify different formats as listed in part 6 of Table 2-1.)

Bit 11: Diskette Write Protected: 0 = Not write protected
1 = Write protected

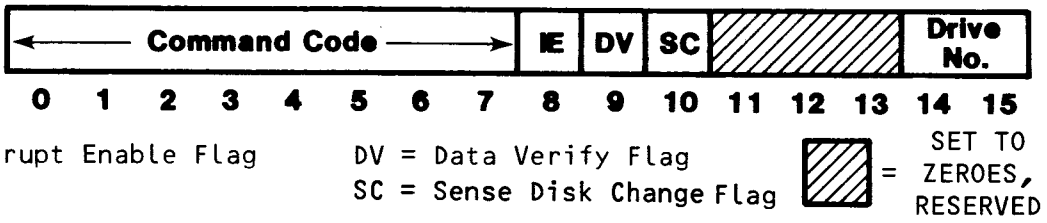
Bit 12: Head at Track 00: 0 = Not at track 00
1 = At track 00

Bit 13: Drive Ready: 0 = Drive not ready
1 = Drive ready

Bit 14: Diskette Changed: 0 = Same diskette (no change)
1 = Diskette changed since last disk access

Bit 15: Number of Diskette Sides: 0 = Single sided
1 = Double sided

3.4.3 Word 2, Commands, Flags, and Drive No.



3.4.3.1 Word 2, Bits 0 to 7, Command Code

These bits contain the command to be executed by the disk controller. These command bytes are summarized in Table 3-1 and explained in detail in Table 3-2.

3.4.3.2 Word 2, Bit 8, Interrupt Enable Flag

When set to one, the disk controller will issue an interrupt to the host when the command is either successfully or unsuccessfully completed. This is the preferred method of command completion since it allows the host to determine corrective action should the command be terminated without successful completion. Interrupts are covered more in detail in section 3.5.

3.4.3.3 Word 2, Bit 9, Data Verify Flag

The Data Verify flag pertains only to read and write commands. If set to one during a read command, data is read again from the diskette and compared to the data stored in host memory during the first read. If the bit is set during a write command, the controller performs a read-after-write and verifies the data written to diskette. A comparison error will set the Data Error flag (bit 9, word 0).

3.4.3.4 Word 2, Bit 10, Sense Disk Change Flag

This flag enables the controller to check to see if the diskette has been changed since the last access to that diskette/drive. This option is useful in systems where multiple accesses are made, and these accesses need to be made to the same diskette. If this flag is set and the controller, through the DISKCHANGE- line (P4-12), senses that a new diskette has been inserted, it will set the CE (Diskette Status Change Error, Word 1, bit 4) error bit, the UE error bit, the ER error bit, and terminate the command.

3.4.3.5 Word 2, Bits 14 & 15, Drive ID

These bits contain the binary ID number of drive units 0 to 3, indicating the drive unit to be acted on by the Command List. Connections to the disk drives of signals DSELECT1- to DSELECT4- from the disk controller select the specified drive. A jumper at the disk drive must correspond to the drive ID as follows: ID 00 (binary) corresponds to the drive jumpered DS1, ID 01 to DS2, ID 10 to DS3, and ID 11 to DS4.

TABLE 3-1. SUMMARY OF COMMANDS TO DISK CONTROLLER

Command (Hex)	Synopsis
00	Store Status Command. Stores the Drive Parameter List status into a specified host memory location.
01	Controller Self-Test Command. Executes disk controller self-test.
02	Reset Command. Resets the controller.
03	Read Data Command. Reads data from diskette to host memory.
04	Write Data Command. Writes data to diskette from host memory.
05	Bootstrap Load Command. Causes controller to execute a bootstrap load.
06	Read Controller Memory Command. Reads data from controller memory; writes it to host RAM.
07	Write Controller Memory. Writes from host memory to controller RAM.
08	Execute Controller Memory. Causes controller to branch to a controller memory location as if it was the start of a two-word vector address of a BLWP instruction. The controller then begins executing the code.
09	Format Track Command. Causes one track of a diskette to be formatted according to specified or default format parameters.
0A	Read Deleted Data Command. Read data from a deleted sector.
0B	Write Deleted Data Command. Writes data to sector(s) and designates sector(s) as deleted sector(s).
0C	Read ID Command. Read ID field at specified track and sector.
0D	Read Unformatted Command. Starting at a specified track and sector, read specified number of bytes without respect to data and control fields.
0E	Seek Command. Places read/write head at specified physical track of diskette.
0F	Restore Arm Command. Places read/write head at track 00.
10	Define Drive Command. Specifies characteristics of the diskette (e.g., mini or standard size, number of surfaces, disk density), and disk drive (e.g., head step time, step settling time, head load and unload time), and format (e.g, number of tracks, sync type, sector interlace factor, bytes per sector, sectors per track).
11	Store Drive Status Command. Stores status of disk drive (read from interface P4) in host memory.
12	Set Default Address for Command List. Sets up a default address in host memory where the command list is stored and shortens the CRU handshaking interface. Saves CRU access time.

TABLE 3-2. COMMANDS TO DISK CONTROLLER IN WORD 2 (page 1 of 14)

Command Code (Hex)	Meaning
00	Store Status Command. A block of disc controller RAM is used to store an eight-word list of parameters describing the diskette and disk drive used. The contents of these eight words is the same as the Drive Parameter List covered in command code 10_{16} and in Figure 3-8. These parameters are default values or as specified (changed) by the Define Disk Drive command (10_{16}). The Store Status command reads these parameters into host memory at the host address specified in words 6 and 7 of the command list. Command List words 3, 4, and 5 must be zeroes. The parameter list in controller RAM can be changed by the Define Drive command. The controller maintains four separate tables, one for each drive. Jumpers J8 and J9 define the default diskette format which is copied from EPROM to the four RAM table areas at powerup.
01	Controller Self-Test Command. This causes the controller self-test to begin. This test is contained in controller firmware and consists of: (1) ROM checksum test (2) RAM walking ones/zeroes (3) RAM transition test (4) TMS 9901 timer verification (5) TMS 9901 interrupt test (6) Disk drive interface & CRC test An error will cause the Self Test error bit (ST, bit 6 in word 1) and the Unit Error bit (UE, bit 15 in word 0) to be set. The LEDs are interpreted to determine specific error conditions. LEDs and test details are listed in the notes to Table 2-2. Command List words 3, 4, and 5 must be zeroes.
02	Reset Command. The Reset command performs the same steps as for a reset request through the floppy disk controller CRU interface (bit 14 as explained in section 3.3.1.5) except that a bootstrap load is not performed even if this operation is jumpered. Command List words 3, 4, and 5 must be zeroes. The Reset command reloads the four drive parameter tables with the format specified by J8, J9, and the TWOSIDED- signal (P4-10), sets up the Timer Workspaces, clears all flags, and indicates track position as unknown. Naturally, this command cannot bring the controller out of a "locked up" condition as commands cannot be sent to the controller when this condition exists. Recovery would be via the CRU RESET bit (3.3.1.5).
03	Read Data Command. (See CAUTION at the end of this command that applies to both read and write data commands.) The read data command reads the specified number of bytes (in Command List Word 5) from the disk storage address (list Words 3 and 4) to the host memory address (list words 6 and 7). Data will be read beginning at sector boundaries only: however, the number of bytes to transfer can include a partial sector (byte count does not have to be a complete sector). Naturally, the controller will not support reads from more than one disk in a single read operation. All reads will be on a word (16 bits) basis; thus, the byte count in word 5 of the Command List must be even.

TABLE 3-2. COMMANDS TO DISK CONTROLLER IN WORD 2 (page 2 of 14)

Command Code (Hex)	Meaning
	NOTES 1. Partial sector transfers are supported for both read data command and write data command. 2. When a partial sector read is specified, the entire sector is read to calculate the cyclical redundancy check (CRC), but only the partial amount specified is transferred. Multisector reads can be specified with a partial read; in such a case, the last sector read would be a partial sector read. 3. During a read or write, the byte count in word 5 is decremented to determine the transfer count. If more bytes remain to be transferred when the end of a track is read, the controller will switch transfer to the first sector on the next logical track (or next surface or next cylinder in two-sided diskettes) and resume the read or write operation. The head is not returned to track 00 after read/write command completion. After this command is read, the host can later check for unit errors returned in case the disk drive was off line (OL error, word 1, bit 0) or track 00 status was not found (SI error, word 1, bit 5), etc. Number of bytes transferred will be returned in word 5. When this command is executed, a seek is executed to the specified track. A read of sector IDs is performed until the correct starting sector is read. If the sector ID is not found, four read retries will be made. Failure to read the sector ID after the fourth retry will terminate the command and set the ID error bit (word 0, bit 11). After finding the correct sector ID, the data from that sector is read on a word-by-word basis. As each word is read, the transfer count from word 5 is decremented. A CRC check is made after each sector is read. If a CRC error is encountered, four retries will be made. Failure to read data with no CRC error after the fourth retry will cause the Data Error (DE, word 0, bit 9) to be set and the command will be terminated. **CAUTION** Do not read or write across 64 K address boundaries (this applies to extended addressing only). For example, to write to host memory address $FE00_{16}$ to $101FE_{16}$, do two writes (or reads) -- one to $FE00_{16}$ to $FFFE_{16}$ and a second to 10000_{16} to $101FE_{16}$. If a read or write operation is attempted that crosses a 64 K boundary, a wraparound will occur which will write over the beginning memory locations of the same page. The map register (word six in the Command List) has to be explicitly changed to reference a different 64 K memory page.

TABLE 3-2. COMMANDS TO DISK CONTROLLER IN WORD 2 (page 3 of 14)

Command Code (Hex)	Meaning
04	Write Data Command. (See CAUTION for command 3.) This command writes the specified number of bytes (Command List word 5) from the host memory address (list words 6 and 7) to the disk storage address (list words 3 and 4). Data will be written to the disk address beginning at a sector boundary, and data is transferred on a word (16 bits) basis. After this command is read, the host can check for unit errors returned; e.g., in case the disk drive was off-line (OL, word 1, bit 0) or track 00 status was not found (SI, word 1, bit 5). The number of bytes transferred will be returned in word 5. When this command is executed, a seek is executed to the specified track (cylinder for two-sided disks). A read of sector IDs is performed until the correct starting sector is read. If the sector ID is not found, three retries to read it will be made. Failure to read the sector ID after four tries will result in command termination and the ID error (word 0, bit 11) being set. Partial-sector writes are supported; this applies to a single- or multiple-sector write. WHEN THE TRANSFER DOES NOT END ON A SECTOR BOUNDARY (LAST SECTOR WRITTEN TO IS NOT FILLED BY THIS WRITE), THE REMAINDER OF THE SECTOR AFTER THE LAST WORD TRANSFERRED IS FILLED WITH ZEROES. After writing to a sector, the CRC value for the sector is computed and written. If the transfer was to the last sector on a track and more data remains to be transferred, the next transfer will be to the first sector on the next track (or next cylinder or next surface in a two-sided disk). This continues until all the sectors are written to, with CRC values calculated and written for each sector.
05	Bootstrap Load Command. This command causes the controller to read in the first sector of track 0 of the diskette and check it for bootload information. This information will be for one of three diskette formats: • 303 format (command list on diskette) • TX4 format • TM990 AMPLUS (TX5) format After reading the bootload sector, the controller first checks for the TM 990/303B bootload format. This is done by adding an 11 word area which begins at byte offset 56 in the bootload sector. If the sum (called the checksum) of this addition is zero, the controller will attempt a 303B bootload. When using a TX4 or an AMPLUS bootload format, the sum of the 11 word area (at byte offset 56 in the bootload sector) must not result in zero or a 303B bootload will be attempted. If the sector length is 128 bytes per sector, a TX4 bootload is attempted. If the sector length is 288 bytes per sector, an AMPLUS bootload is attempted. If, after a non-zero checksum, the sector length is not 128 or 288 bytes, the controller will report an error. If the controller finds an error during bootload execution, 5 retries will be attempted.

TABLE 3-2. COMMANDS TO DISK CONTROLLER IN WORD 2 (page 4 of 14)

Command Code (Hex)	Meaning
	The host software is responsible for monitoring bootload command completion and for knowing where the bootload program is loaded. Care must be taken so that the bootload program loaded into host memory does not erase the software the host was currently executing. The host program is responsible for transferring control to the bootloaded program upon successful bootload completion. FOR THE 303 FORMAT, this means a seek to the first sector, track 00, surface 0 of unit 0 (designated DS1 by drive jumpers). Ignore the first 56 bytes and interpret the next ten words as a Command List for that unit. The Command List checksum is the eleventh word (two's complement of the ten-word Command List summation) and the sum of the eleven words must equal zero. This Command List read from drive 0 may direct the controller to another drive unit to complete the bootload process; however, the initial Command List must be contained on drive 0. The controller then executes the specified command within the Command List. This bootload structure is compatible with the TM 990 UCSD™ p-System file structure. This command does not observe a chain address, and a chain to a next Command List will not occur. There are two command lists involved with this command: (1) the bootstrap command list originating in host memory and (2) the command list on the diskette that will be executed. Upon command completion, values will be returned to words 0, 1, and 5 of the command list in memory for either command list -- for the command list in memory if the bootload was prevented from executing, or for the command list on the diskette after it was brought into controller memory and executed. Values returned (to words 0, 1, and 5 respectively) include the primary status word (word 0) and data for the secondary status word (word 1) which will contain command status data for the command list, and the byte transfer length in word 5. (A power-up bootstrap load differs in that these three values are returned in words 0, 1, and 5 beginning at the host address specified in words 8 and 9 of the command list on the diskette.) The 303 format does not transfer the workspace pointer and program counter parameters of the bootload program to host memory. The host program is responsible for transferring control to the bootloaded program upon successful bootload completion. THE TX4 FORMAT works with IBM single-density, single-sided formatted diskettes. The controller looks for the bootstrap program in sector 1 of track 0 and reads it, sector-by-sector (maximum of 24 contiguous sectors at 128 bytes per sector), into host memory starting at address $00A0_{16}$. In sectors occupied by the program, the last word of each sector will be the sector number -1 except for the final sector in which the last word of the sector is a -1 ($FFFF_{16}$). When a -1 final sector word is found, reading stops, the controller reads the whole bootstrap program over to the host, and host execution is directed to the start

TABLE 3-2. COMMANDS TO DISK CONTROLLER IN WORD 2 (page 5 of 14)

Command Code (Hex)	Meaning
	of the boot program. See following section pertaining to both TX4 and AMPLUS formats. The bootload program will be loaded to host address $00A0_{16}$. The bootload program workspace will begin immediately following the bootload program. The ten-word status list to be returned by the controller will follow immediately behind this workspace. Error checks include: • boot program is not larger than 24 sectors, • operating system is present. The controller determines if an operating system is present by checking two "flag" words which are contained in the TX4 system table (in sector 20 of track 0). An operating system is present if 1) the word at byte offset 34 contains an ASCII "TX" (5458_{16}) and 2) the word at byte offset 32 contains the pointer where the operating system starts. This pointer should be nonzero. • the last word of each boot sector (except for the final sector) is the sector number. WARNING Texas Instruments provides no system software support for the Bootstrap Load Command (command 05) when used with TX4 format on TM 990 hardware (i.e., "TXBOOT" will not execute). WITH TM990 AMPLUS FORMAT, the first sector of track 0 contains the AMPLUS system's "Volume Information Block." This block contains infomation necessary to perform the bootload as well as information that is used for error checking. Required bootload information is shown below. The word at byte offset 36 is the starting track number of the bootload, and the word at byte offset 26 is the length of the bootload program in bytes. The bootload program begins on the first sector of the bootload starting track. The first two words of the bootload program are respectively (1) the host address at which to start execution and (2) the host address where loading of the program is to begin. This means that the program entry point must be at least two words after the beginning of the program. The workspace area will follow immediately after the loaded program. See the section on the following page pertaining to both TX4 and AMPLUS formats. The information required for the AMPLUS bootload is shown below. Byte Offset — Required Value — Description 12 — 0120_{16} — Number of bytes/sector (TI DD) 14 — User Defined — Bootload starting track. Must be the same as word at byte offset 36. 26 — greater than 0 — Bootload program length in bytes. 36 — User Defined — Bootload starting track. Must be the same as word at byte offset 14.

TABLE 3-2. COMMANDS TO DISK CONTROLLER IN WORD 2 (page 6 of 14)

Command Code (Hex)	Meaning
	Several checks are made on the data in sector 0 of track 0: • word 12 is 120_{16} (number of bytes per sector = 288), • words 14 and 36 are the same (boot starting track number), • word 26 is larger than 0 (bootload length), and • bootload program entry address is larger than the program load address by at least four bytes. FOR TX4 AND AMPLUS FORMATS. After completion of the bootload program transfer to host memory, the controller will attempt to write the workspace pointer and program counter of the bootload program into locations of the host at 0000_{16} and 0002_{16}. If RAM is available at these memory locations, the user may, upon bootload command completion, issue a BLWP @ 0000_{16} to begin execution of the bootload program. If RAM is not available, the user's program must contain the transfer vectors to execute the bootload program (since the transfer vectors from the controller were not retained at 0000_{16} and 0002_{16}). These two formats only specify a 15-bit address in host memory, but the controller will bootload into a 19-bit address system. The controller will check for host RAM (to bootload into) at $0XXXX_{16}$ through $FXXXX_{16}$, starting at $0XXXX_{16}$. FOR ALL FORMATS. This command may also be executed during a powerup reset if the bootstrap load is jumpered. It is very useful in a system which does not have a controller initialization routine in ROM. The first sector of track 00 must already have been set up by a previous disk initialization routine. This command is the same as the optional bootstrap load performed at powerup as explained in section 3.6 except for placement of the returned status words. After a powerup bootstrap load, the two status words and transfer length are placed in host memory in the following manner: • 303 format: as if the address in the chain field was the Command List address. • TX4 and TM990 AMPLUS formats: as if the Command List block was located immediately following the host bootload workspace. For example, the Command List will begin at $E080_{16}$ after reading in a 60_{16}-byte bootstrap program to $E000_{16}$ followed by the 20_{16}-byte workspace. These three words are not placed contiguously in the host memory; instead, the two status words will be placed in words 0 and 1 of this ten-word area; the transfer length will be placed in word 5.

TABLE 3-2. COMMANDS TO DISK CONTROLLER IN WORD 2 (page 7 of 14)

Command Code (Hex)	Meaning
06	Read Controller Memory Command. This command causes the number of bytes specified in list word 5 to be read from controller RAM, starting at the address in list words 3 and 4, and then writes these to the host memory address specified in list words 6 and 7. The byte count must be even. The address limits are $F800_{16}$ to $FFFE_{16}$. NOTE Usually, Command List words 3 and 4 specify a diskette address and must be on sector boundaries. For reading, writing, or executing in controller memory, the address in words 3 and 4 does not have to be on sector boundaries (the TM 990/303A requires words 3 and 4 to be on sector boundaries).
07	Write Controller Memory Command. This command causes the number of bytes specified in list word 5 to be written from host memory to the (RAM only) controller memory space starting at the address specified in list words 3 and 4 (see CAUTION below). Starting address of host memory is specified in list words 6 and 7. All data loaded into the controller RAM must be absolute or position independent since the controller has no relocating capability such as that provided by a relocating loader. See NOTE in command 6. CAUTION The host can write only to specific addresses in controller RAM since writing outside of this area could seriously upset system operation. Controller memory which can be written to is from address $FC06_{16}$ to $FFFE_{16}$. If the address is not within this range, a Bad Command error is returned.
08	Execute Controller Memory Command. This command causes the controller CPU to branch to the memory address specified in list words 3 and 4 as if this address was the two-word vector destination of a BLWP instruction and the new workspace pointer and program counter values were stored at this location. This command can be used in conjunction with the Write Controller Memory command (07) where a code sequence is passed to controller RAM, then executed by the Execute Controller Memory command. The code sequence requires a termination with an RTWP command, the same as a normal BLWP-initiated subroutine. Note the restrictions on controller RAM in the CAUTION above. The address in list words 3 and 4 does not have to be a sector boundary. See NOTE in Command 06.

TABLE 3-2. COMMANDS TO DISK CONTROLLER IN WORD 2 (page 8 of 14)

Command Code (Hex)	Meaning
09	Format Track Command. This command causes one track of a diskette to be formatted according to the format parameters generated by the Define Drive command (command 10_{16}). If parameters have not been re-defined by the Define Drive command, the default format at powerup is as specified at jumpers J8 and J9. After formatting the track, a CRC test is made of each sector on the track. The track address is placed in words 3 and 4 of the Command List. The data pattern to place in the sector data field is specified in a 16-bit word in host memory: this memory location of the data pattern is placed in words 6 and 7 of the Command List. The controller will first check signal lines to determine if the disk unit is in a ready status (not off-line) or if the diskette is not write protected: negative status will be indicated in Command List word 1 (OL or WP bit). Any negative condition will terminate the command. When the arm is correctly positioned, the controller will then proceed to format the diskette track according to the format specified in the Define Drive command (or default format). When a CRC error is encountered, a Data Error will be indicated (DE, word 0, bit 9). It will be the host's responsibility to label the track as bad (e.g., generate table lookup of track status). If the disk drive is jumpered incorrectly such that (for example) the Read/Write heads will not load, an SE error (word 0, bit 14) will be indicated. IBM double density diskettes will automatically be formatted with track 00 in single density and the remaining tracks in double density. Diskette formats are explained in detail in Appendix C.
0A	Read Deleted Data Command. This command is similar to the Read Data command (03) except that this command will read and transfer the data in a sector's data field which has been designated as a deleted sector by the Write Deleted Data command (command 0B). Sectors can be designated as deleted sectors if found to be bad. Attempts to read a deleted sector by the Read Data command (03) will result in both an ID Error and Data Error (ID and DE, word 0, bits 11 and 9). More than one sector can be read. Partial sectoring can be done with this command.
0B	Write Deleted Data Command. This command writes data to sectors as in the Write Data command (04) except that this command also writes a specified code in field AM2 of the control fields of a sector to designate the sector as a deleted sector. The specified code will vary depending upon diskette format, discussed in Appendix C. The write function is initiated the same as the Write Data command (04). This command is normally used to indicate bad tracks. This command allows partial sectoring.

TABLE 3-2. COMMANDS TO DISK CONTROLLER IN WORD 2 (page 9 of 14)

Command Code (Hex)	Meaning
0C	Read ID Command. This command causes the controller to seek to the specified track and sector and read the ID Field, which is four or six bytes that follow address mark 1 (AM1). The ID field specifies the track number, head number, record number, and physical record length as shown for the various diskette formats in Appendix C. The ID field contents are written to the host address specified in words 6 and 7.
0D	Read Unformatted Command. This command reads control data as well as user data beginning at a specified sector. It causes a drive to seek to the specified track, wait for the correct address marker (AM1) of the specified sector, and attempt to synchronize with the next address mark (AM2). Following synchronization, data words will be transferred to memory starting at the disk address specified (words 3 and 4 of the Command List) until the specified number of bytes are transferred. However, data following the sector specified may not be read correctly because of possible loss of synchronization. No cyclical redundancy check (CRC) will be made, and reading will continue without respect to data field boundaries (reading to include data bytes, control fields, etc.). However, data following the sector specified may not be read correctly because of possible loss of synchronization.
0E	Seek Command. This command causes the read head to seek the specified physical track. Prior to initiating the seek, the controller determines disk unit status by checking the DRIVEREADY- (P4-22) line which tells if the diskette is installed, door closed, and at least two index holes sensed. If not ready, the Unit Error bit (UE, bit 15, word 0) and Off Line bit (OL, bit 0, word 1) will be set.
0F	Restore Arm Command. This command causes the read head to return to physical track 00. To determine if the head failed to reach track 00, test the Seek Incomplete error (word 1, bit 5).
10	Define Drive Command. This command is used to modify items in the Drive Parameter Lists, generated in controller RAM, which describe the characteristics of the disk drive and its diskette. At the times listed below, the four Drive Parameter Lists are derived from tables in ROM and from the setting of onboard jumpers J8 and J9 (Table 2-1) and drive signal TWOSIDED-: • at powerup • at powerup bootload • at hardware RESET (via CRU) • at a RESET command.

TABLE 3-2. COMMANDS TO DISK CONTROLLER IN WORD 2 (page 10 of 14)

Command Code (Hex)	Meaning
	There are four Drive Parameter Lists in controller RAM, one for each disk drive (DS0 to DS3). NOTES 1. Diskette size in word 0 cannot be changed; it must be the same as specified in J10 and J11 or improper operation will result. 2. There are four Drive Parameter Lists, one for each possible disk drive (drives 00_2 to 11_2 as specified in bits 14 and 15 of Command List word 2). Changes to the Drive Parameter List by this Define Drive command will prevail over jumpers set on the controller module except as noted above in Note 1. 3. The controller will revert back to the original Drive Parameter List in ROM (ignore previous changes) should any of the four resets or powerups occur (listed on the previous page). Thus, any changes must be reinserted using the Define Drive command. In words 6 and 7 of the Command List, place the address in host memory where the newly generated Drive Parameter List will be found. Command List words 3, 4, and 5 must be set to zero, as the controller provides these values. Definitions of the Drive Parameter List words are: Drive Parameter List Words Word 0: Bits 0 to 3: precompensation bits designate which tracks on the diskette will have data precompensation (further explained in section 4.7.4). These 4 bits are interpreted: - All zeroes: Precompensation as was done by the TM 990/303A (precompensation only on the inner half of an 8-in. double-density disk). - Bit 0 a one: Precompensation on 5¼ or 8 in. disks in single or double density; determine what track to begin precompensation (precompensation will be on this track and all higher-numbered tracks) by the following formula: T - X = track to begin precompensation Where: T = (number of tracks on disk) - 1 X = Value T shifted right by the value in bits 1 to 3. For example, on a 77-track disk (tracks 0-76) with Word-0 bits 0-3 = 1011_2: T = 77-1 = 76 = $4C_{16}$ = 0100 1100_2 X = 0100 1100_2 shifted 3 places to right X = 01001_2 = 9 T - X = 76-9 = 67 (i.e., precompensation on tracks 67 to 76 only)

TABLE 3-2. COMMANDS TO DISK CONTROLLER IN WORD 2 (page 11 of 14)

Command Code (Hex)	Meaning
	Drive Parameter List Words (Continued) Word 0: Bits 4 to 7: Diskette Size (NOTE: These bits must indicate the same size as set at jumpers J10 and J11 or improper operation may result): 0000 = mini size, 0001 = standard size Word 0: Bit 8: Motor-on bit: 1 = motor turned on only when drive selected 0 = motor always on Default is 0. Set to 1 for drives that start motor only when drive selected. Mini drives wait until selected before turning on the motor; thus this bit is automatically set by the controller for mini drives jumpered as such at J10 and J11. When set, controller waits 5 disk revolutions before accessing the disk. Word 0: Bits 9 to 15: Diskette Surfaces: 1 or 2 Word 1: Number of Tracks (or cylinders for 2-sided disks) Word 2: Head Step Time in 10-microsecond units (time to step the disk read head one track distance) Word 3: Step Settling Time in 10-microsecond units (wait time for disk read head to settle at a track) Word 4: Head Load Time in 10-microsecond units (time to lower head to diskette) Word 5: Head Unload Timeout in milliseconds (time to wait after a command is complete before unloading head) Word 6: Bit 2: all tracks have/have not 256-byte data records: 0 = all tracks do not have 256-byte data records (e.g., standard IBM double density with data records comprised of 128 bytes in the first track but 256 bytes in the other tracks). 1 = all tracks have 256-byte data records (e.g., modified IBM double density) Word 6: Bit 3: IBM Density: 0 = single density, 1 = double density Word 6: Bits 4 to 7: Sync Type: 0000 = IBM, 0001 = TI

Command Code (Hex)	Meaning
	Drive Parameter List Words (Continued)

Word 6: Bit 8: Sector Skew: 0 = No sector skew, 1 = Use sector skew

The sector skew feature is provided to compensate for the disk travel while the drive head moves from one track to another, thus minimizing rotational latency time. The tracks on a disk can be thought of as rings concentric to the center of the diskette. The first sector of each track can be thought of as being the sector occurring after the index hole; this is the first physical sector location (see figure). When data in a read or write crosses track boundaries, the drive head must move to the next track, requiring head step and settling times.

Since the last logical sector on a track is physically adjacent to the first logical sector on the next track, after movement to the next track, the head will not be over the first physical sector position on the next track due to the head step and settling times. Thus, another disk revolution must occur before the first logical sector can be read when it occupies the first physical sector position. Depending on the necessary head step and settling times, the sector skew feature will offset the first logical sector from the first physical sector position enough sector positions so that after the head step and settling times, the first logical sector will be the first sector available to be read. For example, when sector skew is specified and the head must travel from the last sector of track 0 to the first sector of track 1, the third physical sector of track 1 may be the first logical sector of track 1. Using sector skew, data is arranged in logical sectors on a track which are offset each time by a value determined by the controller skew program. This program uses the head step and settle times in calculating this sector offset value. This is also shown in the figure below which uses a skew of two physical sectors.

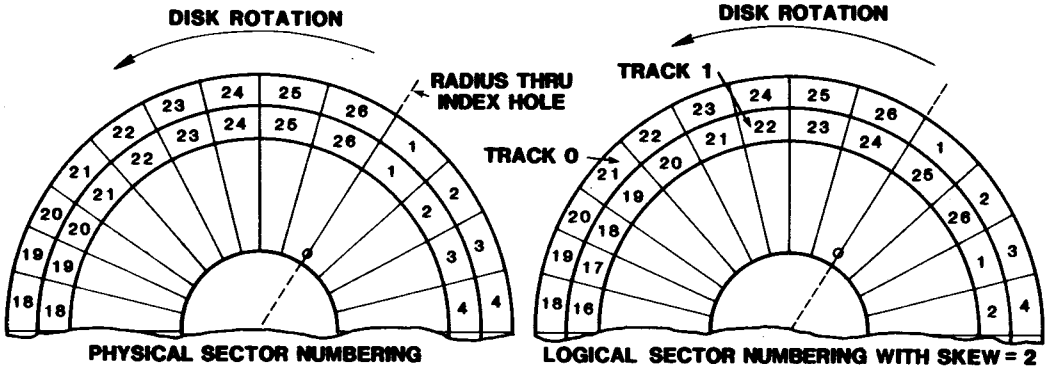
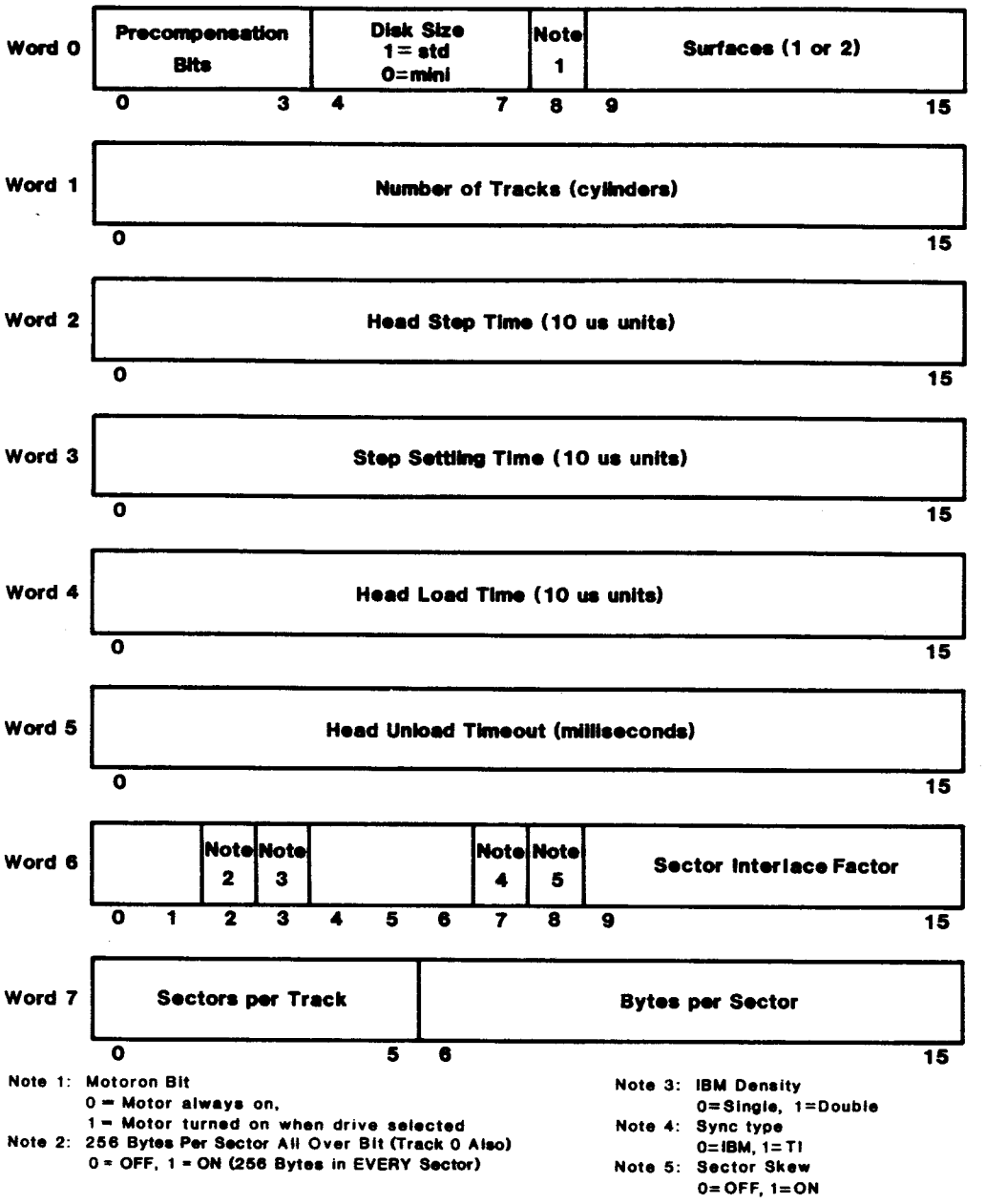


TABLE 3-2. COMMANDS TO DISK CONTROLLER IN WORD 2 (page 13 of 14)

Command Code (Hex)	Meaning
	Word 6: Bits 9 to 15: Sector Interlace Factor: This is the distance, measured in physical sectors, between each contiguous logical sector. Because there is a time factor occurring while the data is being manipulated in a read-sector or write-sector operation, the physical location of contiguous logical sectors can be offset for efficient use of time. The distance between physical locations of contiguous logical sectors allows for normal disk travel during the time needed to manipulate the data written to or read from that sector (e.g., bring it to a buffer or bring it from a buffer). The factor number minus 1 is the number of physical records separating contiguous logical records. Table 3-3 is a list of interlace factors (1 to 25) showing the placement of logical sectors according to physical sector (in the column on the left). Word 7: Bits 0 to 5: Sectors per Track Bits 6 to 15: Bytes per Sector Figure 3-8 shows examples of the Drive Parameter List words and examples of default values that will be brought in from controller ROM depending upon jumpers and a signal from the drive. Appendix A figures show drive parameter values used depending on the diskette format and drive type.
11	Store Drive Status Command. This command allows checking drive parameters at connector P4 as well as jumpers J8 and J9. This status is stored in bits 8 to 15 of word 1 of the Command List such as described in section 3.4.2.7. These bits show the following status: Bit 8: Spare Input (shows logic state of pin P4-24 from disk drive). Bit 9: Jumper J8 Status (Diskette Format): 0 = jumper out, 1 = jumper in Bit 10: Jumper J9 Status (Diskette Format): 0 = jumper out, 1 = jumper in (J8 & J9 specify different formats as listed in Table 2-1.) Bit 11: Diskette Write Protected: 0 = Not write protected 1 = Write protected Bit 12: Head at Track 00: 0 = Not at track 00 1 = At track 00 Bit 13: Drive Ready: 0 = Drive not ready 1 = Drive ready Bit 14: Diskette Changed: 0 = Same diskette (no change) 1 = Diskette changed Bit 15: Number of Diskette Sides: 0 = Single sided 1 = Double sided This status is returned after execution of all commands.

TABLE 3-2. COMMANDS TO DISK CONTROLLER IN WORD 2 (page 14 of 14)

Command Code (Hex)	Meaning
12	Set Default Address for Command List. This command sets up a default address in host memory to get the Command List. To initiate a command to the controller, a 19-bit address is passed over the CRU to the floppy disk controller. This command specifies that each time the floppy disk controller receives an interrupt through the CUE line of the CRU, go to a specified default address set up by this command to get the Command List to execute. The address to be used as a default is contained in list words 6 and 7 of this command. To return to the normal mode of operation, reissue this command with a default address of 0 in words 6 and 7. The CRU interface is shortened to one handshaking of the CRU CUE and ACCEPT lines. The following is the shortened sequence after a Command List has been loaded at the default address: • Wait until ACCEPT and BUSY are zeroes. • Set COMMAND high. • Set CUE high. • Wait for ACCEPT to go high. (BUSY should also be high.) • Set COMMAND low. • Set CUE low. • Wait for ACCEPT to go low. This sequence timing is shown in Figure 4-4 along with the timing where a default address is not used.



Note 1: Motoron Bit
0 = Motor always on,
1 = Motor turned on when drive selected

Note 2: 256 Bytes Per Sector All Over Bit (Track 0 Also)
0 = OFF, 1 = ON (256 Bytes in EVERY Sector)

Note 3: IBM Density
0=Single, 1=Double

Note 4: Sync type
0=IBM, 1=TI

Note 5: Sector Skew
0=OFF, 1=ON

DEFAULT VALUES

J8	J9	Format	Words							
			0	1	2	3	4	5	6	7
In	In	IBM SD*	0101	004D	03E8	05DC	0DAC	03E8	0000	6880
In	Out	IBM DD*	0101	004D	03E8	05DC	0DAC	03E8	1000	6900
Out	In	TI DD*	0101	004D	03E8	05DC	0DAC	03E8	1100	6920
Out	Out	Mini SD**	0001	0023	0FA0	05DC	1388	03E8	0000	4080

* The TWOSIDED- line is checked to determine if the diskette is single or double sided.

** No TWOSIDED- line is present, so the default value is always single sided.

FIGURE 3-8. DRIVE PARAMETER LIST

TABLE 3-3. STANDARD SIZE SECTOR PLACEMENT ACCORDING TO SECTOR INTERLACE FACTOR

Physical Sector Number	Logical Sector According to Interlace Factor*																								
	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
1	1	13	9	13	21	13	15	13	3	13	19	13	2	13	7	13	23	13	11	13	5	13	17	13	25
2	2	1	18	7	16	9	4	10	6	8	12	11	4	2	14	5	20	3	22	4	10	6	8	12	24
3	3	14	1	20	11	22	19	23	9	21	5	24	6	15	21	18	17	16	7	17	15	19	25	25	23
4	4	2	10	1	6	5	8	7	12	3	24	9	8	4	2	10	14	6	18	8	20	12	16	11	22
5	5	15	19	14	1	18	23	20	15	16	17	22	10	17	9	23	11	19	3	21	25	25	7	24	21
6	6	3	2	8	22	1	12	4	18	11	10	7	12	6	16	2	8	9	14	12	4	5	24	10	20
7	7	16	11	21	17	14	1	17	21	24	3	20	14	19	23	15	5	22	25	25	9	18	15	23	19
8	8	4	20	2	12	10	16	1	24	6	22	5	16	8	4	7	2	12	10	3	14	11	6	9	18
9	9	17	3	15	7	23	5	14	1	19	15	18	18	21	11	20	25	25	21	16	19	24	23	22	17
10	10	5	12	9	2	6	20	11	4	1	8	3	20	10	18	12	22	2	6	7	24	4	14	8	16
11	11	18	21	22	23	19	9	24	7	14	1	16	22	23	25	25	19	15	17	20	3	17	5	21	15
12	12	6	4	3	18	2	24	8	10	9	20	1	24	12	6	4	16	5	2	11	8	10	22	7	14
13	13	19	13	16	13	15	13	21	13	22	13	14	1	25	13	17	13	18	13	24	13	23	13	20	13
14	14	7	22	10	8	11	2	5	16	4	6	12	3	1	20	9	10	8	24	2	18	3	4	6	12
15	15	20	5	23	3	24	17	18	19	17	25	25	5	14	1	22	7	21	9	15	23	16	21	19	11
16	16	8	14	4	24	7	6	2	22	12	18	10	7	3	8	1	4	11	20	6	2	9	12	5	10
17	17	21	23	17	19	20	21	15	25	25	11	23	9	16	15	14	1	24	5	19	7	22	3	18	9
18	18	9	6	11	14	3	10	12	2	7	4	8	11	5	22	6	24	1	16	10	12	2	20	4	8
19	19	22	15	24	9	16	25	25	5	20	23	21	13	18	3	19	21	14	1	23	17	15	11	17	7
20	20	10	24	5	4	12	14	9	8	2	16	6	15	7	10	11	18	4	12	1	22	8	2	3	6
21	21	23	7	18	25	25	3	22	11	15	9	19	17	20	17	24	15	17	23	14	1	21	19	16	5
22	22	11	16	12	20	8	18	6	14	10	2	4	19	9	24	3	12	7	8	5	6	1	10	2	4
23	23	24	25	25	15	21	7	19	17	23	21	17	21	22	5	16	9	20	19	18	11	14	1	15	3
24	24	12	8	6	10	4	22	3	20	5	14	2	23	11	12	8	6	10	4	9	16	7	18	1	2
25	25	25	17	19	5	17	11	16	23	18	7	15	25	24	19	21	3	23	15	22	21	20	9	14	1

*NOTES:

1) The above table is for standard sized TI format, To change to standard IBM format, add the value one (1) to each logical sector number in the body of the table. For example, the logical sectors under interlace factor 1 (numbered 0 to 25) would be renumbered with the value 1 added to each (1 to 26).

2) This table does not pertain to mini sized diskettes.

3) Interlace Factor 0 and 1 are the same.

4) Columns under each Interlace Factor number show the physical consecutive (concatenated) postions of the logical sectors starting with sector 0 (top line). For example, looking downward at a diskette and following the sectors counterclockwise, the logical records for a diskette with an Interlace Factor of 2 would be a logical sector 0 followed by logical sector 13, then 1, 14, 2, 15, 3, etc.

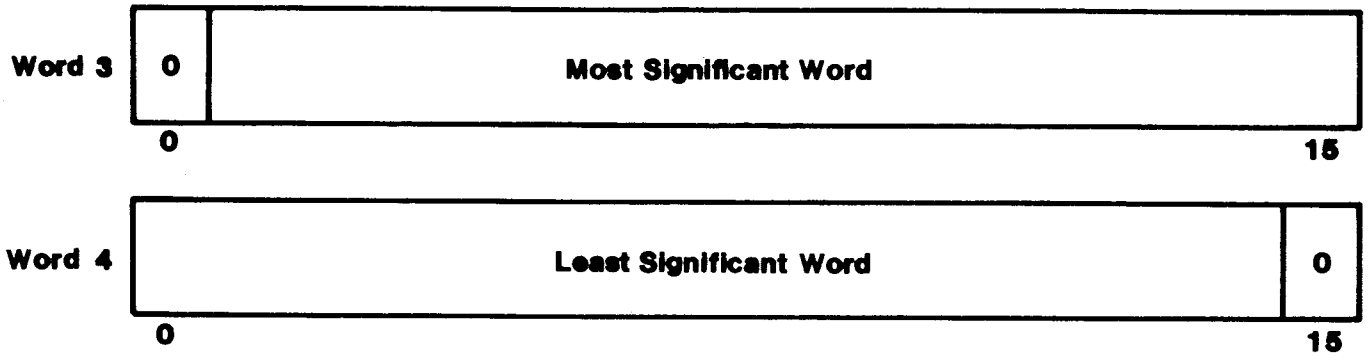
3.4.4 Words 3 and 4, Storage Address on Diskette

Command List Words 3 and 4 may be used in two modes:

- mass storage mode (section 3.4.4.1)
- physical storage mode (section 3.4.4.2)

The modes are specified by a zero (mass storage) or a one (physical storage) in bit zero of word 3.

3.4.4.1 Mass Storage Mode



Note: bit zero of word 3 is a 0 to specify mass storage.

In the mass storage mode, command list words 3 and 4 comprise storage address on the diskette. By dividing this 32-bit number, the disk controller can determine physical track (cylinder), surface (head), and sector address. The mass storage mode address must be on a sector boundary. Diskette storage address (SA) is determined by the following equation:

Diskette Mass Storage Address: $[(T \times N_H \times N_S) + (H \times N_S) + S] \times N_B$

Where:

T = logical track number
H = surface number (0 or 1)
S = sector number (0 to 26)
IBM format: sector number -1
TI format: sector number
N_H = number of surfaces per diskette (1 or 2)
N_S = number of sectors per track (16 or 26)
N_B = number of bytes per sector (128, 256, or 288)

The following are parameters for different diskette formats:

	IBM SD	IBM DD	TI DD	Mini SD	Mini DD	Mod IBM DD
Sectors per Track	26	26	26	16	16	26
Bytes per Sector	128	256	288	128	256	256
First Sector No.	01	01	00	01	01	01
First Track No.	00	00	00	00	00	00

Different diskette formats are shown in Appendix C.

Example 1. Address for logical track 10 (T), surface 0 (H), sector 3 (S) on an IBM single-density single-sided diskette:

Diskette Storage Address: $[(10 \times 1 \times 26) + (0 \times 26) + (3-1)] \times 128$

$(260 + 0 + 2) \times 128$
$(262) \times 128 = 33,536 = 8300_{16}$

Thus, the values entered would be 0000_{16} for word 3 and 8300_{16} for word 4.

Example 2. Address for logical track 12 (T), surface 1 (H), sector 20 (S) on a TI double-sided double-density diskette:

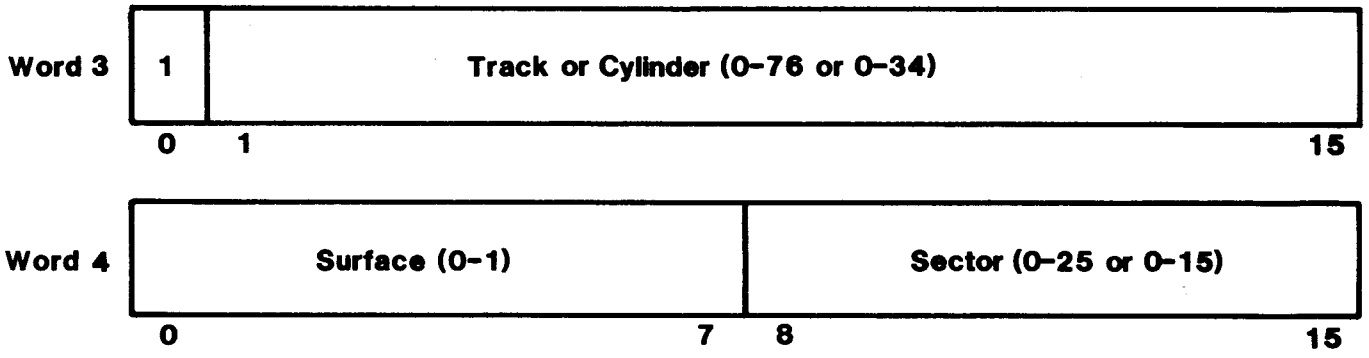
Diskette Storage Address: $[(12 \times 2 \times 26) + (1 \times 26) + 20] \times 288$

$(624 + 26 + 20) \times 288$
$(670) \times 288 = 192,960 = 2F1C0_{16}$

Thus, the values entered would be 0002_{16} for word 3 and $F1C0_{16}$ for word 4.

With a 31-bit storage address, the user has the capability to address up to 2^{31} or 2,147,483,648 bytes. The address range will be 0 to $2,147,483,646_{10}$ or $7FFF\ FFFE_{16}$ bytes.

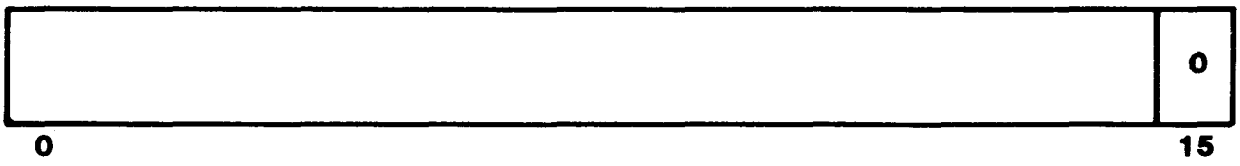
3.4.4.2 Physical Storage Mode



Note: bit zero of word 3 is a 1 to specify physical storage.

The sector field identifies the logical sector on the track. Note that TI double density format begins with sector 00 while the other five diskette formats supported by the TM 990/303B have sectors beginning with 01.

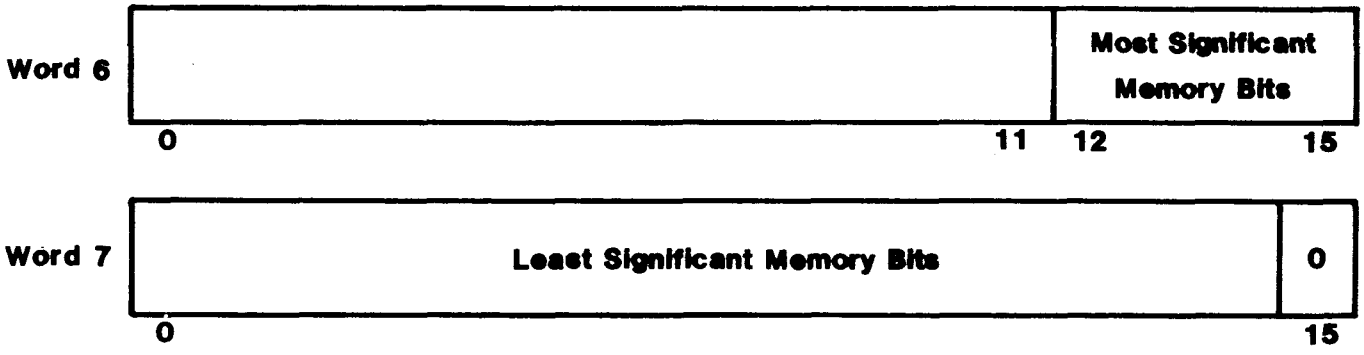
3.4.5 Word 5, Byte Count



Note: Bit 15 must be a zero (even byte count only)

Word 5 contains the amount of bytes to be transferred. This value must be even (bit 15 a zero) and non zero.

3.4.6 Words 6 and 7, Memory Address of Data to Transfer

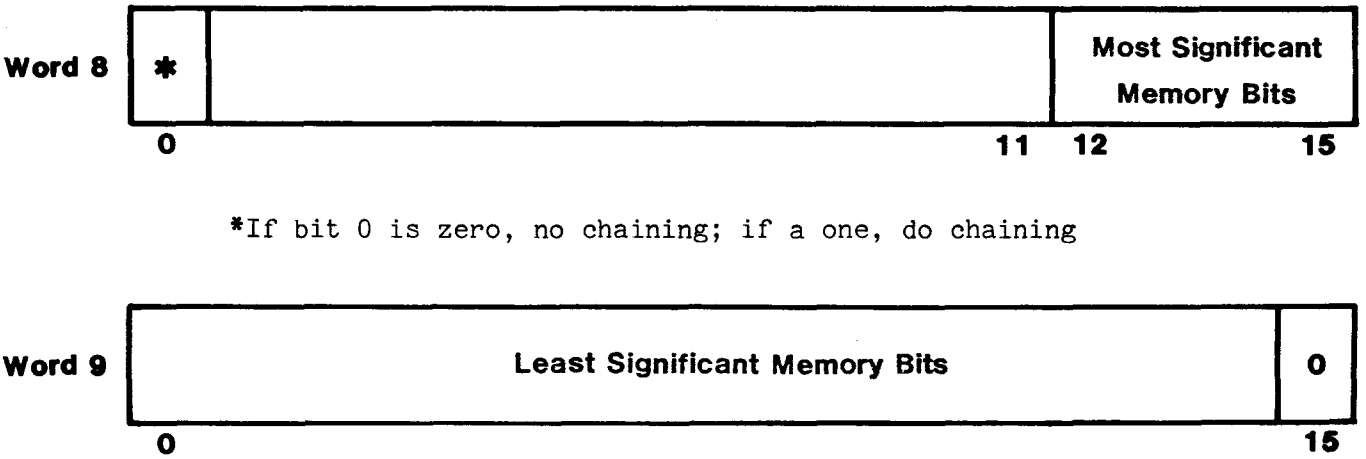


Note: Bit 15 of word 7 must be a zero (word boundary only)

Words 6 and 7 contain the beginning address in host memory for a data transfer. This is a 19-bit address with the extended address bits in bits 12 to 15 of word 6, and the least significant 15 bits in word 7 with bit 15 a zero (word boundary).

3.4.7 Words 8 and 9, Chain Address of Next Command List

In the bootstrap load Command List, these words contain the address where status will be stored.



*If bit 0 is zero, no chaining; if a one, do chaining

Note: Bit 15 of word 9 must be a zero (word boundary only)

If bit 0 of word 8 contains a one, a chaining function will be executed and the disk controller will seek out this address as the beginning of the next Command List to be executed. In this manner, a series of Command Lists may be executed. Words 8 and 9 contain the host memory address of the next Command List to be executed. Bit 0 of word 8 must contain a one in order to enable the chaining function. This is a 19-bit address with the extended address bits in bits 12 to 15 of word 8, and the least significant 16 bits in word 9 with bit 15 a zero (forced word boundary).

3.5 COMMUNICATION THROUGH INTERRUPTS

A third means of communication between the host microcomputer and the disk controller is through one interrupt to the host from the controller and one interrupt to the controller from the host. These interrupts are described as follows:

- From disk controller to host: An interrupt from disk controller software to a jumper-selectable interrupt level on the host tells the host that a command has been completed. To determine the condition of completion (successful, error occurred, etc.), the host should monitor the status bits returned back to the Command List of the respective command. The disk controller is shipped jumpered (at J3) for a level 2 interrupt to the host. An interrupt level of 1 to 9 is selected at jumper J3.

- From host to disk controller: There are two interrupts possible by the host setting the CUE or RESET bit on the CRU (bits 10 and 14 respectively) as shown in Figures 3-2 and 3-3.

  (1) Setting the CUE bit to a one causes an interrupt to level one of the disk controller that tells the controller that the host is to send data to the controller via the CRU. This interrupt is caused by setting the CUE bit to a one as described in section 3.3. Because the controller will not respond to an interrupt if busy, the host should first check the BUSY bit at the CRU (bit C_{16} using software base address 0210_{16}) before issuing the interrupt. This CRU interface is shown in Figures 3-2 and 3-3.

  (2) Setting the RESET bit to a one causes an interrupt level 0 to the controller as explained in section 3.3.1.5.

All interrupts are maskable at the host's TMS 9901 parallel interface as well as at the interrupt mask at the microprocessor (if the mask level is set to a lower number than the interrupt level, the interrupt is masked out).

3.5.1 Command Completion Interrupt from Disk Controller to Host

Through an interrupt, the controller can tell the host microcomputer that a command has been completed. In order for the host to answer interrupts, the host must supply vectors (workspace pointer and program counter values) for the interrupt in lower memory. The TM 990/100M TIBUG monitor comes from the factory with vectors for interrupt traps 3 and 4. The TM 990/101MA TIBUG monitor and TM 990/102 MAPBUG monitor use schemes whereby all interrupt vectors are populated. The following discussion is for a TM 990/101MA microcomputer.

The command-completion interrupt level is jumper selectable on the TM 990/303B at J3 (jumper positions are explained in Figure 2-1 and Table 2-1). For example purposes, the following steps must be accomplished in order to enable a level 2 interrupt to be recognized at the host TM 990/101MA microcomputer (coding to accomplish this is shown in Figure 3-9).

1. The Program Counter interrupt vector for interrupt 2 points to $FF6E_{16}$ (this is contained in memory address $000A_{16}$). Therefore, set up the level 2 interrupt link area at the TM 990/101MA by:

   - Loading memory address $FF6E_{16}$ with 0420_{16} (BLWP instruction).
   - Loading memory address $FF70_{16}$ with the address of the two-word vector area pointing to the interrupt service routine (ISR).
   - Loading memory address $FF72_{16}$ with 0380_{16} (RTWP instruction).

2. At the two-word vector area of the ISR, load the ISR workspace pointer and ISR entry (PC value) addresses. Conclude the ISR with an RTWP. The ISR should toggle the Interrupt Enable CRU bit to shut off the interrupt and reenable it, if desired.

3. Set the jumper on the disk controller at J3 to level 2 to use interrupt 2 in this example (this is the position as shipped).

4. Enable the interrupt level at the host microcomputer module's TMS 9901 so that it recognizes the desired interrupt level (level 2 in this example). To do this:

   - Load the TMS 9901 software base address in R12 (0100_{16} for the TM 990/101MA microcomputer).
   - Through the CRU, set bit 0 to a zero to place the TMS 9901 in the interrupt mode (SBZ 0).
   - Through the CRU, set bit 2 to a one to enable interrupt 2 in this example (SBO 2).

5. Set the interrupt mask at the microprocessor to a level 2 or lower priority (LIMI 2 or higher number in operand for this example).

6. Set the CRU software base address to 210_{16} (controller to host communication). Through the CRU (explained in section 3.3), change (toggle) the Interrupt Enable bit (bit D_{16}) from a logical zero to a logical one.

7. When building the Command List, set the Interrupt Enable bit (bit 8 of word 2) to a one so that completion is signaled with an interrupt.

When the command list is executed and the execution-complete interrupt is issued, the Interrupt Occurred bit in the Command List (bit 2 of word 0) is set to one. To re-enable interrupts, set the Interrupt Enable bit at the CRU to zero, then a one, and repeat steps 1 to 7.

NOTE

It is not practical to use interrupts with chained commands where an interrupt is requested for only the last command. Instead, request an interrupt for each command (not just for the last command). Otherwise, if a error occurs during a command in the middle of the chain and no interrupt is requested, execution of the entire chain halts without informing the host through an interrupt. The host would have to poll the Command Completion bit of each Command List to find the Command List last completed.

The interrupt service routine in the host can be given several responsibilities:

1) Determine the address of the just-completed Command List through a pointer pointing to the address of the last-executed Command List. Keep this pointer updated.

2) Determine that the controller generated an interrupt by checking CPU Interrupt Issued bit and the Operation Complete bit (Word 0, bit 0) of the Command List.

3) Determine if an error occurred during the last command by checking word 0, bit 1 (ER). If a one, check word 0, bit 15 (UE) to see if the error indicator is in word 1; if not, check the error indicators in word 0 (word 0, bit 15 a one means indicator is in word 1, a zero indicates indicator is in word 0).

4) Re-enable interrupts at the CRU and at the TMS 9901 and at the microcomputer interrupt mask of host (see example in Figure 3-9).

3.5.2 Initiate Command Execution Interrupt from Host to Disk Controller

When the user wishes to call the controller to execute a Command List, he calls it through the CRU as explained in section 3.3. This call through the CRU actually causes an interrupt at the controller.

This interrupt is initiated by setting the CUE bit to a one. When this occurs, an interrupt is sent to INT1- of the controller's TMS 9901 interrupt interface. See section 3.3 for a full explanation of CRU communication. The host can also reset the controller throughthe CRU RESET line (see section 3.3.1.5).

NOTE

The EIA port also has jumper selectable interrupt levels (levels 1 - 9) to the host. However, this port is not functionally part of the controller, and is considered to be part of the host system from a software point of view. Interrupts associated with this port are covered in Section 5.

```
                    Source Line
                    Memory Address (Beginning at Address 0000)
                    Object Code (Relative to Beginning Address 0000)

0001                * THIS PROGRAM INITIALIZES A CPU BOARD TO RECEIVE AN
0002                * INTERRUPT LEVEL 2. THE PROGRAM COUNTER VALUE FOR
0003                * INTERRUPT LEVEL 2 IS FOUND AT ADDRESS HEX 000A
0004                      IDT  'INTDEM'
0006                * SET UP INTERRUPT LINK AREA IN HOST RAM
0007 0000 C060            MOV  @>000A,R1    ADDR OF PC VECTOR FOR INT 2
     0002 000A
0008 0004 0202            LI   R2,>0420     BLWP MACHINE CODE
     0006 0420
0009 0008 CC42            MOV  R2,*R1+      MOVE BLWP CODE TO INTERRUPT LINK AREA
0010 000A 0202            LI   R2,>FC00     ISR VECTORS ADDRS (MAKES BLWP @>FC00)
     000C FC00
0011 000E CC42            MOV  R2,*R1+      MOVE TO INTERRUPT LINK AREA
0012 0010 0202            LI   R2,>0380     RTWP MACHINE CODE
     0012 0380
0013 0014 C442            MOV  R2,*R1       MOVE TO INTERRUPT LINK AREA
0014                * SET UP TWO-WORD VECTOR TO INTERRUPT SERVICE ROUTINE (ISR)
0015 0016 0201            LI   R1,>FC00     ADDRESS OF BLWP VECTORS IN HOST RAM
     0018 FC00
0016 001A 0202            LI   R2,>FC80     ISR WORKSPACE POINTER
     001C FC80
0017 001E CC42            MOV  R2,*R1+      MOVE TO VECTOR AREA (WORD 1)
0018 0020 0202            LI   R2,>FCA0     ISR ENTRY (PC VALUE)
     0022 FCA0
0019 0024 C442            MOV  R2,*R1       MOVE TO VECTOR AREA (WORD 2)
0020                * ENABLE TMS 9901 AT HOST TO RECOGNIZE INTERRUPT 2
0021 0026 020C            LI   R12,>100     9901 SOFTWARE BASE ADDRESS
     0028 0100
0022 002A 1E00            SBZ  0            9901 TO INTERRUPT MODE
0023 002C 1D02            SBO  2            ENABLE INTERRUPT 2
0024                * ENABLE HOST INTERRUPT MASK TO RECOGNIZE INTERRUPT 2
0025 002E 0300            LIMI 2
     0030 0002
0026                * ENABLE INTERRUPTS AT CONTROLLER
0027 0032 020C            LI   R12,>210     CRU BASE ADDRESS OF CONTROLLER
     0034 0210
0028 0036 1E0D            SBZ  13           SET TO ZERO
0029 0038 1D0D            SBO  13           TOGGLE BACK TO ONE
0030                * IN BUILDING COMMAND LIST, BE SURE TO SET INTERRUPT
0031                * ENABLE BIT TO A ONE (WORD 2, BIT 8)
```

NOTE

When interrupt 2 is issued, program execution acts as a BLWP to the address in memory address $000A_{16}$. Then it will execute the code as loaded above (BLWP @ >FC00 with WP and PC vectors of >FC80 and >FCA0 at> FC00). Return is via successive RTWPs.

FIGURE 3-9. LISTING TO LOAD VECTORS FOR INTERRUPT LEVEL 2

3.6 POWERUP BOOTSTRAP LOAD OPTION

3.6.1 General

If jumpered, a bootstrap load feature allows the initialization of the system by executing a bootload program placed on a disk by the user. The controller is responsible for determining which of three bootload formats is used by the system (these formats are defined in the Bootload Command, 05, in Table 3-2):

- 303 format where a Command List is read off of disk and executed.
- TX4 or TM990 AMPLUS formats where a program is read off of disk and executed.

If jumpered at J1, J2, and J14 (see Table 2-1), the bootstrap load occurs during one of the following:

- upon powerup of the disk controller,
- upon execution of the software Bootstrap Load command (command 05), whether or not jumpers J1/J2 are set for bootstrap load,
- an active PRES.B- signal on the backplane (pin P1-94).

NOTE

Powerup bootstrap load can be used only with the disk format specified specified at jumpers J8 and J9. See Table 2-1.

The powerup bootstrap load is activated by the PRES.B- signal (not the IORST.B- signal). When executed, the host microprocessor is placed in a hold state until the bootstrap load is completed (see conditions to cause a system lockup in section 3.6.5). After the bootload program is loaded into host memory, the hold state is released on the host processor, and the normal power up level 0 interrupt occurs to start host processor execution. To implement this option, the following jumpers must be set as shown:

- Jumper J1 E1 to E2
- Jumper J2 E4 to E5
- Jumper J14 Installed on CPUs which do not have powerup circuitry that asserts PRES.B- (e.g., TM 990/100MA and TM 990/101MA).

  Removed on CPUs which have powerup circuitry that asserts PRES.B- (e.g., TM 990/102).

Unless bootload jumpers J1 and J2 are installed for automatic bootload, jumper J14 has no meaning.

The bootstrap load causes the controller to read the first sector of the first track. From this information, along with the type of diskette being used (denoted by jumpers J8 and J9), the controller can tell which of the three bootload formats is present on the disk. These three are documented in the Bootload Command, 05, in Table 3-2. (It is advised that you read the explanation of these formats before proceeding.) From this information, the controller will attempt to bootload the host microcomputer.

The controller will return the status of the bootload as follows:

- if a Bootstrap Load Command (05) is used, status is returned to words 0 and 1 and transfer count will be returned to word 5 of the host Command List, or

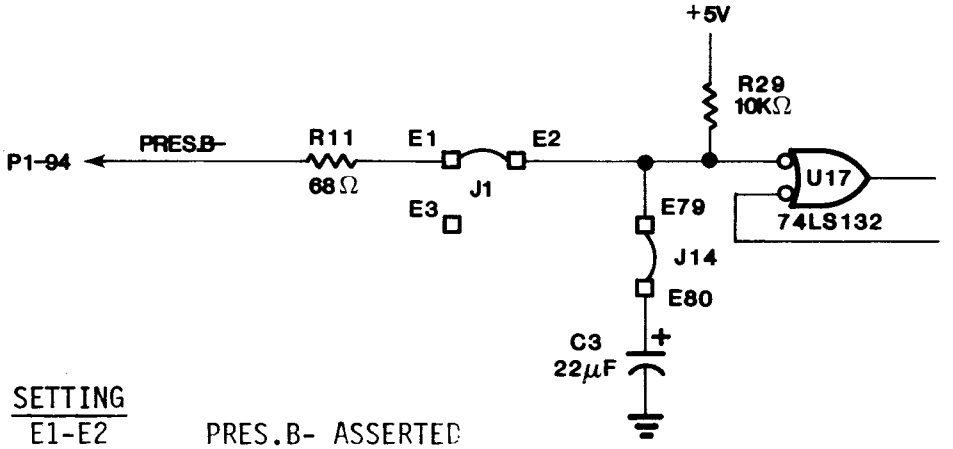
- if a 303-format imbedded Command List, list words 8 and 9 specify the location of a 10-word block. The two status words are placed in words 0 and 1 of this block and the transfer length will be placed in word 5, or

- if TX4 or TM990 AMPLUS, the status is returned to the address immediately after the workspace (this WP is located right after the boot program). It is the responsibility of the host to first format the diskette with the desired bootstrap program in the appropriate sector and track.

Figure 3-10 shows how the auto bootstrap function is asserted to pin P1-94 (PRES.B-) for a powerup bootload. With J14 jumpered to attach a 22 uF capacitor to the PRES.B- line, the +5 V at powerup rises slowly to assert PRES.B-. This is necessary for CPU modules such as the TM 990/100MA or TM 990/101MA, but is not necessary for the TM 990/102 which asserts PRES.B- at powerup via its TL7705 chip. With PRES.B- low, the RESET- pin to the processor is low, placing the processor in a reset state.

While the bootload is being executed, the host processor is held in an idle state by controller hardware holding HOLD.B- low. After the boot has been transferred to the host with no errors detected, the HOLD.B- line is released and the host is allowed to execute an interrupt level zero. If it is desired to execute the loaded program, the PC vector at memory address 0002_{16} should be coordinated with the entry address of the just-loaded program. Note that the TM 990/102 CPU module is powered up in map page zero.

The TM 990/102 module has logic onboard to assert PRES.B- to itself. With a TM 990/102 as the host processor, the bootload RESET logic should be disabled by removing jumper J14.



JUMPER	SETTING	
J1	E1-E2	PRES.B- ASSERTED
J1	E1-E3	PRES.B- NOT ASSERTED
J14	IN	POWERUP PRES.B- ASSERTED (IF J1 = E1-E2)
J14	OUT	POWERUP PRES.B- NOT ASSERTED

FIGURE 3-10. AUTO BOOTSTRAP LOAD CIRCUITRY JUMPERED TO ASSERT PRES.B- TO HOST CPU

3.6.2 Preparing a Diskette for Bootstrap Load

The procedure for preparing the diskette for the bootstrap load is as follows:

1. Write the bootload on the disk in one of three acceptable formats as explained in the Bootload Command (Command 05, Table 3-2).

