EK-OTA79-TM-001

TA79, TU79, TA78, TU78 Magnetic Tape Subsystem

Technical Manual

digital™

EK-OTA79-TM-001

TA79, TU79, TA78, TU78 Magnetic Tape Subsystem

Technical Manual

Prepared by Educational Services
of Digital Equipment Corporation

1st Edition, March 1988

The information in this document is subject to change without notice.
Digital Equipment Corporation assumes no responsibility for any errors that may appear in this document.

LEGAL NOTICE: TMSCP and STI protocols and documentation are the proprietary information of Digital Equipment Corporation. UNIBUS/Q-BUS/ BI-BUS port drivers and documentation for MSCP/TMSCP products are also proprietary information of Digital Equipment Corporation.

FCC NOTICE: This equipment generates, uses, and may emit radio frequency energy. The equipment has been type tested and found to comply with the limits for a Class A computing device pursuant to Subpart J of Part 15 of FCC Rules, which are designed to provide reasonable protection against such radio frequency interference when operated in a commercial environment. Operation of this equipment in a residential area may cause interference in which case the user at his own expense may be required to take measures to correct the interference.

The following are trademarks of Digital Equipment Corporation, Maynard, Massachusetts.

digital™	MASSBUS	Rainbow	VAXstation
DEC	PDP	RSTS	VAXstation II
DECmate	P/OS	RSX	VMS
DECUS	Professional	RT	VT
DECwriter	Q-Bus	UNIBUS	Work Processor
DIBOL	Q22-Bus	VAX	

Mate-N-Lok is a trademark of AMP, Inc.

Contents

CHAPTER 1 — GENERAL DESCRIPTION

CHAPTER 2 — FORMATTER TECHNICAL DESCRIPTION

CHAPTER 3 — TAPE DRIVE, THEORY OF OPERATION

Tables

CHAPTER 1
GENERAL DESCRIPTION

1.1 INTRODUCTION

This manual contains functional and detailed descriptions of the tape drive and formatter hardware. The subsystem is a high performance magnetic tape drive. The TA78/79 provides magnetic tape backup and data interchange for fixed-media disk drives controlled by the HSC, whereas the TU78/79 uses the Massbus directly from the system.

Both tape drives write and read data in 9-track phase-encoded (PE) or group code recording (GCR) format. Bit density is 1600 bits per inch (bits/in) for the PE format and 6250 bits/in in the GCR format. The tape drive can read or write data in the forward direction, or read data in the reverse direction. The read/write tape speed is 125 inches per second (in/s). They also offer the following features.

- Single-track error correction while reading PE or GCR data
- Two-track error correction while reading GCR data
- Resident in-line microdiagnostics
- Error repositioning for retries

1.2 SPECIFICATIONS

The following list includes the mechanical, electrical, functional and environmental specifications of the tape subsystem.

Tape (computer grade)	
Width	12.6492 mm ± 0.0508 mm (0.498 in ± 0.002 in)
Thickness	0.0381 mm (1.5 mil)
Tape tension	2.780 N (10.0 oz +0.0/ 1.0 oz) nominal
Reel diameter (autoload)	266.7 mm (10.5 in) maximum (note 1) and wraparound cartridge 1 and 2
Recording modes	
PE	1600 bits/in (3200 FCI)
GCR	6250 bits/in (9042 FCI)
Magnetic head	Dual stack (with erase head)

Tape speed	
Read/write	3.2 m/s (125 in/s)
Rewind	11.2 m/s (440 in/s)
Instantaneous speed variation	± 1 percent
Long term speed variation	± 1 percent (forward) ± 2 percent (reverse)
Rewind time for 731.5 m (2400 ft) tape	65 seconds nominal (85 seconds maximum)
Tape cleaner	Dual blade type connected to vacuum supply
Interchannel displacement error	
Read	3.81 m (150 in) maximum (note 2)
Read-after-write	8.89 m (350 in) maximum (note 3)
Start/stop time	
Start	1.3 ms ± 0.1 ms
Stop	1.1 ms ± 0.1 ms
Start distance	1.91 mm +0.25/ 0.38 mm (0.075 in +0.010/ 0.015 in)
Stop distance	2.16 mm +0.25/ 0.38 mm (0.85 in +0.010/ 0.015 in)
Beginning of tape (BOT) and end of tape (EOT) detectors (note 4)	Photoelectric
Tape creepage	None
Pneumatic interlock	Tape motion disabled when vacuum is lost in vacuum column
Load time	No greater than 10 seconds without a retry, and 20 seconds with a retry for 10.5-in reels
Unload time	Less than 7 seconds for 10.5-in reels
Write gap to read gap distance	0.381 cm ± 0.013 cm (0.150 in ± 0.005 in)
Error detection	
PE	Single-track error correction
GCR	Single-track error correction always: double-track error correction with pointers: CRC/ACRC check character verification
Maximum record length	65,536 (64K) characters (PE and GCR)

Minimum record length	1 character (PE and GCR)
Data reliability	Established by error rate of transport as follows.

	GCR	PE
Recoverable	Read 1 in 10(9) Write 1 in 10(7)	1 in 10(9) 1 in 10(7)
Nonrecoverable	Read 1 in 10(10)	1 in 10(10)

Note:

(10) = 10,000,000,000
(9) = 1,000,000,000
(7) = 10,000,000.

Operating temperature	15o to 32o C (59o to 90o F) (note 5)
Nonoperating temperature	40o to 66o C (40o to 151o F)
Humidity (operating)	20 to 80% with a maximum wet bulb of 25oC (77oF) and a minimum dry bulb of 2oC (36oF)
Operating altitude	0 m to 2438 m (0 ft to 8000 ft) (note 6)
Nonoperating altitude	9100 m (30,000 ft) maximum
Power	
Volts ac	208 V to 240 V at 60 Hz 208 V to 240 V at 50 Hz
Frequency	50 Hz ± 1 Hz or 60 Hz ± 1 Hz
Kilovolt amp (kVA)	
Standby (loaded)	1.74 kVA maximum
Start/stop	2.90 kVA maximum

Notes:

1. 177.8 mm (7 in) and 216.0 mm (8.5 in) reels may be used but cannot be autoloaded.
2. The maximum displacement between any two bits of a character when reading a master tape using the read section of the tape head.
3. The maximum displacement between any two bits of a character on a tape written with all ones using the write section of the tape head.
4. Approximate distance from detection area to write head gap is 35.6 mm (1.40 in).
5. For data transfer, the operating temperature is dictated by the nature of the tape media.
6. Operation above 610 m (2000 ft) requires installation of high altitude pulleys and belts in tape drive power pack.

1.3 RELATED DOCUMENTS

The following list describes documents related to the TA78/TA79/TU79/TU78 magnetic tape drive.

Title	Document Number
TA78 Magnetic Tape Drive User's Guide	EK-0TA78-UG
TA78 Magnetic Tape Drive Service Manual	EK-0TA78-SV
TA78 Magnetic Tape Drive Maintenance Guide	AA-P673A-TK
TA78 Magnetic Tape Drive IPB	EK-0TA78-IP
874 Power Controller IPB	EK-00874-IP
TA78 Field Maintenance Print Set	MP-01614-00
TA79,TU79 Magnetic Tape Subsystem Service Manual	EK-OTA79-SV
TA78/79 System Error Logging/Error Handling Supplement	EK-OTA79-EL
TA79,TU79 Magnetic Tape Subsystem Installation Manual	EK-OTA79-IN
TA79,TU79 Magnetic Tape Subsystem Owner's Manual	EK-OTA79-OM

1.4 TAPE FORMATS

The formatter used in the tape subsystem is industry compatible with 1600 bits/in PE and 6250 bits/in GCR tape formats. Both formats are described in the following paragraphs.

As tape moves past the read head, magnetic flux transitions on the tape cause the read head to produce voltage pulses. The direction of flux transition determines the polarity of the output pulses. The phase-encoding method of data formatting defines a zero bit as a negative-going transition and a one bit as a positive-going transition. To achieve both increased capacity and data rate, higher density recording is necessary. This requires a different magnetic encoding method because the industry-standard 1600 bits/in code is near its density limit. Group-coded recording (GCR) greatly improves the data packing density, compensating for both mechanical and media imperfections.

GCR is similar to NRZI (nonreturn to zero indiscrete), in that a one bit is represented by a flux transition and a zero bit is represented by the absence of a flux reversal. However, by definition, no more than two zeros may be recorded in succession on a given track; this assures synchronization.

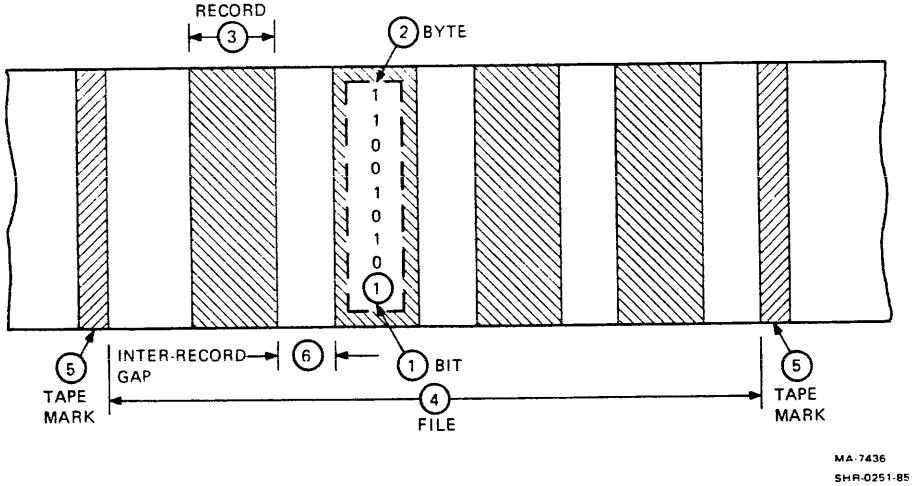
Group coding avoids recording more than two successive zeros by translating four bits into five. Four bits can be grouped in any of 16 combinations from 0000 to 1111. Five bits allow 32 possible combinations; those that begin or end with more than one zero and those that contain more than two internally successive zeros are not used. After elimination, 17 codes remain; of those, the 11111 code is reserved for synchronization

purposes. The remaining 16 combinations of five bits are matched with the 16 combinations of four bits. During GCR recording, each group of four bits in a sequence is translated to the corresponding 5-bit combination and recorded similar to NRZI form. During reading, the opposite occurs: the 5-bit combination is translated to the original 4-bit code. Table 1-1 lists definitions that apply to both PE and GCR formatting modes. The numbers reference portions of Figure 1-1.

Table 1-1 Common Tape Term Definitions

Figure 1-1 Reference	Name	Meaning
1	Bit	A bit is the smallest unit of binary information, either a logic 0 or 1.
2	Byte (or character)	A byte comprises several bits. In this example, a byte on the tape comprises eight bits plus parity.
3	Record (or block)	A record consists of multiple bytes, taken in succession. The programmer determines the number of bytes within a given record.
4	File	A file consists of many records, taken in series separated by interrecord gaps. (See 6, below.) The program determines the number of records in any given file.
5	Tape mark (or file mark)	A tape mark is a unique control record used to separate files.
6	Interrecord gap (or interblock gap)	An IRG is a section of tape that contains no information; it is used to separate records.
NA	Tape format	A tape format comprises the entire set of unique parameters used to define the recording mode.

Figure 1-1 Common Tape Format Features



1.4.1 PE Format (1600 bits/in)

This paragraph describes phase-encoded tape formatting. Figures 1-2 and 1-3 complement the description.

When reading PE data in the forward direction, a logical one data bit is defined as a flux reversal to the polarity of the interblock gap (the tape is magnetized, with the rim end of the tape being a north pole). Conversely, a logical zero data bit is defined as a flux reversal to the polarity opposite that of the interblock gap.

Refer to Figure 1-2. A beginning of tape (BOT) marker (photoreflective strip) on the tape defines the beginning of the permissible recording area. Information is recorded on nine parallel tracks, each having a 0.109 cm (0.043 in) minimum width. The tracks are numbered consecutively, beginning with track 1 at the reference edge, as follows:

Track	1 2 3 4 5 6 7 8 9
Binary Weight	2 0 4 P 5 6 7 1 3

where

P is the parity (odd) bit.

An identification burst (IDB) is recorded near the BOT marker to identify the PE method of recording. The IDB consists of 1600 fr/in (flux reversals per inch) on physical track 4 and erasures on all remaining eight tracks.

Figure 1-3 shows the format of PE recorded data. The numbers on the figure relate to the following definitions.

Figure 1-2 PE Recording Format (1600 Characters per Inch)

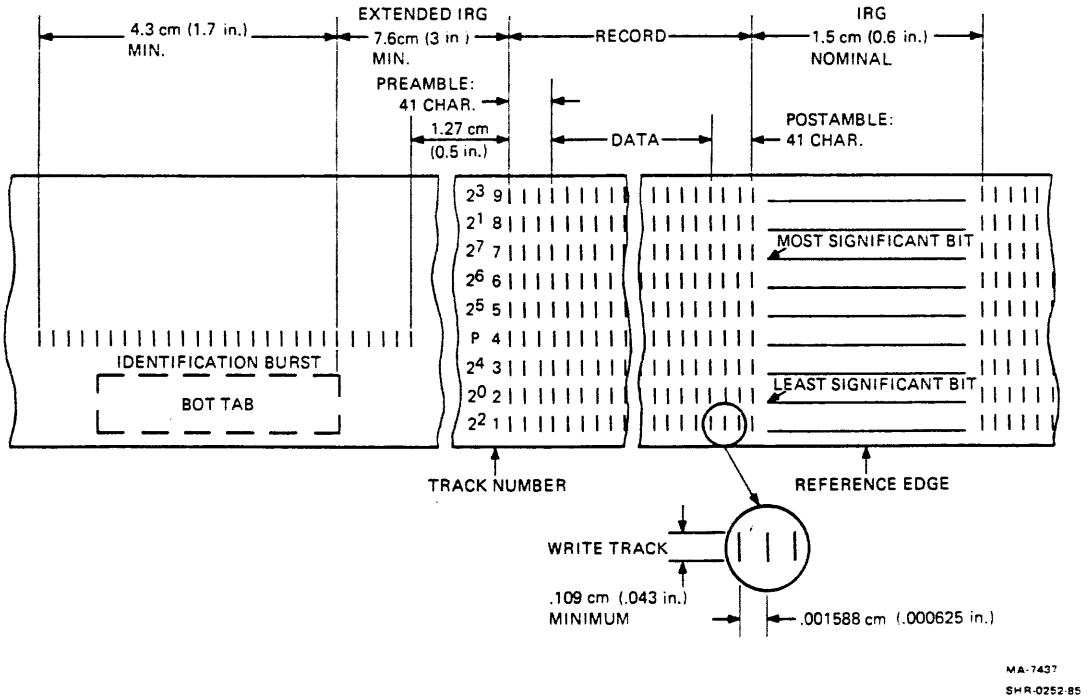
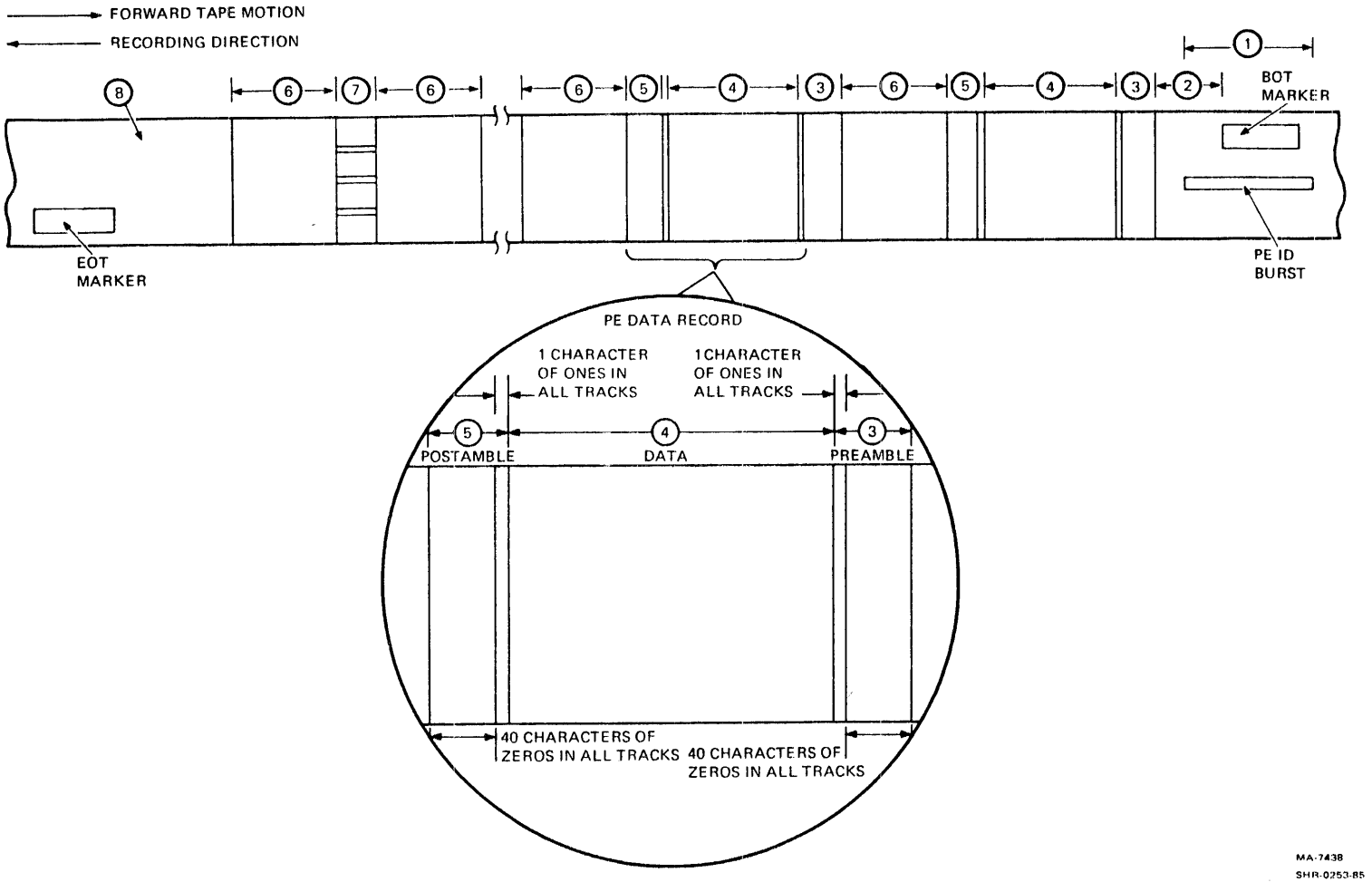


Figure 1-3 PE Tape Format



Density Identification Area — 1

The PE recording method is always identified by a recording burst beginning at the BOT marker, comprising 1600 flux changes per inch (fc/in) on the parity track (physical track 4). The burst must begin at least 4.32 cm (1.7 in) before the trailing edge of the BOT marker, continue past the trailing edge, and end at least 1.27 cm (0.5 in) before the first data record.

Initial Gap — 2

The initial gap is the required space between the trailing edge of the BOT marker and the first recorded character. Its length must be 7.62 cm (3 in) minimum, 7.62 m (25 ft) maximum.

Preamble — 3

A preamble always precedes the data in each data record. It consists of 41 characters; the first 40 characters are all zeros in all tracks, and the remaining character contains all ones in all tracks.

Data Record — 4

The data portion of a record contains from 1 to 65,536 ASCII characters (ANSI standard).

Postamble — 5

A postamble always follows the data in each data record. It consists of 41 characters: the first character contains all ones in all tracks, followed by 40 characters containing zeros in all tracks.

Interrecord Gap — 6

These gaps separate data records. They may vary in length from 1.27 cm (0.5 in) minimum to 7.62 m (25 ft) maximum; 1.524 cm (0.6 in) is nominal.

Tape Marks — 7

Tape marks are special control records (without preamble or postamble) used to signify the end of a data file. They consist of 64 to 256 flux reversals in physical tracks 1, 2, 4, 5, 7, and 8. Tracks 3, 6, and 9 are dc erased.

Reference Edge — 8

The tape reference edge is the top edge when viewing the oxide-coated side of the tape, with the rim end of the tape to the observer's right.

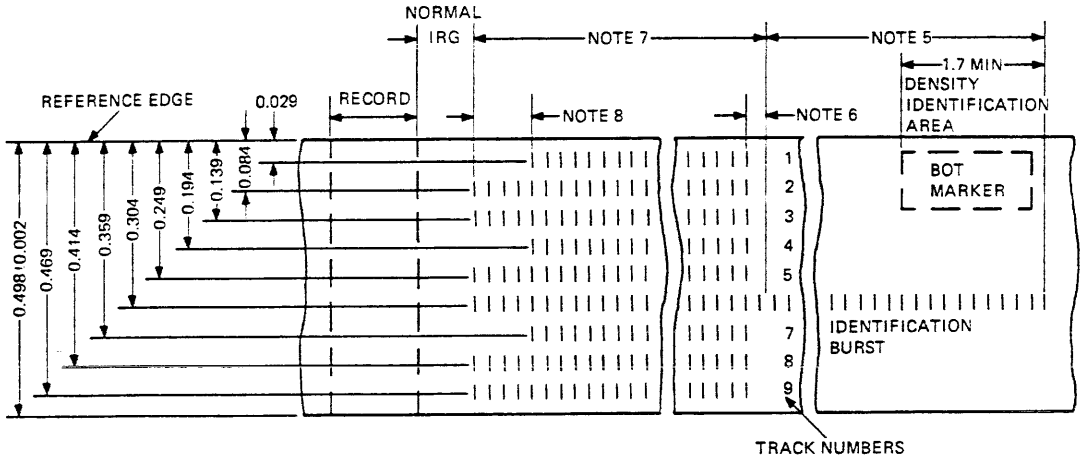
1.4.2 GCR Format (6250 bits/in)

This paragraph describes group-coded recording formatting. Figures 1-4 and 1-5 complement the description.

GCR is a recording technique that collects and encodes groups of characters prior to placing them on tape. It is similar to NRZI recording, in that a one is represented by a reversal of magnetic flux polarity; a zero is represented by no magnetic flux polarity reversal. As mentioned, GCR encoding avoids recording more than two successive zeros by translating four bits into five. This ensures that synchronization, once achieved, is not lost. Data packaging density increases to 6250 bits/in and, with a unique synchronization pattern of all ones, is very reliable operating at 318 cm/sec (125 in/sec).

GCR hardware consists of two major sections: a phase-locked oscillator (PLO) to control data flow through the system, and a 4-to-5 bit data encoder/decoder. During a read or write operation, data applied to the PLO inputs brings the output frequency in phase with the data inputs. While data is read, the PLO follows minor changes in the data rate corresponding to tape speed changes, but averages any rapid changes caused by shifting flux reversals.

Figure 1-4 GCR Recording Format (6250 Characters per Inch)



NOTES:

(1) TAPE IS SHOWN IN 6250 MODE, OXIDE SIDE UP.
(2) ALL DIMENSIONS ARE GIVEN IN INCHES.
(3) TRACK PLACEMENT TOLERANCE IS ± 0.003 FOR EACH TRACK.
(4) TAPE TO BE FULLY SATURATED IN THE ERASE DIRECTION IN THE INTERRECORD GAP AND THE ID AREA.
(5) ID BURST.
(6) UNDEFINED GAP.
(7) ARA BURST.
(8) ARA ID CHARACTERS.

MA-7439
SHR-0254-85

Figure 1-5 GCR Tape Format

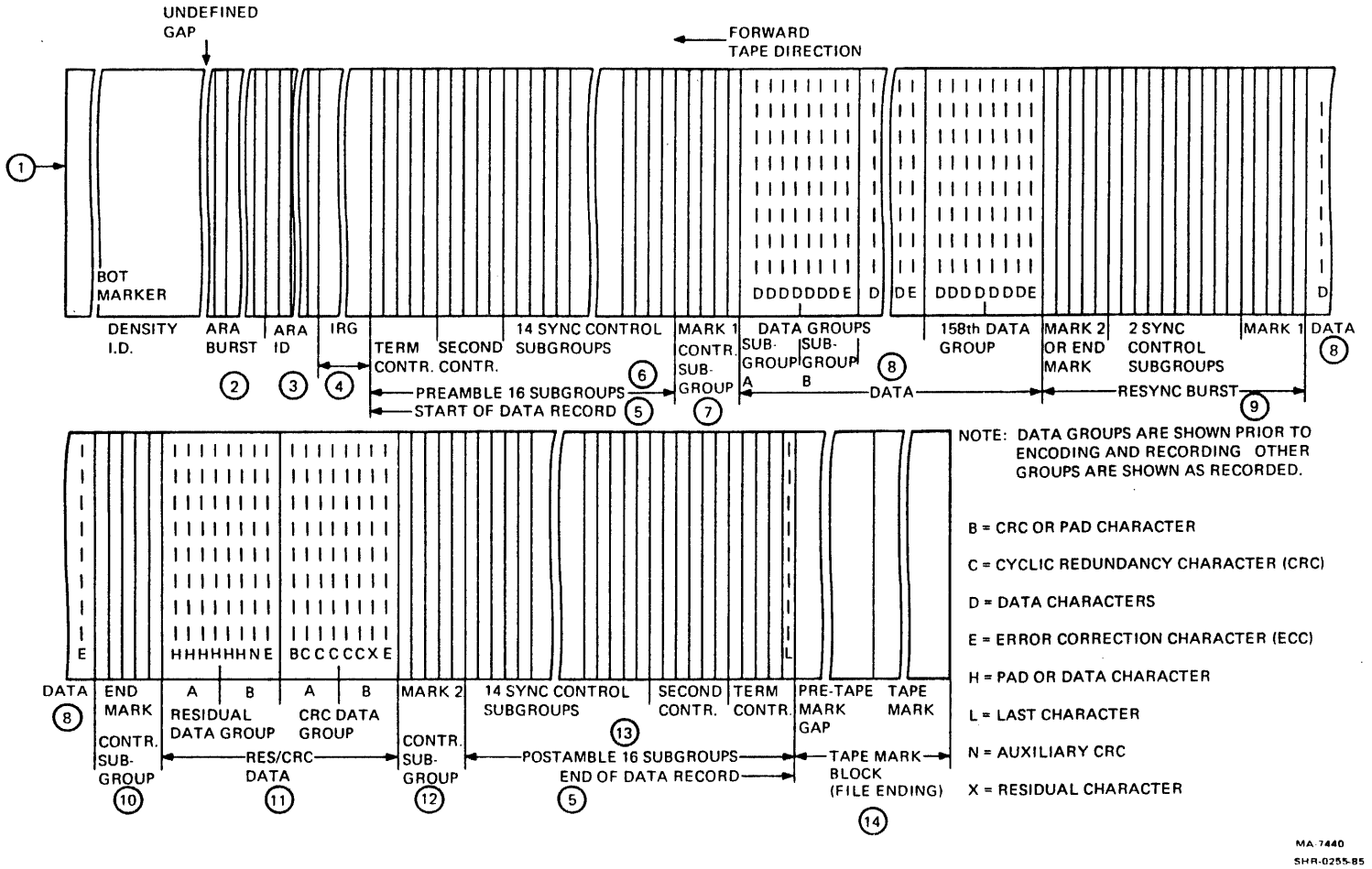


Figure 1-4 shows the tape dimensions and the recording format used in GCR operation. A beginning of tape (BOT) marker (photoreflective strip) on the tape defines the beginning of the permissible recording area. Information is recorded on nine parallel tracks, each having a 0.109 cm (0.043 in) minimum width. The tracks are numbered consecutively, beginning with track 1 at the reference edge, as follows:

Track	1 2 3 4 5 6 7 8 9
Binary Weight	2 0 4 P 5 6 7 1 3

where

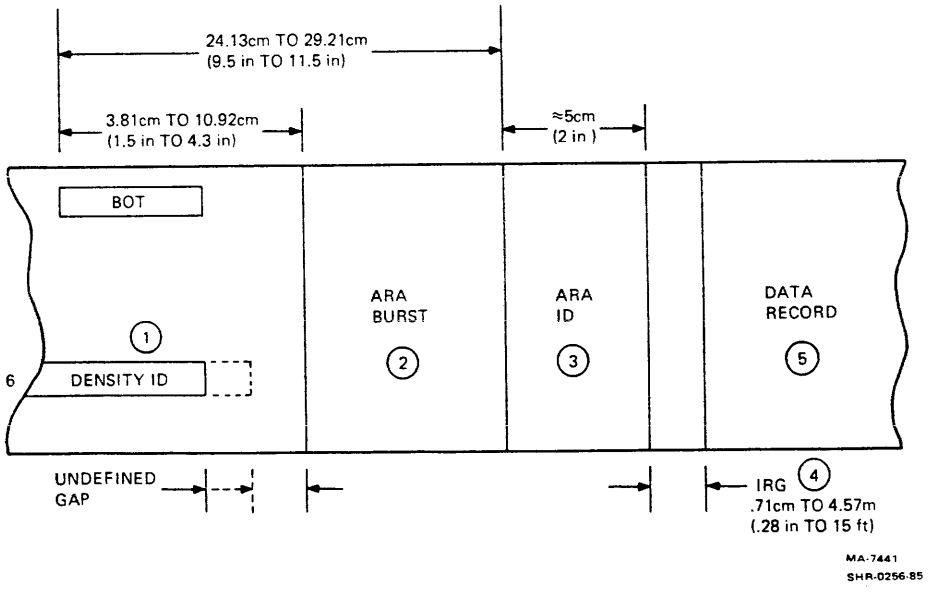
P is the parity (odd) bit.

An identification burst (IDB) is recorded near the BOT marker to identify the GCR method of recording. The burst is in the PE frequency range on physical track 6 only; all other tracks are dc erased. As shown in Figure 1-4, the burst begins at least 4.318 mm (1.7 in) before the trailing edge of the BOT marker and extends past the trailing edge of the BOT marker.

Figure 1-5 is a complete overview of the GCR tape format, identifying all major areas. Each major area has a circled number that references the section of text describing that tape area. Diagrams that detail the major tape area described complement selected descriptions.

Figure 1-6 details items 1 through 5.

Figure 1-6 GCR ID Area



Density ID — 1

The GCR recording method is identified by a recording burst near the BOT marker. The burst is in the PE frequency range on physical track 6; all other tracks are erased. The burst begins at least 4.318 mm (1.7 in) before the trailing edge of the BOT marker and extends past the trailing edge of the marker.

ARA Burst — 2

Immediately following the ID burst is an undefined gap and the ARA burst. The ARA burst verifies the capability of writing all tracks and allows the tape transport read amplifiers to set gains. The burst of all ones (in all tracks) begins no sooner than 3.81 cm (1.5 in) and no later than 10.92 cm (4.3 in) from the leading edge of the BOT marker. The burst ends no sooner than 24.13 cm (9.5 in) and no later than 29.21 cm (11.5 in) from the leading edge of the BOT marker.

ARA ID — 3

The ARA ID character is appended to the ARA burst and warns of the approaching load point in reverse tape operations. This recording burst comprises ones in tracks 2, 3, 5, 6, 8, and 9, and erasure in tracks 1, 4, and 7. The character is approximately 5.08 cm (2 in) long. At least one 0.63 cm (0.25 in) section of the 5.08 cm (2 in) recording burst must be error-free in all tracks at once.

IRG — 4

The interrecord gap (IRG) is a completely erased portion of tape separating two recorded data fields (for example, ARA ID and the preamble). Its length is 0.71 cm (0.28 in) minimum, 4.57 m (15 ft) maximum, and 0.76 cm (0.3 in) nominal.

Data Record — 5

Data records comprise the preamble, data groups, resync burst, control subgroups, and the postamble.

Figure 1-7 details items 6 through 8.

Preamble — 6

The preamble consists of 16 control subgroups of 5 bytes each (80 characters). Fourteen of the subgroups are all ones. The preamble initializes and synchronizes the read circuitry.

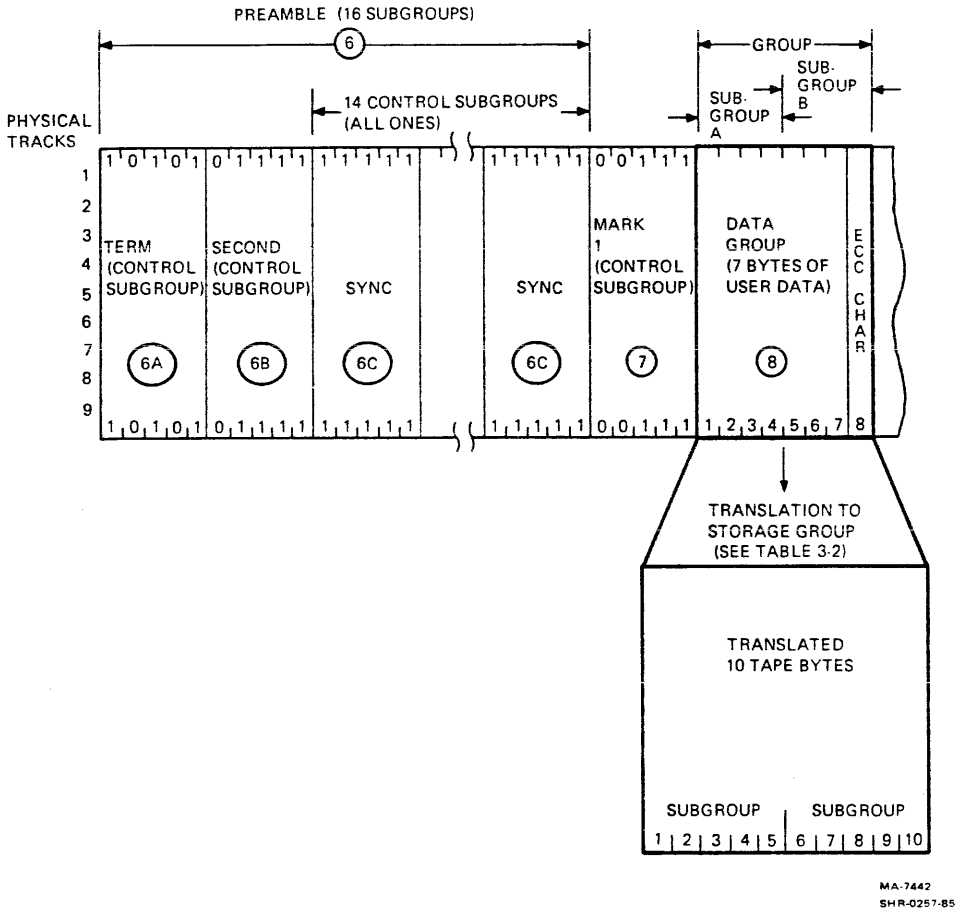
Term — 6A

The terminator control subgroup (TERM) is one set of nine parallel 5-bit serial values of 10101 (in the respective tracks at the BOT end of each block) and 1010L (at the EOT end of each block), where L represents a bit of a last character that restores the magnetic remanence to the erase state (resets the last character).

This subgroup introduces the preamble portion of the data block.

The data pattern in this subgroup ensures high-level read detection circuit signals by providing a long wavelength input at the beginning of a read operation.

Figure 1-7 GCR Preamble and Data



Second — 6B

The second control subgroup is one set of nine parallel 5-bit serial values of 01111 (in the respective tracks at the BOT end of the block) and 11110 (at the EOT end of the block). It follows the TERM subgroup in the data block and introduces the sync control subgroups, of which it is a part.

Sync — 6C

The sync control subgroup is one set of nine parallel 5-bit values of 11111 (the reserved sync code) in the respective tracks. Fourteen of these subgroups comprise the subgroups that synchronize the read-reference oscillator.

Mark 1 — 7

The mark 1 control subgroup is one set of nine parallel 5-bit serial values of 00111 in the respective tracks. Directly following the sync control subgroups ending the preamble, mark 1 indicates data is coming. It correctly sets the buffer counters, so that the data being read is formatted into the right 5-byte groups. This is necessary to retranslate from 5-bit to 4-bit codes during a read operation.

Data — 8

Data is defined as any recorded section of tape that contains only data and the error correction code (ECC) character (no control subgroups). The data is formatted into data groups, of seven data bytes each, followed by the ECC character. The data groups are divided into two data subgroups, A and B. Data subgroup A comprises four data bytes before translation; data subgroup B comprises three data bytes plus the added ECC character before translation.

Vertical parity is always odd, and ECC character parity is recorded on track 4. The ECC character is used in an algorithm that helps in the possible recovery of errors occurring in that data group. Data groups do not appear on tape as shown (Figure 1-7) but are translated to storage groups before being written.

The storage group (Figure 1-7) represents the coded data as written on tape. This code is the 4-to-5 bit translation process described in the beginning of this chapter. Table 1-2 lists the 4-to-5 bit translation codes.

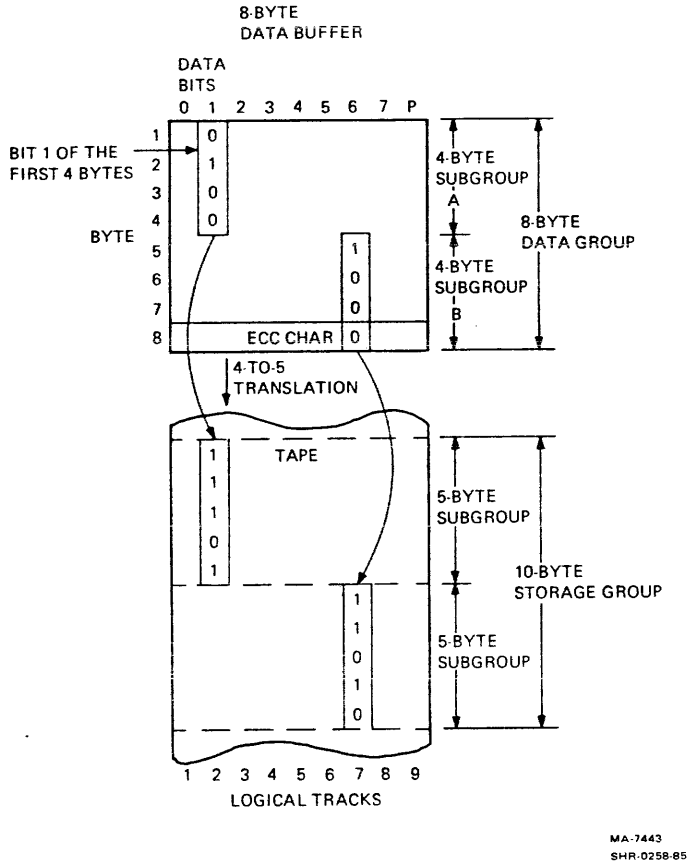
Remember, in GCR recording, four bits are translated into five bits prior to recording. This assures that data applied to tape contains no more than two successive zeros in any track, permitting very reliable read synchronization.

Table 1-2 Four-to-Five Translation Record Codes

4-Bit Data Value (Group Positions: 1234/5678)	5-Bit Recording Value (Group Positions: 12345/678910)
0000	11001
0001	11011
0010	10010
0011	10011
0100	11101
0101	10101
0110	10110
0111	10111
1000	11010
1001	01001
1010	01010
1011	01011
1100	11110
1101	01101
1110	01110
1111	01111

Refer to Figure 1-8. An 8-byte buffer within the formatter holds seven data bytes and the ECC character byte at one time. As mentioned, the 8-byte set is a data group; each 4-byte half is a data subgroup. Remember that this is not yet a physical representation of data on tape, but just a buffered data storage area within the formatter. Grouping the data into 8-byte groups aids in error correction; the subgroups are for translation purposes.

Figure 1-8 Translation and Physical Data Placement on Tape



Data is written by translating corresponding bits (for example, bit 1) of the first four bytes in the 4-to-5 bit code, and then recording them as five bits in the logical track (for example, track 1) corresponding to the bits. This occurs simultaneously for all nine tracks. After the first four bytes in the data group are written on tape, the second four bytes are translated and written in the same manner. The resultant 10-byte group on tape is called a storage group. A data group translates into a storage group and vice versa.

No more than 158 continuous data groups may be recorded in a data block. If more than 158 data groups are in the incoming record, a resync burst is required before the recording can continue. (See Resync Burst.)

Resync Burst — 9 (See Figure 1-9.)

The resync burst comprises the following control subgroups: mark 2, two sync groups, and mark 1. It resynchronizes data when the incoming record exceeds the 158 data group limit, and allows the phase-locked loop to be re-enabled after an error.

Mark 2 — 9A

The mark 2 control subgroup is one set of nine parallel 5-bit serial values of 11100 in the respective tracks. It marks the end of incoming data and the start of nondata information.

Sync — 9B

The two sync control subgroups are all ones in all tracks. These bits resynchronize the read circuitry.

Mark 1 — 9C

The mark 1 control subgroup is one set of nine parallel 5-bit serial values of 00111 in the respective tracks. It indicates new data is arriving next. Note that mark 1 is the reciprocal of mark 2. This allows the same subgroups to be used in reverse operation.

Figure 1-9 Resynchronization Burst Format

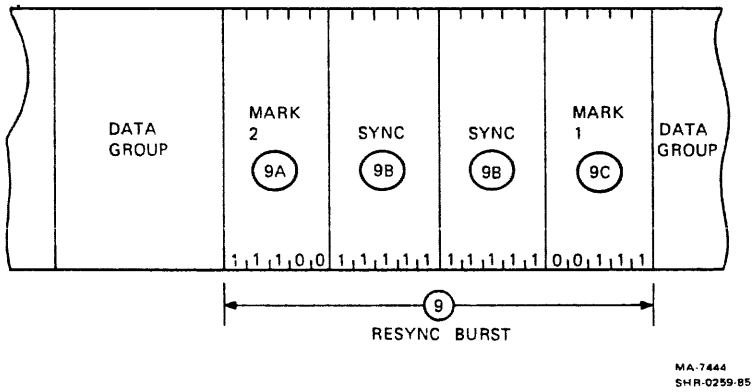


Figure 1-10 details items 10 through 13.

End Mark — 10

The end mark control subgroup consists of one set of 5-bit serial values of 11111 on all tracks. This mark indicates the arrival of the residual data group. (See Residual Data Group.)

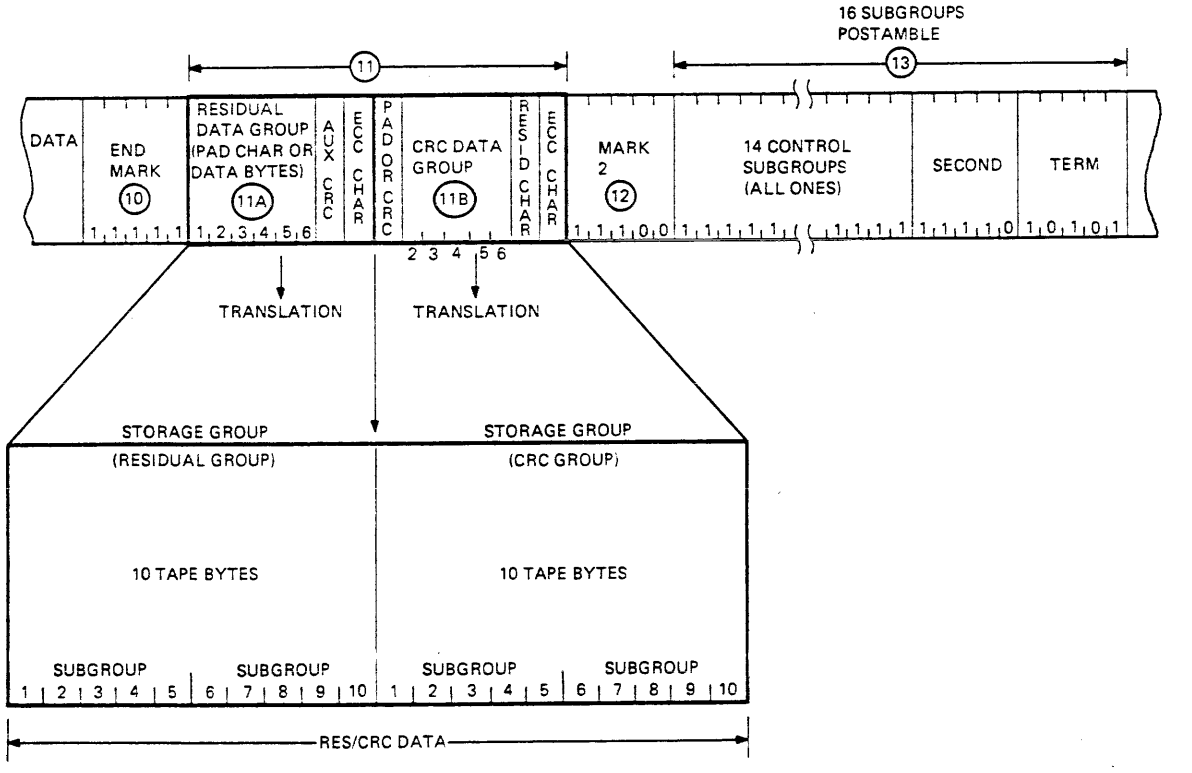
RES/CRC Data — 11

This data comprises both the residual and CRC data groups (explained below). These two groups are always written together and appear at the end of a data record.

Residual Data Group — 11A

This data group is formed when six or less data bytes remain in a data record. The seventh byte is the auxiliary CRC character (a check character for data validity); the eighth byte is the normal ECC character. If less than six residual data bytes remain, "pad" characters of all zeros (with correct parity) are added to the residual data group to create the required six bytes. Keeping in mind that eight bytes are required in GCR mode, note that the residual data group comprises the remaining data bytes, any required pad characters, the auxiliary CRC character, and the ECC.

Figure 1-10 End Data and Postamble



MA-7445
SHR-0260-85

CRC Data Group — 11B

This group includes the odd-parity CRC character, a byte guaranteeing the parity of the CRC character, the residual character, and the ECC character.

Before this data group is written, the CRC character normally has even parity if an odd number of data bytes existed, or odd parity if an even number of data bytes existed. If the record had an odd number of data bytes, then the CRC character is even. The CRC character must then be made odd, because an even parity byte is not permitted in a GCR data group. An additional "pad" byte of all zeros and a parity bit is added to the record to accomplish this. The addition of the byte changes the number of bytes in the CRC generation from odd to even and provides an odd-parity CRC character.

The next five bytes of the CRC data group are identical CRC characters. The additional CRC characters complete the CRC data group, as no more data to be read exists.

The residual character is next in the group. By definition, this character acts as a record data counter. Bits 0—2 of the byte are used as a modulo-7 counter, which indicates how many of the residual data group bytes are data. When the record is read, only the data bytes are retrieved; all pad characters are dropped.

As in all groups, the ECC character verifies data and isolates any errors during a read operation.

Mark 2 — 12

As before, this control subgroup marks the end of data and the start of nondata information.

Postamble — 13

The postamble is a mirror image of the preamble. It works in read reverse operations the same way as the preamble in read forward operations. Refer to the preamble description for further information.

Tape Mark Block — 14 (not shown in diagram)

If the programmer wants to create a tape mark, a pretape mark gap is provided directly after the postamble of the preceding data block. The tape mark is written as a special block by the Write Tape Mark command. The tape mark is specified as 250 to 400 flux changes, all ones at 9042 fr/in in tracks 1, 2, 4, 5, 7, and 8; tracks 3, 6, and 9 are erased. The tape mark provides end-of-file information. More than one file may be written on a reel of tape.

1.5 SYSTEM-LEVEL DESCRIPTION

Section 1.5 introduces the internal system concepts of the TS78 and TM78 formatter and outline the functional components. Section 1.5 also explains how the internal components communicate with the host computer system(s), tape transport(s), and one another.

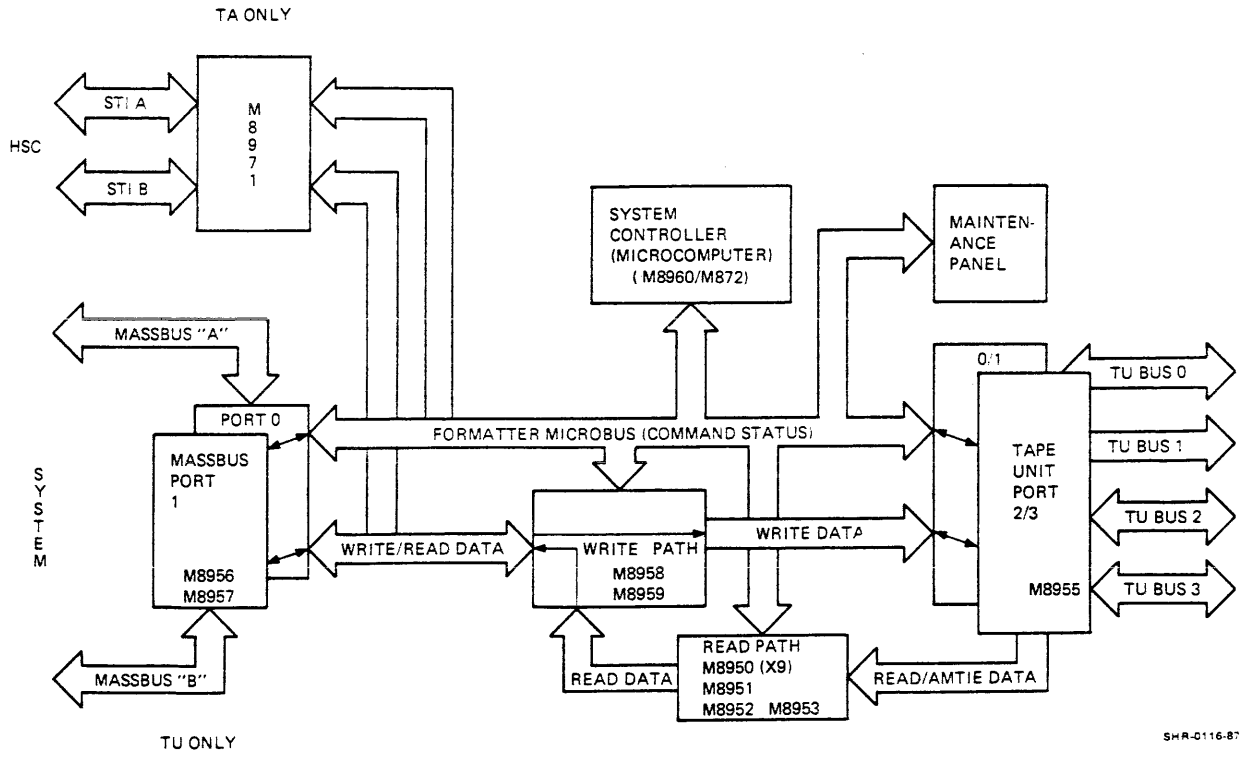
1.5.1 Formatter Functional Components

Six main functional components make up the formatter's internal system: the system controller or microcomputer, the interconnect port(s), the write path, the read path, the tape unit (TU) port(s), and the maintenance panel. Refer to Figure 1-11 for a simplified internal system block diagram.

The task of each main system component is as follows.

System Controller
The system controller module is a continuously running microcomputer system that forms the heart of the formatter. It interprets commands issued by the host system and issues instructions to the other internal components, enabling them to accomplish the system-level commands.

Figure 1-11 Formatter Internal System Block Diagram



Interconnect Port STI Bus

The host computer system communicates with the TS78 over the STI bus and through the HSC to the STI bus port. The host issues subsystem commands and write data, and the TS78 returns status and read data through this port. The port consists of the M8971 STI Interface and M8970 STI protocol modules. In a dual-ported configuration (that is, two host systems communicating with the same TS78) the two port modules are duplicated.

Massbus Port

The host computer system communicates with the TM78 over the Massbus and through the Massbus port. The host issues subsystem commands and write data, and the TM78 returns status and read data through this port. The port consists of the M8956 Massbus data module and the M8957 common address space module. In a dual-ported configuration (that is, two host systems communicating with the same TM78) the two port modules are duplicated.

Write Path

The write path consists of the M8959 write microcontroller module and the M8958 translator module. It accepts data from the system bus port and formats it into tape characters to be written to tape. The data format depends upon the density mode selected by the host system. The write path also creates data for writing BOT ID bursts, tape marks, preambles, postambles, and certain control characters.

Read Path

The read path accepts data read from tape through the tape unit port, and unformats it from tape characters to data intelligible to the host computer system. The way data is unformatted depends upon the density mode that the tape was originally written in. The read path consists of the M8953 read path controller, nine M8950 read channel modules, the M8951 error correction code module, and the M8952 CRC/ACRC module.

Tape Unit Port

The formatter communicates with one to four tape transports over the radial tape unit bus (TU bus) through the tape unit (TU) port. The formatter issues tape unit commands and write data, and the tape unit returns status, read data, and amplitude track in error (AMTIE) data through this port. The TU port consists of one M8955 TU port module for each pair of tape drive transports connected to the formatter. Thus, a full four-transport configuration includes two TU port modules.

Control and Maintenance Panels

The panels allow maintenance personnel to communicate to and from the system controller. Through it they may evoke maintenance microprograms to troubleshoot hardware failures either off-line or on-line, concurrently with host computer commands to the subsystem. The TS78 connects to an external terminal and the TM78 utilizes an on-board keypad.

The system controller (microcomputer) communicates with the other five components (peripherals) over an internal 62-wire bus known as the formatter microbus. The microcomputer addresses these internal peripherals in much the same way as the host processor addresses the formatter. The actual mechanics of addressing and data transfer over the microbus are discussed later in this chapter.

1.5.2 Formatter System Microcomputer Function and Architecture

Paragraphs 1.5.2.1 and 1.5.2.2 describe the microcomputer in terms of internal system functions.

1.5.2.1 Microcomputer Function - The microcomputer is the tape subsystem controller and scheduler. It interprets function codes (commands) sent to the subsystem by the host operating system and/or external terminal or the maintenance panel keypad. Then, based upon the type of command, it starts running the tape transport and certain microcontrollers imbedded within other internal peripherals (read path, write path).

Commands may be sent to the subsystem through 14 different sources.

- One data transfer function code from Massbus/STI bus A
- Four nondata transfer function codes from Massbus/STI bus A
- One TM Clear (Initialize) command from Massbus/STI bus A
- One data transfer function code from Massbus/STI bus B
- Four nondata transfer function codes from Massbus/STI bus B
- One TM Clear (Initialize) command from Massbus/STI bus B
- A command (or command string) from the control or maintenance panel
- A Reset command from the MASTER RESET pushbutton

The commands are sent to the subsystem asynchronously and treated by the microcomputer on a time-shared/priority basis. All commands are handled independently by the microcomputer (although they may appear to be acted upon simultaneously by the host system) according to the following priority schedule.

Priority	Task
1	TS/TM Clear (Initialize) and Reset commands
2	Data Transfer commands
3	Nondata Transfer commands

The two interconnect ports do not have a priority scheme. All commands within a priority level are handled asynchronously.

The microcomputer also performs the following functions.

- Handles Massbus/STI bus command arbitration
- Schedules internal diagnostic routines during subsystem idle time
- Controls Massbus/STI bus data transfer protocol
- Calculates the correct value for the residual group pad count in GCR mode
- Provides master clocks for read/write data transfers
- Controls the access of Massbus/STI bus commands and external terminal or keypad commands to the tape units
- Monitors tape velocity during all write commands
- Interprets microcontroller status and reports errors
- Repositions tape for error retries

The microcomputer does not actually handle any system data through the formatter. All system data to and from the tape transports is processed by the ports and read/write data paths.

1.5.2.2 Microcomputer Architecture - A brief overview of its architecture in terms of system implications follows.

The microcomputer consists of a single-chip microprocessor with supporting ROM and RAM memory. It communicates with other system peripherals over a 62-line microbus. The microbus is divided into a bidirectional address bus (16 lines), a bidirectional data bus (8 lines), and a control/status bus (38 lines).

The memory within the microcomputer totals 20K (8-bit) bytes, divided into 16K bytes of ROM area and 4K bytes of RAM area. The ROM area contains a permanent version of the system monitor, and a special diagnostic monitor used only during execution of host system level standalone diagnostics. The RAM area is dynamic and serves as a scratchpad for the following purposes.

1. It provides a stack area for the microprocessor.
2. It provides an area for four tables (one per tape transport), which contain extended sense information. This information helps maintenance personnel to interpret subsystem errors.
3. It provides an area to load diagnostic programs and is, in a sense, a writable control store. When running standalone diagnostics, microcode is continually executed and overlayed in this area.
4. It provides an area to store information on the current subsystem configuration, that is, how many Massbus ports and tape transports are connected.

1.5.3 Operational Microcode Firmware

The following paragraphs briefly describe some of the functions of the internal operational microcode (or microprogram).

1.5.3.1 Microcode Overview - As mentioned, the microprocessor runs a program in ROM known as the system monitor. The monitor program runs continuously as long as power is applied to the formatter and the host system has not set the hold bit. The monitor normally cycles in a polling routine known as the idle loop. However, before entering the idle loop, the microprocessor executes instructions to place the entire formatter in a reset state. These instructions form the power on/reset flow shown in Figure 1-12. The power on/reset flow also takes a quick subsystem inventory to see what options are presently connected. Figure 1-13 shows the basic functional flow of the idle loop.

Figure 1-12 Power On/Reset Flowchart

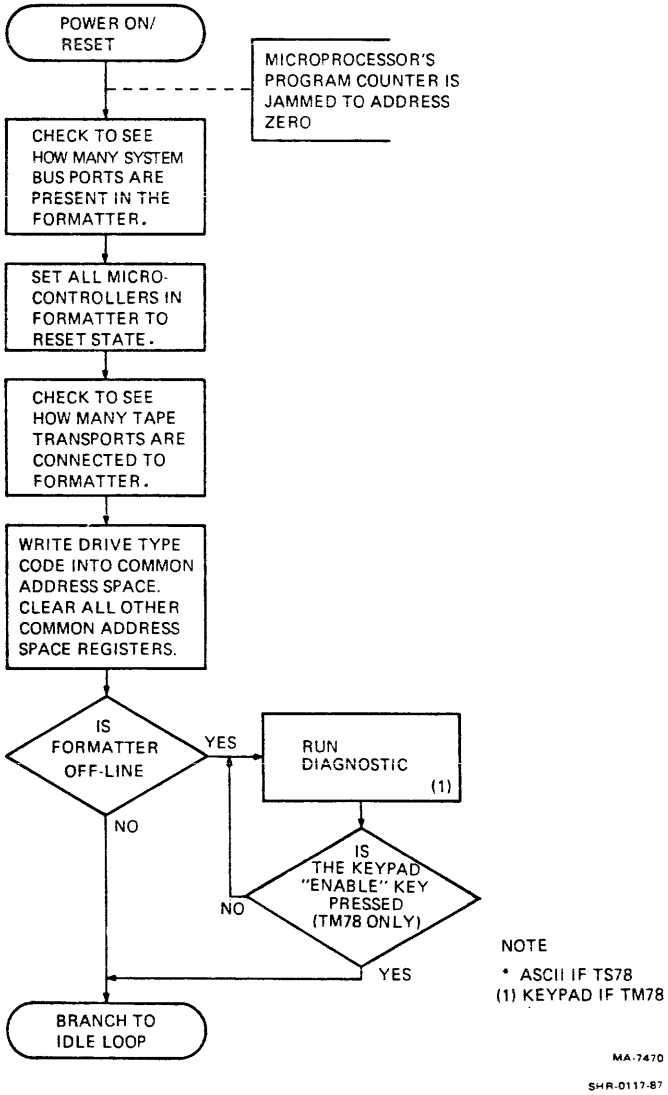
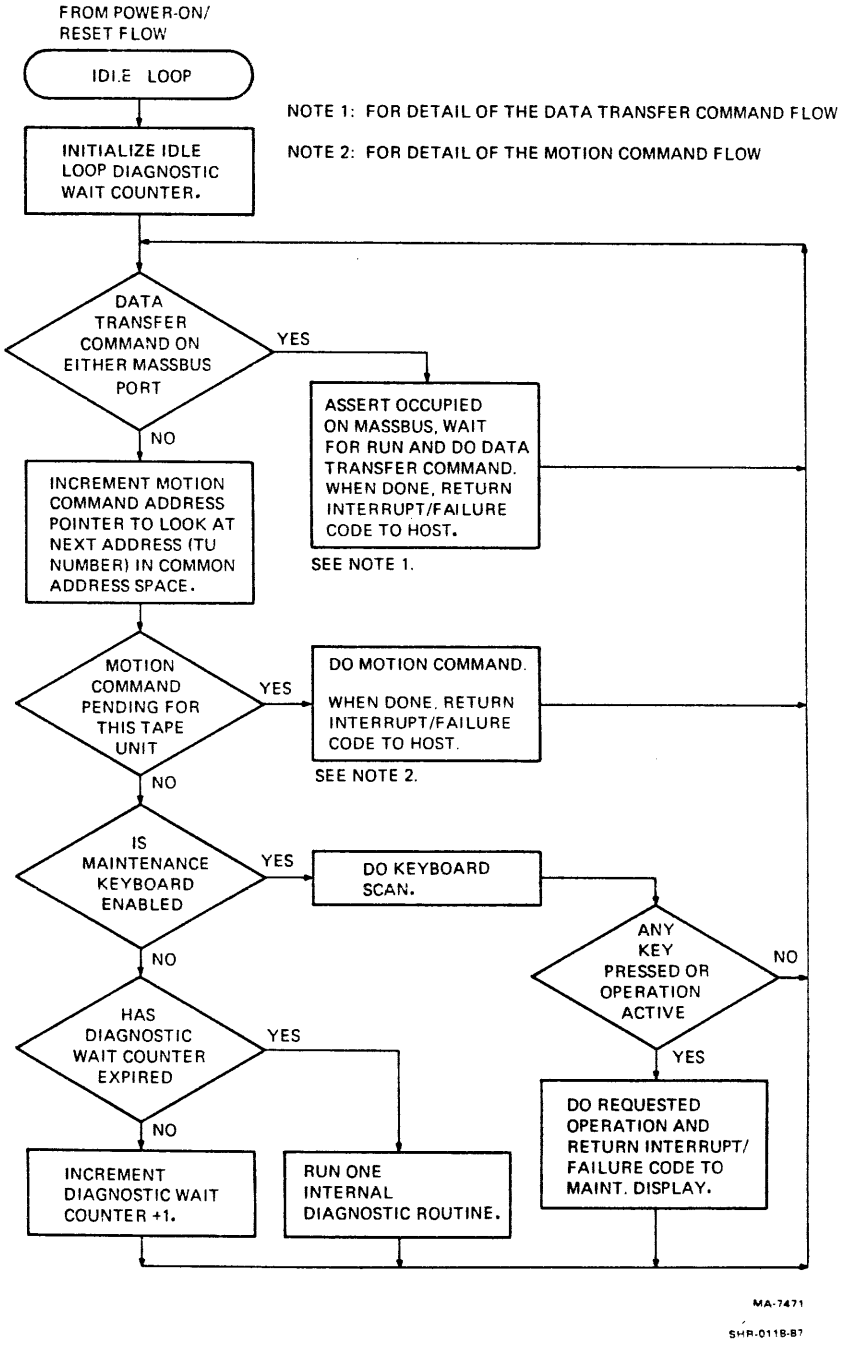


Figure 1-13 Idle Loop Simplified Flowchart



1.5.3.2 Polled Versus Interrupt-Driven Functions - The idle loop polls or scans the same functions over and over, sensing for any changes that it may have to act on. It looks for any command GO bit being set by the host system. This indicates a subsystem command is ready to execute. It also looks for an asserted ready status of a tape transport, indicating that a tape has just been loaded. Other polled functions are as follows.

- Does the control (TS78) or (TM78) maintenance panel require servicing? (Enable key depressed.)
- Has a system bus been connected or disconnected? (FAIL line going true or false.)
- Have N iterations of the idle loop been made without command activity? If so, branch to the internal microdiagnostic routines.

Some internal status information may be too critical in nature for the idle loop to poll, such as internal hardware parity errors or ac/dc power fluctuations. Information of this nature notifies the microcomputer immediately by means of hardware interrupt lines on the microbus. The assertion of these interrupt lines causes the microprocessor to branch to an interrupt-servicing routine in the ROM area. However, before the interrupt is serviced, the microcomputer pushes its current program counter and processor status values onto the stack in RAM area. Table 1-3 lists the types of interrupts sent to the microcomputer, their priority of execution, the microbus signal that causes each interrupt, and the hardware address that is jammed to the program counter.

Table 1-3 TM78 Internal Interrupt Summary

Priority	Interrupt	Microbus Signal That Causes Interrupt	New Microprocessor PC Value
1	Restart 0 (reset)	SW RESTART L or IA CLEAR H	0_{16} (0_8)
2	Trap	AC LO L or a ROM parity error internal to the microcomputer	24_{16} (44_8)
3	Restart 7.5	P CMD PE L	$3C_{16}$ (74_8)
4	Restart 6.5	CL CONTENTION (0) L or CL CONTENTION (1) L	34_{16} (64_8)
5	Restart 5.5	IA CAS PE H or IA ILR H	$2C_{16}$ (54_8)
6	Restart 7	P STAT PE L (interrupt)	38_{16} (70_8)

1.5.4 Formatter System Reliability/Integrity

The formatter has a wide variety of self-check features to ensure data integrity. Both hardware and firmware (microcode) checks continually verify subsystem operation.