2. If the bootstrap load program (see 1, above) is to read data from the disk and write this to host RAM, this data also must have been written onto the disk prior to executing the bootstrap load.

3. If a powerup bootload is to read data into host RAM to be executed, the PC vector for interrupt 0 at address 0002_{16} must point to an entry point in the loaded program.

4. When issuing a bootload command (command 05), check where the boot is to be loaded into host memory. Make sure the loaded program does not wipe out the command list that specified the bootload or the DSR that is waiting for boot completion.

NOTE

This initialization of the system diskette must be done before the diskette is used by the host system bootstrap load routine.

3.6.3 Tests Performed at Bootload

During a powerup bootload, the TM 990/303B performs a self-test operation. If an error is found during self-test, the powerup bootload operation is not completed and the LEDs will be encoded with the type of error found. The LED error setup is explained in Table 2-2. The bootload code performs two additional tests that are not performed by a Self-Test command (Command 01) or a Reset command (Command 02) or a powerup reset. If the bootload is in a TX4 or a TM990 AMPLUS format, the TM 990/303B will attempt to find the map (i.e., which 64 K byte page) that the host will run from. If there are multiple maps used, the TM 990/303B will find the lowest numbered map and use it as the map to attempt the bootload into. If the TM 990/303B does not find a map, the bootload will halt and all the LEDs will be lit. The map test is not performed for an imbedded boot, because the imbedded command list specifies which map to use.

The other test executed is a test of the DMA circuitry and the host system RAM. This test is performed for all bootload types at the host address where the boot is to be placed. The bootload information on the disk specifies where the bootload program is to be loaded into host memory. This test checks a 256 byte block of host RAM. The RAM is tested with data patterns of 0000, 1111, 2222 through $FFFF_{16}$. The TM 990/303B cannot differentiate between a DMA logic error or a host RAM error. If an error does occur, the bootload operation halts, the TM 990/303B returns to the idle state, and the LEDs are encoded as follows: DS3 and DS2 will be on and DS1 will be off.

There are ten allowable formats that can be sensed at bootload. These are:

Format	TWOSIDED- Signal	Jumper	
		J8	J9
(8 in.) IBM S Density, S Sided	Not Active	In	In
(8 in.) IBM S Density, D Sided	Active	In	In
(8 in.) IBM D Density, S Sided	Not Active	In	Out
(8 in.) IBM D Density, D Sided	Active	In	Out
(8 in.) IBM Modified, D Density, S Sided	Not Active	In	Out
(8 in.) IBM Modified, D Density, D Sided	Active	In	Out
(8 in.) TI D Density, S Sided	Not Active	Out	In
(8 in.) TI D Density, D Sided	Active	Out	In
(5¼ in) Mini S Density, S Sided	Not Available	Out	Out
(5¼ in) Mini D Density, D Sided	Not Available	Out	Out

These formats will be sensed depending upon J8 and J9 configuration, along with the TWOSIDED- disk signal line. The parameters supplied in a Define Drive command take precedent over the jumper positions set on the module.

Mini disk drives do not have a TWOSIDED- line, so the diskette is assumed to be single sided. If the bootload fails with a SD-SS mini disk drive, the drive is redefined as DD-DS and the bootload is attempted once again. If the DD-DS bootload attempt fails, the bootload operation ceases.

The modified IBM double density (DD) format will be supported at bootload in the following manner. To bootload from a modified DD IBM diskette, the format jumpers must be set for IBM regular DD format. IBM regular DD format specifies that track 0 will be single density (SD) with remaining tracks double-density. The controller, after failing to read the first track in SD, will redefine the drive as having a diskette with a modified IBM DD format and reattempt the bootload. After redefining the format to modified DD, the first track can be read and the host computer can successfully be bootloaded into.

3.6.5 Conditions That Can Lock Up System at Bootstrap Load

The disk controller could lock up the system during a bootstrap load under one of these conditions:

1. A CRC error in reading the first sector of track 0 from the disk. Either this or the next condition (2) will place the controller in the idle mode without releasing the host microprocessor which remains in the hold mode.

2. An error during execution of the Command List built by the controller.

3. The controller fails the self-test operation in which case the LEDs will be lit with the appropriate error code (as shown in Table 2-2).

4. The controller fails to find the map for which the host is configured, in which case the LEDs DS1, DS2, and DS3 will be lit (on).

5. The controller does not successfully self-test its DMA circuitry. LEDs DS3 and DS2 will be on and DS1 will be off.

6. There is no disk drive, no diskette, or the diskette is in an incorrect format.

SECTION 4

HARDWARE DESCRIPTION

4.1 GENERAL

An overview of a typical system using the TM 990/303B floppy disk controller will be presented first, followed by a detailed description of the disk controller including the local processor, disk drive interface, host system interface, read/write controller, and EIA port.

4.2 SYSTEM DESCRIPTION

A typical microcomputer disk system is shown in Figure 4-1. The major elements in the system are the TM 990/303B floppy disk controller, TM 990/101MA microcomputer, TM 990/203 memory, TM 990/510A card cage, terminal, and one to four disk drives.

The TM 990/303B floppy disk controller communicates with the microcomputer system using the Communications Register Unit (CRU) for initialization and direct memory access (DMA), for command, status, and data transfer. The disk controller can be configured to automatically load a program into memory upon system powerup (this is called a bootload).

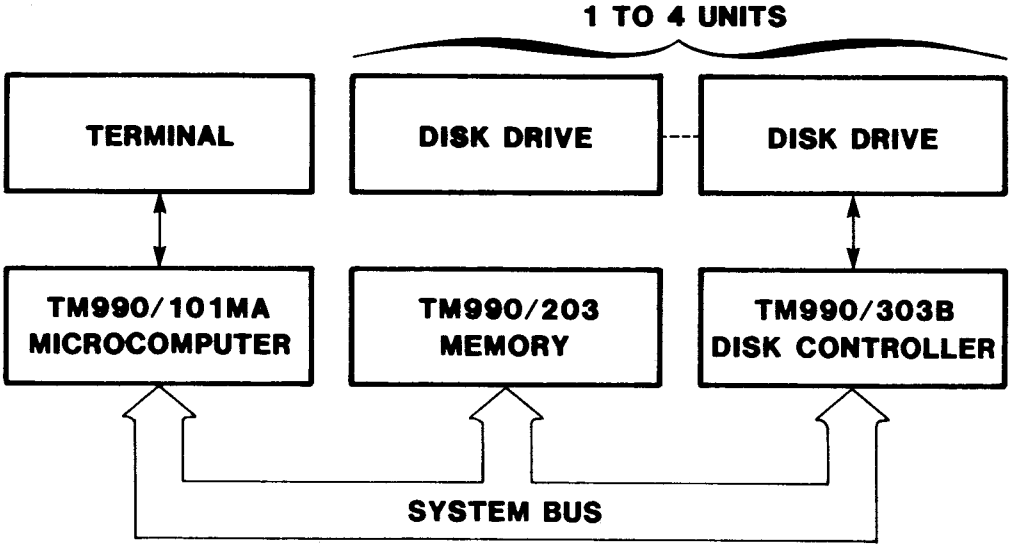
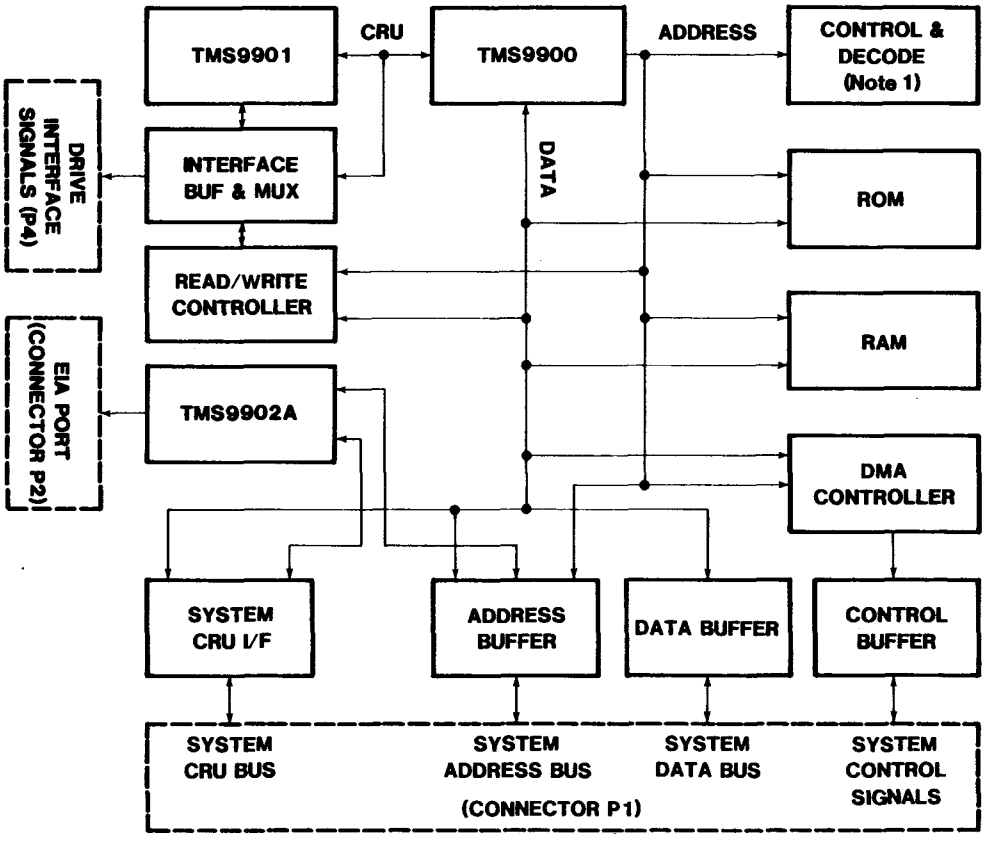


FIGURE 4-1. TYPICAL SYSTEM BLOCK DIAGRAM

4.3 CONTROLLER DESCRIPTION

The TM 990/303B floppy disk controller consists of a local processor system (section 4.4), disk drive interface (section 4.5), host system interface (section 4.6), read/write controller (section 4.7), and EIA port (section 4.8) as shown in Figure 4-2. The local processor system contains a TMS 9900 microprocessor, TIM 9904A clock generator, read-only memory (EPROM), read-write memory (RAM), decode PROM, and wait-state control logic. The disk drive interface contains a TMS 9901 programmable systems interface, a multiplexer for additional inputs, and disk drive interface driver and receiver circuits. The host system interface consists of a CRU interface and a DMA interface. The DMA interface contains an input data register, an output data register, a memory address register, and memory access control logic. The read/write controller contains two ROM controllers for data bit and word processing, a data shift register, a cyclic redundancy check (CRC) circuit, a phase-locked loop (PLL) for data synchronization and control logic.



Note 1: Control & Decode Block Functions:

- **RAM/ROM Access**
- **DMA Enable**
- **Read/Write Controller Enable**

FIGURE 4-2. DISK CONTROLLER BLOCK DIAGRAM

4.4 LOCAL PROCESSOR SYSTEM

The local processor system contains a TMS 9900 microprocessor operating at 3 MHz. The clocks for the processor are provided by a TIM 9904A clock generator using a 12 MHz crystal oscillator which is divided down to give the four-phase clocks for the processor. The clock generator also provides synchronization of the processor RESET signal as well as crystal-controlled clock signals for the read/write controller.

The local processor memory consists of 8192 bytes of ROM and 2048 bytes of RAM. The local memory bus is also used for local, memory-mapped I/O. Local memory addresses are decoded by a 32 X 8 PROM and by a 3 to 8-line decoder which is gated with write enable (MWE-) to generate load signals for local RAM and data registers.

The DMA address, DMA data, command data (from host system CRU), and read/write controller data register are memory-mapped I/O devices. The local processor memory map is given in Table 4-1. Certain memory address bits are used during memory-mapped I/O to the DMA and read/write controllers in addition to the data bus. The DMA and read/write controllers will delay the local processor using wait states via the READY signal if they are accessed prior to data being available.

TABLE 4-1. PROCESSOR MEMORY ADDRESS MAP

Address	Read Function	Write Function
0000-1FFE	ROM	
2000-3FFE	Reserved	Reserved
4000-5FFE		DMA Address
6000-7FFE	*DMA Data	*DMA Data
8000-9FFE	*Read/Write Controller	
A000-BFFE		*Read/Write Controller
C000-DFFE	Command Data	
E000-F7EE	Reserved	Reserved
F800-FFFE	RAM	RAM

*Processor wait states are used for synchronization.

4.5 DISK DRIVE INTERFACE

The disk drive interface consists of a TMS 9901 programmable interface adapter, an 8-input multiplexer, a drive-select decoder, output buffers, and input receiver circuits. The TMS 9901 contains parallel I/O, interrupt, and timer sections interfaced to the local processor through the CRU. The parallel I/O port CRU bit assignments are given in Table 4-2. Fifteen ports are used as outputs and one port is used as an input. The ACCEPT, BUSY, SYSINT-, and CCOMMAND signals are used by the host system interface. The DDENSITY, RFMTWPCOMP, LED- and RWRST- signals are used by the read/write controller. The other I/O ports are used as outputs to the disk drive interface circuit.

The disk drive select signals, DSELECT1- through DSELECT4-, are generated by decoding the DSEL0 and DSEL1 signals from the TMS 9901. The drive select signals and all other drive control signals except HEADLOAD- and WRITEDATA- are controlled by the DRIVENABLE- output from the TMS 9901. The DRIVENABLE- signal is pulled-up to the inactive state whenever the TMS 9901 is reset (All

TMS 9901 outputs become inputs). The HEADLOAD signal will be activated when the TMS 9901 is reset, but no headload will occur because the drive select will be pulled high. (This assumes the correct headload option is jumpered on the drive).

TABLE 4-2. TMS 9901 PARALLEL I/O PORT BIT ASSIGNMENT (page 1 of 2)

Onboard 303B CRU Base Address (R12) is $1F60_{16}$.			
BIT ADDRESS	IN/OUT	SIGNATURE	FUNCTION
0	Out	DSEL0	Disk drive unit select LSB.
1	Out	DSEL1	Disk drive unit select MSB.
2	Out	WGATE	Disk drive write gate. When active (high), WGATE enables data to be written on the disk.
3	Out	DIR	Disk drive step direction.
4	Out	SIDE	Disk drive side select.
5	Out	DDENSITY	When active (high), DDENSITY causes the data separator to interpret read data in double density mode. After changing this bit, allow at least 12 microseconds before trying a read or write operation. When inactive, DS3 is lit.
6	Out	RFMTWPCOMP	When active (high), RFMTWPCOMP selects the TI sync format in read-mode and enables precompensation in write mode. When inactive in read-mode, IBM sync format is selected and precompensation is disabled in write mode. When inactive, DS2 is lit.
7	Out	STEP	Disk drive step.
8	Out	HLOAD	Disk drive head load.
9	Out	ACCEPT	When active (high),ACCEPT indicates to the host system that a command or data byte has been read by the disk controller.
10	Out	BUSY	When active (high), BUSY indicates to the host system that the disk controller is busy and cannot accept a new command.
11	Out	RWRST-	When active (low), RWRST- clears all flip-flops, registers, and counters in the read/write controller except for the data separator and clears the read/write controller address bit functions (All R/W controller signals are inactive except RESET- and PRESETCRC-).
12	Out	DRIVENABLE-	When active (low), DRIVENABLE- enables all disk drive control signals except HEADLOAD and WRITE DATA (outputs from the controller). DRIVENABLE- is inactivated when the controller is reset.

TABLE 4-2. TMS 9901 PARALLEL I/O PORT BIT ASSIGNMENT (page 2 of 2)

BIT ADDRESS	IN/OUT	SIGNATURE	FUNCTION
13	Out	SYSINT-	System interrupt. The high-to-low transition of SYSINT- sets the INT flip-flop. This will cause an interrupt to be issued to the host system if interrupts have been enabled through the CRU. When active (low), SYSINT- clears the automatic bootload flip-flop.
14	Out	LED-	When active (low), LED- turns on disk controller status indicator DS1.
15	In	CCOMMAND	During a CRU Command List address transfer the CCOMMAND, if active (high), indicates that the first byte of the 3 byte address is being transferred. This signal can also be used as interrupt 7 by the TMS 9901.

The TMS 9901 interrupt/input assignment is given in Table 4-3. These signals can be used as either interrupts or inputs as required by the firmware. The TMS 9901 timer section is used by the firmware to provide timing for various disk operations. CCUE- is the only interrupt line actually used as an interrupt.

An 8-bit multiplexer is also interfaced to the local TM 990/303B processor through the CRU. This multiplexer is connected to input port P4. The CRU bit assignments are given in Table 4-4. In addition to disk drive inputs, two jumpers (J8 and J9) are connected to this input port. These jumpers are used in conjunction with the two-sided signal (TWOSIDED-) to define the disk drive type to be used during powerup or reset operation. J8 and J9 are used to determine the default value of the disk format (IBM or TI or mini) during a bootload. The jumper configurations are given in Table 4-5.

TABLE 4-3. TMS 9901 INTERRUPT/INPUT ASSIGNMENT

Onboard 303B CRU Base Address (R12) is $1F40_{16}$.			
BIT ADDRESS	INTERRUPT LEVEL	SIGNATURE	FUNCTION
0			Control bit. Set to zero to write mask or read interrupts.
1	1	CCUE-	When active (low), CCUE- indicates that the host system is sending data or command to the disk controller.
2	2	-------	Not connected.
3	3	ALHOLDQ-	Automatic bootload. When active (low), ALHOLDQ- indicates that the automatic bootload flip-flop is set as the result of system power-up or the power-on reset signal (PRES.B-).
4	4	OVERRUNQ-	Data overrun error. When active (low), OVERRUNQ- indicates that a data timing error has occurred during a disk read or write operation.
5	5	CRCERRORQ-	Cyclic redundancy check error. When low (active), CRCERRORQ- indicates that a data error has occurred during a disk read operation.
6	6	INDX-	Disk drive index. When active (low), INDX- indicates that the index hole in the diskette is being sensed by the disk drive.
7	7	CCOMMAND	During a CRU Command List address transfer the CCOMMAND, if active (high), indicates that the first byte of the 3 byte address is being transferred.

TABLE 4-4. PARALLEL INPUT PORT BIT ASSIGNMENT (P4)

Onboard 303B CRU Base Address (R12) = $1F80_{16}$.		
BIT ADDRESS	SIGNATURE	FUNCTION ADDRESS
0	TWOSD	Disk drive two sided.
1	DCHG	Disk drive disk change.
2	DRDY	Disk drive ready.
3	TZERO	Disk drive track zero.
4	WPROT	Disk drive write protect.
5	FMTJ9	*If jumper J9 is jumpered, this is HIGH.
6	FMTJ8	*If jumper J8 is jumpered, this is HIGH.
7	SPAREIN	Spare input. This bit indicates the state of pin 24 of the disk drive connector P4.

* See Table 2-1 for J8 and J9 settings which define default formats.

TABLE 4-5. BOOTLOAD FORMAT SELECTION

Format Selected	Jumper Connections		TWOSIDED- Signal
	J8 E81 to E82	J9 E84 to E83	
IBM S Density, S Sided	Install	Install	Not Active
IBM S Density, D Sided	Install	Install	Active
IBM D Density, S Sided	Install	Remove	Not Active
IBM D Density, D Sided	Install	Remove	Active
IBM Modified D Density, S Sided	Install	Remove	Not Active
IBM Modified D Density, D Sided	Install	Remove	Active
TI D Density, S Sided	Remove	Install	Not Active
TI D Density, D Sided	Remove	Install	Active
Mini S Density, S Sided	Remove	Remove	---
Mini D Density, D Sided	Remove	Remove	---

NOTE

A powerup bootload or powerup reset requires the correct sided diskette in the drive. The correct diskette will result in pin P4-10 being a "0" for two-sided diskettes and a "1" for one-sided diskettes correctly placed in the drive. This signal is inverted at U49 and is valid only when the drive is selected for access. P4-10 is not used for mini drives because mini drives do not have this interface line.

4.6 HOST SYSTEM INTERFACE

The TM 990/303B disk controller contains a host system CRU interface for initialization and a host system DMA interface for command and data transfer.

4.6.1 Host System CRU Interface

A block diagram of the host system CRU interface is shown in Figure 4-3. The host system software CRU base address is set to 0210_{16} by a 256 X 4 PROM which is installed in the disk controller prior to shipment (section 3.3 and Figure 3-3 explain the CRU software base address and the resulting 32 CRU bits used). If some other CRU base address is required by the user, the PROM can be replaced with an appropriate PROM (Refer to Appendix E for PROM coding requirements). The disk controller is assigned 32 consecutive CRU bit addresses starting at the base address. 16 bits are assigned to the general-purpose CRU interface scheme as shown in Table 4-6. The other 16 bits are reserved.

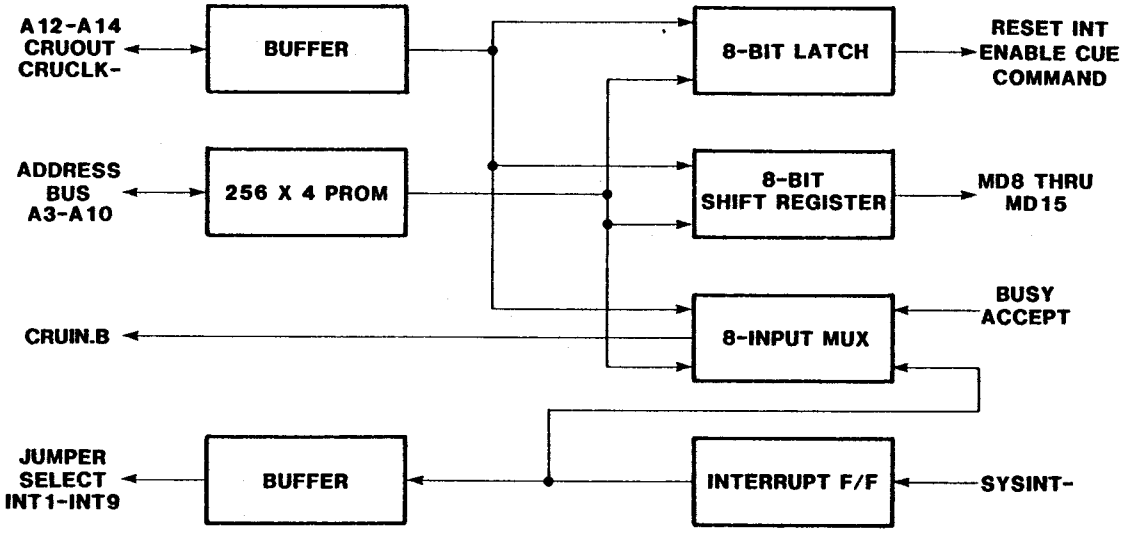


FIGURE 4-3. SYSTEM CRU INTERFACE BLOCK DIAGRAM

TABLE 4-6. GENERAL-PURPOSE CRU INTERFACE

Host System CRU Base Address is 0210_{16}.		
Host System Relative CRU Bit Address	Output Function (to Disk Controller)	Input Function (from Disk Controller)
0_{16}	LSB	--
1_{16}		--
2_{16}		--
3_{16}	Output Data	--
4_{16}		--
5_{16}		--
6_{16}		--
7_{16}	MSB	--
8_{16}	COMMAND	--
9_{16}	--	--
A_{16}	CUE	--
B_{16}	--	ACCEPT
C_{16}	--	BUSY
D_{16}	INTERRUPT ENABLE	--
E_{16}	RESET	--
F_{16}	--	INTERRUPT ISSUED

The general-purpose CRU interface provides a means to transfer data between two systems. The disk controller implements a subset of the general-purpose CRU interface as shown in Table 4-6. Eight-bit data fields are transferred from the host system to the disk controller using the CUE and ACCEPT signals in a handshaking scheme as shown in Figure 4-4a. The handshaking proceeds sequentially as shown. The CUE signal is activated by the host system when ACCEPT is inactive and after the data field and COMMAND signal are valid. The ACCEPT signal is activated by the disk controller after the data field and COMMAND signal have been read. The CUE and ACCEPT signals are then inactivated in sequence as shown. The first activation of CUE causes interrupt level 1 on the controller. Then the controller masks off all interrupts and the CCUE- line is used as an input to the TMS 9901.

The 19-bit command list address is sent to the disk controller in three 8-bit data transfers as shown in Figure 4-4b. The COMMAND signal is activated only for the first data byte. The BUSY signal is activated by the disk controller

after accepting the first data byte and remains set until the controller can accept a new command. Command 12 allows the user to modify the CRU interface such that the controller knows where the Command List will be in memory (see section 3.4). The alternate interface is shown in Figure 4-4c.

The general-purpose CRU interface scheme allows the host system to reset the disk controller by activating the RESET signal. (The RESET signal causes a hardware reset of the TM 990/303B.) This causes the local 303B processor to reset and execute its initialization firmware when the RESET signal is inactivated. The host system can control disk controller interrupts through the INTERRUPT ENABLE signal. Activating this signal allows the disk controller to interrupt the host system. Inactivating this signal masks off any disk controller interrupt to the host system.

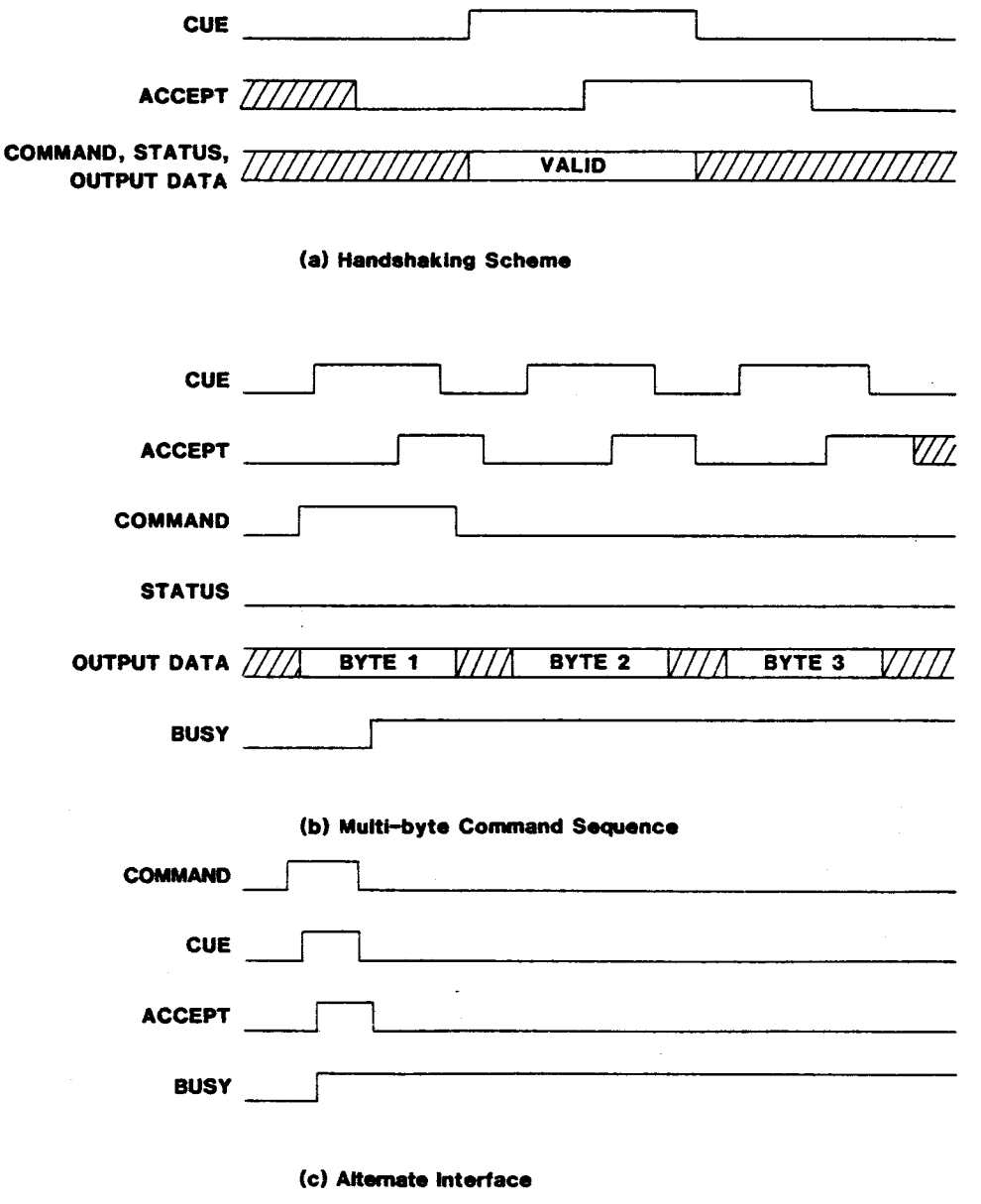


FIGURE 4-4. GENERAL-PURPOSE CRU INTERFACE

An interrupt is reset and re-enabled by inactivating, then activating the INTERRUPT ENABLE signal. If an interrupt is being issued by the disk controller, the INTERRUPT ISSUED signal is activated. This signal can be tested by the host system to determine if the disk controller is interrupting in systems which allow other devices to share the same interrupt level as the disk controller. Once issued, the disk controller interrupt is latched until reset by the host system.

The general-purpose CRU interface data field is transferred from the host system CRU into an 8-bit, parallel-output shift register (memory-mapped at $C001_{16}$). This shift register is a memory-mapped I/O device on the local processor memory bus (bits 8 through 15). The shift register must be loaded by an 8-bit Load CRU (LDCR) instruction prior to setting the CUE bit.

4.6.2 Host System DMA Interface

The host system DMA interface contains two 16-bit data registers (one for input, one for output), a 19-bit address register, memory-access control logic, control signal buffers, and the automatic bootload latch. The data registers allow data to be transferred between the host system memory bus and the local processor memory bus. The DMA Controller local memory address space is from 4000_{16} to $7FFE_{16}$. These registers are memory-mapped I/O devices on the local memory bus. The host system memory address is stored in a 19-bit register. This register is also a memory-mapped I/O device on the local memory bus. The address register must be loaded with a new address prior to each host system DMA cycle. The fifteen least-significant address bits are loaded from the local memory data bus; the four most-significant address bits are loaded from the local memory address bus. Thus all nineteen address bits can be loaded with one firmware instruction. In addition, the data transfer direction is also defined through the local memory address bus. The DMA controller address bit are defined in Table 4-7.

The extended address and transfer direction are loaded into the DMA controller from the address bus at the same time the lower 16 address bits are loaded from the data bus using a base address of 4000_{16}.

TABLE 4-7. DMA CONTROLLER ADDRESS BITS

Address Bit	Signature	Function
7	WRITEQ-	When active (low), this bit allows the DMA controller to control the system data bus during direct memory access and write data into host memory. When inactive (high), a direct memory access will read data from system memory.
11	XA0	These bits are the four most significant bits of the 19-bit system memory address.
12	XA1	
13	XA2	
14	XA3	

The disk controller performs acknowledged ready direct memory access (ARMA) each time the memory address register is loaded by the firmware. Loading the host system memory address sets the DMABUSYQ flip-flop. The DMABUSYQ flip-flop is reset at the end of each memory access by the high-to-low transistion of

the MEMCYCQ signal. If either of the data registers is accessed by the local processor during a DMA cycle, the local processor is held in a wait state until the DMA cycle is complete. The DMA controller memory access timing is shown in Figure 4-5. The DMA logic equations are given in Table 4-8. The subscripts used in the table refer to the respective flip-flop inputs.

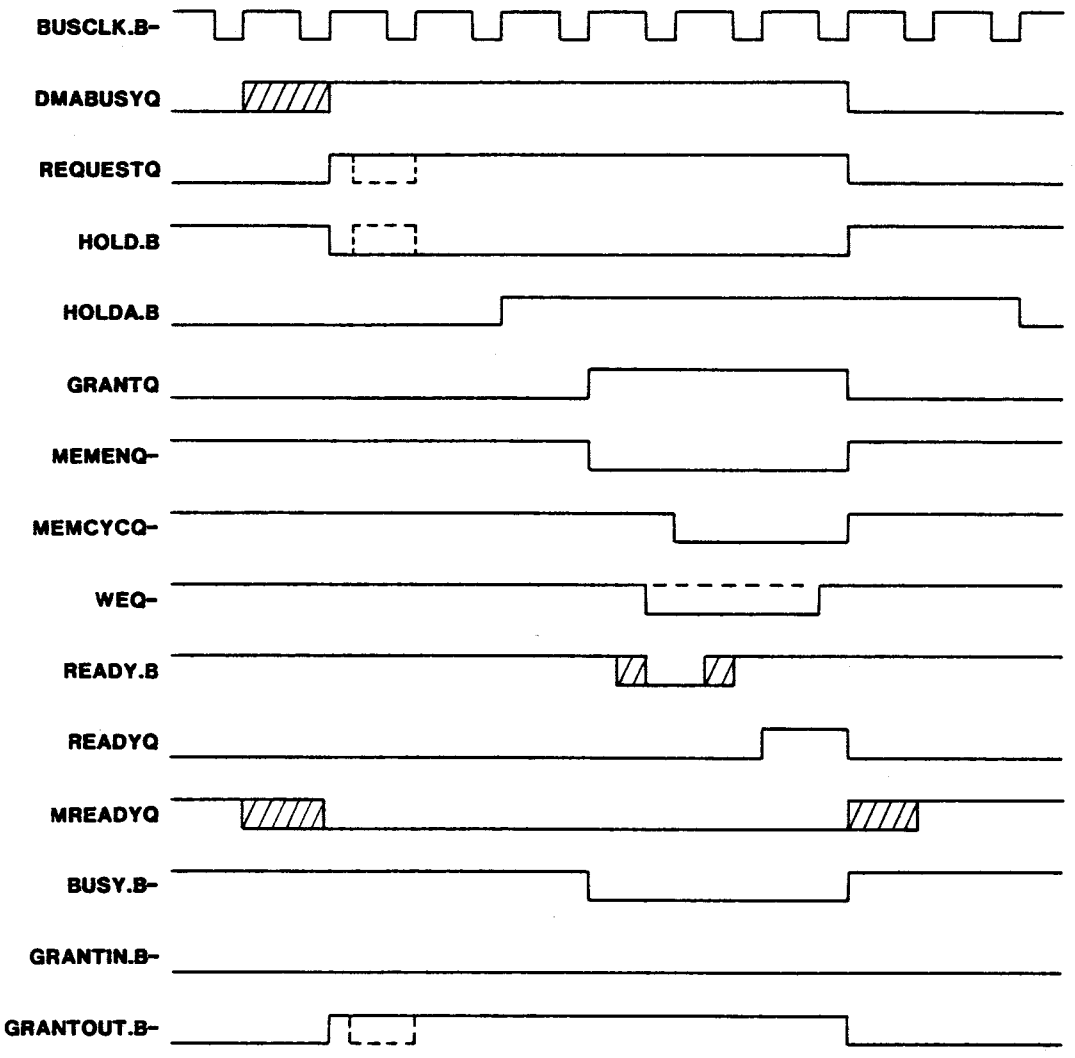


FIGURE 4-5. DMA MEMORY ACCESS TIMING (1 WAIT STATE)

TABLE 4-8. DMA CONTROLLER LOGIC EQUATIONS

$REQUESTQ_J$ = DMABUSYQ * (HOLD.B- + HOLDA.B)

$REQUESTQ_K$ = READYQ

$GRANTQ_{PRE-}$ = ALHOLDQ-

$GRANTQ_J$ = REQUESTQ * GRANTIN.B * HOLDA.B * BUSY.B-

$GRANTQ_K$ = READYQ * ALHOLDQ-

$MEMCYCQ_J$ = REQUESTQ * GRANTQ

$MEMCYCQ_K$ = READYQ

$READYQ_J$ = REQUESTQ * GRANTQ * READY.B

$READYQ_K$ = READYQ

WEQ_J = REQUESTQ * GRANTQ

WEQ_K = READYQ

WEQ_{CLR-} = WRITEQ

MEMENQ = REQUESTQ * GRANTQ

GRANTOUT.B- = (GRANTIN.B * REQUESTQ-)-

HOLD.B- = (REQUESTQ + GRANTQ)-

BUSY.B- = GRANTQ-

The DMA interface also contains the automatic bootload latch ALHOLDQ. This latch is set during system powerup by either the PRES.B- signal or by an on-board RC timing circuit. The automatic bootload can be disabled by jumpers J1 and J2. When the ALHOLDQ latch is set, the disk controller activates the HOLD.B- signal thus disabling the host processor. The ALHOLDQ latch remains active until reset by the firmware's activation of the SYSINT- signal. The HOLD.B- signal remains active until the end of the next DMA cycle. The automatic bootload timing is shown in Figure 4-6. The processor memory timing with wait states is shown in Figure 4-7.

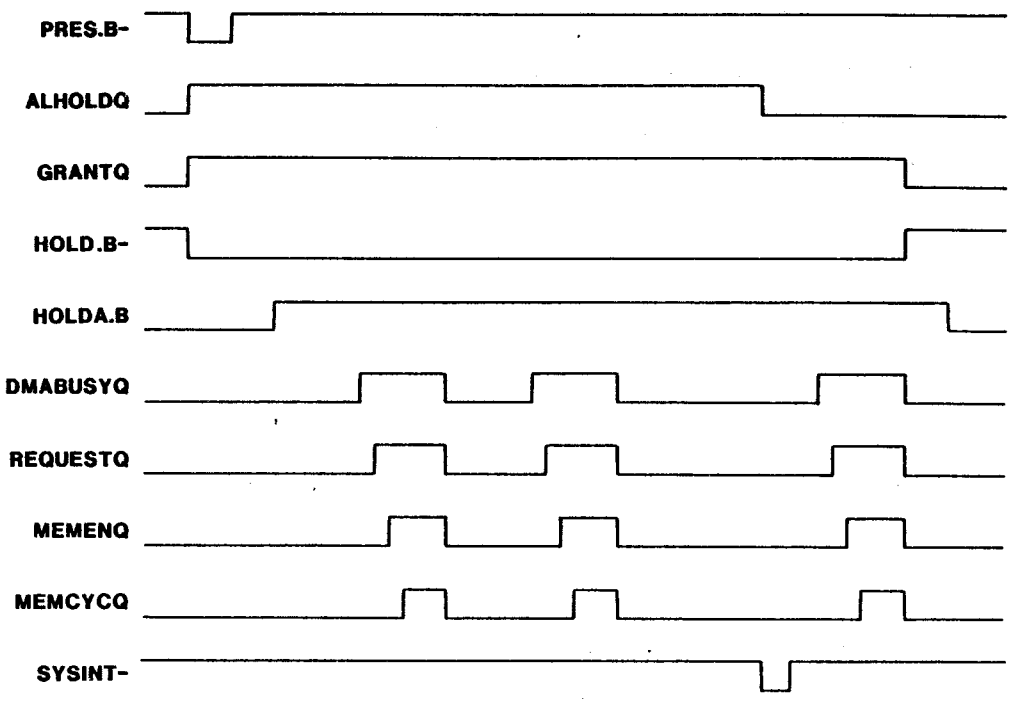


FIGURE 4-6. DMA TIMING - AUTOMATIC BOOTLOAD

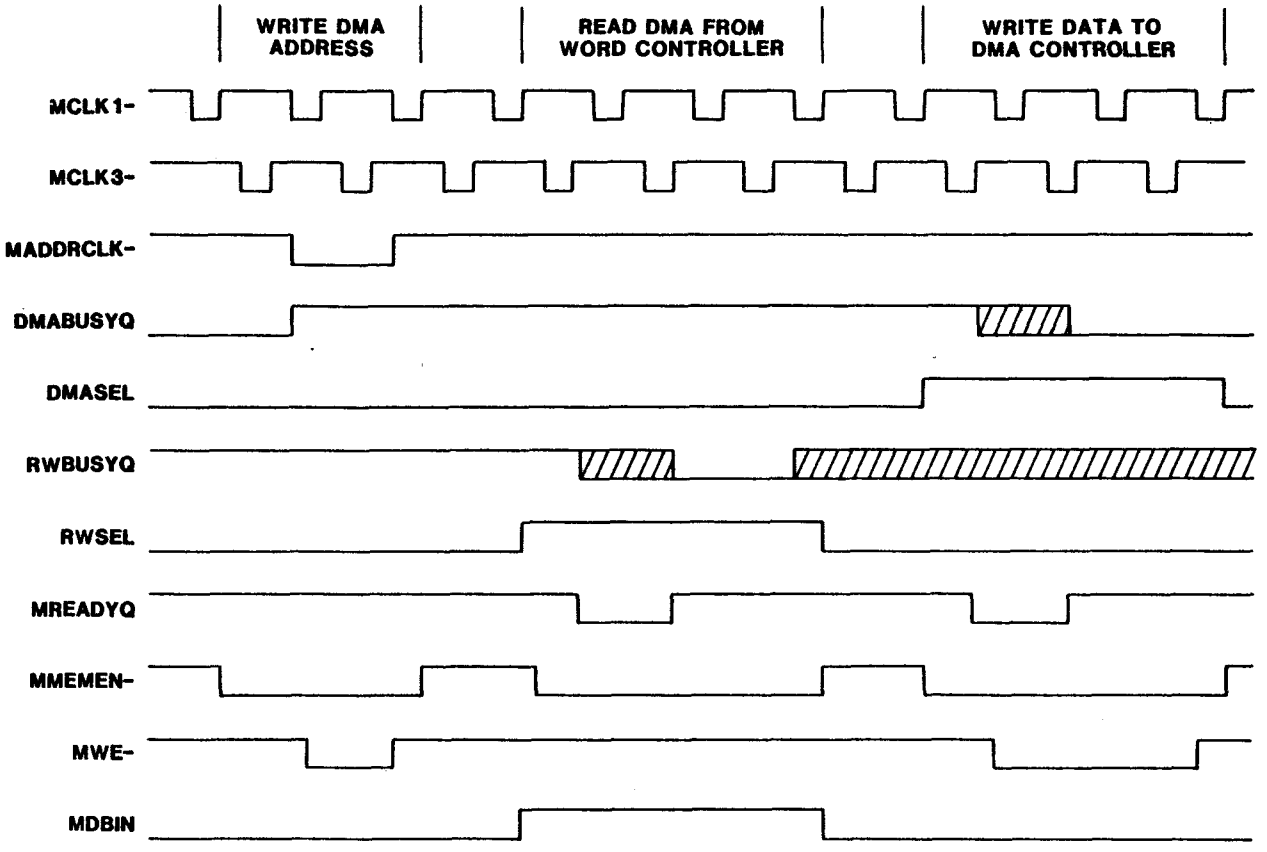


FIGURE 4-7. PROCESSOR MEMORY TIMING WITH WAIT STATES

4.7 READ/WRITE CONTROLLER

The read/write controller provides the interface between the local processor memory bus and the disk drive serial data stream. A block diagram of the read/write controller showing the major components is given in Figure 4-8. The read/write controller contains a phase-locked loop (PLL) for data synchronization, a PROM-based bit controller, a synchronization and precompensation decode PROM, a flip-flop for data and clock separation, a PROM-based word controller, a 16-bit data shift register, a cyclic redundancy check (CRC) function, and a control register.

The read/write controller is a memory-mapped I/O device on the local processor memory bus. The read/write controller local memory address space is from 8000_{16} to $BFFE_{16}$. The local processor memory address is used to control the operation of the read/write controller. The read/write address bit functions are given in Table 4-9. These bits (MA7 through MA14) are loaded into the control register during a memory write operation to the read/write controller. The control register configures the bit controller, word controller, and the data path for read or write operations.

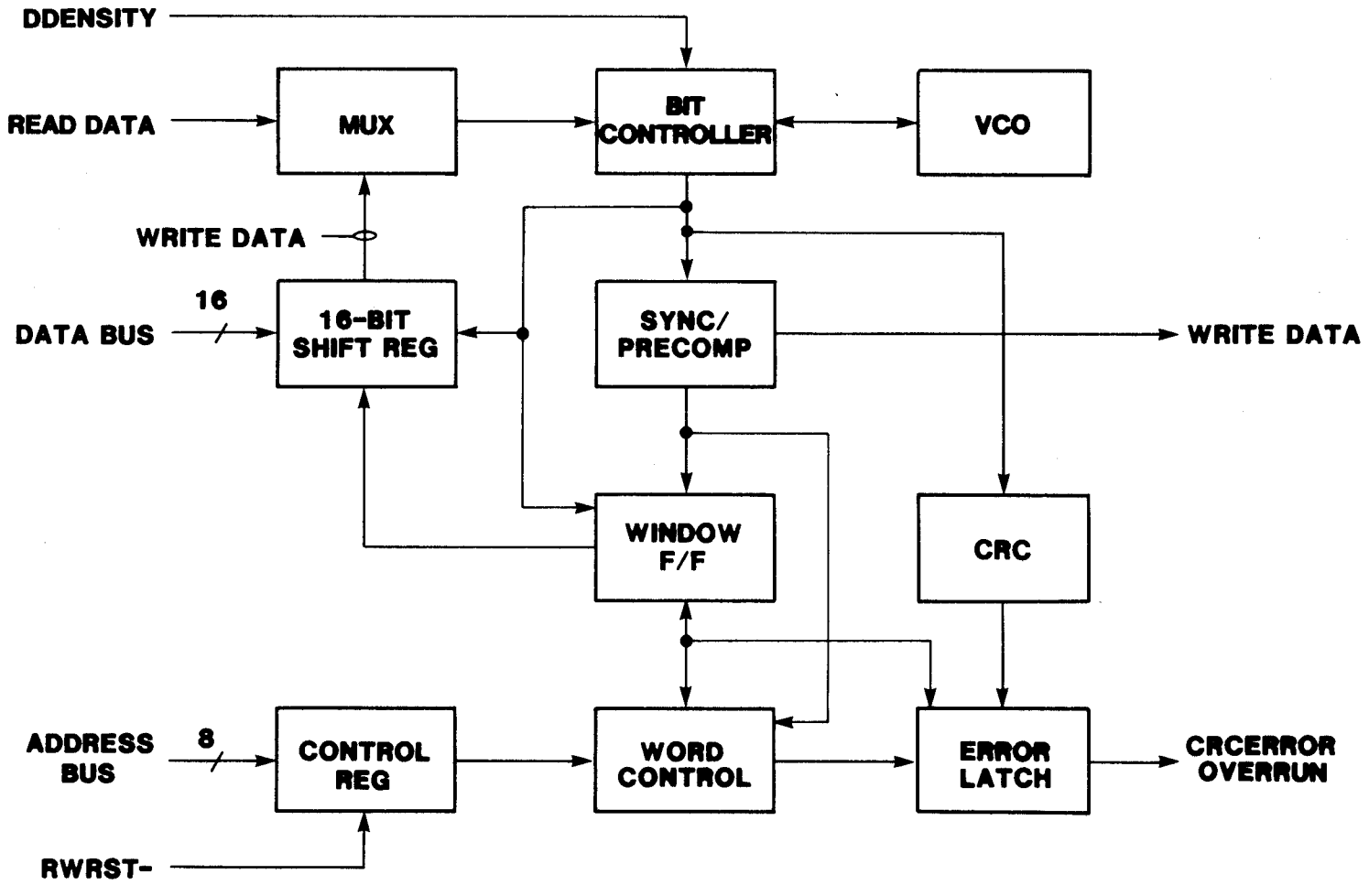


FIGURE 4-8. READ/WRITE CONTROLLER BLOCK DIAGRAM

TABLE 4-9. READ/WRITE CONTROLLER BITS

Base Address When Writing To Read/Write Controller Is $A000_{16}$		
Address Bit	Signature	Function
7	RESET-	When active (low), RESET- clears OVERRUNQ and CRCERRORQ flip-flops.
8	EXPROM	This is not used.
9	READENABLE	When active (high), READENABLE allows the read/write controller to search for address marks and assemble data words.
10	BCREAD	When active (high), BCREAD enables the phase-lock loop to lock onto read data. When inactive the phase-lock loop locks onto the crystal oscillator.
11	READ	When active (high), READ sets the data path multiplexer and data separator to the read mode.
12	WRITEMARK	When active (high), WRITEMARK enables the read/write controller to write an 8-bit address mark using the most-significant data byte as the clock pattern (0 = delete clock, 1 = enable clock).
13	ACCUMCRC	When active (high), ACCUMCRC enables the CRC generator to accumulate the CRC check word and selects the data shift register to write data. When inactive, the CRC check word is shifted to write data.
14	PRESETCRC-	When active (low), PRESETCRC- initializes the CRC check word to all ones.

4.7.1 Read/Write Data Path

The read/write data path is shown in Figure 4-9. The data path is configured for read or write operations by the control register. Read operations use all blocks in the data path except the write data multiplexer and the precompensation shift register. Write operations use all blocks in the data path.

Write data is selected through the write data multiplexer as shown in Table 4-10. Write data is selected from either 8-bit section of the 16-bit data shift register or from the CRC generator. While an address mark is being written, the deleted clock pattern is contained in the least-significant half of the data shift register. The bit controller selects data or clock information through the write data multiplexer as required for data encoding.

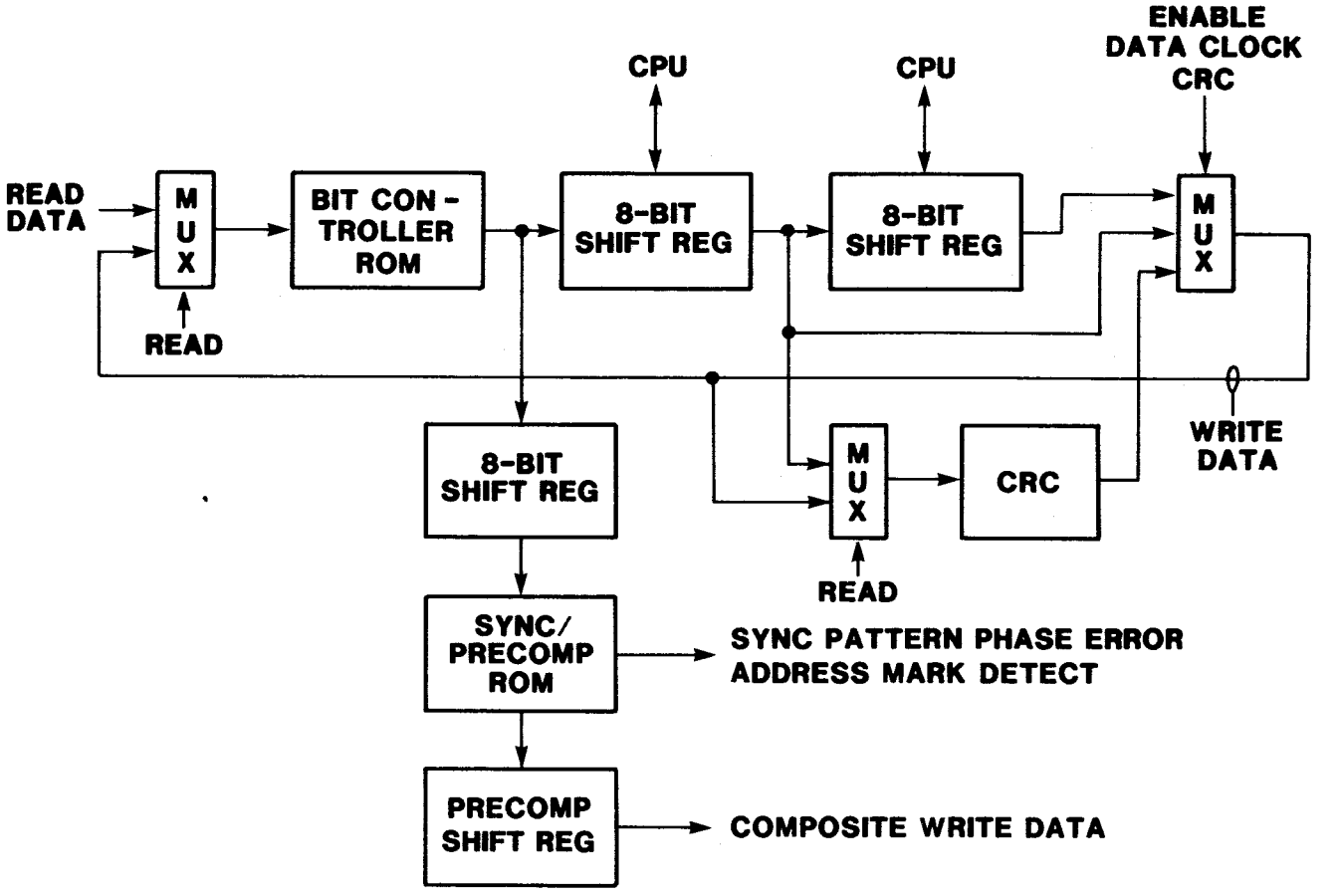


FIGURE 4-9. READ/WRITE DATA PATH

TABLE 4-10. WRITE DATA MULTIPLEXER
(74LS151)

Select Input			Signal Routed to Output	
UPQ(SDWINDOW) SEL C	ACCUMCRC SEL B	WRITEMARK SEL A	Output Y	Output W
0	0	0	CRCOUT	CRCOUT-
0	0	1	CRCOUT	CRCOUT-
0	1	0	SRDATAMSB	SRDATAMSB-
0	1	1	SRDATAMSB	SRDATAMSB-
1	0	0	1	0
1	0	1	SRDATALSB	SRDATALSB-
1	1	0	1	0
1	1	1	SRDATALSB	SRDATALSB-

NOTE: SELECT- is held low.

The data path is configured for read or write operation by the READ multiplexer (shown in Table 4-11) and by the BCREAD multiplexer (shown in Table 4-12). The BCREAD multiplexer also controls the VCO phase detector inputs which causes the VCO to synchronize to either read data or the local processor clock.

TABLE 4-11. READ MULTIPLEXER

Output Signal	Signal Selected When READ = 0	Signal Selected When READ = 1
SETCRC-	PRESETCRC-	RW3Q(READSYNCQ-)
WINDOWJ	UPQ(SDWINDOW)	PHASEGOOD
DATAQ-	WRITE DATA MUX FALSE OUTPUT	RDATAQQ-
CRCIN	WRITE DATA MUX TRUE OUTPUT	SRDATALSB

TABLE 4-12. BCREAD MULTIPLEXER

Output Signal	Signal Selected When BCREAD = 0	Signal Selected When BCREAD = 1
UPCLK	XTAL6MHZ	UPQ(SDWINDOW)
DNCLK	VCO6MHZ	DNQ(PCLOADQ)
CLOCKSHIFT	Logic ONE	CLOCKWINDOW
BCTRIN	MA12	RW1Q(RBCTRINQ)

4.7.2 Bit Controller

The bit controller, shown in Figure 4-10, provides read data bit synchronization and write data encoding. The bit controller provides inputs to the PLL phase detector during reading and provides a synchronized data and clock stream BC5Q(SDATAQ) to the data path. Data and clock information is not separated by the bit controller. A half-bit-cell clock BC4Q(HBCLK) is provided to the window flip-flop which performs the data separation function. During writing, the bit controller encodes the write data into a data and clock stream BC5Q(SDATAQ), sets the window flip-flop to the correct state using BC4Q(HBCLKQ) and UPQ(SDWINDOW), and provides a load signal DNQ(PCLOADQ) for the precompensation shift register.

The phase-locked loop (PLL), shown in Figure 4-11, consists of a phase detector, low-pass filter, voltage-controlled oscillator (VCO), a clock rate multiplier, the bit controller, and part of the BCREAD multiplexer. During write operations, the PLL synchronizes the VCO to a 6-MHz clock from the local processor. During read operations, the bit controller synchronizes the VCO to the incoming data stream producing a nominal 6-MHz clock which tracks the read data.

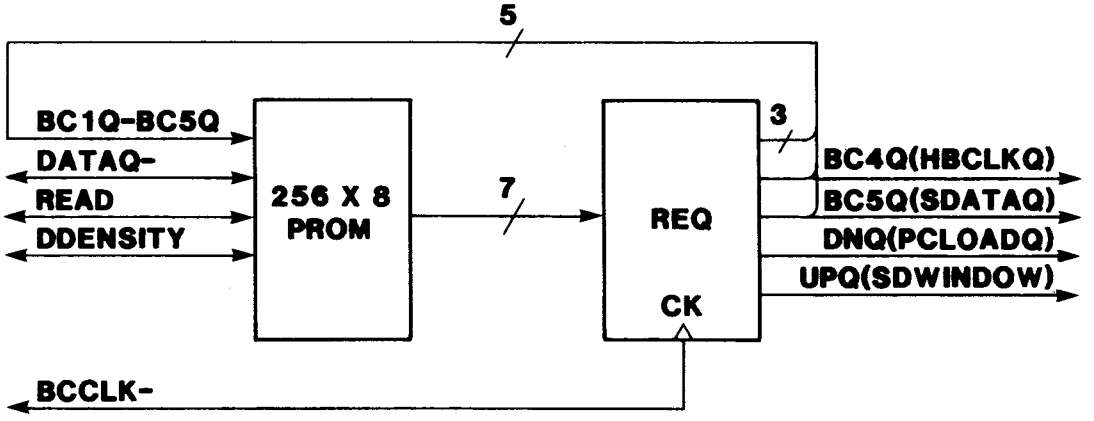


FIGURE 4-10. BIT CONTROLLER BLOCK DIAGRAM

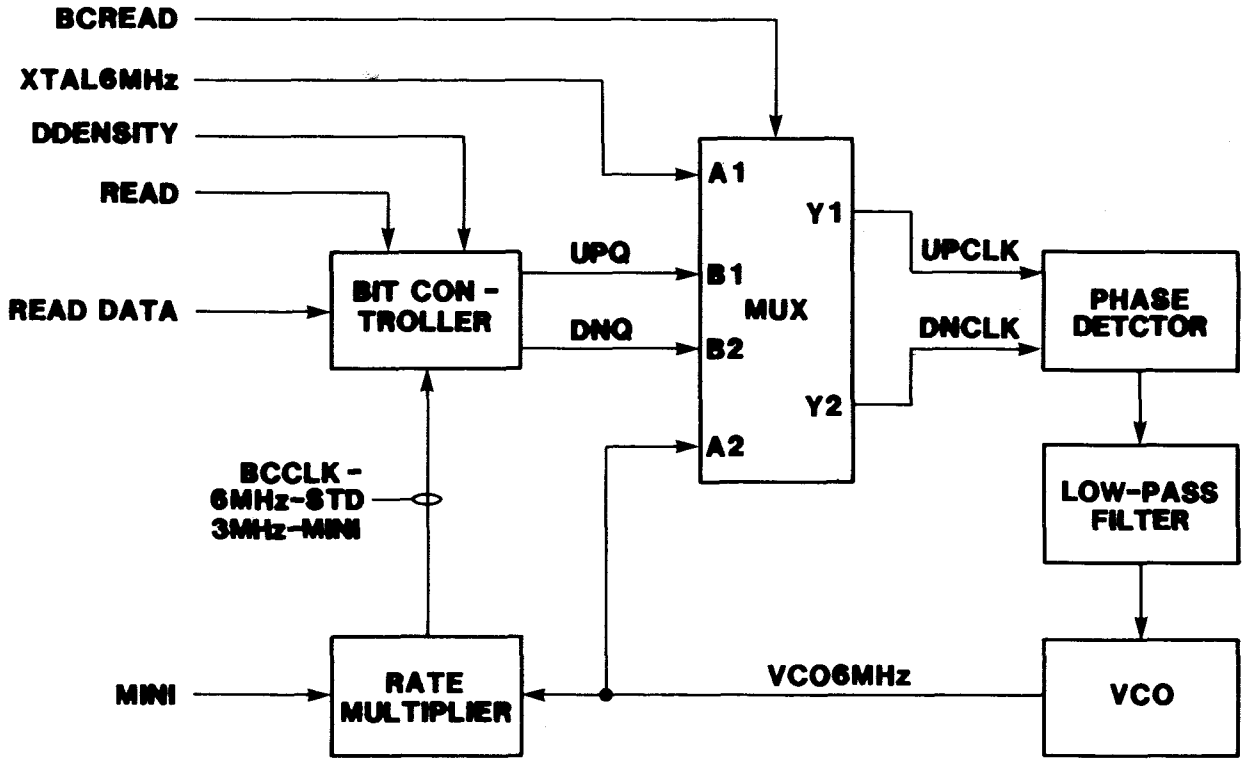


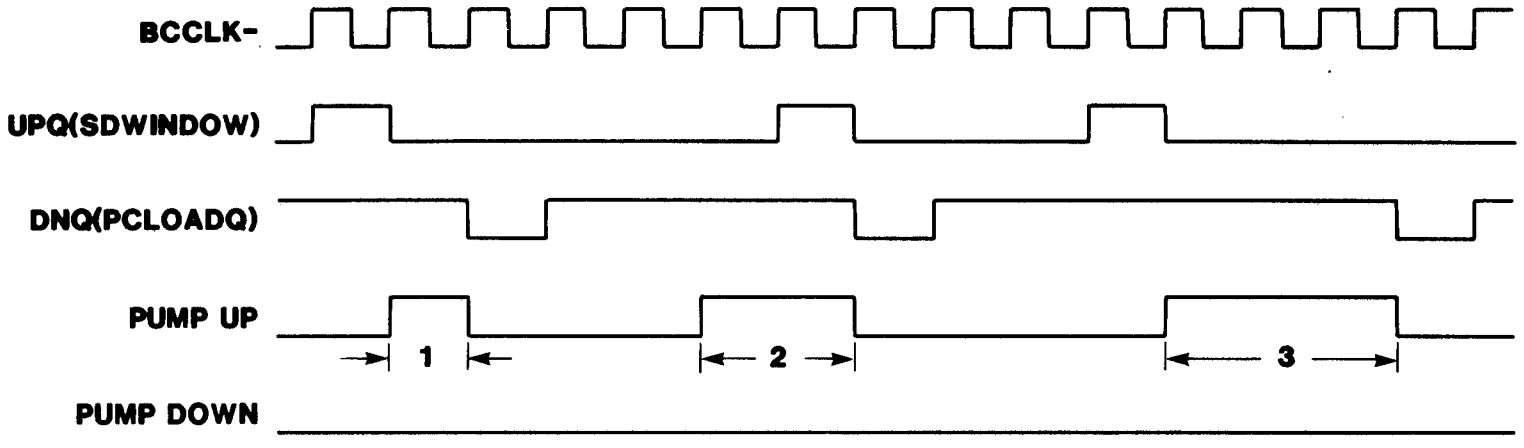
FIGURE 4-11. PHASE-LOCKED LOOP BLOCK DIAGRAM

The VCO output is used directly (6 MHz) for standard-size diskettes or divided by two (3 MHz) for mini-sized diskettes by the rate multiplier. The read/write controller operates synchronous to the VCO output clock.

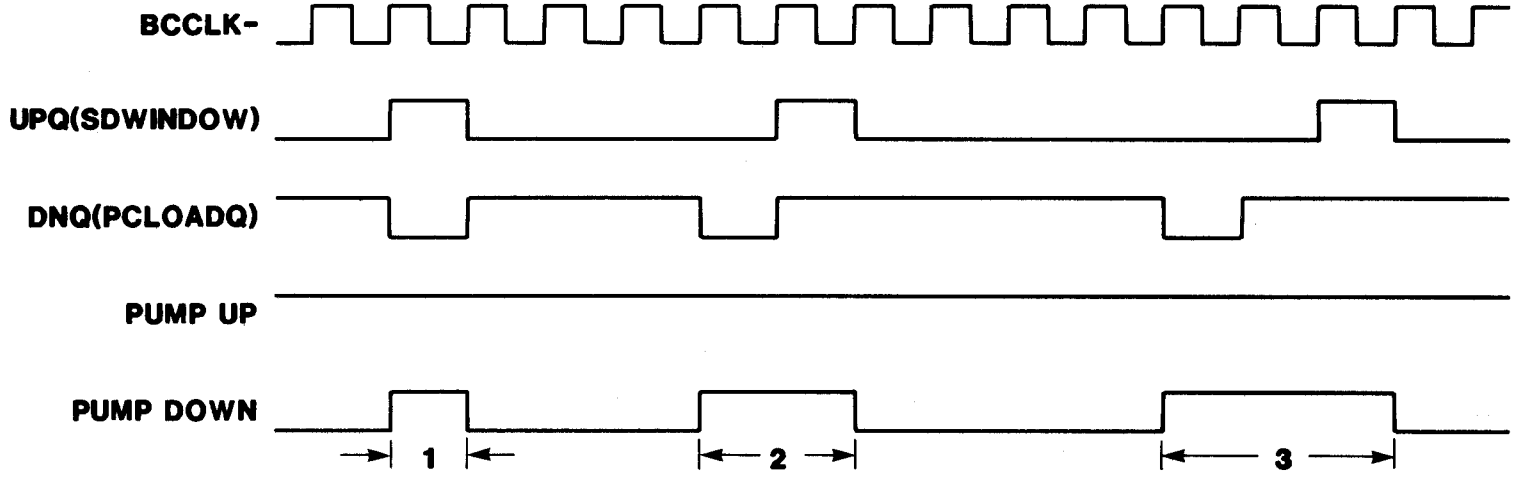
Typical phase detector timing for double-density read operations is shown in Figure 4-12. The phase detector inputs, UPQ(SDWINDOW) and DNQ(PCLOADQ) are used to cause phase corrections of 1, 2, or 3 bit controller clock periods. Note that the phase detector has negative-edge triggered inputs.

The bit controller state diagrams for writing single density (FM) and double density (MFM) are shown in Figures 4-13 and 4-15. The state transitions occur vertically from top to bottom of the diagram except as shown. The bit controller timing diagrams for writing FM and MFM are shown in Figures 4-14 and 4-16.