Control and data parity is checked externally by separate hardware parity lines on the bus (control and data portions) and the transport buses (write/control/status, read and AMTIE portions). Internally, data parity is checked through all phases of the read/write data paths. The microcomputer and all system microcontrollers perform data parity self-checks on their internal ROM microprogram storage areas.

As noted, the microcomputer schedules internal microdiagnostics to run during subsystem idle time. Some of these microdiagnostics are set up and started by the microcomputer, but are actually run by microcontrollers in the read/write data paths. Other diagnostics generate and route data internally from the write path through the TU ports and read path. They also route data externally, sending data out on the TU bus and looping it back through the tape transport interface module. The microcomputer runs these internal microdiagnostics at normal operating speeds first, then switches in alternate master clock frequencies to margin the circuitry at 20 percent of the normal operating speed.

1.5.5 Formatter Internal Communication

The microcomputer communicates with other peripheral devices over the internal microbus. The microbus contains 62 signal lines, divided into a bidirectional address bus (16 lines), a bidirectional data bus (8 lines), and a control/status bus (38 lines). Figure 1-14 shows the system interconnection on the bus. The control/status portion is broken out line-by-line and shows that not all lines go to and from each internal peripheral device, as do the data lines and most address lines. This figure does not show how the various peripheral devices communicate data to one another. Data transfer and manipulation is discussed in later theory sections.

All address, data, and portions of the control/status buses are bidirectional. In addition, some of these bidirectional lines are tri-state in nature. Tri-state lines permit driving heavily loaded buses without using open-collector/pull-up resistor schemes. A tri-state line has a third level in addition to its normal ground and high TTL levels. This third level is a high-impedance (open) state, so the line neither represents a significant load (as in a grounded output) or acts as a driver (as in a HI state). An integrated circuit (IC) with a tri-state output has a corresponding data input and an enable input. With the enable input true, the output is in one of two states, HI or LO, depending upon the data input. However, with the enable input false, the output is "open" regardless of the data input's state. Thus, a line connected to this output in an open state can be driven by another tri-state enabled output. This scheme of tri-state line driving is used extensively throughout the formatter.

Figure 1-14 TM78 Microbus System Interconnection

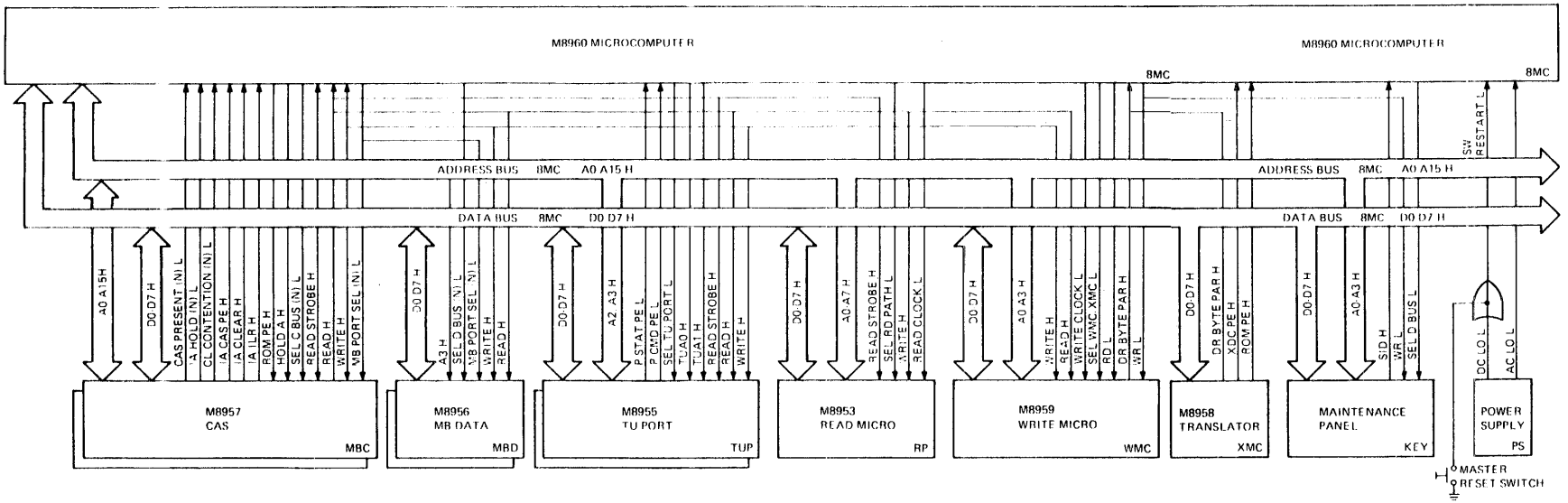
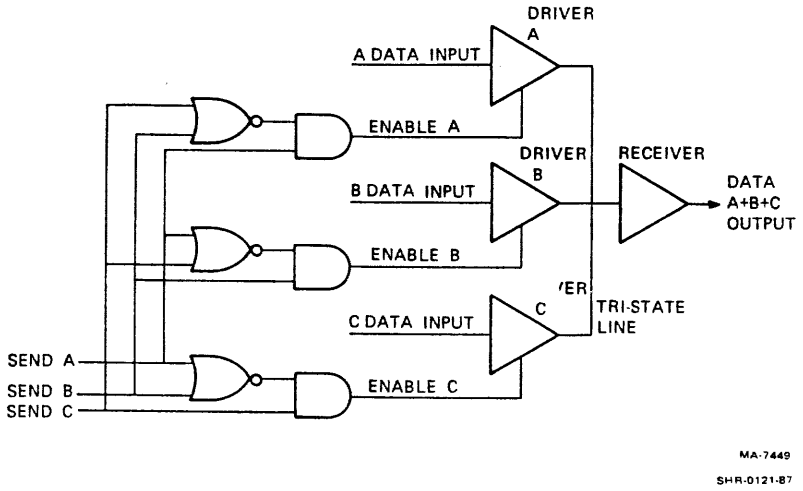


Figure 1-15 shows a hypothetical tri-state line with three drivers and a receiver connected. This line could conceivably be expanded to include twenty drivers and receivers in an actual system. Unless a hardware failure occurs, only one tri-state driver at a time should be enabled to drive a line HI or LO.

Figure 1-15 Typical Tri-State-DRNEW Line (Decoding Scheme)



1.5.6 Interface-Level Description

Paragraphs 1.5.6.1 through 1.5.6.3 outline and describe the three major functional areas of the formatter relative to communication over the bus interface. They are the recognition and processing of commands to the subsystem, write data processing in the write path, and read data processing in the read path.

1.5.6.1 Subsystem Command Execution - All operations of the tape transport subsystem can be grouped into three categories.

- Data transfer operations in which the subsystem handles data in conjunction with the host computer
- Motion operations in which the subsystem functions in a nondata transfer mode
- Subsystem sensing operations in which the host computer asks the drive for the status of a tape transport and/or internal registers

A brief discussion of how the drive executes each of these categories follows. They have, however, a number of common characteristics which are summarized here.

1. The host computer issues a command to TS78 STI interface over the STI bus from the HSC, or TM78 common address space over the Massbus control bus, along with related parameter information.
2. The drive interrogates the command and causes the tape transport to execute it.
3. Based upon the success or failure of execution, the drive issues an interrupt/failure code to the host computer and raises the bus attention (ATTN) or end of block (EBL) line.
4. The host computer reads the interrupt/failure code from common address space and takes appropriate action.

Data Transfer Operations - The four tape transports share registers in the M8971 and M8957 modules for data transfer operations. This means only one transfer per bus may be active at a time. The microcomputer continuously polls the data transfer command register in its idle loop, looking for the GO bit. When GO is set, the microcomputer assumes that the host computer previously wrote the command code, byte count, format code, skip count (if necessary), and TU number into the M8971 and M8957 modules. While executing the command, the microcomputer draws upon these parameters and sets up the internal write/read path registers accordingly.

Figure 1-16 shows an example of how the microcomputer executes a subsystem data transfer command. It executes the Write and Read data commands in much the same way, the difference being in the direction of data transfer and the type of commands issued to the write/read paths. For this reason, only a Write command is shown here.

Motion Operations - There is a separate motion command register M8971 and M8957 modules for each of the four tape transports. Up to four motion commands per system bus port may be active (one for each transport) at any time. The formatter microcomputer polls the motion command registers sequentially in its idle loop, looking for any GO bit. When GO is set, the microcomputer assumes that the host computer previously wrote the command code and command count into CAS and draws upon those parameters to set up the operation. Figure 1-17 shows the basic flow of execution for a motion command. The flow diagram does not cover the specifics of each type of motion command operation, but highlights the functions common to all.

The fifth process block (labeled *) in Figure 1-17 may represent as many as 17 different commands. For instance, the Write Tape Mark and Close File commands require both the read and write data paths. Tape mark information is created by the write path and verified by the read path. Spacing and erase gap operations involve only the read path, which provides status to the microcomputer as records, tape marks, and erased areas pass over the tape transport read head. Operations such as rewind and data security erase are executed by the microcomputer alone and do not involve the

read or write data paths. For a rewind operation, the microcomputer sends a Rewind command to the tape transport. Then, it polls one of the TU status bytes for BOT each time a pass through the idle loop is made. For a data security erase operation, the microcomputer sends Forward and Erase commands to the tape transport and polls for EOT status.

Figure 1-16 Subsystem Data Transfer Command (Sheet 1 of 4)

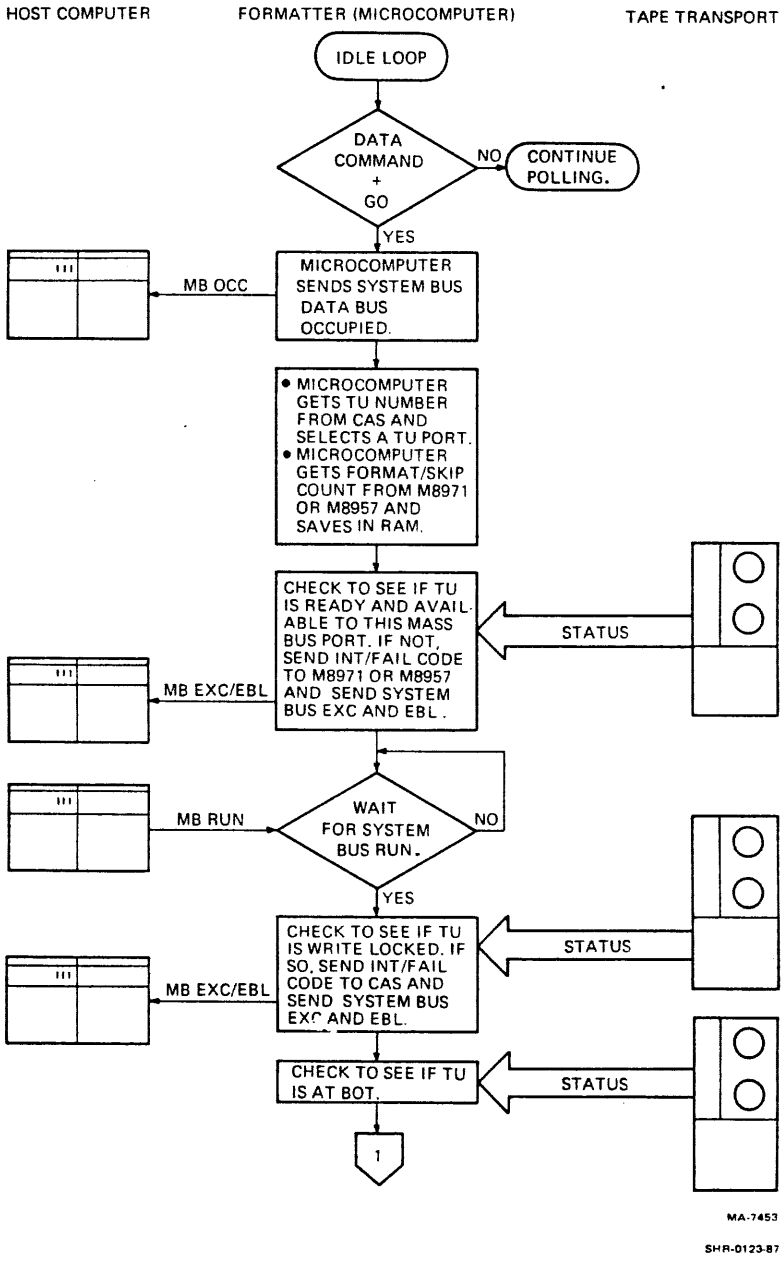


Figure 1-16 Subsystem Data Transfer Command (Sheet 2 of 4)

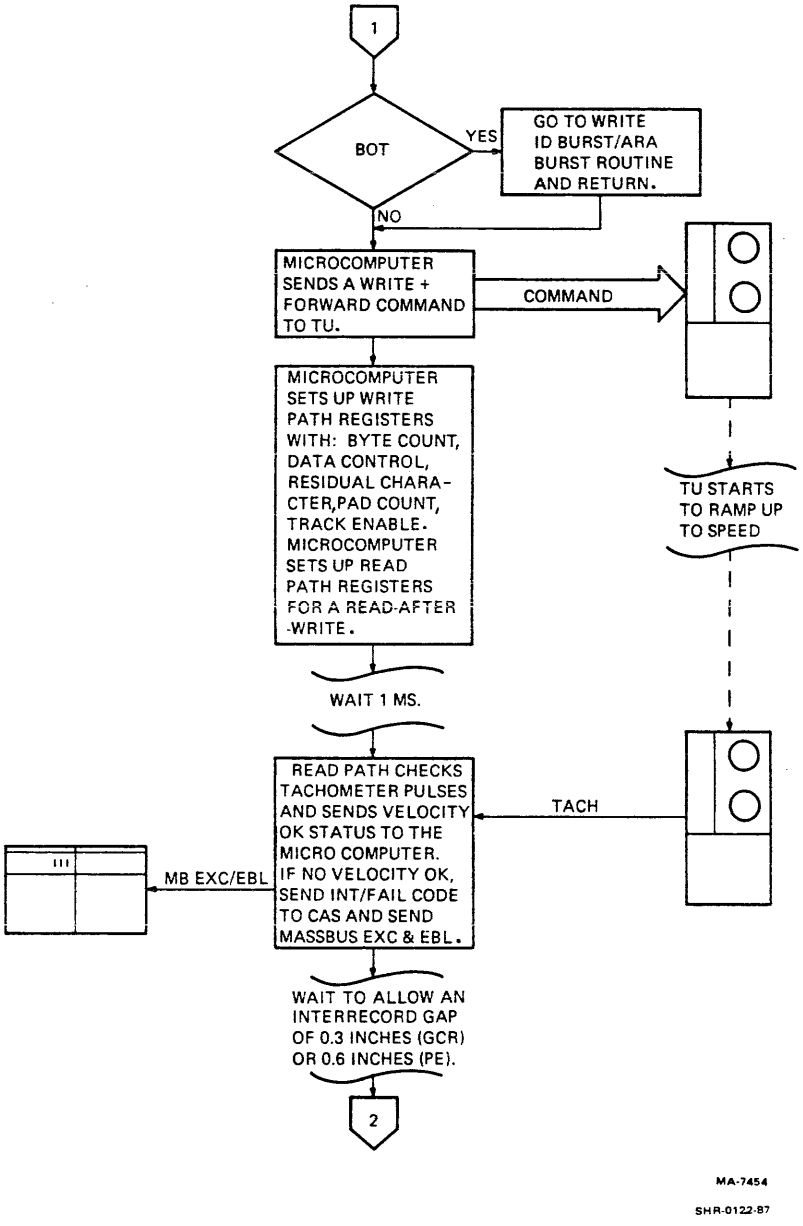


Figure 1-16 Subsystem Data Transfer Command (Sheet 3 of 4)

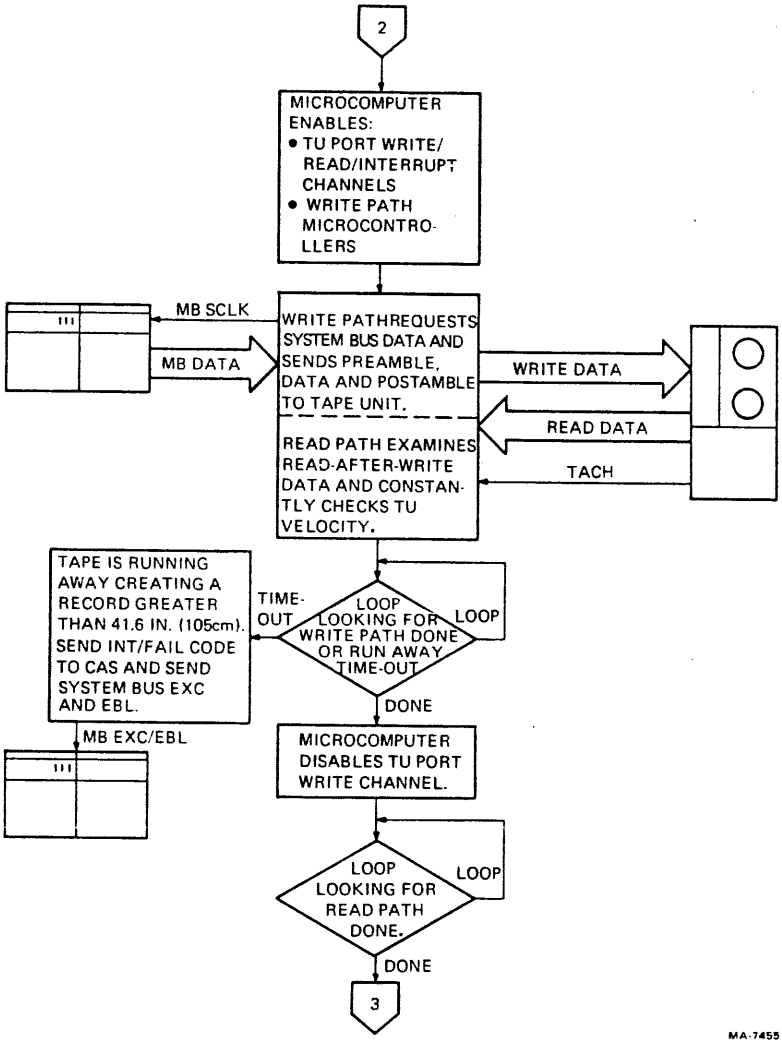


Figure 1-16 Subsystem Data Transfer Command (Sheet 4 of 4)

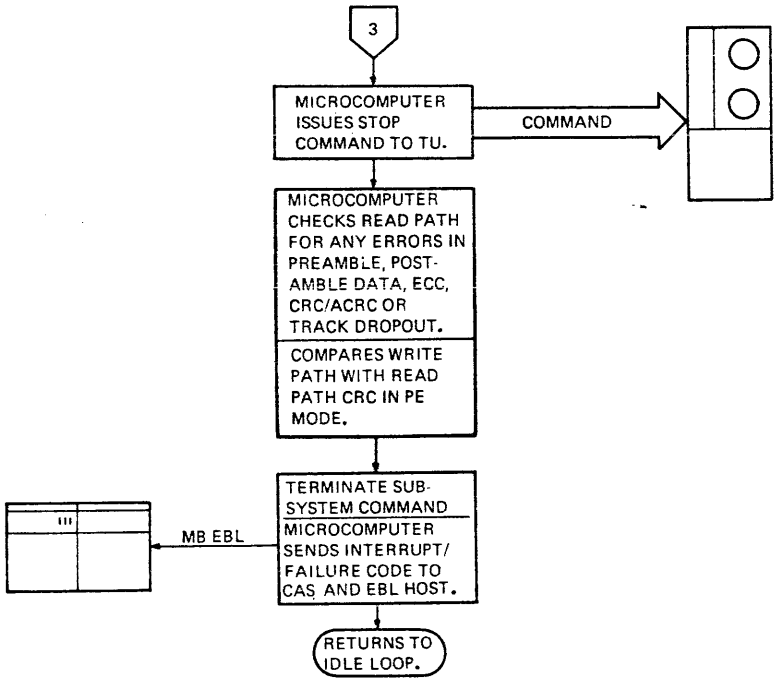


Figure 1-17 Motion Command (Sheet 1 of 2)

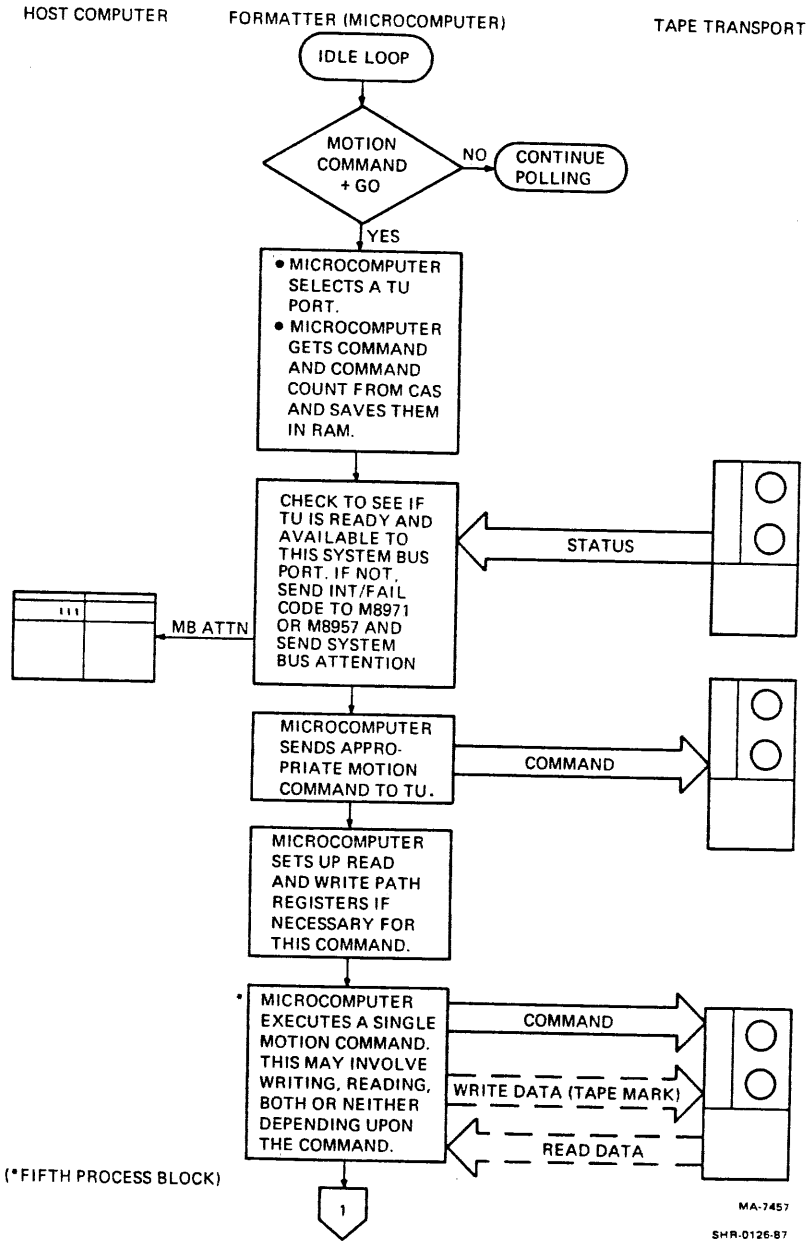
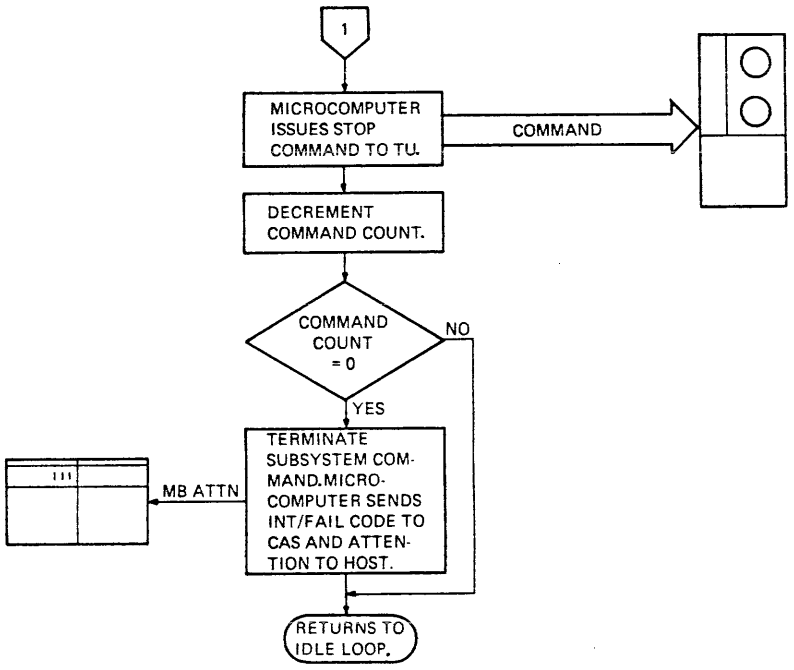


Figure 1-17 Motion Command (Sheet 2 of 2)



Sensing Operations - The formatter recognizes two types of sense commands, Sense and Extended Sense. Both commands provide information to the host computer about a specific tape transport. However, they are different in terms of how they are executed by the microcomputer, and when the information is sensed.

The Sense command is issued to the motion command registers and provides a limited amount of status to selected registers in the M8971 and M8957 modules.

When the microcomputer recognizes a Sense command: it samples some of the tape transport status registers; puts the appropriate information into registers 6, 7, and 10 on the M8971 and M8957 modules; and raises the xBUS ATTN line (where x is STI or MASS). This provides the host with information that is valid at the time the Sense command is issued.

The Extended Sense command, however, provides a great deal of sense information stored away in RAM, when an error occurred during a prior data transfer or motion command. It serves primarily for error-logging applications, where information pertaining to a failure in the subsystem or media must be analyzed. Figure 1-18 shows the events that take place in the formatter during an Extended Sense command. This command is sent to the data transfer command register and must be accompanied by the tape transport number in the CMD ADR field. It functions like the Read Forward command, except that related parameter information (format, byte count, etc.) is not necessary. When the Extended Sense command is recognized, the microcomputer sets up an internal byte count of 6010 and initiates a system bus data transfer. Sense data moves internally from RAM through the write path and out to the host computer, two bytes at a time, over the Massbus data bus. After thirty such transfers, the microcomputer raises the system BUS EBL line, signaling the command completion.

1.5.6.2 Write Path Description - The write path is defined as that portion of logic from where data enters the byte assembly logic to where it leaves the translator. This comprises the M8959 write microcontroller/byte assembly logic and the M8958 translator (Refer to Figure 1-19).

Figure 1-18 Extended Sense Command

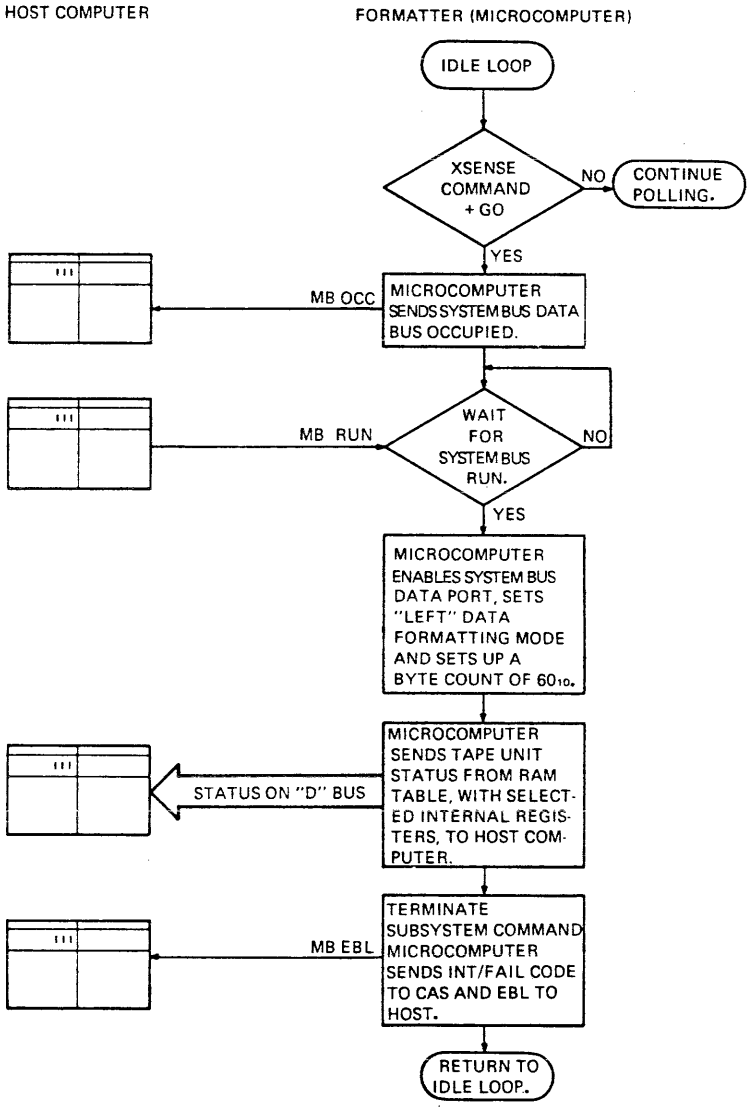
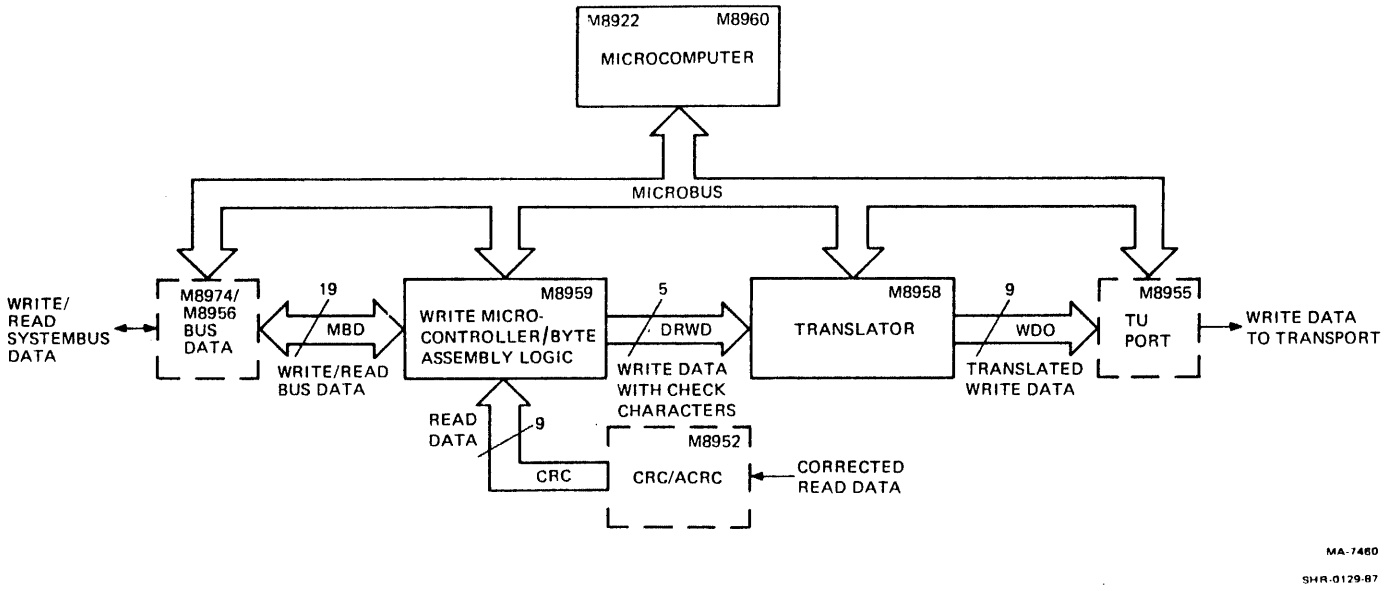


Figure 1-19 Basic Write Logic Block Diagram



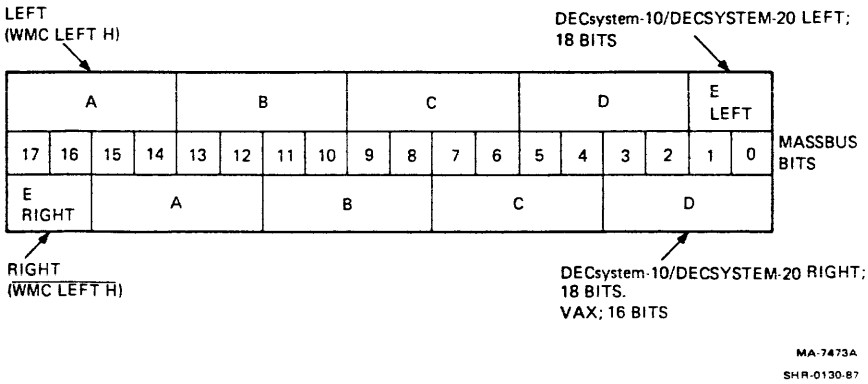
1.5.6.2.1 Overview of the Byte Assembly Process - Write data originates from the host computer system and passes over the system bus data bus to the formatter. On its way to the write path, the data passes through the STI bus interface module (M8971) or Massbus data module (M8956), where it is shifted left or right. The write microcontroller/byte assembly logic temporarily stores the system bus data words one at a time and breaks them down into 4-bit elements called nibbles. The translator requests and assembles one data subgroup consisting of eight nibbles. Then it either translates the data subgroup into a storage subgroup of five tape characters (GCR mode), or multiplexes the data subgroup into four tape characters (PE mode). In either case, the translator sends a 9-bit-wide stream of tape characters to the selected tape transport through the TU port module (M8955).

The process of requesting and assembling system bus data is under complete control of the write microcontroller module (M8959). It requests a data word by toggling the xbus sync clock (SCLK) line, and tells the STI bus Interface module (M8971) or Massbus data module (M8956) what data format (left/right) to latch into its nibble holding registers. Figure 1-20 shows the 18 Massbus data bits and how they correspond to a designated nibble (A through E). Note that the nibble groups align with different Massbus bits, depending on the state of the write microcontroller LEFT signal. When the host computer is a VAX, the write microcontroller specifies right format only; Massbus bits 15:0 are loaded into nibble registers A through D. Then A through D are sent one at a time to the translator as part of a data subgroup.

When the host computer is a DECsystem-10 or DECSYSTEM-20 the write microcontroller alternates between left and right formats, depending upon which half of the 36-bit system word will be transferred. In right format, system bits 18:35 correspond to Massbus bits 17:0 and load into one half of nibble E and nibbles A through D.

Then, as in PDP-11 operation, nibbles A through D go to the translator subgroup buffers. The write microcontroller specifies left format on the next Massbus word transfer and system bits 0:17 (Massbus bits 17:0) load into nibbles A through D and the other half of nibble E. Now nibble E is full and contains system bits 16:19. To fulfill the balance of the translator subgroup buffer, the write microcontroller sends nibbles E, A, B, and C to the translator. Nibble D forms the start of the next subgroup.

Figure 1-20 Massbus to Write Microcontroller Nibble

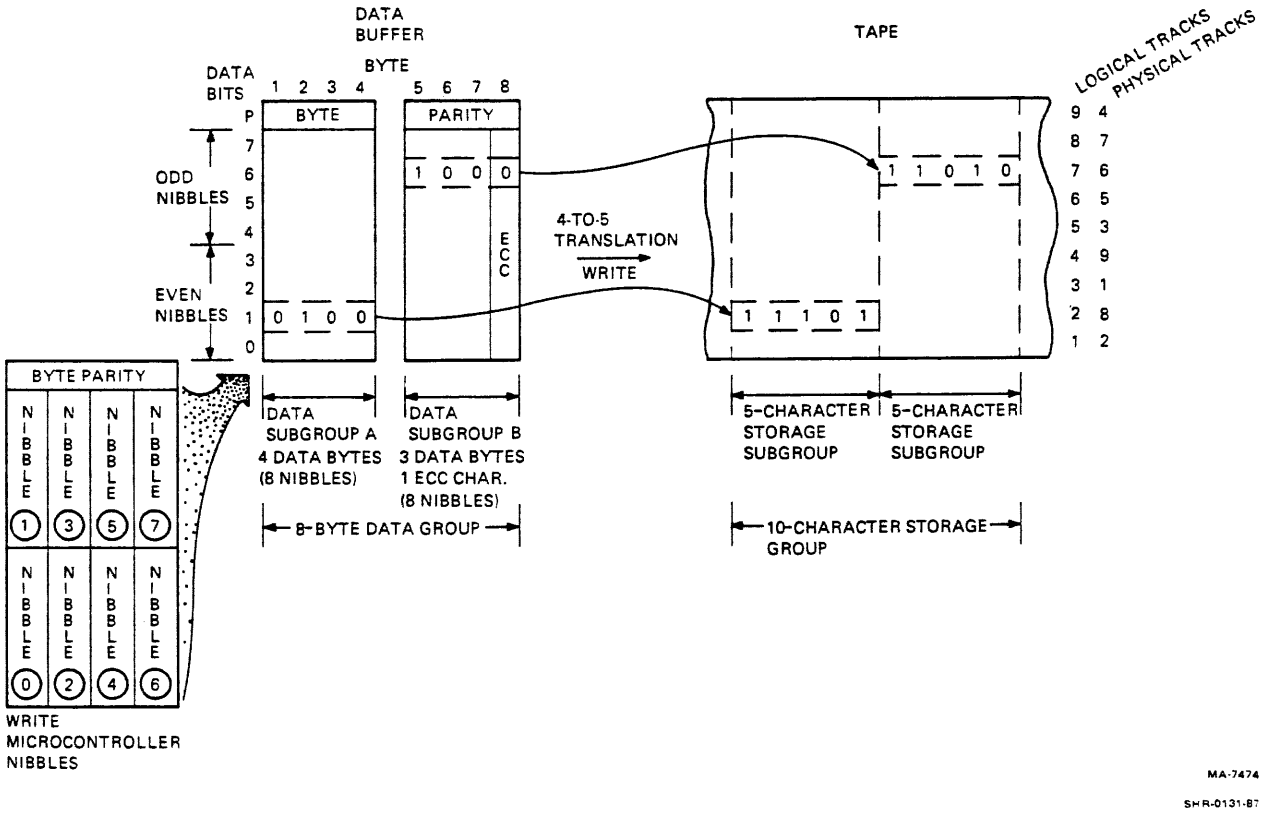


The preceding paragraph describes the order of events for subgroup A, the first of two subgroups in a data group. If it were subgroup B, the last two nibbles (7 and 8) would be an error correcting code (ECC) byte. The write microcontroller generates and sends the ECC byte as the last byte in data subgroup B in both PE and GCR modes. In PE mode, the translator simply ignores the ECC byte, processes the preceding three bytes (six nibbles), and requests the next subgroup. In GCR mode, both data subgroups A and B translate into storage subgroups. Figure 1-21 shows a graphic representation of how this translation takes place.

In summary, the write microcontroller sends data and check characters to the translator on a nibble basis. This continues till eight nibbles fill the translator subgroup buffer. The translator processes the data and sends it to the TU port on a tape character-by-character (9-bit) basis.

In addition to processing write data, the byte assembly logic handles read data from the CRC/ACRC module (M8952). During a read operation the read path outputs bytes of data (two nibbles) assembled in the nibble holding registers. When enough data is assembled, the write microcontroller initiates a system bus transfer to the host computer, again specifying left or right mode to the STI bus Interface module (M8971) or Massbus data module (M8956). Then, having emptied the nibble holding registers, it requests another series of bytes from the read path.

Figure 1-21 Translation and Physical Data Placement on Tape (Detail)



Write Path Operation - The write path consists of traditional logic elements such as AND/OR gates, registers, decoders, etc. However, much of the functional operation for processing a write command is delegated to microprograms that run in the system microcomputer, write microcontroller, and translator microcontroller. The following discussion describes the sequence of events that take place in the write path when the microcomputer determines that the host computer has issued a write command to the subsystem. Refer to Figure 1-22 for a graphic representation of the write path flow while reading this section.

At the start of a write operation, the write path is set up and commanded to write by the microcomputer. During the write operation the write path takes over and performs all the necessary data transfer handshaking with the host computer. It also determines the rate at which data is clocked out to the tape transport. After deciding that the host computer has sent a Write Data Transfer command, the microcomputer sets up certain registers in the write path. It calculates and sends the following parameters to the write microcontroller: a 16-bit byte count, an 8-bit data control word (format and skip count), and in GCR mode the residual character (modulo 7/32 counts) and the pad count (how many pad characters will be written in the residual data group).

Following this, the microcomputer sends a double word track enable parameter to the translator. This enables all of the write buffer registers at the output of the translation logic. Then the microcomputer enables the write path by sending a command word to the write microcontroller. The command word contains enabling bits for the write and translator microcontrollers, a density select bit, a direction bit, and a write bit. The density, direction, and write bits also go to the read path, so that the read-after-write process conforms to the selected parameters. The enable translator microcontroller and density select bits pass on to the translator module.

Now both write path microcontrollers are enabled to perform the write task and branch out of their idle loops, where they have been running internal diagnostic routines. The translator starts assembling and sending preamble characters to the TU port and requests the first data subgroup from the write microcontroller. The write microcontroller requests the first system bus data word from the host computer and starts to send data nibbles to the translator's subgroup buffer. When the translator finishes writing the preamble, it requests the second data subgroup. This request causes the first subgroup to transfer to a second level of buffering, while allowing the second data subgroup to shift into the first buffer. Now the translator either multiplexes (PE) or translates (GCR) the data, and gates it out to the TU port.

This handshaking process between the system bus, write microcontroller and translator microcontroller continues until certain counters overflow, depending upon the selected density mode. In PE mode, the write microcontroller signals the translator as it is sending the last byte of a record. When the byte counter overflows, the write microcontroller asserts the byte parity line during an even nibble transfer. (Normally the byte parity line transmits parity on odd nibble transfers.) The translator sees the byte count end status, multiplexes the last byte of data, and starts assembling and sending PE postamble characters to the TU port.

Figure 1-22 Write Path Operational Flowchart (Sheet 1 of 3)

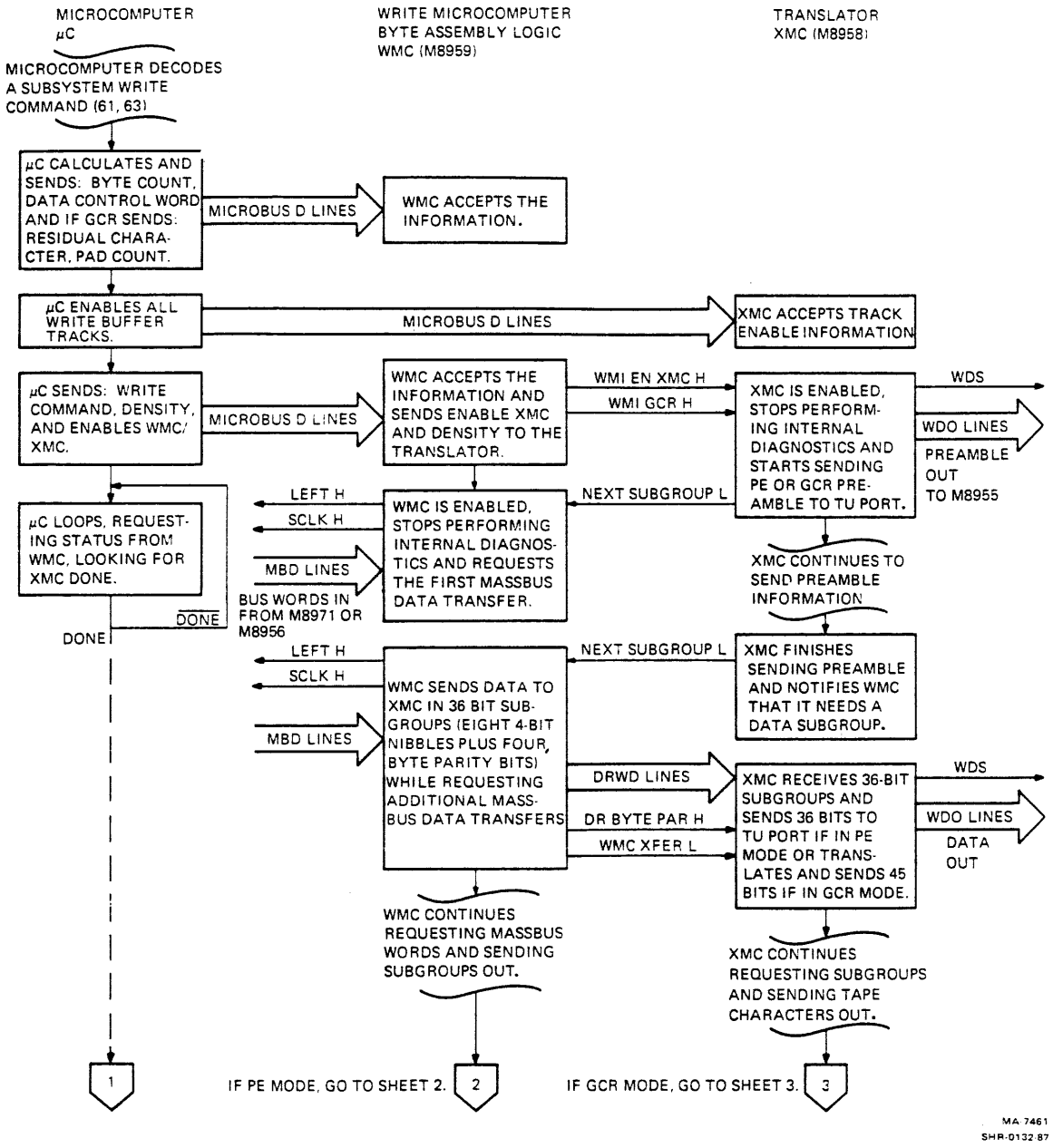


Figure 1-22 Write Path Operational Flowchart (Sheet 2 of 3)

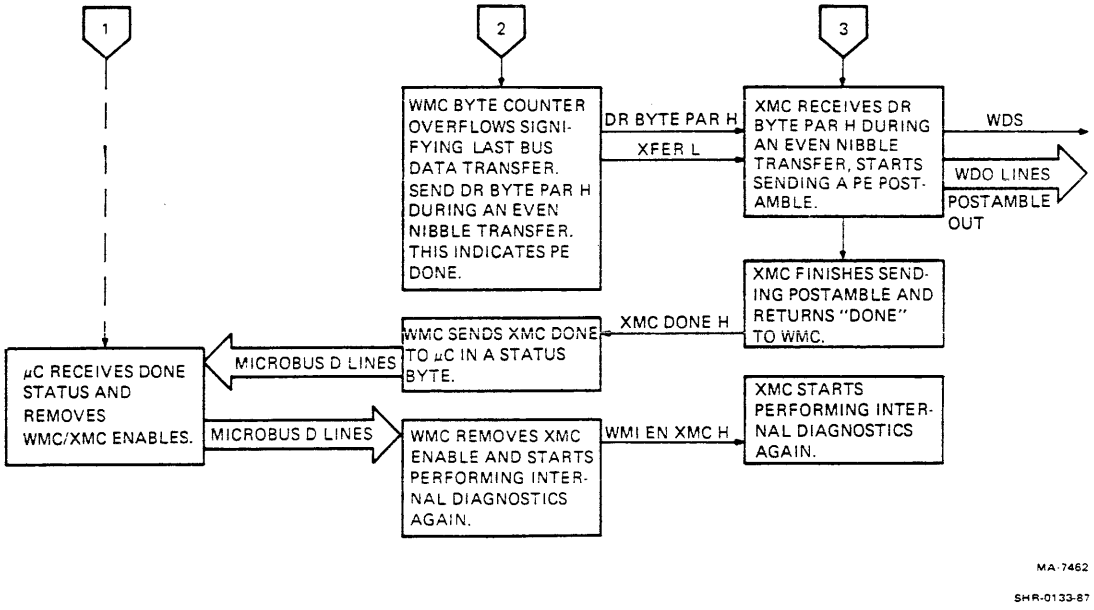
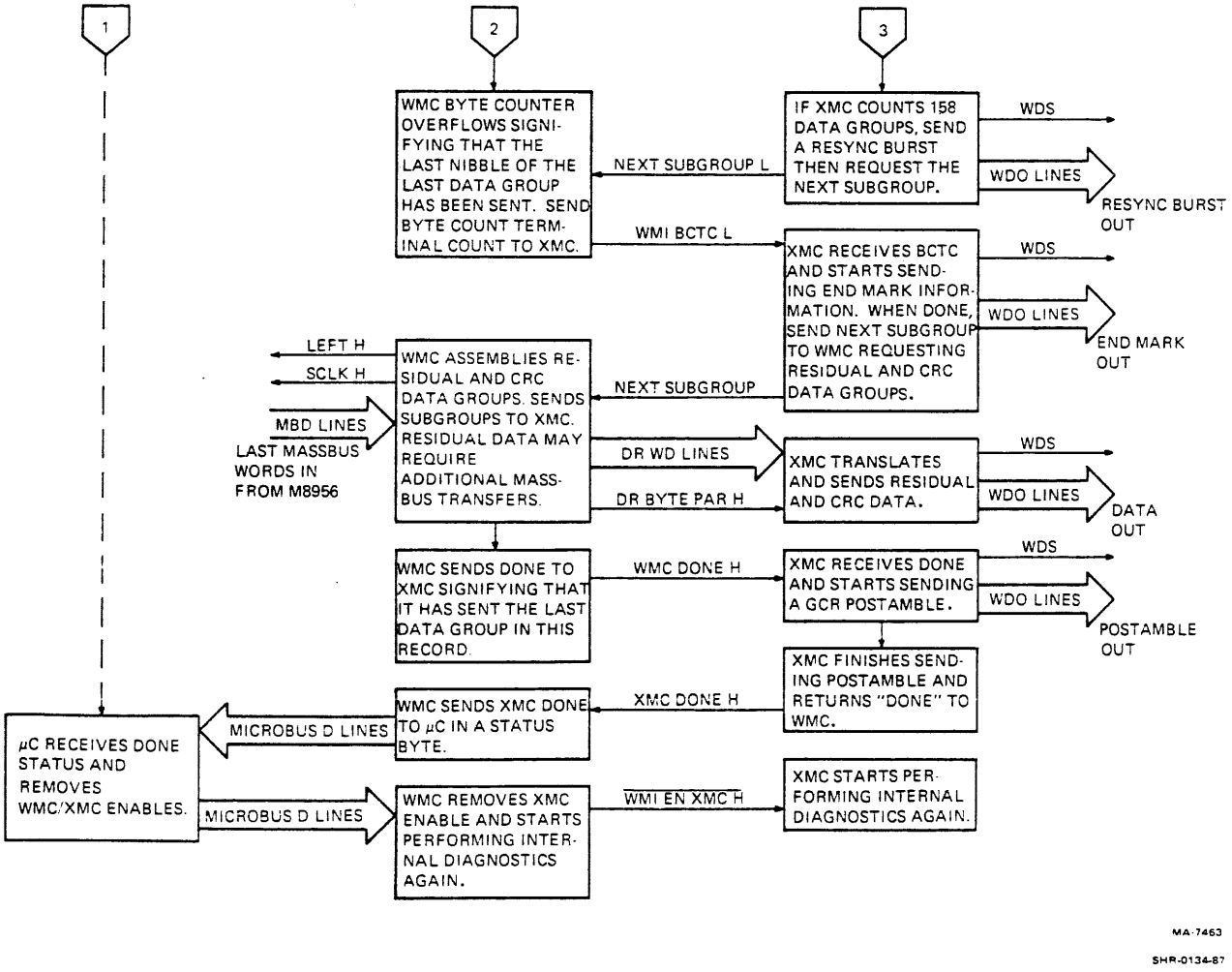


Figure 1-22 Write Path Operational Flowchart (Sheet 3 of 3)



In GCR mode the translator maintains a count of the number of data groups it processes. If the count reaches 1106_{10} data bytes, the group counter overflows and signals for a resync burst. After sending the resync burst, the translator reinitializes the group counter and requests the next subgroup. When the write microcontroller sends the last byte of the last data subgroup, it also sends the byte count terminal count to the translator. This tells the translator that it must translate the last data subgroup and send the end mark.

While the translator is doing this, the write microcontroller assembles subgroup A of the residual data group. If the byte count is some number other than a multiple of seven (seven data bytes to a data group), the residual data bytes must be requested and sent over the system bus. The translator continues requesting and translating subgroups for the residual and CRC data groups. When the write microcontroller sends subgroup B of the CRC data group, it also sends done status to the translator. This tells the translator that the write microcontroller will not send any more data or check characters, and that it must start to assemble and send the GCR postamble characters to the TU port.

The balance of the write operation is common to both density modes. As soon as the translator finishes sending the postamble, it returns done status to the write microcontroller. The microcomputer, which has been polling the write microcontroller status word, sees "done" and promptly removes the microcontroller enable bits. The disabled write path microcontrollers now return to their idle loop diagnostic routines. This concludes the write path operational flow.

1.5.6.3 Read Path Description - The read path is defined as that portion of logic from where data enters the read channels to where it leaves the check character logic. This comprises the read channel (nine modules), read path controller, error correcting code, and CRC/ACRC modules.

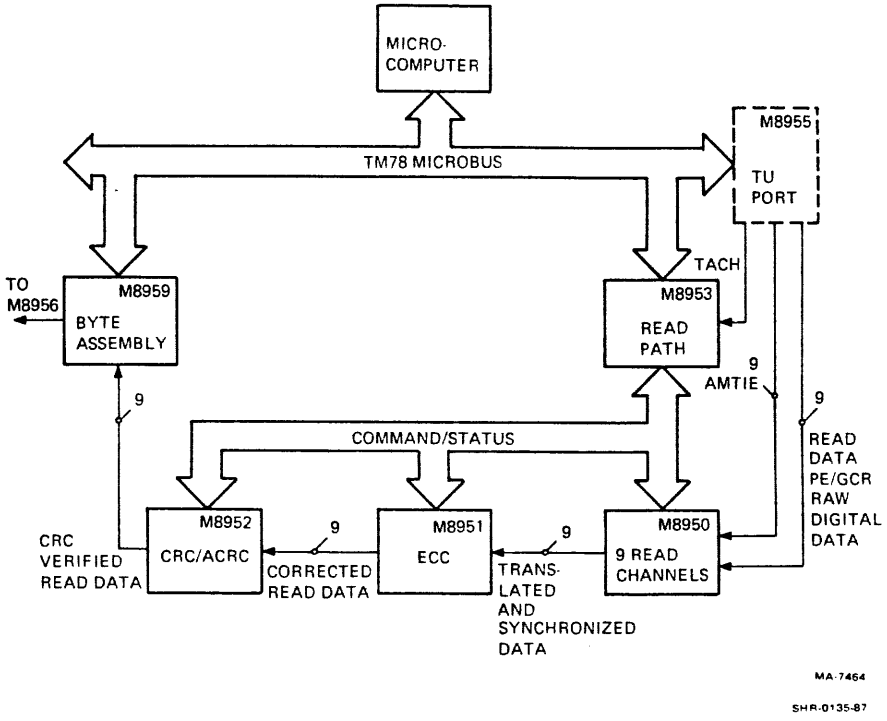
This section includes a variation of the diagram, highlighting the read logic. Refer to the basic module descriptions (as necessary) and Figure 1-23, the basic read logic block diagram, while reading the following paragraphs.

General - Read data originates from a prewritten magnetic tape passing over the tape transport's read head. The analog signal leaving the head must be amplified, conditioned, and converted to digital data by electronics within the tape transport. The digital data then transfers over the transport bus cable, through the TU port module (M8955), to the read channel modules (M8950).

An ID burst, near the beginning of the tape, identifies the format in which the data was written (PE ID burst — track 4, GCR ID burst — track 6). The read path determines which density ID is recorded as the tape moves off BOT. Amplitude track in error (AMTIE) lines handle envelope detection.

Digital data enters the read channel modules from the TU port module, where data decoding is performed using phase-locked loops (PLL). Then the read signals are fed to the first-in-first-out (FIFO) deskew buffer.

Figure 1-23 Basic Read Logic Block Diagram



In addition to deskewing data, the read channel modules provide initial error detection. This helps to detect bad (dead) tracks and develop an error pointer for error correction.

The read data leaves the deskew buffer and enters the translator logic. Here the 5-bit GCR encoded data is retranslated to its original 4-bit format. Translation is performed bit-by-bit, following a translation algorithm. Data then passes to the ECC module (M8951). If correctable errors exist (as determined by the error-correction algorithms), they are corrected by the error-correction logic. Next, the data passes to the CRC module (M8952) for CRC and ACRC check character verification before passing on to the byte assembly logic.

Data Paths - Nine channels of digital data enter the read channel modules from the TU port module. PLLs that were locked into each channel's preamble provide a window around the expected data transition time. The polarity of transition determines whether the bit is a zero or one in PE mode; the absence or presence of a transition determines the bit in GCR mode.

The PLL develops two signals to indicate signal quality to the ECC module. PHTIE (phase track in error) and FATAL. PHTIE becomes active whenever a data transition occurs too near either edge of the expected bit cell time. Data cannot be guaranteed if transitions occur during these periods. FATAL asserts when either a transition occurs within a six percent time at the beginning of the window, or two transitions occur during the window period.

For example, FATAL asserts if one read channel PLL is failing. The read channel where the signal originated shuts down, preventing it from interfering with the other read channels. Subsequent error-correction circuitry fills in the missing data.

Illegal 5-bit GCR codes detected by the read channel microcontrollers generate pointers to the ECC module (along with PHTIE and AMTIE) to aid in error correction. The read channel microcontroller is a ROM controller with a program counter and 512 words of 16-bit ROM. It performs the following functions.

1. Performs the 5-to-4 bit translation algorithm and detects illegal codes and control codes.
2. Controls the PLLs so that each track has its PLL locked in when its data appears. If FATAL asserts, the microcontroller attempts to lock up the PLL again; this allows the PLL to participate when a Resync burst occurs.
3. In PE mode, provides a pointer for the error-correction circuits, in case there is no transition within a track's bit cell or window.
4. In both modes, reports to the read path controller when all PLLs are synchronized.
5. "Knows" when data arrives by detecting the sync character (mark 1 in GCR, all ones in PE). In GCR, it also detects the other control characters (for example, mark 2).
6. Passes along the amplitude track in error (AMTIE) with the data to the ECC logic. The AMTIE signal comes from the transport, indicating a weak signal.
7. Stores statistical information about the record, for example:
    a. Did any illegal 5-to-4 translations occur?
    b. Did AMTIE or PHTIE ever assert?
    c. Did any pointerless errors occur that the ECC module still detected?
8. Sends status back to the read path controller.

The read path module controller (M8953) acts as a supervisor for all nine read channels, the ECC, and CRC modules, directing the jobs performed and sending status back to the microcomputer. Besides performing its own self-testing, this module's ROM controller also aids in the read channel self-testing. The read path controller also checks special records like tape marks and assures that the ARA ID is written accurately. The ARA character is approximately 5.08 cm (2 in) long. This recording burst comprises 1s in tracks 2, 3, 5, 6, 8, and 9 and erasure in tracks 1, 4, and 7. At least one 0.63 cm (0.25 in) section of the 5.08 cm (2 in) recording burst must be error free in all tracks at once.

Each read channel module has an 8-bit shift register that contains an entire data group's 5-bit storage data translated into 4-bit data. The data remains in the shift registers until it is ready in all nine modules. Then the ECC module receives the DATA RDY signal from the read path controller. This signal asserts at the beginning of each data group. Once the signal asserts, the ECC module shifts the data out to the CRC/ACRC module at a 10 MHz rate. As the data shifts out, the ECC module scans all nine tracks, one at a time, observing the TIE bus 4-bit code that indicates how bad a particular track was. For example, if both an illegal 5-to-4 translation and an AMTIE occur, it is worse than just an AMTIE error occurring. The 4-bit track in error (TIE) bus comprises (in descending order of value) an illegal 5-to-4 translation (definite data error), AMTIE, PHTIE, and a history (HIST) bit. The HIST bit indicates that nothing was noticeably wrong with the data, but earlier in the record something was wrong with it.

After shifting the data and TIE information, the ECC module notes the two worst pointers and their respective track numbers. The module's microcontroller program uses this information to determine if more than two, exactly two, one, or no tracks had pointers. The program then goes to one of two routines, the single- or two-track error-correction algorithm. The single-track error-correction algorithm has an advantage because it does not require pointers; however, it can only locate and correct one track in error. Pointers are required for the two-track error-correction algorithm.

The ECC module algorithms correct the data as much as possible. Data then passes into a FIFO on the CRC module. Also, some status bits are developed during error correction, namely: uncorrectable error, two-track error correction used, single-track error correction used, or a pointer mismatch problem occurred (i.e., wrong pointer for the word). This last problem might occur if the pointer indicated track 1, but the error was actually on track 2.

The CRC/ACRC module accepts data from the ECC module. Normal (for example, nonpad character) data enters both the CRC and the ACRC checkers. If the data is the residual character, the CRC logic holds it in a register and interprets it as follows: how many characters are data, how many are pad characters, how much of the residual group is pad, and how much is data. While the data transfers to the write microcontroller's byte assembly logic, this interpreted information is placed in the checkers. If the word does not transfer back to the byte assembly logic, the checkers will indicate BAD CRC and BAD ACRC.

Altogether, three 8-bit bytes of status-type information come from the CRC and ECC modules: the corrected data (used only by diagnostic programs); the CRC character read; and the word made up of pointer mismatch, uncorrectable error, two-track error correction used, single-track error correction used, CRC bad, ACRC bad, ECC ROM parity error, and AMTIE occurred during record. These bits determine the need to rewrite or reread the record.

That is how data travels from the TU port module to the byte assembly logic, which then sends it over the system bus to the host computer during a read operation. On a write operation (read-after-write) the same things happen, except the data does not transfer back to the CPU. The data travels all the way through to the CRC module.

but does not go out on the lines to the byte assembly logic. A read is retried only when the CRC or the ACRC (or both) do not match, or when the ECC has an uncorrectable error in GCR mode. In PE mode, too long a postamble voids a read.

Control and Supervisory Logic - The read path controller is the supervisor in the read logic, keeping track of the exact record location at all times, measuring the length and quality of the preamble/postamble, checking the quality of both the writing of tape marks and the information near BOT, and continually monitoring the tape velocity during write operations.

The read logic uses 12 ROM microcontrollers, 1 on each read channel, ECC, CRC/ACRC, and read path controller module. They all function in a similar manner. Basically, each is a program counter (PC) controller whose outputs drive ROM address lines. The ROM outputs (in all cases at least eight bits) connect to presets on the PC. The entire operation is based on each ROM's branch opcode output, gated with a true/false selected condition flip-flop signal: various conditions are selectable. A unique feature of this architecture is that the selected condition specified in the previous instruction is the one used. This procedure saves time: once the branching instruction starts, the fact that it will or will not branch is already known.

In summary, the read path controller is the supervisor of the read operation. Interfaced with the microcomputer, the read path controller sends demands to and receives status from the read channels. It can inspect all status information read by the read channels. It can set bits (used as internal flags), set an entire word at one time (for example, set a task code to be performed by the read channels), load a status register readable by the microcomputer, and preset a general-purpose counter that counts at a 10 MHz rate for a predetermined period. Also, the read path controller has a ROM to observe both AMTIE and illegal 5-to-4 lines.

The read path controller's ROM microcontroller has many programs. One is a check interblock gap program that verifies all nine tracks are inactive (erased). Observing the tracks every 100 ns, this program detects noise in the gap and then notifies the microcomputer. Other programs allow the read path controller to look for PE ID, GCR ID, ARA ID, ARA burst, and tape mark problems for write quality checking.

Additional ROM microcontroller programs perform the following functions.

- Read.
- Perform self-tests.
- Test all nine read channel modules.
- Detect PE or GCR mode by reading the ID burst.
- Check status registers (diagnostic commands).
- Force a read operation (diagnostic commands).
- Clear everything that has a clear in it. (This program usually runs prior to diagnostics. It helps with the ECC self-test by timing out the test period and running the command lines to tell the ECC when to run the test.)
- Continually observe the AMTIE lines. This AMTIE WATCH program can space over pieces of records without losing the tape position.

1.6 UNIT-LEVEL DESCRIPTION

The following paragraphs describe the formatter logic circuits and power supply. They also discuss the TU bus protocol and characteristics. The text provides the most detailed technical description in this manual. Flowcharts of microcode/hardware tasks appear with their corresponding function. Individual flowchart blocks do not necessarily represent firmware instructions or groups of instructions.

1.6.1 Tape Unit Bus Description

The tape unit (TU) bus is a radial bus, with one tape unit per cable set. A cable set consists of two physical cables running between the formatter and a tape transport. A single formatter can drive up to four tape transports; thus a full configuration requires four independent radial TU buses.

A single TU bus consists of 33 active signal lines. Figure 1-24 shows the TU bus signals and the direction(s) in which they assert. Table 1-4 summarizes the TU bus signals and their functions.