The bit controller state diagrams for reading FM and MFM are shown in Figures 4-17 and 4-18. In the absense of a data pulse from the disk drive, the state transitions occur vertically from top to bottom of the figure. If a data pulse occurs, the horizontal state transition path (if shown) is taken and the indicated phase correction is made (E^1, E^2, etc.) using UPQ(SDWINDOW) and DNQ(PCLOADQ)



(a) Data Occurs Early in Window



(b) Data Occurs Late in Window

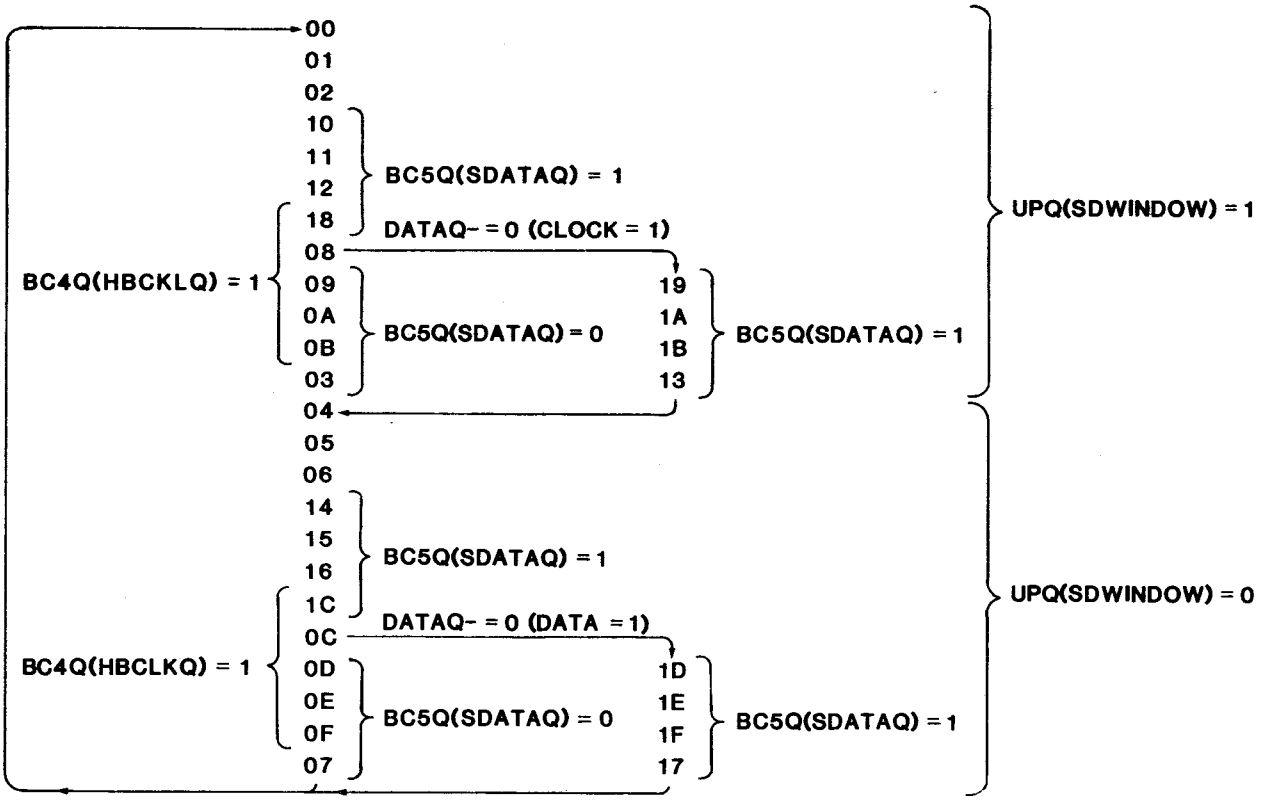


FIGURE 4-13. BIT CONTROLLER WRITE FM STATE DIAGRAM

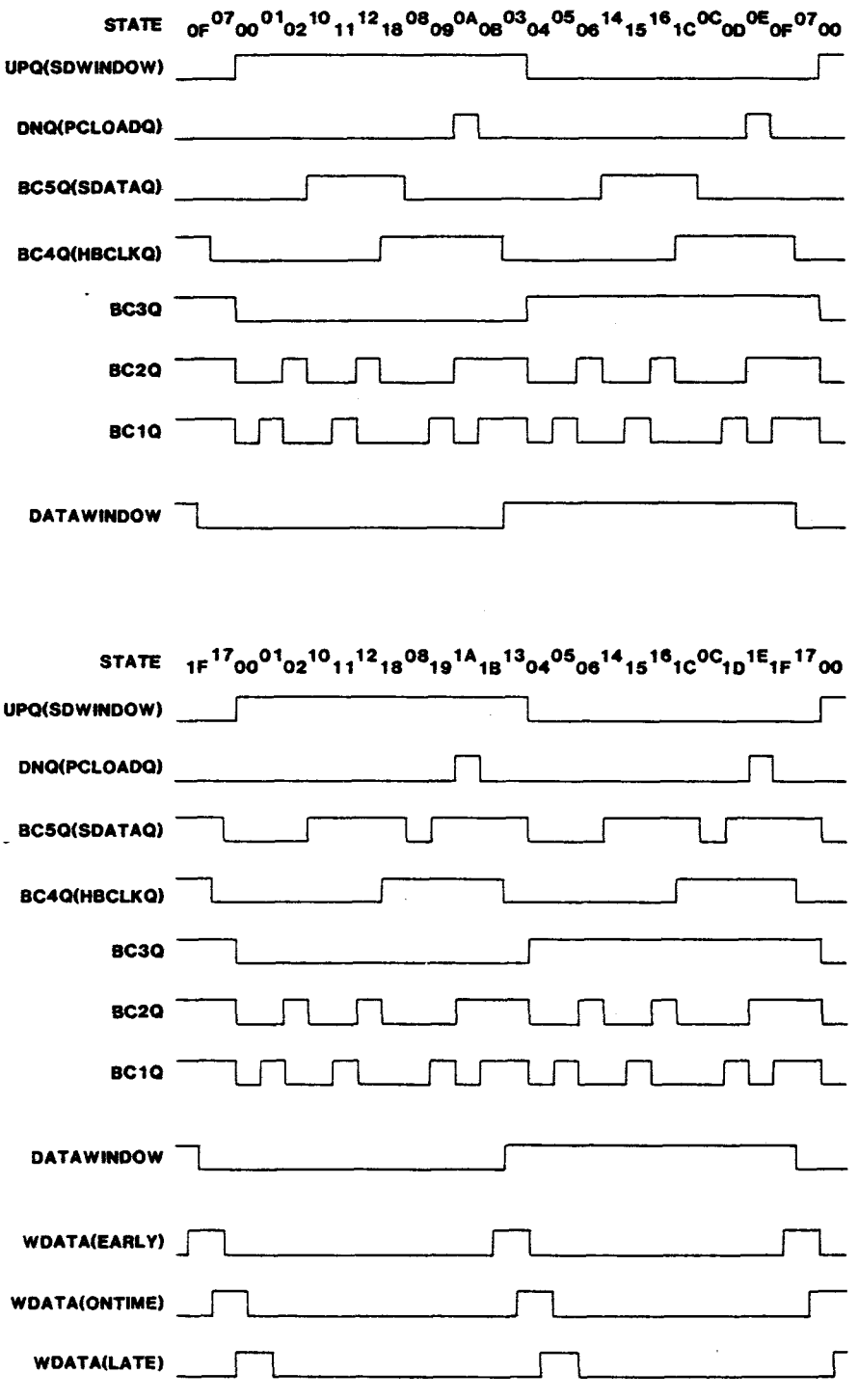


FIGURE 4-14. WRITE FM TIMING

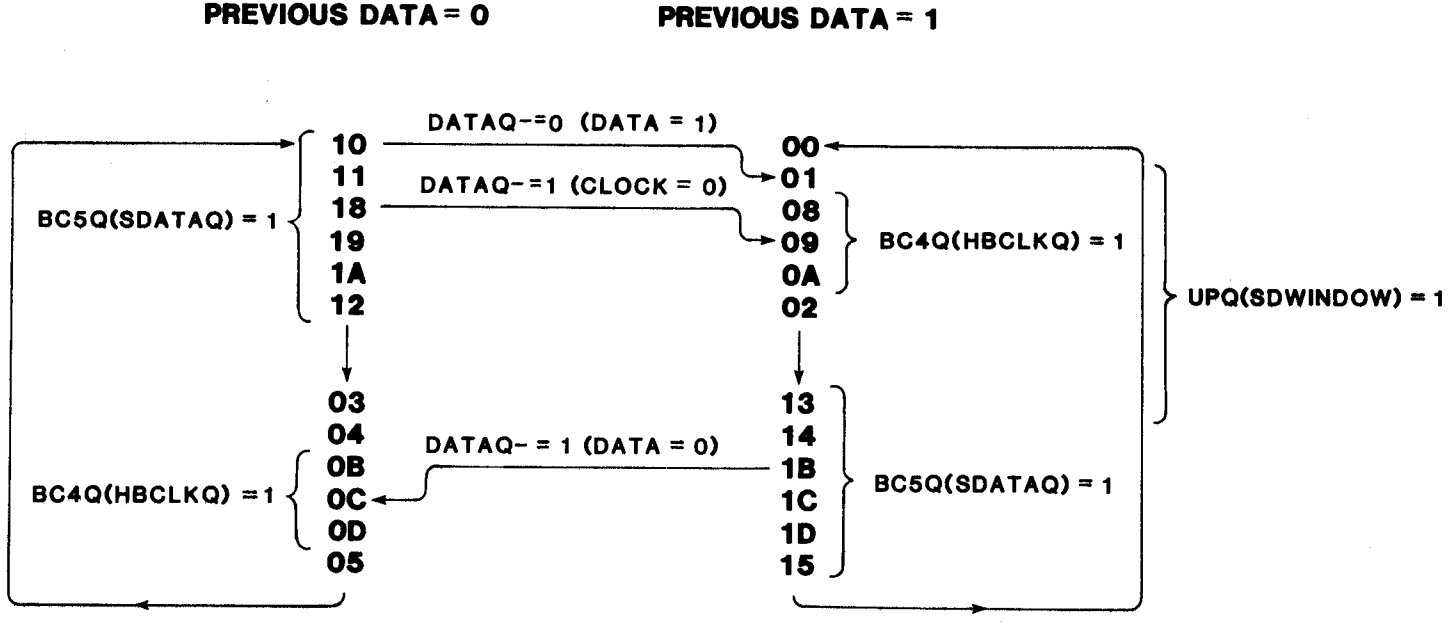


FIGURE 4-15. BIT CONTROLLER WRITE MFM STATE DIAGRAM

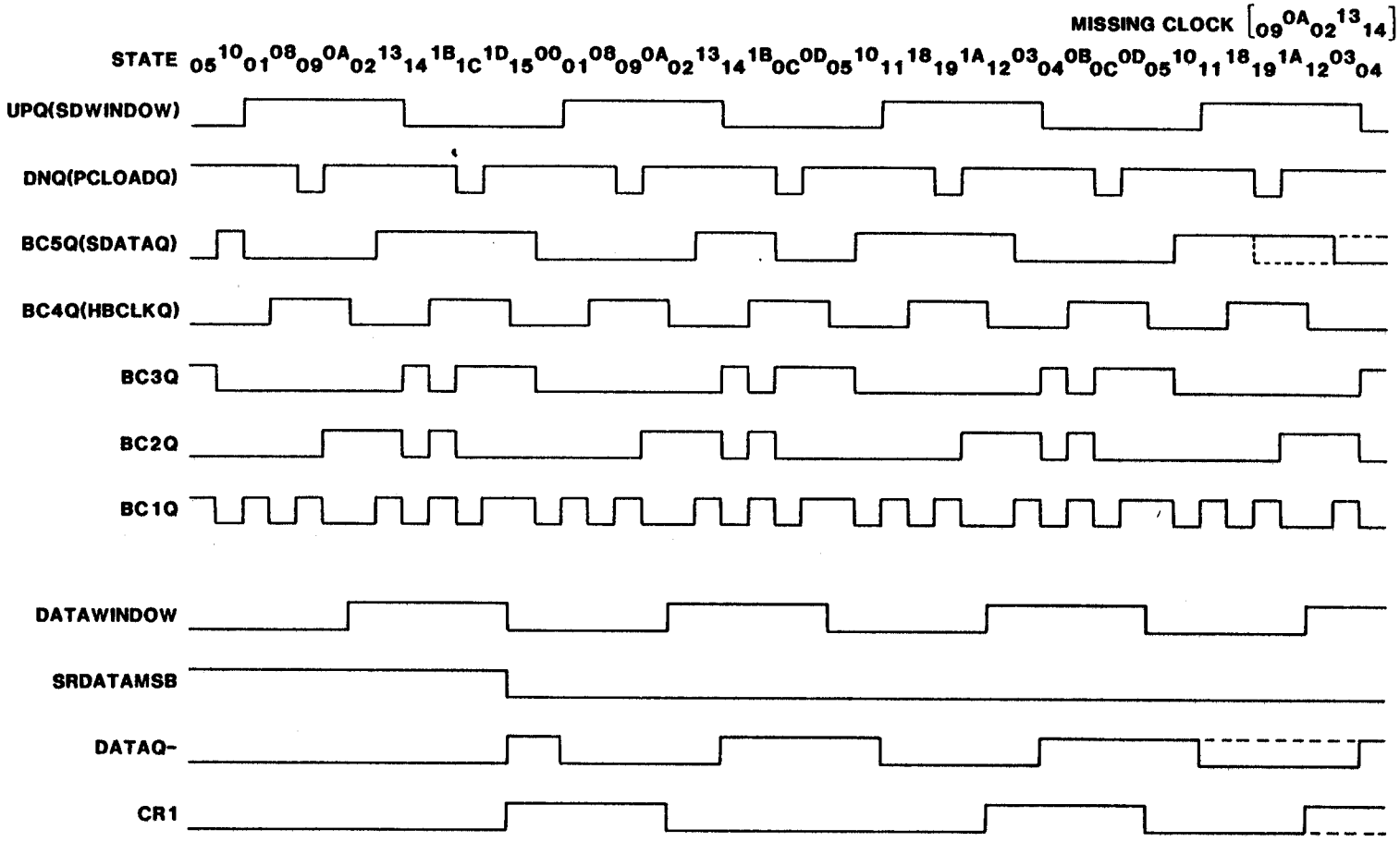


FIGURE 4-16. WRITE MFM

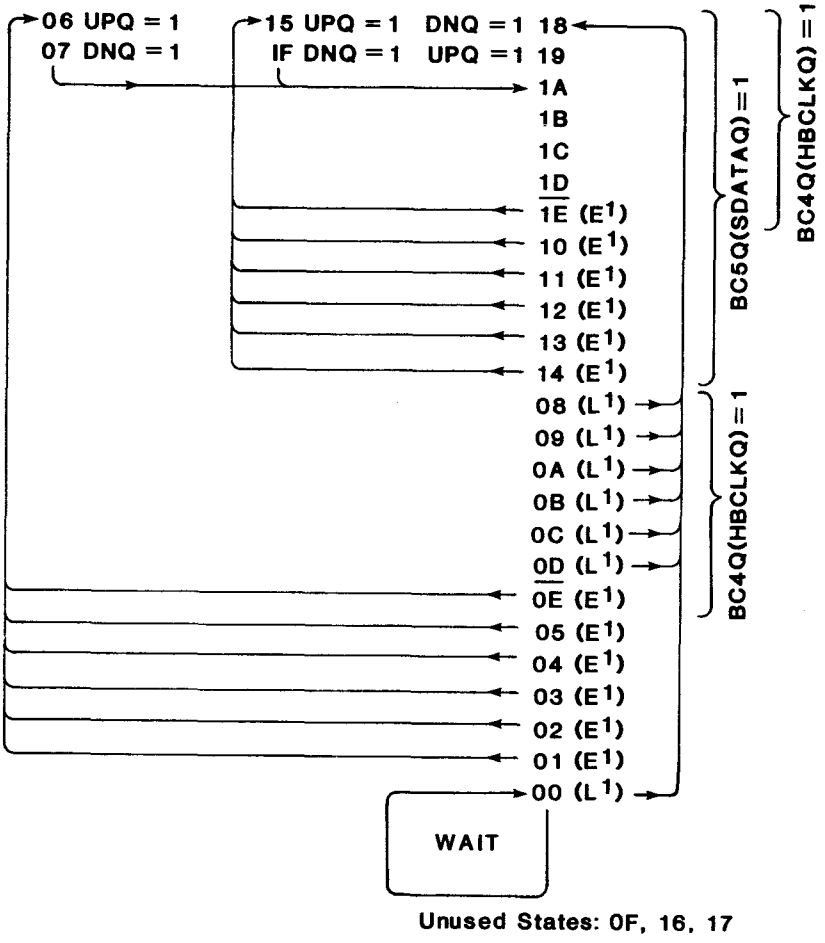


FIGURE 4-17. BIT CONTROLLER READ FM

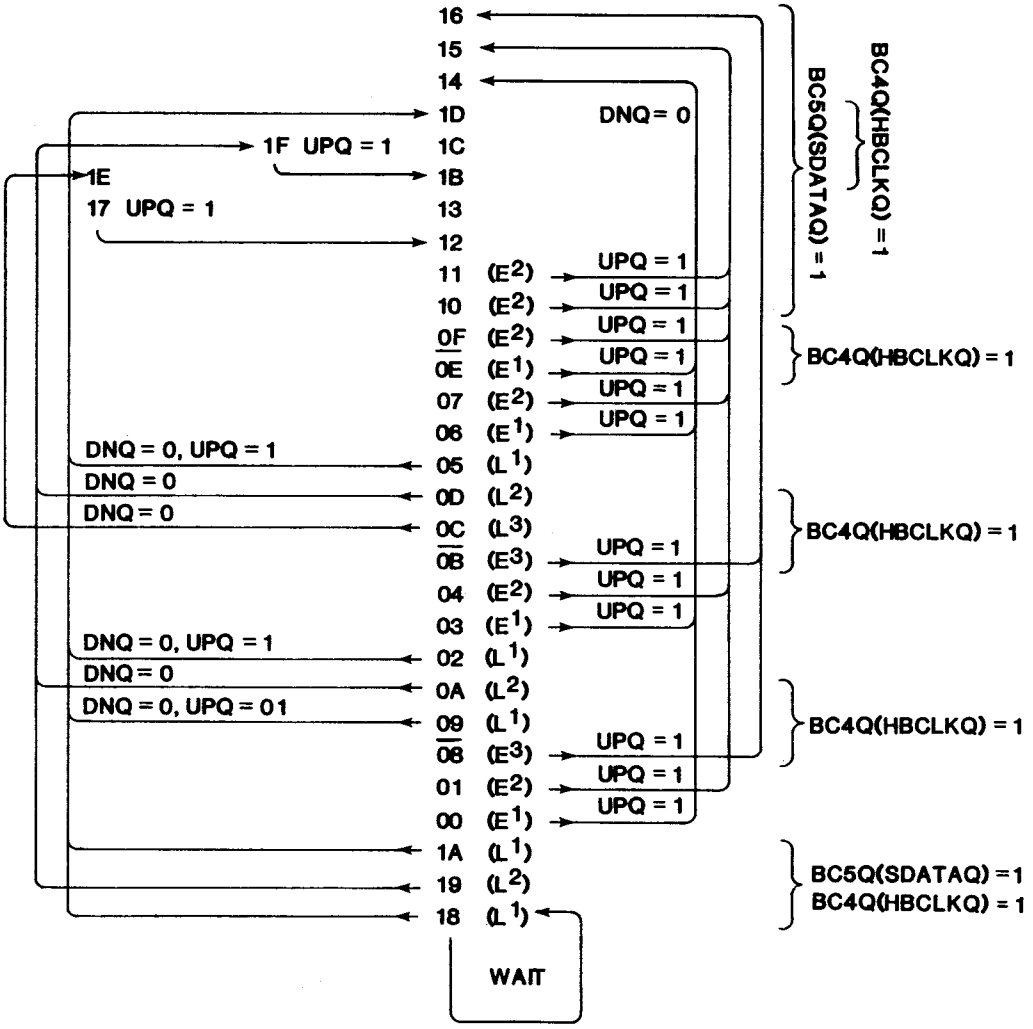


FIGURE 4-18. READ MFM

4.7.3 Synchronization and Address Mark Detection

Synchronization and address mark patterns are recorded on the disk preceding ID and data fields. Synchronization patterns allow the phase-locked loop to lock into sync with the incoming data and provide a data and clock reference for bit synchronization. Address marks provide a reference for byte synchronization. Address marks are unique bit patterns which violate the encoding rules by deleting clock bits. The synchronization and address mark patterns for various recording formats are given in Table 4-13.

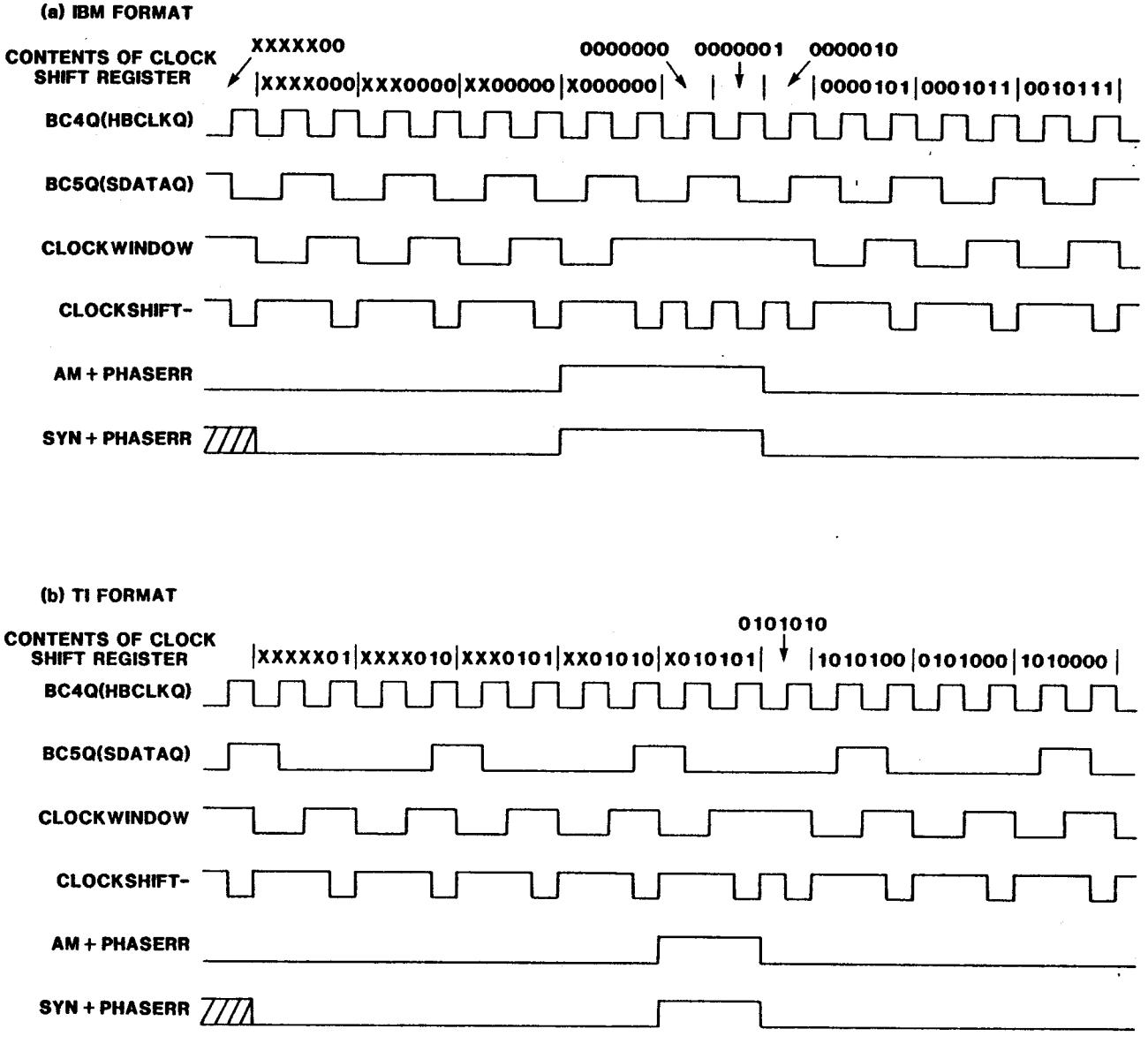
The synchronization and address mark patterns are detected by the sync/precompensation PROM. The incoming clock bit pattern is shifted into an 8-bit shift register and decoded by the PROM. If a phase error condition exists during a synchronization pattern, the window flip-flop is held in the clock window state until the phase is correct. This process is shown in Figure 4-19. The sync detect, address mark detect, and phase error conditions are encoded into the AM+PHASERR and SYN+PHASERR signals for use by the word controller.

TABLE 4-13. SYNCHRONIZATION AND ADDRESS MARK PATTERNS

PATTERN	IBM-FM FORMAT	IBM-MFM FORMAT	TI-MFM FORMAT
Sync Data	00000000	00000000	01010101
Sync Clock	11111111	11111111	00000000
Track AM Data	11111100	11000010	
Track AM Clock	11010111	00010100	
ID AM Data	11111110	10100001	00001010
ID AM Clock	11000111	00001010	01010000
Data AM Data	11111011	10100001	00001011
Data AM Clock	11000111	00001010	01010000
Delete AM Data	11111000	10100001	
Delete AM Clock	11000111	00001010	
Synchronization Phase Error Clock Pattern	XXX00000	XXX00000	XX010101

NOTE

IBM-MFM address marks are repeated 3 times and followed by the appropriate IBM-FM address mark data pattern and normal clock pattern.



Note: False address mark detection will be ignored by read/write controller.

FIGURE 4-19. PHASE ERROR RECOVERY TIMING DIAGRAM

4.7.4 Precompensation

Data precompensation is a method of compensating for bit shift caused by the magnetic media. The write data is shifted in the direction opposite of the expected shift due to the media prior to recording the data. The precompensation process is performed by two 8-bit shift registers and a PROM (a 74LS166 at U63) which is also used for synchronization and address mark detection.

Data and clock bits are shifted into an 8-bit shift register. The shift register parallel outputs are encoded by the sync/precompensation PROM. The precompensation patterns are given in Table 4-14. The PROM outputs are loaded into another 8-bit shift register and shifted by a 6 MHz clock to produce a 167-nanosecond precompensation. The output shift register timing is shown in Figure 4-20.

The four most significant bits of Word 0 of the Drive Parameter Table allow the user to specify where precompensation is to be applied on the diskette. Table 3-2, command 10_{16}, word 0 contains instructions for precompensation value setup.

TABLE 4-14. ROM-GENERATED PRECOMPENSATION PATTERNS

ROM INPUT SIGNAL								ROM OUTPUT SIGNAL					COMMENT
RFMTWPCOMP	CR1	CR2	CR3	CR4	CR5	CR6	CR7	PCE	PCE0	PC0	PCL0	PCL	
1	X	X	0	0	1	0	0	0	1	1	1	0	On-Time
1	X	X	0	0	1	0	1	0	0	0	1	1	Late
1	X	X	0	0	1	1	0	0	0	0	1	1	Late*
1	X	X	0	0	1	1	1	0	0	0	1	1	Late*
1	X	X	0	1	1	0	0	1	1	0	0	0	Early*
1	X	X	0	1	1	0	1	1	1	0	0	0	Early*
1	X	X	0	1	1	1	0	0	1	1	1	0	On-Time*
1	X	X	0	1	1	1	1	0	1	1	1	0	On-Time*
1	X	X	1	0	1	0	0	1	1	0	0	0	Early
1	X	X	1	0	1	0	1	0	1	1	1	0	On-Time
1	X	X	1	0	1	1	0	0	0	0	1	1	Late*
1	X	X	1	0	1	1	1	0	0	0	1	1	Late*
1	X	X	1	1	1	0	0	1	1	0	0	0	Early*
1	X	X	1	1	1	0	1	1	1	0	0	0	Early*
1	X	X	1	1	1	1	0	0	1	1	1	0	On-Time*
1	X	X	1	1	1	1	1	0	1	1	1	0	On-Time*
X	X	X	X	X	0	X	X	0	0	0	0	0	No Data
0	X	X	X	X	1	X	X	0	1	1	1	0	Disabled

NOTES

1. X is a don't care condition.

2. Patterns marked with an asterisk (*) do not occur for double-density (MFM) encoding.

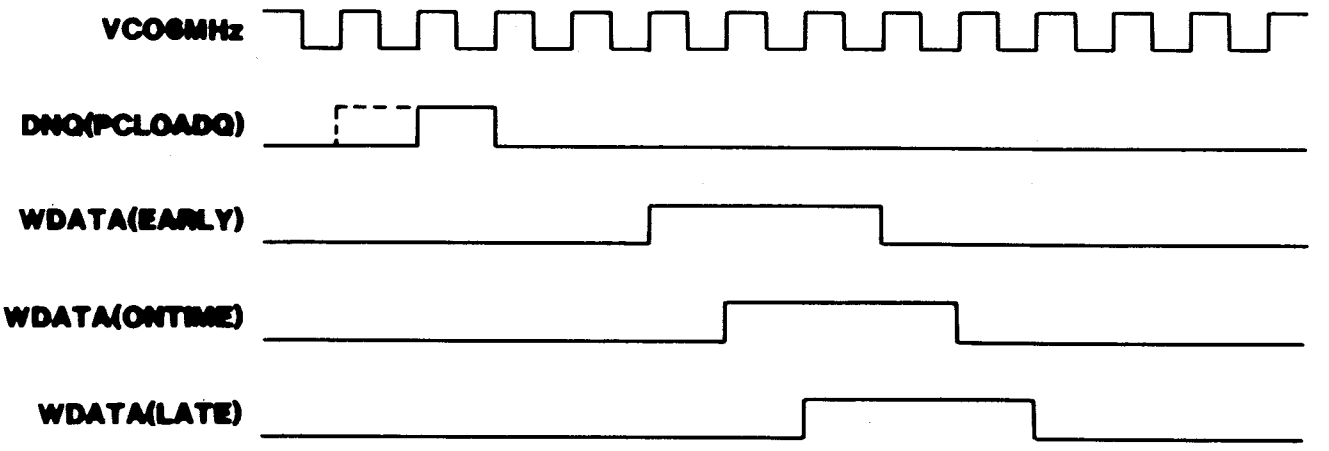


FIGURE 4-20. PRECOMPENSATION SHIFT REGISTER TIMING

4.7.5 Word Controller

The word controller, shown in Figure 4-21, provides data synchronization on a 16-bit word basis. During reading, the word controller operates as a PROM-based controller to check for a synchronization pattern followed by an address mark before establishing word synchronization. The word controller read mode flowchart is given in Figure 4-22. If a phase error is encountered prior to word synchronization, the word controller corrects the window flip-flop using the PHASEGOOD signal. During writing, the word controller operates as a logic array. The word controller logic equations are given in Table 4-15.

The word controller provides the interface between the read/write controller and the local processor. When the read/write controller is accessed as a memory-mapped I/O device, the word controller causes the local processor to enter a wait state until the data transfer occurs by using the RWBUSYQ signal. If the processor is not accessing the read/write controller at the time when a data transfer must occur, the OVERRUN latch is set.

The word controller uses a 4-bit counter to count data bits. During data transfer, the counter is cleared and sixteen data bits are counted. During address mark transfer, the counter is set to eight, and eight data bits are counted. When the counter carry occurs, data is transferred between the data shift register and the local processor. The word controller timing is shown in Figures 4-23, 4-24, and 4-25.

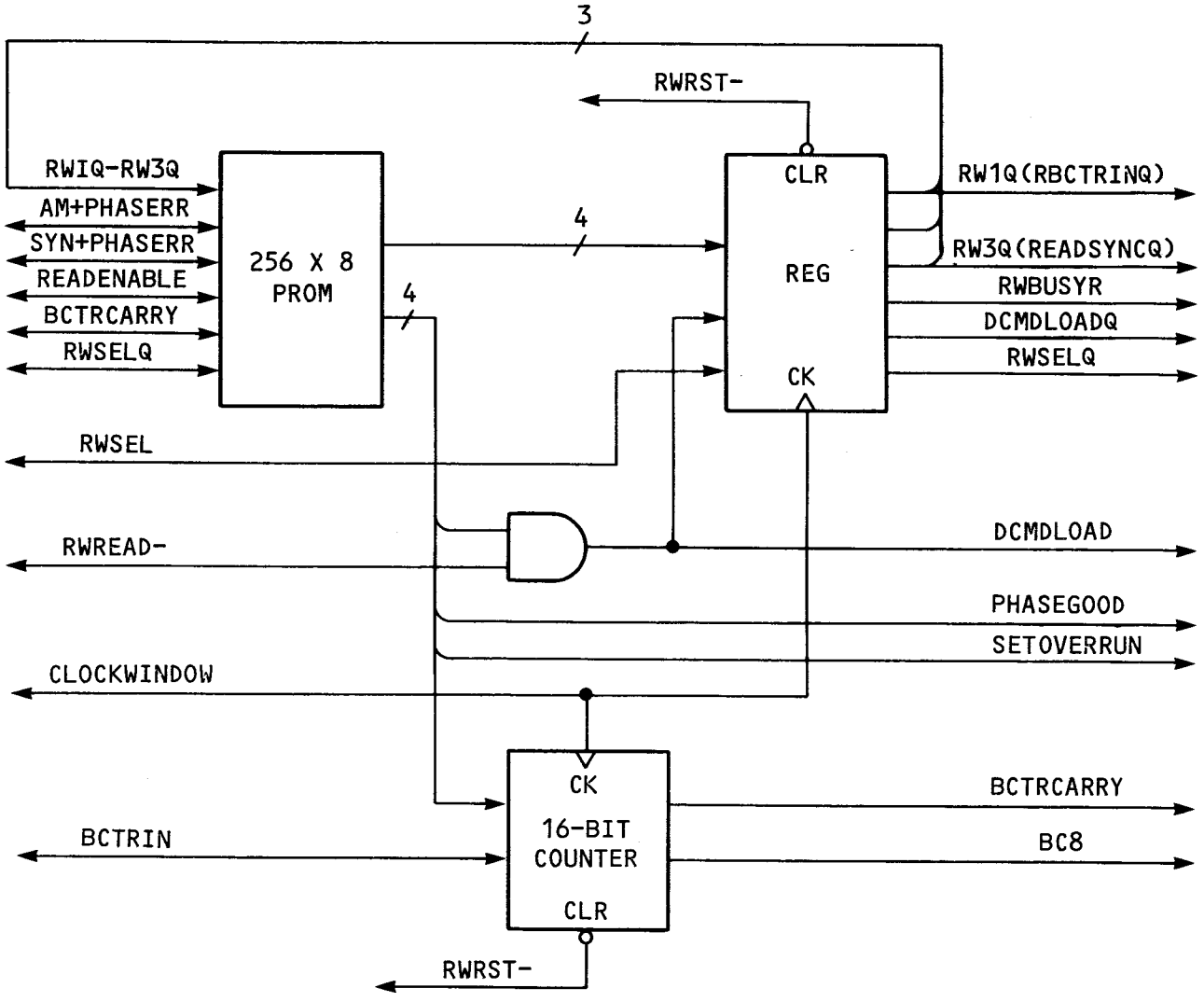
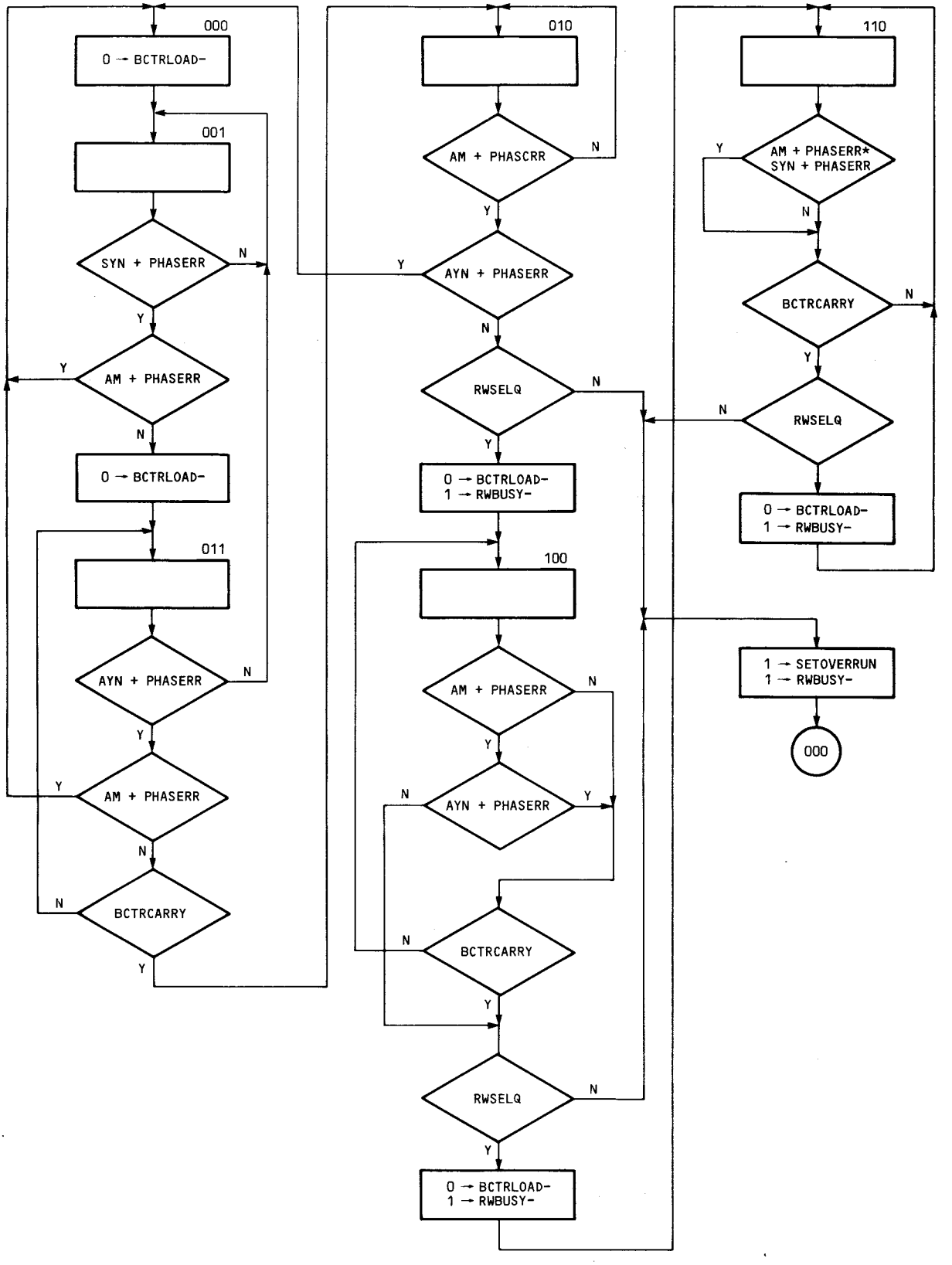


FIGURE 4-21. WORD CONTROLLER BLOCK DIAGRAM



NOTES

1. **Binary States (000 to 110) are shown atop some rectangular blocks; some blocks are left blank (used only to show state).**
2. **The final procedure block in the lower right is the last block to states 010, 100, and 110.**

FIGURE 4-22. WORD CONTROLLER READ MODE FLOWCHART

TABLE 4-15. WORD CONTROLLER LOGIC EQUATIONS

```
PHASEGOOD =  READENABLE * AM+PHASERR * SYN+PHASERR * ( 000 + 001 + 010 + 011 )

RWBUSY- = READENABLE * RWSELQ * BCTRLOAD + READENABLE- * RWSELQ * BCTRCARRY

SETOVERRUN = READENABLE * RWSELQ- * BCTRLOAD * + READENABLE- * RWSELQ-
             * BCTRCARRY

BCTRLOAD = READENABLE * ( 001 * SYN * AM- +  011  * BCTRCARRY * (SYN * AM)-
           + 010  * AM * SYN- +  100  * (BCTRCARRY + AM * SYN-) + 110 *
           BCTRCARRY) + READENABLE- * RWSELQ * BCTRCARRY

DCMDLOAD = RWBUSY- * RWREAD-
```

NOTE

Binary numbers (e.g. 011) represent the word controller state defined by the signals RW3Q, RW2Q, and RW1Q.

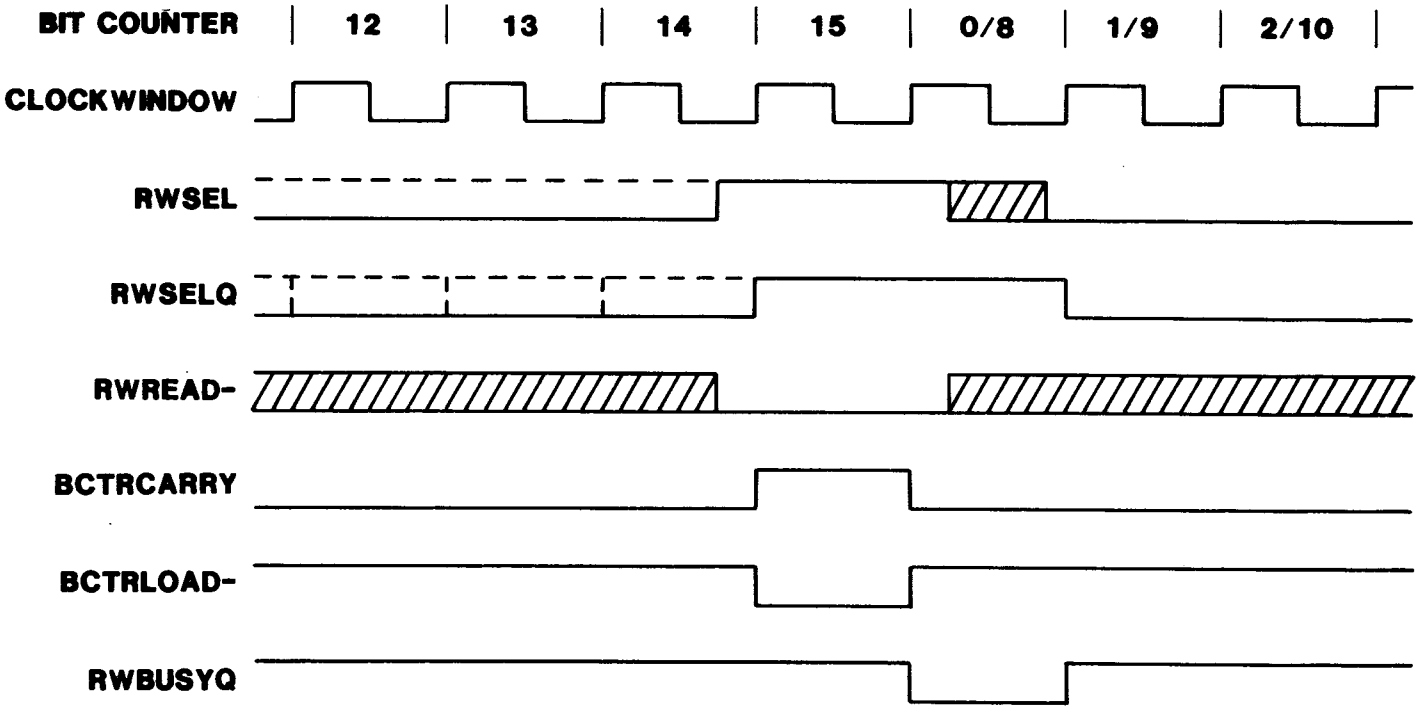


FIGURE 4-23. WORD CONTROLLER READ TIMING

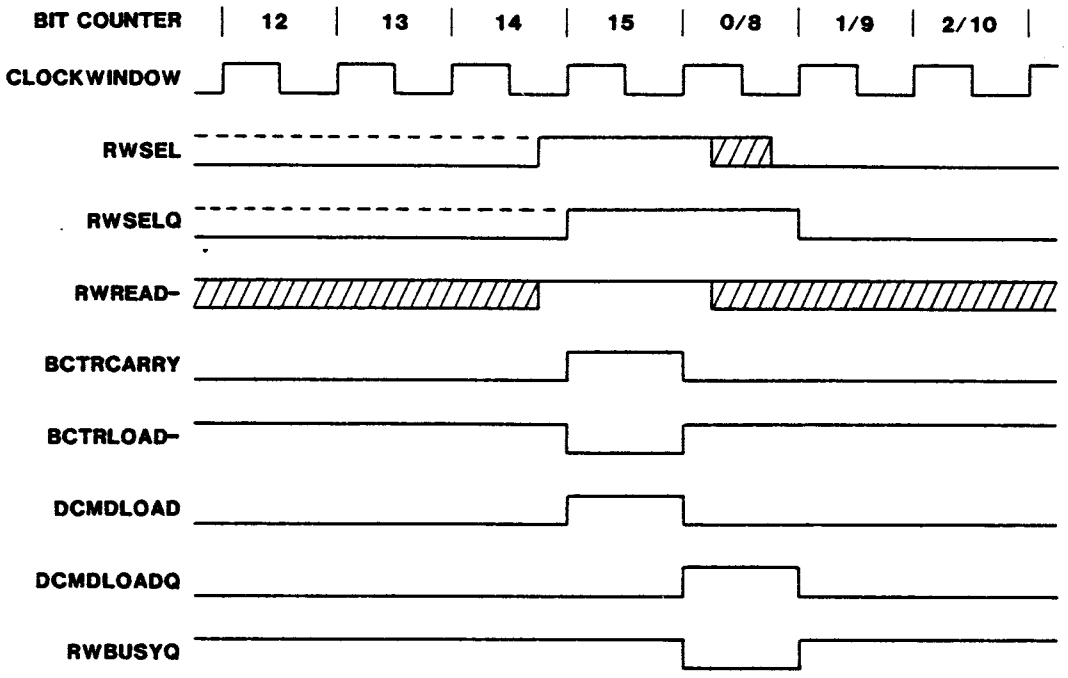


FIGURE 4-24. WORD CONTROLLER WRITE TIMING

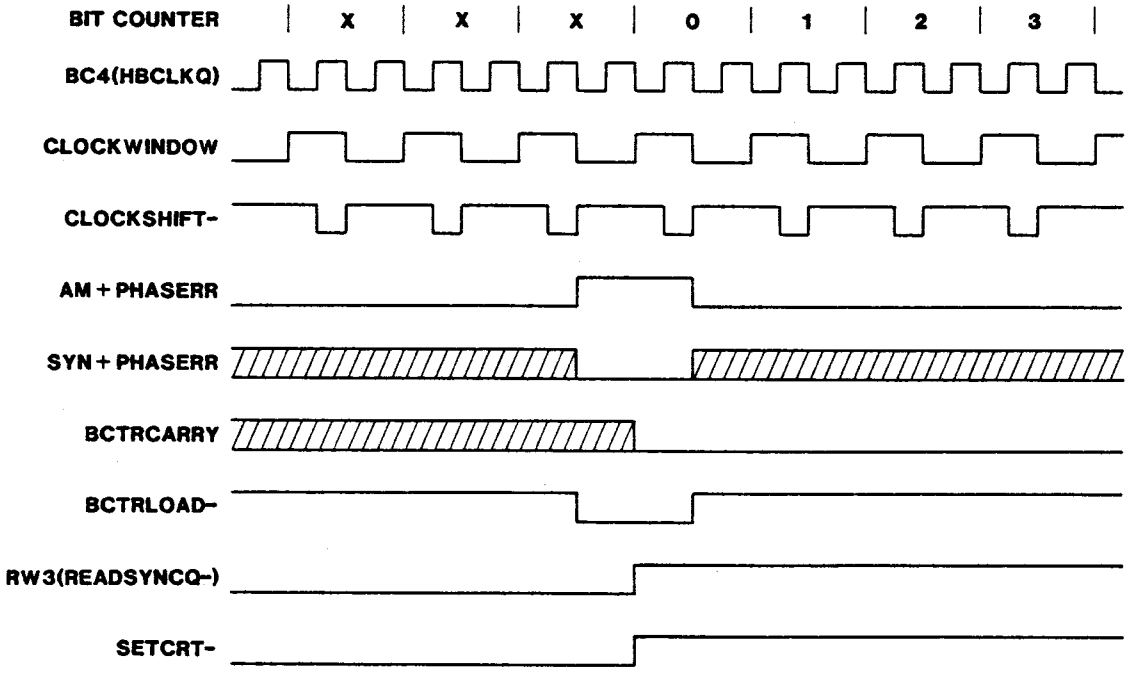


FIGURE 4-25. WORD CONTROLLER READ TIMING-ADDRESS MARK DETECT

4.7.6 Control of Read and Write Operations

Read and write operations are controlled by the CRU output bits DDENSITY and RFMTWPCOMP and by the control register. Writing an ID or data field is accomplished by changing the control register contents while loading the data into the read/write controller. The ID field write timing is shown in Figures 4-26 and 4-28. Reading an ID or data field is accomplished by using the control register to start the word controller. The word controller detects the address mark, synchronizes the data, and starts the CRC checking process. The ID field read timing is given in Figures 4-27 and 4-29.

The CRC check output is latched in the CRCERROR flip-flop and held valid for one 16-bit word time. The CRCERROR signal must be sampled during this time at the end of a read operation. The CRC error latch timing is shown in Figure 4-30.

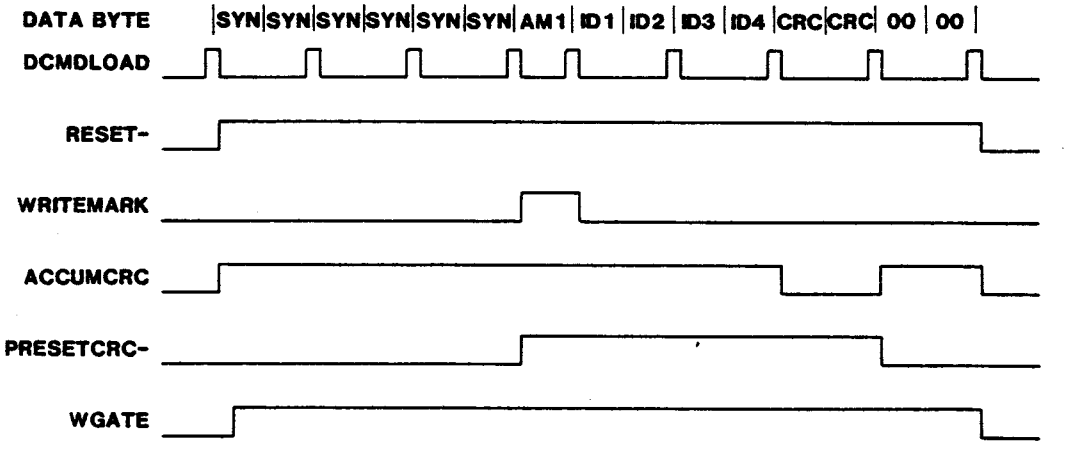


FIGURE 4-26. ID FIELD WRITE TIMING - SINGLE DENSITY

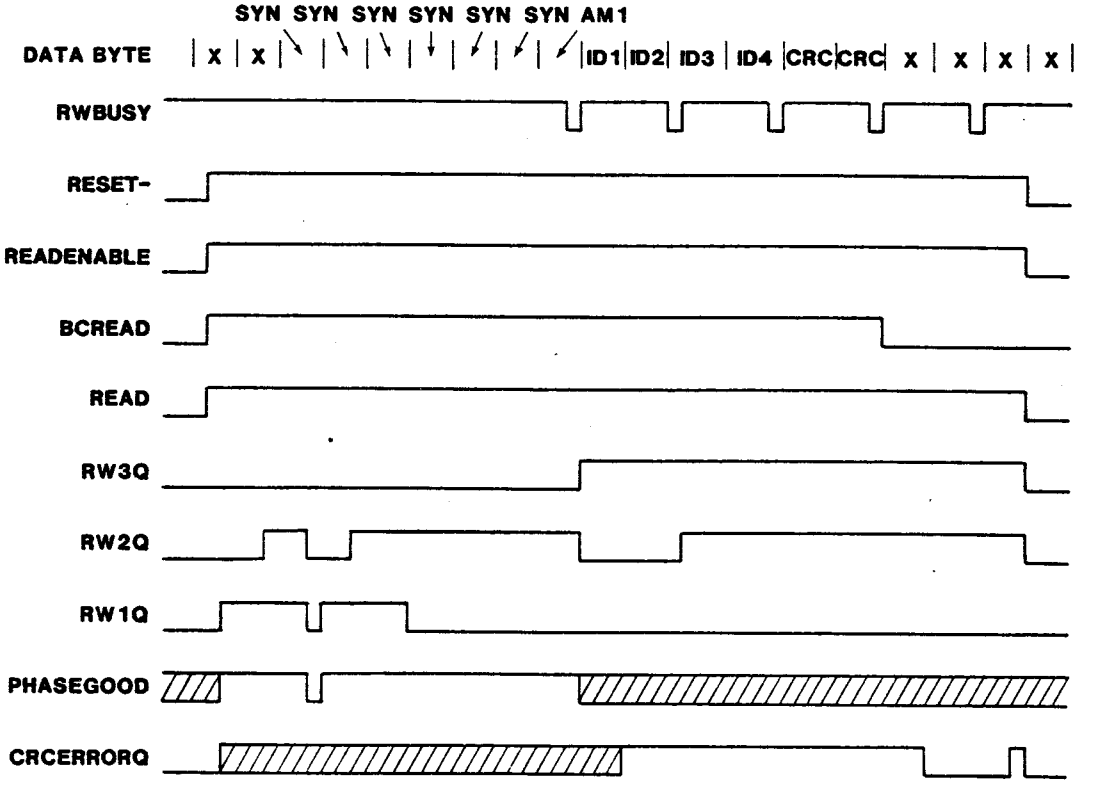


FIGURE 4-27. ID FIELD READ TIMING - SINGLE DENSITY

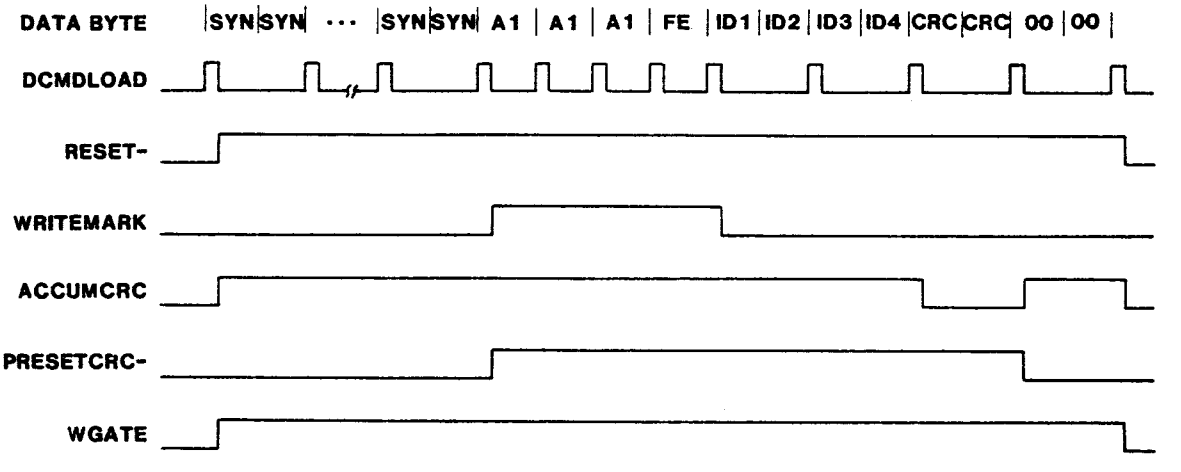


FIGURE 4-28. ID FIELD WRITE TIMING - IBM DOUBLE DENSITY

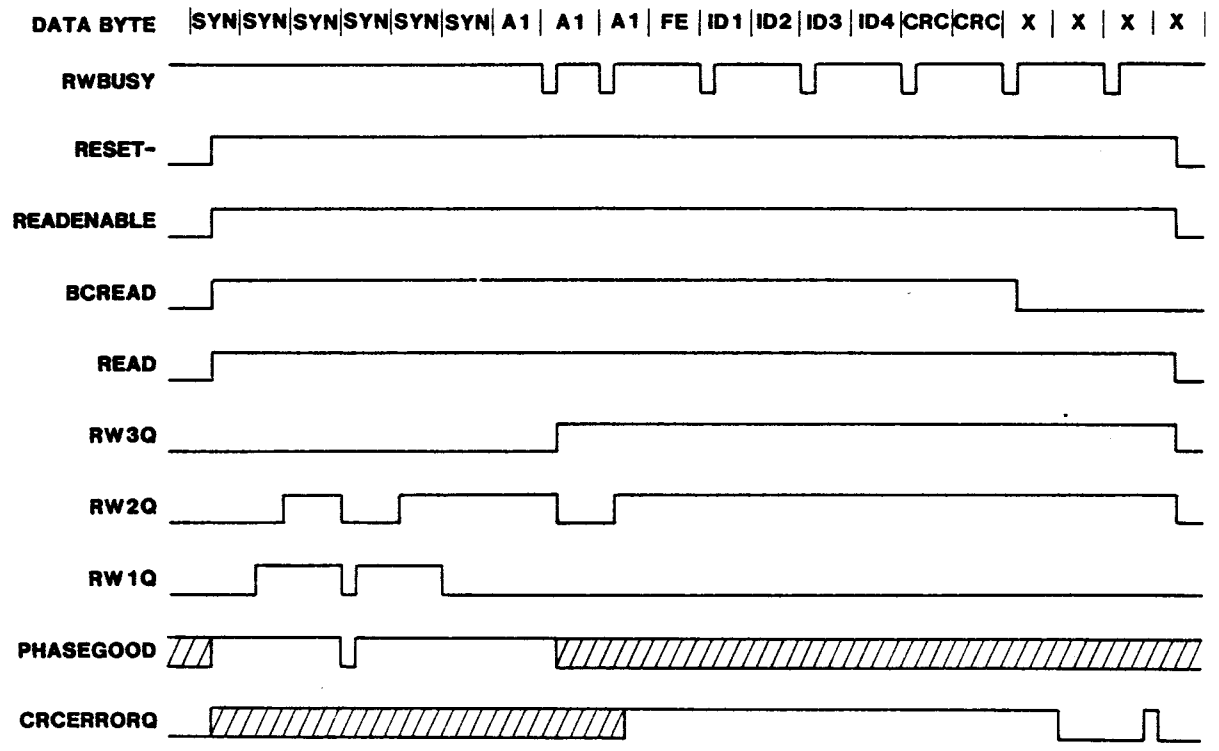


FIGURE 4-29. ID FIELD READ TIMING - IBM DOUBLE DENSITY

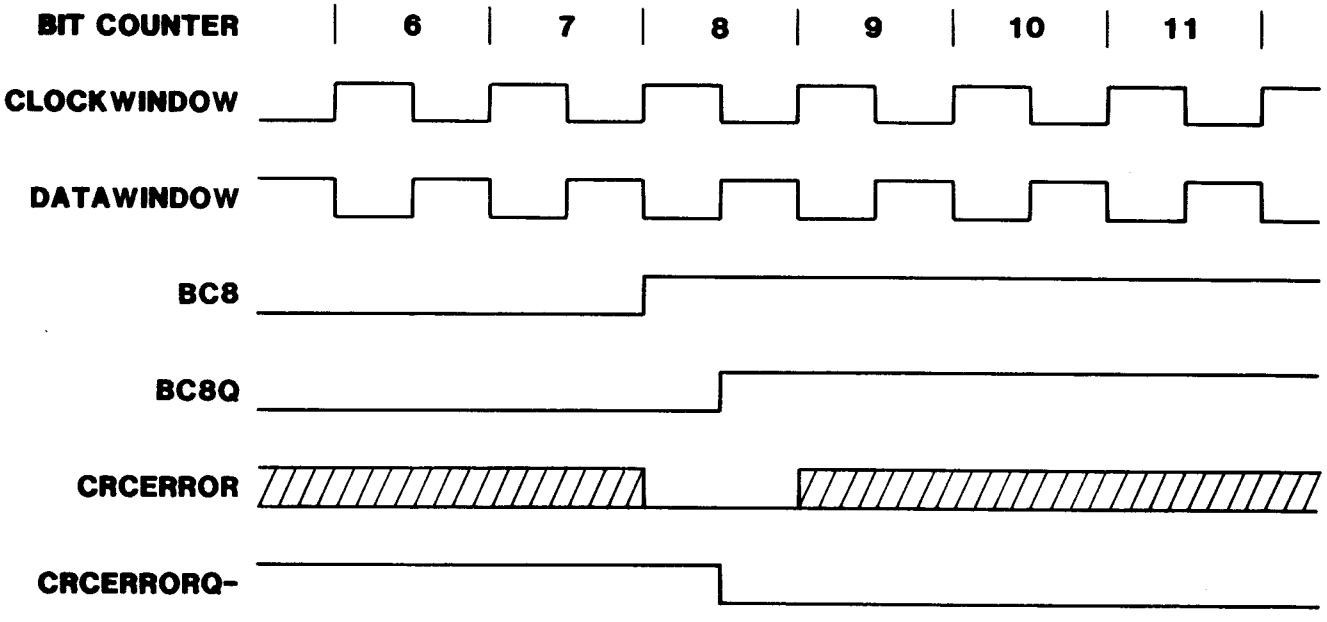


FIGURE 4-30. CRC ERROR LATCH TIMING

4.8 EIA PORT

The TM 990/303B floppy disk controller module has an EIA port onboard which is particularly useful because new microcomputers, such as the TM 990/102 module, only have one EIA port onboard. The processor on the TM 990/303B will have no control over or interface to the port. This EIA port is configured by a decode PROM in the host's CRU map. Jumper J13 can be jumpered to select the EIA port at a CRU software base address of either 180_{16} or 300_{16}. The TM 990/303B module will be shipped with the EIA port having a CRU base address of 180_{16}. The user, by programming another PROM (74S287), may configure the EIA port to any CRU address desired. The EIA port will be controlled by the onboard TMS 9902A. See Figure 5-1 (facing page) for EIA port logic diagram.

Interrupts will be provided for use with the TMS 9902A. The interrupt level is jumper-selectable at jumper J12. The TMS 9902A interrupt level cannot be the same interrupt level as the TM 990/303B or it will conflict with the interrupt level designated for the TM 990/303B. The jumper plugs on the module are placed such that the TMS 9902A and the TM 990/303B cannot have the same interrupt level. Note that the TM 990/303B interrupt jumpers are right next to the TMS 9902A jumpers on the module.

4.8.1 Changing the CRU Address of the EIA Port

The CRU address of this port may be changed in the following manner. The decode PROM is a 74S287 which has 256 4-bit locations. The four outputs are D01, D02, D03, and D04. D01 and D04 are currently being used by the controller. These two locations must remain the same when configuring a new PROM or the controller will not be able to be communicated with. The TMS 9902A which controls the port is enabled by either the D02 or the D03 output of the PROM. Jumper J13 selects the D02 (E78-E77, 180_{16}) or the D03 (E76-E77, 300_{16}) output of the PROM. Address line A10 is gated into the PROM as well as into the TMS 9902A: therefore, two of the D02 or D03 outputs must be low to provide for A10 being both high and low (the TMS 9902A is enabled by a low level signal). Address lines A3 through A9 are used to generate the Chip Enable-signal to the TMS 9902A, and address lines A10 through A14 are gated directly into the TMS 9902A to generate that device's 32-bit CRU space. See Appendix E for PROM programming instructions.

4.8.2 EIA Port DTR/DSR Signals Affected by R43

This EIA port deviates from the EIA standard in regard to the DTR (Data Terminal Ready) signal at pin P2-20 (see Figure 5-1). With R43 installed, the DTR signal may be read at P2-20 (to show if the terminal is ready) unless the cable is disconnected or the terminal power is off, in which case the DTR signal will always appear ready. This feature allows data to be transmitted despite the fact that the majority of terminals and printers have no connection to DTR through the cable. However, if data is transmitted while the cable is disconnected or terminal power is off, the data will be lost.

In order to detect cable disconnection or the terminal power-off condition as well as the DTR signal, resistor R43 must be removed from the board. This makes the DTR implementation compatible with EIA standard RS-232, but now terminals and printers must provide the DTR signal through the cable.

The DTR signal is actually read by software via the DSR (Data Set Ready) status bit, CRU bit 27 of the TMS 9902A. The following table illustrates four software functions in response to the status of CRU bit 27, depending on whether or R43 is installed.

Case	Hardware	Software	Function
1	R43 Installed	Reads Bit 27	Data Terminal Ready condition is detected regardless of cable disconnection or power off condition so that data is transmitted where DTR is not implemented.
2	R43 Installed	Ignores Bit 27	Data Communication is independent of the status of the terminal.
3	R43 Removed	Reads Bit 27	Fully EIA standard compatible, i.e., data is transmitted only when the cable is connected and power is on and the data terminal is ready.
4	R43 Removed	Ignores Bit 27	Same as Case 2 above.

NOTE

An interrupt is generated when bit 27 changes logic level. This applies to all four cases mentioned above.

SECTION 5

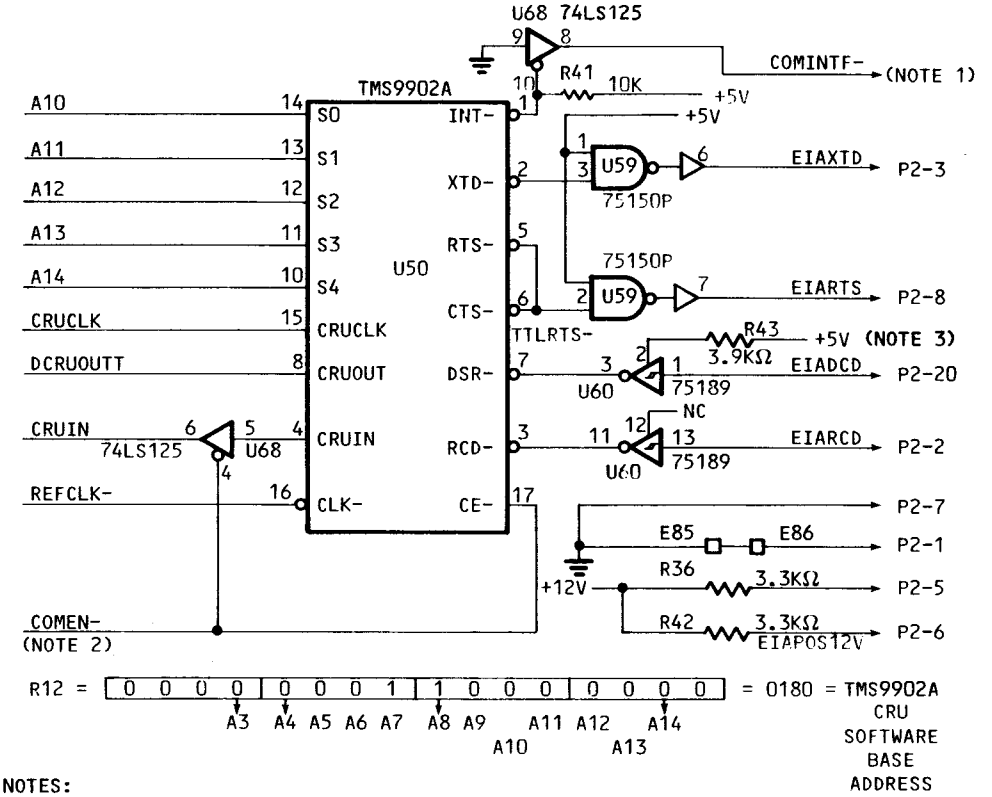
PROGRAMMING TMS 9902A EIA PORT CONTROLLER AT PORT P2

5.1 GENERAL

Connector P2, the EIA port of the TM 990/303B, is provided as an additional 25-pin terminal connection for your microcomputer system. Figure 5-1 shows the interconnections to the port's TMS 9902A asynchronous controller. Note that this port does not drive TTY devices. The TMS 9902A can be programmed to do the following:

- read characters coming in to the TMS 9902A from connector P2,
- transmit characters via the TMS 9902A and connector P2,
- be an interval timer, interrupt driven or polled.

Baud rates, parity, character length, and other data transmission details are programmed into TMS 9902A registers.



NOTES:
1. To System Bus via jumpered interrupt level at J12
2. From desired CRU Software Base Address Selected at jumper J13:
   J13 E77-E78 = 0180 CRU Software Base Address
   J13 E77-E76 = 0300 CRU Software Base Address
   J13 E76-E75 = TMS9902A Deselected
3. R43 is futher explained in section 4.8

FIGURE 5-1. INTERCONNECTIONS TO PORT P2'S TMS 9902A CONTROLLER

5.2 ADDRESS STRUCTURE OF THE CRU

NOTE

Because this port has a unique CRU address (explained below), it cannot be driven by the standard software on the monitors accompanying the TM 990 CPU modules, unless the CRU address to the read/write controllers (TMS 9902A's) is modified. Procedures in this section will guide the writing of software to control this port.

Programming the TMS 9902A is through the Communications Register Unit (CRU) which is the serial input and output port on the TMS 9900 microprocessor. Your microcomputer user's guide explains CRU instructions and addressing, and the TMS 9902A data manual accompanying your microcomputer provides a detailed description of the TMS 9902A. NOTE: If not familiar with the CRU, read your microcomputer user's guide before proceeding.

In general, there are five CRU instructions, three single bit (SBZ, SBO, and TB) and two multibit (LDCR and STCR) CRU instructions. When executed, CRU instructions cause an address to be placed on the address lines and then one of the following:

- send one (or more) bits over the CRUOUT line of the microprocessor as well as strobe the CRUCLK line of the microprocessor, or
- receive one (or more) bits at the CRUIN line, or
- test the logic value at the CRUIN line and place this value in the EQ bit of the Status Register.

In CRU programming, a device is selected to read or to be written to via the serial CRUIN or CRUOUT pins of the microprocessor. This selection is made by first entering a CRU Software Base Address in register 12 before a CRU instruction is executed. During execution, bits 3 to 14 of the register are used to make up a value to be placed on address bus lines A3 to A14 during CRU instruction execution. An address-line decoder then enables the peripheral device (e.g., TMS 9902A) so that it can be accessed through the CRUIN or CRUOUT lines.

On the TM 990/303B, the CRU Software Base Address for the TMS 9902A can be one of two selected at jumper J13:

Jumper J13 Setting	CRU Software Base Address
E77-E78	0180_{16}
E77-E76	0300_{16}
E76-E75	(Port deselected, not programmable)

NOTE

In examples within this section, a CRU Software Base Address of 0300_{16} will be used with jumper J13 set E77-E76.