Figure 1-24 Tape Unit Bus Signals

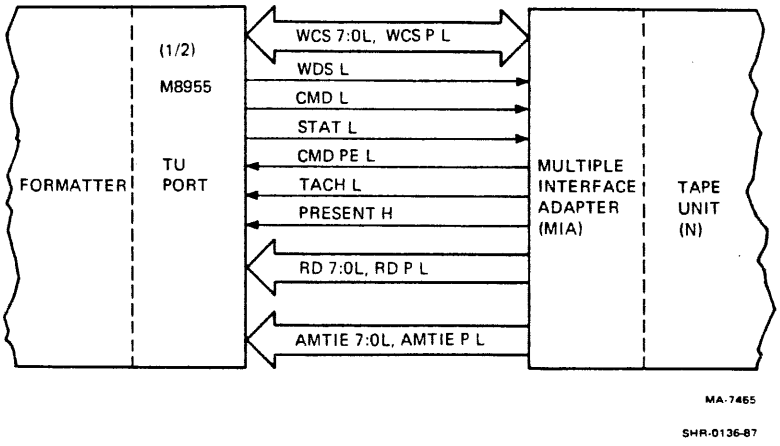


Table 1-4 TU Bus Signals

Signal	Function
WCS 7:0L	These eight write/command/status lines are bidirectional and multiplexed for the three different functions. Information on the WCS lines is determined by the assertion of the WDS, CMD, or STAT control lines. The WCS lines are asserted by the formatter during a write data or command transfer, and asserted by the tape transport during a status transfer.
WCS P L	This single, bidirectional, multiplexed WCS line carries the odd parity bit for command/status transfers, and the parity channel for a write data transfer. WCS P is asserted by the formatter during command or write data transfers, and asserted by the tape transport during status transfers.

Table 1-4 TU Bus Signals (Cont)

Signal	Function
WDS L	The write data strobe line indicates that the WCS lines contain data to be written on tape. WDS is asserted by the formatter.*
CMD L	The command line indicates that the WCS lines contain a command byte. CMD is asserted by the formatter.*
STAT L	The status line enables the tape unit to place a status byte on the WCS lines. STAT is asserted by the formatter.*
CMD PE L	The command parity error line indicates that a CMD/STAT address or command byte has been received by the tape transport with even parity. CMD PE is asserted by the tape transport. This line remains asserted until a clear command is received with odd parity.
TACH L	The tachometer line reflects the digital output of the tape transport's capstan servo motor tachometer.
PRESENT H	The present line is asserted by the tape transport when it is connected to the TU bus and has power applied.
RD7:0 L, RD P L	The nine read data lines are asserted by the tape transport when read enabled. They may also be asserted during the tape transport's loop write-to-read diagnostic mode.†
AMTIE 7:0 L, AMTIE P L	The nine amplitude track in error signals may be asserted by the tape transport when read enabled. They each indicate that the read back amplitude on the track for which they are named has not come up to a predetermined threshold value. They may also be asserted during the tape transport loop write-to-read diagnostic mode.†

* The WDS, CMD, and STAT lines are mutually exclusive. That is, only one of the three (indicating one of three types of information on the WCS lines) may be logically true at any time.

† The RD and AMTIE lines may be disabled at the transport by writing a zero into bit 6 of command address 3 (the read enable bit of the threshold command). Refer to Appendix C for an explanation of the read enable bit. The RD and AMTIE lines are disabled during the TU port's loop write-to-read diagnostic mode.

1.6.1.1 Electrical Characteristics - TU bus communication uses two 40-conductor cables. The cable pair running between the formatter and each tape transport cannot exceed 15.25 m (50 ft). The 33 signal lines make up 33 signal pairs. Figure 1-25 shows a typical signal pair and its termination network. Figure 1-26 shows the PRESENT H signal line, which is always asserted when a tape transport is connected to the TU bus and has power applied.

All TU bus signal functions (except for PRESENT H) are asserted when the bus line is at a logic low level. The lines are driven low at the source end by open collector type drivers; bidirectional WCS lines may be driven from either end. Typical voltage levels are indicated on each figure.

Figure 1-25 Tape Unit Bus Signal Line Pair

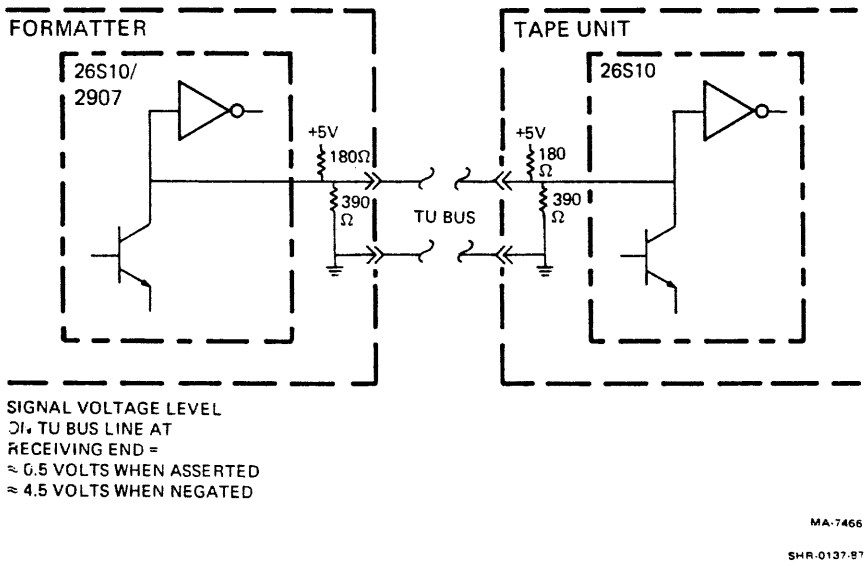
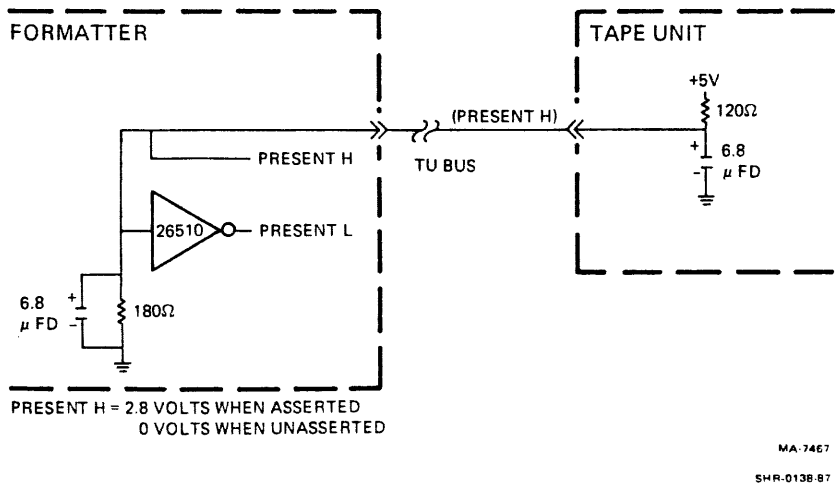


Figure 1-26 Tape Unit Bus Present H Signal Line



1.6.1.2 Timing - The microcomputer's operating system monitor establishes the TU bus protocol. The microcomputer is the single decision-making element for issuing commands, requesting status, and enabling read/write gates in the TU bus/TU port configuration. It also provides all critical TU bus timing relationships.

The three major TU bus components are the write/command/status (WCS) lines, the read lines, and the AMTIE lines. The WCS lines are synchronous with the microcomputer clock and the microbus; the read and AMTIE lines are asserted relative to the speed and quality of data being read from tape.

The multiplexed WCS lines have the strictest timing considerations of these components. In a typical subsystem write data operation, the microcomputer requires (and controls) the bidirectional WCS lines to transfer information in the following order. (Parentheses indicate microbus transfer only — not WCS.)

- (Microcomputer requests port status.)
- Microcomputer requests status byte from transport.
- (Microcomputer issues command byte to port.)
- Microcomputer issues multiple command bytes to transport.
- (Microcomputer enables port write and read gates.)
- (Write data transfers from the formatter to the transport.)
- Microcomputer requests multiple status bytes from transport after completion of write operation.

Figures 1-27 and 1-28 show the timing relationships between the instruction issued to the TU port/TU bus by the microcomputer (command or status request) and the subsequent communication over the WCS/control lines to the tape unit. Figure 1-27 shows the timing relationship between the formatter microbus and the TU bus WCS lines during a command transfer. This figure also shows two TU magnetic interface adapter (MIA) signals (Command Strobe one-shot and Execute one-shot) which strobe the command from the WCS lines and cause it to execute. Figure 1-28 shows the timing relationship between the formatter microbus and the TU bus WCS lines during a status request. Each figure lists the signal name and the source in parentheses.

The third function of the WCS lines is to transmit write data from the formatter to the tape transport. During a write data transfer, no WCS/microbus handshaking takes place. This is because all write data signals (nine parallel bits per character) and the Write Data Strobe pulse come from the formatter write path logic. The TU port/TU bus WCS lines act simply as an open channel during the actual data transfer.

Figure 1-27 Tape Unit Bus Command Transfer Timing

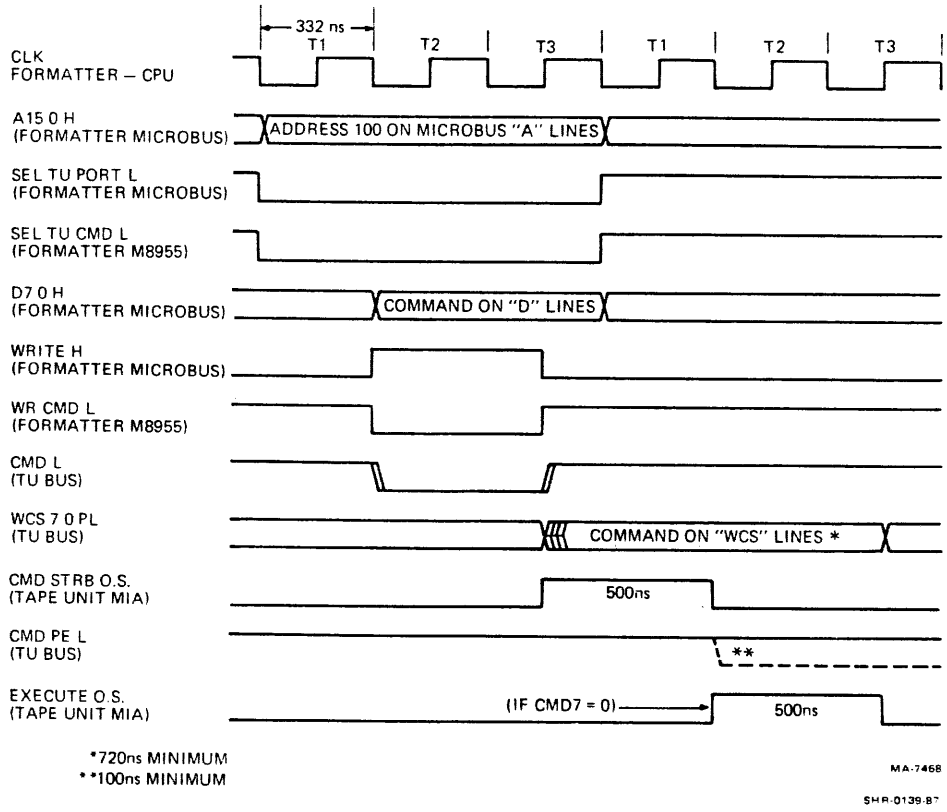
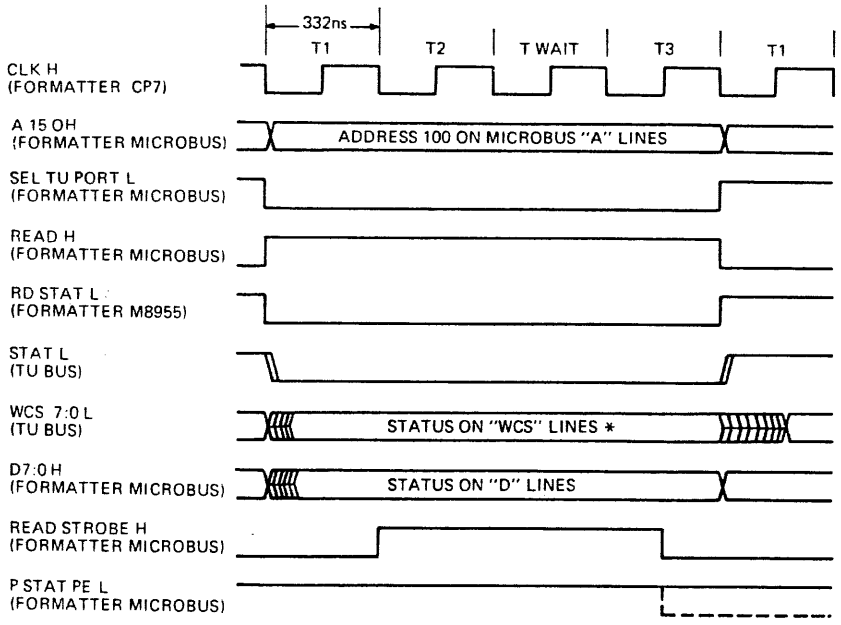


Figure 1-28 Tape Unit Bus Status Transfer Timing



* THE STATUS INFORMATION ON THE WCS LINES WILL NOT CHANGE DURING THE REQUEST CYCLE EVEN THOUGH TRANSPORT STATUS MAY CHANGE SUDDENLY.

MA-7469

SHR-0140-87

CHAPTER 2
FORMATTER TECHNICAL DESCRIPTION

2.1 INTRODUCTION

This chapter provides a brief functional description of each of the formatter modules. Then, each module is described in detail including the circuit modules, the maintenance panel, and the power supply systems. All descriptions are supported by detailed block diagrams.

2.2 FUNCTIONAL DESCRIPTION

The following modules are briefly described on a functional level (Refer to Figure 2-1).

STI Microcomputer (M8972)
STI Protocol Microcontroller (M8970)
STI Interface (M8971)
Extended Memory (M8973)

Massbus Microcomputer (M8960)
Common Address Space (M8957)
Data Bus Interface (M8956)

Write Microcontroller (M8959)
Translator (M8958)
Tape Unit Port (M8955)
Read Channel (M8950)
Read Path Controller (M8953)
ECC Controller (M8951)
Cyclic Redundancy Checker (M8952)

Figure 2-1 Magnetic Tape Subsystem Functional Block Diagram (Sheet 1 of 2)

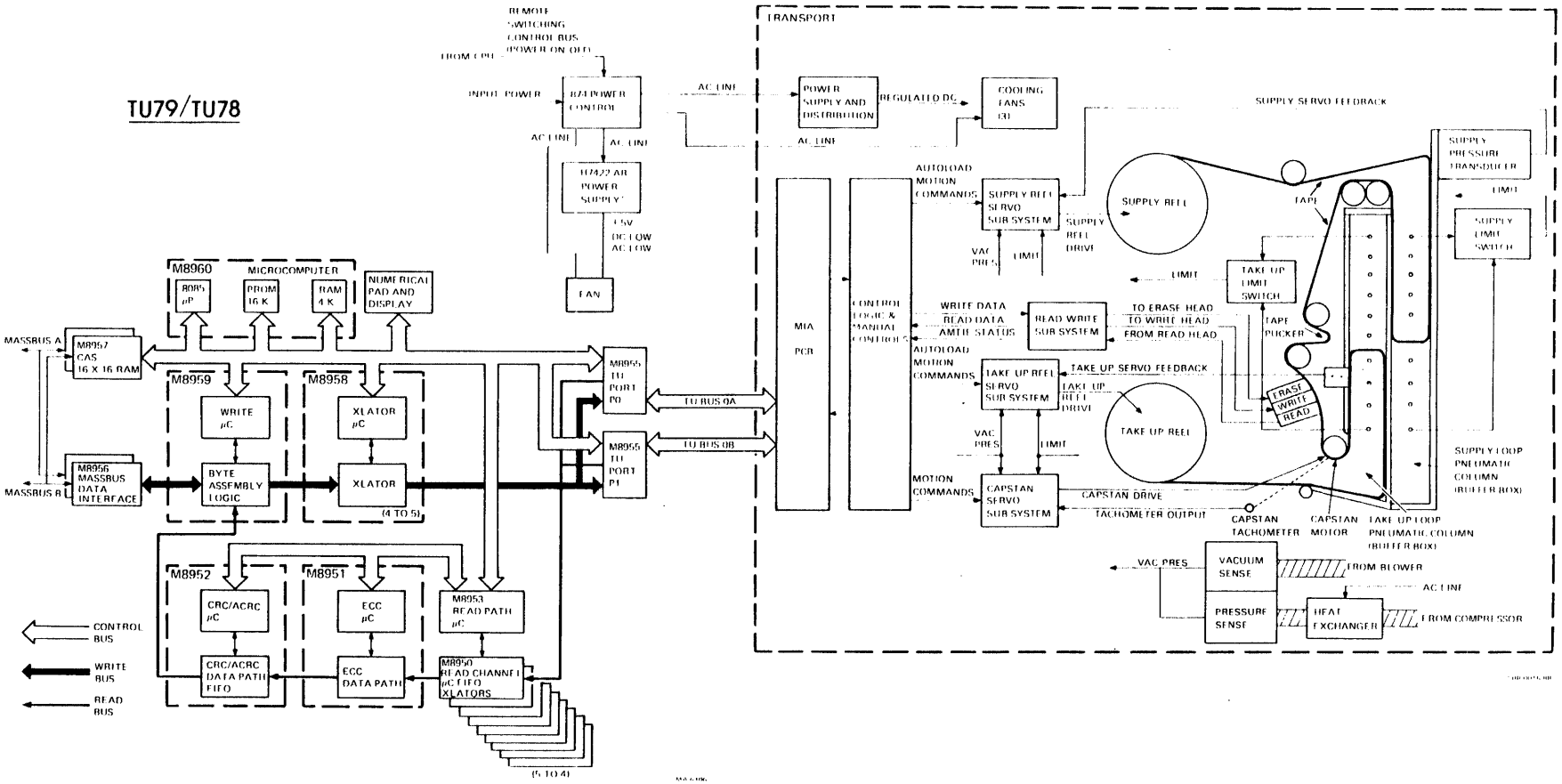
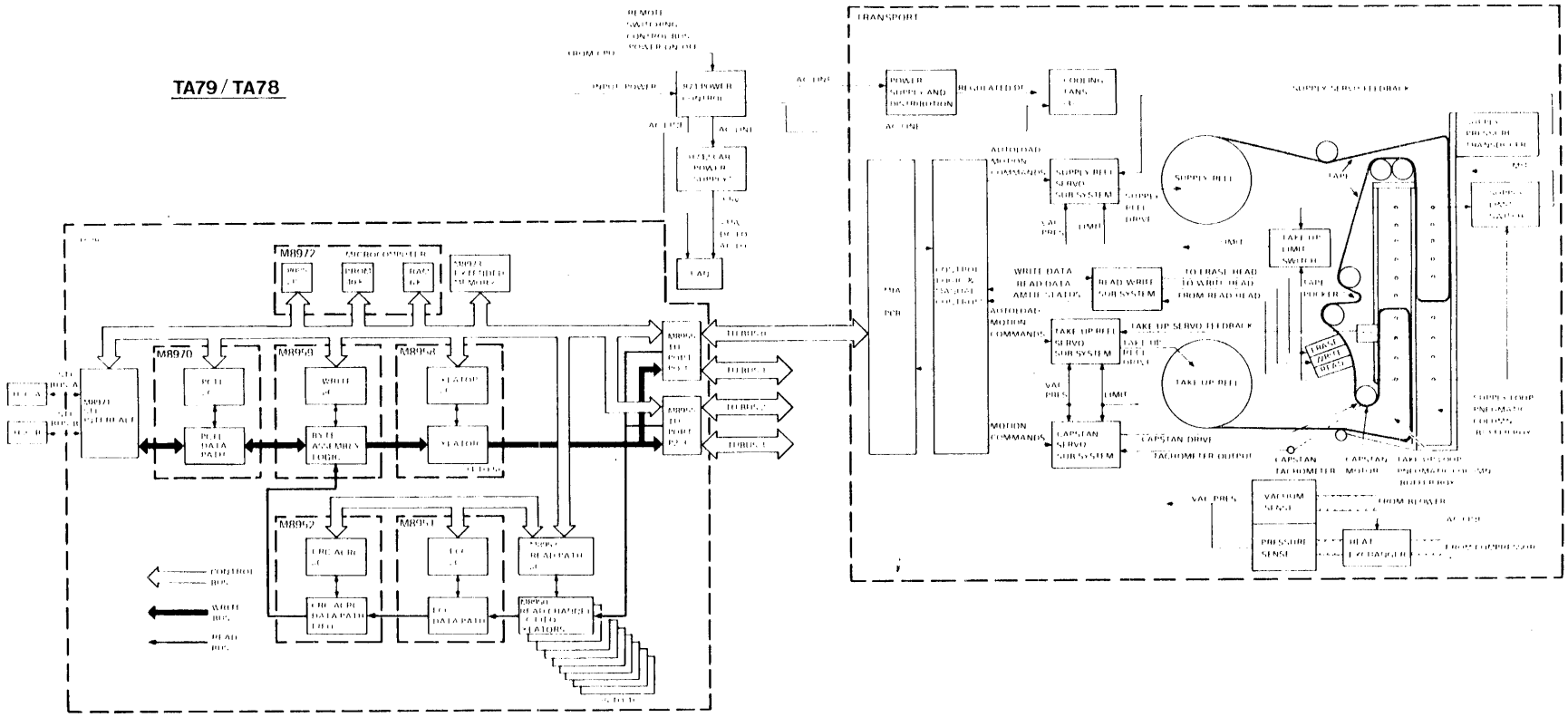


Figure 2-1 Magnetic Tape Subsystem Functional Block Diagram (Sheet 2 of 2)



Write Microcontroller (M8959)

This module is a ROM microcontroller dispatched on various tasks and monitored by the system microcomputer. In addition to allowing diagnostic data injection to the write logic during diagnostic testing, the write micro portion of the module functions in both write and read modes.

Write Mode

1. Receives data from subsystem bus (STI or Massbus) data module.
2. Disassembles data for translator in one of seven formats.
3. Checks data parity from subsystem bus (STI or Massbus).
4. Generates data parity to translator.
5. Performs CRC character calculation.
6. Performs ECC character calculation (CR).
7. Performs ACRC character calculation (GCR).
8. Formats residual and CRC data groups (GCR).

Read Mode

1. Receives data from read logic.
2. Assembles data for subsystem bus (STI or Massbus) in one of seven formats.
3. Generates data parity to subsystem bus (STI or Massbus).
4. Checks data parity from read logic.
5. Facilitates data byte skip count.

Translator (M8958)

This module is a ROM microcontroller initialized by the system microcomputer. Once initialized, it is controlled by the write micro portion of the M8959 module. The translator contains a microprogram ROM, subgroup buffers, translation ROMs, and control logic needed to translate 4-bit data bytes into the 5-bit characters needed in GCR formatting. In general, during a GCR write operation, this module performs the following functions.

1. Generates preamble and postamble.
2. Performs 4-to-5 translation.
3. Generates Resync burst.
4. Generates End Mark.
5. Controls flux reversal rate.
6. Provides track enable/disable functions.
7. Writes all ones.

The module also performs the following functions during PE write operations.

1. Generates preamble and postamble.
2. Controls flux reversal rate.
3. Provides track enable/disable functions.

Microcomputer (M8972 and M8960)

This module uses an 8085 microprocessor chip to act as the heart of the formatter control bus (microbus). PROM space contains the operating system and diagnostic monitors. Specific diagnostic routines are loaded from the host computer system into 4K of additional RAM spaces before they are run.

The microcomputer directs formatter operation over the TS78 microbus. It interfaces with the host CPU through the STI bus data module and the common address space (CAS). The microcomputer's addressable memory space extends to the CAS. In this context, the host CPU and microcomputer form a tightly coupled asymmetrical multiprocessor system.

Tape Unit Port (M8955)

This module interfaces the formatter with the tape transport. Two tape unit port modules may be used in each formatter, each module interfacing with two tape transports. During a write operation, the tape unit port module accepts either GCR- or PE-formatted data from the M8958, and transmits it to the tape transport to be written onto tape. Under complete microprogram control, bits in a TU command word inform the tape transport the format of the preceding data.

During formatter read or write operations, the tape unit port module accepts either GCR or PE data. This time the data is sent to the formatter from the tape transport. Again, the TU command word defines the data format.

This module also performs the following functions when the formatter is running in diagnostic mode.

1. Loops translator to microcomputer.
2. Loops translator to read path.
3. Loops microcomputer to read path.
4. Loops transport command register to transport status register.
5. Loops microcomputer to amplitude track in error (AMTIE) lines.
6. Loops AMTIE lines to microcomputer.
7. Loops read data to microcomputer.

Read Channel (M8950)

Nine read channel modules are used in the formatter, one for each tape track. Read channels are ROM microcontrollers, dispatched and monitored by the read path microcontroller (M8953). Read information feeds through the read channels, read data path, byte assembly, and data bus interface to the STI bus, and on to the CPU. During the read-after-write portion of the write cycle, information goes only as far as read data path logic, where ECC and CRC/ACRC checking are performed.

The basic functions of the M8950 module during PE read/write operation are as follows.

1. Searches for preamble.
2. Passes data to ECC.
3. Deskews data.
4. Performs read reverse correction.
5. Verifies PE flux reversal.

The main function of the module during GCR read/write operation is to retranslate the 5-bit GCR-formatted data into its original 4-bit format. Additional functions are as follows.

1. Searches for Mark 1.
2. Passes data to ECC.
3. Deskews data.
4. Performs read reverse correction.
5. Performs AMTIE signal detection.
6. Performs TIE pointer generation.

Read Path Controller (M8953)

This module is a ROM microcontroller, dispatched and monitored by the system microcomputer. The read path dispatches and controls the nine read channel modules (M8950), and provides control information for the microcontrol portion of the ECC module. This module also serves as a central area of status reporting for the entire read electronics (read channels, ECC, and check character microcontroller parts).

The basic module functions for the respective formatter operations are as follows.

Write PE

1. Tests IRG (interrecord gap).
2. Tests ID burst.
3. Tests tape mark.
4. Performs read-after-write.

Write GCR

1. Tests IRG.
2. Tests ID.
3. Tests tape mark.
4. Tests ARA ID.
5. Tests ARA burst.
6. Performs read-after-write.

Read GCR

1. Reads GCR forward.
2. Reads GCR reverse.
3. Samples density at BOT.
4. Detects ARA ID.
5. Detects tape mark.

Read PE

1. Reads PE forward.
2. Reads PE reverse.
3. Samples density at BOT.
4. Detects tape mark.

Diagnostic Mode

1. Self-tests diagnostic.
2. Performs status register test.
3. Loads/reads FIFO data.

ECC Controller (M8951)

The error-correcting code (ECC) controller module is a ROM microcontroller initialized by the system microcomputer, and thereafter controlled by the read path (M8953). This microcontroller implements the error-correction algorithms in both GCR and PE data formats, and in both read and write modes.

Read PE

1. Performs single-track error correction.

Write/Read GCR

1. Performs single-track error correction always.
2. Performs double-track error correction with pointers.

Cyclic Redundancy Checker (M8952)

The cyclic redundancy checker (CRC and ACRC) module is a ROM microcontroller initialized by the microcomputer, and thereafter controlled by the ECC controller (M8951). This microcontroller performs check-character algorithms in both read and write modes, PE and GCR data formats.

Write GCR

1. Verifies CRC and ACRC check characters.

Read GCR

1. Verifies CRC and ACRC check characters.
2. Passes data to byte assembly logic (M8959).

STI Protocol Microcontroller (M8970)

The M8970 handles protocol characters and performs data conversions involving message frames and data on the STI bus data lines. The M8970 also generates a 16-bit EDC character for validating data, and generates clock signals for the M8971 that provide the basis for timing on the STI bus data lines. In accordance with the M8972, the M8970 provides port control signals and specifies the formatter receiver ready state bit, for the M8971. The M8970 also provides a series of MCLK interrupts used by the M8972 to implement certain formatter timing functions.

STI Interface (M8971)

The M8971 handles waveform processing for the port A and B STI bus lines. It also handles data conversions involving information on the STI real-time state lines. In addition, the M8971 includes STI error logic and four register/buffer locations in the microprocessor I/O address space. Three of these locations involve data conversion and error identification. The fourth location, the Light (LITO) register, controls the three indicators on the TS78 control panel in accordance with the M8972.

Extended Memory (M8973)

The M8973 contains 56K of read-only memory (ROM), which stores 38 off-line microdiagnostic tests (that is, all off-line tests except Pass and Fail). The M8973 also contains jumpers to specify the hardware revision level.

The memory consists of seven 8K × 8-bit chips with tri-state outputs. The M8973 also contains a socket reserved for future use by an eighth chip. The M8972 accesses the contents of the extended memory through I/O locations using three data transfers. Two transfers set up the extended memory address and the third transfer reads the memory data.

Common Address Space (M8957)

The M8957 common address space (CAS) module is the element of the Massbus port that enables Massbus control bus (CBus) communication between the host computer and the TM78 formatter. It contains circuitry that stores command/status information and handles Massbus CBus protocol independently, without the aid of the microcomputer (M8960).

Data Bus Interface (M8956)

The M8956 Massbus Data module interfaces the TM78 read/write data paths to the Massbus controller. It also contains circuitry to interface the Massbus DBus/CBus control lines to the microcomputer (M8960). During a write operation the Massbus data module receives 18-bit data inputs from the Massbus and transfers all 18 bits to the write microcontroller module (M8959). The particular Massbus controller used may in fact transfer either 16 or 18 bits across the Massbus, but the Massbus data module does not make this distinction. During a read operation the Massbus data module receives 16- or 18-bit data inputs from the write microcontroller module and transmits them to the Massbus controller.

2.3 STI BUS INTERFACE

2.3.1 M8972 Microcomputer

The M8972 module is a completely self-contained microcomputer that supervises all formatting and data handling functions of the TS78. The M8972 also generates the read path and write path clocks, and it contains the ASCII port interface logic allowing an external terminal to access the tape unit for maintenance purposes. Figure 2-2 is a block diagram of the M8972.

The M8972 communicates with the rest of the formatter by means of the microbus. It reaches the STI bus(es) through the M8970 and M8971, and reaches the transport(s) through the M8955s.

The M8972 sees the entire tape drive as a series of I/O registers that can be accessed by means of the microbus. The M8972 issues commands with write data transfers and monitors system status with read data transfers. In addition, the M8972 services certain logic sections and error conditions on an interrupt basis.

The following paragraphs discuss the five functional elements of the M8972. Data transfer implementation and interrupt handling are also discussed.

Central Processing Unit

The CPU includes a single-chip 8085 microprocessor, several buffers/registers, and associated gates. The 8085 uses 8-bit parallel construction. It executes instructions from ROM/RAM storage elements at a 1.3 us cycle time. It can directly access a 64K memory address space or 256-byte I/O address space. And it has an integral system controller interfaced to the microbus through an external buffer. The 8085 also has five interrupt inputs. Table 2-1 describes each functional I/O pin on the 8085, and how it applies to the microcomputer and the TS78 formatter system.

Figure 2-2 M8972 Block Diagram

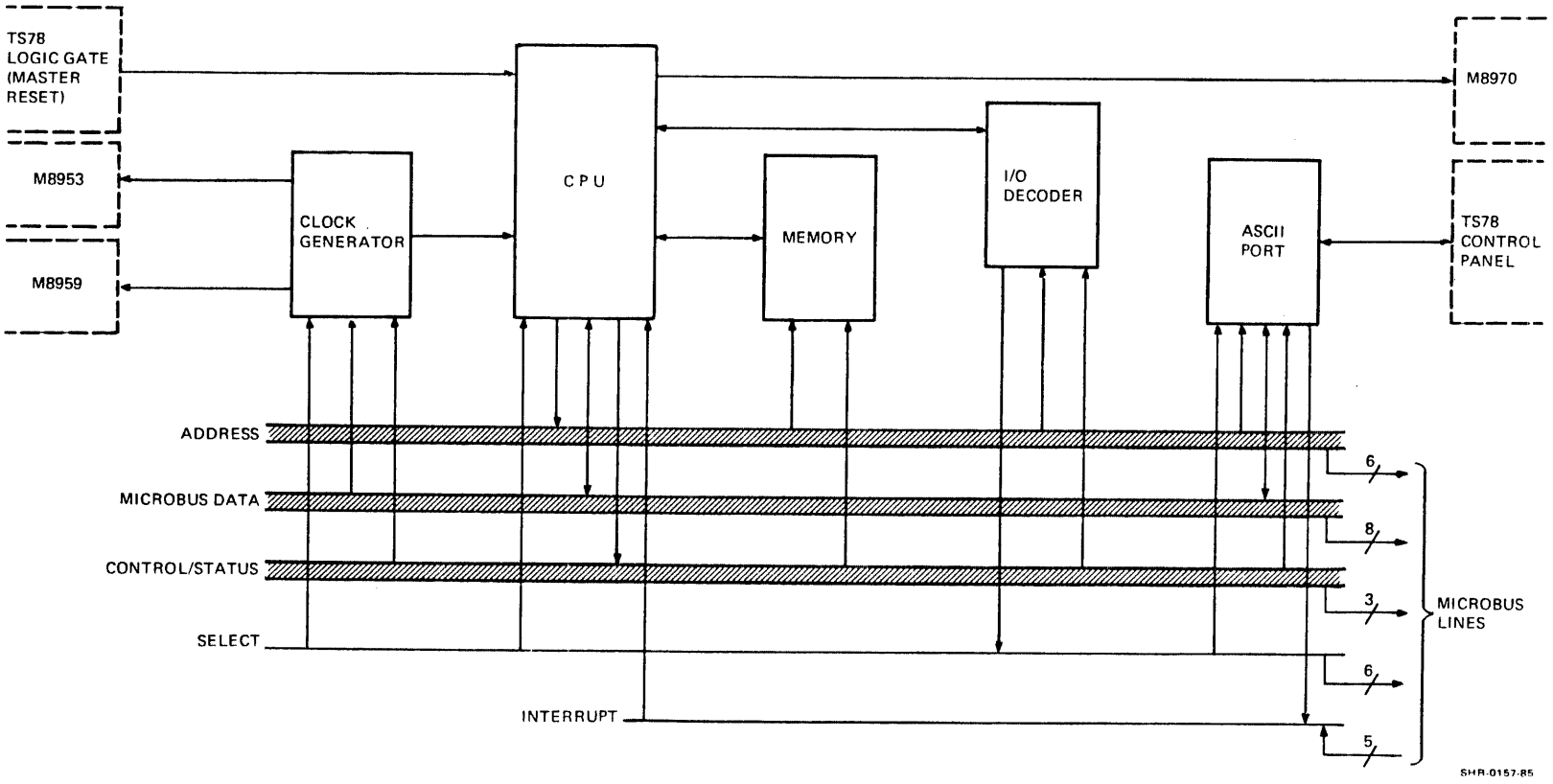


Table 2-1 Microcomputer Pins

Pin	Description
A08-A15	These are the eight most significant bits of the memory address or the entire I/O address. These lines are tri-stated during hold and reset modes.*
A00-A07	These are the multiplexed address/data lines. The least significant eight bits of the memory address or the entire I/O address appear during the T1 cycle. These lines are for data during the T2 and T3 cycles.*
ALE (8MC1 ALE H)	The address latch enable signal goes high during the T1 time state allowing the new partial address on the A07-A00 lines to appear on the microbus. When ALE goes low, the partial address is latched.
S1 (8MC1 S1 H)	This machine cycle status signal when high indicates that a memory or I/O read cycle is in progress. This line is buffered to produce the microbus 8MC1 Read H control line.
S0 (8MC1 S0 H)	This machine cycle status signal when high indicates that a memory or I/O write cycle is in progress.
IO/M (8MC1 IO H)	The I/O-or-memory status signal is high for data transfers involving I/O address space and low for data transfers involving memory address space. It is tri-stated during hold.
RD (8MC1 RD L)	The read control signal when low indicates that the addressed memory or I/O location will be read from. It is tri-stated during hold and reset modes. This line is inverted to produce the microbus 8MC1 Read Strobe H control line.
RST 6.5	Restart 6.5 when raised causes the microprocessor to jump to PC address 34_{16}. It is not used in the TS78 (that is, tied to ground).
RST 7.5	Restart 7.5 is an interrupt raised by a TU command parity error from the currently addressed TU port. It causes the microprocessor to jump to PC address $3C_{16}$.
Trap	Trap is an interrupt raised by a power supply AC LO condition. It causes the microprocessor to jump to PC address 24_{16}.

Table 2-1 Microcomputer Pins (Cont)

Pin	Description
Reset In	Reset is a general-purpose microprocessor clear input that when low jams the program counter (PC) to zero and resets a hold condition. Reset is enabled by a TS78 power-up or pressing the TS78 Master Reset button. When the reset signal goes high again, the microprocessor starts its power-up sequence.
Reset Out (8MC1 Clear H)	The reset output signal when high indicates the microprocessor is being reset. This signal is used to reset the M8970.
X1 (8MC4 8085 Clock H)	X1 is the crystal clock timing input.
CLK (8MC1 CLK H)	Clock output is the inverted master system clock that reflects the internal microprocessor time-state periods. The CLK period is twice that of the X1 input.
SID (8MC1 SID H)	The serial input data signal is monitored by a special microprocessor instruction. In the TS78, this input is not used.
SOD (8MC1 SOD H)	The serial output data signal is set high or low by a special microprocessor instruction. In the TS78, this output is not used.
WR (8MC1 WR L)	The write control signal when low indicates that the addressed memory or I/O location will be written to. It is tri-stated during hold and reset modes. This line is inverted to produce the microbus 8MC1 Write H control line.
RDY (8MC2 RDY H)	The ready input signal is normally high, indicating that the addressed memory or I/O location is ready to send or receive data. When low, this signal causes the microprocessor to extend the data transfer. In effect, the microprocessor waits for a slow memory or I/O device.
Hold	The hold input signal when high causes the microprocessor to release the bus at the completion of the current data transfer. In the TS78, this input is not used (that is, tied to ground).
HLDA (8MC1 HLD A H)	The hold acknowledge output signal when high indicates the microprocessor is honoring a hold request and has therefore released control of the bus. In the TS78, this output stays low since the hold input is grounded.

Table 2-1 Microcomputer Pins (Cont)

Pin	Description
INTR	A raised interrupt request signal causes the microprocessor to assert its INTA output and execute the instruction specified on the A07-A00 lines.
INTA (8MC1 INT A L)	The interrupt acknowledge control output signal when high indicates the microprocessor is honoring a request from the INTR line. It causes the restart instruction generator to place the Restart opcode on the A07-A00 lines.
RST 5.5	Restart 5.5 is an interrupt raised by the assertion of the KINI bit on an enabled real-time controller state line. It causes the microprocessor to jump to PC address $2C_{16}$ starting an STI bus initialization.

* I/O registers may be accessed using a memory address or I/O address. ROM/RAM bytes can only be accessed with a memory address.

Clock Generator

This logic provides the source for the master system clock and also provides the read path and write path clocks.

Master System Clock — This clock is based on an 18.084 MHz crystal oscillator (E41). The crystal output has a 55.3 ns period and is designated 8MC4 Normal CLK H. A binary counter (E40) divides the crystal output by three to yield 8MC4 8085 Clock H. This signal is a square wave with a 166 ns period and is the clock input (X1) to the microprocessor.

The microprocessor in turn divides its X1 input by two to yield the master system clock. The master system clock is, therefore, a square wave with a 332 ns period. It is the basis for all microprocessor instruction, data transfer and interrupt timing. It is also inverted to produce the microprocessor clock output designated 8MC1 CLK H. Signal 8MC1 CLK H is used for synchronization of TS78 peripherals since it reflects the microprocessor's internal time states.

Read/Write Path Clocks — Clock generation logic on the M8972 produces a read path clock and write path clock. Both clocks are used for formatter microcontroller sequencing but the write path clock is also used for data path sequencing. The logic involved with generating these clocks includes: four crystal oscillators (E41, E46, E38, E42), two 8:1 multiplexers (E33, E37), and one 8-bit parallel register (E10).

The four crystal oscillator outputs and a single-step signal (8MC4 Step H) are the source signals. The two multiplexers route the proper source signals to M8972 output pins to yield write path and read path clocks. Multiplexer E37 provides the write path clock (8MC4 Write Clock L) used by the M8959, and multiplexer E33 provides the read path clock (8MC4 Read Clock L) used by the M8953. Table 2-2 lists the frequency of the source signals with their uses as read path and write path clocks.

Table 2-2 Read/Write Path Clock Sources

Clock Frequency	Write Path	Read Path
18.084 MHz	Normal write path clock	—10% clock used for retries
21.84 MHz	+20% clock	+10% clock used for retries
20.0 MHz	+10% clock used for the byte assembly logic during a GCR read	Normal read path clock
15.6 MHz	—20% clock used for the byte assembly logic during a PE read	-30% clock
Single step	A single clock pulse provided for the read and/or write path each time CLKCTL bit 6 is set and reset (toggled). It is used for diagnostic purposes.	

Register E10 is the clock control (CLKCTL) register which the microprocessor accesses as location F0 in its I/O address space. This register specifies MUX codes for E37 and E33, selecting the appropriate source signals routed to the read path and write path. So in effect, the CLKCTL register specifies the write path and read path clocks. CLKCTL bits 2—0 specify the MUX code for E37, while CLKCTL bits 5—3 specify the MUX code for E33. In addition, CLKCTL bit 6 is the single-step source signal.

Data Transfers

Executing a microprocessor instruction involves at least one 8-bit data transfer with memory or an I/O register. Each memory byte and I/O register is assigned an address within a 64K memory address space. Each I/O register is also assigned an address in a 256-byte I/O address space. Executing an input or output instruction causes a read or write data transfer involving the I/O address space. Executing any other instruction causes one or more data transfers in the memory address space.

During a data transfer, the 8085 specifies 16 address bits and the states of 3 status lines and 3 control lines (Table 2-1). During a write operation, the 8085 also provides 8-bit data. For a data transfer in memory address space, the 8085 negates the IO/M status line, and the sixteen address bits specify one location in the 64K address space.

For a data transfer in I/O address space, the 8085 asserts the IO/M status line. The eight least-significant address bits specify one I/O register location within the 256-byte address space. The eight most-significant address bits are a copy of the I/O register address. The eight most-significant address bits are therefore redundant and are not used by other microcomputer logic for data transfers in the I/O address space.

Two bidirectional data buffers (E14 and E18) interface the 8085s multiplexed address/data lines to two sets of data lines. The two sets are: the microbus data lines (8MC1 <7:0> H) used for data transfers with I/O registers, and the memory data lines (8MC1 D<7:0)A H) used for data transfers with memory bytes.

An 8085 data transfer usually consists of three time states. The first time state (T1) is for set-up, the second and third time states (T2 and T3) are for the actual transferring of data. Figures 2-3 and 2-4 show the timing of key signals during read and write data transfers.

Figure 2-3 Write Data Transfer Timing

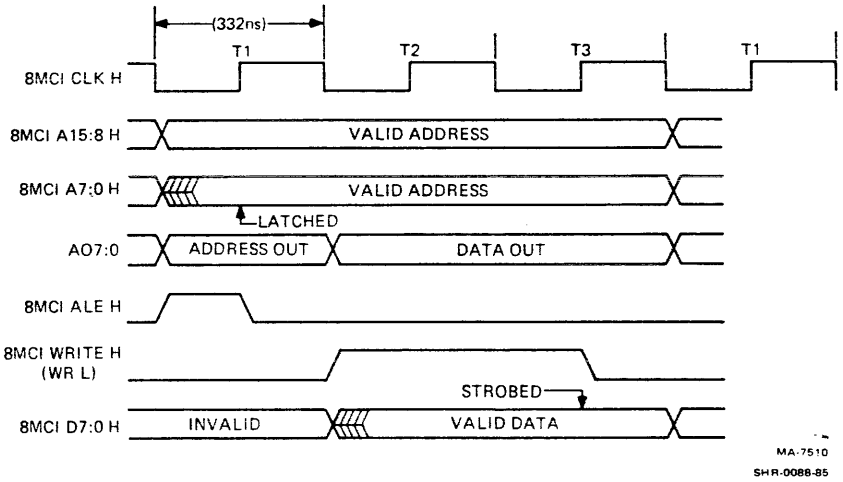
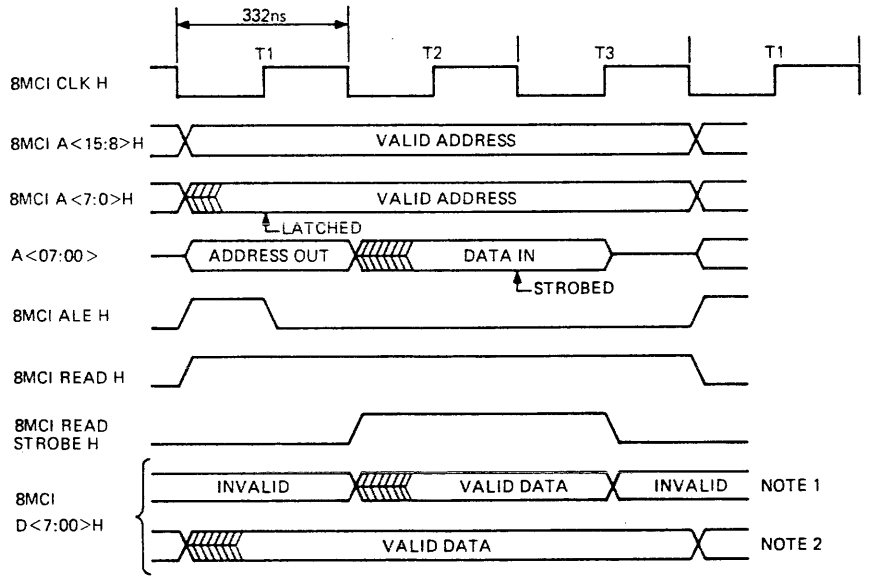


Figure 2-4 Read Data Transfer Timing



1. WHEN READING DATA FROM I/O REGISTERS ON M8970, M8971, M8973 AND READ PATH MODULES; THE DATA APPEARS ON MICROBUS DATA LINES DURING 8MCI READ STROBE H TIME.
   WHEN READING DATA FROM M8972 ROM OR RAM, DATA APPEARS ON MEMORY DATA LINES DURING 8MCI READ STROBE H TIME.
2. WHEN READING DATA FROM I/O REGISTERS OR M8958, M8959, M8955 AND M8972, DATA APPEARS ON MICROBUS DATA LINES DURING 8MCI READ H TIME.

The address bits and correct status-line states appear during T1. The eight most-significant address bits (A15—A08) and the status lines remain latched for the entire data transfer. Buffer E13 places A15—A08 on the 8MC1 A<15:8> H lines. Status bit S1 is inverted twice to produce the 8MC1 Read H microbus line.

The eight least-significant address bits appear on the multiplexed address/data lines but only during T1. Signal 8MC1 ALE H goes high in T1, allowing buffer E29 to place these address bits on microbus lines 8MC1 A<7:0> H. 8MC1 ALE H then goes low again, latching the eight least-significant address bits on the microbus for the remainder of the data transfer.

Also during T1, I/O decoding logic on the M8972 determines which data buffer should be enabled/disabled for the data transfer. For a data transfer with an I/O register, the I/O logic enables the I/O buffer (E14) and disables the memory buffer (E18). So in effect, the microbus data lines become an extension of the multiplexed address/data lines. Otherwise, E18 is enabled and E14 is disabled so the memory data lines become an extension of the multiplexed address/data lines.

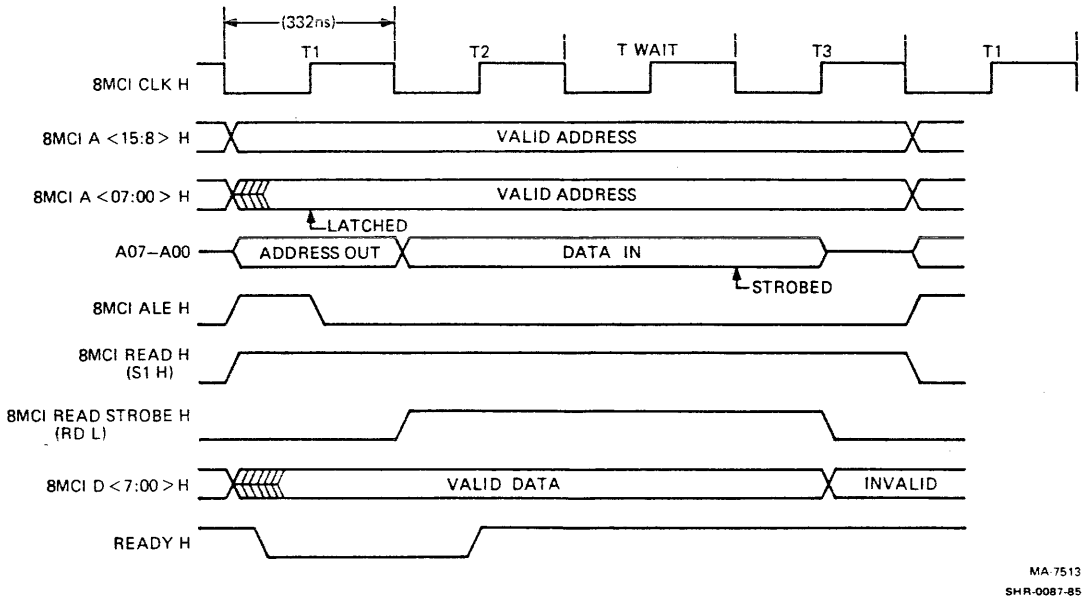
During time states T2 and T3, the multiplexed address/data lines are for data and the 8085 control outputs are valid. Buffer E1 inverts the two read/write control lines to produce microbus lines 8MC1 Read Strobe H and 8MC1 Write H. In T2 and T3, 8MC1 Read Strobe H specifies the direction of the data transfer to the enabled M8972 data buffer. For a read operation, 8MC1 Read Strobe H goes high so the enabled data buffer channels data "to" the multiplexed address/data lines. For a write operation, 8MC1 Read Strobe H remains low so the enabled data buffer channels data "from" the multiplexed address/data lines.

Reading TU78 status requires extra time for the data transfer. To generate the delay, the I/O decoding logic lowers the 8085's ready input from the beginning of T1 to the rising edge of 8MC1 CLK H in T2. At the beginning of T2, the 8085 samples the ready input and since ready is low the 8085 introduces one Twait state. During Twait, the 8085 data transfer signals are frozen. The 8085 samples the ready input again and since ready is now high the 8085 follows the Twait state with T3 finishing the read cycle. As a result, the read cycle is delayed by one time state which is equivalent to one 8MC1 CLK H period (Figure 2-5).

Memory

The microcomputer memory includes 40K bytes of read-only memory (ROM) and 6K bytes of random access memory (RAM). ROM stores all the operational firmware, the 16 in-line microdiagnostics, 2 off-line microdiagnostics (Pass, Fail), and the 11 prewritten tape motion routines. RAM stores data that may change during operation. For example, the extended drive status areas are stored in RAM.

Figure 2-5 Extended Read Data Transfer



The ROM storage area consists of five 8K 8-bit chips which make up the 0—40K (0—$9FFF_{16}$) (0—117777_8) address range. The M8972 also contains a socket reserved for future use by a sixth chip. The corresponding unused ROM address space is 40—48K (A000—$BFFF_{16}$) (120000—137777_8). Address bits A15—A13 specify the desired memory location to one 8K bank. With the address in the 0—$9FFF_{16}$ (0—117777_8) range, A15—A13 cause the ROM decoder (E33) to enable one memory chip. Address bits A12—A00 then specify the desired locaton within the enabled chip. The ROM chips have tri-state outputs which are controlled by the read strobe signal (8MC1 Read Strobe H). When the read strobe arrives, the enabled chip places the contents of the addressed location onto the memory bus data lines (8MC1 D<7:0>A H).

The RAM storage area consists of three 2K 8-bit chips which make up the 54—60K (D800—$EFFF_{16}$) (154000—167777_8) address range. The M8972 also contains three sockets reserved for future use by three additional chips. The corresponding unused RAM address space is 48—54K (C000—$D7FF_{16}$) (140000—153777_8). The microprocessor IO/M output and address bits A15 and A14 enable the RAM decoder (E29) when the desired memory location is within the C000—$EFFF_{16}$ (140000—167777_8) RAM address range. Then E29 decodes A13—A11 which specify the address to a 2K bank. If the address is between $D800_{16}$ (154000_8) and $EFFF_{16}$ (167777_8), E29 enables one of the RAM chips. Address bits A10—A00 then specify the desired memory location within the enabled chip.

A write signal (8MC1 Write H) and the read strobe (8MC1 Read Strobe H) are logically ORed producing a control signal to activate the enabled chip's input/output buffer. A write signal (8MC1 Write L) specifies the direction of the data transfer. A low level causes the enabled chip to strobe in data from the memory data lines and store it in the addressed location. A high signal level for 8MC1 Write L specifies a read operation with the enabled chip placing the desired data on the memory data lines when the read strobe arrives.

During a data transfer in the I/O address space, the I/O signals are asserted disabling the ROM decoder (E33) and the RAM decoder (E29). In effect, this disables the microcomputer memory.

I/O Registers

The microprocessor I/O registers are on various TS78 modules. The microprocessor may access an I/O register as a location in the 64K memory address space or the 256-byte I/O address space. In the memory address space, the I/O registers reside in the 60—64K (F000—FFFF16) range. In either case, address bits A07—A00 specify a register within the range. Appendix B lists and defines the STI bus I/O registers.

I/O decoding logic on the M8972 determines if a microprocessor data transfer involves an I/O register and if so partially decodes the microbus address. This logic consists of one dual 2:4 decoder (E35), one flip-flop (E43), and associated gates. The two counters in E35 are cascaded to generate eight outputs based on four address-bit inputs.

To detect a data transfer involving an I/O register, the I/O decoding logic monitors: four address bits (8MC1 A<15:12> H), an I/O signal (8MC1 IO L), and two read/write signals (8MC1 Read H, 8MC1 S0 H). To specify an I/O register in the memory address space, A15—A12 are all 1s. To specify an I/O register in the I/O address space, 8MC1 IO L is asserted. In either case, the I/O decoding logic asserts 8MC2 EN IO L and 8MC2 IO ADR H. Signal 8MC2 EN IO L then enables E35 to decode address bits A07—A04 and also enables the I/O data buffer. Signal 8MC2 IO ADR H disables the memory data buffer.

Decoding A07—A04 yields a negative pulse for one of eight select signals. The module which contains the accessed I/O register then responds directly or indirectly to the data transfer. Table 2-3 lists the select signal and responding modules for various address ranges in the 256-byte I/O address space. Two select signals are used only with the M8972.

Table 2-3 Microcomputer I/O Decoding

Bus Address	Decoded Select Line	Responding Modules	
		Direct	Indirect
00-3F	8MC2 SEL RD Path L	M8953	M8950s, M8951, M8952
40-7F	8MC2 SEL TU Port L	M8955	TU78 MIA
80-BF	8MC2 SEL PMIC L	M8970	M8971
C0-CF	8MC2 SEL D BUS L	M8972 M8973	
D0-DF	8MC2 SEL WMC/XMC L	M8959	M8958, M8955
E0-EF	8MC2 SEL INT STAT L	M8972	
F0-FF	8MC2 SEL CLK CTRL L	M8972	

The remaining six select lines are used by the M8972 ASCII port USART and other TS78 modules designated "direct". Asserting one of these six lines enables microbus interface logic on the selected "direct" module. This enabled logic in turn decodes address bits A03—A00 and read/write signals, to generate a select signal for a register on the "direct" module or on a corresponding TS78 "indirect" module. For example, the M8970 generates select signals for registers on the M8970 and M8971.

The microbus data lines connect to nine of 20 TS78 modules. Eleven TS78 modules (M8950s, M8951, M8952) transfer information to/from the microbus data lines by means of the M8953.

Microbus address 40 is the tape unit command/status byte which accesses eight registers on a tape unit MIA module. Information is transferred between these eight registers and the microbus data lines by means of the TU bus and M8955. To prepare for a tape unit command/status transfer, the microprocessor writes an MIA register address into location 40 which is then transferred to the tape unit. (Setting location 40 bit 7 to 1 indicates bits 6—0 contain an MIA register address.) Appendix C lists and defines the MIA registers.

Reading tape unit status requires extra time for the data transfer. To delay the microprocessor, flip-flop E43 manipulates the 8085's ready input. When a TU port address is decoded (8MC2 SEL TU Port L) during a read cycle (8MC1 S1 L) at address latch enable time (8MC1 ALE H), the ready flip-flop is directly reset. This lowers the ready input. Ready is then clocked set on the rising edge of the next 8MC1 CLK H pulse, following the negation of 8MC1 ALE H. Therefore, the microprocessor adds one wait state during which it continues to produce 8MC1 CLK H but does not perform any internal processing functions. As a result, the read cycle is delayed (Figure 2-5).

The M8972 contains seven registers, four of which are in the M8972 ASCII port USART. Asserting 8MC2 SEL D BUS L with address bit 3 equal to 0 selects the USART for a data transfer. Asserting 8MC2 SEL INT STAT L selects either the TU-ASCII-Speed-Select (TUSEL) register for a write data transfer or the Interrupt Status (INTSTA) register for a read data transfer. Asserting 8MC2 SEL CLK CTRL L selects the Clock Control (CLKCTL) register.

The 8085 services the M8970 interrupt timer, M8972 ASCII port, and several error conditions on an interrupt basis. To accomplish this, formatter logic external to the 8085 generates seven interrupts. The interrupts include one trap interrupt, two restart interrupts and four vector interrupts. Every interrupt is associated with an activating formatter control signal.

The 8085 has five interrupt inputs: TRAP, RST 7.5, RST 6.5, RST 5.5 and INTR (Table 2-4). The 8085 samples all five inputs during the next to last clock cycle of every instruction. If more than one interrupt input is sampled as active, the 8085 performs the interrupt with the highest priority.

Table 2-4 TS78 Interrupt Summary

Priority	Interrupt Type	Control Signal That Activates Interrupt	New 8085 PC (Hex Value)	Interrupt Description
1	Trap	PS AC LO L	24	AC LO power condition.
2	Restart 7.5	TUP2 P CMD PE L	3C	TU port command parity error.
3	Restart 6.5	GND	34	Not used.
4	Restart 5.5	BI4 KINI H	2C	STI bus initialization.
5	Restart 7	TUP2 P STAT PE L	38	TU port status parity error.
5	Restart 6	PM6 MCLK INT L	30	Interrupt timer (MCLK) time-out.
5	Restart 5	8MC5 TX INT L	28	ASCII port USART transmitter ready and empty.
5	Restart 3	8MC5 RX INT L	18	ASCII port USART receiver ready.

Notes:

1. Restart 7-0 are the vector interrupts. Restart 0, 1, 2, and 4 are not used.
2. All interrupts except Trap are enabled and disabled by the EI and DI instructions respectively. Restart 7.5, 6.5 and 5.5 can also be individually masked out using the SIM instruction. The trap interrupt cannot be masked or disabled with software. Activating the 8085's reset input disables the vector interrupts and masks Restart 7.5, 6.5 and 5.5.

The INTR input is for vectored interrupts and has the lowest priority. Asserting any one of the four vector-interrupt control signals causes the INTR input to go high. If the 8085 samples only INTR as active and if interrupted are enabled, the 8085 inhibits the incrementing of the program counter (PC), and the 8085 starts an interrupt acknowledge machine cycle to obtain an opcode. This cycle has three time states and is similar to a read data transfer.

During T1, 8MC1 ALE H goes high, clocking the RESTART instruction opcode into register E32. During T2 and T3, the 8085 asserts its INTA control output, enabling E32 to place the RESTART opcode on the multiplexed address/data lines. The 8085 decodes the received opcode and executes the specified instruction (that is, a RESET instruction). Note that three vector-interrupt control signals specify RESET opcode bits 5, 4, and 3, which form 3-bit restart address selecting one of eight reserved memory locations.

Executing the RESTART instruction disables further interrupts, pushes the current contents of the PC onto the stack, and decrements the stack pointer. The 8085 then calculates a new PC value, mapping the 3-bit restart address into PC bit 5—3. All other PC bits are 0 (zero). So in effect, the 3-bit restart address is multiplied by eight to yield the new PC value. The 8085 then branches to the memory location specified by the new PC value.

The TRAP and two RESTART inputs each have a formatter control signal associated with them. Asserting one of these three control signals activates the corresponding interrupt, causing the 8085 to execute a RESTART instruction. However, no external vector is used to generate the new PC value. The 8085 loads a reserved value associated with the interrupt into the PC. Table 2-4 lists the interrupts, their priority of execution, the formatter control signal that activates every instruction, and the PC address to which the 8085 branches.

The memory location to which the 8085 branches in a RESTART usually contains the starting address for a verification routine. Executing the verification routine causes the 8085 to read the INSTSTA and PSTAT register. The 8085 then checks the appropriate bit to verify that the proper formatter control signal has initiated the interrupt. Reception of an unverified (unexpected) interrupt is classified as a fatal formatter error. If the received interrupt is verified, the 8085 branches to the interrupt's service routine. Note that for the STI bus initialization interrupt, the 8085 branches directly to the interrupt's service routine.

After executing any interrupt service routine, the stack is popped, interrupts are enabled, and the interrupted program continues. Note that an interrupt's service routine may contain an EI instruction that enables the interrupt before the entire routine is expected.

ASCII Port

This logic allows an RS232-compatible, external terminal to communicate with the TS78 microprocessor (8085). The ASCII port consists of an 8251A programmable universal synchronous/asynchronous receiver-transmitter (USART) and baud rate generator logic. The USART contains a receiver which performs serial-to-parallel conversion, and a transmitter which performs parallel-to-serial conversion. The baud rate generator provides the transmitter and receiver clock signals for the USART.

The 8085 programming specifies the communication format. The 8085 accesses the USART through two read/write I/O address space locations which access four internal registers/buffers.

The USART communicates with the 8085 by means of receiver and transmitter interrupts. As part of an interrupt response, the 8085 manipulates flags in RAM. When the TS78 is off-line, the 8085 periodically samples these flags and carries out any pending operations in accordance with these RAM flags.

For a TS78 master reset or power-up, the USART does an internal reset and the 8085 sets up certain flags in RAM. The 8085 then executes a baud-change operation, initializing the ASCII port for asynchronous operation at 9600 baud. However, only reception is enabled.

Character Reception — Transmitted output from an external terminal is routed through the TS78 control panel, inverted on the M8972, and applied to the USART receiver data input on the M8972. The receiver data line has a high signal level when idle. The USART samples the line at the raising edge of the receiver clock input. A negative edge on the receiver data line indicates the beginning of the start bit and activates USART internal circuits. Thirty-two clock cycles later the USART samples the line again, confirming the presence of the start bit if the line's level is low. With start confirmed, the USART samples the line every sixty-four clock cycles, shifting each bit into an internal register until the entire character is loaded. Since parity is disabled, the USART does not check for the parity bit, but it does check the stop location (that is, right after the eighth data bit) for a high signal level. The USART then loads the 8-bit data into the receiver buffer and asserts the receiver ready (8MC5 RX INT L) signal. This signal is an 8085 interrupt signal; asserting it indicates that ASCII port data is available in the USART. The data remains available to the 8085 until another complete character has shifted in.

In response to the receiver ready interrupt, the 8085 reads location C1, retrieving the USART status word. But it does not check the status word. The 8085 then reads the location C0 to retrieve the data in the USART receiver buffer. The USART in turn negates the receiver ready signal. The 8085 masks bit 7 of the retrieved data leaving a 7-bit ASCII character code.

8085 Received Character Handling — If this is the first received character, the 8085 determines if the character is a CTRL C or CRTL Y. If the character is not recognized, the 8085 assumes ASCII port baud rate does not match the external terminal baud rate. A new baud rate is then calculated, and flags in RAM are set up so that a baud-change operation is carried out at a later time. The 8085 then writes a command word into C1 causing the USART to reset.

If the first received character is recognized as a CRTL C or CRTL Y, the 8085 sets up for certain flags, vectors, and buffer areas in RAM to prepare for further operation. The 8085 also writes a command word into C1 to enable the USART transmitter as well as the receiver. The 8085 then sets up the ASCII port to echo CTRL C.

If this is not the first received character, the 8085 classifies the character and takes appropriate action. Alphanumeric characters are stored in a RAM buffer area and echoed by the ASCII port. When a RETURN character arrives, the 8085 sets up flags in RAM to decode the contents of the RAM buffer area and executes the command at a later time.

Most received control characters are echoed and ignored. Certain control characters (for example, CRTL U) have defined functions. These predefined characters are echoed and executed immediately upon receipt.

Character Transmission — The 8085 places transmission messages (for example, echoed characters) into a RAM buffer area. ASCII port transmission is interrupt driven.

When the USART transmitter is ready to accept new data (T RDY) and has no data internally to send (T EMPTY), the transmitter interrupt signal (8MC5 TX INT L) is asserted. The 8085 reacts to the transmitter interrupt by retrieving a byte from the RAM buffer. It then clears bit 7 of the retrieved byte and writes the resulting byte into the USART transmitter buffer at location C0.

Upon receiving the new data byte from the 8085, the USART negates the transmitter interrupt. The USART transmitter then adds a start and stop bit to the received b-bit data to yield a composite character, and it converts this character to a serial bit stream which appears at the transmitter data output lines. The transmitter shifts out one bit of the character every 64 clock cycles of the transmitter clock input, at the falling edge of the clock. The transmitter data output is inverted by the M8972, routed through the TS78 control panel, and made available to the external terminal.