TABLE 5-1. CRU PROGRAMMABLE BITS ON THE TMS 9902A

Displacement (Base 10)	Device and Device Signals	
	Input Description (Read)	Output Description (Write)
00	RBR0 Rec buffer data	REG0 Register data bit 0
01	RBR1	REG1
02	RBR2	REG2
03	RBR3	REG3
04	RBR4	REG4
05	RBR5	REG5 (Note 2)
06	RBR6	REG6
07	RBR7 Rec buffer data	REG7
08	0	REG8
09	RCVERR Receive error	REG9
10	RPER Receive parity error	REG10 Register data bit 10
11	ROVER Receive overun error	LXDR Load transmit data register
12	RFER Receive framing error	LRDR Load receive data register
13	RFBD Receive full bit	LDIR Load interval register
14	RSBD Receive start bit	LDCTRL Load control register
15	RIN RIN pin status	TSTMD Test mode
16	RBINT Receive interrupt	RTSON Request to send is on
17	XBINT Transmit intrrupt	BRKON Break on
18	0	RIENB Receive interrupt enable
19	TIMINT Timer interrupt	XBIENB Xmit interrupt enable
20	DSCINT Data set change interrpt	TIMENB Timer interrupt enable
21	RBRL Rcv buffer reg loaded	DSCENB Data set chg intr enable
22	XBRE Trnsmt buffer empty	
23	XSRE Trnsmt shft reg empty	
24	TIMERR Timer error	
25	TIMELP Timer elapsed	
26	RTS Request to send	
27	DSR Data term ready (Note 3)	
28	CTS Clear to send	
29	DSCH Data set status change	
30	FLAG Register load/break	
31	INT Interrupt	RESET Reset TMS 9902A

NOTES

1. The values on address lines A10 to A14 are also the displacement in hexadecimal from the CRU hardware base address.

2. The TMS 9902A Registers, loaded via data bits 1 - 10, include the following internal registers:

   - Control Register (character length, timer time base, parity, and number of stop bits),
   - Interval Register (interval timer setting),
   - Receive Data Rate Register (set receive baud rate), and
   - Transmit Data Rate Register (set baud rate if different from that in the Receive Rate Register).

3. The DSR bit is further explained in section 4.8.

Note that when running with a TMS 9900 based CPU, the CRU bit address is the resulting value on address lines A3 to A14 of the microcomputer and the system bus (bus value at address lines on port P1). To derive this address-line value, add the binary values in bits 3 to 14 of register 12 (bit 15 of register 12 is ignored) to the displacement noted in the CRU instruction. This is applied to address lines A3 to A14 of the microprocessor (and subsequently the system bus). For example, Table 5-1 lists the bits that are accessible on the TMS 9902A via the CRU. For example, to test the RCVERR bit (displacement of 9), the following code can be used:

```
LI    R12,>0300     SET CRU SFTWR BASE ADDR OF 0300
TB    9             TEST BIT 09 (RCVERR)
JEQ   ERROR         JUMP IF BIT IS A ONE
```

5.3 LOADING THE PRINCIPAL INTERNAL REGISTERS OF THE TMS 9902A

NOTES

1. In this section, mention is made of the registers and CRU bit functions of the TMS 9902A. It is presumed that the reader is familiar with the TMS 9902A; if not, the TMS 9902A data book is provided with your microcomputer for reference. Other programming examples can be found in that data book.

2. In the coding examples in this section, the label TM9902 represents the software base address (R12 contents of 0300_{16}) for the TMS 9902A. It is as if the following assembler directive was used:

```
TM9902    EQU >300      LABEL "TM9902" = HEX 0300
```

5.3.1 Principal Registers of the TMS 9902A

Before the TMS 9902A can be operational, one or more internal registers must be programmed to initial values. These registers are depicted in Figure 5-2, and the use of their stored values is listed below:

- Control Register: Its value designates
  - number of stop bits,
  - parity,
  - interval-timer time base, and
  - character length.
- Interval Register: Length of timer countdown to zero.
- Receive Data Rate Register: Receive baud rate.
- Transmit Data Rate Register: Transmit baud rate.

Control Register

7	6	5	4	3	2	1	0
SBS1	SBS2	PENB	PODD	CLK4M	NOT USED	RCL1	RCL0
MSB							LSB
Stop Bit Select		Parity Enable	Odd Parity Select	Clock Divide Select		Character Length Select	

Control Register. Stop bits, parity, clock division, and character length are set by writing to the TMS 9902A Control Register. Figure 5-3 shows this register in detail.

The clock divide value allows choosing to divide the external clock input at pin 16 by a value of 3 or 4 using the following:

- CLK4M a one means divide by 4 (f_{int} = 1 MHz @ 4 MHz clk)
- CLK4M a zero means divide by 3 (f_{int} = 1 MHz @ 3 MHz clk)

NOTE: All TM 990 systems use a 3 MHz clock for the TMS 9902A.

Interval Register

7	6	5	4	3	2	1	0
TMR7	TMR6	TMR5	TMR4	TMR3	TMR2	TMR1	TMR0
MSB							LSB

Interval Register. The Interval Register contains the binary value to be counted down for timing. When the countdown reaches zero, an interrupt is issued if enabled by writing a one to CRU bit 20 (TIMENB). If f_{int} = 1 MHz (see Control Register), each count lasts 64 usec; i.e., 64 x ($1/f_{int}$). Interrupt level is selected at jumper J12. An example of timer programming is shown in section 5.4.3.

Receive and Transmit Data Rate Register

10	9	8	7	6	5	4	3	2	1	0
DV8	DR9	DR8	DR7	DR6	DR5	DR4	DR3	DR2	DR1	DR0
MSB										LSB

Receive Data Rate Register. This register is to set the desired baud rate constant of the incoming data. For the Receive Data Rate and Transmit Data Rate Registers, CRU bits 0 to 10 are used to load the register value. If f_{int} = 1 MHz (see Control Register), then the baud rate is equal to 1 M bps/(register value x 2) if bit 10 is a zero, or 1 M bps/(register value x 2 x8) if bit 10 is a one. Figure 5-4 (c) describes loading the Data Rate Register for 110 baud.

Transmit Data Rate Register. This register is set to the desired baud rate constant of the data to be transmitted (see Receive Data Rate Register above).

FIGURE 5-2. TMS 9902A PROGRAMMABLE REGISTERS

5.3.2 Programming the Control Register

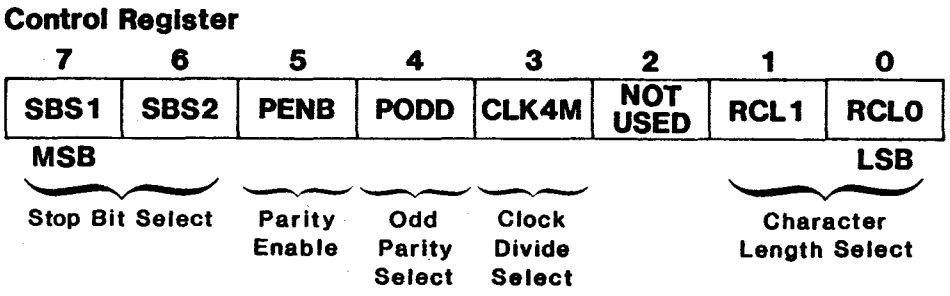


Figure 5-3 shows the Control Register bit combinations corresponding to character length, clock period division, parity selection, and stop bit selection. At the bottom of the figure is example code to set the values shown at the top of the figure. The values to be set in this example are:

- character length of seven bits,
- clock divide value of 3 (used on a 3 MHz system)
- even parity
- two stop bits.

As shown in Figure 5-3, the clock divide value (bit 3 or CLK4M) is used to establish a time base for the Interval Timer and baud rates for the transmitter and receiver. The clock divide value is used to derive two values used as a basis for computing other timing:

$$f_{int} = \text{(Input Clock Frequency)/(divider selected by CLK4M)}$$

$$t_{int} = 1/f_{int}$$

The CLK4M value divides the external clock input at pin 16 by a value of 3 or 4 using the following:

- CLK4M a one means divide by 4 (e.g., f_{int} = 1 MHz @ 4 MHz clock)
- CLK4M a zero means divide by 3 (e.g., f_{int} = 1 MHz @ 3 MHz clock)
  Note: All TM 990 systems use a 3 MHz clock for the TMS 9902A.

For example, to find f_{int} with a 3 MHz input and bit CLK4M set to zero:

f_{int} = (Input Clock Frequency)/(Frequency divider selected)
= (3 MHz)/(3)
= 1 MHz

To find t_{int}, find the inverse of f_{int}. For example:

$1/f_{int} = 1/1\text{-MHz} = t_{int}$
t_{int} = 1 usec

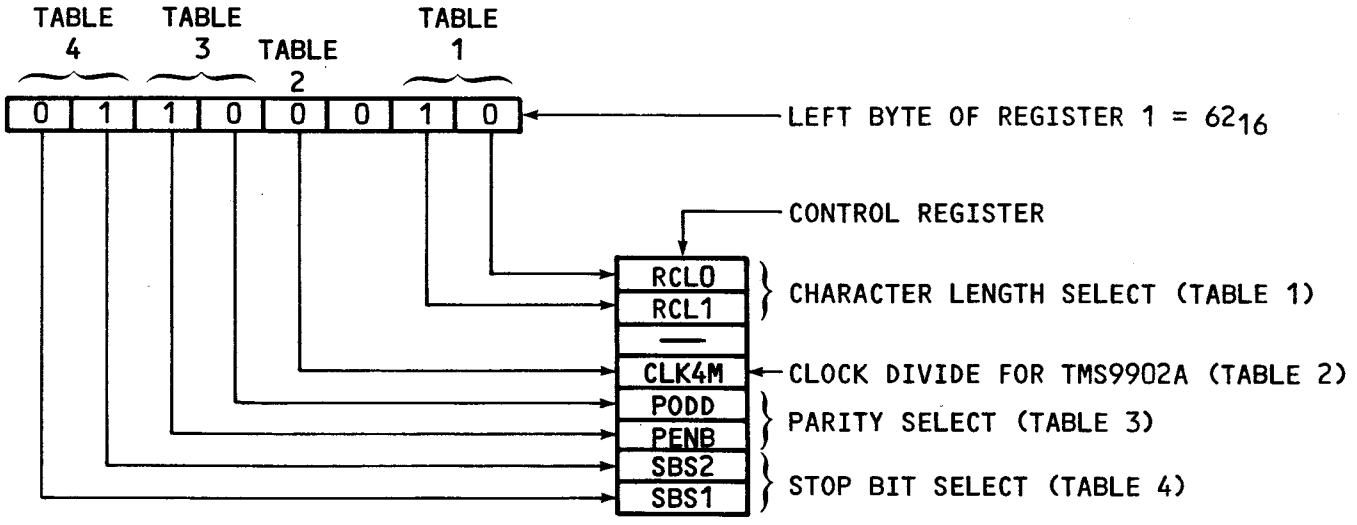


TABLE 1 CHARACTER LENGTH SELECTION

RCL1 BIT 1	RCL0 BIT 0	CHARACTER LENGTH
0	0	5 BITS
0	1	6 BITS
1	0	7 BITS
1	1	8 BITS

TABLE 2 CLOCK PERIOD DIVIDER

CLK4M	INTERNAL CLOCK PERIOD (t_{int}) AT 3 MHz CLOCK
0	1/(3 MHz/3) = 1 µs
1	1/(3 MHz/4) = 0.75 µs

TABLE 3 PARITY SELECTION

PENB BIT 5	PODD BIT 4	PARITY
0	0	NONE
0	1	NONE
1	0	EVEN
1	1	ODD

TABLE 4 STOP BIT SELECTION

SBS1 BIT 7	SBS2 BIT 6	NUMBER OF TRANSMITTED STOP BITS
0	0	1½
0	1	2
1	0	1
1	1	1

```
LI      R1,>6200        LEFT BYTE CONTAINS VALUE FOR CONTROL REGISTER
LI      R12,TM9902      SET PORT P2 CRU SOFTWARE BASE ADDRESS (HEX 0300)
SBO     31              RESET TMS 9902A
LDCR    R1,8            APPLY BITS TO CONTROL REGISTER
```

FIGURE 5-3. LOADING THE TMS 9902A CONTROL REGISTER

5.3.3 Programming the Interval Register

Interval Register

7	6	5	4	3	2	1	0
TMR7	TMR6	TMR5	TMR4	TMR3	TMR2	TMR1	TMR0
MSB							LSB

The use of bit 3 on the Control Register (CK4M) is explained in section 5.3.2. This bit value is used to derive f_{int} and t_{int} which affect the countdown value that is loaded into the Interval Register (I.R.) as well as the baud rates. The countdown period of the Interval Register is computed thus:

$$\text{Countdown Time} = 64 \times (\text{I.R. Value}) \times t_{int}$$

For example, with t_{int} = 1 microsecond and the Interval Register loaded with all ones (FF_{16} or 255_{10}), the following countdown time is derived:

$$\text{Countdown Time} = 64 \times 255 \times 1 \text{ usec}$$
$$= 16{,}320 \text{ usec} = 16.320 \text{ ms}$$

To load the Interval Register:

1) Set CRU Software Base Address
2) Set bit 14 (LDCTRL) to zero
3) Set bit 13 (LDIR) to one
4) Load 8 bits of data into I.R.

Example Code

```
LI     R12,TM9902
SBZ    14
SBO    13
LDCR   @COUNT,8
```

A faster alternate loading method is explained in section 5.3.5. When LDIR is reset to zero (SBZ 13), the countdown begins. LDIR is automatically reset to zero by the eighth bit loaded into the Interval Register.

A derivation of the above countdown formula can be used to derive the Interval Register value required for a desired interval:

$$\text{I.R. Value} = \frac{\text{Countdown-Time in usec}}{64 \times t_{int}}$$

For example, to determine the I.R. value for an interval of 10,000 usec:

$$\text{I.R. Value} = \frac{10{,}000 \text{ usec}}{64 \times 1 \text{ usec}} = \frac{10{,}000}{64}$$

$$= 156_{10} = 9C_{16}$$

To have an interrupt issued at countdown, set bit 20 (TIMENB) to one, set the desired interrupt value at jumper J12, and enable interrupts at the microcomputer (these latter include microprocessor interrupt mask, settings at the interrupt controller (such as the TMS 9901), interrupt vector values, and the interrupt service routine).

When countdown to zero occurs, read-bit 25 (TIMELP) becomes a one and (if set up) an interrupt is issued. Then the Interval Register contents are reloaded into the timer and the countdown begins again. TIMELP is reset by writing a one or zero to bit 20 (TIMENB), which can be done by the interrupt service routine. (For bit 20, writing a one to it will enable the next interrupt while a zero will disable interrupts.) If TIMELP (bit 25) was still set when a countdown occurs again, read-bit 24 (TIMERR) will be set.

5.3.4 Loading the Receive Data Rate and Transmit Data Rate Registers

Receive and Transmit Data Rate Register

10	9	8	7	6	5	4	3	2	1	0
DV8	DR9	DR8	DR7	DR6	DR5	DR4	DR3	DR2	DR1	DR0
MSB										LSB

These two registers select the baud rate at which the TMS 9902A will receive and transmit characters.

The Receive Data Rate Register contains eleven bits; these represent two divisors of the f_{int} value specified in the Control Register. The least-significant ten bits (bits 0 to 9) are one divisor. The second divisor is specified in the register's most significant bit (bit 10) which designates whether an additional division-by-eight should be made. Bit 10 specifies:

Bit 10 a one = Divide by eight
Bit 10 a zero = Divide by one (i.e, no change)

The following formula is used to derive the baud rate:

$$\text{Baud Rate} = f_{int}/[2 \times (\text{bits } 0\text{-}9) \times (\text{bit 10 value})]$$

For example, with f_{int} = 1 usec (specified in Control Register), a baud rate of 110 can be specifed by bit 10 a one and bits 9 to 0 containing 568_{10}:

$$\text{Baud Rate} = \frac{1 \text{ MHz}}{2 \times 568 \times 8} = \frac{1 \text{ MHz}}{9088} = 110.03521 \text{ bits per second}$$

For a known baud rate, the division rate can be computed thus:

$$\text{Division Rate} = \text{Bits } 0 - 9 = f_{int}/[2 \times (\text{baud rate})]$$

For example, to determine the division rate for 110 baud:

$$\text{Division Rate} = \text{Bits } 0 - 9 = \frac{1{,}000{,}000}{2 \times 110} = \frac{1{,}000{,}000}{220} = 4545$$

Since a maximum value of 1023 can be placed in bits 0 to 9, use the divide-by-eight bit also. Dividing 4545 by 8 = 568. Then load 568_{10} (238_{16}) into the Receive Data Rate Register with bit 10 a one (apply 638_{16} to the register).

To load a value into the Receive Data Rate Register, write-bits 14 and 13 (LDCTRL and LDIR) must be zeroes and write-bit 12 (LRDR) must be a one. To load the Receive Data Rate Register:

	Example Code	
1) Set CRU Software Base Address	LI	R12,TM9902
2) Set bit 14 (LDCTRL) to zero	SBZ	14
3) Set bit 13 (LDIR) to zero	SBZ	13
4) Set bit 12 (LRDR) to one	SBO	12
5) Load 10-11 bits of data into the register	LDCR	@VALUE,11

TABLE 5-2. CRU-BIT LOGIC TO ENABLE TMS 9902A REGISTERS

Register(s) to be Loaded	CRU Bit Values to Enable Register (Bit number is the displacement from CRU Base Address)				
	LDCTRL Bit 14	LDIR Bit 13	LRDR Bit 12	LXDR Bit 11	Register Occupies Bits
Control Register	1	X	X	X	0 to 7
Interval Register	0	1	X	X	0 to 7
Both Receive and Transmit D. R. Regs*	0	0	1	1	0 to 10
Receive Data Rate Register only	0	0	1	X	0 to 10
Transmit Data Rate Register only	0	0	X	1	0 to 10
Transmit Buffer (data to send/rcve)	0	0	0	0	0 to 7

*Loading the Receive and Transmit Data Rate Registers at the same time loads both with the same baud rate (this can save a programming step).
X = don't care

A faster method of loading is explained in section 5.3.5. Computing and loading the baud rate for the Transmit Data Rate Register is similar as shown for the Receive Data Rate Register with one exception: all of bits 14 and 13 must be first set to zeroes and bit 11 (LXDR) set to one.

5.3.5 Ease in Sequentially Loading Several Registers

As explained in the TMS 9902A data manual, the TMS 9902A four internal registers are loaded by writing values to bits 0-7 or 0-10 at the TMS 9902A CRU base address. The first part of the register-load operation is to enable the register desired; this is done by setting CRU bits 11 to 14 as shown in Table 5-2 to select the desired register. The next step is to write the desired register value to bits 0-7 or 0-10 as shown in the table.

Note that the register-enable bits (in the first step above) are set to all ones by a RESET operation (write a one to bit 31). This would set up for loading the Control Register. After a register is loaded, its register-enable bit is automatically reset to zero. This sets up for loading the next register in the order shown in Table 5-2, top to bottom (e.g., after loading the Control Register, bit 14 is automatically set to zero; this sets up for loading the Interval Register, etc.). The register contents are accessed via CRU bits 0-7 or bits 0-10, depending on the register size. Figure 5-4 is an example of a RESET followed by four LDCR instructions which load the four registers.

After writing a one to bit 31 for a RESET, do not read or write to the TMS 9902A for 11 clock cycles. A time delay to wait 11 cycles can be two NOPs (two JMP $+2) on a TMS 9900 processor, or a decrement of a register from 7 to 0 on a faster TMS 99110 processor at 6 MHz.

Figure 5-4 shows example code to conveniently load all four registers. The enabling bits for these registers will be set to one upon a RESET (CRU bit 31 set to one); this allows writing to the registers using the logic shown in Figure 5-4.

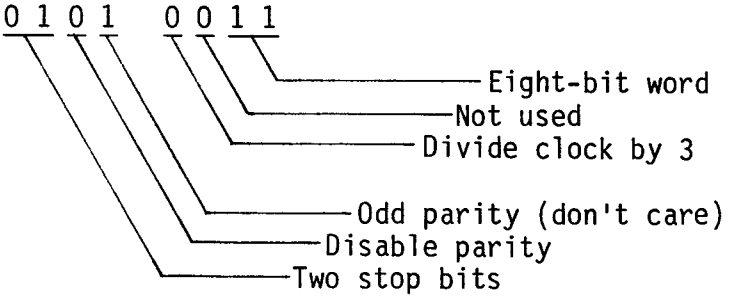
```
        LI    R12,TM9902    CRU BASE ADDRESS IN R12 (>0300)
*RESET THE TMS 9902A (FOLLOW WITH NO-OP FOR STABILIZATION) (SEE NOTE 2)
        SBO   31            RESET TMS 9902A, ALL REG LOAD BITS = 1
        NOP                 NO-OPS TO USE 11 CYCLES OF 9902A CLOCK
        NOP                 TWO NO-OPS NEEDED (SEE NOTE 2)
*LOAD THE CONTROL REGISTER (REG. CONTENTS EXPLAINED BELOW)
        LI    R1,>5300      SET UP CONTROL REGISTER CONTENTS
        LDCR  R1,8          APPLY TO CONTROL REGISTER
** THE LOAD CONTROL REGISTER BIT WAS AUTOMATICALLY RESET TO ZERO
*LOAD THE INTERVAL REGISTER
        LI    R1,>9C00      >9C YIELDS 9.984 MILLISECONDS
        LDCR  R1,8          APPLY TO INTERVAL REGISTER
** THE INTERVAL REGISTER BIT WAS AUTOMATICALLY RESET TO ZERO
*LOAD THE RECEIVE DATA RATE REGISTER (REG. EXPLAINED BELOW)
        LI    R1,>638       >0638 YIELDS 110.04 BITS PER SECOND
        LDCR  R1,11         APPLY TO RECEIVE DATA RATE REGISTER
** THE RCV DATA RATE REGISTER BIT WAS AUTOMATICALLY RESET TO ZERO
*LOAD THE TRANSMIT DATA RATE REGISTER
        LI    R1,>01A0      >01A0 YIELDS 1200 BITS PER SECOND
        LDCR  R1,12         APPLY TO TRANSMIT DATA RATE REGISTER
** TWELVETH BIT (ZERO) CLEARS TRANSMIT DATA RATE BIT, READIES FOR
** LOADING THE TRANSMIT BUFFER REGISTER WITH A CHARACTER
```

NOTES

1. If the same baud rate is needed for transmit as well as receive, delete the Load Receive Data Rate Register code above; then the Load Transmit Rate Register code also loads the Receive Data Rate Register as both CRU bits 11 and 12 are ones.
2. The two NOPs listed above are for a TMS 9900 processor at 3 MHz. Decrement a register from 7 to 0 on a TMS 99110 processor at 6 MHz.

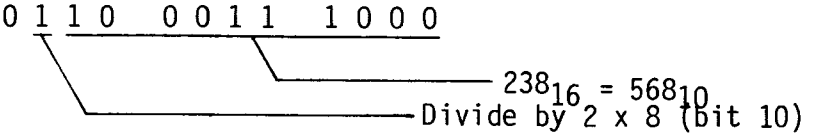
(a) Code to Initialize the TMS 9902A

```
LI      R1,>5300
LDCR    R1,8
```



(b) Load Control Register Code

```
LI    R1,>638
LDCR  R1,11
```



1 M/(568 x 2 x 8) = 110.04 bps baud rate

(c) Load Receive Data Rate Register Code

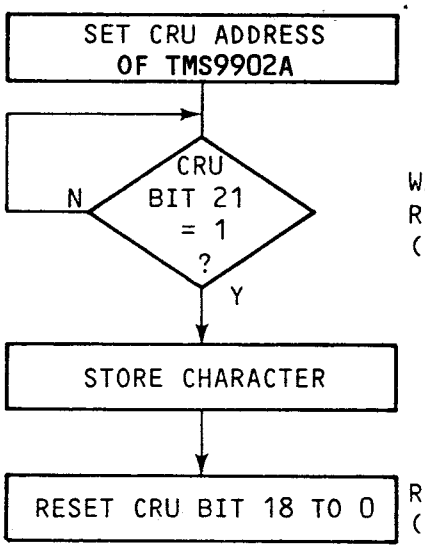
FIGURE 5-4. CODING TO LOAD ALL FOUR TMS 9902A REGISTERS

5.4 PROGRAMMING EXAMPLES

5.4.1 Receive Character by Polling (Figure 5-5)

A high-to-low transition on external line RIN (pin 3) of the TMS 9902A activates its receiver circuitry. Pin RIN then is used to serially receive the character. Figure 5-5 shows code to poll for a character received. When the complete character is received, bit 21 (RBRL) is set to one. In the third line of code in Figure 5-5, bit 21 is polled for a full character received. Writing to bit 18 (RIENB) resets bit 21 so that it can be checked again for the next character received. Note that after resetting the TMS 9902A via the RESET bit (31), no input operation should occur for 11 clock cycles. See Note 2 on Figure 5-4.

Figure 5-5 shows checking for character received by polling bit 21; however, if enabled, an interrupt can be generated to the microcomputer upon receipt of a character. To enable this interrupt, set bit 18 (RIENB) to a one. Of course, the interrupt level jumper, the interrupt controller (e.g., TMS 9901) and microprocessor interrupt mask on the CPU module must be preprogrammed along with the values in the interrupt vector areas and the interrupt service routine.



WAIT FOR CHARACTER TO ENTER RECEIVE BUFFER REGISTER (RBRL = 1)

RESET THE RBRL FLAG (RECEIVE BUFFER REGISTER LOADED)

```
        LI    R12,TM9902     TMS 9902A CRU ADDRESS IN R12 (>0300)
        LI    R1,STORE       MEMORY ADDRESS TO STORE CHARACTERS
        .
        .
        .
LOOP    TB    21             HAS CHARACTER BEEN RECEIVED (RBRL=1?)
        JNE   LOOP           NO, LOOP TO WAIT FOR CHARACTER
        STCR  R2,8           BRING 8-BIT CHARACTER TO R2
        CI    R2,>0400       WAS CHARACTER AN EOT?
        JEQ   EXIT           YES, EXIT
        MOVB  R2,*R1+        STORE CHARACTER IN MEMORY
        SBZ   18             RESET RBRL FLAG (OUTPUT BIT 21)
        JMP   LOOP           GET NEXT CHARACTER
EXIT
        .
        .
```

FIGURE 5-5. RECEIVE CHARACTER(S) BY POLLING

5.4.2 Transmit Character(s) Through Port P2 (Figure 5-6)

Figure 5-6 is split into two separate parts on the next two pages. The first sheet of this figure is a flow chart showing the sequence of TMS 9902A bit manipulation to send characters over the CRU. The second sheet is an example of coding to affect this transfer. This program polls bits to determine completion of the various stages of character transmission.

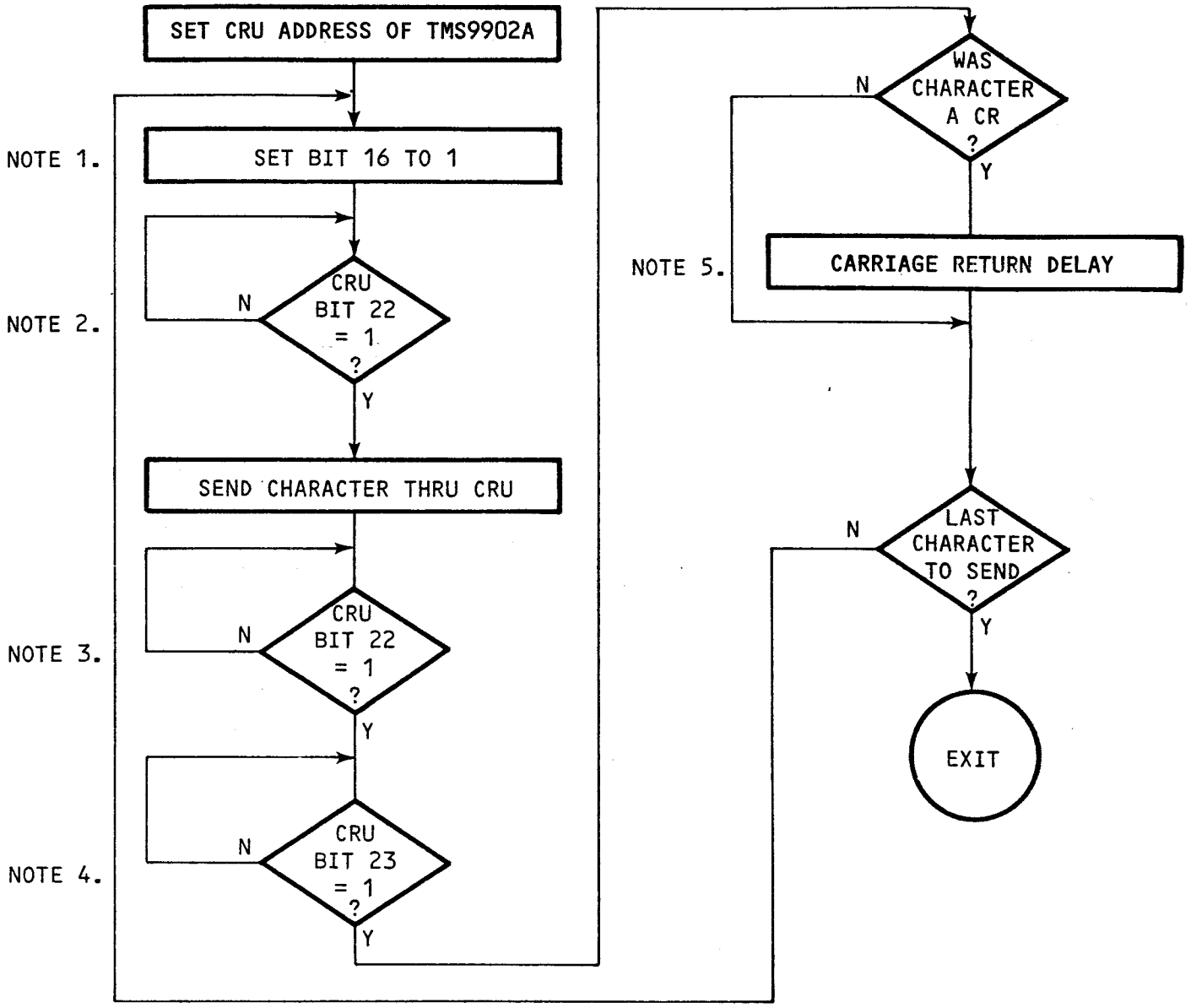
Write a one to bit 16 (RTSON) to drive the external RTS- (request to send) pin (pin 5) low which is interpreted by the peripheral receiving device. The peripheral receiving device enables the TMS 9902A transmitter by setting the TMS 9902A's external CTS- (clear to send) line (pin 6) to a low value.

In a polled transmit operation, first test bit 22 (XBRE) to verify that the Transmit Buffer Register is presently empty. Then write the character to the TMS 9902A via the CRUOUT line (LDCR instruction) of the host. Next check bit 22 (XBRE) for a one to see that the character has exited the TMS 9902A Receive Buffer Register; this bit is reset when a character is sent to the Transmit Buffer Register. Then check bit 23 (XSRE) for a one to see that the character has exited the Transmit Buffer Register so that (if required) a timing loop can be entered for a particular baud rate should a delay be needed (e.g., for a mechanical carriage return). Such a delay is shown in Figure 5-6 to delay a total of 200 ms for a mechanical carriage return. Another delay is used to pad each character 25 ms for a printer that operates at 300 baud while interfaced to a transmitter operating at 1200 baud (at 1200 baud, the 120 characters-per-second (cps) rate takes 1000/120 or 8.3 ms for each character to be sent, while at 30 cps, each requires 1000/30 or 33.3 ms to be sent). Characters are transmitted from the TMS 9902A via its XOUT pin (pin 2).

CAUTION

After resetting the TMS 9902A via the RESET bit (31), prevent output operations from occurring for 11 clock cycles. This can be done with no-ops (JMP $+2) using a TMS 9900 processor or a decrement of a register from 7 to 0 on a faster TMS 99110 processor at 6 MHz. For example:

```
      LI   R5,7     COUNT OF 7 TO DECREMENT
LOOP  DEC  R7       DECREMENT R7 BY ONE
      JNE  LOOP     LOOP UNTIL R7 = 0
           .
           .
           .
```



NOTES:
1. ACTIVATE $\overline{\text{RTS}}$ (LOW)
2. WAIT FOR TMS9902A TRANSMIT BUFFER REGISTER TO BE EMPTY (XBRE = 1)
3. WAIT FOR TMS9902A TRANSMIT BUFFER REGISTER TO BE EMPTY (XBRE = 1)
4. WAIT FOR TMS9902A TRANSMIT SHIFT REGISTER TO BE EMPTY (XSRE = 1)
5. A CARRIAGE RETURN DELAY MAY BE NEEDED FOR SOME MECHANICAL PRINTHEADS

(a) Flow Chart

FIGURE 5-6. TRANSMIT CHARACTER(S) BY POLLING
(Sheet 1 of 2)

```
          *INITIALIZE REGISTERS
                  CLR   R2              INITIALIZE TEMPORARY STORAGE
                  LI    R12,>0300       CRU SOFTWARE BASE ADDRESS
                  LI    R1,CHARS        STORAGE FOR CHARACTERS TO BE SENT
                  .
                  .
                  .

NOTE 1    LOOP6   EQU   $
                  LI    R3,DELAY        LOOP COUNT FOR CR 175 MS DELAY
                  SBO   16              SET RTS TO ONE
          LOOP1   TB    22              IS TRANSMIT BUFFER EMPTY?
                  JNE   LOOP1           NO, WAIT UNTIL EMPTY
                  LDCR  *R1,8           CHARACTER TO TMS 9902A VIA CRUOUT
          LOOP2   TB    22              IS TRANSMIT BUFFER EMPTY?
                  JNE   LOOP2           NO, WAIT UNTIL CHARACTER MOVED
          LOOP3   TB    23              IS TRANSMIT SHIFT REGISTER EMPTY?
                  JNE   LOOP3           NO, WAIT UNTIL CHARACTER SENT
                  MOVB  *R1+,R2         MOVE CHARACTER TO R2
NOTE 1            CI    R2,>0D00        WAS IT A CARRIAGE RETURN?
NOTE 1            JNE   NEXT            NO, SKIP EXTRA 175 MS DELAY
NOTE 1    LOOP4   DEC   R3              DECREMENT COUNTER FOR 175 MS
                  JNE   LOOP4           LOOP FOR 175 MS
          NEXT    CI    R2,0            WAS THAT LAST CHARACTER = 00?
                  JNE   LOOP6           NO, GET NEXT CHARACTER
                   .
                   .
                   .
```

NOTES

1. These four source lines are needed only if a mechanical printhead mechanism is used requiring a carriage return delay.

2. Count value in DELAY will vary depending upon the processor used. Please compute this value using your data manual. For example, a decrement/loop count of 26,253 will cause a 175 ms delay using a TMS 9900 processor at 3 MHz and a count of 7 decremented to 0 using a TMS 99110 processor at 6 MHz.

(b) Assembly Language Coding

FIGURE 5-6. TRANSMIT CHARACTER(S) BY POLLING
(Sheet 2 of 2)

5.4.3 Programming the Interval Timer (Figure 5-7)

A flow chart for execution of the TMS 9902A interval timer is shown in Figure 5-7. Circled numbers next to the various steps show user programming interaction with the TMS 9902A; unnumbered steps are internal to the TMS 9902A.

In the example below, an interrupt is issued by the TMS 9902A when the timer counts down to zero. This interrupt signal is programmed as a level-seven interrupt; thus, interrupt-level jumper J12 must be set E54-E72 on the TM 990/303B module to set up INT7- on the system bus to the microcomputer module.

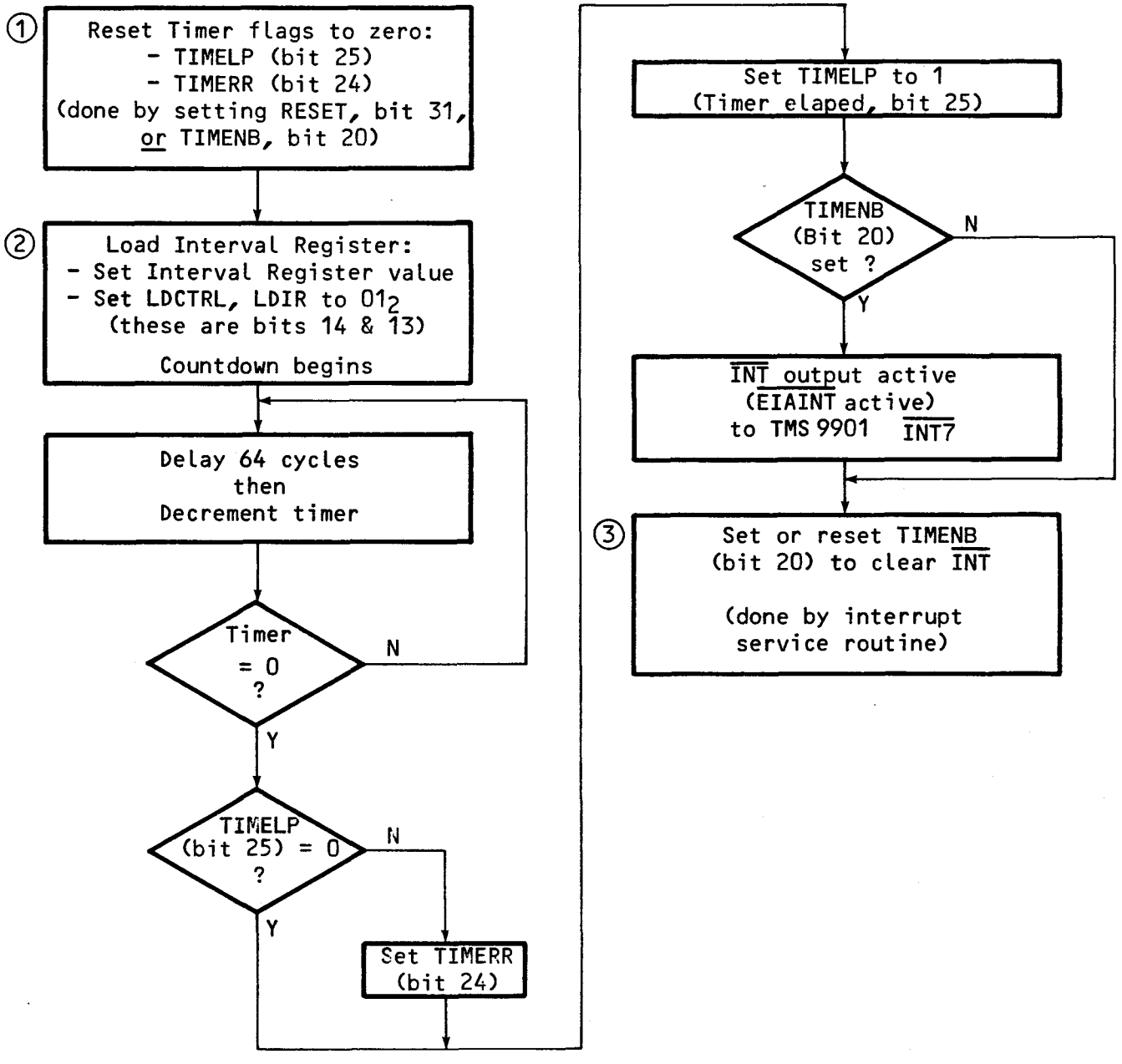
This example sets up the TMS 9902A as well as the microcomputer module's interrupt controller (a TMS 9901) to receive an interrupt level 7 from the TM 990/303B. The controller module's TMS 9902A is set up as follows:

- Reset the TMS 9902A which causes
  - TIMELP flag to be reset (timer elapsed, bit 25)
  - TIMERR flag to be reset (timer error, bit 24)
  - LDCTRL and LDIR bits to be set to ones (bits 14 and 13)
- Set up interval value for 15 milliseconds in R2 (using 3 MHz clock)
- Set TIMENB at bit 20 which allows an interrupt issued at pin INT- of the TMS 9902A when the Interval Register counts down to zero.
- Load Interval Register which also starts the countdown.

```
*RESET TMS 9902, SET COUNTER, AND ENABLE ISSUANCE OF INTERRUPT
*NOTE: INTERNAL CLOCK PERIOD SET AT CONTROL REGISTER FOR 1 US (3 MHZ CLOCK)
       LI    R12,TM9902      PLACE TMS 9902A SFTWR BASE ADDR IN R12 (>0300)
       SBO   31              RESET (SETS LDCTRL, LDIR; RESETS TIMELP, TIMERR)
       NOP                   NOP AND LI FOR 11 CYCLE DELAY ON TMS 9900
       LI    R2,>EA00        15-MILLISECOND TIMER VALUE IN R2
       SBZ   14              RESET LDCTRL: SET UP FOR LOADING INTERVAL REG.
       SBO   20              SET TIMENB (ALLOWS INT- TO BUS & CPU MODULE)
       LDCR  R2,8            8 MSB'S OF R2 TO INTERVAL REG.: START COUNTDOWN
*ENABLE INT7 AT CPU MODULE'S TMS 9901 (INT7 JUMPERED AT TM 990/303B)
       LI    R12,TM9901      PLACE TMS 9901 SFTWR BASE ADDR IN R12
       SBZ   0               ENTER INTERRUPT MODE
       SBO   7               ENABLE INTERRUPT 7
*ENABLE INTERRUPT 7 AT TMS 9900
       LIMI  7               ENABLE INTERRUPTS 1 TO 7 AT TMS 9900
```

When the countdown occurs, INT- from the TMS 9902A becomes active low to INT7- of the bus and the TMS 9901 at the microcomputer module. This interrupt is enabled at the TMS 9901 and TMS 9900, and the interrupt-7 vectors (at memory addresses $001C_{16}$, $001E_{16}$) contain the WP and PC values to trap to an interrupt service routine. Included in this routine should be code to disable (or reset) the interrupt at the TMS 9901 and reprogram the timer at the TMS 9902A.

If the timer counts to zero and remains enabled, its Interval Register is automatically reloaded with the initial countdown value and the countdown starts over again. If it counts down to zero again and TIMELP (read-bit 25) has not been reset, another interrupt will be issued; however, TIMERR (read-bit 24) will also be active. TIMERR means an interrupt occurred while TIMELP was also active (was not reset after becoming active from a previous countdown). Writing to bit 20 (TIMENB) resets both TIMELP and TIMERR. However, writing a <u>one</u> to bit 20 will also cause pin INT- to be active at the next countdown; writing a <u>zero</u> will prevent INT- from being active.



NOTE: Flowchart symbols with circled numbers are user program inputs; others are internal to the TMS 9902A.

FIGURE 5-7. LOADING AND EXECUTING THE INTERVAL TIMER USING INTERRUPTS

5.4.4 Example Interrupt Operation to Receive Characters (Figure 5-8)

Figure 5-8 uses interrupts to receive and store characters sent to the TMS 9902A. Although the TMS 9902A can be polled for character receipt, interrupt operation allows the processor to do other chores in between receipts of characters. Several factors must be considered in interrupt operations:

- Interrupt level must be enabled at microcomputer module's interrupt controller (e.g., TMS 9901) and in the microprocessor's interrupt mask. In this example (shown in Figure 5-8), interrupt level 7 is enabled at a TMS 9900 and TMS 9901.

- On the TM 990/303B, jumper J12 for interrupt level 7 (E54 to E72).

- Vectors must be present in lower memory for the interrupts used. Since the example in Figure 5-8 uses interrupt level 7 for the TMS 9902A, WP and PC vectors must be already loaded in memory addresses (hex) 001C and 001E in order to branch to the interrupt service routine.

- Four events can cause interrupts (if enabled) to be issued by the TMS 9902A (shown in TMS 9902A data manual.) These are:

  a) Receipt of a character (Receive Buffer Full).
  b) Character is ready to be transmitted (character shifted from Transmit Buffer Register to Transmit Shift Register).
  c) Interval Timer counts down to zero.
  d) DSR- or CTS- inputs change (data set change) at P2-20 or P2-8.

  In the example in Figure 5-8, event marked (a) above is used to cause an interrupt. This occurs when a character is sensed at pin RIN of the TMS 9902A. In this case, the receiver-interrupt enable bit must already be set at the TMS 9902A (bit 18).

- The interrupt service routine should reset (re-enable) the interrupt at the TMS 9902A to prevent it from being continuously requested. When the interrupt is reset at the TMS 9902A, it will go inactive at the TMS 9901 output. The TMS 9902A interrupt is reset by writing a one or zero to bit 18 (RIENB); a one will also rearm the interrupt circuitry while a zero will disable the character-received interrupt.

- Before the interrupt service routine is exited, it should restore the interrupt conditions desired. For example, if it is desired that the next character received should cause an interrupt:

  - at the TMS 9901, re-enable (if disabled) the interrupt (INT7-),

  - at the TMS 9902A, turn off RBRL (Receive Buffer Register loaded) and re-enable the receiver interrupt; both can be done by writing a one to bit 18 (RIENB).

As shown in the Figure 5-8 example and described in the TMS 9902A data manual, the receiver interrupt is enabled by writing a one to bit 18 (RIENB) on the TMS 9902A. In this example, an interrupt will be issued from the TMS 9902A (INT- low) to the INT7- input on the TMS 9901. Before the interrupt can be serviced, INT7- must have been enabled at the TMS 9901 on the CPU module and at jumper J12 on the TM 990/303B.

```
*SET UP FOR INTERRUPT IN MAIN BODY OF PROGRAM
**CHARACTER STORAGE AREA
COUNTR DATA 0              CHARACTER STORAGE COUNT ACCUMULATOR
STORE  BSS  200            RESERVE 200 BYTES
       .
       .
       .

**ROUTINE TO ENABLE INTERRUPT 7 AT TMS 9901
       LI   R12,>0100      ADDRESS TMS 9901 ON MICROCOMPUTER MODULE
       SBZ  0              ENTER INTERRUPT MODE
       SBO  7              ENABLE INTERRUPT 7 AT TMS 9901
       LIMI 7              ENABLE INTERRUPT 7 AT TMS 9900
**ROUTINE TO ENABLE THE ISSUING OF AN INTERRUPT FROM TMS 9902A
**WHEN A CHARACTER IS RECEIVED AT THE TMS 9902A
       LI   R12,>0300      ADDRESS OF TMS 9902A
       SBO  18             ENABLE RECEIVER INTERRUPT ENABLE BIT (BIT 18)
**ROUTINE TO PLACE STORAGE-AREA ADDRESS IN R0 OF INTERRUPT SERVICE ROUTINE
       MOV  @>001C,R1      PLACE ADDRESS OF INTERRUPT WP START IN R1
       LI   R2,STORE       PLACE ADDRESS OF STORAGE START IN MAIN PROGM R2
       MOV  R2,*R1         MOVE STORAGE START TO R0 OF INTERRUPT SERV. RTN.
       CLR  @COUNTR        CLEAR CHARACTER COUNTER
       .
       .
       .
```

(1) Set Up Main Program for Interrupt

FIGURE 5-8. EXAMPLE PROGRAM USING INTERRUPTS TO RECEIVE CHARACTERS
(Sheet 1 of 2, see next page)

Figure 5-8 contains example code to use interrupts to receive characters at the EIA port. This example shows the major considerations of using interrupts, and may be expanded as needed by the user. This example is not intended as a complete example of an interrupt service routine (e.g., not shown are full error checking routines, a string termination character check, etc.).

Sheet 1 of the figure contains code to enable interrupt level 7 at the TMS 9901 and TMS 9900 on the microcomputer module. It also loads an address into register 0 of the interrupt service routine; this register is located by using the workspace-pointer vector in lower memory (address $001C_{16}$). A counter used by the interrupt service routine is also set to zero.

```
*INTERRUPT SERVICE ROUTINE
**TURN OFF RECEIVER INTERRUPT AT TMS 9902A
       LI   R12,>0300      ADDRESS CONTROLLER TMS 9902A
       SBZ  18             SET BIT 18 TO ZERO TO DISABLE RCVER INTERRUPT
**CHECK FOR ERRORS, IF NONE, RECEIVE AND STORE CHARACTER
       TB   21             IS CHARACTER IN RECEIVE BUFFER OF 9902A?
       JNE  ERROR1         NO, CHARACTER NOT RECEIVED, CHECK FOR ERROR
       TB   9              YES; DID ERROR OCCUR IN CHARACTER RECEIPT?
       JEQ  ERROR2         YES, GO TO ERROR ROUTINE
       STCR R1,8           NO, CHARACTER RECEIVED OK: MOVE TO 8 MSB OF R1
       JMP  STOR           GO STORE CHARACTER RECEIVED IN PORT BUFFER AREA
**ERROR ROUTINES FOR CHARACTER RECEIPT
ERROR1
       .
       .
       .
ERROR2
       .
       .
       .

**STORE CHARACTER, KEEP TRACK OF CHARACTER COUNT
STOR   MOVB R1,*R0+        MOVE CHARACTER TO STORAGE AREA
       INC  @COUNTR        INCREMENT CHARACTER QTY COUNTER
**SET UP TO EXIT INTERRUPT SERVICE ROUTINE
***AT TMS 9902A
       LI   R12,>0300      TMS 9902A CRU BASE ADDRESS
       SBO  18             BIT 18 TO ONE TO RE-ENABLE RECEIVER INTERRUPT
       RTWP
```

(2) Major Parts of Interrupt Service Routine

FIGURE 5-8. EXAMPLE PROGRAM USING INTERRUPTS TO RECEIVE CHARACTERS (Sheet 2 of 2)

Sheet 2 of Figure 5-8 shows the interrupt service routine. Not shown is the code for the error checking routines. After receipt, the character is stored.

5.4.5 Program TMS 9902A, Then Echo Characters (Figure 5-9)

This example program has two main parts:

- A setup area where the TMS 9902A is programmed to receive and send characters and where a means is made so that the interrupt vectors for interrupt 7 will point to the interrupt service routine.

- The interrupt service routine that answers an interrupt caused by a key being pressed on a terminal attached to port P2; the character is received and then transmitted back (echoed) to the terminal screen or printer.

Figure 5-9 shows the code to execute this. Its principal parts are explained below, keyed to source lines in the figure (leftmost column in the figure). To load this program into memory, enter the object code in the third column into sequential memory addresses; however, add a bias to relocatable code which is identified by a tic mark such as 0200'. For example, if the program is loaded at A000, then this value must be added to relocatable and the 0200' becomes A200. Select interrupt 7 at jumper 12 and 0300 base address at jumper J13.

- Source lines 0012 to 0021: Set up a means so that the vectors for interrupt 7 point to the interrupt service routine. The interrupt 7 PC vector is at memory address $001E_{16}$. To have this vector point to the interrupt service routine beginning at INTRP, a BLWP routine is placed in memory so that the interrupt 7 PC entry address points to it. This BLWP branches to the two vectors at WP, in the first two lines of the program. These vectors point to a workspace 200 bytes away and to INTRP, the beginning of the interrupt service routine.

- Source lines 0027 to 0038: Program the TMS 9902A Control Register for:

  - 7-bit character length
  - 3 MHz clock
  - Even parity
  - 2 stop bits

  (Note: The terminal used must also be set to these values.)

  Also programs the Receive and Transmit Data Rate Registers for 1200 baud by transferring $01A0_{16}$ to both registers. Values for baud rates at 3 MHz (source line 0030) include:

Baud Rate	Hex Value for Data Rate Registers
110	638
300	4D0
600	340
1200	1A0
2400	D0
4800	68
9600	34
19200	1A

  Source line 0038 enables bit RIENB so that an interrupt will be issued from the TMS 9902A's INT- pin to the jumpered interrupt level 7 at jumper J12. With the interrupt enabled, a high-to-low transition on pin RIN will enable the receiver circuitry, receive the character from the terminal, and bring INT- low to issue an interrupt.

- Source lines 0042 to 0046: The first sheet of Figure 5-9 sets up the TMS 9902A, then sets up interrupts and places the program into a loop, waiting for an interrupt caused by a character received. Lines 0042 to 0046 set up to receive an interrupt 7 at the CPU module's TMS 9901 and TMS 9900, then goes into a loop (JMP $ in line 0046).

When character receipt is sensed at the TMS 9902A (high-to-low transition on pin RIN), the character is received and pin INT- on the TMS 9902A goes low. This low value goes to jumper J12 where the interrupt level is selected to the CPU module. In Figure 5-9, interrupt level 7 is assumed. Interrupt level 7 uses the vector at address $001C_{16}$ for the workspace pointer and $001E_{16}$ for the program counter. The PC vector points to a three-word program (shown in source lines 0085 and 0086 and is loaded into memory via source lines 0017 to 0021 in order to be executed by an interrupt 7).

```
0085    006A    0420    BLWP @WP
        006C    0000'
0086    006E    0380    RTWP
```

The BLWP instruction directs execution to the interrupt service routine via the BLWP vectors at location WP (source lines 0012 and 0013). These vectors direct execution to location INTRP (source line 0052) which is the beginning of the interrupt service routine.

- Source lines 0050 to 0058: These lines begin the interrupt service routine. They set the CRU base address of the TMS 9902A and test that a receive interrupt occurred. If so, control is given to receive the interrupt; if not, an error occurs since only receive-character interrupts are enabled, and control exits to the monitor.

- Source lines 0064 to 0068: These lines set up register 2 to receive the character, then check to see if it is caused by an ESCape key (ASCII $1B_{16}$). If so, execution goes to the monitor.

- Source lines 0072 to 0081: These lines transmit the character back to the terminal (echo it). These lines monitor the character through the TMS 9902A registers (first to the Transmit Buffer Register then finally to the Transmit Shift Register). When it exits to the terminal, the RTWP returns execution to the executed program (JMP $ on source line 0046).

- Source lines 0085 and 0086: These lines represent a program that is placed into memory by source lines 0017 to 0021. This small program is pointed to by the interrupt vectors and is the first code executed when the interrupt becomes active. In essence, it points the interrupt to the interrupt service routine (which begins at source line 0052) via the vectors at source lines 0012 and 0013.

```
0001                    *      THIS PROGRAM RECEIVES A CHARACTER FROM A TERMINAL     *
0002                    *      AT 1200 BAUD AND TRANSMITS IT BACK TO THE SCREEN.     *
0003                    *      THIS PROGRAM USES CRU SOFTWARE BASE ADDRESS OF        *
0004                    *      HEX 0300 WHICH MUST BE JUMPERED AT J13. ALSO, IT      *
0005                    *      USES EIA INTERRUPT 7 WHICH MUST BE JUMPERED AT J12.   *
0006                    *      A 3 MHZ SYSTEM IS PRESUMED.    J.J.W. 9/23/82        *
0007                    ************************************************************
0008                           IDT  'ECHO'
0009                    ************************************************************
0010                    *  SET UP INTERMEDIATE INTERRUPT AREA FOR INTERRUPT SEVEN
0011                    *  1. VECTOR VALUES FOR INTERRUPT BLWP
0012 0000 0200'  WP     DATA $+>200       WORKSPACE OF INTERRUPT SERV ROUTN
0013 0002 003E'  PC     DATA INTRP        BEGINNING OF INTERRUPT SERV ROUTN
0014                    *  2. ROUTINE TO SET UP INTERRUPT BRANCH IN ORDER TO BRANCH
0015                    *     FROM INTERRUPT TRAP VECTORS TO THE PROGRAM
0016 0004 02E0   START  LWPI VECTOR+6     WORKSPACE FOLLOWS PROGRAM AREA
     0006 0070'
0017 0008 0201          LI   R1,VECTOR    ADDRESS OF THREE VECTOR WORDS
     000A 006A'
0018 000C C0A0          MOV  @>1E,R2      INTERRUPT 7 PC VECTOR TO R2
     000E 001E
0019 0010 CCB1          MOV  *R1+,*R2+    MOVE BLWP OBJECT TO PC DESTINATION
0020 0012 CCB1          MOV  *R1+,*R2+    MOVE VECTOR OBJECT TO PC DEST+2
0021 0014 C491          MOV  *R1,*R2      MOVE RTWP OBJECT TO PC DEST+4
0022                    ************************************************************
0023                    *  SET UP TMS 9902A REGISTER VALUES
0024                    *  1. ROUTINE TO SET UP CONTROL REGISTER ON TMS 9902A
0025                    *     SET UP FOR:  *7-BIT CHARACTER LENGTH    *EVEN PARITY
0026                    *                  *CLOCK DIVIDE OF 3         *2 STOP BITS
0027 0016 0201          LI   R1,>6200     8 CONTROL REG. VALUES IN LEFT BYTE
     0018 6200
0028                    *  2. ROUTINE TO SET UP RECEIVE/TRANSMIT BAUD RATE OF 1200
0029 001A 0202          LI   R2,>1A0      SET UP FOR 1200 BAUD (3 MHZ SYSTEM)
     001C 01A0
0030                    *  3. PROGRAM THE PRINCIPAL TMS 9902A REGISTERS
0031 001E 020C          LI   R12,>0300    TMS 9902A SOFTWARE BASE ADDRESS
     0020 0300
0032 0022 1D1F          SBO  31           RESET TMS 9902A/SET FOR REGISTER LOAD
0033 0024 1000          NOP               TWO NO-OPS NEEDED TO CAUSE DELAY
0034 0026 1000          NOP                 OF AT LEAST 11 CLOCK CYCLES
0035 0028 3201          LDCR R1,8         LOAD CONTROL REGISTER, RESET BIT 1
0036 002A 1E0D          SBZ  13           RESET BIT 13, LDIR
0037 002C 3302          LDCR R2,12        LOAD RCV/XMT REGS
0038 002E 1D12          SBO  18           SET RIENB TO ALLOW RECEIVE-CHAR INTRP
0039                    ************************************************************
0040                    *  ENABLE INTERRUPTS AT CPU MODULE (AT TMS 9901 & TMS 9900)
0041                    *  AND WAIT FOR INTERRUPT FROM KEYBOARD INPUT
0042 0030 020C          LI   R12,>0100    9901 SFWR BASE ADDR (INTERRUPT HNDLR)
     0032 0100
0043 0034 1E00          SBZ  0            ENTER INTERRUPT MODE ON 9901
0044 0036 1D07          SBO  7            ENABLE INTERRUPT SEVEN AT TMS 9901
0045 0038 0300          LIMI 7            ENABLE INTERRUPT SEVEN AT TMS 9900
     003A 0007
0046 003C 10FF          JMP  $            WAIT HERE FOR INTERRUPT TO OCCUR
```

FIGURE 5-9. CODING TO SET UP TMS 9902A AND ECHO CHARACTERS (Page 1 of 2)

```
0047                ************************************************************
0048                *  INTERRUPT SERVICE ROUTINE CHECKS FOR CHARACTER RECEIVED
0049                ************************************************************
0050                *  INTERRUPT COMING FROM THE TMS 9902A (I.E.,
0051                *  A CHARACTER HAS BEEN RECEIVED FROM KEYBOARD)
0052 003E 020C     INTRP  LI   R12,>0300    TMS 9902A CRU BASE ADDRESS IN R12
     0040 0300
0053 0042 1F15            TB   21           CHARACTER BEEN RECEIVED (RBRL=1)?
0054 0044 1302            JEQ  RCVE         YES, GO RECEIVE CHARACTER
0055                *  THE HEX 0080 VALUE IS THE MONITOR STARTING ADDRESS FOR THE
0056                *  TM 990/401-3 TIBUG MONITOR. THIS VALUE MAY HAVE TO BE
0057                *  CHANGED IF ANOTHER MONITOR IS USED.
0058 0046 0460     EXIT   B    @>0080       BIT 21 NOT SET, GO TO MONITOR
     0048 00A0
0059                ************************************************************
0060                *  BIT 21 (RBRL) = 1 = CHARACTER RECEIVED
0061                *  INTERRUPT SERVICE ROUTINE TO RECEIVE ONE CHARACTER FROM
0062                *  TERMINAL AND CHECK THIS FOR PROPER CHARACTER
0063                ************************************************************
0064 004A 04C2     RCVE   CLR  R2           CLEAR CHARACTER RECEIVE AREA
0065 004C 3602            STCR R2,8         STORE CHARACTER IN LEFT BYTE R2
0066 004E 0282            CI   R2,>1B00     WAS ESCAPE KEY PRESSED?
     0050 1B00
0067 0052 13F9            JEQ  EXIT         YES, GO TO MONITOR
0068 0054 1000            JMP  TRANS        NO, TRANSMIT CHARACTER RECEIVED
0069                ************************************************************
0070                *  ROUTINE TO TRANSMIT CHARACTERS
0071                *  CRU SOFTWARE BASE ADDRESS SET IN R12
0072 0056 1D10     TRANS  SBO  16           SET RTS TO ONE
0073 0058 1F16     LOOP1  TB   22           IS TRANSMIT BUFFER EMPTY?
0074 005A 16FE            JNE  LOOP1        NO, WAIT UNTIL EMPTY
0075 005C 3202            LDCR R2,8         YES, APPLY CHARACTER TO TMS 9902A
0076 005E 1F16     LOOP2  TB   22           IS TRANSMIT BUFFER EMPTY?
0077 0060 16FE            JNE  LOOP2        NO, WAIT UNTIL CHARACTER MOVED
0078 0062 1F17     LOOP3  TB   23           IS TRANSMIT SHIFT REGISTER EMPTY?
0079 0064 16FE            JNE  LOOP3        NO, WAIT UNTIL CHARACTER SENT
0080 0066 1D12     LOOP6  SBO  18           RE-ENABLE CHAR RCV INT, RESET BIT
0081 0068 0380            RTWP              RETURN TO CALLING PROGRAM
0082                ************************************************************
0083                ************************************************************
0084                *  VECTORS TO BE BRANCHED TO BY LEVEL-SEVEN INTERRUPTS
0085 006A 0420     VECTOR BLWP @WP          FIRST TWO WORDS, BRANCH TO PGM START
     006C 0000'
0086 006E 0380            RTWP              THIRD WORD, RETURN TO INTERRUPTED PGM
0087                *                        (THIS WAS THE JMP $ AT LOCATION 003C)
```

FIGURE 5-9. CODING TO SET UP TMS 9902A AND ECHO CHARACTERS (Page 2 of 2)

APPENDIX A

DRIVE PARAMETER LIST ENTRIES AND DISK DRIVE JUMPERING

TI cannot assume any responsibility for the information in the following pages or guarantee that it is has not been changed by the various disk drive manufacturers. This section is provided as an aid; see the respective disk drive user's manual for detailed information.

A.1 INTRODUCTION

This appendix has three major sections:

- Drive parameter tables, by format type
- Drive parameters, by drive
- Recommended jumper settings, by drive

The following is a list of the drives covered, and the section and page where the drive information is located. Table A-1 lists the dc power requirements for the disk drives.

TABLE A-1. POWER REQUIREMENTS FOR DISK DRIVES

Company and Model	+5 V		-5 V		+12 V		+24 V		Comment
	Typ	Max	Typ	Max	Typ	Max	Typ	Max	
Eight-Inch Disk Drives (values in amps)									
CDC 9404B	0.7						1.3		
CDC 9406-4	0.6						0.7		
Qume DT-8	0.9	1.3					0.7	1.0	
Shugart SA800	0.8	1.0	0.05	0.07			1.3	1.7	
Shugart SA801	0.8	1.0	0.05	0.07			1.3	1.7	
Shugart SA850/851	1.0	1.1					0.85	1.0	
Siemens FDD100-8		1.0						1.8	
Siemens FDD200-8		1.0						1.8	
Tandon TM848-1	1.0						1.0		
Tandon TM848-2	1.0						1.0		
Five (5¼) Inch Disk Drives (values in amps)									
BASF 6106	0.7				1.75	2.4			Note 1
CDC 9408	0.5	0.7			0.9	1.8			Note 1
CDC 9409T	0.5	0.7			0.9	1.8			Note 1
Qume DT-5	0.5				1.35				Note 1
Shugart SA400	0.5	0.7			0.9	1.8			
Shugart SA450	0.5	0.7			0.9	1.8			
Shugart SA410	0.5	0.7			1.2	2.0			
Shugart SA460	0.5	0.7			1.2	2.0			
Siemens FDD100-5	0.4	0.7			0.8	1.8			Note 1
Siemens FDD200-5	0.4	0.7			0.8	1.8			Note 1
Siemens FDD196-5	0.4				0.8				
Siemens FDD296-5	0.4				0.8				
Tandon TM100-1		0.6				0.9			
Tandon TM100-3		0.6				0.9			Note 2
Tandon TM100-4		0.6				0.9			Note 2

Note 1: 100 mV p-p ripple on 12 V and 50 mV p-p ripple on 5 V.
Note 2: 100 mV p-p ripple on 5 V.

The following twelve tables list the suggested drive parameters, by format type for both 8 inch and 5¼ inch disk drives.

TABLE A-2. IBM 8-INCH SS SD DRIVE PARAMETERS

DRIVE	WORD0	WORD1	WORD2	WORD3	WORD4	WORD5	WORD6	WORD7
CDC 9404B	0101	004D	0ABE	05DC	1770	03E8	0000	6880
CDC 9406-4	9101	004D	012C	05DC	0DAC	03E8	0000	6880
QUME DT-8	0101	004D	012C	05DC	0DAC	03E8	0000	6880
SHUGART SA800	0101	004D	0320	05DC	0DAC	03E8	0000	6880
SHUGART SA801	0101	004D	0320	0320	0DAC	03E8	0000	6880
SHUGART SA851/850	9101	004D	012C	05DC	1388	03E8	0000	6880
SIEMENS FDD100-8	9101	004D	0320	0578	09C4	03E8	0000	6880
SIEMENS FDD200-8	9101	004D	0258	0460	09C4	03E8	0000	6880
TANDON TM848-1	0181	004D	012C	05DC	0002	4E20	0000	6880
TANDON TM848-2	0181	004D	012C	05DC	0002	4E20	0000	6880

NOTES

1. All values are hexadecimal.
2. MOTORON is specified for Tandon drives in word 0.
3. The head unload timeout specified for the Tandon drives is 20 seconds. The drives, as shipped from the factory, will run for 20 seconds after an access unless jumpered otherwise.

TABLE A-3. IBM 8-INCH DS SD DRIVE PARAMETERS

DRIVE	WORD0	WORD1	WORD2	WORD3	WORD4	WORD5	WORD6	WORD7
CDC 9404B								
CDC 9406-4	9102	004D	012C	05DC	0DAC	03E8	0000	6880
QUME DT-8	0102	004D	012C	05DC	0DAC	03E8	0000	6880
SHUGART SA800								
SHUGART SA801								
SHUGART SA851/850	9102	004D	012C	05DC	1388	03E8	0000	6880
SIEMENS FDD100-8								
SIEMENS FDD200-8	9102	004D	0258	0460	09C4	03E8	0000	6880
TANDON TM848-1								
TANDON TM848-2	0182	004D	012C	05DC	0002	4E20	0000	6880

NOTES

1. All values are hexadecimal.
2. MOTORON is specified for Tandon drives in word 0.
3. The head unload timeout specified for the Tandon drives is 20 seconds. The drives, as shipped from the factory, will run for 20 seconds after an access unless jumpered otherwise.

TABLE A-4. IBM 8-INCH SS DD DRIVE PARAMETERS

DRIVE	WORD0	WORD1	WORD2	WORD3	WORD4	WORD5	WORD6	WORD7
CDC 9404B								
CDC 9406-4	A101	004D	012C	05DC	0DAC	03E8	1000	6900
QUME DT-8	9101	004D	012C	05DC	0DAC	03E8	1000	6900
SHUGART SA800	9101	004D	0320	05DC	0DAC	03E8	1000	6900
SHUGART SA801	9101	004D	0320	0320	0DAC	03E8	1000	6900
SHUGART SA851/850	9101	004D	012C	05DC	1388	03E8	1000	6900
SIEMENS FDD100-8	B101	004D	0320	0578	09C4	03E8	1000	6900
SIEMENS FDD200-8	B102	004D	0258	0460	09C4	03E8	1000	6900
TANDON TM848-1	A181	004D	012C	05DC	0002	4E20	1000	6900
TANDON TM848-2	9181	004D	012C	05DC	0002	4E20	1000	6900

NOTES

1. All values are hexadecimal.
2. MOTORON is specified for Tandon drives in word 0.
3. The head unload timeout specified for the Tandon drives is 20 seconds. The drives, as shipped from the factory, will run for 20 seconds after an access unless jumpered otherwise.

TABLE A-5. IBM 8-INCH DS DD DRIVE PARAMETERS

DRIVE	WORD0	WORD1	WORD2	WORD3	WORD4	WORD5	WORD6	WORD7
CDC 9404B								
CDC 9406-4	A102	004D	012C	05DC	0DAC	03E8	1000	6900
QUME DT-8	9102	004D	012C	05DC	0DAC	03E8	1000	6900
SHUGART SA800								
SHUGART SA801								
SHUGART SA851/850	9102	004D	012C	05DC	1388	03E8	1000	6900
SIEMENS FDD100-8								
SIEMENS FDD200-8	B102	004D	0258	0460	09C4	03E8	1000	6900
TANDON TM848-1								
TANDON TM848-2	9182	004D	012C	05DC	0002	4E20	1000	6900

NOTES

1. All values are hexadecimal.
2. MOTORON is specified for Tandon drives in word 0.
3. The head unload timeout specified for the Tandon drives is 20 seconds. The drives, as shipped from the factory, will run for 20 seconds after an access unless jumpered otherwise.