Interrupt driven transmission continues until the RAM buffer is empty. The 8085 then writes a command word into location C1, disabling the transmitter. The 8085 also decodes and executes any commands that were delayed during transmission. Note that the USART transmitter data output line has a high signal level when idle.

Baud Rate Generator — This logic consists of a 2.4567 MHz crystal oscillator (E3), dual 4-bit binary counter (E11), and two 8x1 multiplexers (E6 and E7). The counter divides the crystal output to produce eight signals with frequencies corresponding to eight standard baud rates. The multiplexers route the proper signals for specified baud rates to the USART clock inputs. Multiplexer E7 provides the USART transmitter clock, and E6 provides the USART receiver clock. The USART divides these clocks by 64 to yield the transmitter and receiver baud rates.

The TUSEL register specifies MUX codes for E6 and E7, selecting the appropriate counter outputs for routing to the transmitter and receiver clock inputs. So in effect, the TUSEL register selects the transmitter and receiver baud rates.

8085 Baud Rate Handling — To change baud rates, the 8085 first disables the USART by writing all 0s mode and command words into location C1. The 8085 then writes into the TUSEL register, specifying the transmitter and receiver baud rates. The 8085 also writes a command word into C1 to reset the USART and writes a mode word into C1 to initialize the USART. This mode word sets up asynchronous communication with: 1 stop bit, parity disabled, 8-bit data, and a baud rate factor of 64. The 8085 then writes a command word into C1 to enable the receiver, disable the transmitter, and reset the status error flag. Finally, the 8085 reads C0 to clear the receiver buffer.

For a TS78 master reset or power-up, the 8085 initializes the receiver and transmitter at 9600 baud. If the first received character is not recognized as a CRTL C or CRTL Y, the 8085 goes through an autobaud sequence. Based on the received unrecognized character, the 8085 calculates a lower MUX code for E6 and E7. The 8085 then sets up flags in RAM for a baud-change operation at a later time (that is, the next time the 8085 goes through the off-line idle loop). When a CTRL C or CTRL Y is repeatedly keyed on the external terminal, the process repeats.

As a result, the transmitter and receiver baud rates are lowered until the received character is recognized as a CTRL C or CTRL Y or until the baud rate 300 is reached. If 300 baud is reached, the 8085 starts again at 9600 baud for the next change. Once CRTL C or CRTL Y is recognized, the baud rate stays the same until termination with CRTL Z or until SET TERMINAL SPEED. Use of this command is the only way to introduce split baud rates.

Modem Signals — Through the TS78 control panel, the ASCII port logic accepts one modem signal and provides a second modem signal. The received signal is ASCII DSR H which inverts on the M8972 and is applied to the DATA SET READY input of the USART. The received signals remain high as long as an external terminal is connected and powered up. As long as this is the case, the transmitter is enabled when programmed to do so by the 8085. A low level for ASCII DSR H prevents the ASCII port from transmitting to an unusable terminal.

The ASCII port provides a DATA TERMINAL READY signal (8MC5 DTR H). The signal remains high as long as the TS78 is powered up, which indicates to a connected external terminal that the ASCII port is available.

2.3.2 M8970 STI Protocol Microcontroller

The M8970 handles protocol characters and performs data conversions involving message frames and data on the STI bus data lines. The M8970 also generates a 16-bit EDC character for validating data, and clock signals for the M8971 that provide the basis for timing on the STI bus data lines. In accordance with the M8972, the M8970 provides port control signals and specifies the formatter receiver ready state bit for the M8971. And the M8970 provides a series of MCLK interrupts used by the M8972 to implement certain formatter timing functions.

Figure 2-6 is a block diagram of the M8970. It divides the M8970 into nine major functional areas. The following paragraphs discuss each functional area separately.

Data Path

The M8970 data path (Figure 2-7) transfers information in four types of operations: write data reception, command message reception, read data transmission, and response message transmission. Write data and command messages are received from the M8971 in serial form and converted to parallel bytes. The write data goes to the M8959, and the command message bytes go to the M8972. Read data bytes come from the M8959, are converted to serial form, and sent to the M8971. Response message bytes come from the M8972, are converted, and passed to the M8971.

The data path also handles STI protocol characters. It receives and decodes one or more sync characters during write data or command message reception. And it generates one or more sync characters during read data or response message transmission. The data path also decodes error detection codes (EDC) during command message reception, and generates EDC codes during response message transmission.

Table 2-5 relates the four operation types to ten key control signals. Four columns are valid only while transferring data or message bytes during an operation. Two columns are valid only when generating or receiving a sync character during an operation.

Figure 2-6 M8970 Block Diagram

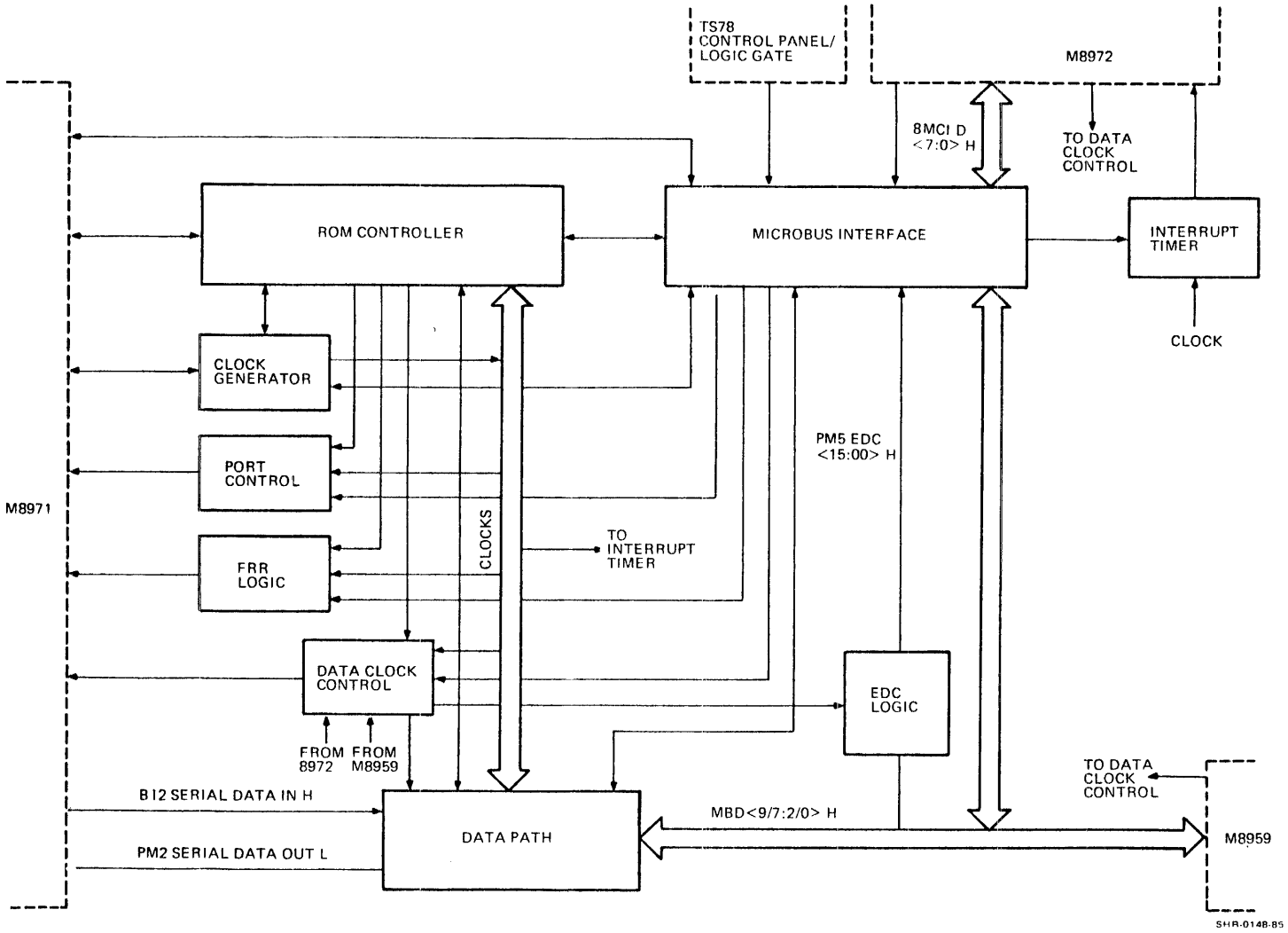


Figure 2-7 M8970 Data Path and EDC Logic

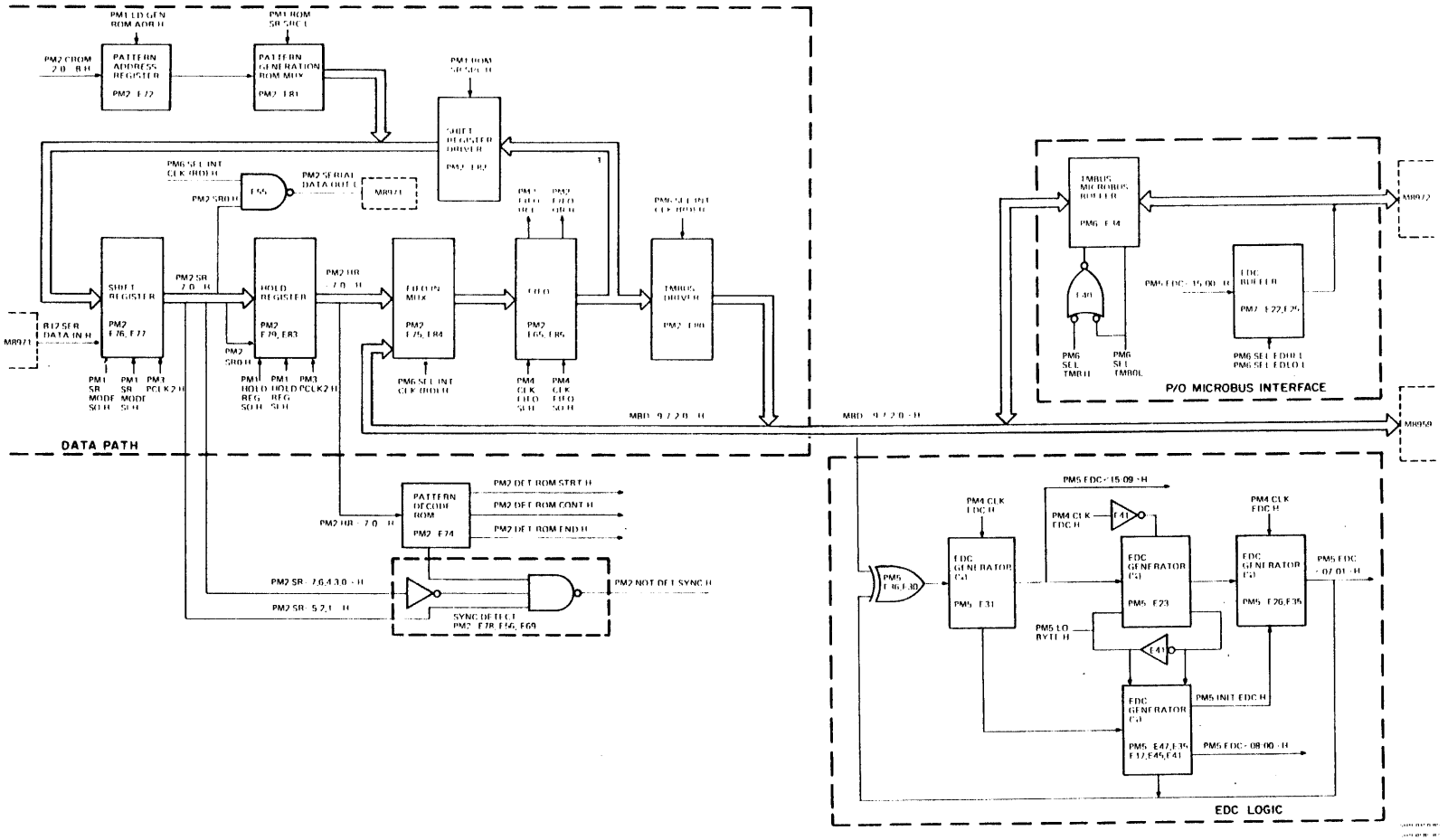


Table 2-5 Data Path Operation Control

Signal	Receive Write Data	Transmit Read Data	Receive Sync	Generate Sync	Receive Command Message	Transmit Response Message
PM1 SR Mode S0 H	LO	_! !_	LO	_! !_! !_	LO	_! !_
PM1 SR Mode S1 H	HI	_! !_	HI	_! !_! !_	HI	_! !_
PM1 Hold REG Mode S0 H	_! !_	LO	LO	LO	_! !_	LO
PM1 Hold REG Mode S1 H	_! !_	LO	HI	LO	_! !_	LO
PM6 SEL INT CLK (RD) H	LO	HI	LO	HI	LO	HI
PM1 ROM SR SRC L	HI	HI	HI	LO	HI	HI
PM1 ROM SR SRC H	LO	LO	LO	HI	LO	LO
PM1 LD GEN ROM ADR H	LO	LO	LO	_! !_! !_	LO	LO
PM4 CLK FIFO SI H	_! !_	_! !_	--	--	_! !_	_! !_
PM4 CLK FIFO SO H	_! !_	_! !_	--	--	_! !_	_! !_

NOTES
-- = irrelevant

HI = remains high during stated process

LO = remains low during stated process

_! !_ = positive pulse required to transfer one data or message byte

_! !_! !_ = two positive pulses necessary to generate entire sync character

The following paragraphs discuss how the data path logic implements the four types of transfer operations. Note that the shift register and hold register can receive input in a parallel mode (one byte at a time) or in a serial mode (one bit at a time).

Write Data Reception — During write data reception, the HSC sends the formatter a 16-bit sync character, 16-bit data words, and a 16-bit EDC character. The M8971 routes this information to the serial input of the shift register in the M8970. For every word or character, the low byte appears first. Within every byte, the least significant bit (LSB) appears first. The shift register is always in shift mode. Successive clock pulses shift the serial input to the eight parallel outputs, from PM2 SR 7 H to PM2 SR 0 H. The PM2 SR 0 H line is the serial input for the hold register.

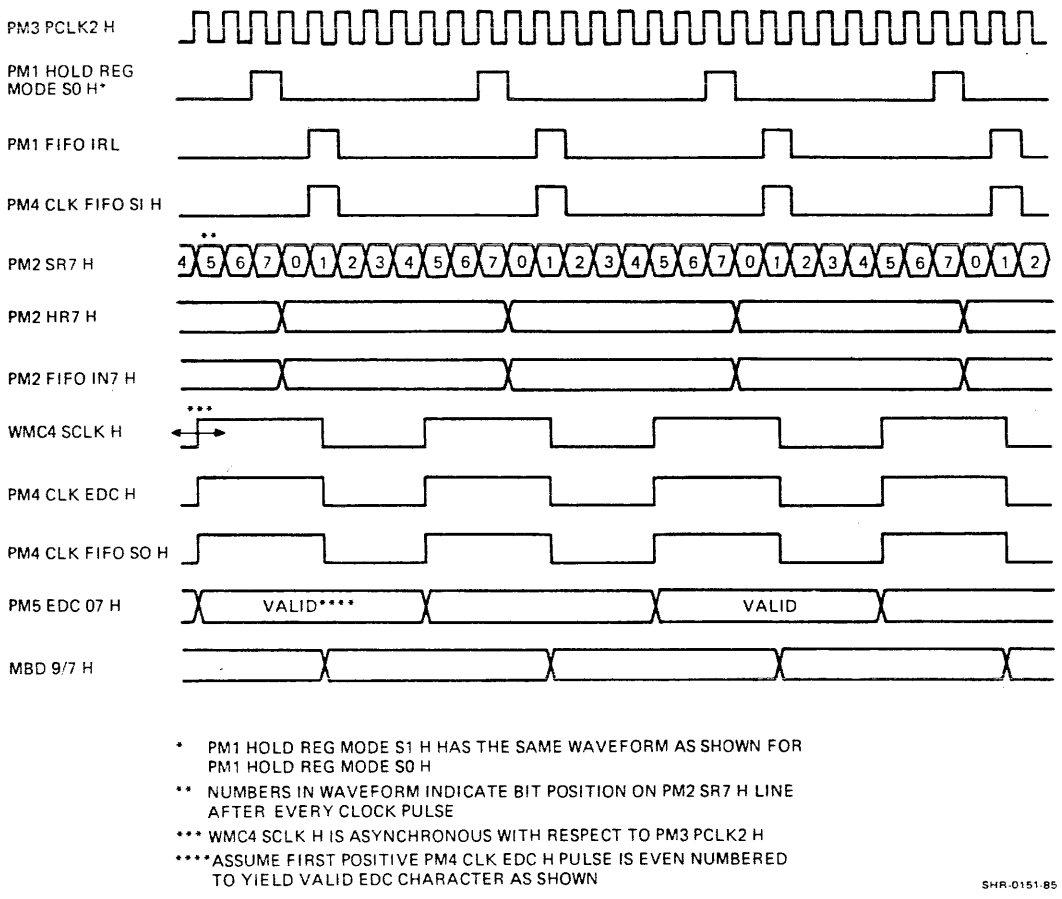
To receive the sync character, the hold register is also in shift mode. In effect, the shift register and hold register act as one 16-bit shift register. The pattern decode ROM and sync detect gates determine if the shift and hold register outputs correspond to the sync character. Signal PM2 NOT DET SYNC H goes low to indicate reception of the sync character to the ROM controller.

After receiving the sync character, the hold register is in either parallel load or hold mode. Every eight clock pulses, a full byte is present at the shift register output, and the hold register goes into parallel load mode. The next clock pulse loads the entire data byte into the data register. The FIFO-in MUX routes the data byte to the FIFO and then the FIFO receives a positive clock pulse (PM4 CLK FIFO SI H) to shift in the data byte. The same clock pulse that loads the byte into the hold register also loads the LSB of the next byte onto the PM2 SR 7 H output of the shift register.

The data clock control logic produces FIFO-input clock pulse PM4 CLK FIFO SI H in autoshift mode. This clock pulse is based on a hold register mode signal (PM1 Hold REG S0 H) from the ROM controller, and the FIFO-input ready signal (PM2 FIFO IR L) from the data path. When the FIFO is full, PM2 FIFO IR L stays high, inhibiting further FIFO-input clock pulses. With PM2 FIFO IR L high for four M8970 clock cycles, the M8971 inhibits the STI line clock. The HSC then stops sending data to the formatter. This inhibits the M8970 shift register and hold register since the clock for these registers is based on the incoming data rate. Therefore, no data is lost because of FIFO overflow. Two M8970 clock cycles after the FIFO input is ready, the M8971 starts the STI line clock and HSC transmission and formatter reception resume.

The TMBUS driver routes the FIFO data output to the MBD lines (MBD 9/7:2/0 H). The data then appears at the EDC logic, microbus interface, and M8959 module. The first data byte to enter the FIFO appears on the MBD lines, and the EDC logic produces XOR values based on the data. The data clock control logic produces two positive clock pulses (PM4 CLK FIFO SO H and PM4 CLK EDC H) based on a positive clock pulse (WMC4 SCLK H) from the M8959. At the leading (rising) edge of WMC4 SCLK H, the XOR values based on the present MBD data byte load into an EDC logic register. At the trailing (falling) edge of WMC4 SCLK H, the present MBD byte loads into the M8959 and the FIFO shifts data out so the next data byte appears on the MBD lines. The M8959 issues clock pulses until all the data bytes are loaded. Figure 2-8 shows the timing relationships for typical write data flow through the M8970 data path.

Figure 2-8 Write Data Flow Timing (M8970 Data Path)



The byte count, stored in M8972 memory, represents the number of data bytes to be written in a record. For an odd-numbered byte count, the HSC provides an extra byte of 0s. The M8959 leaves the byte in the M8970 FIFO but the M8972 retrieves it. An EDC logic register clocks in the XOR value based on the 0s and the FIFO shifts the 0s out. The low byte of the supplied EDC character then appears on the MBD lines. With the extra pulse, the total number of EDC clocks for the record is even. This is a requirement for proper EDC character calculation. The extra byte of 0s is already part of the EDC character supplied by the HSC.

For an even-numbered byte count, an extra clock pulse is not required. When the M8959 is finished, the calculated EDC character is valid and the low byte of the HSC supplied EDC character is already on the MBD lines.

The M8972 reads the calculated EDC character through the EDC buffer, using two data cycles, one for each byte. In a similar manner, the M8972 reads the supplied EDC character through the TMBUS/MICROBUS buffer. The M8972 then compares the calculated and HSC supplied EDC characters. A character match indicates no detected transmission or protocol error.

Read Data Transmission — During read data transmission, the M8970 supplies the sync character, data, byte count, and EDC character to the M8971. The M8971, in turn, routes the information to the HSC through the STI bus. The M8959 supplies the data to the MBD lines on the M8970. With each byte, the M8959 also supplies a positive WMC4 SCLK H clock pulse.

The M8970 shift register parallel loads bytes to its eight outputs and shifts every byte from PM2 SR 7 H to PM2 SR 0 H. Gate E55 inverts PM2 SR 0 H routing the bytes in a serial stream to the M8970 output. For every byte, the least significant bit (LSB) appears first.

At the start of the read data transmission cycle, the FIFO is empty and the shift register receives bytes from the pattern generation ROM. The four bits loaded into the pattern address register select the appropriate STI code applied by the generator. Until data is ready for shifting, the selected pattern is an all 0s byte. Therefore, the tape drive clocks a steady stream of 0s to the HSC until the M8970 receives data from the M8959.

Signal PM2 FIFO OR H goes high indicating to the ROM controller that the first data byte is available for shifting out. As a result, the pattern generation ROM supplies the low byte and then the high byte of the sync character. The shift register parallel loads and shifts each byte. The pattern generation ROM is then disabled and data is shifted out.

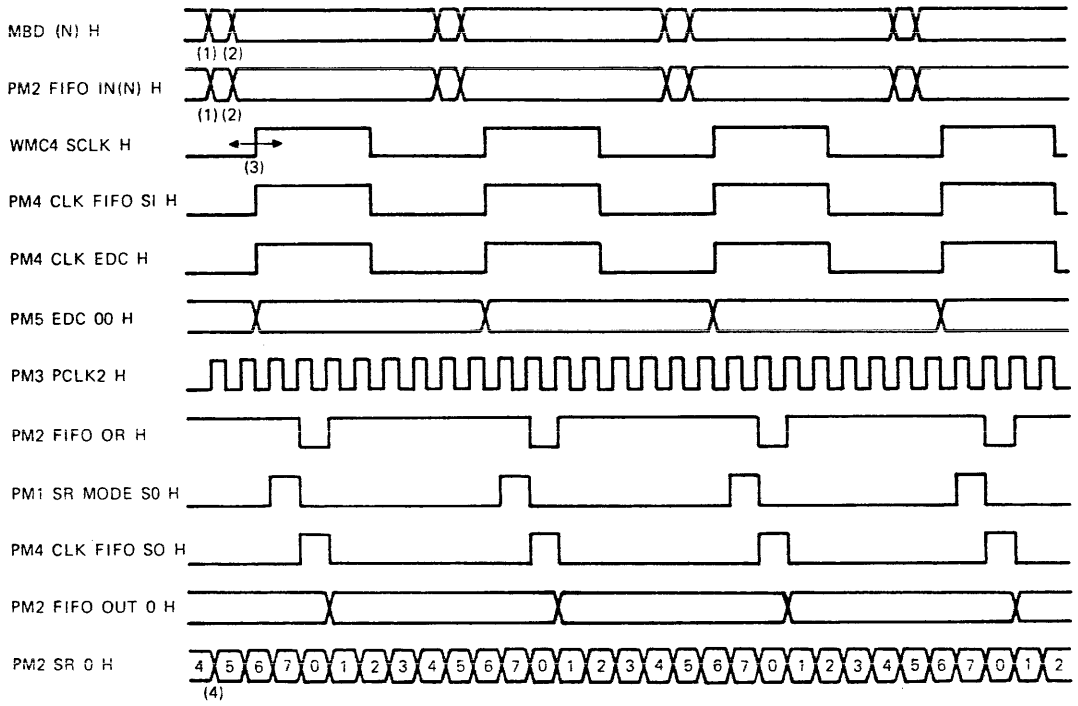
Data from the M8959 appears at the data path and EDC logic. The FIFO-in MUX routes the data to the FIFO. The data clock control logic generates two positive clock pulses (PM4 CLK FIFO SI H and PM4 CLK EDC H) based on WMC4 SCLK H. At the leading (rising) edge of WMC4 SCLK H, the FIFO shifts in the data and the EDC logic calculates a new byte.

The shift register driver routes the data to the shift register, which converts it to a serial stream. Every eight clock pulses the shift register goes into parallel load mode. Otherwise, it is in shift mode. The M8971 strobes in the last bit from PM2 SR 0 H in the same clock period that the LSB of the new byte appears on PM2 SR 0 H.

While a byte is shifting out, the FIFO receives a FIFO-out clock pulse, deleting the data from the FIFO. The PM2 FIFO OR H signal then goes high to indicate that a new data byte is available. The cycle repeats, loading the new byte into the shift register as long as the M8970 receives data. Note that the data clock control logic generates the FIFO-out clock pulses based on a ROM controller strobe signal. Figure 2-9 shows the timing relationships for typical read data flow through the M8970 data path.

If the FIFO is empty, PM2 FIFO OR H stays low, the shift register goes into hold mode and the M8971 inhibits the STI line clock. Therefore, the HSC sees nothing on the enabled STI read/response data line. Note that three bits stay in the shift register. After the FIFO output is ready again, data transmission is enabled and the cycle continues, shifting out the three remaining bits first.

Figure 2-9 Read Data Flow Timing (M8970 Data Path)



NOTES

1. M8959 PLACES BITS 7:4 OF NEW BYTE ON MBD<3/7:6/4>H WHICH THEN APPEAR ON FIFO IN<7:4>H
2. M8959 PLACES BITS 3:0 OF NEW BYTE ON MBD<5/3:2/0>H WHICH THEN APPEAR ON FIFO IN<3:0> H. THE ENTIRE BYTE IS NOW VALID ON THE MBD AND FIFO IN LINES.
3. WMC4 SCLK H IS ASYNCHRONOUS WITH RESPECT TO PM3 PCLK2 H
4. NUMBERS IN WAVEFORM INDICATE BIT POSITION ON PM2 SR0 H LINE AFTER EVERY CLOCK PLUSE

SHR-0152-85

After data transmission, the M8972 calculates the byte count based on the record. For an odd-numbered byte count, the M8972 loads a fill byte of 0s onto the MBD lines of the M8970. The M8972 then loads the byte count using two data cycles, supplying the low byte first. The EDC logic calculates a new character for each byte and the FIFO strobes in each byte. With the loading of the fill byte, the total number of EDC clock pulses for the record is even. This is a requirement for proper EDC character calculation.

For an even-numbered byte count, a fill byte is not needed since the EDC character is already valid. Therefore, the M8972 loads only the byte count.

The shift register shifts all but the last three bits of the byte count to the M8971 since the FIFO is then empty. Inhibiting the serial bit stream now yields enough time for the M8972 to clear the data ready bit in FSTO. Therefore, the HSC receives a 0 data ready state bit after reception of data but before reception of the entire byte count.

The calculated EDC character is based on the data and byte count. The M8972 reads the calculated EDC character through the EDC buffer using two data cycles, one for each byte. By means of the TMBUS/MICROBUS buffer, the M8972 then loads the EDC character onto the MBD lines, loading the low byte first. The last three bits of the byte count, the EDC low byte, and five bits of the EDC high byte are shifted out.

When the FIFO is empty, the M8972 issues a flag instructing the M8970 to end the cycle. The pattern generation ROM is enabled and the all 0s pattern selected. The last three bits of the EDC character are then shifted out followed by 16 zeroes.

Command Message Reception — During a command message reception, the HSC sends the formatter from 1 to 64 frames. A frame consists of a 16-bit sync character, an 8-bit message or checksum, and an 8-bit hamming code. The M8971 routes the 4-byte frame(s) to the input of the shift register on the M8970. For every byte, the least significant bit (LSB) appears first. The shift register is always in shift mode. The serial stream shifts from PM2 SR 7 H to PM2 SR 0 H. The PM2 SR 0 H line is the serial input for the hold register.

As in write data reception, the hold register is in shift mode to receive the sync character. In effect, the hold and shift registers are cascaded. The outputs of these registers appear at the pattern detect ROM and sync detect gates. Signal PM2 NOT DET SYNC H goes low to indicate reception of the 2-byte sync character.

The hold register acts as a parallel load buffer to receive the next two frame bytes. The pattern detect ROM monitors the hold register outputs to determine if a start, continue or end hamming code is present. These three hamming codes indicate the relative position of the frame in a multi-frame message.

Eight clock periods after the sync, the entire message byte is present at the shift register output, and the hold register goes into parallel load mode. The next clock pulse loads the entire byte into the hold register. The FIFO-in MUX routes the byte to the FIFO and the FIFO receives a clock pulse to shift in the message byte. The TMBUS driver routes the FIFO output to the MBD lines. The same clock pulse that loads the message byte into the hold register also loads the LSB of the hamming code byte onto PM2 SR 7 H.

The hamming code byte, when present at the shift register output, loads into the hold register. It then appears at the FIFO input and pattern decode ROM. If the first frame does not have a start hamming code, the FIFO receives a second clock pulse to shift in the hamming code. The message and hamming code in the FIFO make up a one-frame level 1 message (for example, read forward) or an error has occurred. By means of the TMBUS/MICROBUS buffer, the M8972 retrieves the message and hamming code, and it sets a flag if an error has occurred.

If the first frame has a start hamming code, PM2 DET ROM STRT H goes high. The hold register then goes into shift mode to receive the sync character for the second frame. The FIFO does not receive a second clock pulse, so only the message byte is in the FIFO.

After a valid start frame, the M8970 continues to receive frame(s), storing only the first byte after the sync character. As long as the last byte contains the continue hamming code (PM2 DET ROM CONT H = HI), the hold register goes into shift mode for the next sync. Signal PM2 DET ROM END H goes high ending reception when the end hamming code appears at the hold register output. This hamming code also identifies the last byte in the FIFO as the checksum.

As already mentioned, the M8972 sets a flag if the first command frame contains an error. Once the flag is set, the M8970 continues reception until the end frame is seen. However, the FIFO does not strobe in the message byte(s) or checksum byte. Note that the M8970 sets a flag and continues operation if it sees an invalid hamming code after the start frame. But the message is stored in the FIFO.

When reception of a level 2 message is complete, the M8972 retrieves the contents of the M8970 FIFO. The TMBUS driver routes the contents of the FIFO to the MBD lines.

To retrieve a byte, the M8972 causes the microbus interface to produce a negative PM6 SEL TMBI L pulse. The TMBUS/MICROBUS buffer then places the byte onto the microbus data lines. The data clock control logic generates two positive clock pulses (PM4 CLK FIFO SO H and PM4 CLK EDC H) based on PM6 SEL TMBI L and microprocessor clock 8MC1 CLK H. Although EDC clocks are generated, the EDC character calculation is not valid during command message reception. After PM6 SEL TMBI L goes high again, the FIFO shifts out the byte at the trailing (falling) edge of the next microprocessor clock pulse. The next message byte or checksum byte then appears on the MBD lines. FIFO-output ready signal PM2 FIFO OR H goes high to indicate the next byte is available.

The M8972 samples FIFO-output ready through the microbus interface and continues to retrieve bytes until it samples a low signal value. FIFO-output ready stays low when the FIFO is empty (that is, all message byte(s) and the checksum byte have been retrieved).

Response Message Transmission — During a response message transmission, the M8970 supplies from 2 to 64 frames to the M8971. The M8971 in turn routes the information to the HSC by means of the STI bus. A frame consists of a 16-bit sync character, an 8-bit message or checksum byte, and an 8-bit hamming code. The M8972 supplies the message byte(s) and checksum.

The shift register parallel loads bytes to its eight outputs and shifts every byte from PM2 SR 7 H to PM2 SR 0 H. In effect, this converts the 8-bit bytes to a 1-bit wide serial bit stream. Gate E55 inverts PM2 SR 0 H routing the bit stream to the M8970 output. For every byte, the least significant bit (LSB) appears first.

The shift register receives all 8-bit bytes except message and checksum from the pattern generation ROM. The four bits loaded into the pattern address register select the appropriate STI code applied by the generator. At the start of the response message transmission, the selected pattern is an all 0s byte. The tape drive clocks a steady stream of 0s onto the STI bus until the controller receiver is ready and a message byte is available for shifting.