TABLE A-6. TI 8-INCH SS DD DRIVE PARAMETERS

DRIVE	WORD0	WORD1	WORD2	WORD3	WORD4	WORD5	WORD6	WORD7
CDC 9404B								
CDC 9406-4	A101	004D	012C	05DC	0DAC	03E8	1100	6920
QUME DT-8	9101	004D	012C	05DC	0DAC	03E8	1100	6920
SHUGART SA800	9101	004D	0320	05DC	0DAC	03E8	1100	6920
SHUGART SA801	9101	004D	0320	0320	0DAC	03E8	1100	6920
SHUGART SA851/850	9101	004D	012C	05DC	1388	03E8	1100	6920
SIEMENS FDD100-8	B101	004D	0320	0578	09C4	03E8	1100	6920
SIEMENS FDD200-8	B102	004D	0258	0460	09C4	03E8	1100	6920
TANDON TM848-1	A181	004D	012C	05DC	0002	4E20	1100	6920
TANDON TM848-2	9181	004D	012C	05DC	0002	4E20	1100	6920

NOTES

1. All values are hexadecimal.
2. MOTORON is specified for Tandon drives in word 0.
3. The head unload timeout specified for the Tandon drives is 20 seconds. The drives, as shipped from the factory, will run for 20 seconds after an access unless jumpered otherwise.

TABLE A-7. TI 8-INCH DS DD DRIVE PARAMETERS

DRIVE	WORD0	WORD1	WORD2	WORD3	WORD4	WORD5	WORD6	WORD7
CDC 9404B								
CDC 9406-4	A102	004D	012C	05DC	0DAC	03E8	1100	6920
QUME DT-8	9102	004D	012C	05DC	0DAC	03E8	1100	6920
SHUGART SA800								
SHUGART SA801								
SHUGART SA851/850	9102	004D	012C	05DC	1388	03E8	1100	6920
SIEMENS FDD100-8								
SIEMENS FDD200-8	B102	004D	0258	0460	09C4	03E8	1100	6920
TANDON TM848-1								
TANDON TM848-2	9182	004D	012C	05DC	0002	4E20	1100	6920

NOTES

1. All values are hexadecimal.
2. MOTORON is specified for Tandon drives in word 0.
3. The head unload timeout specified for the Tandon drives is 20 seconds. The drives, as shipped from the factory, will run for 20 seconds after an access unless jumpered otherwise.

TABLE A-8. IBM 8-INCH MODIFIED SS DD DRIVE PARAMETERS

DRIVE	WORD0	WORD1	WORD2	WORD3	WORD4	WORD5	WORD6	WORD7
CDC 9404B								
CDC 9406-4	A101	004D	012C	05DC	0DAC	03E8	3000	6900
QUME DT-8	9101	004D	012C	05DC	0DAC	03E8	3000	6900
SHUGART SA800	9101	004D	0320	05DC	0DAC	03E8	3000	6900
SHUGART SA801	9101	004D	0320	0320	0DAC	03E8	3000	6900
SHUGART SA851/850	9101	004D	012C	05DC	1388	03E8	3000	6900
SIEMENS FDD100-8	B101	004D	0320	0578	09C4	03E8	3000	6900
SIEMENS FDD200-8	B102	004D	0258	0460	09C4	03E8	3000	6900
TANDON TM848-1	A181	004D	012C	05DC	0002	4E20	3000	6900
TANDON TM848-2	9181	004D	012C	05DC	0002	4E20	3000	6900

NOTES

1. All values are hexadecimal.
2. MOTORON is specified for Tandon drives in word 0.
3. The head unload timeout specified for the Tandon drives is 20 seconds. The drives, as shipped from the factory, will run for 20 seconds after an access unless jumpered otherwise.
4. IBM Modified DD is specified in this table in word 6.

TABLE A-9. IBM 8-INCH MODIFIED DS DD DRIVE PARAMETERS

DRIVE	WORD0	WORD1	WORD2	WORD3	WORD4	WORD5	WORD6	WORD7
CDC 9404B								
CDC 9406-4	A102	004D	012C	05DC	0DAC	03E8	3000	6900
QUME DT-8	9102	004D	012C	05DC	0DAC	03E8	3000	6900
SHUGART SA800								
SHUGART SA801								
SHUGART SA851/850	9102	004D	012C	05DC	1388	03E8	3000	6900
SIEMENS FDD100-8								
SIEMENS FDD200-8	B102	004D	0258	0460	09C4	03E8	3000	6900
TANDON TM848-1								
TANDON TM848-2	9182	004D	012C	05DC	0002	4E20	3000	6900

NOTES

1. All values are hexadecimal.
2. MOTORON is specified for Tandon drives in word 0.
3. The head unload timeout specified for the Tandon drives is 20 seconds. The drives, as shipped from the factory, will run for 20 seconds after an access unless jumpered otherwise.
4. IBM Modified DD is specified in this table in word 6.

TABLE A-10. IBM 5-INCH SS SD DRIVE PARAMETERS

DRIVE	WORD0	WORD1	WORD2	WORD3	WORD4	WORD5	WORD6	WORD7
BASF 6106	0001	0028	0460	12C0	0DAC	03E8	0000	4080
CDC 9408	9001	0023	07D0	05DC	1388	03E8	0000	4080
CDC 9409T	8001	0050	01F4	05DC	1388	03E8	0000	4080
QUME DT-5	0001	0028	07D0	05DC	1388	03E8	0000	4080
SHUGART SA400	0001	0023	0FA0	03E8	1D4C	03E8	0000	4080
SHUGART SA450	0001	0028	09C4	05DC	1388	03E8	0000	4080
SHUGART SA410	8001	0050	0258	05DC	0002	03E8	0000	4080
SHUGART SA460	9001	0050	0258	05DC	0002	03E8	0000	4080
SIEMENS FDD100-5	0001	0028	09C4	05DC	1388	03E8	0000	4080
SIEMENS FDD200-5	0001	0028	09C4	05DC	1388	03E8	0000	4080
SIEMENS FDD196-5	9001	0050	07D0	05DC	1388	03E8	0000	4080
SIEMENS FDD296-5	0001	0050	0258	05DC	1388	03E8	0000	4080
TANDON TM100-1	9001	0028	01F4	05DC	0002	03E8	0000	4080
TANDON TM100-3	9001	0050	012C	05DC	1388	03E8	0000	4080
TANDON TM100-4	9001	0050	012C	05DC	1388	03E8	0000	4080

NOTE: All values in hexadecimal.

TABLE A-11. IBM 5-INCH DS SD DRIVE PARAMETERS

DRIVE	WORD0	WORD1	WORD2	WORD3	WORD4	WORD5	WORD6	WORD7
BASF 6106								
CDC 9408								
CDC 9409T	8002	0050	01F4	05DC	1388	03E8	0000	4080
QUME DT-5	0002	0028	07D0	05DC	1388	03E8	0000	4080
SHUGART SA400								
SHUGART SA450	0002	0028	09C4	05DC	1388	03E8	0000	4080
SHUGART SA410								
SHUGART SA460	9002	0050	0258	05DC	0002	03E8	0000	4080
SIEMENS FDD100-5								
SIEMENS FDD200-5	0002	0028	09C4	05DC	1388	03E8	0000	4080
SIEMENS FDD196-5								
SIEMENS FDD296-5	0002	0050	0258	05DC	1388	03E8	0000	4080
TANDON TM100-1								
TANDON TM100-3								
TANDON TM100-4	9002	0050	012C	05DC	1388	03E8	0000	4080

NOTE: All values in hexadecimal.

TABLE A-12. IBM 5-INCH SS DD DRIVE PARAMETERS

DRIVE	WORD0	WORD1	WORD2	WORD3	WORD4	WORD5	WORD6	WORD7
BASF 6106	9001	0028	0460	12C0	0DAC	03E8	1000	4100
CDC 9408	8001	0023	07D0	05DC	1388	03E8	1000	4100
CDC 9409T	9001	0050	01F4	05DC	1388	03E8	1000	4100
QUME DT-5	8001	0028	07D0	05DC	1388	03E8	1000	4100
SHUGART SA400	9001	0023	0FA0	03E8	1D4C	03E8	1000	4100
SHUGART SA450	9001	0028	09C4	05DC	1388	03E8	1000	4100
SHUGART SA410	9001	0050	0258	05DC	0002	03E8	1000	4100
SHUGART SA460								
SIEMENS FDD100-5	9001	0028	09C4	05DC	1388	03E8	1000	4100
SIEMENS FDD200-5	8001	0028	09C4	05DC	1388	03E8	1000	4100
SIEMENS FDD196-5								
SIEMENS FDD296-5								
TANDON TM100-1	9001	0028	01F4	05DC	0002	03E8	1000	4100
TANDON TM100-3	9001	0050	012C	05DC	1388	03E8	1000	4100
TANDON TM100-4	9001	0050	012C	05DC	1388	03E8	1000	4100

NOTE: All values in hexadecimal.

TABLE A-13. IBM 5-INCH DS DD DRIVE PARAMETERS

DRIVE	WORD0	WORD1	WORD2	WORD3	WORD4	WORD5	WORD6	WORD7
BASF 6106								
CDC 9408								
CDC 9409T	9002	0050	01F4	05DC	1388	03E8	1000	4100
QUME DT-5	8002	0028	07D0	05DC	1388	03E8	1000	4100
SHUGART SA400								
SHUGART SA450	9002	0028	09C4	05DC	1388	03E8	1000	4100
SHUGART SA410								
SHUGART SA460								
SIEMENS FDD100-5								
SIEMENS FDD200-5	8002	0028	09C4	05DC	1388	03E8	1000	4100
SIEMENS FDD196-5								
SIEMENS FDD296-5								
TANDON TM100-1								
TANDON TM100-3								
TANDON TM100-4	9002	0050	012C	05DC	1388	03E8	1000	4100

NOTE: All values in hexadecimal.

A.2 CDC 9404B EIGHT INCH DISK DRIVE

This section includes specifications, drive parameter lists, jumper locations, and jumper settings. For the most up-to-date information, consult your disk drive user's manual.

A.2.1 Specifications for Drive Parameter Lists

Word 0	Number of Sides: 1 Single Density Write Precompensation: none Double Density Write Precompensation: none Motor On with Drive Select: No, motor always on
Word 1	Maximum Number of Tracks per side: 77 004D (hex) (77 tracks total)
Word 2	Head Step Time: 27.5 ms 0ABE (hex)
Word 3	Step Settling Time: 15 ms 05DC (hex)
Word 4	Head Load Time: 60 ms 1770 (hex)
Word 5	Head Unload Timeout: 1 sec 03E8 (hex)
Word 6	Density: Single Sync Type: IBM
Word 7	Sectors per Track: 26 = 001A (hex) Bytes per Sector: 128

A.2.2 Recommended Drive Parameter Lists

	IBM SS SD	IBM DS SD	IBM SS DD	IBM DS DD	TI SS DD	TI DS DD	Mod IBM SS DD	Mod IBM DS DD
Word 0	0101							
Word 1	004D							
Word 2	0ABE							
Word 3	05DC							
Word 4	1770							
Word 5	03E8							
Word 6	0000							
Word 7	6880							

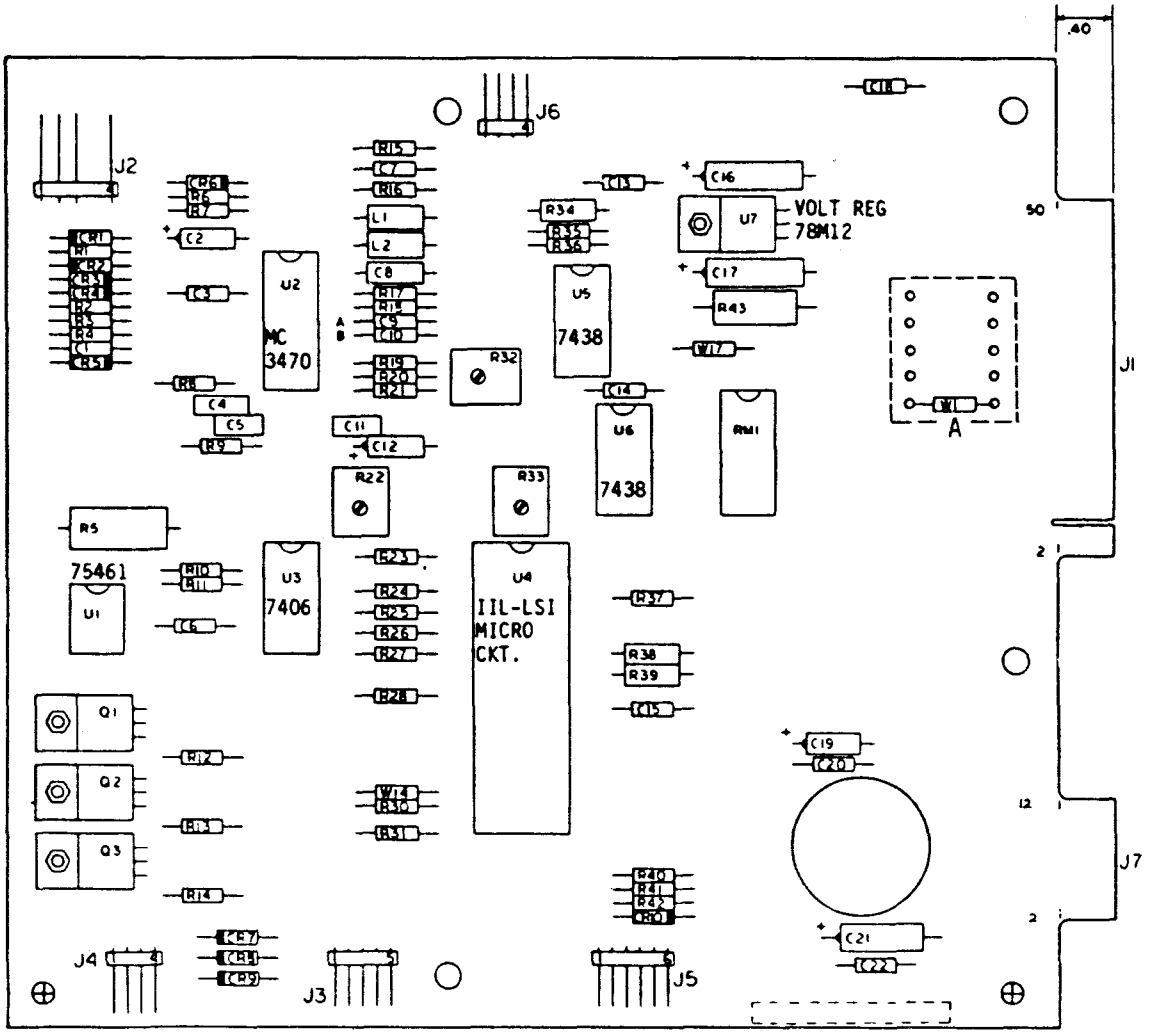


FIGURE A-1. CDC 9404B JUMPER LOCATIONS

A.2.3 Recommended CDC 9404B Jumper Settings

Jumper Name	Termination Board Jumper In or Out	Intermediate Board Jumper In or Out
W1/W5	See Note 1	See Note 1
W2/W5	See Note 1	See Note 1
W3/W5	See Note 1	See Note 1
W4/W5	See Note 1	See Note 1
Terminator Pack Beckman R220/330	In	Out

NOTE 1: Drive designators W1 to W4 correspond to drive select numbers 1 to 4. One of these is jumpered to W5 to designate the disk drive number which corresponds to the disk drive select number in the two LSBs of Command Word 2. As shipped from the factory, W1 is jumpered to W5 to designate the drive as drive 1, selected by a 00_2 in bits 14 and 15 of Command List Word 2. For example:

Disk Drive Select Number	Install Jumper
1	W1/W5
2	W2/W5
3	W3/W5
4	W4/W5

To select drive 2, 3, or 4, the user must first remove the soldered jumper installed at the factory between W1 and W5 (to select drive 1). It is recommended that the user substitute a four-switch standard DIP package (AMP 435166-2 or 435626-1, 7000 series) between W5 and W1 to W4. This allows the user to easily designate the drive number by setting one of the four switches to ON.

A.3 CDC 9406-4 EIGHT INCH DISK DRIVE

This section includes specifications, drive parameter lists, jumper locations, and jumper settings. For the most up-to-date information, consult your disk drive user's manual.

A.3.1 Specifications for Drive Parameter Lists

Word 0	Number of Sides: 2 Single Density Write Precompensation: inner ½ Double Density Write Precompensation: inner ¼ Motor On with Drive Select: No, motor always on
Word 1	Maximum Number of Tracks per side: 77 004D (hex) (154 tracks total)
Word 2	Head Step Time: 3 ms 012C (hex)
Word 3	Step Settling Time: 15 ms 05DC (hex)
Word 4	Head Load Time: 35 ms 0DAC (hex)
Word 5	Head Unload Timeout: 1 sec 03E8 (hex)
Word 6	Density: Single or Double Sync Type: TI or IBM
Word 7	Sectors per Track: 26 = 001A (hex) Bytes per Sector: 128 or 256 or 288

A.3.2 Recommended Drive Parameter Lists

	IBM SS SD	IBM DS SD	IBM SS DD	IBM DS DD	TI SS DD	TI DS DD	Mod IBM SS DD	Mod IBM DS DD
Word 0	9101	9102	A101	A102	A101	A102	A101	A102
Word 1	004D	004D	004D	004D	004D	004D	004D	004D
Word 2	012C	012C	012C	012C	012C	012C	012C	012C
Word 3	05DC	05DC	05DC	05DC	05DC	05DC	05DC	05DC
Word 4	0DAC	0DAC	0DAC	0DAC	0DAC	0DAC	0DAC	0DAC
Word 5	03E8	03E8	03E8	03E8	03E8	03E8	03E8	03E8
Word 6	0000	0000	1000	1000	1100	1100	3000	3000
Word 7	6880	6880	6900	6900	6920	6920	6900	6900

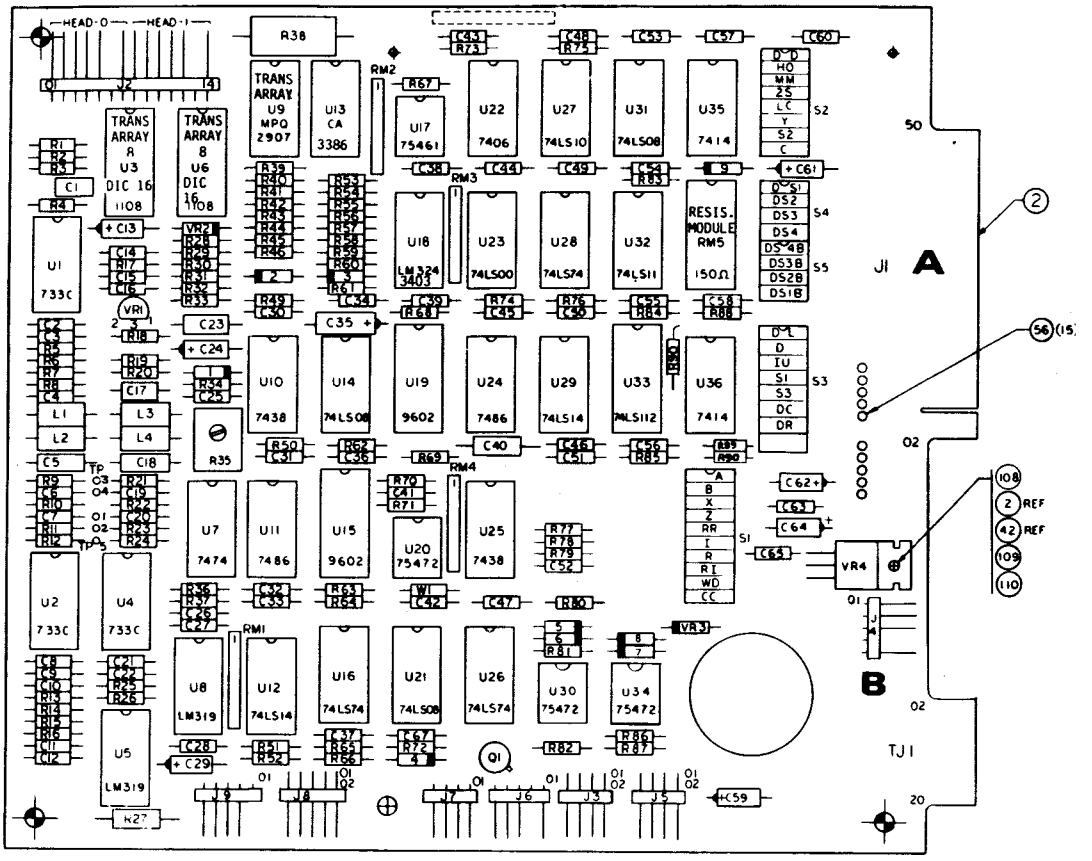


FIGURE A-2. CDC 9406-4 JUMPER LOCATIONS

A.3.3 Recommended CDC 9406-4 Jumper Settings

Jumper Name	Termination Board Jumper In or Out	Intermediate Board Jumper In or Out
DD	In	In
HO	In	In
MM	Out	Out
2S	In	In
LC	In	In
Y	Out	Out
C	In	In
DS1	See Note 1	See Note 1
DS2	See Note 1	See Note 1
DS3	See Note 1	See Note 1
DS4	See Note 1	See Note 1
DS4B	Out	Out
DS3B	Out	Out
DS2B	Out	Out
DS1B	Out	Out
DL	Out	Out
D	Out	Out
IU	Out	Out
S1	Out	Out
S2	In	In
S3	Out	Out
DC	In	In
DR	Out	Out
A	Out	Out
B	Out	Out
X	In	In
Z	In	In
RR	In	In
I	In	In
R	In	In
RI	Out	Out
WP	In	In
CC	In	In

Note 1: Only 1 of these can be jumpered at a time.

A.4 QUME DT-8 EIGHT INCH DISK DRIVE

This section includes specifications, drive parameter lists, jumper locations, and jumper settings. For the most up-to-date information, consult your disk drive user's manual.

A.4.1 Specifications for Drive Parameter Lists

Word 0	Number of Sides: 2 Single Density Write Precompensation: none Double Density Write Precompensation: inner ½ Motor On with Drive Select: No, motor always on
Word 1	Maximum Number of Tracks per side: 77 004D (hex) (154 tracks total)
Word 2	Head Step Time: 3 ms 012C (hex)
Word 3	Step Settling Time: 15 ms 05DC (hex)
Word 4	Head Load Time: 35 ms 0DAC (hex)
Word 5	Head Unload Timeout: 1 sec 03E8 (hex)
Word 6	Density: Single or Double Sync Type: TI or IBM
Word 7	Sectors per Track: 26 = 001A (hex) Bytes per Sector: 128 or 256 or 288

A.4.2 Recommended Drive Parameter Lists

	IBM SS SD	IBM DS SD	IBM SS DD	IBM DS DD	TI SS DD	TI DS DD	Mod IBM SS DD	Mod IBM DS DD
Word 0	0101	0102	9101	9102	9101	9102	9101	9102
Word 1	004D	004D	004D	004D	004D	004D	004D	004D
Word 2	012C	012C	012C	012C	012C	012C	012C	012C
Word 3	05DC	05DC	05DC	05DC	05DC	05DC	05DC	05DC
Word 4	0DAC	0DAC	0DAC	0DAC	0DAC	0DAC	0DAC	0DAC
Word 5	03E8	03E8	03E8	03E8	03E8	03E8	03E8	03E8
Word 6	0000	0000	1000	1000	1100	1100	3000	3000
Word 7	6880	6880	6900	6900	6920	6920	6900	6900

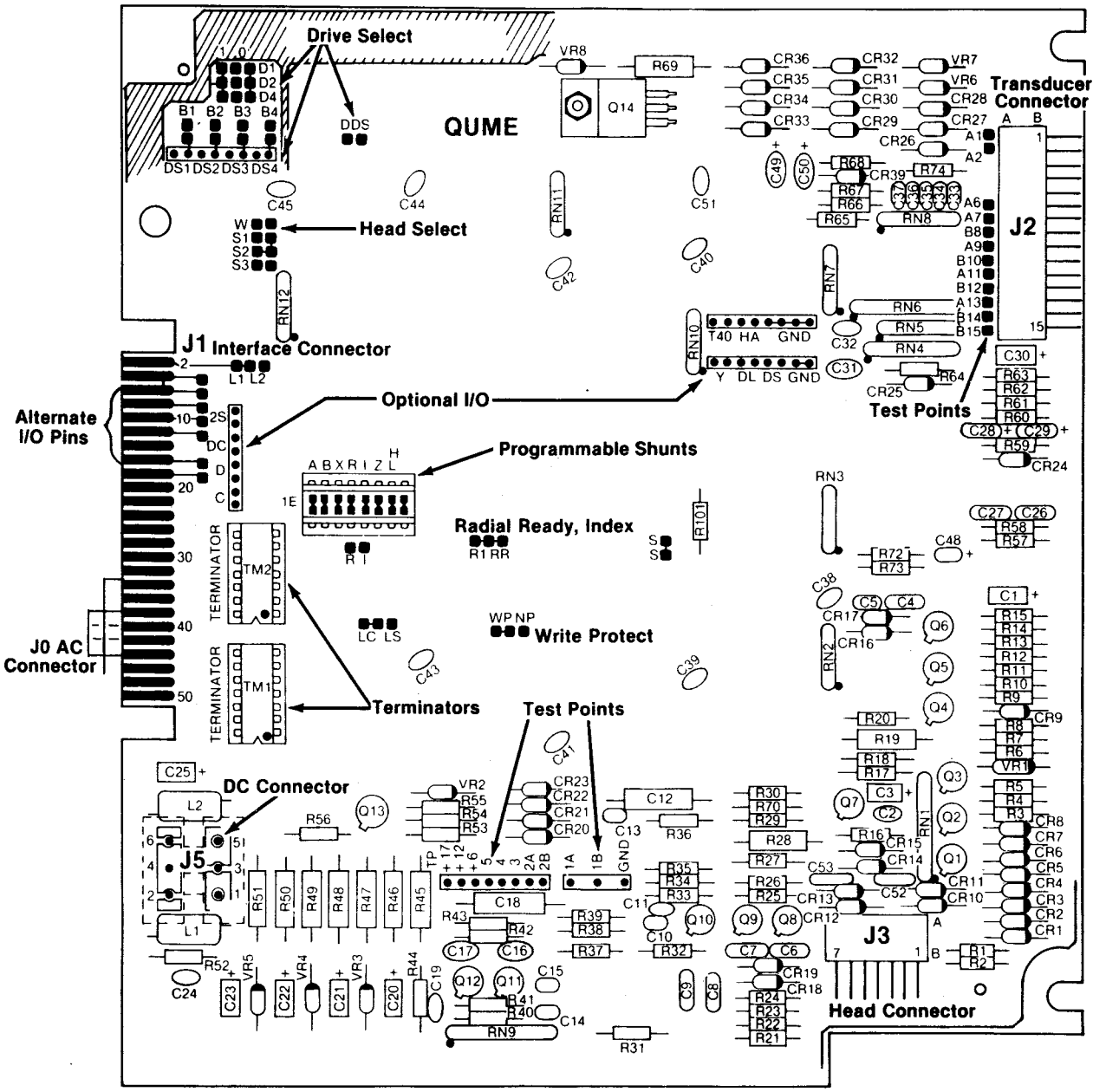


FIGURE A-3. QUME DT-8 JUMPER LOCATIONS

A.4.3 Recommended Qume DT-8 Jumper Settings

Jumper Name	Termination Board Jumper In or Out	Intermediate Board Jumper In or Out
A	In	In
B	In	In
B1	Out	Out
B2	Out	Out
B3	Out	Out
B4	Out	Out
C	In	In
D	Out	Out
DC	In	In
DDS	Out	Out
DL	Out	Out
DS	Out	Out
DS1	See Note 1	See Note 1
DS2	See Note 1	See Note 1
DS3	See Note 1	See Note 1
DS4	See Note 1	See Note 1
GND	Out	Out
GND	Out	Out
HA	Out	Out
HL	Out	Out
I	In	In
R	In	In
S1	Out	Out
S2	In	In
S3	Out	Out
X	Out	Out
Y	Out	Out
Z	In	In
2S	In	In
1TM	In	Out
2TM	In	Out

Note 1: Only one of the DS1 to DS4 (Drive Select 1 to 4) is jumpered to designate a disk drive number; this number (DS1 to DS4) corresponds to the disk drive select number specified in the LSBs of Command Word 2 (e.g., to designate a disk drive as number 2, jumper DS2).

A.5 SHUGART SA800 EIGHT INCH DISK DRIVE

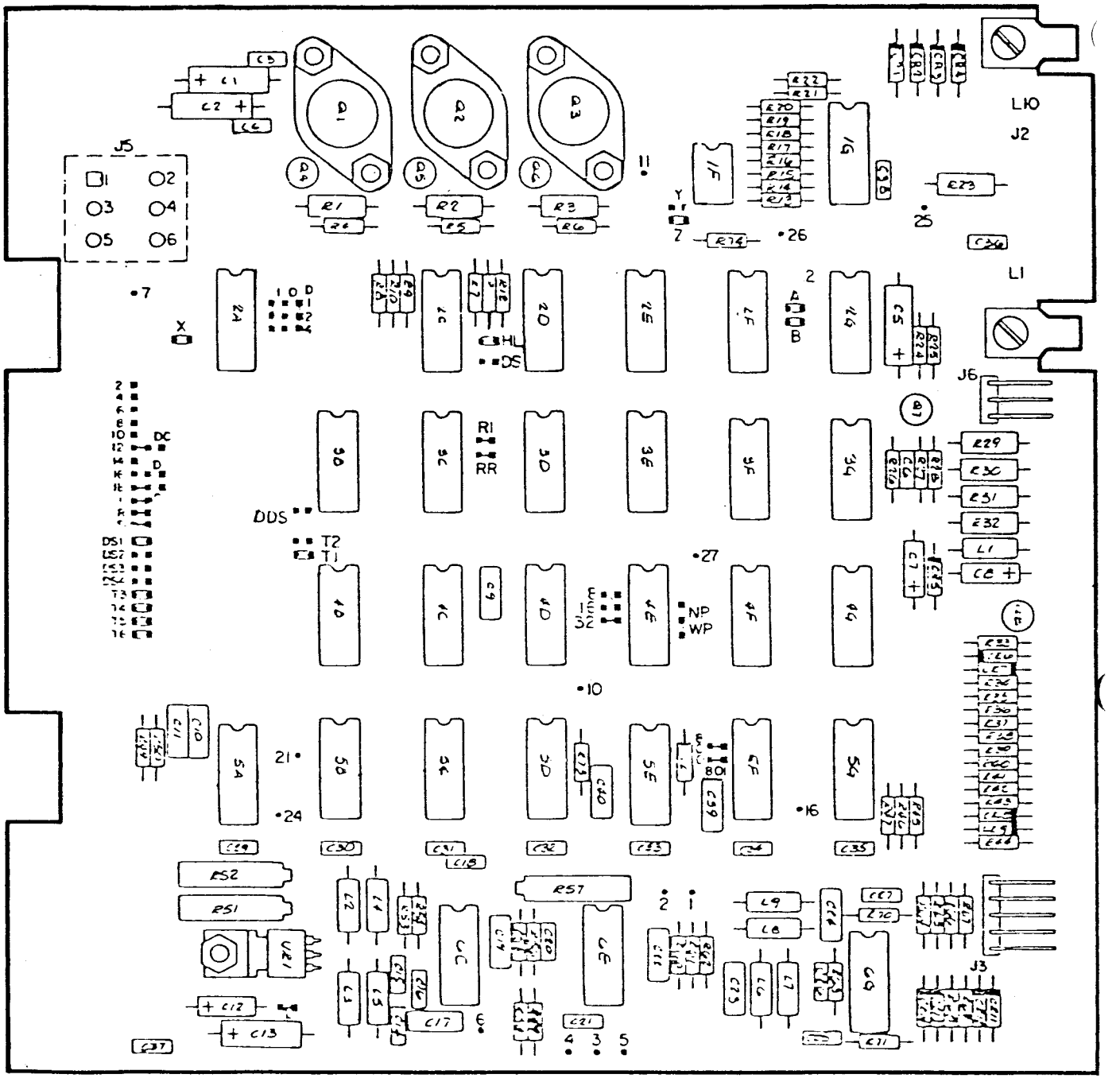
This section includes specifications, drive parameter lists, jumper locations, and jumper settings. For the most up-to-date information, consult your disk drive user's manual.

A.5.1 Specifications for Drive Parameter Lists

Word 0	Number of Sides: 1 Single Density Write Precompensation: none Double Density Write Precompensation: inner ½ Motor On with Drive Select: No, motor always on
Word 1	Maximum Number of Tracks per side: 77 004D (hex) (77 tracks total)
Word 2	Head Step Time: 8 ms 0320 (hex)
Word 3	Step Settling Time: 15 ms 05DC (hex)
Word 4	Head Load Time: 35 ms 0DAC (hex)
Word 5	Head Unload Timeout: 1 sec 03E8 (hex)
Word 6	Density: Single or Double Sync Type: TI or IBM
Word 7	Sectors per Track: 26 = 001A (hex) Bytes per Sector: 128 or 256 or 288

A.5.2 Recommended Drive Parameter Lists

	IBM SS SD	IBM DS SD	IBM SS DD	IBM DS DD	TI SS DD	TI DS DD	Mod IBM SS DD	Mod IBM DS DD
Word 0	0101		9101		9101		9101	
Word 1	004D		004D		004D		004D	
Word 2	0320		0320		0320		0320	
Word 3	05DC		05DC		05DC		05DC	
Word 4	0DAC		0DAC		0DAC		0DAC	
Word 5	03E8		03E8		03E8		03E8	
Word 6	0000		1000		1100		3000	
Word 7	6880		6900		6920		6900	



Jumper Plug Installed as Shipped

Test Point

FIGURE A-4. SHUGART SA800 JUMPER LOCATIONS

A.5.3 Recommended Shugart SA800 Jumper Settings

Jumper Name	Board Position	Termination Board Jumper In or Out	Intermediate Board Jumper In or Out
A	A	In	In
B	A	In	In
C	B	In	In
D	B	Out	Out
DC	B	In	In
DDS	C	Out	Out
DS	D	In	In
DS1	E	See Note 1	See Note 1
DS2	E	See Note 1	See Note 1
DS3	E	See Note 1	See Note 1
DS4	E	See Note 1	See Note 1
HL	D	Out	Out
T1	C	In	Out
T2	C	In	Out
T3	E	In	Out
T4	E	In	Out
T5	E	In	Out
T6	E	In	Out
X	F	Out	Out
Y	G	Out	Out
Z	G	In	In
800	H	In	In
801	H	Out	Out

Note 1: Only one of the DS1 to DS4 (Drive select 1 to 4) is jumpered to select a disk drive; this number (1 to 4) corresponds to the disk drive select number specified in the two LSBs of Command Word 2.

A.6 SHUGART SA801 EIGHT INCH DISK DRIVE

This section includes specifications, drive parameter lists, jumper locations, and jumper settings. For the most up-to-date information, consult your disk drive user's manual.

A.6.1 Specifications for Drive Parameter Lists

Word 0	Number of Sides: 1 Single Density Write Precompensation: none Double Density Write Precompensation: inner ½ Motor On with Drive Select: No, motor always on
Word 1	Maximum Number of Tracks per side: 77 004D (hex) (77 tracks total)
Word 2	Head Step Time: 8 ms 0320 (hex)
Word 3	Step Settling Time: 8 ms 0320 (hex)
Word 4	Head Load Time: 35 ms 0DAC (hex)
Word 5	Head Unload Timeout: 1 sec 03E8 (hex)
Word 6	Density: Single or Double Sync Type: TI or IBM
Word 7	Sectors per Track: 26 = 001A (hex) Bytes per Sector: 128 or 256 or 288

A.6.2 Recommended Drive Parameter Lists

	IBM SS SD	IBM DS SD	IBM SS DD	IBM DS DD	TI SS DD	TI DS DD	Mod IBM SS DD	Mod IBM DS DD
Word 0	0101		9101		9101		9101	
Word 1	004D		004D		004D		004D	
Word 2	0320		0320		0320		0320	
Word 3	0320		0320		0320		0320	
Word 4	0DAC		0DAC		0DAC		0DAC	
Word 5	03E8		03E8		03E8		03E8	
Word 6	0000		1000		1100		3000	
Word 7	6880		6900		6920		6900	

SA801 Current Requirements

DC Voltage	Tolerance	Amps Current Max	Typ	Max Ripple (p-p)
24 volts	± 1.2 Vdc	1.7	1.3	100 mV
-5 Vdc	± .25 Vdc	0.07	0.05	50 mV
+5 V	± .25 Vdc	1.0	0.8	50 mV

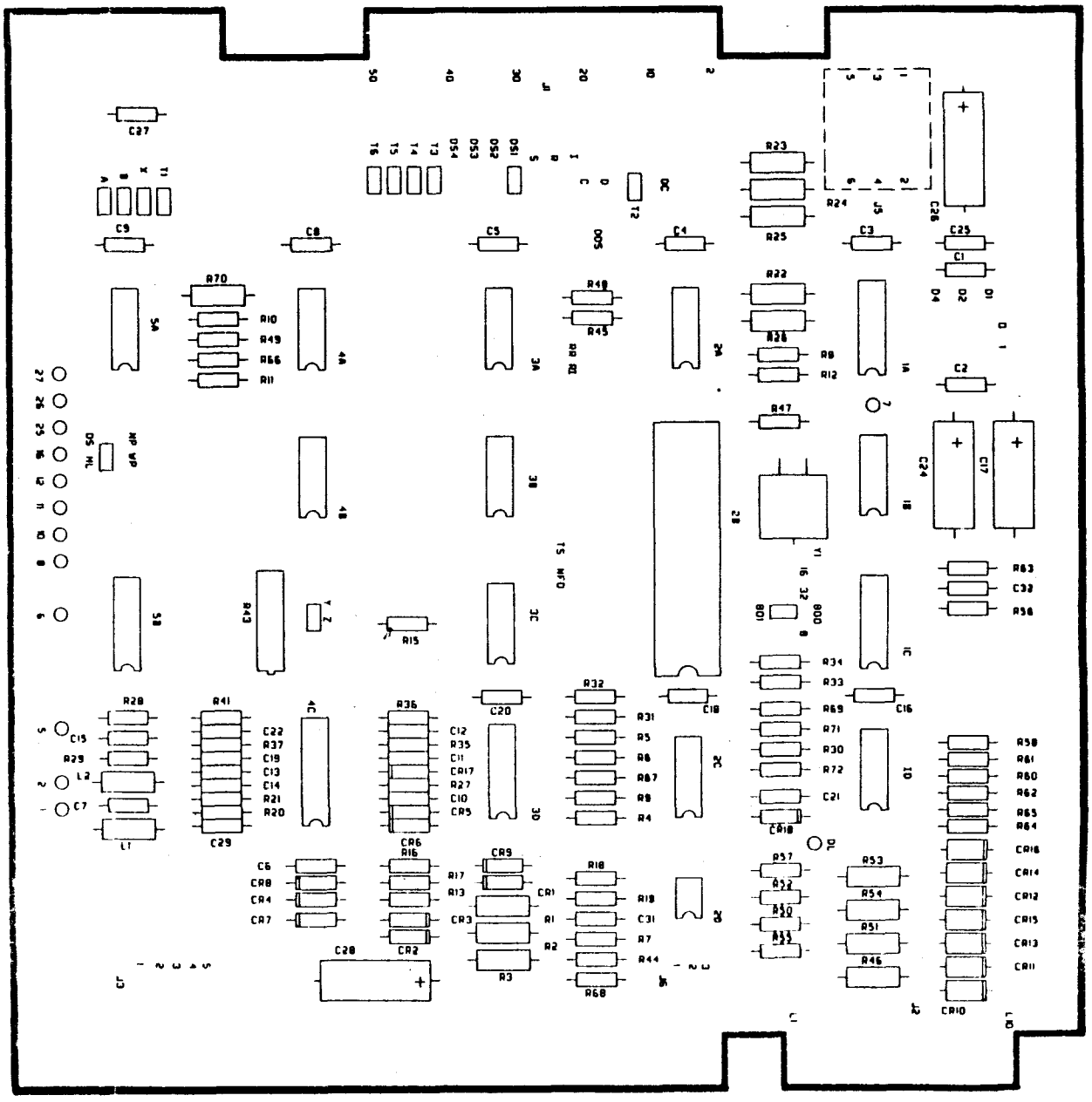


FIGURE A-5. SHUGART SA801 JUMPER LOCATIONS

A.6.3 Recommended Shugart SA801 Jumper Settings

Jumper Name	Termination Board Jumper In or Out	Intermediate Board Jumper In or Out
T1	In	Out
T2	In	Out
T3	In	Out
T4	In	Out
T5	In	Out
T6	In	Out
DS1	See Note 1	See Note 1
DS2	See Note 1	See Note 1
DS3	See Note 1	See Note 1
DS4	See Note 1	See Note 1
RR	In	In
RI	In	In
R	In	In
I	In	In
S	In	In
HL	Out	Out
DS	In	In
WP	In	In
NP	Out	Out
8,16,32	Out	Out
D	Out	Out
D1, D2	Out	Out
D4, DDS	Out	Out
A	In	In
B	In	In
X	Out	Out
C	In	In
Y	Out	Out
Z	In	In
DC	In	In
NFO	Out	Out
TS	Out	Out
800	In	In
801	Out	Out

Note 1: Only 1 of these can be jumpered at a time.

A.7 SHUGART SA851/850 EIGHT INCH DISK DRIVE

This section includes specifications, drive parameter lists, jumper locations, and jumper settings. For the most up-to-date information, consult your disk drive user's manual.

A.7.1 Specifications for Drive Parameter Lists

Word 0	Number of Sides: 2 Single Density Write Precompensation: inner ½ Double Density Write Precompensation: inner ½ Motor On with Drive Select: No, motor always on
Word 1	Maximum Number of Tracks per side: 77 004D (hex) (154 tracks total)
Word 2	Head Step Time: 3 ms 012C (hex)
Word 3	Step Settling Time: 15 ms 05DC (hex)
Word 4	Head Load Time: 50 ms 1388 (hex)
Word 5	Head Unload Timeout: 1 sec 03E8 (hex)
Word 6	Density: Single or Double Sync Type: TI or IBM
Word 7	Sectors per Track: 26 = 001A (hex) Bytes per Sector: 128 or 256 or 288

A.7.2 Recommended Drive Parameter Lists

	IBM SS SD	IBM DS SD	IBM SS DD	IBM DS DD	TI SS DD	TI DS DD	Mod IBM SS DD	Mod IBM DS DD
Word 0	9101	9102	9101	9102	9101	9102	9101	9102
Word 1	004D	004D	004D	004D	004D	004D	004D	004D
Word 2	012C	012C	012C	012C	012C	012C	012C	012C
Word 3	05DC	05DC	05DC	05DC	05DC	05DC	05DC	05DC
Word 4	1388	1388	1388	1388	1388	1388	1388	1388
Word 5	03E8	03E8	03E8	03E8	03E8	03E8	03E8	03E8
Word 6	0000	0000	1000	1000	1100	1100	3000	3000
Word 7	6880	6880	6900	6900	6920	6920	6900	6900

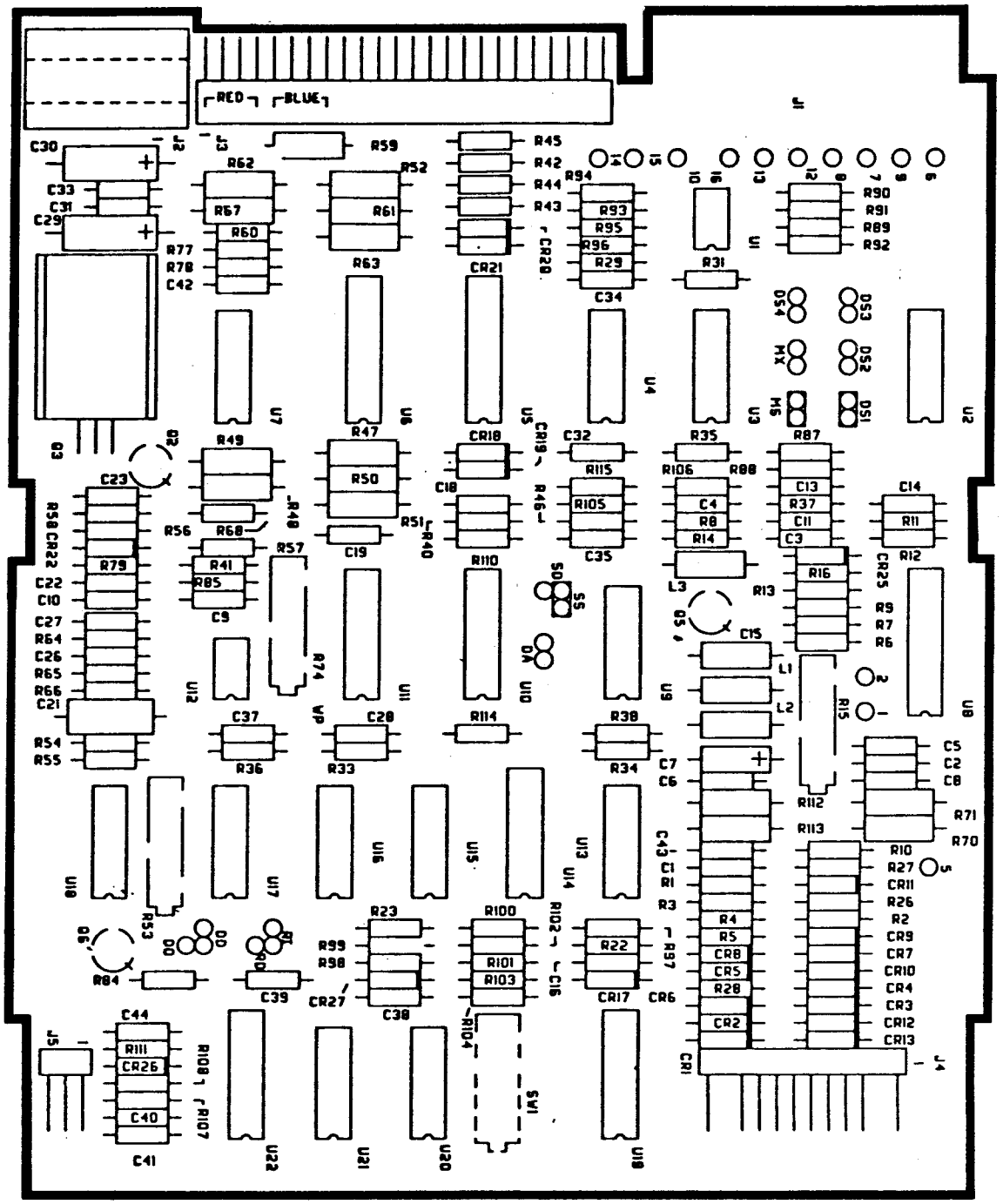


FIGURE A-6. SHUGART SA851/SA850 JUMPER LOCATIONS

A.7.3 Recommended Shugart SA851 or SA850 Jumper Settings

Jumper Name	Termination Board Jumper In or Out	Intermediate Board Jumper In or Out
I (shunt 4F)	In	In
1B	Out	Out
2B	Out	Out
2S	In	In
3B	Out	Out
4B	Out	Out
850	In	In
851	Out	Out
A (shunt 4F)	In	In
AF	In	In
B (shunt 4F)	In	In
C	Out	Out
D	Out	Out
DC	In	In
DL	In	In
DS	Out	Out
F	Out	Out
FS	In	In
HI	Out	Out
HL (shunt 4F)	Out	Out
HLL	Out	Out
IT	In	In
IW	In	In
M	In	In
NF	Out	Out
NP	Out	Out
R (shunt 4F)	In	In
RI	In	In
RM	Out	Out
RR	In	In
RS	In	In
S (shunt 4F)	In	In
S1	Out	Out
S2	In	In
S3	Out	Out
TS	Out	Out
WP	In	In
X (shunt 4F)	Out	Out
Y	Out	Out
Z (shunt 4F)	In	In
DS1 to DS4	See Note 1	See Note 1
Resistor Termination Pack	In	Out

Note 1: Only 1 of these can be jumpered at a time.

A.8 SIEMENS FDD100-8 EIGHT INCH DISK DRIVE

This section includes specifications, drive parameter lists, jumper locations, and jumper settings. For the most up-to-date information, consult your disk drive user's manual.

A.8.1 Specifications for Drive Parameter Lists

Word 0	Number of Sides: 1 Single Density Write Precompensation: inner ½ Double Density Write Precompensation: inner 1/8 Motor On with Drive Select: No, motor always on
Word 1	Maximum Number of Tracks per side: 77 004D (hex) (77 tracks total)
Word 2	Head Step Time: 8 ms 0320 (hex)
Word 3	Step Settling Time: 14 ms 0578 (hex)
Word 4	Head Load Time: 25 ms 09C4 (hex)
Word 5	Head Unload Timeout: 1 sec 03E8 (hex)
Word 6	Density: Single or Double Sync Type: TI or IBM
Word 7	Sectors per Track: 26 = 001A (hex) Bytes per Sector: 128 or 256 or 288

A.8.2 Recommended Drive Parameter Lists

	IBM SS SD	IBM DS SD	IBM SS DD	IBM DS DD	TI SS DD	TI DS DD	Mod IBM SS DD	Mod IBM DS DD
Word 0	9101		B101		B101		B101	
Word 1	004D		004D		004D		004D	
Word 2	0320		0320		0320		0320	
Word 3	0578		0578		0578		0578	
Word 4	09C4		09C4		09C4		09C4	
Word 5	03E8		03E8		03E8		03E8	
Word 6	0000		1000		1100		3000	
Word 7	6880		6900		6920		6900	

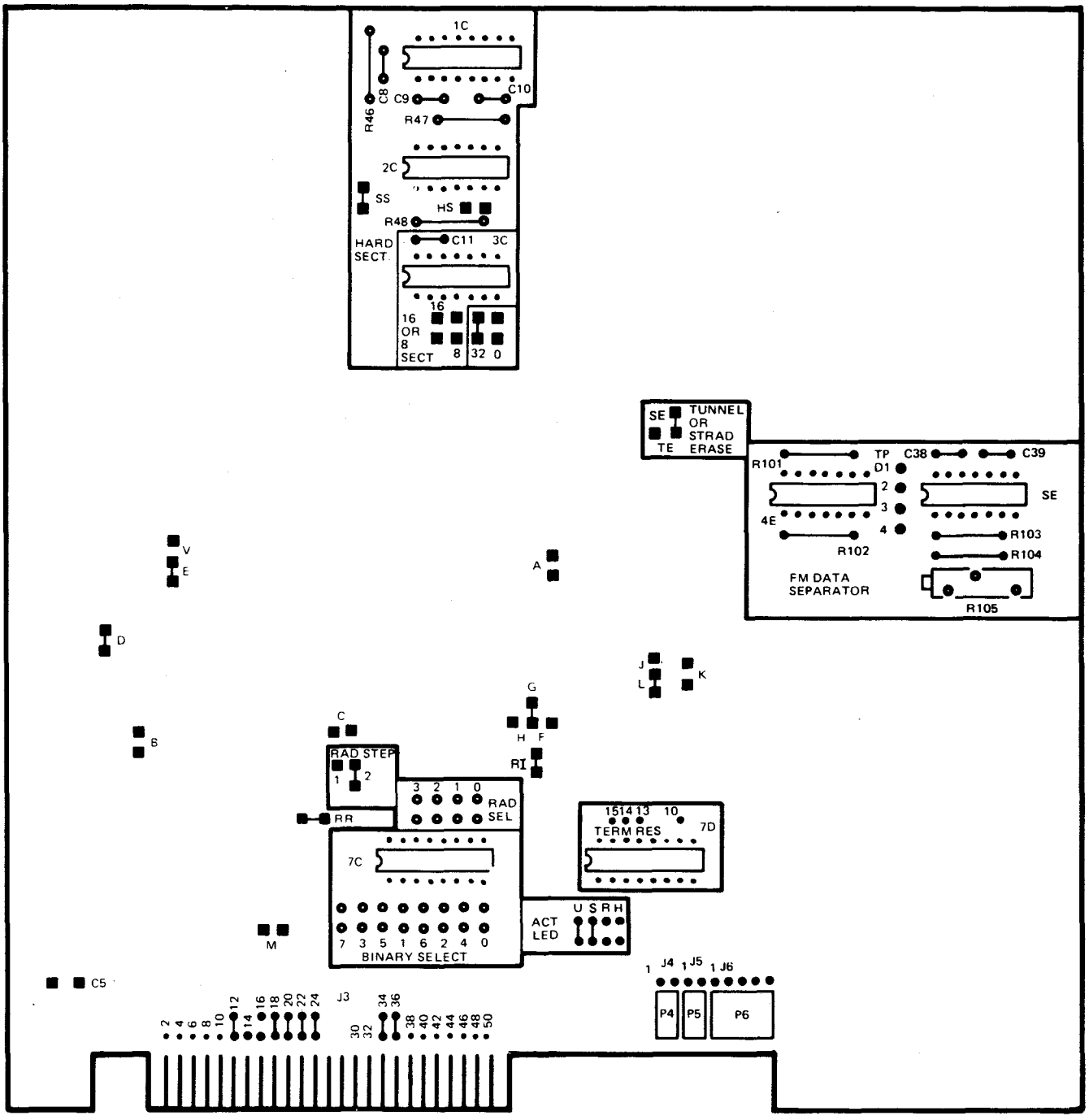


FIGURE A-7. SIEMENS FDD100-8 JUMPER LOCATIONS

A.8.3 Recommended Siemens FDD100-8 Jumper Settings

Jumper Name	Termination Board Jumper In or Out	Intermediate Board Jumper In or Out
RAO Select 0-3	See Note 1	See Note 1
Binary Select 0-7	Out	Out
Radial Step 1	Out	Out
Radial Step 2	In	In
36	In	In
A	Out	Out
34	In	In
B	Out	Out
RR	In	In
22	In	In
RI	In	In
20	In	In
C	Out	Out
24	In	In
L	In	In
J	Out	Out
18	In	In
M	Out	Out
K	Out	Out
SS	In	In
HS	Out	Out
32	In	In
16	Out	Out
8	Out	Out
0	Out	Out
U	In	In
S	Out	Out
R	Out	Out
SE	Out	Out
TE	In	In
E	In	In
V	Out	Out
12	In	In
G	Out	Out
F	In	In
H	Out	Out
Termination Resistor Pack at 7D	In	Out

Note 1. Only one of these can be jumpered at a time.

A.9 SIEMENS FDD200-8 EIGHT INCH DISK DRIVE

This section includes specifications, drive parameter lists, jumper locations, and jumper settings. For the most up-to-date information, consult your disk drive user's manual.

A.9.1 Specifications for Drive Parameter Lists

Word 0	Number of Sides: 2 Single Density Write Precompensation: inner ½ Double Density Write Precompensation: inner 1/8 Motor On with Drive Select: No, motor always on
Word 1	Maximum Number of Tracks per side: 77 004D (hex) (154 tracks total)
Word 2	Head Step Time: 6 ms 0258 (hex)
Word 3	Step Settling Time: 12 ms 0460 (hex)
Word 4	Head Load Time: 25 ms 09C4 (hex)
Word 5	Head Unload Timeout: 1 sec 03E8 (hex)
Word 6	Density: Single or Double Sync Type: TI or IBM
Word 7	Sectors per Track: 26 = 001A (hex) Bytes per Sector: 128 or 256 or 288

A.9.2 Recommended Drive Parameter Lists

	IBM SS SD	IBM DS SD	IBM SS DD	IBM DS DD	TI SS DD	TI DS DD	Mod IBM SS DD	Mod IBM DS DD
Word 0	9101	9102	B102	B102	B102	B102	B102	B102
Word 1	004D	004D	004D	004D	004D	004D	004D	004D
Word 2	0258	0258	0258	0258	0258	0258	0258	0258
Word 3	0460	0460	0460	0460	0460	0460	0460	0460
Word 4	09C4	09C4	09C4	09C4	09C4	09C4	09C4	09C4
Word 5	03E8	03E8	03E8	03E8	03E8	03E8	03E8	03E8
Word 6	0000	0000	1000	1000	1100	1100	3000	3000
Word 7	6880	6880	6900	6900	6920	6920	6900	6900

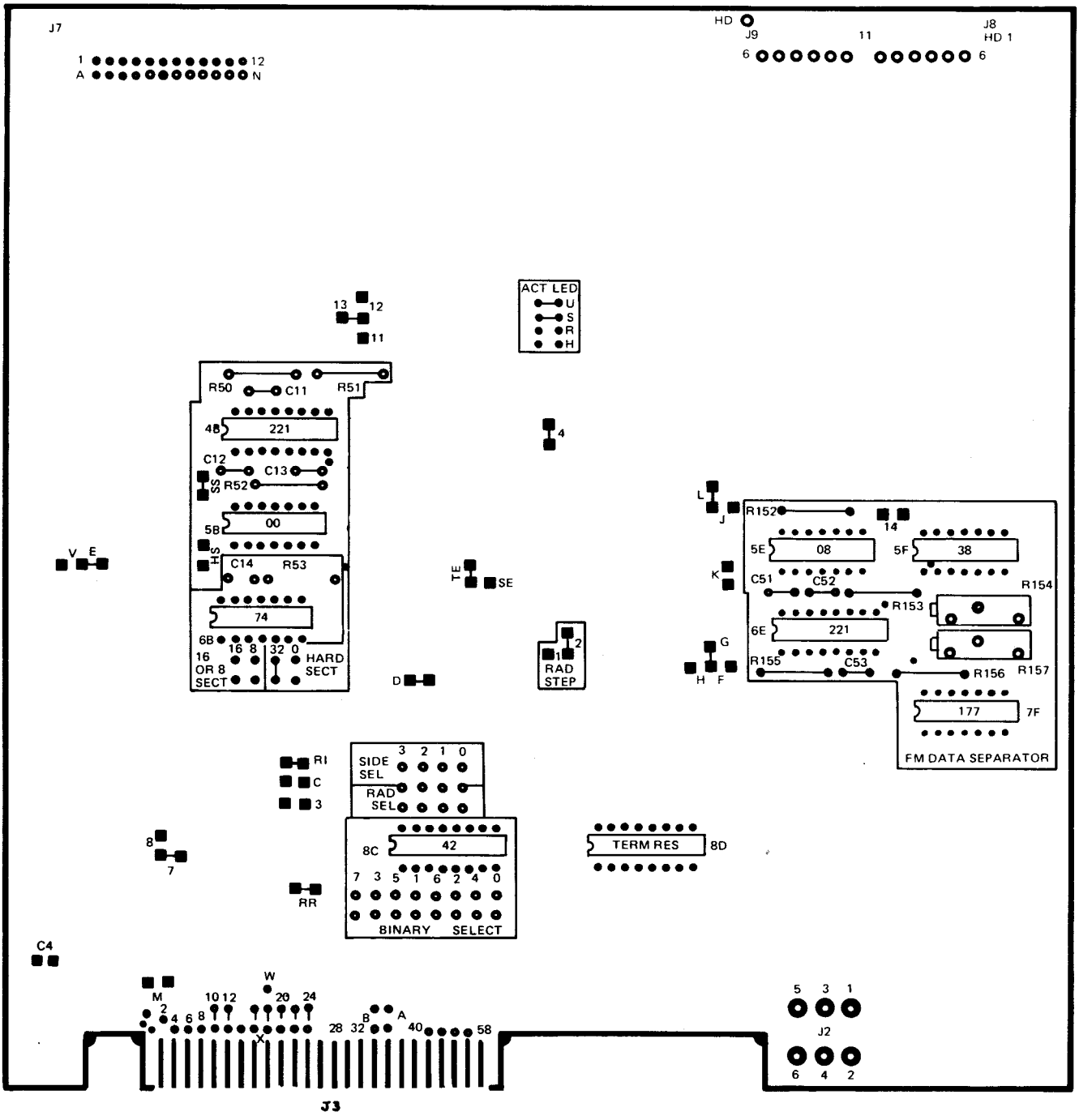


FIGURE A-8. SIEMENS FDD200-8 JUMPER LOCATIONS

A.9.3 Recommended Siemens FDD200-8 Jumper Settings

Jumper Name	Termination Board Jumper In or Out	Intermediate Board Jumper In or Out
11	Out	Out
12	Out	Out
13	In	In
14	Out	Out
16	Out	Out
18	In	In
20	In	In
22	In	In
24	In	In
3	Out	Out
4	In	In
7	In	In
8	Out	Out
C	Out	Out
E	In	In
F	In	In
G	Out	Out
H	Out	Out
HS	Out	Out
J	Out	Out
K	Out	Out
L	In	In
M	Out	Out
R	Out	Out
RI	In	In
RR	In	In
S	In	In
SE	Out	Out
SS	In	In
TE	In	In
U	In	In
V	Out	Out
X	In	In
Binary Sel 0-7	Out	Out
Hard Sec 0,8,16,32	Out	Out
Rad Sel 0-3	See Note 1	See Note 1
Rad Step 1	Out	Out
Rad Step 2	In	In
Side Sel 0-3	Out	Out
Termination Resistor Pack at 8D	In	Out

Note 1. Only one of these can be jumpered at a time.

A.10 TANDON TM848-1 EIGHT INCH DISK DRIVE

This section includes specifications, drive parameter lists, jumper locations, and jumper settings. For the most up-to-date information, consult your disk drive user's manual.

A.10.1 Specifications for Drive Parameter Lists

Options: No Head Load Solenoid, the heads are loaded all the time.

Word 0	Number of Sides: 1 Single Density Write Precompensation: none Double Density Write Precompensation: inner $\frac{1}{4}$ Motor On with Drive Select: Yes (option for motor on all the time available)
Word 1	Maximum Number of Tracks per side: 77 004D (hex) (77 tracks total)
Word 2	Head Step Time: 3 ms 012C (hex) 10 us units
Word 3	Step Settling Time: 15 ms 05DC (hex) 10 us units
Word 4	Head Load Time: 0 ms 0002 (hex)
Word 5	Head Unload Timeout: 20 sec 4E20 (hex) (drive will run for 20 seconds unless other option is installed)
Word 6	Density: Single or Double Sync Type: TI or IBM
Word 7	Sectors per Track: 26 = 001A (hex) Bytes per Sector: 128 or 256 or 288

A.10.2 Recommended Drive Parameter Lists

	IBM SS SD	IBM DS SD	IBM SS DD	IBM DS DD	TI SS DD	TI DS DD	Mod IBM SS DD	Mod IBM DS DD
Word 0	0181		A181		A181		A181	
Word 1	004D		004D		004D		004D	
Word 2	012C		012C		012C		012C	
Word 3	05DC		05DC		05DC		05DC	
Word 4	0002		0002		0002		0002	
Word 5	4E20		4E20		4E20		4E20	
Word 6	0000		1000		1100		3000	
Word 7	6880		6900		6920		6900	

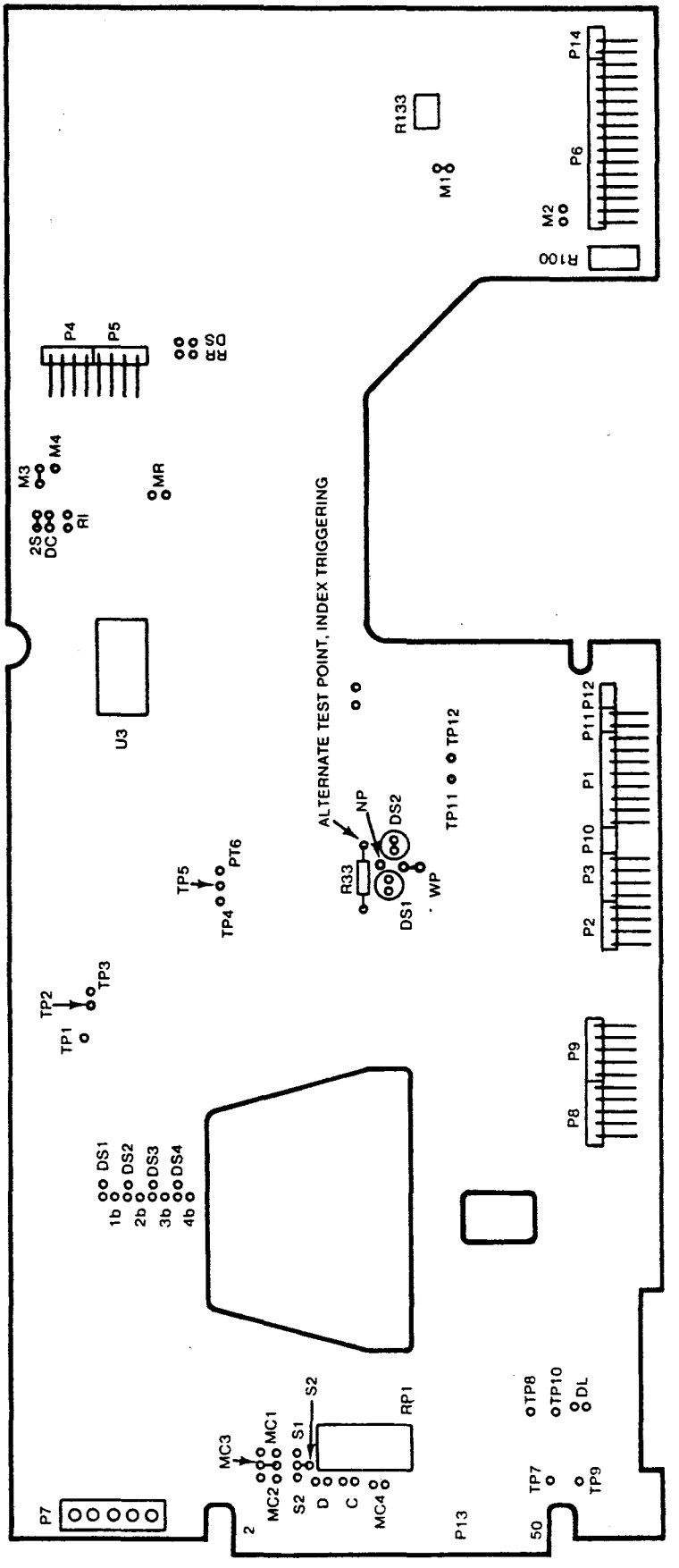


FIGURE A-9. TANDON TM848-1 JUMPER LOCATIONS

A.10.3 Recommended Tandon TM848-1 Jumper Settings

Jumper Name	Termination Board Jumper In or Out	Intermediate Board Jumper In or Out
1B	Out	Out
2B	Out	Out
2S	Out	Out
3B	Out	Out
4B	Out	Out
D	Out	Out
DC	Out	Out
DL	Out	Out
DS	Out	Out
DS1	See Note 1	See Note 1
DS2	See Note 1	See Note 1
DS3	See Note 1	See Note 1
DS4	See Note 1	See Note 1
I	In	In
M1	In	In
M2	Out	Out
M3	In	In
M4	Out	Out
MC1	Out	Out
MC2	Out	Out
MC3	Out	Out
MC4	Out	Out
NP	Out	Out
R	In	In
RI	In	In
RM	Out	Out
RR	In	In
S1	Out	Out
S2	In	In
S3	Out	Out
WP	In	In
Z	In	In
Termination Resistor Pack	In	Out

Note 1: Only one of these can be jumpered at a time.

A.11 TANDON TM848-2 EIGHT INCH DISK DRIVE

This section includes specifications, drive parameter lists, jumper locations, and jumper settings. For the most up-to-date information, consult your disk drive user's manual.

A.11.1 Specifications for Drive Parameter Lists

Options: No Head Load Solenoid, the heads are loaded all the time.

Word 0	Number of Sides: 2 Single Density Write Precompensation: none Double Density Write Precompensation: inner ½ Motor On with Drive Select: Yes (jumper selectable options)
Word 1	Maximum Number of Tracks per side: 77 004D (hex) (154 tracks total)
Word 2	Head Step Time: 3 ms 012C (hex) 10 us units
Word 3	Step Settling Time: 15 ms 05DC (hex) 10 us units
Word 4	Head Load Time: 0 ms 0002 (hex)
Word 5	Head Unload Timeout: 20 sec 4E20 (hex) (jumper selectable)
Word 6	Density: Single or Double Sync Type: TI or IBM
Word 7	Sectors per Track: 26 = 001A (hex) Bytes per Sector: 128 or 256 or 288

A.11.2 Recommended Drive Parameter Lists

	IBM SS SD	IBM DS SD	IBM SS DD	IBM DS DD	TI SS DD	TI DS DD	Mod IBM SS DD	Mod IBM DS DD
Word 0	0181	0182	9181	9182	9181	9182	9181	9182
Word 1	004D	004D	004D	004D	004D	004D	004D	004D
Word 2	012C	012C	012C	012C	012C	012C	012C	012C
Word 3	05DC	05DC	05DC	05DC	05DC	05DC	05DC	05DC
Word 4	0002	0002	0002	0002	0002	0002	0002	0002
Word 5	4E20	4E20	4E20	4E20	4E20	4E20	4E20	4E20
Word 6	0000	0000	1000	1000	1100	1100	3000	3000
Word 7	6880	6880	6900	6900	6920	6920	6900	6900

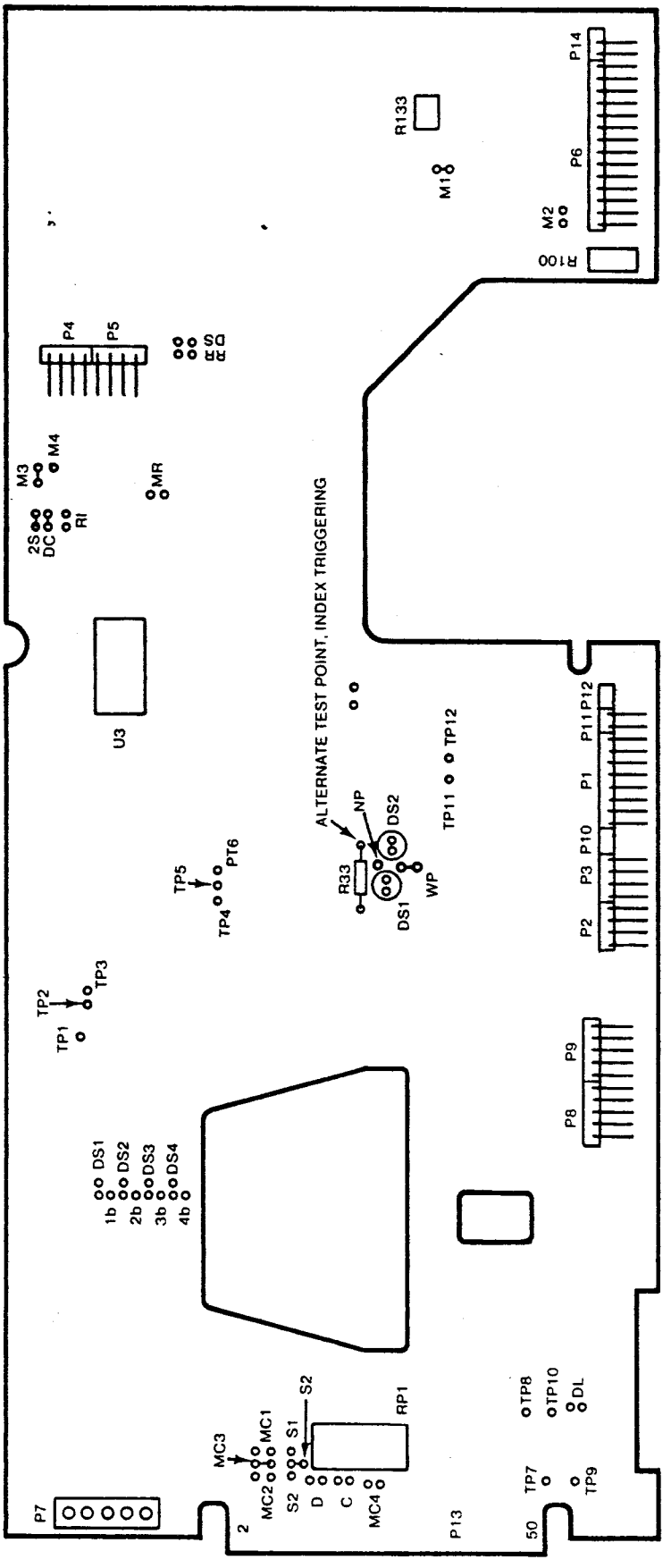


FIGURE A-10. TANDON TM848-2 JUMPER LOCATIONS

A.11.3 Recommended Tandon TM848-2 Jumper Settings

Jumper Name	Termination Board Jumper In or Out	Intermediate Board Jumper In or Out
1B	Out	Out
2B	Out	Out
2S	Out *IN	Out
3B	Out	Out
4B	Out	Out
D	Out	Out
DC	Out	Out
DL	Out	Out
DS	Out	Out
DS1	See Note 1	See Note 1
DS2	See Note 1	See Note 1
DS3	See Note 1	See Note 1
DS4	See Note 1	See Note 1
I	In	In
M1	In	In
M2	Out	Out
M3	In	In
M4	Out	Out
MC1	Out	Out
MC2	Out	Out
MC3	Out	Out
MC4	Out	Out
NP	Out	Out
R	In	In
RI	In	In
RM	Out	Out
RR	In	In
S1	Out	Out
S2	In	In
S3	Out	Out
WP	In	In
Z	In	In
Termination Resistor Pack	In	Out

Note 1: Only 1 of these can be jumpered at a time.

A.12 BASF 6106 FIVE INCH DISK DRIVE

MANUFACTURER BASF

MODEL 6106

SIZE $5\frac{1}{4}$ inch

A.12.1 Specifications for Drive Parameter Lists

Word 0	Number of Sides: 1 Single Density Write Precompensation: none Double Density Write Precompensation: inner $\frac{1}{2}$ Motor On with Drive Select: Yes
Word 1	Maximum Number of Tracks per side: 40 0028 (hex) (40 tracks total)
Word 2	Head Step Time: 12 ms 0460 (hex)
Word 3	Step Settling Time: 48 ms to 50 ms 12C0 (hex)
Word 4	Head Load Time: 35.0 ms 0DAC (hex)
Word 5	Head Unload Timeout: 1 sec 03E8 (hex)
Word 6	Density: Single or Double Sync Type: IBM
Word 7	Sectors per Track: 16 = 0010 (hex) Bytes per Sector: 128 or 256

A.12.2 Recommended Drive Parameter Lists

	IBM SS SD	IBM DS SD	IBM SS DD	IBM DS DD	TI SS DD	TI DS DD	Mod IBM SS DD	Mod IBM DS DD
Word 0	0001		9001					
Word 1	0028		0028					
Word 2	0460		0460					
Word 3	12C0		12C0					
Word 4	0DAC		0DAC					
Word 5	03E8		03E8					
Word 6	0000		1000					
Word 7	4080		4100					

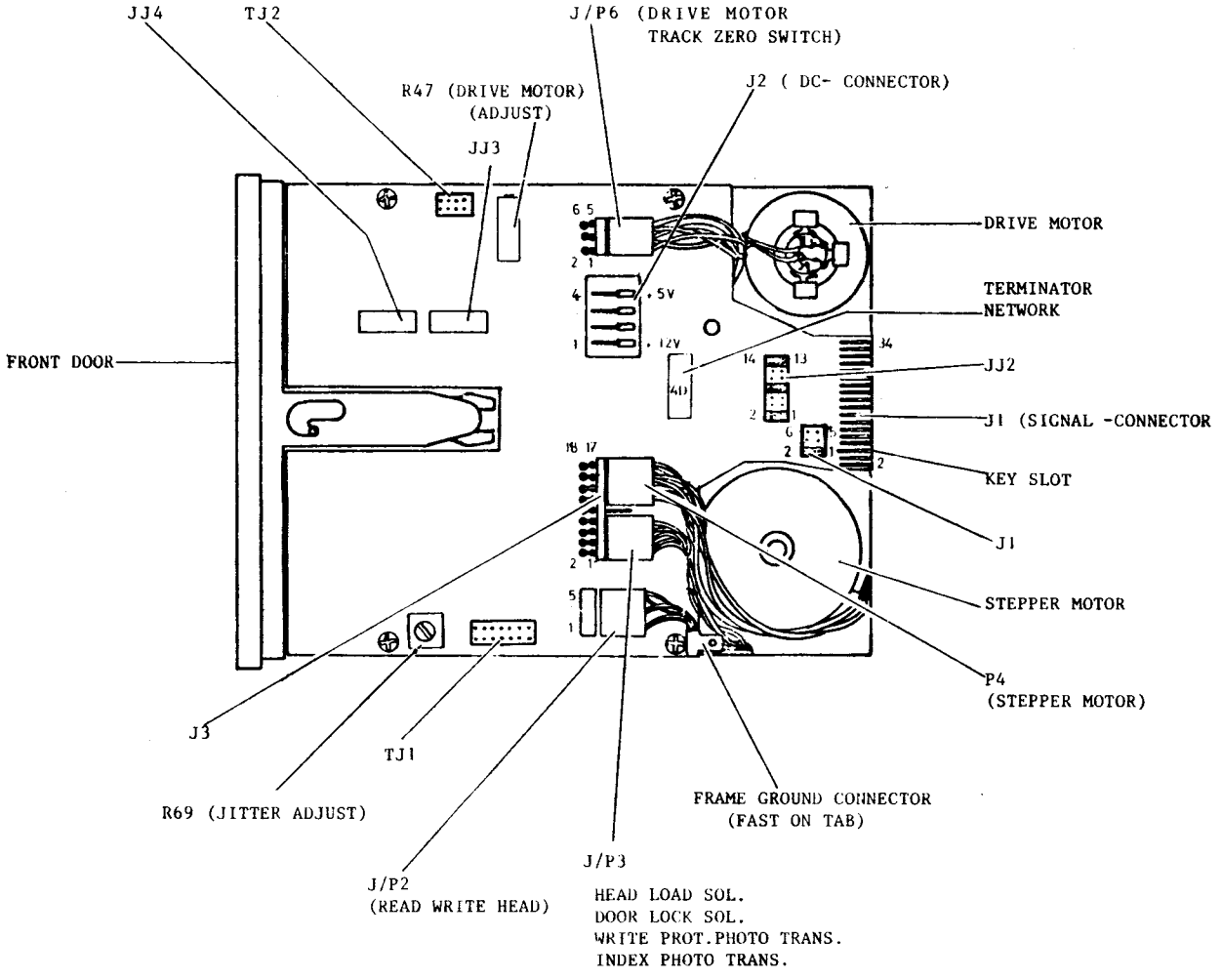


FIGURE A-11. BASF 6106 JUMPER LOCATIONS

A.12.3 Recommended BASF 6106 Jumper Settings

Jumper Name	Termination Board Jumper In or Out	Intermediate Board Jumper In or Out
JJ1/DS1	See Note 1	See Note 1
JJ1/DS2	See Note 1	See Note 1
JJ1/DS3	See Note 1	See Note 1
JJ2/1-2	In	In
JJ2/3-4	Out (auto head load)	Out
JJ2/5-6	In (auto select)	In
JJ2/7-8	Out (auto select)	Out
JJ2/9-10	Out (auto head load)	Out
JJ2/11-12	In (disk change option)	In
JJ2/13-14	Out (disk change option)	Out (disk change not needed)
JJ3/1-2	Out (door locks upon HDload)	Out
JJ3/3-4	Out (write protect)	Out
JJ3/5-6	In (write protect)	In
JJ3/7-8	Out (door lock)	Out (door lock)
JJ3/9-10	Out (door lock)	Out
JJ3/11-12	In (door locks upon HDload)	In
JJ3/13-14	In (auto head select)	In
JJ4/1-2	Out (stepper enable = motor on)	Out
JJ4/3-4	In (write protect)	In
JJ4/5-6	Out (activate LED)	Out
JJ4/7-8	In (activate LED)	In
JJ4/9-10	Out (door lock)	Out
JJ4/11-12	Out (door lock)	Out
JJ4/13-14	Out	Out
Termination Resistor Pack at 4D	In	Out

Note 1: Only 1 of these can be jumpered at a time.

A.13 CDC 9408 FIVE INCH DISK DRIVE

This section includes specifications, drive parameter lists, jumper locations, and jumper settings. For the most up-to-date information, consult your disk drive user's manual.

A.13.1 Specifications for Drive Parameter Lists

Word 0	Number of Sides: 1 Single Density Write Precompensation: inner ½ Double Density Write Precompensation: all over disk Motor On with Drive Select: Yes
Word 1	Maximum Number of Tracks per side: 35 0023 (hex) (35 tracks total)
Word 2	Head Step Time: 20 ms 07D0 (hex)
Word 3	Step Settling Time: 15 ms 05DC (hex)
Word 4	Head Load Time: 50 ms 1388 (hex)
Word 5	Head Unload Timeout: 1 sec 03E8 (hex)
Word 6	Density: Single or Double Sync Type: IBM
Word 7	Sectors per Track: 16 = 0010 (hex) Bytes per Sector: 128 or 256

A.13.2 Recommended Drive Parameter Lists

	IBM SS SD	IBM DS SD	IBM SS DD	IBM DS DD	TI SS DD	TI DS DD	Mod IBM SS DD	Mod IBM DS DD
Word 0	9001		8001					
Word 1	0023		0023					
Word 2	07D0		07D0					
Word 3	05DC		05DC					
Word 4	1388		1388					
Word 5	03E8		03E8					
Word 6	0000		1000					
Word 7	4080		4100					

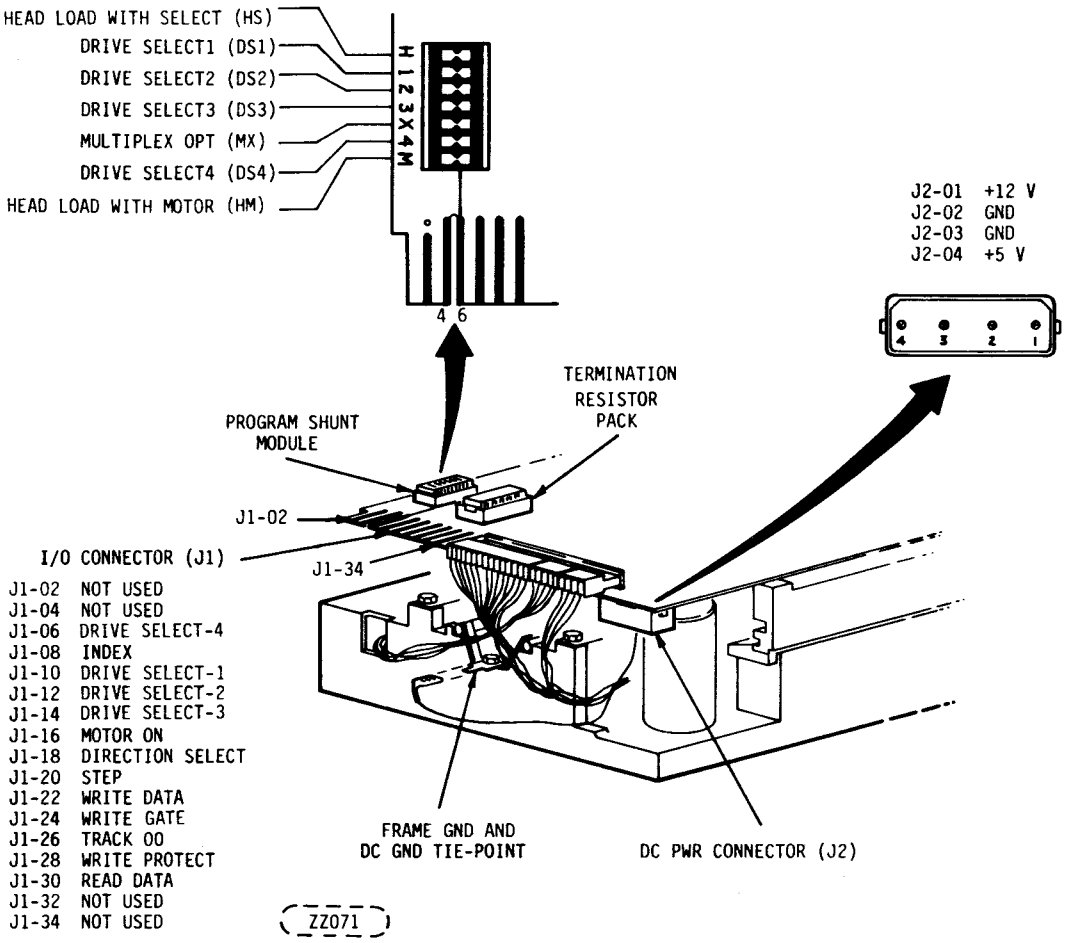


FIGURE A-12. CDC 9408 JUMPER LOCATIONS

A.13.3 Recommended CDC 9408 Jumper Settings

Jumper Name	Termination Board Jumper In or Out	Intermediate Board Jumper In or Out
DS1	See Note 1	See Note 1
DS2	See Note 1	See Note 1
DS3	See Note 1	See Note 1
DS4	See Note 1	See Note 1
HM	Out	Out (load head w/motor on - signal)
HS	In	In (load head w/select - signal)
MX	Out	Out
Termination Resistor Pack	In	Out

Note 1: Only 1 of these can be jumpered at a time.

A.14 CDC 9409T FIVE INCH DISK DRIVE

This section includes specifications, drive parameter lists, jumper locations, and jumper settings. For the most up-to-date information, consult your disk drive user's manual.

A.14.1 Specifications for Drive Parameter Lists

Word 0	Number of Sides: 2 Single Density Write Precompensation: all over disk Double Density Write Precompensation: inner ½ Motor On with Drive Select: Yes	
Word 1	Maximum Number of Tracks per side: 80	0050 (hex) (96 TPI, 180 tracks total)
Word 2	Head Step Time: 5 ms	01F4 (hex)
Word 3	Step Settling Time: 15 ms	05DC (hex)
Word 4	Head Load Time: 50 ms	1388 (hex)
Word 5	Head Unload Timeout: 1 sec	03E8 (hex)
Word 6	Density: Single or Double Sync Type: IBM	
Word 7	Sectors per Track: 16 = 0010 (hex) Bytes per Sector: 128 or 256	

A.14.2 Recommended Drive Parameter Lists

	IBM SS SD	IBM DS SD	IBM SS DD	IBM DS DD	TI SS DD	TI DS DD	Mod IBM SS DD	Mod IBM DS DD
Word 0	8001	8002	9001	9002				
Word 1	0050	0050	0050	0050				
Word 2	01F4	01F4	01F4	01F4				
Word 3	05DC	05DC	05DC	05DC				
Word 4	1388	1388	1388	1388				
Word 5	03E8	03E8	03E8	03E8				
Word 6	0000	0000	1000	1000				
Word 7	4080	4080	4100	4100				

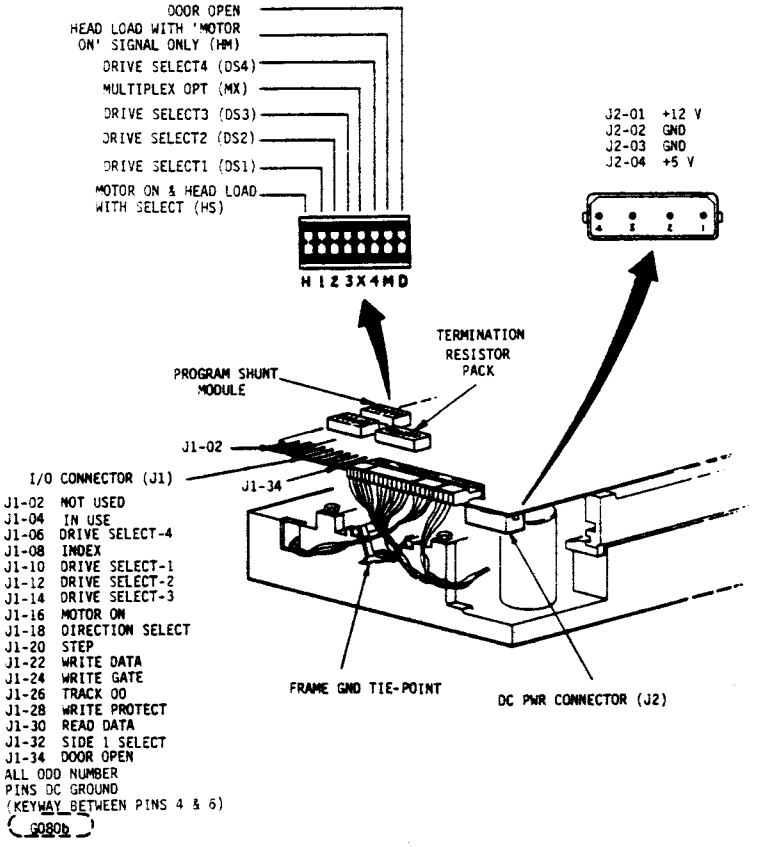


FIGURE A-13. CDC 9409T JUMPER LOCATIONS

A.14.3 Recommended CDC 9409T Jumper Settings

Jumper Name	Termination Board Jumper In or Out	Intermediate Board Jumper In or Out
DO	Out	Out (Door Open)
DS1	See Note 1	See Note 1
DS2	See Note 1	See Note 1
DS3	See Note 1	See Note 1
DS4	See Note 1	See Note 1
HM	Out	Out
HS	In	In (Motor on with any drive sel. sig)
MX	Out	Out
Termination Resistor Pack	In	Out

Note 1: Only 1 of these can be jumpered at a time.

A.15 QUME DT-5 FIVE INCH DISK DRIVE

This section includes specifications, drive parameter lists, jumper locations, and jumper settings. For the most up-to-date information, consult your disk drive user's manual.

A.15.1 Specifications for Drive Parameter Lists

Word 0	Number of Sides: 2 Single Density Write Precompensation: none Double Density Write Precompensation: all over disk Motor On with Drive Select: Yes
Word 1	Maximum Number of Tracks per side: 40 0028 (hex) (80 tracks total)
Word 2	Head Step Time: 20 ms 07D0 (hex)
Word 3	Step Settling Time: 15 ms 05DC (hex)
Word 4	Head Load Time: 50 ms 1388 (hex)
Word 5	Head Unload Timeout: 1 sec 03E8 (hex)
Word 6	Density: Single or Double Sync Type: IBM
Word 7	Sectors per Track: 16 = 0010 (hex) Bytes per Sector: 128 or 256

A.15.2 Recommended Drive Parameter Lists

	IBM SS SD	IBM DS SD	IBM SS DD	IBM DS DD	TI SS DD	TI DS DD	Mod IBM SS DD	Mod IBM DS DD
Word 0	0001	0002	8001	8002				
Word 1	0028	0028	0028	0028				
Word 2	07D0	07D0	07D0	07D0				
Word 3	05DC	05DC	05DC	05DC				
Word 4	1388	1388	1388	1388				
Word 5	03E8	03E8	03E8	03E8				
Word 6	0000	0000	1000	1000				
Word 7	4080	4080	4100	4100				

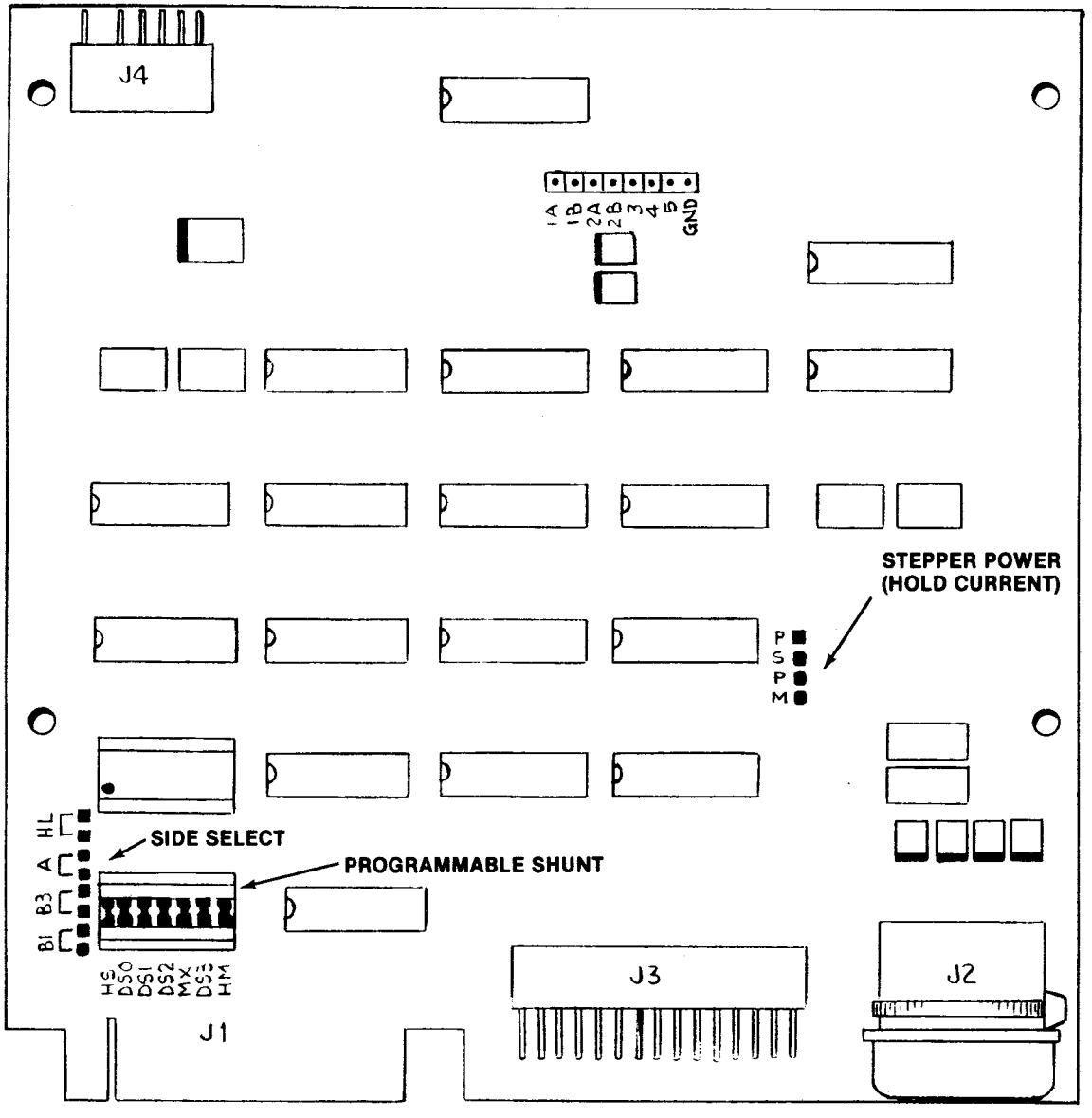


FIGURE A-14. QUME DT-5 JUMPER LOCATIONS

A.15.3 Recommended Qume DT-5 Jumper Settings

Jumper Name	Termination Board Jumper In or Out	Intermediate Board Jumper In or Out
DS1	See Note 1	See Note 1
DS2	See Note 1	See Note 1
DS3	See Note 1	See Note 1
DS4	See Note 1	See Note 1
HM	Out	Out
HS	In	In
MX	Out	Out
Termination Resistor Pack	In	Out

Note 1: Only 1 of these can be jumpered at a time.

A.16 SHUGART SA400 FIVE INCH DISK DRIVE

This section includes specifications, drive parameter lists, jumper locations, and jumper settings. For the most up-to-date information, consult your disk drive user's manual.

A.16.1 Specifications for Drive Parameter Lists

Word 0	Number of Sides: 1 Single Density Write Precompensation: none Double Density Write Precompensation: inner ½ Motor On with Drive Select: Yes
Word 1	Maximum Number of Tracks per side: 35 0023 (hex) (35 tracks total)
Word 2	Head Step Time: 40 ms 0FA0 (hex)
Word 3	Step Settling Time: 10 ms 03E8 (hex)
Word 4	Head Load Time: 75 ms 1D4C (hex)
Word 5	Head Unload Timeout: 1 sec 03E8 (hex)
Word 6	Density: Single or Double Sync Type: IBM
Word 7	Sectors per Track: 16 = 0010 (hex) Bytes per Sector: 128 or 256

A.16.2 Recommended Drive Parameter Lists

	IBM SS SD	IBM DS SD	IBM SS DD	IBM DS DD	TI SS DD	TI DS DD	Mod IBM SS DD	Mod IBM DS DD
Word 0	0001		9001					
Word 1	0023		0023					
Word 2	0FA0		0FA0					
Word 3	03E8		03E8					
Word 4	1D4C		1D4C					
Word 5	03E8		03E8					
Word 6	0000		1000					
Word 7	4080		4100					

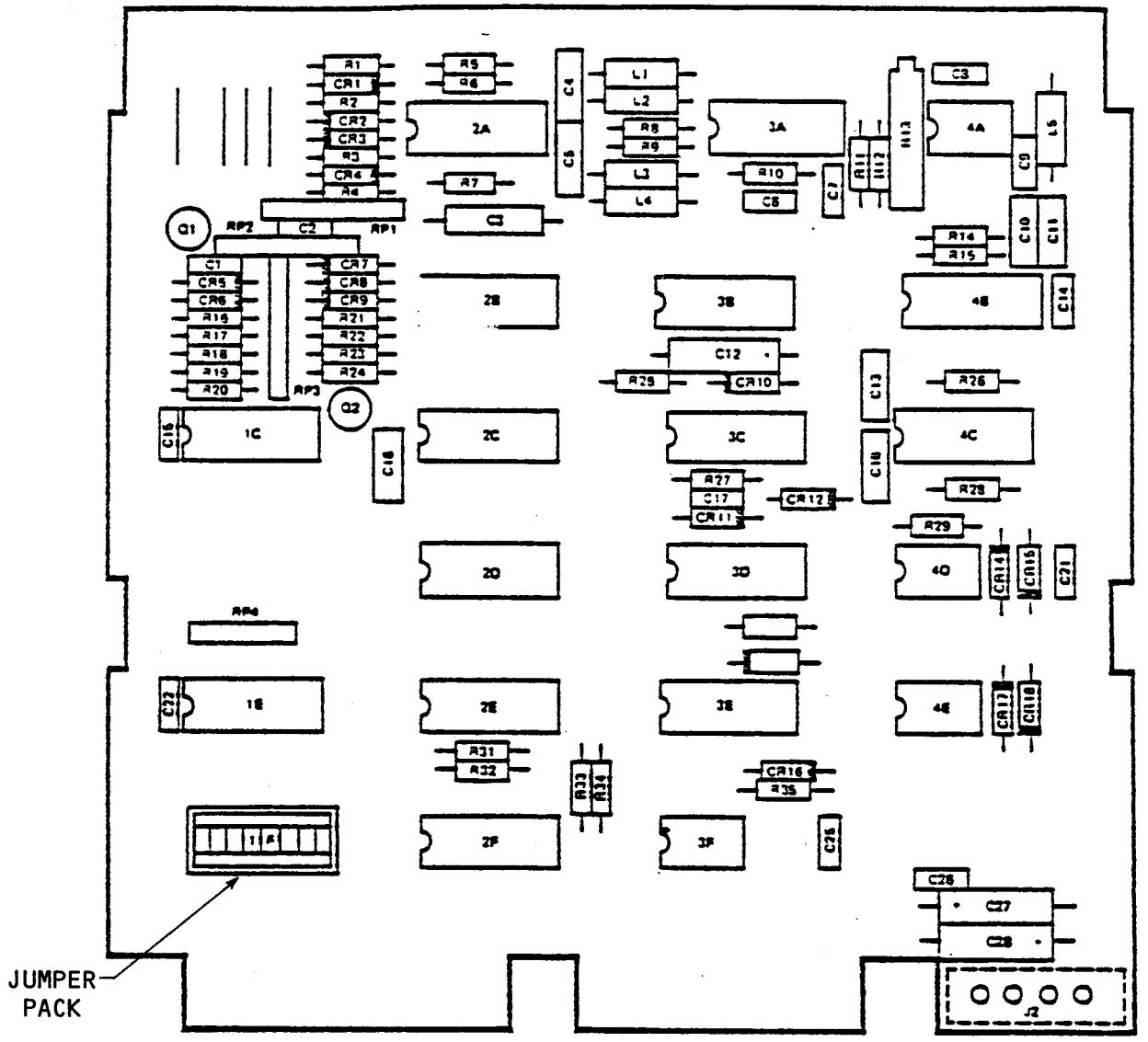


FIGURE A-15. SHUGART SA400 JUMPER LOCATIONS

A.16.3 Recommended Shugart SA400 Jumper Settings

Jumper Name	Termination Board Jumper In or Out	Intermediate Board Jumper In or Out
DL	Out	Out
DS1	See Note 1	See Note 1
DS2	See Note 1	See Note 1
DS3	See Note 1	See Note 1
HL	In	In
HM	Out	Out
MX	Out	Out
Termination Resistor Pack	In	Out

Note 1: Only 1 of these can be jumpered at a time.

A.17 SHUGART SA450 FIVE INCH DISK DRIVE

This section includes specifications, drive parameter lists, jumper locations, and jumper settings. For the most up-to-date information, consult your disk drive user's manual.

A.17.1 Specifications for Drive Parameter Lists

Word 0	Number of Sides: 2 Single Density Write Precompensation: none Double Density Write Precompensation: inner ½ Motor On with Drive Select: Yes
Word 1	Maximum Number of Tracks per side: 40 0028 (hex) (80 tracks total)
Word 2	Head Step Time: 25 ms 09C4 (hex)
Word 3	Step Settling Time: 15 ms 05DC (hex)
Word 4	Head Load Time: 50 ms 1388 (hex)
Word 5	Head Unload Timeout: 1 sec 03E8 (hex)
Word 6	Density: Single or Double Sync Type: IBM
Word 7	Sectors per Track: 16 = 0010 (hex) Bytes per Sector: 128 or 256

A.17.2 Recommended Drive Parameter Lists

	IBM SS SD	IBM DS SD	IBM SS DD	IBM DS DD	TI SS DD	TI DS DD	Mod IBM SS DD	Mod IB DS DDM
Word 0	0001	0002	9001	9002				
Word 1	0028	0028	0028	0028				
Word 2	09C4	09C4	09C4	09C4				
Word 3	05DC	05DC	05DC	05DC				
Word 4	1388	1388	1388	1388				
Word 5	03E8	03E8	03E8	03E8				
Word 6	0000	0000	1000	1000				
Word 7	4080	4080	4100	4100				

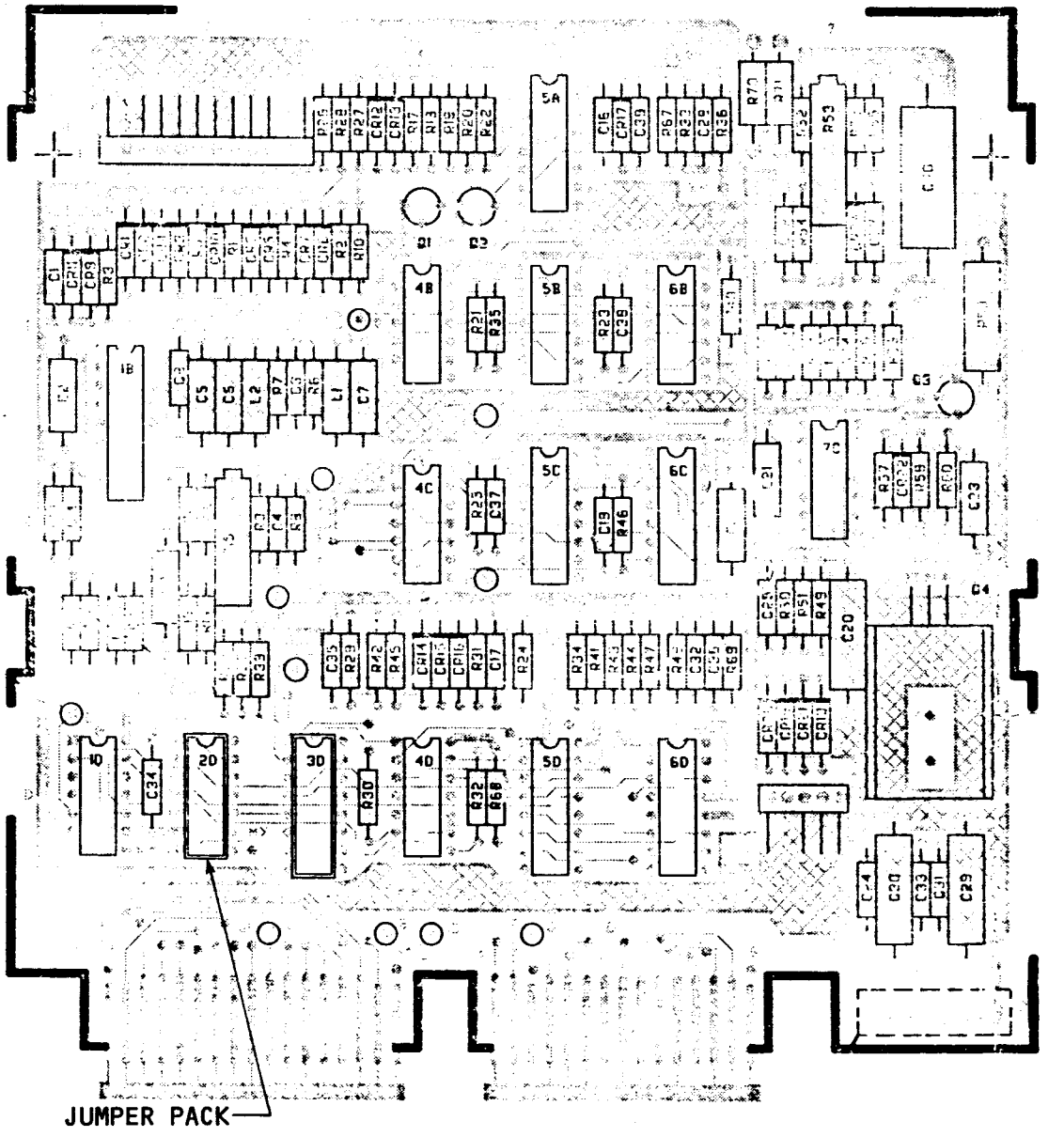


FIGURE A-16. SHUGART SA450 JUMPER LOCATIONS

A.17.3 Recommended Shugart SA450 Jumper Settings

Jumper Name	Termination Board Jumper In or Out	Intermediate Board Jumper In or Out
DS1	See Note 1	See Note 1
DS2	See Note 1	See Note 1
DS3	See Note 1	See Note 1
DS4	See Note 1	See Note 1
MM	Out	Out
MS	In	In
MX	Out	Out
Termination Resistor Pack @ Location 3D	In	Out

Note 1: Only 1 of these can be jumpered at a time.

A.18 SHUGART SA410 FIVE INCH DISK DRIVE

This section includes specifications, drive parameter lists, jumper locations, and jumper settings. For the most up-to-date information, consult your disk drive user's manual.

A.18.1 Specifications for Drive Parameter Lists

Options: No Head Load Solenoid, the heads are loaded all the time.

Word 0	Number of Sides: 1 Single Density Write Precompensation: all over disk Double Density Write Precompensation: inner ½ Motor On with Drive Select: Yes
Word 1	Maximum Number of Tracks per side: 80 0050 (hex) (96 TPI, 80 tracks total)
Word 2	Head Step Time: 6 ms 0258 (hex)
Word 3	Step Settling Time: 15 ms 05DC (hex)
Word 4	Head Load Time: 0 ms 0002 (hex)
Word 5	Head Unload Timeout: 1 sec 03E8 (hex)
Word 6	Density: Single or Double Sync Type: IBM
Word 7	Sectors per Track: 16 = 0010 (hex) Bytes per Sector: 128 or 256

A.18.2 Recommended Drive Parameter Lists

	IBM SS SD	IBM DS SD	IBM SS DD	IBM DS DD	TI SS DD	TI DS DD	Mod IBM SS DD	Mod IB DS DDM
Word 0	8001		9001					
Word 1	0050		0050					
Word 2	0258		0258					
Word 3	05DC		05DC					
Word 4	0002		0002					
Word 5	03E8		03E8					
Word 6	0000		1000					
Word 7	4080		4100					

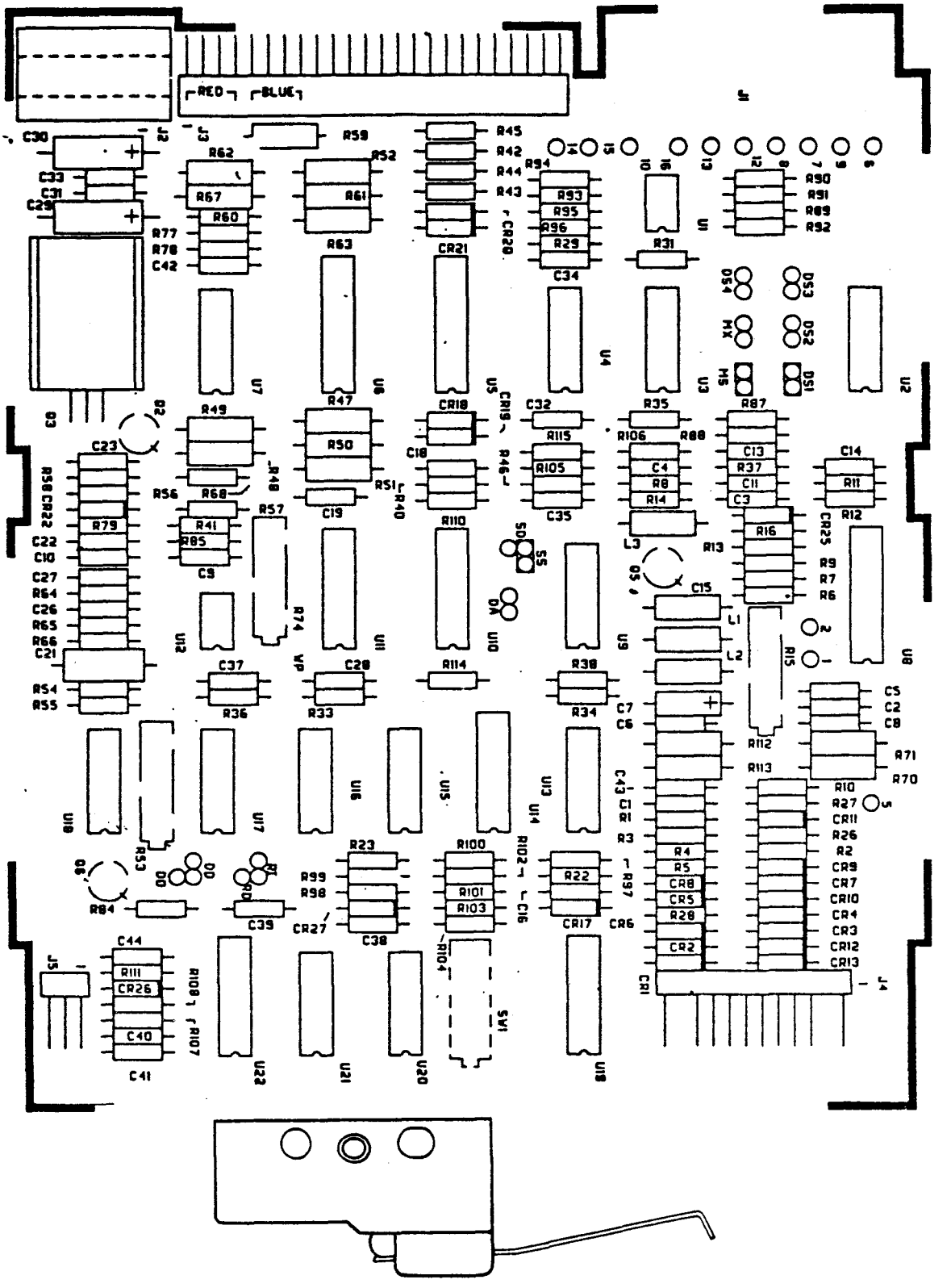


FIGURE A-17. SHUGART SA410 JUMPER LOCATIONS

A.18.3 Recommended Shugart SA410 Jumper Settings

Jumper Name	Termination Board Jumper In or Out	Intermediate Board Jumper In or Out
DS1	See Note 1	See Note 1
DS2	See Note 1	See Note 1
DS3	See Note 1	See Note 1
DS4	See Note 1	See Note 1
MM	Out	Out
MS	In	In
MX	Out	Out
Termination Resistor Pack	In	Out

Note 1: Only 1 of these can be jumpered at a time.

A.19 SHUGART SA460 FIVE INCH DISK DRIVE

This section includes specifications, drive parameter lists, jumper locations, and jumper settings. For the most up-to-date information, consult your disk drive user's manual.

A.19.1 Specifications for Drive Parameter Lists

Word 0	Number of Sides: 2 Single Density Write Precompensation: inner ½ Motor On with Drive Select: Yes	
Word 1	Maximum Number of Tracks per side: 80	0050 (hex) (96 TPI, 160 tracks total)
Word 2	Head Step Time: 6 ms	0258 (hex)
Word 3	Head Settle Time: 15 ms	05DC (hex)
Word 4	Head Load Time: 0 ms	0002 (hex)
Word 5	Head Unload Time: 1 sec	03E8 (hex)
Word 6	Density: Single Sync Type: IBM	
Word 7	Sectors per Track: 16 = 0010 (hex) Bytes per Sector: 128	

A.19.2 Recommended Drive Parameter Lists

	IBM SS SD	IBM DS SD	IBM SS DD	IBM DS DD	TI SS DD	TI DS DD	Mod IBM SS DD	Mod IBM DS DD
Word 0	9001	9002						
Word 1	0050	0050						
Word 2	0258	0258						
Word 3	05DC	05DC						
Word 4	0002	0002						
Word 5	03E8	03E8						
Word 6	0000	0000						
Word 7	4080	4080						

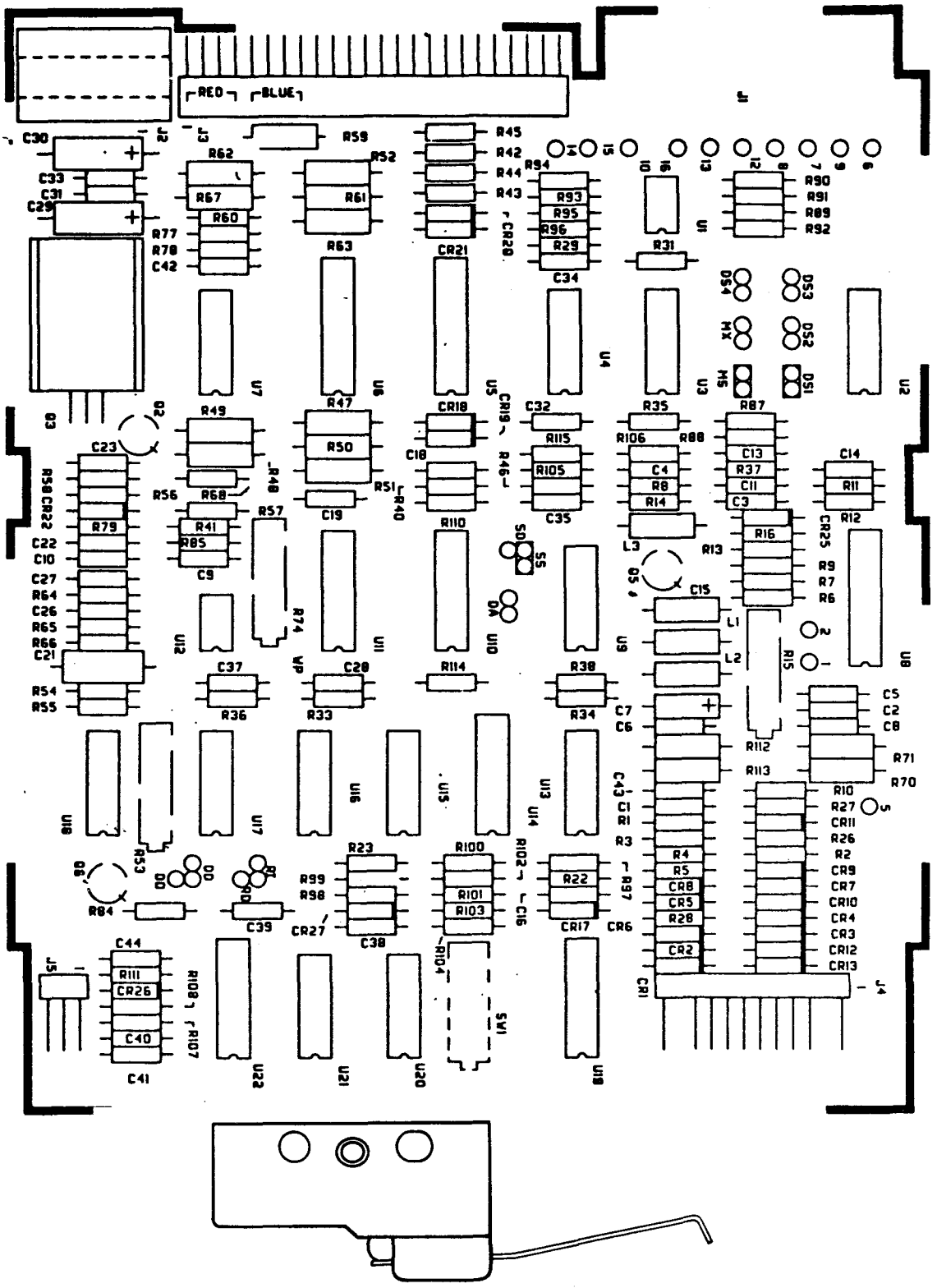


FIGURE A-18. SHUGART SA460 JUMPER LOCATIONS

A.19.3 Recommended Shugart SA460 Jumper Settings

Jumper Name	Termination Board Jumper In or Out	Intermediate Board Jumper In or Out
DS1	See Note 1	See Note 1
DS2	See Note 1	See Note 1
DS3	See Note 1	See Note 1
DS4	See Note 1	See Note 1
MX	Out	Out
MS	In	In
Termination Resistor Pack	In	Out

Note 1: Only 1 of these can be jumpered at a time.

A.20 SIEMENS FDD100-5 FIVE INCH DISK DRIVE

This section includes specifications, drive parameter lists, jumper locations, and jumper settings. For the most up-to-date information, consult your disk drive user's manual.

A.20.1 Specifications for Drive Parameter Lists

Word 0	Number of Sides: 1 Single Density Write Precompensation: none Double Density Write Precompensation: inner ½ Motor On with Drive Select: Yes
Word 1	Maximum Number of Tracks per side: 40 0028 (hex) (40 tracks total)
Word 2	Head Step Time: 25 ms 09C4 (hex)
Word 3	Step Settling Time: 15 ms 05DC (hex)
Word 4	Head Load Time: 50 ms 1388 (hex)
Word 5	Head Unload Timeout: 1 sec 03E8 (hex)
Word 6	Density: Single or Double Sync Type: IBM
Word 7	Sectors per Track: 16 = 0010 (hex) Bytes per Sector: 128 or 256

A.20.2 Recommended Drive Parameter Lists

	IBM SS SD	IBM DS SD	IBM SS DD	IBM DS DD	TI SS DD	TI DS DD	Mod IBM SS DD	Mod IBM DS DD
Word 0	0001		9001					
Word 1	0028		0028					
Word 2	09C4		09C4					
Word 3	05DC		05DC					
Word 4	1388		1388					
Word 5	03E8		03E8					
Word 6	0000		1000					
Word 7	4080		4100					

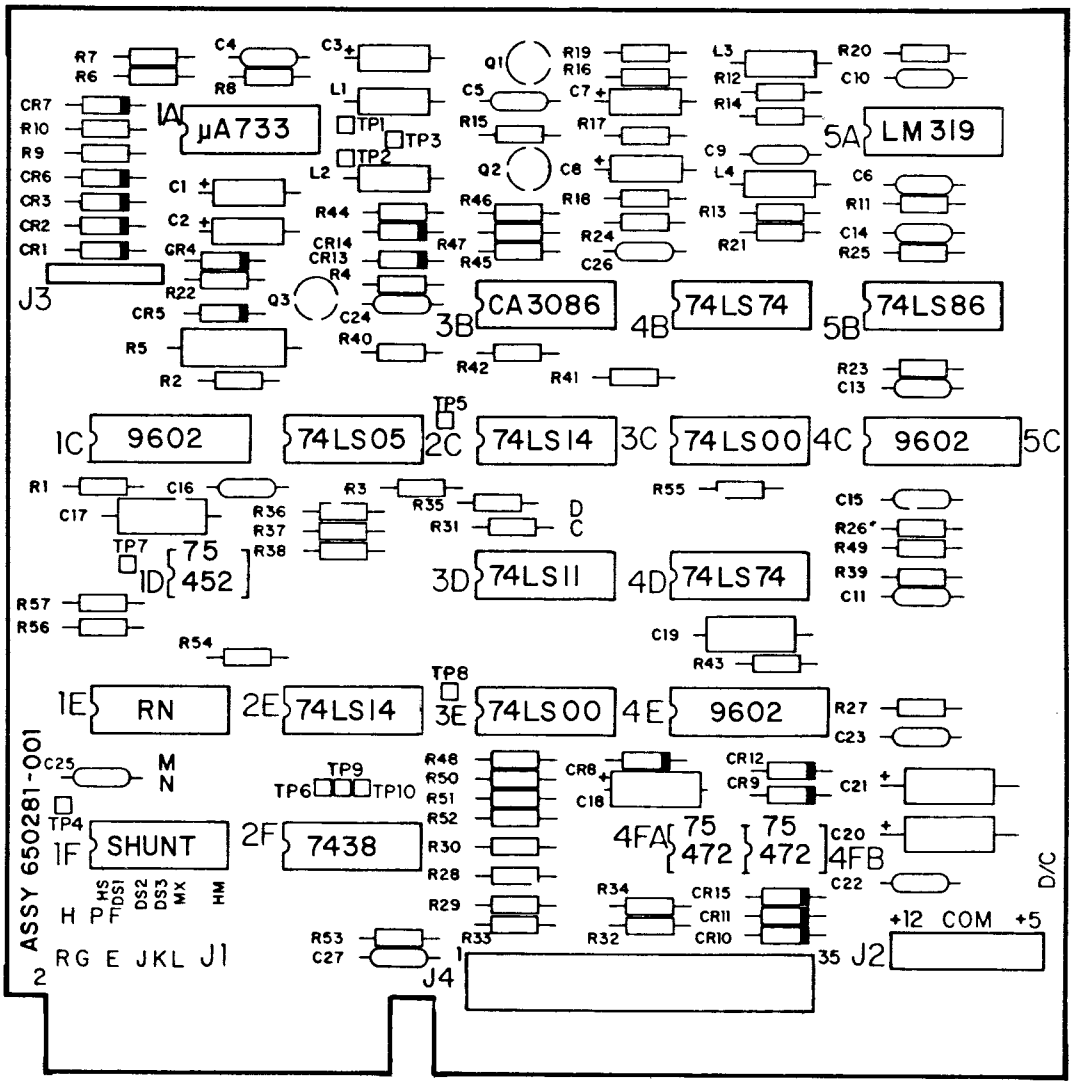


FIGURE A-19. SIEMENS FDD100-5 JUMPER LOCATIONS

A.20.3 Recommended Siemens FDD100-5 Jumper Settings

Jumper Name	Termination Board Jumper In or Out	Intermediate Board Jumper In or Out
HS	In	In
DS1	See Note 1	See Note 1
DS2	See Note 1	See Note 1
DS3	See Note 1	See Note 1
DS0	See Note 1	See Note 1
MX	Out	Out
DL	In	In
HM	Out	Out
Termination Resistor Pack at Position 1E	In	Out

Note 1: Only 1 of these can be jumpered at a time.

A.21 SIEMENS FDD200-5 FIVE INCH DISK DRIVE

This section includes specifications, drive parameter lists, jumper locations, and jumper settings. For the most up-to-date information, consult your disk drive user's manual.

A.21.1 Specifications for Drive Parameter Lists

Word 0	Number of Sides: 2 Single Density Write Precompensation: none Double Density Write Precompensation: all over disk Motor On with Drive Select: Yes
Word 1	Maximum Number of Tracks per side: 40 0028 (hex) (80 tracks total)
Word 2	Head Step Time: 25 ms 09C4 (hex)
Word 3	Step Settling Time: 15 ms 05DC (hex)
Word 4	Head Load Time: 50 ms 1388 (hex)
Word 5	Head Unload Timeout: 1 sec 03E8 (hex)
Word 6	Density: Single or Double Sync Type: IBM
Word 7	Sectors per Track: 16 = 0010 (hex) Bytes per Sector: 128 or 256

A.21.2 Recommended Drive Parameter Lists

	IBM SS SD	IBM DS SD	IBM SS DD	IBM DS DD	TI SS DD	TI DS DD	Mod IBM SS DD	Mod IBM DS DD
Word 0	0001	0002	8001	8002				
Word 1	0028	0028	0028	0028				
Word 2	09C4	09C4	09C4	09C4				
Word 3	05DC	05DC	05DC	05DC				
Word 4	1388	1388	1388	1388				
Word 5	03E8	03E8	03E8	03E8				
Word 6	0000	0000	1000	1000				
Word 7	4080	4080	4100	4100				

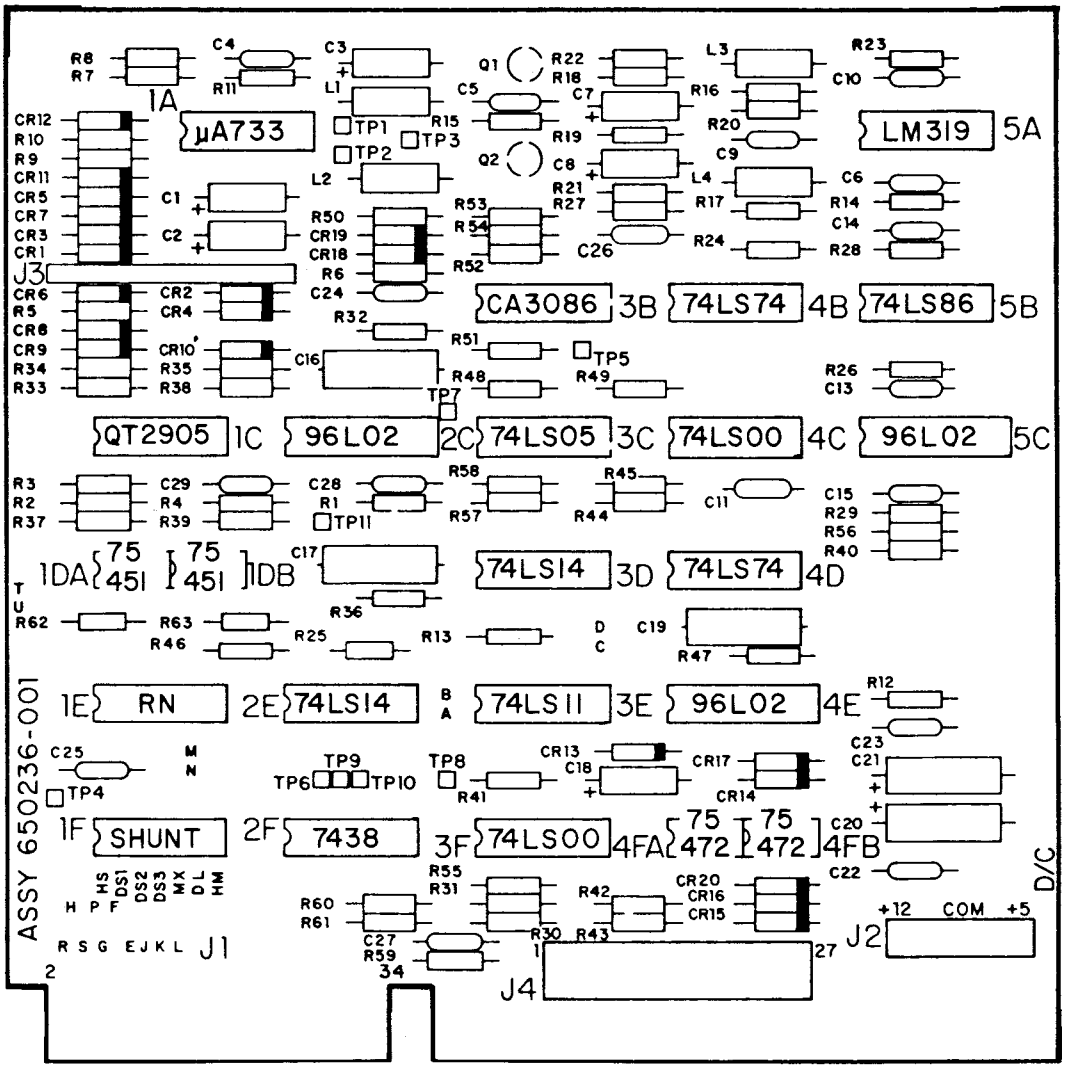


FIGURE A-20. SIEMENS FDD200-5 JUMPER LOCATIONS

A.21.3 Recommended Siemens FDD200-5 Jumper Settings

Jumper Name	Termination Board Jumper In or Out	Intermediate Board Jumper In or Out
HS	In	In
DS1	See Note 1	See Note 1
DS2	See Note 1	See Note 1
DS3	See Note 1	See Note 1
DS0	See Note 1	See Note 1
MX	Out	Out
DL	In	In
HM	Out	Out
Termination Resistor Pack at Position 1E	In	Out

Note 1: Only 1 of these can be jumpered at a time.

A.22 SIEMENS FDD196-5 FIVE INCH DISK DRIVE

This section includes specifications, drive parameter lists, jumper locations, and jumper settings. For the most up-to-date information, consult your disk drive user's manual.

A.22.1 Specifications for Drive Parameter Lists

Word 0	Number of Sides: 1 Single Density Write Precompensation: inner ½ Motor On with Drive Select: Yes
Word 1	Maximum Number of Tracks per side: 80 0050 (hex) (96 TPI, 80 tracks total)
Word 2	Head Step Time: 20 ms 07D0 (hex)
Word 3	Step Settling Time: 15 ms 05DC (hex)
Word 4	Head Load Time: 50 ms 1388 (hex)
Word 5	Head Unload Timeout: 1 sec 03E8 (hex)
Word 6	Density: Single Sync Type: IBM
Word 7	Sectors per Track: 16 = 0010 (hex) Bytes per Sector: 128

A.22.2 Recommended Drive Parameter Lists

	IBM SS SD	IBM DS SD	IBM SS DD	IBM DS DD	TI SS DD	TI DS DD	Mod IBM SS DD	Mod IBM DS DD
Word 0	9001							
Word 1	0050							
Word 2	07D0							
Word 3	05DC							
Word 4	1388							
Word 5	03E8							
Word 6	0000							
Word 7	4080							

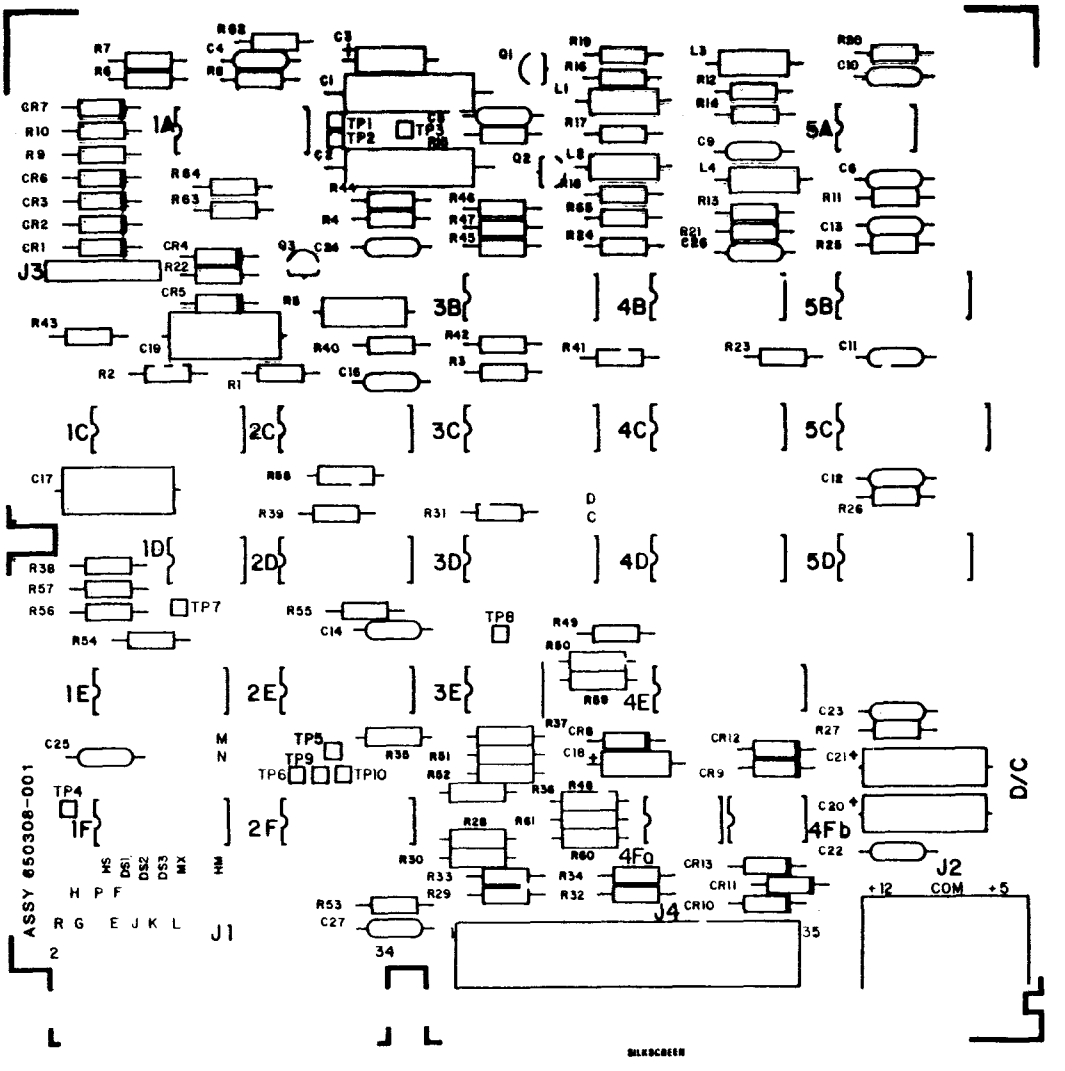


FIGURE A-21. SIEMENS FDD196-5 JUMPER LOCATIONS

A.22.3 Recommended Siemens FDD196-5 Jumper Settings

Jumper Name	Termination Board Jumper In or Out	Intermediate Board Jumper In or Out
HS	In	In
DS1	See Note 1	See Note 1
DS2	See Note 1	See Note 1
DS3	See Note 1	See Note 1
MX	Out	Out
Spare	Out	Out
HM	Out	Out
Termination Resistor Pack	In	Out

Note 1: Only 1 of these can be jumpered at a time.

A.23 SIEMENS FDD296-5 FIVE INCH DISK DRIVE

This section includes specifications, drive parameter lists, jumper locations, and jumper settings. For the most up-to-date information, consult your disk drive user's manual.

A.23.1 Specifications for Drive Parameter Lists

Word 0	Number of Sides: 2 Single Density Write Precompensation: none Motor On with Drive Select: Yes	
Word 1	Maximum Number of Tracks per side: 80	0050 (hex) (96 TPI, 160 tracks total)
Word 2	Head Step Time: 6 ms	0258 (hex)
Word 3	Step Settling Time: 15 ms	05DC (hex)
Word 4	Head Load Time: 50 ms	1388 (hex)
Word 5	Head Unload Timeout: 1 sec	03E8 (hex)
Word 6	Density: Single Sync Type: IBM	
Word 7	Sectors per Track: 16 = 0010 (hex) Bytes per Sector: 128	

A.23.2 Recommended Drive Parameter Lists

	IBM SS SD	IBM DS SD	IBM SS DD	IBM DS DD	TI SS DD	TI DS DD	Mod IBM SS DD	Mod IBM DS DD
Word 0	0001	0002						
Word 1	0050	0050						
Word 2	0258	0258						
Word 3	05DC	05DC						
Word 4	1388	1388						
Word 5	03E8	03E8						
Word 6	0000	0000						
Word 7	4080	4080						

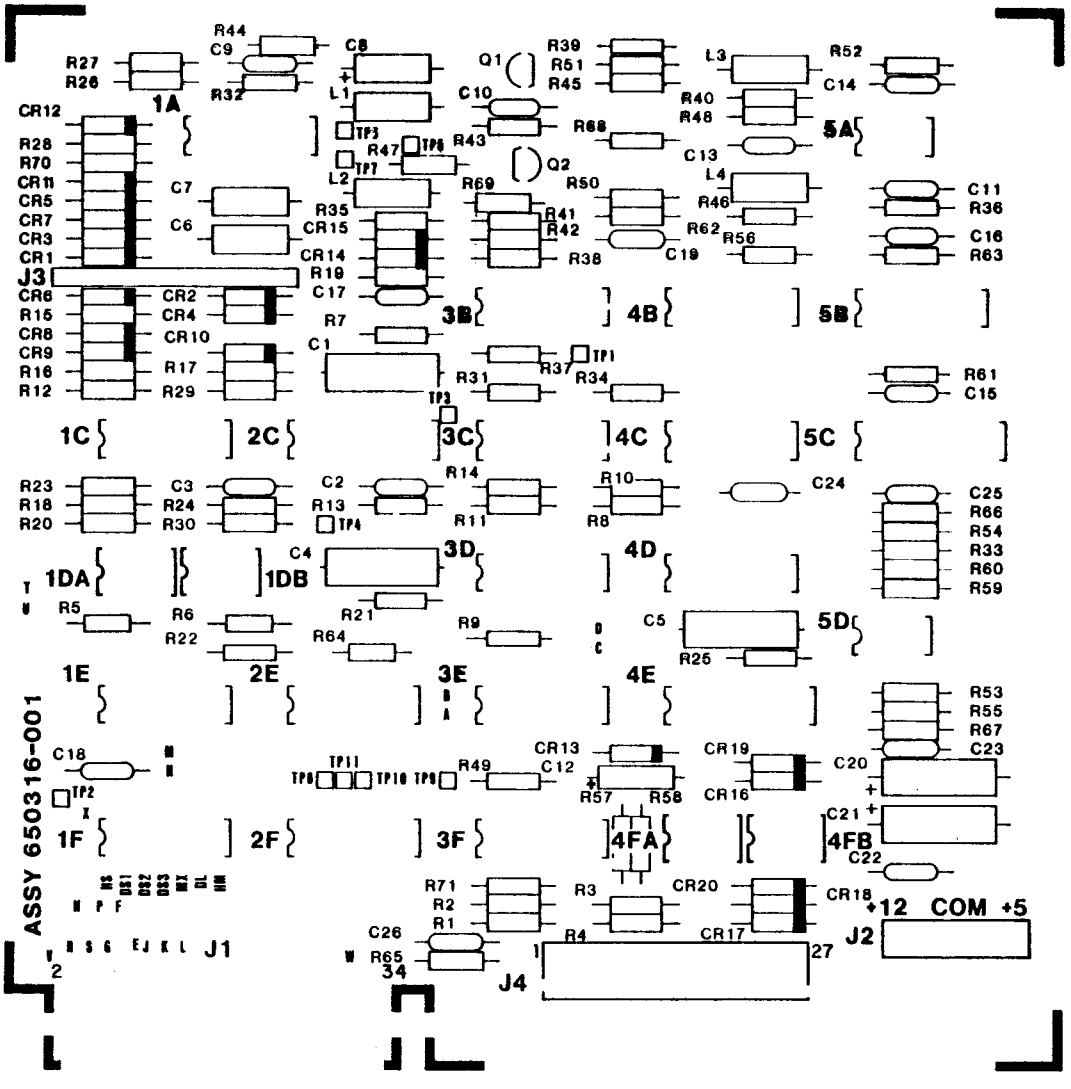


FIGURE A-22. SIEMENS FDD296-5 JUMPER LOCATIONS

A.23.3 Recommended Siemens FDD296-5 Jumper Settings

Jumper Name	Termination Board Jumper In or Out	Intermediate Board Jumper In or Out
HS	In	In
DS1	See Note 1	See Note 1
DS2	See Note 1	See Note 1
DS3	See Note 1	See Note 1
DS0	See Note 1	See Note 1
MX	Out	Out
DL	In	In (Door Lock Option)
HM	Out	Out
Termination Resistor Pack at Position 1E	In	Out

Note 1: Only 1 of these can be jumpered at a time.

A.24 TANDON TM100-1 FIVE INCH DISK DRIVE

This section includes specifications, drive parameter lists, jumper locations, and jumper settings. For the most up-to-date information, consult your disk drive user's manual.

A.24.1 Specifications for Drive Parameter Lists

Options: No Head Load Solenoid, the heads are loaded all the time.

Word 0	Number of Sides: 1 Single Density Write Precompensation: inner ½ Double Density Write Precompensation: inner ½ Motor On with Drive Select: Yes
Word 1	Maximum Number of Tracks per side: 40 0028 (hex) (40 tracks total)
Word 2	Head Step Time: 5 ms 01F4 (hex)
Word 3	Step Settling Time: 15 05DC (hex)
Word 4	Head Load Time: 0 ms 0002 (hex)
Word 5	Head Unload Timeout: 1 sec 03E8 (hex)
Word 6	Density: Single or Double Sync Type: IBM
Word 7	Sectors per Track: 16 = 0010 (hex) Bytes per Sector: 128 or 256

A.24.2 Recommended Drive Parameter Lists

	IBM SS SD	IBM DS SD	IBM SS DD	IBM DS DD	TI SS DD	TI DS DD	Mod IBM SS DD	Mod IBM DS DD
Word 0	9001		9001					
Word 1	0028		0028					
Word 2	01F4		01F4					
Word 3	05DC		05DC					
Word 4	0002		0002					
Word 5	03E8		03E8					
Word 6	0000		1000					
Word 7	4080		4100					

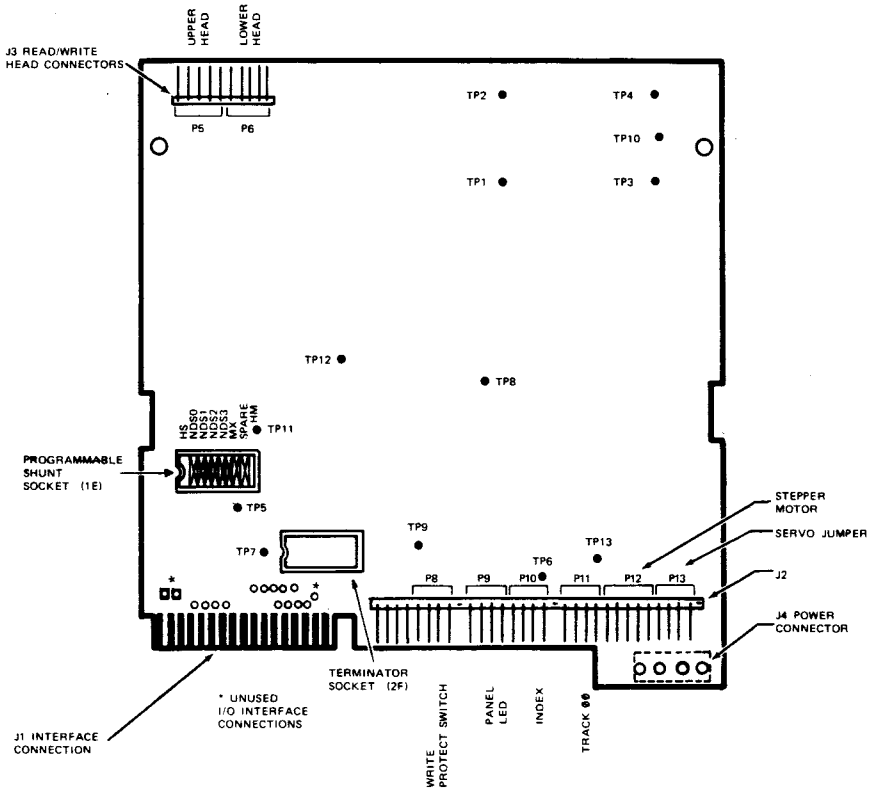


FIGURE A-23. TANDON TM100-1 JUMPER LOCATIONS

A.24.3 Recommended Tandon TM100-1 Jumper Settings

Jumper Name	Termination Board Jumper In or Out	Intermediate Board Jumper In or Out
HS	In	In
NDS0	See Note 1	See Note 1
NDS1	See Note 1	See Note 1
NDS2	See Note 1	See Note 1
NDS3	See Note 1	See Note 1
MX	Out	Out
HM	Out	Out
Termination Resistor Pack At Position 2F	In	Out

Note 1: Only 1 of these can be jumpered at a time.

A.25 TANDON TM100-3 FIVE INCH DISK DRIVE

This section includes specifications, drive parameter lists, jumper locations, and jumper settings. For the most up-to-date information, consult your disk drive user's manual.

A.25.1 Specifications for Drive Parameter Lists

Word 0	Number of Sides: 1 Single Density Write Precompensation: inner ½ Double Density Write Precompensation: inner ½ Motor On with Drive Select: Yes	
Word 1	Maximum Number of Tracks per side: 80	0050 (hex) (96 TPI, 80 tracks total)
Word 2	Head Step Time: 3 ms	012C (hex)
Word 3	Step Settling Time: 15 ms	05DC (hex)
Word 4	Head Load Time: 50 ms	1388 (hex)
Word 5	Head Unload Timeout: 1 sec	03E8 (hex)
Word 6	Density: Single or Double Sync Type: IBM	
Word 7	Sectors per Track: 16 = 0010 (hex) Bytes per Sector: 128 or 256	

A.25.2 Recommended Drive Parameter Lists

	IBM SS SD	IBM DS SD	IBM SS DD	IBM DS DD	TI SS DD	TI DS DD	Mod IBM SS DD	Mod IBM DS DD
Word 0	9001		9001					
Word 1	0050		0050					
Word 2	012C		012C					
Word 3	05DC		05DC					
Word 4	1388		1388					
Word 5	03E8		03E8					
Word 6	0000		1000					
Word 7	4080		4100					

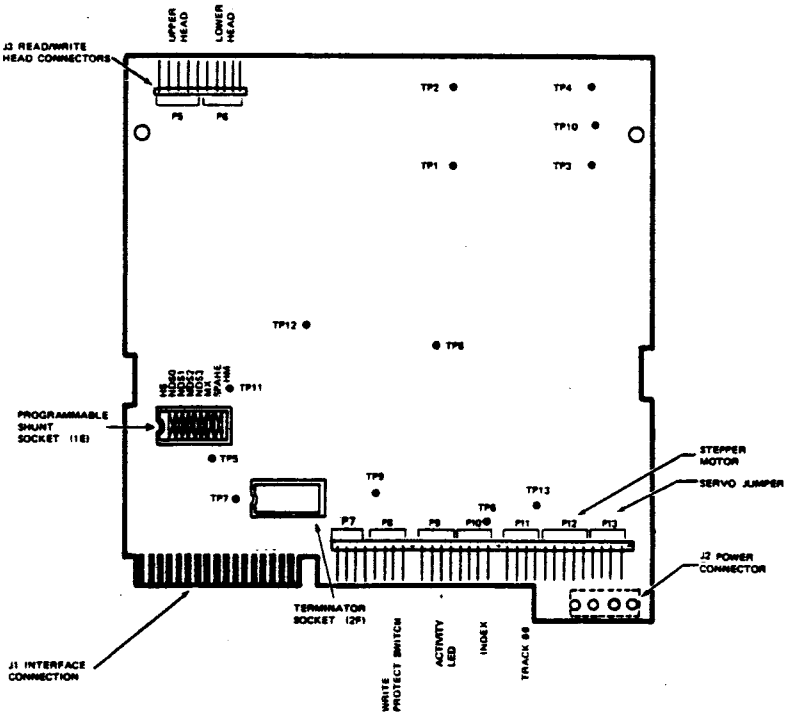


FIGURE A-24. TANDON TM100-3 JUMPER LOCATIONS

A.25.3 Recommended Tandon TM100-3 Jumper Settings

Jumper Name	Termination Board Jumper In or Out	Intermediate Board Jumper In or Out
HS	In	In
NDS0	See Note 1	See Note 1
NDS1	See Note 1	See Note 1
NDS2	See Note 1	See Note 1
NDS3	See Note 1	See Note 1
MX	Out	Out
HM	Out	Out
Termination Resistor Pack	In	Out

Note 1: Only 1 of these can be jumpered at a time.

A.26 TANDON TM100-4 FIVE INCH DISK DRIVE

This section includes specifications, drive parameter lists, jumper locations, and jumper settings. For the most up-to-date information, consult your disk drive user's manual.

A.26.1 Specifications for Drive Parameter Lists

Word 0	Number of Sides: 2 Single Density Write Precompensation: inner ½ Double Density Write Precompensation: inner ½ Motor On with Drive Select: Yes
Word 1	Maximum Number of Tracks per side: 80 0050 (hex) (96 TPI, 160 tracks total)
Word 2	Head Step Time: 3 ms 012C (hex)
Word 3	Step Settling Time: 15 ms 05DC (hex)
Word 4	Head Load Time: 50 ms 1388 (hex)
Word 5	Head Unload Timeout: 1 sec 03E8 (hex)
Word 6	Density: Single or Double Sync Type: IBM
Word 7	Sectors per Track: 16 = 0010 (hex) Bytes per Sector: 128 or 256

A.26.2 Recommended Drive Parameter Lists

	IBM SS SD	IBM DS SD	IBM SS DD	IBM DS DD	TI SS DD	TI DS DD	Mod IBM SS DD	Mod IBM DS DD
Word 0	9001	9002	9001	9002				
Word 1	0050	0050	0050	0050				
Word 2	012C	012C	012C	012C				
Word 3	05DC	05DC	05DC	05DC				
Word 4	1388	1388	1388	1388				
Word 5	03E8	03E8	03E8	03E8				
Word 6	0000	0000	1000	1000				
Word 7	4080	4080	4100	4100				

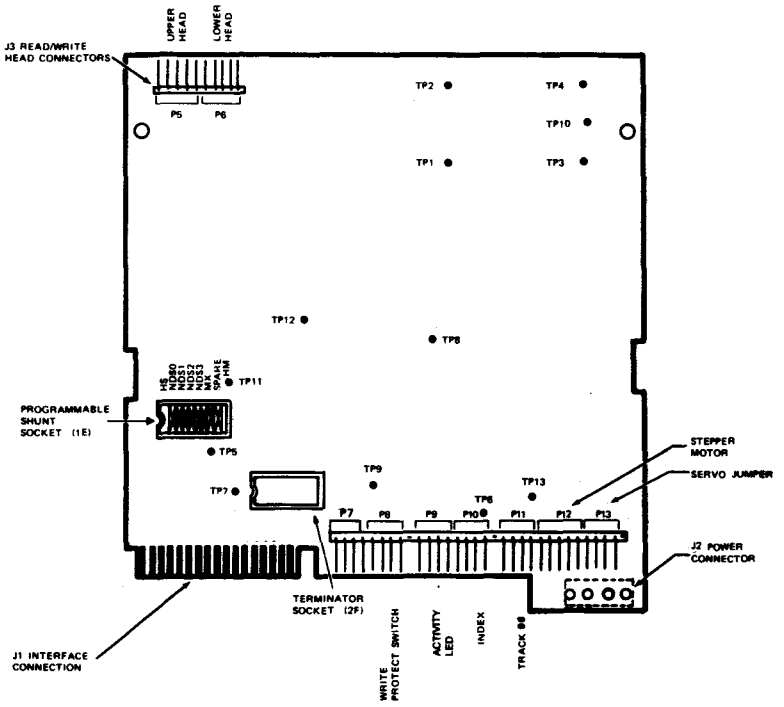


FIGURE A-25. TANDON TM100-4 JUMPER LOCATIONS

A.26.3 Recommended Tandon TM100-4 Jumper Settings

Jumper Name	Termination Board Jumper In or Out	Intermediate Board Jumper In or Out
HS	In	In
NDS0	See Note 1	See Note 1
NDS1	See Note 1	See Note 1
NDS2	See Note 1	See Note 1
NDS3	See Note 1	See Note 1
MX	Out	Out
Spare	Out	Out
HM	Out	Out

Note 1: Only 1 of these can be jumpered at a time.

APPENDIX B

DISK DRIVE SPECIFICATIONS

TABLE B-1. MINI (5¼ INCH) FLOPPY DRIVE SPECIFICATIONS

	Single Density	Double Density	Measure
PERFORMANCE SPECIFICATIONS			
Diskette Capacity			
Unformatted			
Per Surface	875	1750	kilobits
Per Track	25	50	kilobits
IBM Format			
Per Surface	71,680	143,360	bytes
Per Track	2,048	4,096	bytes
Per Sector	128	256	bytes
Transfer Rate			
Total	125	250	kilobits/sec
Data	81	162	kilobits/sec
Rotational Latency (Avg)	100	100	milliseconds
Seek Time			
Track to Track	40	40	milliseconds
Average	463	463	milliseconds
Settling Time	10	10	milliseconds
Head Load Time	75	75	milliseconds
FUNCTIONAL SPECIFICATIONS			
Rotational Speed	300	300	rpm
Recording Density (Inside Track)	2581	5162	bpi
Flux Density	5162	5162	fci
Track Density	48	48	tpi
Tracks	35	35	
Sectors per Track (Soft)	16	16	
Index	1	1	
Encoding Method	FM	MFM	
Media Requirements (2 Sided)	SA104	SA104	

TABLE B-2. STANDARD (8 INCH) FLOPPY DRIVE SPECIFICATIONS

	Single Density	Double Density	Measure
PERFORMANCE SPECIFICATIONS			
Diskette Capacity			
Unformatted			
Per Surface	3.2	6.4	megabits
Per Track	41.7	83.3	kilobits
IBM Format			
Per Surface	256,256	512,512	bytes
Per Track	3,328	6,656	bytes
Per Sector	128	256	bytes
TI Format			
Per Surface		576,576	bytes
Per Track		7,488	bytes
Per Sector		288	bytes
Transfer Rate			
Total	250	500	kilobits/sec
Data	159	318	kilobits/sec
Rotational Latency (Avg)	83	83	milliseconds
Seek Time			
Track to Track	10	10	milliseconds
Average	260	260	milliseconds
Settling Time	8	8	milliseconds
Head Load Time	35	35	milliseconds
FUNCTIONAL SPECIFICAITIONS			
Rotational Speed	360	360	rpm
Recording Density (Inside Track)	3200	6400	bpi
Flux Density	6400	6400	fci
Track Density	48	48	tpi
Tracks	77	77	
Sectors per Track (Soft)	26	26	
Index	1	1	
Encoding Method	FM	MFM	
Media Requirements	SA100/IBM Diskette II	SA100/IBM Diskette II	

APPENDIX C

DISKETTE TRACK FORMATS

C.1 GENERAL

This appendix shows the format associated with the five formats compatible with the TM 990/303B floppy disk controller:

Format	Figure
Standard-size, single-density track format	C-1
IBM-compatible, standard-size, double-density track format	C-2
IBM-compatible, standard-size, modified double-density track format	C-3
TI-compatible, standard-size, double-density track format	C-4
Mini-size, single-density track format	C-5
Mini-size, double-density track format	C-6

C.2 NOMENCLATURE

In the figures in this appendix, circled numbers are used to identify various fields and markers. These numbers represent the following:

① Sync Field. This field synchronizes the diskette drive circuitry to the information being read from the diskette.

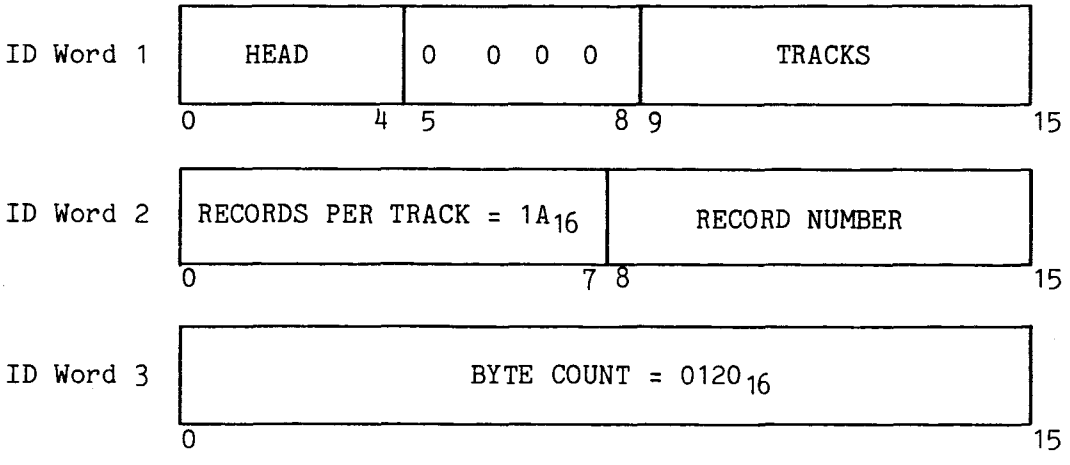
② AM1 (Address Marker 1). This marker identifies the information that follows as being the ID field.

③ ID Field.

a. Formats other than TI double density. The ID field consists of four bytes of information identifying the address and size of the sector.

- First byte. Track number (00_{16} thru $4C_{16}$)
- Second byte. Head number (00_{16} = side 0, 01_{16} = side 1)
- Third byte. Record number with 1 record per sector (1 thru 26)
- Fourth byte. Physical record length in bytes:
  00_{16} = 128 bytes
  01_{16} = 256 bytes

b. TI double density format. Three words as follows:



④ Cyclic Redundancy Check (CRC). The CRC is the 16-bit remainder value generated on a write data or write format operation by performing a polynomial division of the string of bits including the address mark and the data field using the following polynomial divisor:

$$X^{16} + X^{12} + X^5 + 1$$

In addition, to reduce the possibility of a false CRC check, a partial remainder value of $FFFF_{16}$ is preset into the CRC generator prior to any CRC generation or checking operation.

During an ID location or read data operation, the address mark and following data, including the previously written CRC value, are again divided by the divisor polynomial. If the data does not contain any errors, the resulting remainder value in the CRC generator wil be 0000.

⑤ Gap 2. This contains the fixed number of gap data bytes.

⑥ AM2 (Address Marker 2). This identifies the field that follows as being a data record or a deleted data record.

⑦ Data Record. A Data Record field contains the bytes of data.

⑧ Gap 3. This contains two bytes of binary zeroes followed by a variable number of gap data bytes.

⑨ Gap 4B. This is the pre-index gap which consists of a variable number of gap data bytes.

⑩ Gap 4A. This is the post-index gap which consists of a fixed number of gap bytes.

⑪ Track AM. This follows the index mark and identifies the start of a track.

⑫ Gap 1. This consists of two bytes of zeroes followed by a fixed number of gap data bytes.

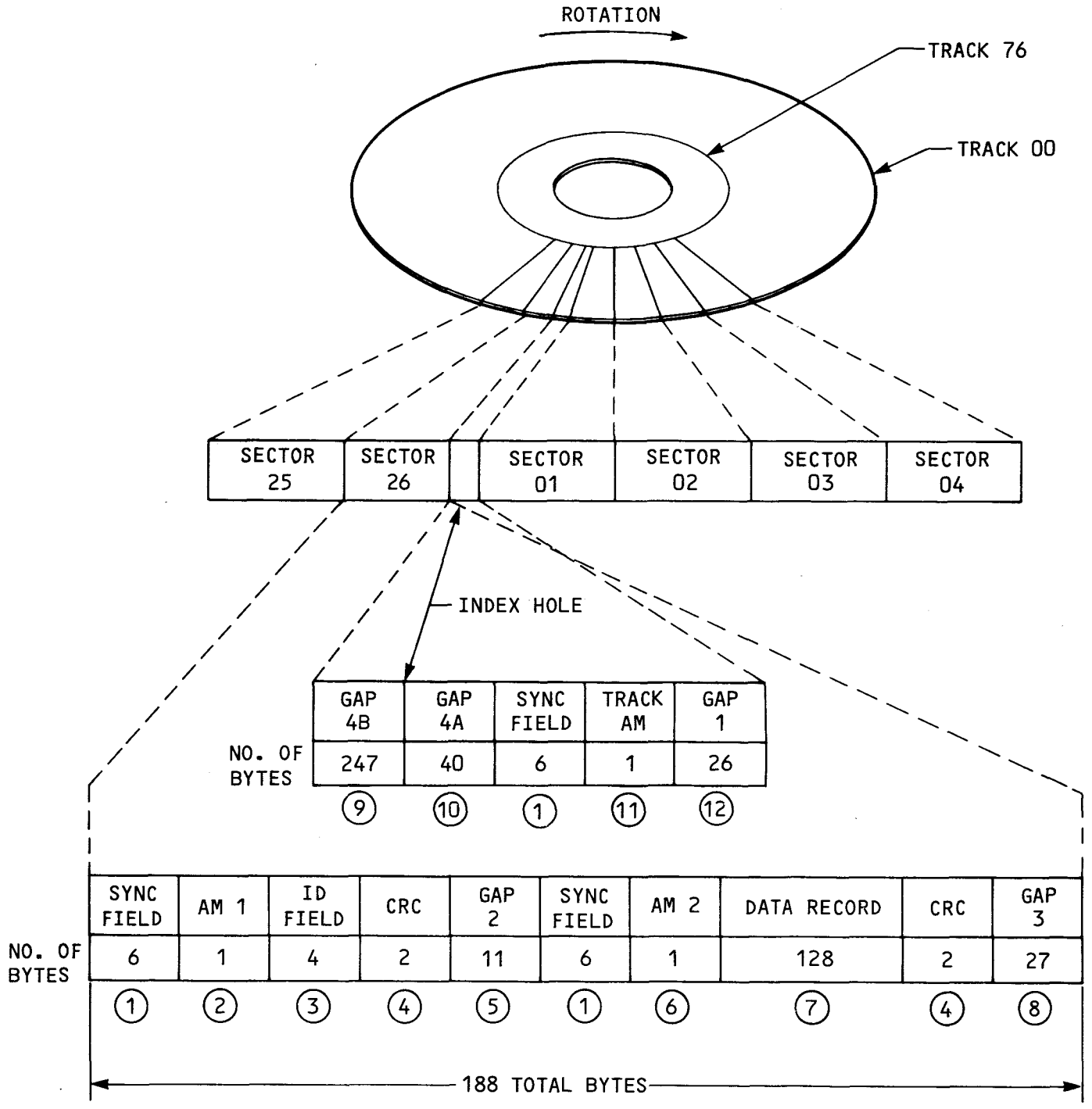
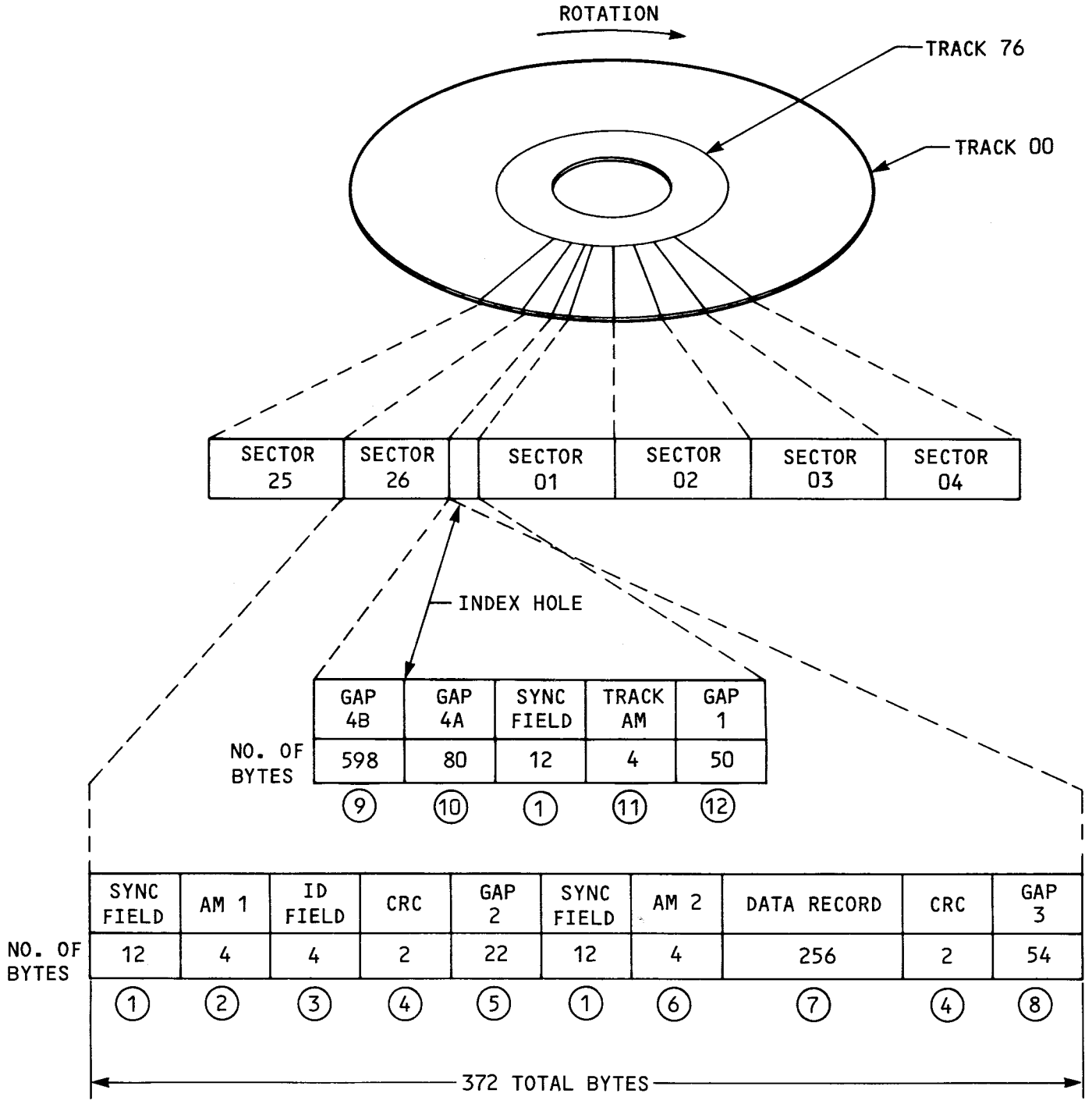


FIGURE C-1. STANDARD-SIZE, SINGLE-DENSITY TRACK FORMAT

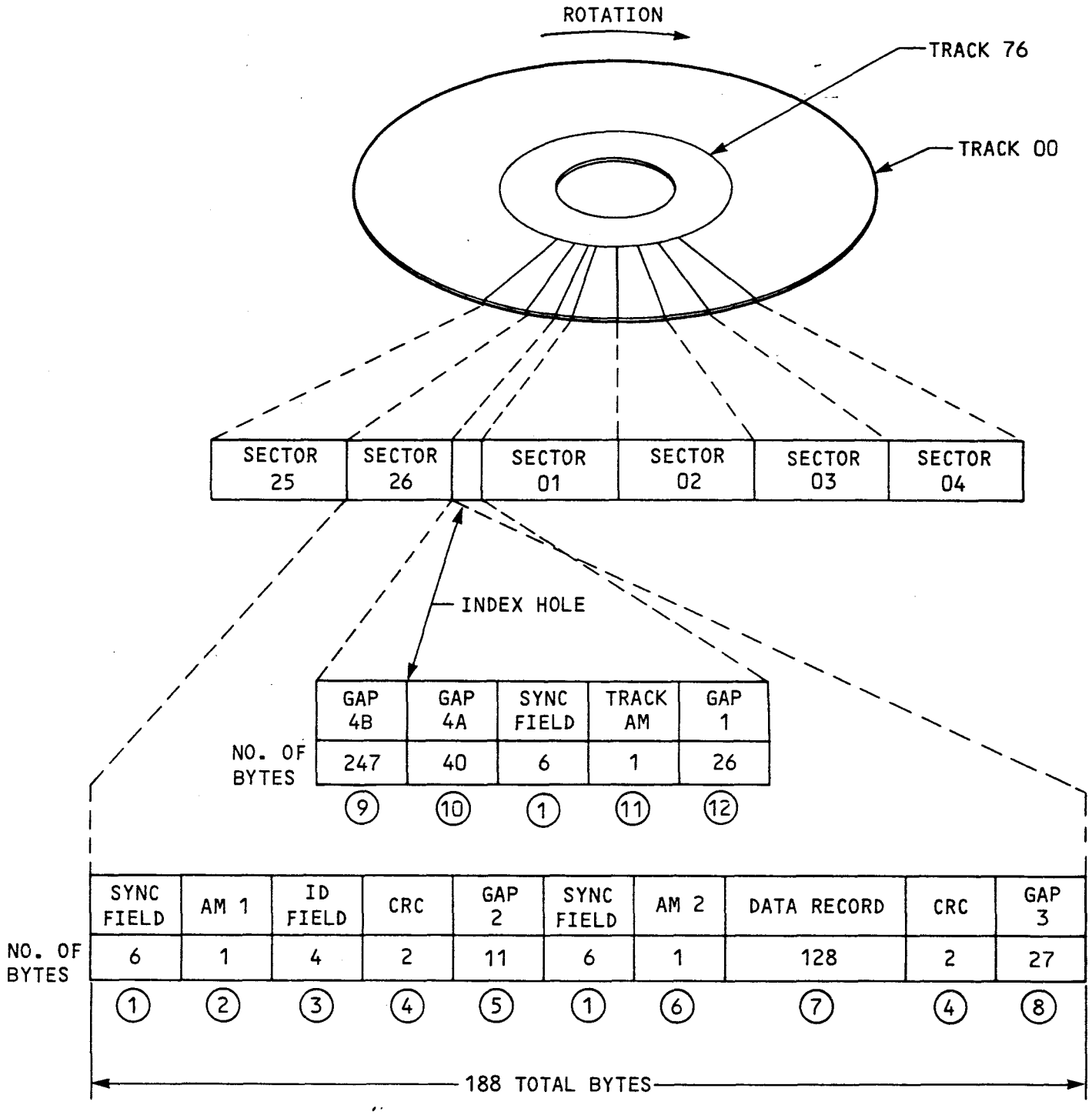
IBM diskette types are explained in greater detail in "The IBM Diskette General Information Manual," IBM publication number GA21-9182-5.



NOTE

Shown above is for all tracks except track 0, side 0; track 0, side 0 shown in next page.

FIGURE C-2. IBM-COMPATIBLE STANDARD-SIZE DOUBLE DENSITY TRACK FORMAT (PAGE 1 OF 2)

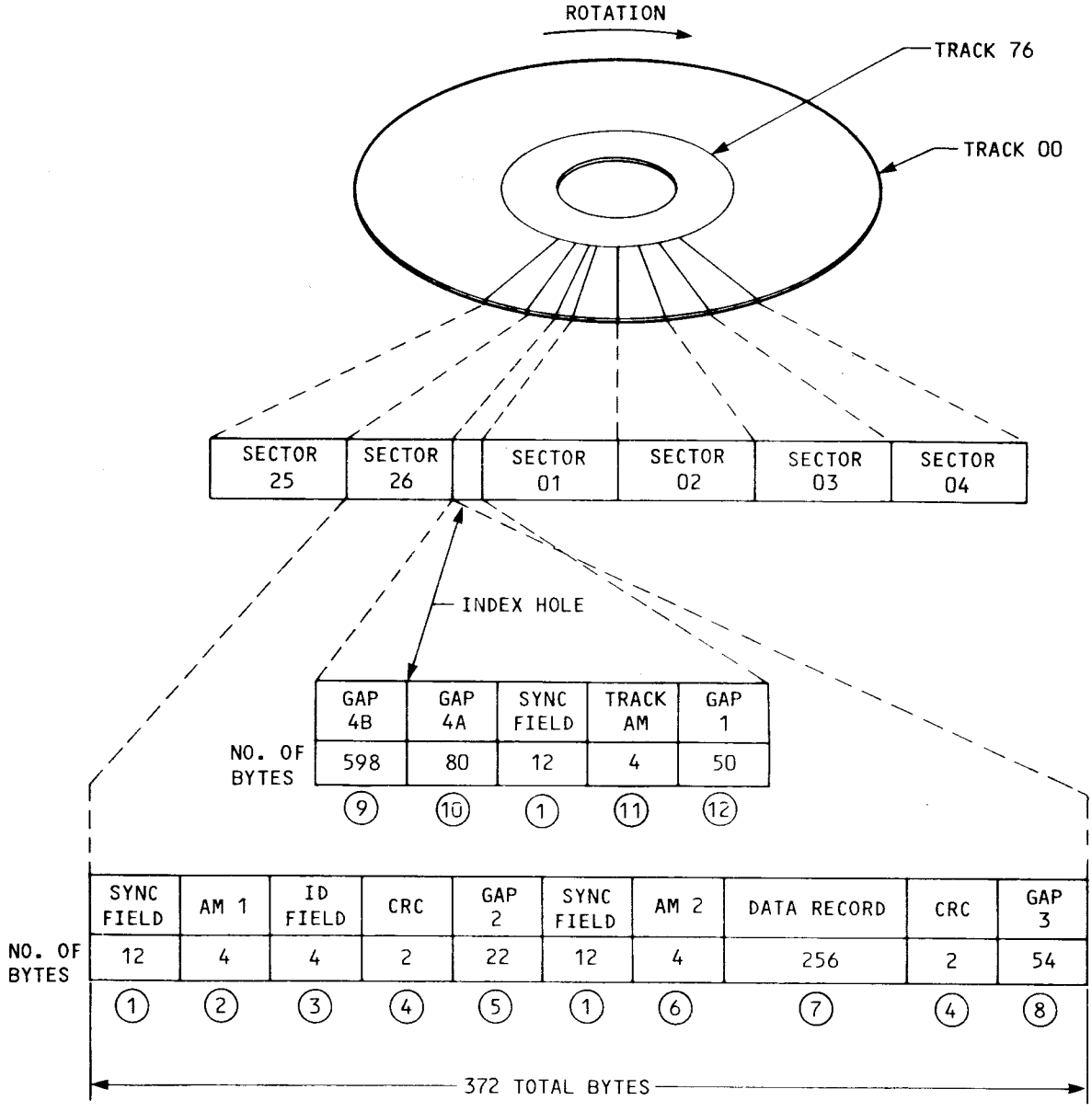


NOTES

1. Shown above is track 0, side 0; other tracks on side 0 and all tracks on side 1 on previous page.

2. Shown are physical sector numbers. In this format for sector 0 only, only logical sectors 1 to 13 are present.

FIGURE C-2. IBM-COMPATIBLE STANDARD-SIZE DOUBLE DENSITY TRACK FORMAT (PAGE 2 OF 2)



NOTE

Shown above is IBM modified double density format which has 256 bytes/sector on all tracks.

FIGURE C-3. IBM-COMPATIBLE STANDARD-SIZE MODIFIED DOUBLE DENSITY TRACK FORMAT

TI double-sided, double-density diskette formats are explained in greater detail in TI's "DSG Group Drawing #2261686 or in the FD1000 flexible system manual (DSG manual #2261886-9701).

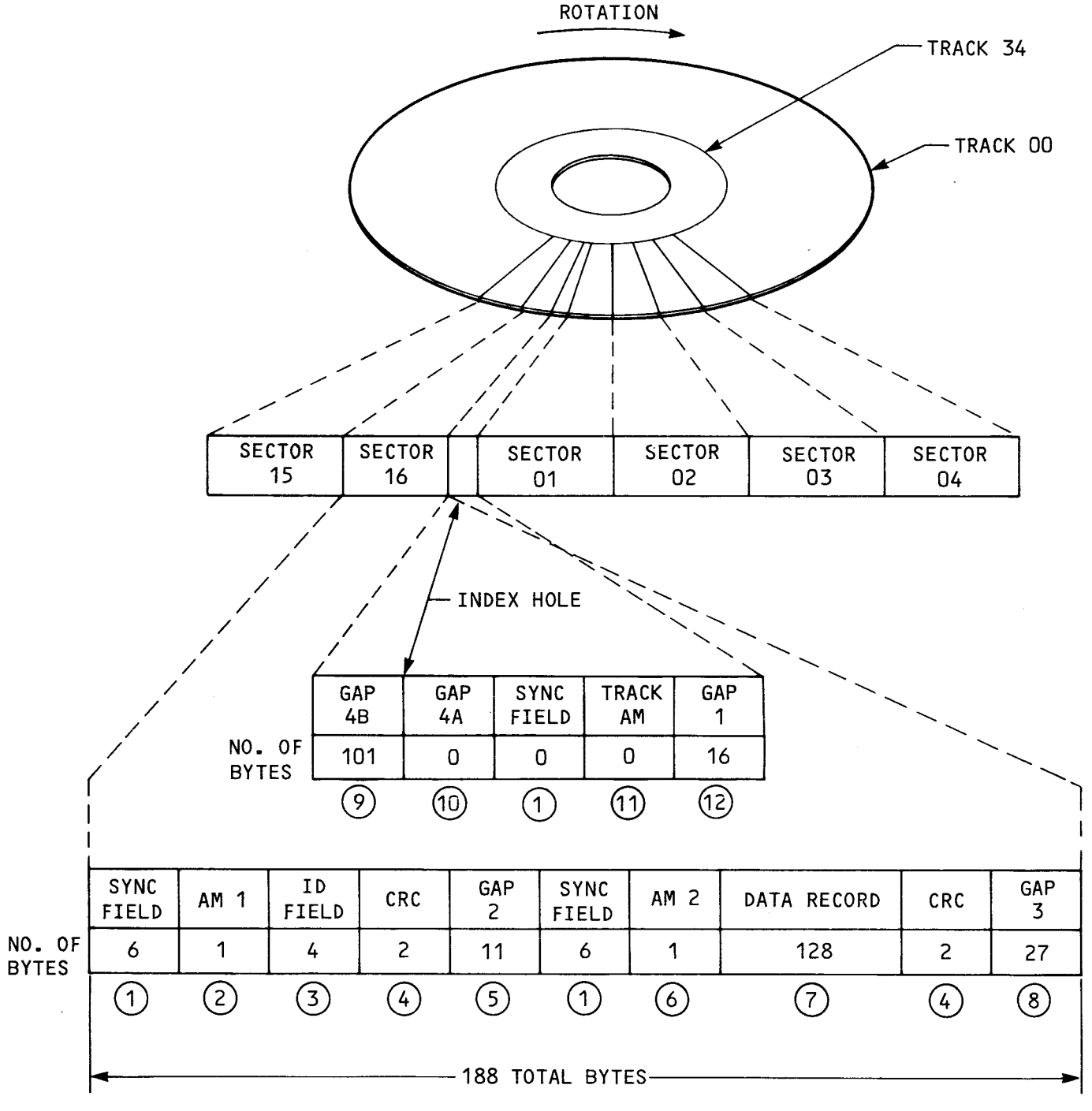


FIGURE C-4. TI-COMPATIBLE, STANDARD-SIZE, DOUBLE-DENSITY TRACK FORMAT

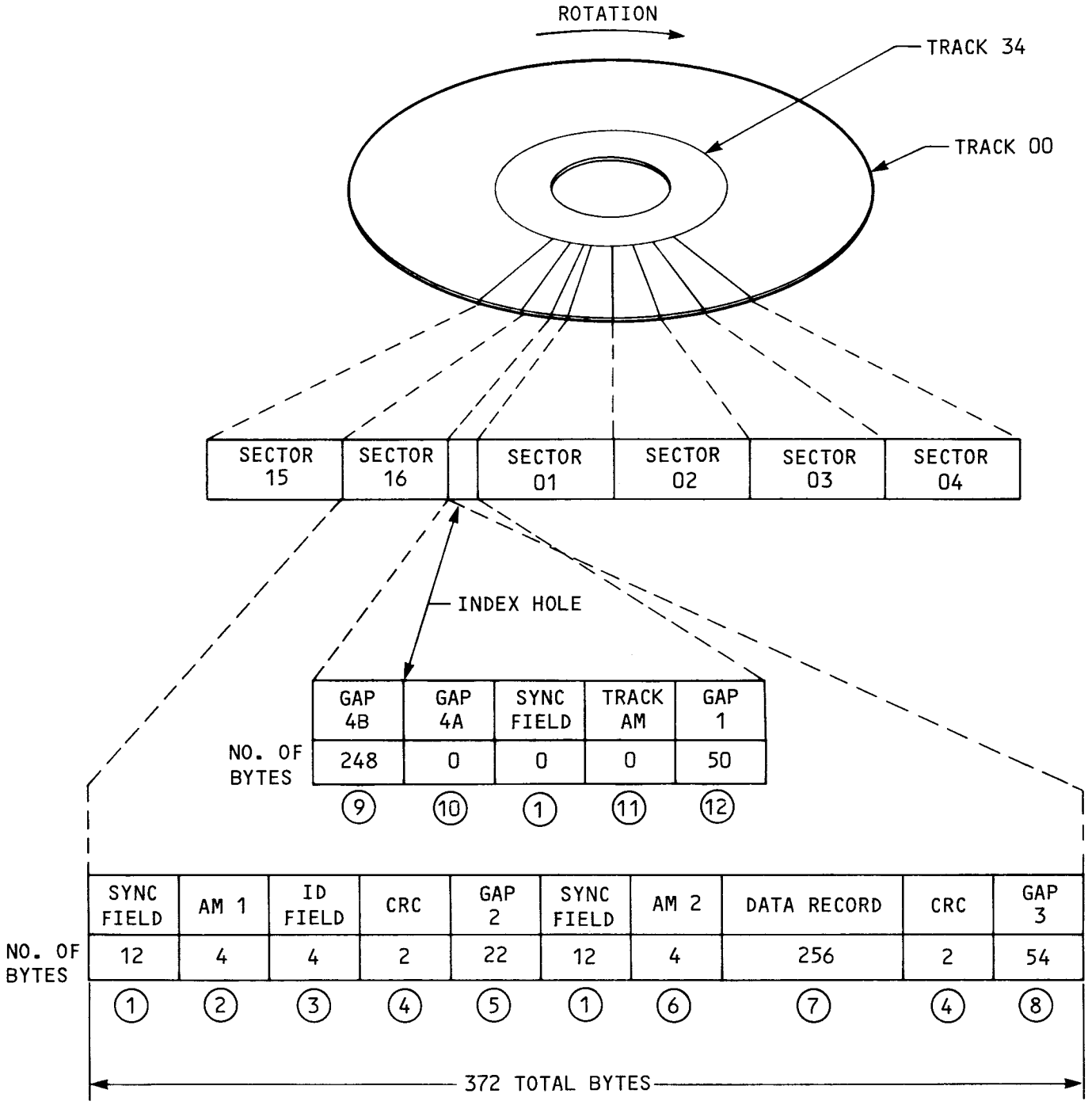


FIGURE C-5. MINI-SIZE, SINGLE-DENSITY TRACK FORMAT

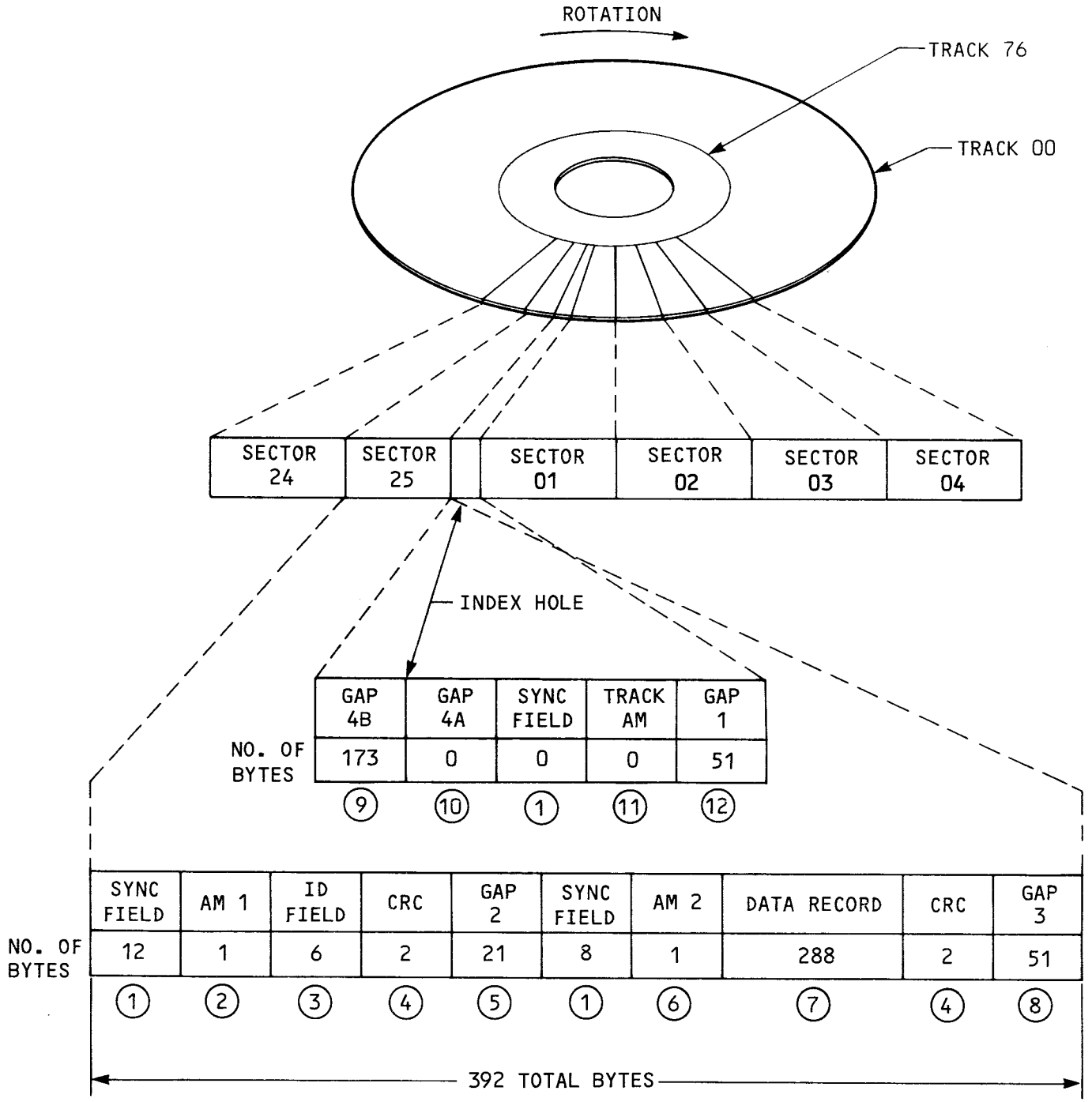


FIGURE C-6. MINI-SIZE, DOUBLE-DENSITY TRACK FORMAT

APPENDIX D

SCHEMATICS

NOTES UNLESS OTHERWISE SPECIFIED

1. ALL CAPACITOR VALUES ARE IN MICROFARADS
2. ALL RESISTOR VALUES ARE IN OHMS
3. ALL RESISTORS TO BE .25 W, 5 %

[4] 14 PIN DEVICES USE PIN 7 FOR GND AND PIN 14 FOR +5V
16 PIN DEVICES USE PIN 8 FOR GND AND PIN 16 FOR +5V
20 PIN DEVICES USE PIN 10 FOR GND AND PIN 20 FOR +5V

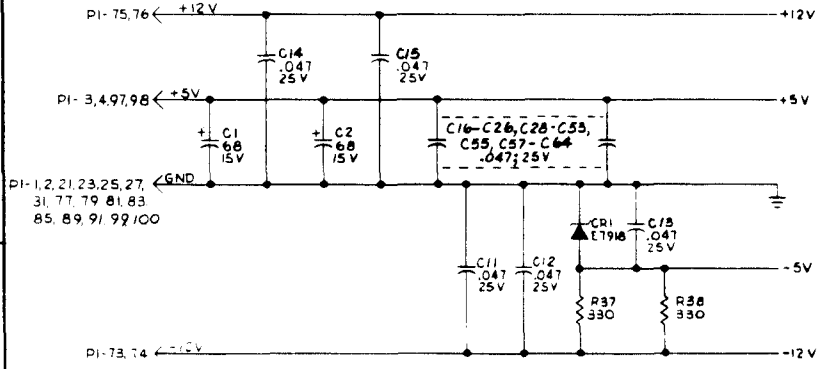
[5] PROGRAMMED DEVICE

[4] DEVICE	GND	+5V	-5V	+12V
TMS2532 (U1, U2)	12	24		
TMS4045 (U3-U6)	9	18		
TMS9900 (U32)	26,40	2,59	1	27
TMS9901 (U48)	16	40		
TMS9902A (U50)	9	18		
TIM9904A (U31)	3,10	20		13
75150 P (U59)	4	8		

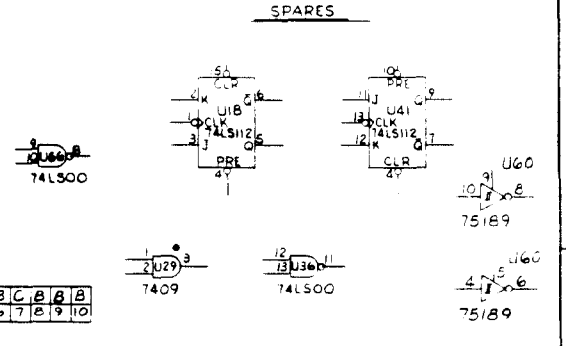
REFERENCE DESIGNATORS	
USED	NOT USED
C1 - C64	C27, C54, C56
CR1	
DS1 - DS3	
E1 - E86	E12-14, E21-44, E50
J1 - J14	J4 - J7
L1, L2	
P1, P2, P4	P3
R1 - R43	R23
U1 - U68	
Y1	

JUMPER CONFIGURATION	
JUMPER	POSITION
J1	E1, E3
J2	E5, E6
J3	E8, E56
J8	E81, E82
J9	E84, E83
J10	E19, E18
J11	E16, E17
J12	E53, E71
J13	E78, E77
J14	E80, E79

REV	DESCRIPTION	AUTH	COMMENTS
A	INFORMAL DESIGN CHANGE	2/1/82	
B	INF. DESIGN CHANGE	3/25/82	
C	CN494999 INF.	5/14/82	



SPARES



REV STATUS OF SHEETS	REV	C	C	B	B	B	B	C	B	B	B
	SH	1	2	3	4	5	6	7	8	9	10

ITEM NO	QTY	PART OR IDENTIFYING NUMBER	NOMENCLATURE OR DESCRIPTION	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 PROCESSING
- PARENTHETICAL INFO FOR REF ONLY

HOLE TOLERANCE

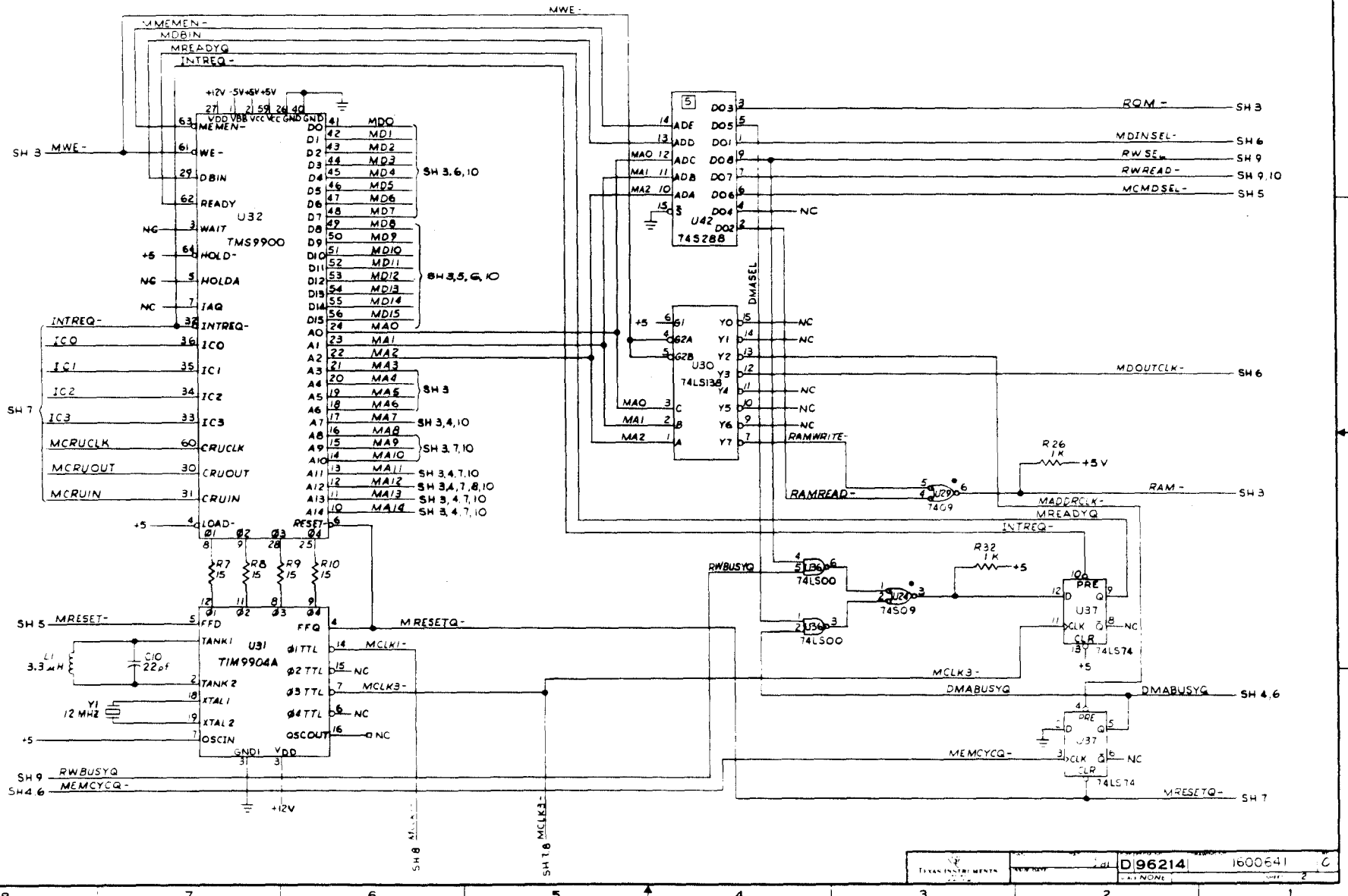
.013 THRU .029 +.004 -.001	.29 THRU .750 +.10 -.03	.75 THRU .900 +.008 -.003
.901 THRU 1.29 +.008 -.001	.751 THRU .900 +.010 -.001	1.001 THRU 2.000 +.012 -.001

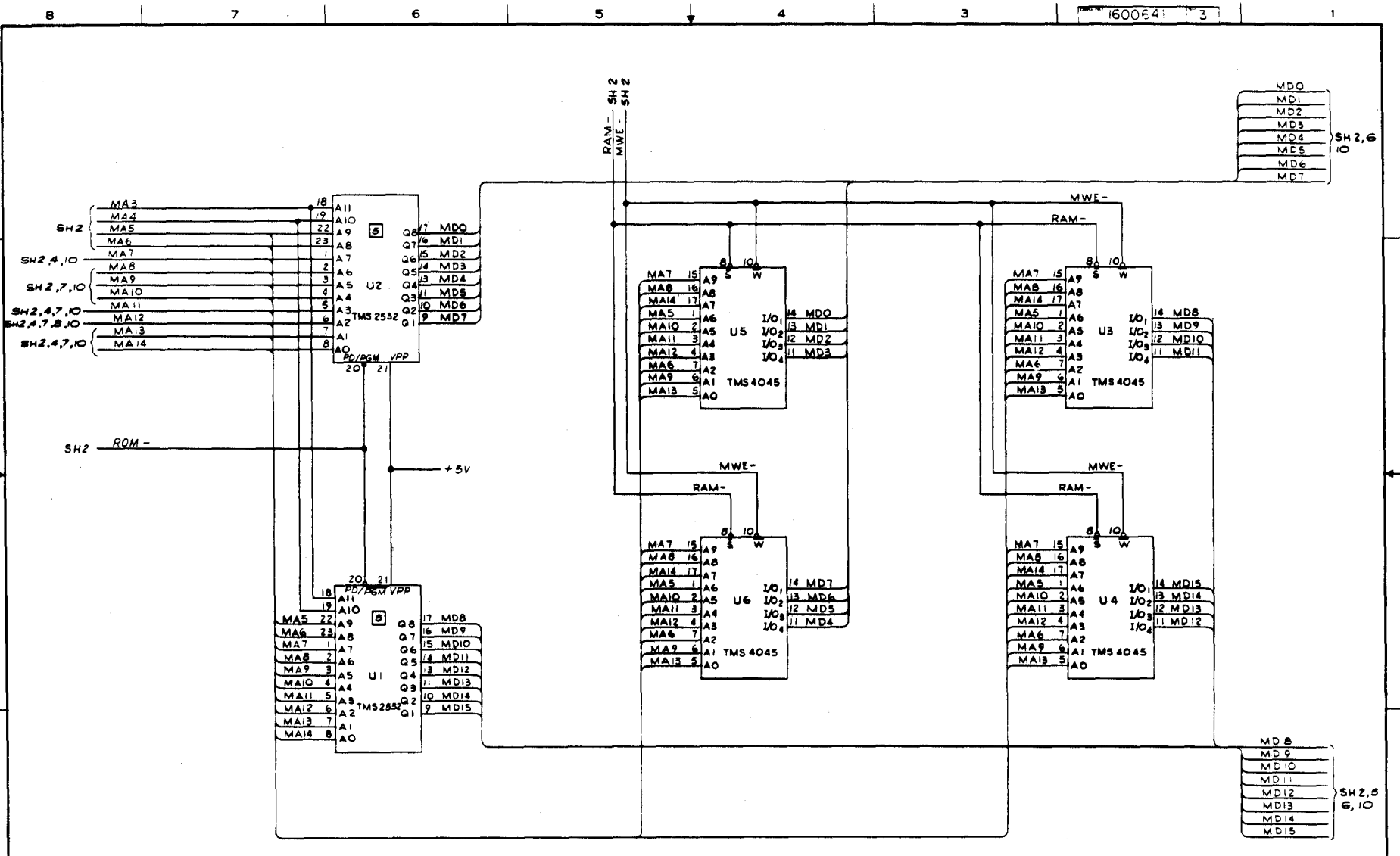
1600642	8117
NEXT ASSY	USED ON
APPLICATION	

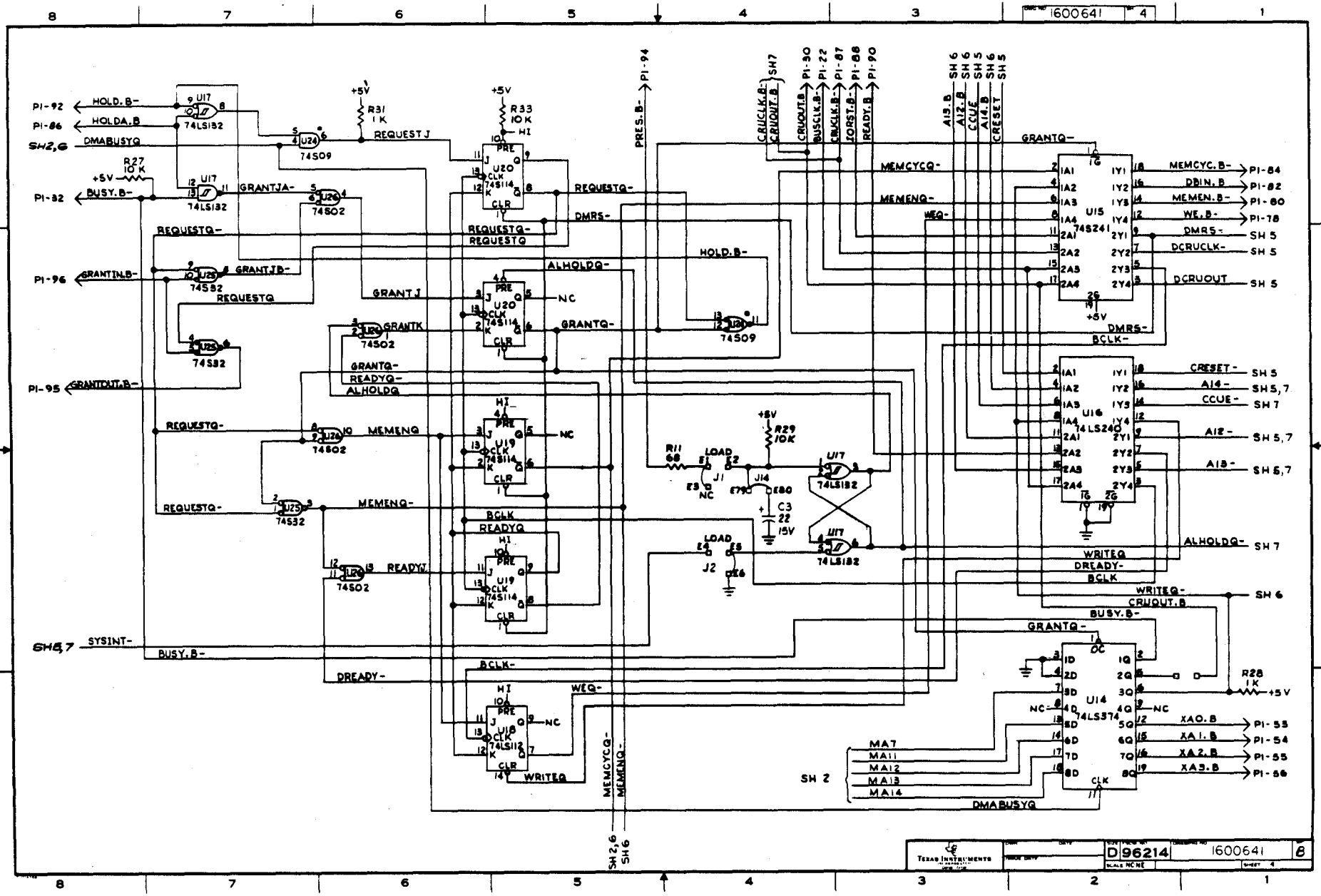
TEXAS INSTRUMENTS

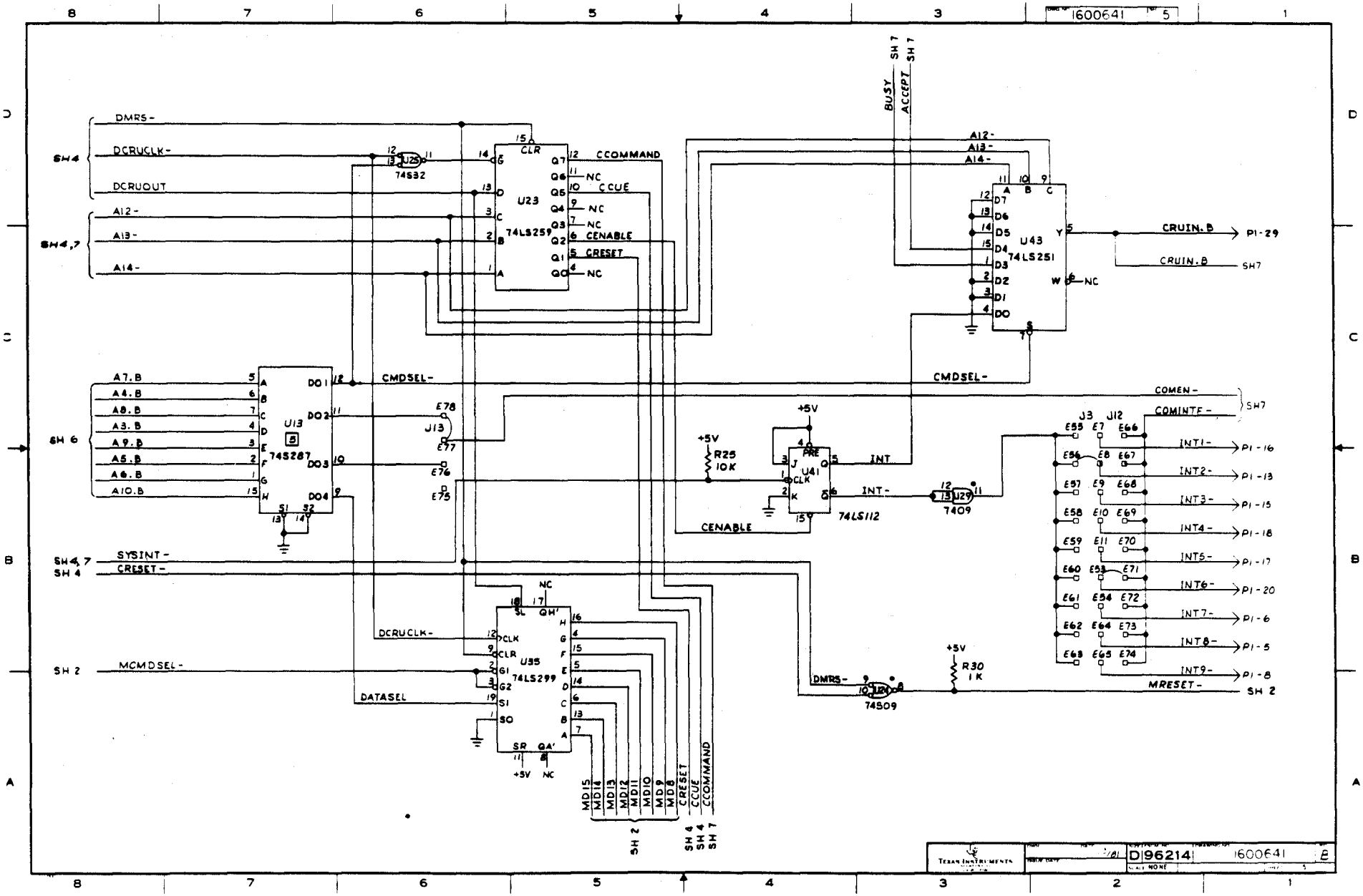
DIAGRAM, LOGIC
TM 990/303B

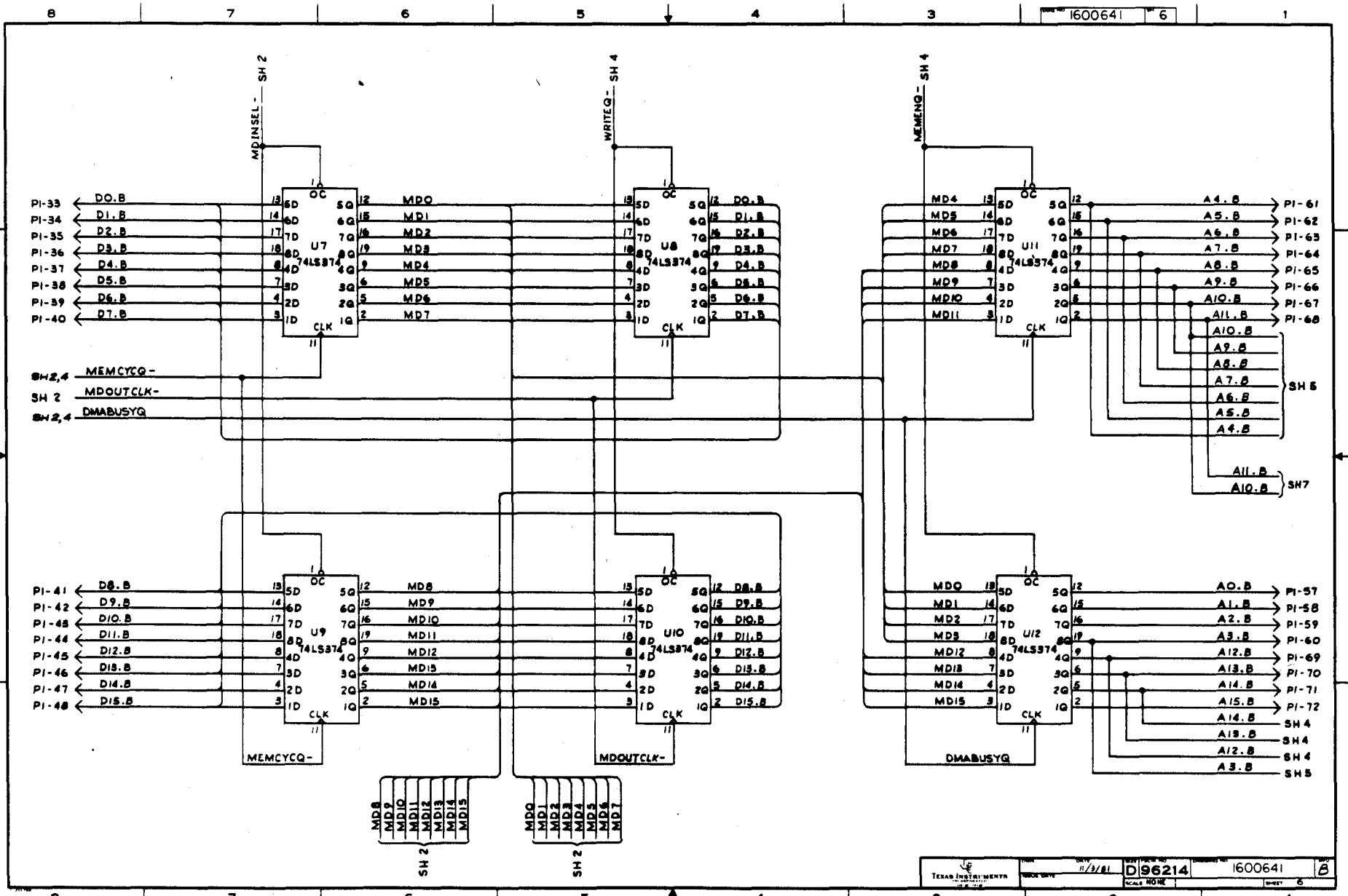
D 96214

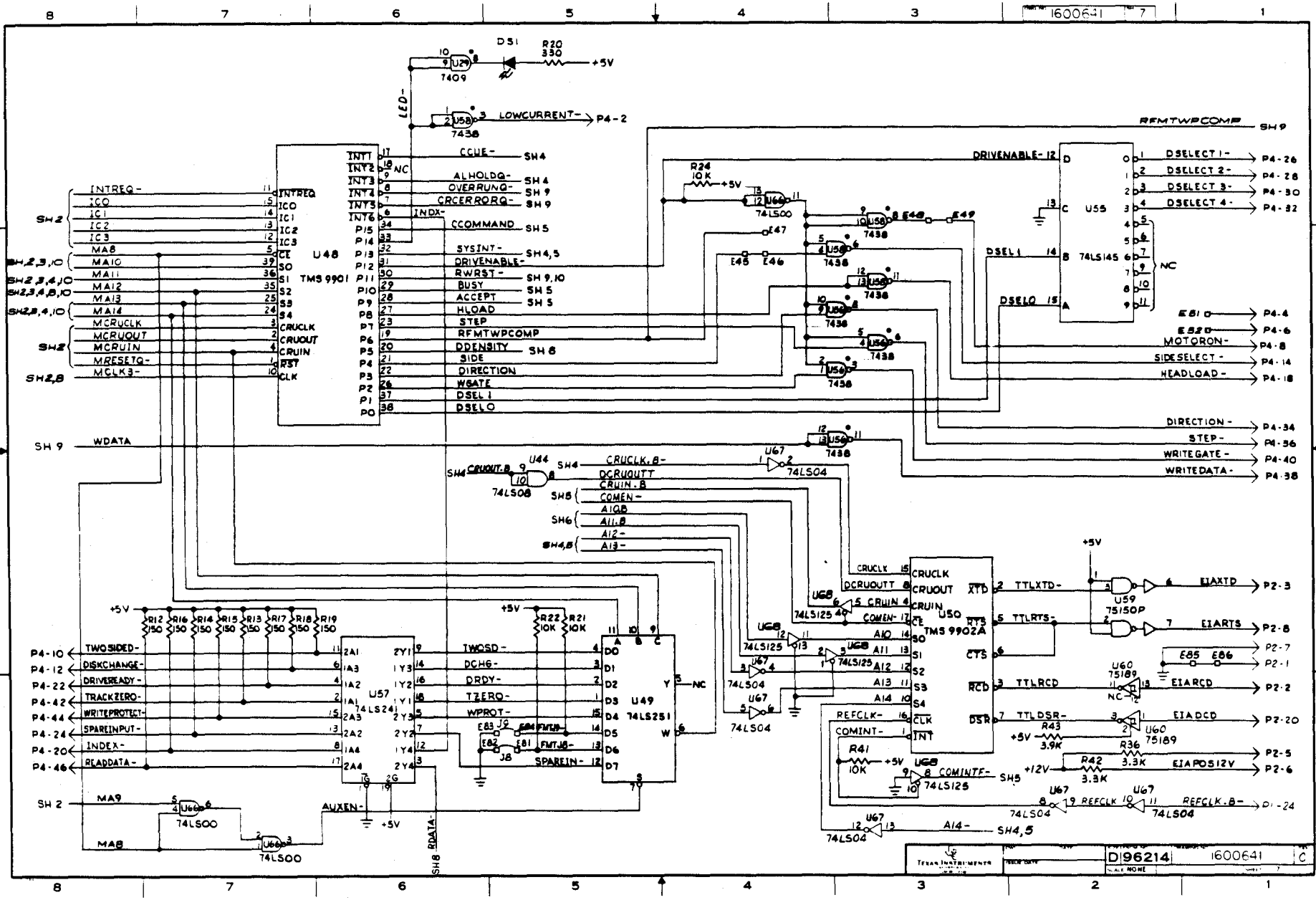

MWE-
MMEMEN-
MDBIN
MREADYQ
INTREQ-
+12V -5V +5V +5V
U32
TMS9900
MEMEN-
WE-
DBIN
READY
WAIT
HOLD-
HOLDA
IAQ
INTREQ-
ICO
IC1
IC2
IC3
MCRUCLK
MCRUOUT
MCRUIN
LOAD-
RESET-
MD0
MD1
MD2
MD3
MD4
MD5
MD6
MD7
MD8
MD9
MD10
MD11
MD12
MD13
MD14
MD15
MA0
MA1
MA2
MA3
MA4
MA5
MA6
MA7
MA8
MA9
MA10
MA11
MA12
MA13
MA14
SH 3, 6, 10
SH 3, 5, 6, 10
SH 3
SH 3, 4, 10
SH 3, 7, 10
SH 3, 4, 7, 10
SH 3, 4, 7, 8, 10
SH 3, 4, 7, 10
R7 15
R8 15
R9 15
R10 15
U31
TIM9904A
MRESET-
FFD
FFQ
MRESETQ-
TANK1
TANK2
XTAL1
XTAL2
OSCIN
L1 3.3 µH
C10 22 pf
Y1 12 MHz
Ø1 TTL
Ø2 TTL
Ø3 TTL
Ø4 TTL
QSCOUT
MCLK1-
MCLK3-
RWBUSYQ
MEMCYCQ-
GND1
VDD
+12V
U42
74S288
DO3
DO5
DO1
DO8
DO7
DO6
DO4
DO2
ADE
ADD
ADC
ADB
ADA
ROM-
MDINSEL-
RWSEL-
RWREAD-
MCMDSEL-
SH 3
SH 6
SH 9
SH 9, 10
SH 5
NC
DMASEL
U30
74LS138
G1
G2A
G2B
Y0
Y1
Y2
Y3
Y4
Y5
Y6
Y7
MDOUTCLK-
RAMWRITE-
RAMREAD-
U29
7409
R26 1K
+5V
RAM-
MADDRCLK-
MREADYQ
INTREQ-
R32 1K
U36
74LS00
U24
74S09
U37
PRE
CLR
74LS74
DMABUSYQ
SH 4, 6
SH 7
SH 8
SH 10

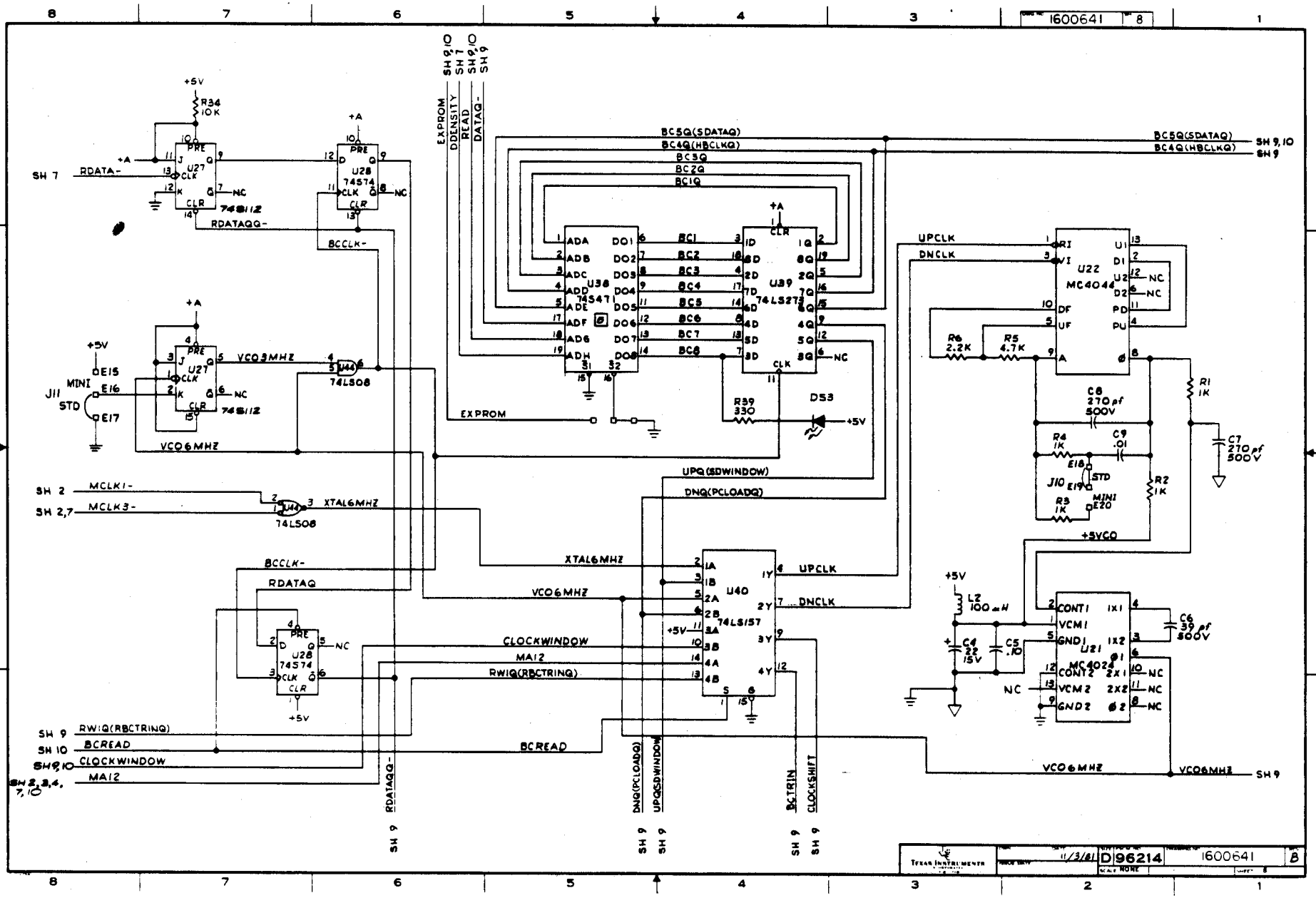

1600641 3
MDO
MD1
MD2
MD3
MD4
MD5
MD6
MD7
SH2,6 IO
SH2
RAM-
MWE-
MA3
MA4
MA5
MA6
MA7
MA8
MA9
MA10
MA11
MA12
MA13
MA14
SH2,4,10
SH2,7,10
SH2,4,7,10
SH2,4,7,8,10
U2
TMS 2532
PD/PGM
VPP
SH2 ROM-
+5V
U5
U6
U3
U4
TMS 4045
MD8
MD9
MD10
MD11
MD12
MD13
MD14
MD15
SH2,5 6,10
D96214
1600641
B


1600641 4
D96214
1600641 B
TEXAS INSTRUMENTS
PI-92 HOLD.B-
PI-86 HOLDA.B
SH2,6 DMABUSYQ
PI-82 BUSY.B-
PI-96 GRANTIN.B-
PI-95 GRANTOUT.B-
SH4,7 SYSINT-
74LS132
74S09
74S02
74S32
74S114
74LS112
74S241
74LS240
74LS374
R27 10K
R31 1K
R33 10K
R29 10K
R28 1K
R11 68
C3 22 15V
REQUESTJ
REQUESTQ-
GRANTJA-
GRANTJB-
GRANTJ
GRANTK
GRANTQ-
NC
HOLD.B-
ALHOLDQ-
READYQ-
MEMENQ-
READYJ
BCLK
BCLK-
DREADY-
WEQ-
WRITEQ
MEMCYCQ-
PRES.B-
CRUCLK.B-
CRUOUT.B
CRUOUT.B
BUSCLK.B-
CRUCLK.B-
IORST.B-
READY.B
A13.B
A12.B
CCUE
A14.B
CRESET
MEMCYC.B-
DBIN.B
MEMEN.B-
WE.B-
DMRS-
DCRUCLK-
DCRUOUT
CRESET-
A14-
CCUE-
A12-
A13-
WRITEQ-
CRUOUT.B
XA0.B
XA1.B
XA2.B
XA3.B
MA7
MA11
MA12
MA13
MA14
SH 2
SH 2,6
SH6
D-5

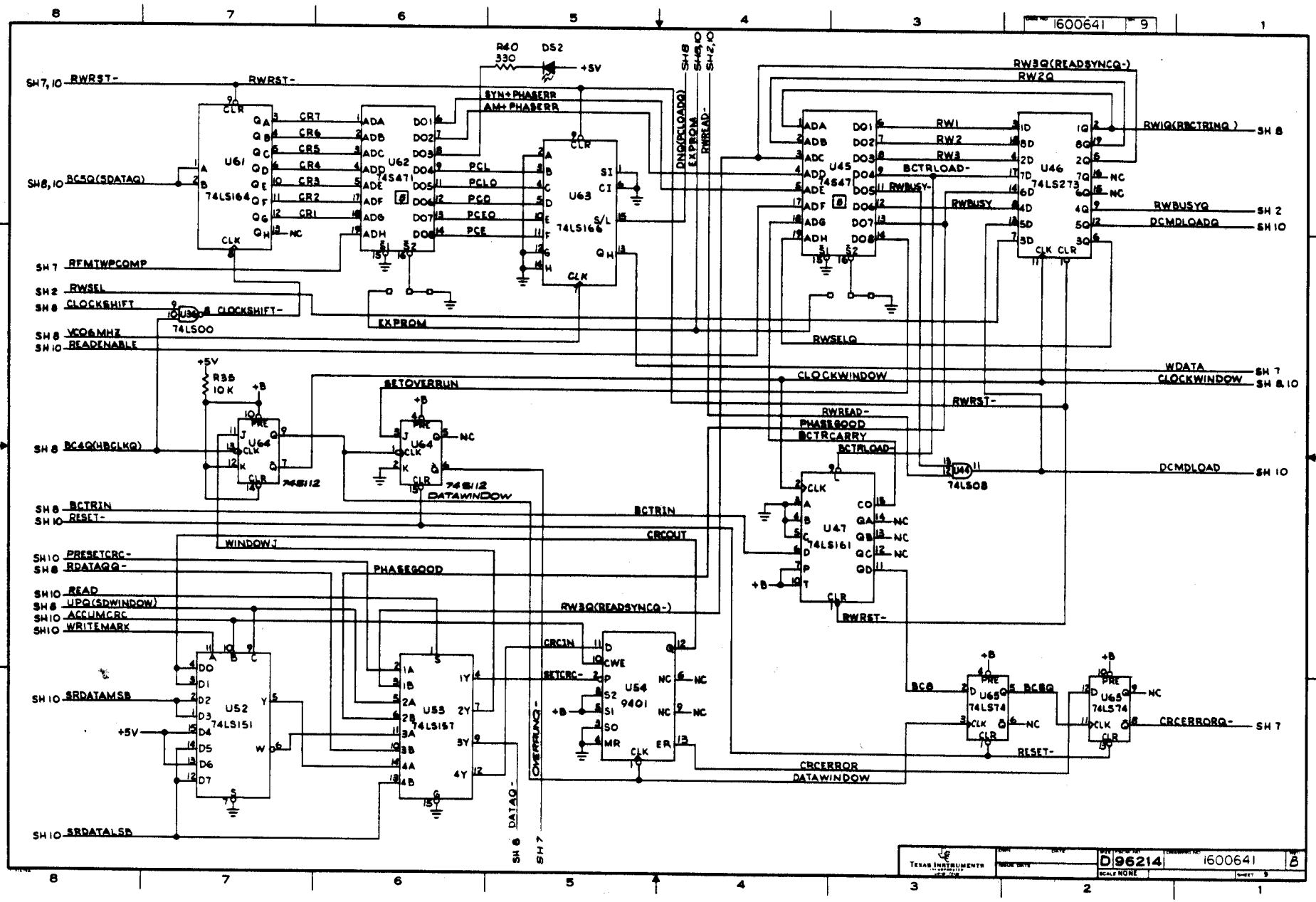

1600641 5
DMRS-
DCRUCLK-
DCRUOUT
A12-
A13-
A14-
SH4
SH4,7
U25
74S32
U23
74LS259
CCOMMAND
NC
CCUE
CENABLE
CRESET
BUSY
ACCEPT
SH 7
U43
74LS251
CRUIN.B
P1-29
CRUIN.B
SH7
A7.B
A4.B
A8.B
A3.B
A9.B
A5.B
A6.B
A10.B
SH 6
U13
74S287
CMDSEL-
E78
J13
E77
E76
E75
+5V
R25
10K
U41
74LS112
INT
INT-
CENABLE
U29
7409
COMEN-
COMINTF-
J3 J12
INT1-
INT2-
INT3-
INT4-
INT5-
INT6-
INT7-
INT8-
INT9-
P1-16
P1-13
P1-15
P1-18
P1-17
P1-20
P1-6
P1-5
P1-8
SYSINT-
CRESET-
SH4,7
SH 4
SH 2
MCMDSEL-
DATASEL
DCRUCLK-
U35
74LS299
MD15
MD14
MD13
MD12
MD11
MD10
MD9
MD8
CRESET
CCUE
CCOMMAND
DMRS-
74S09
R30
1K
MRESET-
SH 2
TEXAS INSTRUMENTS
D96214
1600641
B
SIZE NONE

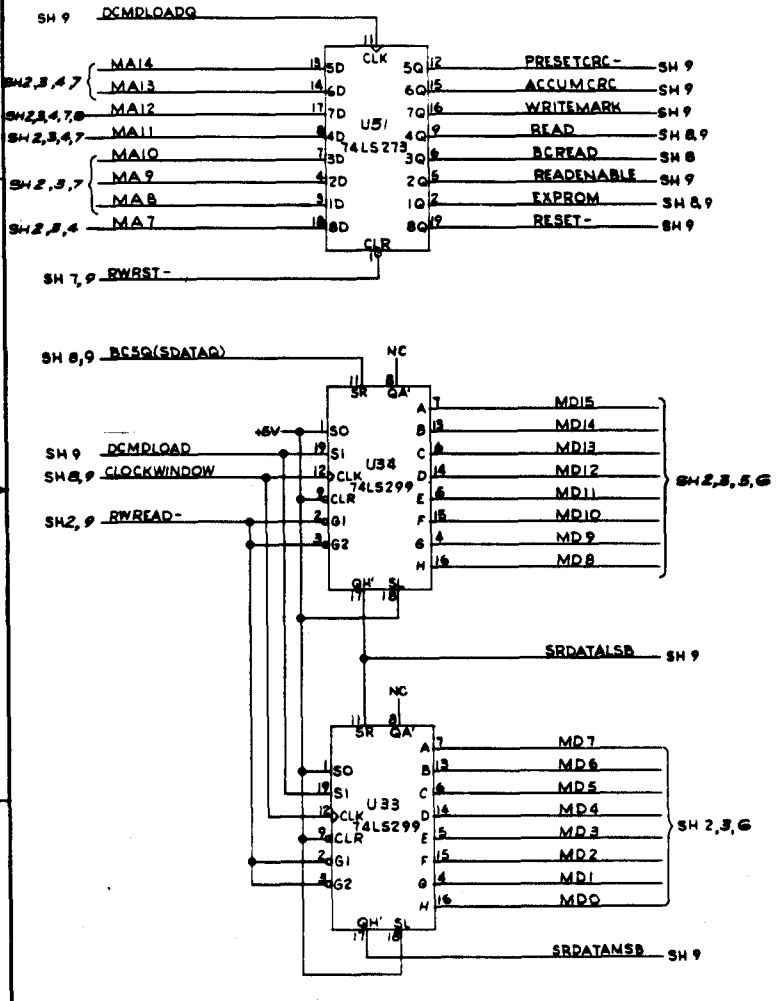

1600641 6
MDINSEL- SH 2
WRITEQ- SH 4
MEMENQ- SH 4
U7
U8
U11
U9
U10
U12
74LS374
OC
CLK
PI-33 DO.B
PI-34 D1.B
PI-35 D2.B
PI-36 D3.B
PI-37 D4.B
PI-38 D5.B
PI-39 D6.B
PI-40 D7.B
MD0
MD1
MD2
MD3
MD4
MD5
MD6
MD7
DO.B
D1.B
D2.B
D3.B
D4.B
D5.B
D6.B
D7.B
A4.B
A5.B
A6.B
A7.B
A8.B
A9.B
A10.B
A11.B
PI-61
PI-62
PI-63
PI-64
PI-65
PI-66
PI-67
PI-68
SH5
SH7
SH2,4 MEMCYCQ-
SH 2 MDOUTCLK-
SH2,4 DMABUSYQ
PI-41 D8.B
PI-42 D9.B
PI-43 D10.B
PI-44 D11.B
PI-45 D12.B
PI-46 D13.B
PI-47 D14.B
PI-48 D15.B
MD8
MD9
MD10
MD11
MD12
MD13
MD14
MD15
D8.B
D9.B
D10.B
D11.B
D12.B
D13.B
D14.B
D15.B
A0.B
A1.B
A2.B
A3.B
A12.B
A13.B
A14.B
A15.B
PI-57
PI-58
PI-59
PI-60
PI-69
PI-70
PI-71
PI-72
SH4
SH5
MEMCYCQ-
MDOUTCLK-
DMABUSYQ
SH 2
SH 2
Texas Instruments
11/3/81
D96214
1600641
B
SCALE NONE
SHEET 6


1600641
DS1
R20 330
+5V
LED-
LOWCURRENT-
P4-2
U48
TMS 9901
INTREQ-
ICO
IC1
IC2
IC3
MA8
MA10
MA11
MA12
MA13
MA14
MCRUCLK
MCRUOUT
MCRUIN
MRESETQ-
MCLK3-
CCUE-
ALHOLDQ-
OVERRUNQ-
CRCERRORQ-
INDX-
CCOMMAND
SYSINT-
DRIVENABLE-
RWRST-
BUSY
ACCEPT
HLOAD
STEP
RFMTWPCOMP
DDENSITY
SIDE
DIRECTION
WGATE
DSEL1
DSEL0
RFMTWPCOMP
DRIVENABLE-
U55
74LS145
DSELECT1-
DSELECT2-
DSELECT3-
DSELECT4-
MOTORON-
SIDESELECT-
HEADLOAD-
WDATA
DIRECTION-
STEP-
WRITEGATE-
WRITEDATA-
U44
74LS08
CRUCLK.B-
DCRUOUTT
CRUIN.B
COMEN-
AIQB
AI1.B
AI2-
AI3-
U57
74LS241
U49
74LS251
TWOSIDED-
DISKCHANGE-
DRIVEREADY-
TRACKZERO-
WRITEPROTECT-
SPAREINPUT-
INDEX-
READDATA-
TWOSD-
DCHG-
DRDY-
TZERO-
WPROT-
SPAREIN-
FMT.J6-
J8
U50
TMS 9902A
CRUCLK
DCRUOUTT
CRUIN
COMEN-
AIQ
AI1
AI2
AI3
AI4
REFCLK-
COMINT-
XTD
RTS
CTS
RCD
DSR
INT
TTLXTD-
TTLRTS-
TTLRCD
TTLDSR-
U59
75150P
U60
75189
EIAXTD
EIARTS
EIARCD
EIADCD
EIAPOS12V
COMINTF-
REFCLK.B-
AI4-
MA9
MA8
AUXEN-
RDATA-


1600641
8
+5V
R34
10K
+A
RDATA-
SH 7
U27
U28
74S74
74S112
NC
RDATAQQ-
BCCLK-
VCO3MHZ
VCO6MHZ
74LS08
J11
MINI
STD
E15
E16
E17
SH 2
MCLK1-
SH 2,7
MCLK3-
XTAL6MHZ
RDATAQ
CLOCKWINDOW
MAI2
RWIQ(RBCTRINQ)
BCREAD
SH 9
SH 10
SH9,10
SH 2,3,4, 7, 10
SH 9,10
SH 7
SH 9
EXPROM
DDENSITY
READ
DATAQ-
BC5Q(SDATAQ)
BC4Q(HBCLKQ)
BC3Q
BC2Q
BC1Q
U38
74S471
U39
74LS273
R39
330
DS3
UPQ(SDWINDOW)
DNQ(PCLOADQ)
U40
74LS157
UPCLK
DNCLK
U22
MC4044
R6
2.2K
R5
4.7K
R1
1K
C8
270 pf
500V
R4
1K
C9
.01
R2
1K
R3
1K
J10
C7
270 pf
500V
+5VCO
L2
100 uH
C4
22
15V
C5
.10
U21
MC4024
C6
39 pf
500V
RDATAQQ-
DNQ(PCLOADQ)
UPQ(SDWINDOW)
BCTRIN
CLOCKSHIFT
VCO6MHZ
SH 9
11/3/81
D96214
1600641
B
NONE
TEXAS INSTRUMENTS

D-9


1600641 9
R40 330
DS2
+5V
SH7,10 RWRST-
RWRST-
SH8,10 BC5Q(SDATAQ)
U61
74LS164
CR7
CR6
CR5
CR4
CR3
CR2
CR1
U62
74S471
PCL
PCLQ
PCQ
PCEQ
PCE
SYN+PHASERR
AM+PHASERR
U63
74LS166
SH7 RFMTWPCOMP
SH2 RWSEL
SH8 CLOCKSHIFT
U36
74LS00
CLOCKSHIFT-
SH8 VCO&MHZ
SH10 READENABLE
EXPROM
DNQ(RELOADQ)
EXPROM
RWREAD-
SH8
SH5,10
SH2,10
RW3Q(READSYNCQ-)
RW2Q
U45
74S472
RW1
RW2
RW3
BCTRLOAD-
RWBUSY-
RWBUSY
U46
74LS273
RW1Q(RBCTRINQ) SH 8
RWBUSYQ SH 2
DCMDLOADQ SH 10
RWSELQ
+5V
R35
10K
+B
U64
74S112
SETOVERRUN
DATAWINDOW
SH8 BC4Q(HBCLKQ)
SH8 BCTRIN
SH10 RESET-
BCTRIN
CRCOUT
WINDOWJ
PHASEGOOD
SH10 PRESETCRC-
SH8 RDATAQQ-
SH10 READ
SH8 UPQ(SDWINDOW)
SH10 ACCUMCRC
SH10 WRITEMARK
RWREAD-
PHASEGOOD
BCTRCARRY
BCTRLOAD-
CLOCKWINDOW
WDATA SH 7
CLOCKWINDOW SH 8,10
RWRST-
U44
74LS08
DCMDLOAD SH 10
U47
74LS161
RWRST-
U52
74LS151
U53
74LS157
SH10 SRDATAMSB
SH10 SRDATALSB
+5V
CRC1N
RESETC-
U54
9401
CRCERROR
DATAWINDOW
OVERRUNQ-
SH6 DATAQ-
SH7
BCQ
U65
74LS74
BCRQ
CRCERRORQ- SH 7
RESET-
TEXAS INSTRUMENTS
D96214
1600641
SCALE NONE
SHEET 9
8
A
D-10



D-11

TEXAS INSTRUMENTS	DATE 11/3/81	SIZE D	96214	1600641	REV B
		SCALE NONE			SHEET 10

APPENDIX E

PROGRAMMING PROM FOR UNIQUE CRU ADDRESS

E.1 GENERAL

The 74S287 PROM at U13 monitors address lines A3 through A10 in order to specify CRU addresses for:

- transferring Command List memory location to TM 990/303B,
- issuing commands to the TM 990/303B and receiving status of the command transfer.
- the two jumper-selectable addresses for the EIA port (port 2).

These CRU addresses are selected using a programmed 74S287 at U13. The address-line inputs are the eight most-significant bits of the CRU address (A3 to A10). This address is placed by the programmer into register 12 and placed on the address bus during a CRU instruction. These address lines are decoded by the PROM which, in turn, places a four-bit data byte, selected by the address, on its DO1 to DO4 output bits to enable circuitry as follows:

- DO1 a zero and CRU write instruction: Write command for address transfer to TM 990/303B. As-shipped CRU software base address = 220_{16} or 230_{16}.

  DO1 a zero and CRU read instruction: Read status of address transfer at TM 990/303B. As-shipped CRU software base address = 220_{16} or 230_{16}.

- DO2 a zero: Enable port P2 if jumper J13 is set E78-E77. As-shipped CRU software base address = 0180_{16}.

- DO3 a zero: Enable port P2 if jumper J13 is set E76-E77. As-shipped CRU software base address = 0300_{16}.

- DO4 a one: Transfer the Command List address to the TM 990/303B. As-shipped CRU software base address = 200_{16} or 210_{16}.

Address lines and data output lines are shown in Figure E-1. The mode is selected by address-line inputs and PROM memory outputs which set the values on mode-select lines. Because mode selection is caused by PROM address contents, you can reprogram the PROM to obtain a custom CRU address scheme.

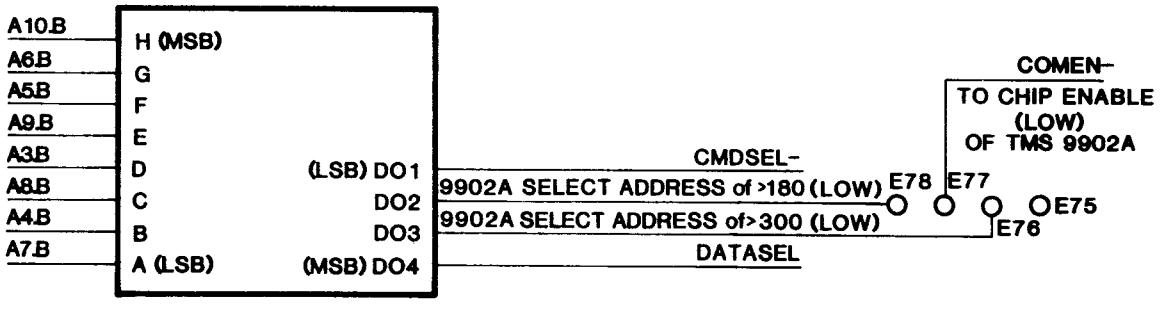


FIGURE E-1. PROM U13 ADDRESS-INPUT PINS AND DATA-OUTPUT PINS

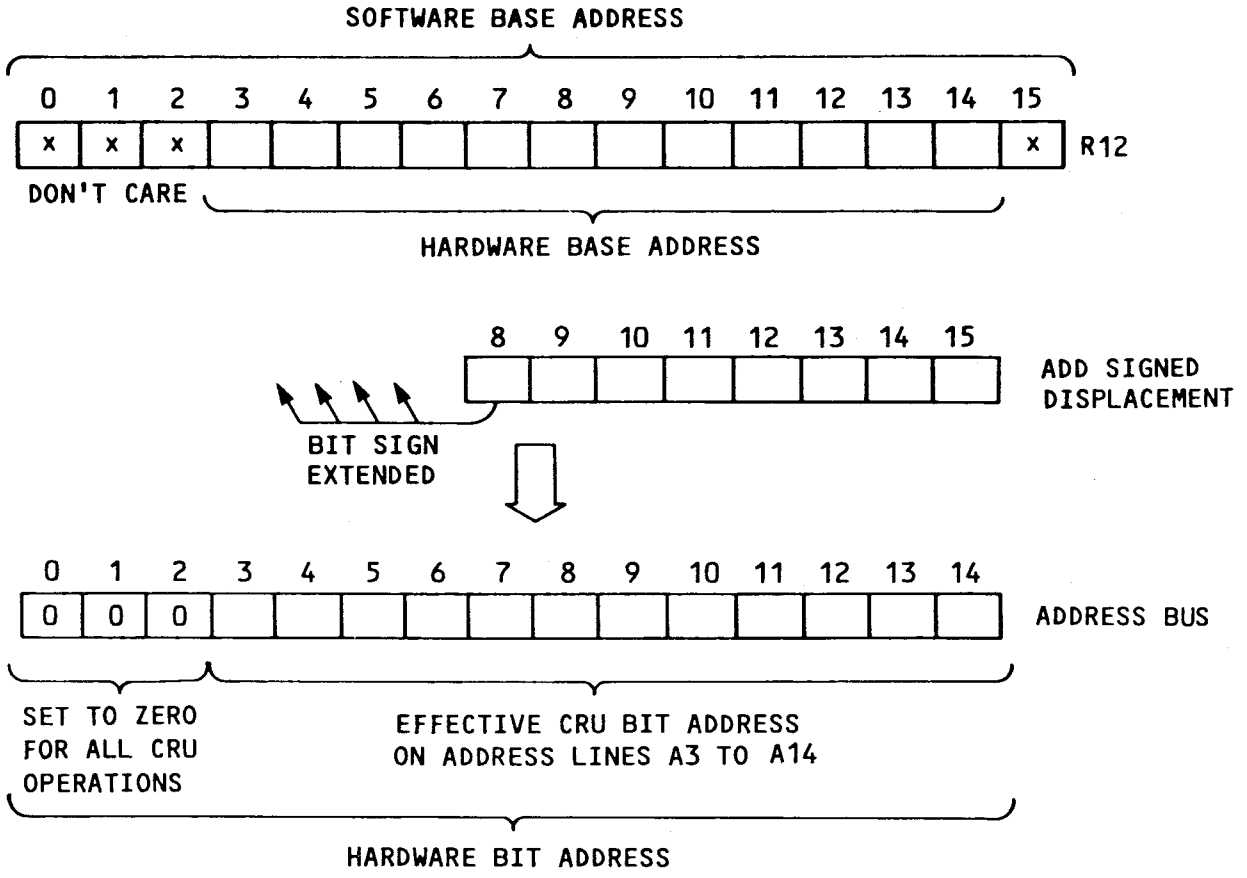


FIGURE E-2. CRU ADDRESS NOMENCLATURE

This writeup presumes a background in CRU addressing and CRU nomenclature. CRU address nomenclature is depicted in Figure E-2, which shows how the CRU bit address is derived from the software base address and the displacement of the executing CRU instruction. A detailed explanation of the five CRU addresses is provided in your CPU manual.

E.2 COMMAND COMMUNICATION VIA THE CRU

Two different CRU base address blocks are used to pass command addresses:

- one eight-bit block to transfer the three-byte address where the Command List resides (hardware base address 0108_{16} or software base address 0210_{16}) and
- one block to access handshaking signals for Command List transfer (hardware base address 0110_{16} or software base address 0220_{16}).

These addresses are the result of the PROM monitoring address lines A3 to A10 and thus outputting an enabling/disabling code on D01 or D04, which are the command transfer signal (CMDSEL-) or the data transfer signal (DATASEL) as shown in Figure E-1.

Note in Figure E-3 that the first eight CRU bits (at hardware base address 0100_{16}) are for the address (data) bytes to be transferred serially to the disk controller. These are repeated on the next eight CRU bits (hardware base address 0108_{16}). This is the same for the next group of eight CRU bits which contain the command data for the address-byte transfer -- the first eight bits (hardware base address 0110_{16}) are repeated eight bits later.

To Disk Controller	CRU Bit (Hardware Base Addresses in parentheses)	From Disk Controller	Displacement From Hardware Base Address
LSB	Base + 0 (100)	0	
↑	1	0	
Command	2	0	
List	3	0	
Address	4	0	
↓	5	0	
	6	0	
MSB	7	0	
LSB	Base + 8 (108)	0	0
↑	9	0	1
Command	A	0	2
List	B	0	3
Address	C	0	4
↓	D	0	5
	E	0	6
MSB	F	0	7
COMMAND	Base + 10 (110)	0	8
0	11	0	9
CUE	12	0	10
0	13	ACCEPT	11
0	14	BUSY	12
INTERRUPT ENABLE	15	0	13
RESET	16	0	14
0	Base + 17 (117)	INTERRUPT ISSUED	15
COMMAND	18 (118)	0	
0	19	0	
CUE	1A	0	
0	1B	ACCEPT	
0	1C	BUSY	
INTERRUPT ENABLE	1D	0	
RESET	1E	0	
0	Base + 1F (11F)	INTERRUPT ISSUED	

FIGURE E-3. CRU ADDRESS SCHEME FOR TRANSFERRING COMMAND LIST ADDRESS AS SHIPPED FROM FACTORY

The reason for this repetition is that address line A11 is not monitored in this address scheme. Only address lines A3 to A10 are decoded by the PROM at U13; thus, the value on A11 can be a one or zero and not change the CRU address. If the value on A3 to A10 remains the same while A11 is toggled, the PROM address remains the same, and the PROM output remains the same. As shown in Figure E-3, the 16 bits in the center of the 32-bit scheme are contiguous. If this center block is addressed (CRU software base address of 210_{16}), bytes can be transferred by an LDCR while handshaking can be via single-bit CRU instructions without changing the base address in R12.

PROM output summary for transfer of Command List address:

```
DO1 = 0
DO2 = 1   Send/monitor handshaking signals for address transfer
DO3 = 1
DO4 = 1

DO1 = 1
DO2 = 1   Transfer Command List address bytes
DO3 = 1
DO4 = 1
```

As has been stated, PROM output corresponds to its programmed values.

E.3 EIA PORT ADDRESS

The CRU software base address for the EIA port is chosen in a similar manner. Jumper J13 connects the U13 PROM output at either DO2 or DO3 to chip enable (low) of the TMS 9902A communication controller. A CRU software base address of 180_{16} causes a zero output at PROM output DO2 which is routed to the TMS 9902A by jumper J13 set E77-E78. Likewise, a CRU base address of 300_{16} causes a zero output at DO3 which is routed to the TMS 9902A with jumper J13 set E76-E77. PROM output summary for CRU base address:

```
DO1 = 1                                  DO1 = 1
DO2 = 0   Selects Address 180(16)        DO2 = 1   Selects Address 300(16)
DO3 = 1                                  DO3 = 0
DO4 = 0                                  DO4 = 0
```

E.4 OTHER ADDRESSES

For addresses other than those in E.2 and E.3, the PROM is programmed so that its output will not enable either address transfer or TMS 9902A circuitry:

```
DO1 = 1
DO2 = 1
DO3 = 1
DO4 = 0
```

Thus, the PROM contents determine function according to address-line inputs.

E.5 CONSIDERATIONS

The arrangement of bus address lines to PROM address inputs is not straightforward; that is, the most significant bus address line A3.B is not connected to the most significant PROM address input line H. Because of this, the user must take care in interpreting address line values corresponding to the internal PROM address. Figure E-4 shows the relationship of address line to PROM address inputs.

The software base address must be a value between 0020_{16} (hardware base address 0010_{16}) and $1FE0_{16}$ (hardware base address $0FF0_{16}$).

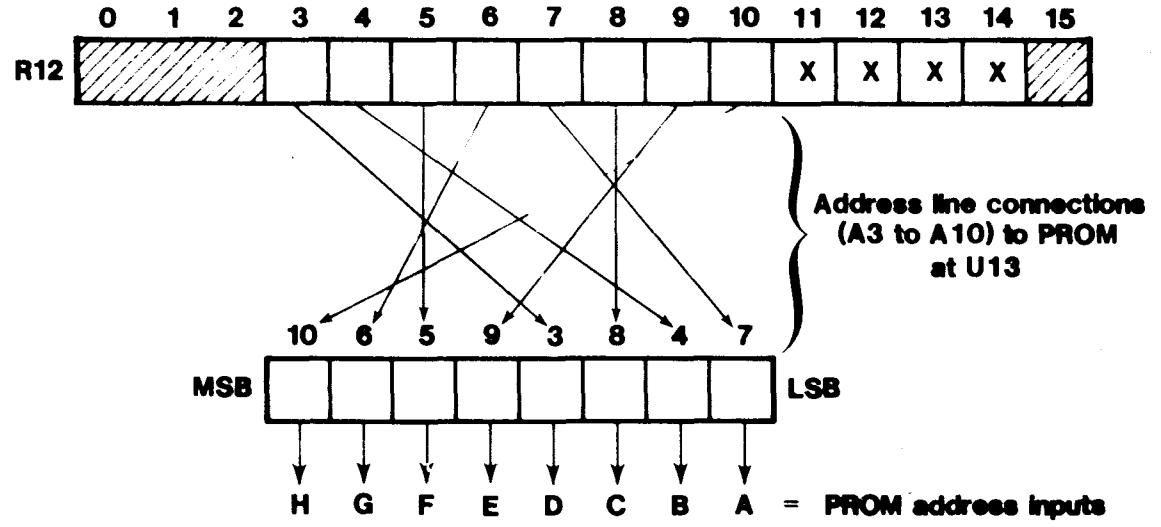


FIGURE E-4. INTERPRETING HARDWARE BASE ADDRESS AS ADDRESS INPUT TO PROM AT U13

For port P2 CRU software base addresses, address line A10 must be a zero. This is because address lines A10 to A14 select what part of the 32-bit address space within the TMS 9902A is to be addressed. Thus, A10 a zero addresses the bottom 16 address bits while A10 a one addresses the higher 16 address bits. It is the 32-bit address space offset from the CRU base address contained in R12 that requires the PROM to handle address line A10 being a 0 or a 1, even though address line A10 contained in R12 must be a zero. A more detailed look at the TMS 9902A is given in Section 5.

Because address line A11.B is not decoded in Command List transfer, the user selects a CRU area consisting of 16 CRU bits for each address input, even though only eight bits are needed. Thus a total CRU address space of 32 CRU bits is needed for both the data transfer and the command transfer (each at their own CRU address). As shipped, the data (address) transfer uses CRU software base address 0200_{16} and command transfer uses CRU software base address 0220_{16}. However, for convenience, software base addresses 0210_{16} (for data) and 0220_{16} (for command) can be used to make one contiguous CRU address space. With 0210_{16} in register 12, an LDCR instruction can load the Command List address while single-bit instructions can set and monitor control bits (all at CRU software base address 0210_{16}).

E.6 HOW TO DETERMINE THE PROM CONTENTS

Use the following steps to determine to PROM addresses and their respective contents:

1) Compute the desired CRU hardware base address (in register 12, this is bits 3 to 14) or CRU software base address (all 16 bits).

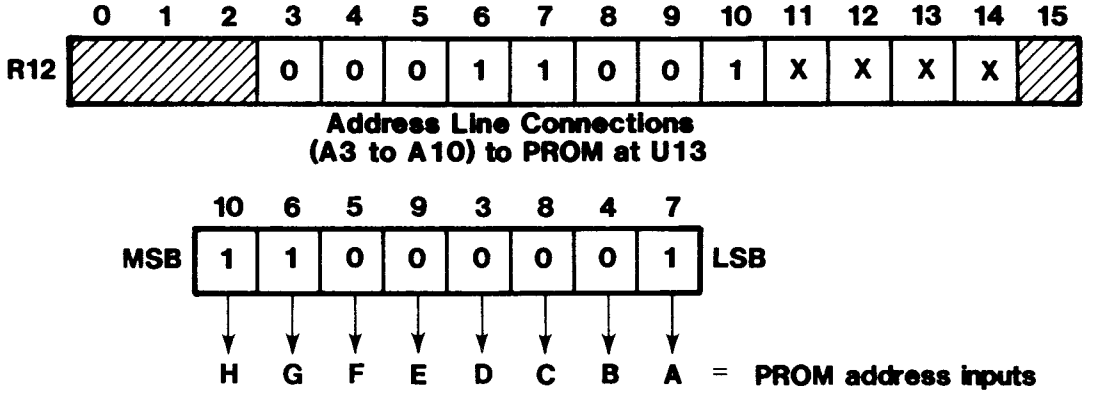
2) Insert the hardware base address into the 12 blanks of R12 as shown at the top of Figure E-4. Note that only the values in R12 bits 3 to 10 are relevent since only these will be decoded by the PROM address input lines. Do not enter values for R12 bits 0-3 or 11-15.

3) Following the lines between R12 and the PROM, copy the same binary values into the PROM address input boxes. The resulting eight-bit value will be the PROM address at which to program the respective nibble contents, depending upon desired function, as explained in the tables in sections E.2, E.3, and E.4.

E.7 EXAMPLE 1

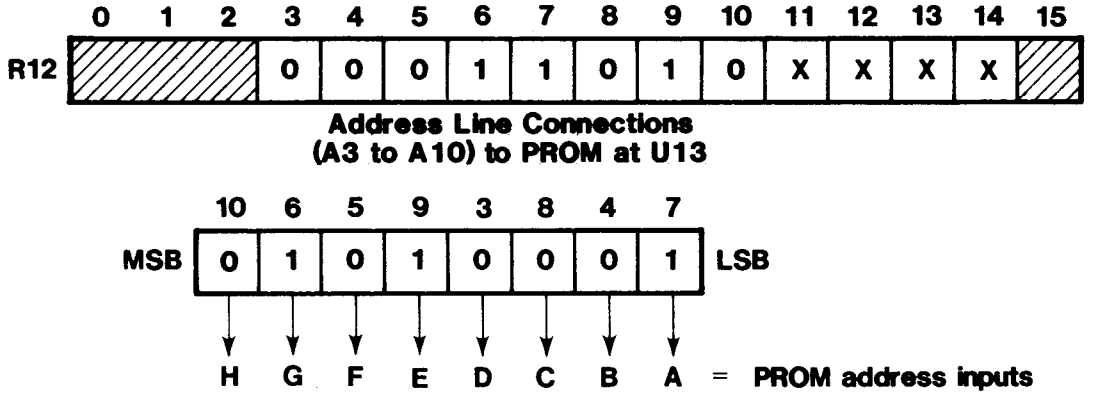
- Desired software base address for data transfer: 0320_{16}
- Desired software base address for command transfer: 0340_{16}

The results are shown in Figure E-5.



Result: Program PROM address $C1_{16}$ with 1111_2.

(a) Determine Data Transfer PROM Nibble Contents (Sftwr Base Addr 0320_{16})



Result: Program PROM address 51_{16} with 0110_2.

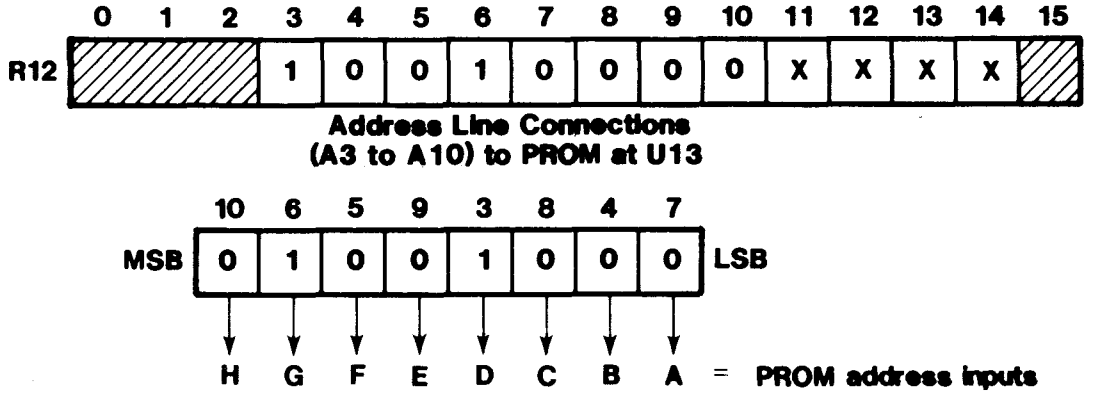
(b) Determine Command Transfer PROM Nibble Contents (Sftwr Base Addr 0340_{16})

FIGURE E-5. EXAMPLE 1 RESULTS

E.8 EXAMPLE 2

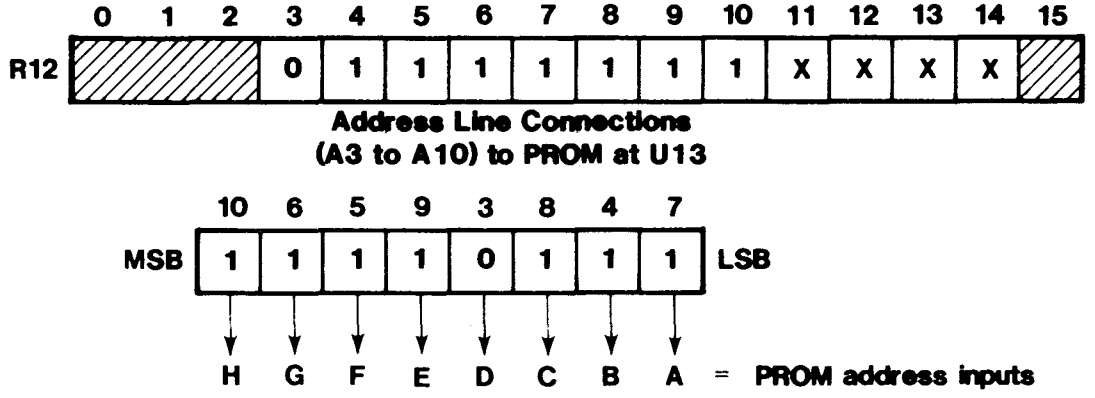
- Desired software base address for data transfer: 1200_{16}
- Desired software base address for command transfer: $0FE0_{16}$

The results are shown in Figure E-6.



Result: Program PROM address 48_{16} with 1111_2.

(a) Determine Data Transfer Nibble Contents (Sftwr Base Addr 1200_{16})



Result: Program PROM address $F7_{16}$ with 0110_2.

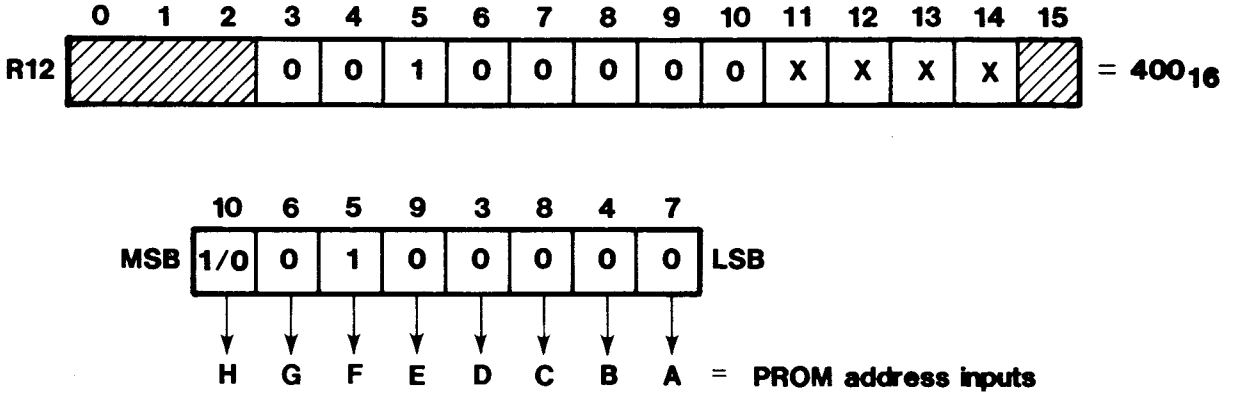
(b) Determine Command Transfer PROM Nibble Contents (Sftwr Base Addr $0FE0_{16}$)

FIGURE E-6. EXAMPLE 2 RESULTS

E.9 EXAMPLE 3

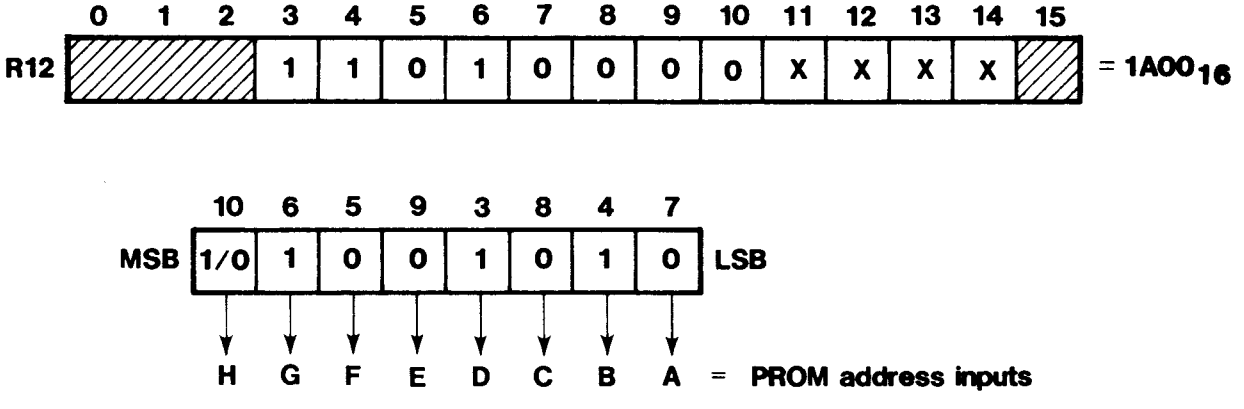
- Desired software base address for port P2 enabled via D02 (thus jumper 13 position E78-E77): 0400_{16}
- Desired software base address for port P2 enabled via D03 (thus jumper 13 position E76-E77): $1A00_{16}$

The results are shown in Figure E-7.



Result: Program PROM address 20_{16} and $A0_{16}$ with 0101_2.

(a) Determine EIA Port PROM Nibble Contents for D02 Enable (Sftwr Base Addr 400_{16})



Result: Program PROM address $4A_{16}$ and CA_{16} with 0011_2.

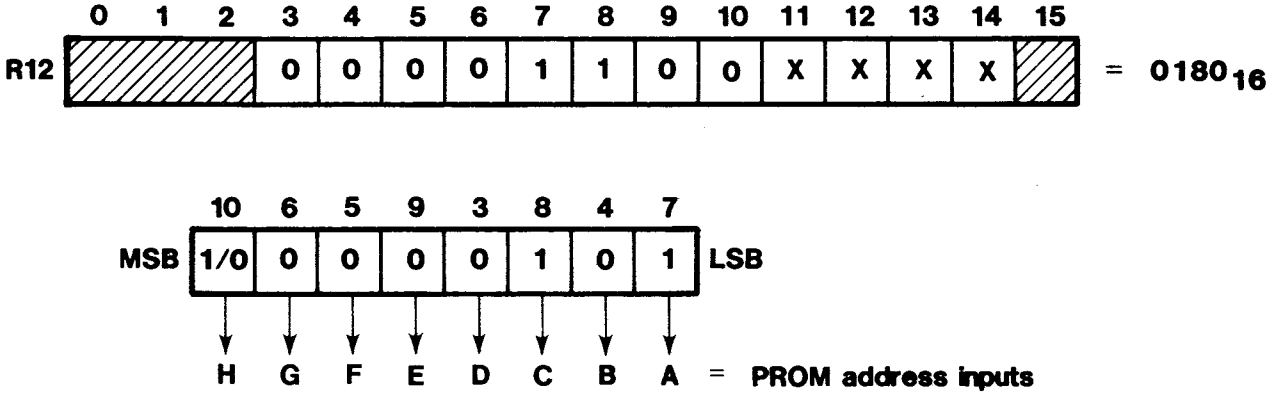
(b) Determine EIA Port PROM Nibble Contents for Port Enable from D03 (Sftwr Base Addr $1A00_{16}$)

FIGURE E-7. EXAMPLE 3 RESULTS

E.10 EXAMPLE 4

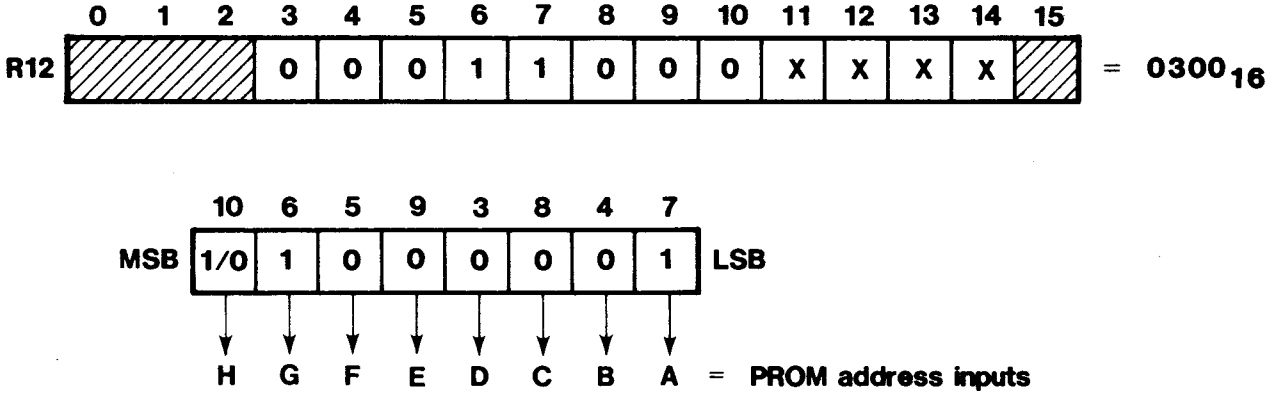
- Desired software base address for port P2 enabled via D02 (thus jumper 13 position E78-E77): 018_{16}
- Desired software base address for port P2 enabled via D03 (thus jumper 13 position E76-E77): 030_{16}

The results are shown in Figure E-8.



Result: Program PROM address 05_{16} and 85_{16} with 0101_2.

(a) Determine EIA Port PROM Nibble Contents for D02 Enable (Sftwr Base Addr 0180_{16})



Result: Program PROM address 41_{16} and $C1_{16}$ with 0011_2.

(b) Determine EIA Port PROM Nibble Contents for Port Enable from D03 (Sftwr Base Addr 0300_{16})

FIGURE E-8. EXAMPLE 4 RESULTS

APPENDIX F

PIN LIST FOR CONTROLLER-TO-DRIVE CABLES

TABLE F-1. PIN LIST FOR TM 990/527 CABLE FOR MODEL 800 DRIVE

Pin at Connector P4	Pin at Disk Drive	Signal
P4-2	2	LOW CURRENT (Not used by Shugart models)
P4-4	4	E51 (unpopulated jumper pin)
P4-6	6	E52 (unpopulated jumper pin)
P4-8	8	MOTORON-
P4-10	10	TWOSIDED-
P4-12	12	DISKCHANGE-
P4-14	14	SIDESELECT-
P4-18	18	HEADLOAD-
P4-20	20	INDEX-
P4-22	22	DRIVEREADY-
P4-24	24	SPAREINPUT-
P4-26	26	DSELECT1-
P4-28	28	DSELECT2-
P4-30	30	DSELECT3-
P4-32	32	DSELECT4-
P4-34	34	DIRECTION-
P4-36	36	STEP-
P4-38	38	WRITEDATA-
P4-40	40	WRITEGATE-
P4-42	42	TRACKZERO-
P4-44	44	WRITEPROTECT-
P4-46	46	READDATA-

NOTES

1. All odd-numbered lines are tied to ground. Odd-numbered pins at connector P4 are on the bottom side of the PC board.

2. Note that the dash number at connector P4 is the pin number at the disk drive edge connector.

3. Jumper pins E50, E51, and E52 are <u>not</u> provided on the board as shipped; instead plated through holes are connected to the pins at connector P4. These are provided for future use.

4. Pin 2 at connector P4 is connected to the color-coded ribbon-cable edge.

TABLE F-2. PIN LIST FOR TM 990/535B CABLE FOR MODEL 400 DRIVE

50-Pin[1] Connector P4	34-Pin[2] Output at P5	Signal
P4-8	P5-16	MOTORON-
P4-14	P5-32	SIDE SELECT-
P4-20	P5-8	INDEX-
P4-26	P5-10	DSELECT1-
P4-28	P5-12	DSELECT2-
P4-30	P5-14	DSELECT3-
P4-32	P5-6	DSELECT4-
P4-34	P5-18	DIRECTION-
P4-36	P5-20	STEP-
P4-38	P5-22	WRITEDATA-
P4-40	P5-24	WRITEGATE-
P4-42	P5-26	TRACKZERO-
P4-44	P5-28	WRITEPROTECT-
P4-46	P5-30	READDATA-

NOTES

1. PC board 1600135-0001 is connected to the 50-pin connector at P4 to provide cross-over wiring to a 34-pin output compatible with the 34-pin connector on the mini disk drive. This interface board is included as part of the 34-wire cable, TM 990/535B. P5 is the 34-pin output of this interface board.

2. The dash number at P5 is the corresponding pin number at the disk drive connector. P5 is the 34-pin output of this interface board.

3. Odd-numbered pins are connected to ground only. Those odd-numbered ground pins in the cable are P4-7 (P5-1), P4-11 (P5-3), P4-19 (P5-5), P4-21 (P5-7), and the odd numbers from P4-25 to P4-45 (P5-9 to P5-29) inclusive.

4. Pin 2 at connector P4 is connected to the color-coded ribbon-cable edge.

APPENDIX G

PARTS LIST

Symbol	Description	Qty
C1, C2	Capacitor, 68.0 uFd, 15 V, 10%	2
C3, C4	Capacitor, 22.0 uFd, 15 V, 10%	2
C5	Capacitor, 0.10 uFd, 50 V, 5%, ceramic	1
C6	Capacitor, 39.0 pFd, 500 V, 5%, mica	1
C7, C8	Capacitor, 270.0 pFd, 500 V, 5%, mica	2
C9	Capacitor, 0.010 uFd, 100 V, 10%, ceramic	1
C10	Capacitor, 22.0 pFd, 200 V, 10%, ceramic	1
C11-C26, C28-C53, C55, C57-C64	Capacitor, 0.047 uFd, 25 V, +80%/-20%, axial lead	51
CR1	Diode, silicon zener, 1%, 5V (E7918)	1
DS1-DS3	LED, Rt angle, Red	3
L1	Coil, RF, 3.3 uH	1
L2	Coil, RF, 100 uH, 4.5 ohm, 133 mA	
R1-R4, R26, R28, R30-R32	Resistor, 1.0 kilohm, 5%, 0.25 W	9
R5	Resistor, 4.7 kilohm, 5%, 0.25 W	1
R6	Resistor, 2.2 kilohm, 5%, 0.25 W	1
R7-R10	Resistor, 15 ohm, 5%, 0.25 W	4
R11	Resistor, 68 ohm, 5%, 0.25 W	1
R12-R19	Resistor, 150 ohm, 5%, 0.25 W	8
R20, R37-R40	Resistor, 330 ohm, 5%, 0.25 W	5
R21, R22, R24, R25, R27, R29, R33, R34, R35, R41	Resistor, 10 kilohm, 5%, 0.25 W	10
R36, R42	Resistor, 3.3 kilohm, 5%, 0.25 W	2
R43	Resistor, 3.9 kilohm, 5%, 0.25 W	1
U1	EPROM, TMS 2532, LSB	1
U2	EPROM, TMS 2532, MSB	1
U3-U6	Static RAM, TMS 2114	4
U7-U12, U14	IC, SN74LS374N	7
U13	PROM, CRU decode, 74S287	1
U15	IC, SN74S241N	1
U16	Network, SN74LS240N	1
U17	Network, SN74LS132N	1
U18, U41	Network, SN74LS112N	2
U19, U20	Network, SN74S114N	2
U21	IC, MC4024P	1
U22	IC, MC4044P	1
U23	IC, SN74LS259N	1
U24	IC, SN74S09N	1
U25	Network, SN74S32N	1
U26	Network, SN74S02N	1
U27, U64	Network, SN74S112N	2
U28	Network, SN74S74N	1
U29	Network, SN7409N	1
U30	Network, SN74LS138N	1

Symbol	Description	Qty
U31	TIM 9904A four-phase clock generator driver	1
U32	Microprocessor, TMS 9900	1
U33, U34, U35	IC, SN74LS299N	3
U36, U66	Network, SN74LS00N	2
U37, U65	Network, SN74LS74N	2
U38	PROM, data separator, 74S471	1
U39, U46, U51	IC, SN74LS273N	3
U40, U53	Network, SN74LS157N	2
U42	PROM, processor decode, 74S288	1
U43, U49	Network, SN74LS251N	2
U44	Network, SN74LS08N	1
U45	PROM, read/write controller, 74S471	1
U47	IC, SN74LS161N	1
U48	TMS 9901 programmable systems interface	1
U50	IC TMS 9902A, EIA Controller	1
U52	Network, SN74LS151N	1
U54	IC, CRC Generator, FCD 9401	1
U55	Network, SN74LS145	1
U56, U58	Network, SN7438N	2
U57	IC, SN74LS241N	1
U59	Network, 75150P, Dual Line Driver	1
U60	Network, 75189N, Quad Line Receiver	1
U61	Network, SN74LS164N	1
U62	PROM, sync/precomp,74S471	1
U63	IC, SN74LS166N	1
U67	Network, SN74LS04N	1
U68	Network, SN74LS125N	1
Y1	Crystal, quartz, 12 MHz	1
Jumper Plugs	These plugs are available from: Berg Electronic, Inc. Rt. 3 New Cumberland, Penn. 17070 Berg part number 65474-005	

INDEX

In the page number list of this alphabetical index, subject matter is covered by a table if a T precedes the page number, or is covered by a figure if an F precedes the page number.