The M8972 supplies the first message byte on the microbus data lines and causes the M8970 microbus interface to produce a negative PM6 SEL TMBO L pulse. The TMBUS/MICROBUS buffer then places the byte on the M8970 MBD lines, and the FIFO-in MUX routes the byte to the FIFO. The data clock control logic generates two positive clock pulses (PM4 CLK FIFO SI H and PM4 CLK EDC H) based on PM6 SEL TMBO L and microprocessor clock 8MC1 CLK H. Although EDC clocks are generated, the EDC character calculation is not valid during response message transmission. After PM6 SEL TMBO L goes low, the FIFO strobes in the byte at the leading (rising) edge of the next microprocessor clock pulse. Signal PM2 FIFO OR H goes high indicating to the ROM controller that the first message byte is available for shifting out.

To shift out a 4-byte frame, the pattern generation ROM supplies the low byte of the sync character. The shift register then goes into parallel load mode, and the next clock pulse loads the low sync byte into the shift register. The pattern generation ROM then supplies the high byte of the sync character.

After loading the low sync byte, the shift register strobes in three more frame bytes, one byte every eight clock pulses. The high sync byte is the next frame byte. After this byte is loaded, the pattern generation ROM is disabled and the shift register driver routes the message byte from the FIFO to the shift register. At the appropriate time, the shift register parallel loads the message byte. The M8971 strobes in the last bit of the sync character at PM2 SR 0 H in the same clock period that the LSB of the message byte appears on PM2 SR 0 H.

While the message byte is shifting out, the FIFO receives a FIFO-out clock pulse, deleting the byte from the FIFO. The shift register driver is then disabled and the pattern generation ROM supplies the start hamming code. The code is then parallel loaded and shifted out. Note that PM2 FIFO OR H stays low until a new message byte or checksum byte is available at the FIFO output.

After the hamming code, the all 0s byte appears at the shift register input. The TA78 shifts out at least 80 zeroes before the first bit of the next frame appears.

The cycle continues as long as a new message byte is available at the FIFO output. Starting with the second frame, the M8970 assigns a continue hamming code to any frame that contains a message byte. The M8972 sets a flag to indicate that the last message byte has been loaded into the FIFO. The M8970 shifts out extra 0s. The M8972 then clears the flag allowing the M8970 to shift out any remaining message bytes in the FIFO.

The M8972 sets another flag to indicate the checksum has been loaded. The M8970 then checks PM2 FIFO OR H before adding the hamming code. The FIFO Output Ready signal stays low after the checksum has been deleted from the FIFO since the FIFO is now empty. The pattern generation ROM then applies the end hamming code to the shift register. And the end hamming code is shifted out right after the checksum.

Error Detection Code (EDC) Logic

The EDC logic calculates a 16-bit EDC character based on read or write data on the MBD lines. For read data transmission, the EDC character also includes the byte count. The logic implements an XOR and shift left algorithm to yield the EDC character. Gates E30 and E36 form the XOR network. Registers E31, E23, and E26 are involved with shifting and temporary storage. The remaining circuitry involves bits 8 and 0, and initialization.

At the start of a record, all registers and flip-flops are cleared to 0, and PM5 INIT EDC H is asserted. Therefore, in the initial EDC character, bits 6, 2, and 0 are 1s and the remaining bits are 0s. During a write data reception, the EDC logic receives data from the M8970 data path. During a read data transmission, the EDC logic receives data from the M8959 and the byte count from the M8972. In either operation, the EDC logic receives a positive clock pulse (PM4 CLK EDC H) from the data clock control logic for every byte that appears on the MBD lines.

To start the calculation, the low byte of the first HSC word appears on the MBD lines and is XORed with the low byte of the initial EDC character. At the leading (rising) edge of the first clock pulse, E31 strobes in the XORed values and E26 strobes in all 0s. In effect, E31 shifts the XORed low byte to the left so bits 0—7 become bits 1—8. At the trailing (falling) edge of the first clock pulse, E23 strobes in bits 1—7 and flip-flop E17 strobes in bit 8 of the new EDC character. In addition, flip-flop E47 negates PM5 INIT EDC H so PM5 EDC <6,2,0> H reflect E26 and E31 for the rest of the operation. The all 0s high byte of the initial EDC character is now fed back for XORing with the high byte of the first HSC word.

At the leading (rising) edge of the second clock pulse, E31 shifts XORed bits 8—15 to 9—15 and 0 while E26 strobes in bits 1—7. The high byte is at PM5 EDC <15:8> H (E31, E17) and the low byte is at PM5 EDC <7:0> H (E26, E35). The EDC character for the first HSC word is now valid. At the trailing (falling) edge of the second clock pulse, E23 strobes in bits 9-15 but E17 is not affected. The cycle continues until all the data has been transferred and included in EDC calculation.

In general after an odd-numbered clock pulse, E31 contains bits 8—1 of the new EDC character, and E26 contains bits 15—8 of the old character. After an even-numbered clock pulse, E31 contains bits 15—9 and 0 of the new EDC character and E26 contains bits 7—1 of the new character. Gate E45 determines whether to feed back bit 0 or 8 of the present EDC character to generate an XORed byte for the new EDC character.

Gate E35 reflects E31 and produces bit 0 at PM5 EDC 0 H only after the even-numbered clock pulses, since only then is the shifted bit 15 in E31. Gate E41 combines with E23 to yield a divide-by-two counter producing clock pulses for E47 and E17. Flip-flop E17 clocks in data only at the trailing edge of an odd-numbered clock pulse, when E31 contains the shifted bit 7. Therefore, bit 8 is already valid when the even-numbered clock pulse arrives.

The EDC character is valid only after an even-numbered clock pulse. If a record has an odd-numbered byte count, an all 0s fill byte appears on the MBD lines after the data stream. The extra clock pulse that comes with the fill byte yields a valid EDC character. However, the fill byte is included in the EDC character calculation.

In a write data reception, the M8972 reads the calculated EDC character through the EDC buffer using two data cycles, one for each byte. In a similar manner, the M8972 reads the HSC supplied EDC character through the TMBUS/MICROBUS buffer. The M8972 then compares the calculated and HSC supplied EDC characters. A character match indicates no detected transmission or protocol error in the data stream. Note that the supplied EDC character includes the all 0s fill byte.

In a read data transmission, the M8972 supplies a 16-bit byte count by means of the TMBUS/MICROBUS buffer. The byte count is included in the EDC calculation. The M8972 reads the calculated EDC character through the EDC buffer and then, by means of the TMBUS/MICROBUS buffer, supplies the EDC character to the MBD lines for shifting to the HSC.

Data Clock Control Logic

The data clock control logic generates: two data path FIFO clock pulses (PM4 CLK FIFO SI H and PM4 CLK FIFO SO H), the EDC logic clock pulse (PM4 CLK EDC H), and two error signals (PM4 Data Late H and PM4 BD ERR L).

Signal PM4 CLK FIFO SI H is the FIFO-input clock and PM4 CLK FIFO SO H is the FIFO-output clock. The logic generates both FIFO clock pulses during each of the four information transfer operations through the M8970 data path. The signals are generated in accordance with control inputs from: the ROM controller, microbus interface, data path, M8959 module and M8972 module. Table 2-6 lists the key signals for generating the FIFO clock pulses in each type of information transfer.

Table 2-6 FIFO Clock Signal Sources

Operation	FIFO Clock Sources PM4 CLK FIFO SI H	PM4 CLK FIFO SO H
Write data reception	PM1 Hold REG Mode S0 H PM4 FIFO IR L	WMC4 SCLK H
Read data transmission	WMC4 SCLK H	PM1 STB FIFO SO L
Command message reception	PM1 STB FIFO SI L	PM6 SEL TMBI L
Response message transmission	PM6 SEL TMBO L	PM1 STB FIFO SO L

Three signals in Table 2-6 control transfers on the MBD lines. Signal WMC4 SCLK H involves data transfers to or from the M8959. Signals PM6 SEL TMBI L and PM6 SEL TMBO L involve message byte transfers to or from the M8972 by means of the TMBUS/MICROBUS buffer. For message byte transfers, the microprocessor clock (8MC1 CLK H) provides additional sequencing for the FIFO clocks.

Two additional signals from the microbus interface indicate direction of transfer through the data path. Signal PM6 SEL INT CLK (RD) H is high, enabling gate E60 when data or message bytes are transferred from the MBD lines to the M8971. This is the case during a read data or response message transmission. Conversely, PM6 SEL EXT CLK (WRT) H is high, enabling E60 when data or message bytes are transferred from the M8971 to the MBD lines. This is the case during write data or command message reception.

Asserting one of the three MDB transfer control signals causes the data clock control logic to generate a positive PM4 CLK EDC H clock pulse. This is the clock for the EDC logic. The EDC logic receives its clock pulse when the FIFO receives the appropriate clock involving an MBD line transfer. Although the EDC logic receives clock pulses during all four transfer operations, the resulting EDC character is only valid for data operations.

Four signals in Table 2-6 control the FIFO clocks to regulate transfers to and from the M8970 shift register. Signals PM1 STB FIFO SI L and PM1 STB FIFO SO L are strobe signals from the ROM controller. These signals are issued in accordance with the microprogram for the transfer operation. The hold register mode signal (PM1 HOLD REG S0 H) from the ROM controller, and the FIFO-input ready signal (PM2 FIFO IR L) from the data path, are the basis for the FIFO-input clock in autoshift mode during a write data reception.

Dual J-K flip-flop E63 and flip-flop E53 are the major components used in autoshift mode to generate the FIFO-input clock. In autoshift mode, PM6 AUTO SI MODE H is high, allowing the hold register mode signal to affect E63. Every eight clock pulses, PM1 Hold REG S0 H goes high, placing the data path hold register in parallel load mode. The next clock pulse strobes a 1 into E63 at the same time that a hold register strobes in a new data byte. If the FIFO input is also ready, gate E50 pin 3 goes low, and E53 samples it, strobing in a 0. At the start of the next clock cycle, the other half of E63 strobes in a 1, asserting PM4 RST HR FLG L. This causes the data clock control logic to generate a positive FIFO-input clock pulse so the data path FIFO strobes in the new byte. Asserting PM4 RST HR FLG L also clears E63 in preparation for the next time the hold register goes into parallel load mode. In summary, the FIFO normally strobes in the new data about 1 clock cycle after the hold register strobes in the data.

When the FIFO is full, the FIFO-input ready signal remains negated. If this is the case for more than four M8970 clock cycles, the M8971 inhibits the STI line clock, the HSC inhibits data tansmission, and the M8970 clock halts. Therefore, the FIFO receives no FIFO-input clocks to strobe data into a full FIFO. The operation restarts about two clock cycles after the FIFO input is ready.

With the clock running, the FIFO input must be ready within 6.5 clock cycles after the hold register strobes in new data. Otherwise E63 is still set to 1 when the hold register goes into parallel load mode again. Flip-flop E47 then strobes in a 1, causing the M8970 to issue a data late signal (PM4 Data Late H asserted) and a board error signal (PM4 BD ERR L asserted). Therefore, a data late indicates that the hold register has strobed in new data before the FIFO could strobe in the previous data.

Strobing a 0 into either part of dual flip-flop E44 also sets E47, issuing data late and board error signals. This is the case if an attempt was made to retrieve unavailable data from the FIFO, or an attempt was made to store data in the FIFO before the input was ready. The data clock control logic issues data late and board error signals until the M8970 is cleared (PM6 CLR PC L asserted) at the end of the operation. Note that data late is classified as a formatter hardware error.

Asserting the board error signal (PM4 BD ERR L) indicates to the M8971 that an M8970 error has occurred. The M8971 then issues a composite error signal to the M8972 through the M8970 PSTI buffer. In addition to data late, two other conditions cause the assertion of PM4 BD ERR L. The first condition is a CROM parity error (PM1 CROM PAR ERR H asserted) which indicates an M8970 microprogram parity error. This is classified as a formatter hardware error. The second condition is a level 1 protocol error (PM1 Protocol ERR H asserted). This indicates that after the first frame of a level 2 message, a received hamming code is not recognized as a valid continue or end code. A level 1 protocol error is classified as an STI transmission error. The ROM controller generates both these error signals and once issued an error signal remains asserted until the M8970 is cleared at the end of the operation.

Two diagnostic signals from the microbus interface asynchronously control PM4 RST HR FLG L through one part of E63. The microbus interface asserts PM7 FRC SI H, forcing PM4 RST HR FLG L to go low and stay low. This causes the FIFO-input clock to go high so the FIFO strobes in 1 byte. Conversely, the microbus interface asserts PM7 FRC Data Late H, forcing PM4 RST HR FLG L to stay high. Therefore, E63 is never cleared, and the data clock control logic issues the data late and board error signals.

Clock Generator

The M8970 clock generator produces seven clock signals for the M8971 and other parts of the M8970. The generator also produces a Clock Enable signal for the M8971 and has a 64 count delay counter.

All the clocks are based on a 33.33 MHz crystal oscillator. In normal operation (PM7 PMIC RUN L = LO), dual flip-flop E48 divides the crystal frequency by four, yielding clocks PM3 XCLK H, PM3 XCLK L, and PM3 FCLK H. The clocks are all square waves with a period of about 120 ns. Signal PM3 XCLK H is inverted with respect to the other two clocks and is therefore 180 degrees out of phase. The M8970 data clock control logic uses signal PM3 FCLK H while M8971 uses PM3 XCLK H and PM3 XCLK L.

Signal PM3 XCLK L also drives a control circuit consisting of flip-flop E54, shift register E46, and gate E45. During every clock cycle, E54 samples the level at E58 pin 6 and E46 samples the inverted flip-flop output.

Signal PM3 XCLK H is the master clock, serving as the basis for the M8972 STI line clock and other M8970 clocks. Based on PM3 XCLK H, the M8971 clocks all 0s onto the enabled STI read/response data line while the tape drive is idle. However, no other M8970 clock based on the master clock is produced until the M8972 activates the M8970.

To activate the M8970, the M8972 causes the microbus interface to assert PM6 ENBL PMIC H. Since the M8970 is never initially in autoshift mode, E58 pin 6 then goes low. Within two clock cycles, gate E45 asserts PM3 PCLK ENBL L, enabling the remainder of the clock generation circuit. Asserting PM3 PCLK ENBL L also causes the microbus interface to negate a clear signal (PM6 CLR PC L), allowing the M8970 to react to the clocks.

The remainder of the circuit produces four clocks: PM3 PCLK1 L, PM3 PCLK2 H, PM3 PCLK15 L, and PM3 CROM LTCH L. All four are used by the ROM controller. Signal PM3 PCLK2 H is also the clock for the shift register and hold register in the data path. And PM3 PCLK1 L is also used by the data clock control logic.

For a read data or response message transmission, the microbus interface holds PM6 SEL EXT (WRT) H at a low signal level. Therefore, PM3 PCLK ENBL L remains low, and the four remaining M8970 clocks are based on the master clock (PM3 XCLK H). So PM3 PCLK1 L and PM3 PCLK2 H are square waves with periods of about 120 ns but are 180 degrees out of phase. Delay line E59 introduces a phase lag to produce the other two clocks. Signal PM3 PCLK15 L lags the leading (falling) edge of PM3 PCLK1 L by about 15 ns. Signal PM3 CROM LTCH L is a positive pulse about 45 ns wide that lags the leading (falling) edge of PM3 PCLK1 L by about 30 ns.

In addition to the four M8970 clocks, the M8971 STI line clock is also based on master clock PM3 XCLK H. Therefore, data or message frames are clocked through the M8970 data path, through the M8971, and onto the enabled STI read/response data line at the master clock rate. However, during read data transmission, the ROM controller inhibits the data path and STI line clock if the data path FIFO is empty for more than five clock cycles.

For a write data or command message reception, PM6 SEL EXT CLK (WRT) H is high. This yields two results. First, the four M8970 clocks are based on an external clock derived from the incoming message or data rate. The second result is that the M8971 clocks a stream of 0s onto the enabled STI read/response data line based on the master clock. The HSC in turn uses the read/response clock rate to transmit command messages or data. So in effect, the four M8970 clocks are based on the master clock even during write data or command message reception. However, the clock phase is shifted because of cable propagation delay. Using the external clock allows the M8970 to compensate for phase differences, matching control of the M8971 and M8970 data path to the incoming information.

For a command message reception, the M8970 is never in autoshift mode, and the stream of 0s on the enabled STI read/response data line is not interrupted. The maximum number of message frames is 64, and the capacity of the data path FIFO is 64 bytes. Therefore, there is no danger of FIFO overflow. However, this is not the case for a write data reception.

After activating the M8970 for a write data reception, the M8972 monitors PM3 PCLK ENBL L through the microbus interface. When this signal is low, it indicates all clocks are running so the HSC can transmit data. The M8972 then places the M8970 and M8971 in autoshift mode (PM6 Auto SI Mode H = HI). Signal PM3 PCLK ENBL L and the STI line clock are then controlled by the data path FIFO-input ready signal (PM2 FIFO IR L). If the FIFO is full, PM2 FIFO IR L stays high, and E54 strobes in a 1 for every clock cycle. If PM2 FIFO IR L stays high for four clock cycles, E46 contains three 0s, and E45 disables PM3 PCLK ENBL L. The M8971 then stops sending 0s on the enabled STI read/response data line, and the HSC stops transmitting data. Therefore, the four M8970 clocks are inhibited, since these clocks are based on the incoming data rate. The ROM controller halts, the data path shift register and hold register halt, and no FIFO-input clocks are generated. So data can only be retrieved from the FIFO.

When the FIFO is no longer full, PM2 FIFO IR L goes low. After two clock cycles, E46 contains a 1, and E45 enables PM3 PCLK ENBL L. The M8971 resumes clocking 0s on the enabled STI read/response data line, the HSC resumes data transmission, and the M8970 ROM controller and the full data path restart. The M8972 takes the M8970 and M8971 out of autoshift mode before deactivating the M8970.

To deactivate the M8970, the M8972 causes the microbus interface to negate PM6 ENBL PMIC H. After four clock cycles, E45 disables PM3 PCLK ENBL L. Since the M8970 is never in autoshift mode at this time, the microbus interface then asserts PM6 CLR PC L, which halts and resets the M8970 logic.

For diagnostic purposes, PM7 PMIC RUN L is high, allowing single stepping of the clock circuitry. In accordance with the M8972, the microbus interface toggles PM7 PMIC STEP H once for every clock pulse generated.

The 64 count delay counter consists of a dual 4-bit binary counter (E4), which is wired as a single 8-bit counter. The counter is used during response message transmission. Signal PM3 PCLK2 H is the counter clock that is also used by the data path. The ROM controller asserts PM1 RST DLY L to start the counter. Sixty-four clock periods later, E4 asserts PM3 DLY H. This indicates to the ROM controller that enough time has passed to clock 64 0s onto the enabled STI read/response data line. The ROM controller then negates PM1 RST DLY L clearing the counter.

ROM Controller

The ROM controller directs the data path and other M8970 logic to implement four types of information transfer and port switching. The major component of the ROM controller is a 512 8-bit ROM (E19). The ROM contains eight microprogram routines, with each routine associated with an operational or diagnostic function.

The ROM receives nine address bits and a strobe signal (PM3 CROM LTCH L). The three most significant address bits are PM6 PC <8:6> H, which select one of eight microprogram routines. The remaining six address bits (PM1 PC <5:0> H) come from the program counter. The program counter, in turn, consists of two synchronous 4-bit binary counters (E32 and E37) wired to act as one 8-bit counter.

At the start of an operation, the M8972 provides address bits PM6 PC <8:6> H by means of the microbus interface, selecting the appropriate microprogram routine. By means of the microbus interface, the M8972 then activates the clock generator and allows the other logic sections to function (PM6 CLR PC L negated). The ROM controller receives four clocks from the clock generator. These clocks synchronize the ROM controller with the data path and data clock control logic.

As long as no branch instruction appears, the program counter increments the ROM address at the start of every clock cycle. The ROM strobe is negated, and a new microprogram instruction appears on the ROM output lines (PM1 CROM <7:0> H). The ROM strobe is then asserted again, latching the new instruction at the output. Signals PM1 CROM <5:0> H are decoded to generate control signals in accordance with the microprogram instruction. For a non-branching instruction, PM1 CROM 6 H is always a 0. As long as this is the case, the contents of the ROM are retrieved in sequence.

The ROM controller allows conditional or unconditional branching based on synchronous or asynchronous inputs. Branching circuitry includes two 8:1 multiplexers: the synchronous branch MUX (E73) and the asynchronous branch MUX (E21). Signal PM3 PCLK2 H synchronizes the output of the asynchronous branch MUX through flip-flop E54. In effect, the two branch MUXs are cascaded through the flip-flop. Each MUX also has a register associated with it that contains ROM outputs PM1 CROM <2:0> H from a previous instruction. These registers select the desired multiplexer input.

For a branch instruction, PM1 CROM 6 H is a 1, and PM1 CROM <5:0> H is the branch address. A logical 1 on PM1 CROM 6 H enables E73, and, if the selected condition is true, E73 pin 6 goes low. At the start of the next clock cycle, the program counter then parallel loads PM1 CROM <5:0> H and the program branches, retrieving the specified location. Note that selecting input 1 of E73 yields an unconditional branch instruction since this input is always true (that is, tied to a high signal level).

Selecting input 0 of E73 specifies the asynchronous conditions. A separate instruction loads PM1 CROM <2:0> H into E29, selecting one of the asynchronous inputs. If the selected asynchronous input is true, E54 strobes in a 1 at the start of the next clock cycle. If a 1 then appears on PM1 CROM 6 H, E73 enables the program counter to parallel load at the start of the following cycle. The ROM branches to the new address.

Parity generator/checker E27 calculates the parity of the ROM outputs. Signal PM1 CROM 7 H is the parity bit. The assertion of PM1 CROM ODD PAR H indicates that the parity of the ROM outputs is odd. This is the correct parity. If even parity is calculated, J-K flip-flop E66 strobes in a 1, indicating a parity error. The program counter is inhibited from incrementing or parallel loading, and PM1 CROM PARR ERR H goes high, causing the data clock control logic to issue the M8970 board error signal to the M8971. The M8971 then issues a composite error signal to the M8972 through the M8970 PSTI buffer. The assertion of PM1 CROM PARR ERR H also reports the parity error to the M8972 by means of the PERR register on the M8971. Once issued, the ROM controller generates the parity error signals until E66 is cleared at the end of the current operation.

For diagnostic purposes, PM7 ENB CRMO H is high, disabling the ROM. By means of the microbus interface, the M8972 then supplies program instructions onto the PM1 CROM <7:0> H lines.

Port Control Logic

In accordance with the M8972, the port control logic generates four control signals for the M8971. The M8970 ROM controller contains the port switch routine, which is used in conjunction with the port control logic. Figure 2-10 shows the port control logic and two associated registers/buffers from the microbus interface.

The M8972 periodically reads the PSWI buffer to determine the state of the TS78 control panel switches. Bits 6, 5, and 4 represent the Fault, Port Select A and Port Select B switches respectively. Reading a 0 for one of these bits indicates that the corresponding switch is in the enabled (in) position.

The M8972 periodically causes the M8970 to execute the port switch routine so the enabled STI port reflects the TS78 control panel switch configuration. However, the routine is run even if the switch configuration has not changed. Before running the routine, the M8972 loads the appropriate code into the PRTO register. The M8972 then supplies the port switch routine address and activates the M8970. The ROM controller then executes the port switch routine.

In accordance with the routine, the ROM controller sets the port transition bit. At the start of the next clock cycle, J-K flip-flop E66 strobes in a 1 asserting PM3 INH STATUS S/R L. Asserting this signal clears a previous data-line pulse error condition, resets the M8971 deserializer circuitry, and causes the M8971 to send all 0s on the presently enabled STI real-time formatter-state line(s).

The ROM controller also causes the M8970 data path to shift out a stream of 0s. The M8971 in turn sends these 0s on the presently enabled STI read/response data line.

Eighteen clock cycles after setting port transition, the ROM controller clears the port transition bit. At the start of the next clock cycle, E66 strobes in a 0. Signal PM3 INH Status S/R L goes high, and the port control sync register strobes in the code stored in PRTO. In accordance with the code, the port control sync register asserts the appropriate signals, causing the M8971 to enable the desired port. Table 2-7 shows the PRTO codes, the asserted control signals after running the port switch routine, and the functional effects based on the various TS78 control panel switch configurations.

When the ROM controller finishes running the port switch routine, the M8972 reads the PSWI register to determine the state of the M8970 port control outputs. The M8972 then compares the code for these outputs to the port control code put in PRTO. A match indicates the port switch routine was successful.

For a TS78 master reset or power-up, the M8972 causes the microbus interface to assert PM6 Clear L for several microprocessor clock cycles. This clears the port control sync register. All the M8970 port control outputs are negated and both STI ports are off-line. Note that the port control sync register and therefore the STI port lines are not affected when the M8970 is deactivated (PM6 CLR PC L asserted).

Figure 2-10 M8970 Port Control Logic and Associated Registers

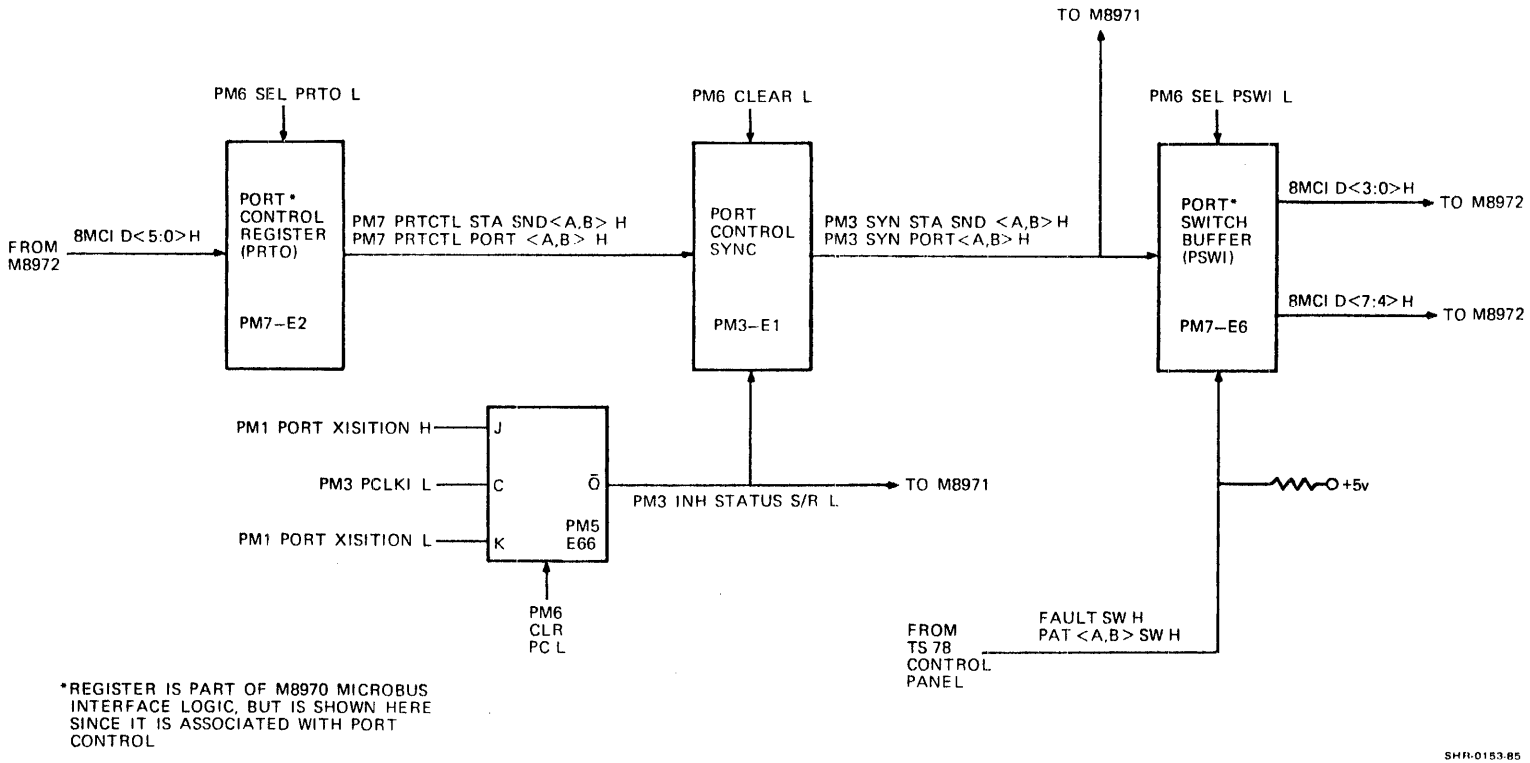


Table 2-7 STI Port Control

Switch Position Port Select A	Port Select B	Port Control (PRTO) Code	Asserted M8970 Port Control Output(s)	Function
Disabled	Disabled	00	--	Formatter is off-line; formatter reception and transmission disabled on port A and port B data lines and real-time state lines.
Enabled	Disabled	0A	PM3 SYN Port A H	Formatter reception and transmission enabled on port A data lines, and formatter reception enabled on port A real-time controller-state line.
			PM3 SYN STA SND A H	Formatter transmission enabled on port A real-time formatter-state line.
Disabled	Enabled	05	PM3 SYN PORT B H PM3 SYN STA SND B H	Definition of these two signals for port B is analogous to definition of previous two signals for port A.
Enabled	Enabled	0E or 0D	PM3 SYN PORT <A,B> H*	Formatter reception and transmission enabled on port A or port B data lines, and formatter reception enabled on port A or port B real-time controller-state line.
			PM3 SYN STA SND <A,B> H	Formatter transmission enabled on port A and port B real-time formatter-state lines.

*The M8970 asserts the two port control signals alternately at 4.2 ms intervals sampling each port, until one receives a command to go on-line. The formatter-state bits reflect the active port.

Formatter Receiver Ready (FRR) Control Logic

The FRR control logic generates two signals used by the M8971: PM6 FRR H and PM6 VALID L. Signal PM6 FRR H, when loaded into the M8971 serializer logic, becomes the formatter receiver ready state bit. Signal PM6 VALID L is a clear signal for the serializer logic.

uJ-K flip-flop E33, flip-flop E53 and several gates make up the FRR control logic. When the M8970 is deactivated, PM6 CLR PC L is asserted and PM1 INH PRO H is negated. Both flops are cleared so PM6 FRR H and PM6 VALID L are both low. The formatter receiver ready state bit is a 0 on the enabled STI real-time formatter-state line(s). When the M8970 is activated, the FRR control logic is controlled by the ROM controller or the M8972 by means of the microbus interface.

The M8972 controls the FRR control logic through microbus interface signals PM6 SFRR H and PM6 SEL ENPMO L. Signal PM6 SFRR H is bit 6 in the PCTO register. The M8972 specifies this bit along with the operation and other initial conditions. A 1 for this bit indicates the formatter receiver ready bit should be set at the start of the operation. Signal PM6 SEL ENPMO L is a key signal in the activation of the M8970.

To activate the M8970, the M8972 causes the microbus interface to provide a negative PM6 SEL ENPMO L pulse. This negates PM6 CLR PC L and enables the clock generator. The FRR control logic, as well as other M8970 logic sections, can then operate. At the trailing (rising) edge of PM6 SEL ENPMO L, E33 in effect samples PM6 FRR H to specify the state of the formatter receiver ready bit. For a write data or command message reception, E33 strobes in a 1, asserting PM6 FRR H, so the formatter receiver ready bit then becomes a 1. This indicates to the HSC that the formatter is ready to receive a new message frame or data. Note that when an operation is in progress, the ROM controller asserts PM1 IN PRO H inhibiting PM6 SEL ENPMO L from affecting E33.

The ROM controller affects the FRR control logic with two strobe signals (PM1 STB FRR 0 L and PM1 STB FRR 1 L). The ROM controller provides a negative PM1 STB FRR 0 L pulse to clear E33 so formatter receiver ready becomes 0. This is the case when the formatter receives a sync character during a write data or command message reception. Conversely, the ROM controller provides a negative PM1 STB FRR 1 L pulse to set E33 so formatter receiver ready becomes 1. This is the case when a start or continue frame has been processed during command message reception.

Flip-flop E53 controls the PM6 Valid L signal. Once every clock cycle, E53 samples the level at E39 pin 8. When the FRR control logic receives a negative pulse that can change the state bit, a positive pulse appears at E39 pin 8. E53 strobes in a 1 for at least one clock cycle, producing a positive PM6 Valid L pulse. This causes the M8971 to send 0s on an enabled STI real-time formatter-state line(s). The 0s continue until the M8971 serializer logic clocks in a new state byte. The logic clocks in a new byte, with possibly a new formatter receiver ready bit, within 16 clock cycles of the trailing (falling) edge of PM6 Valid L.

To change more than the formatter receiver ready bit, the M8972 writes into the FSTO register. To access this register, the microbus interface provides a negative PM6 SEL CSTO L pulse. This also causes a positive PM6 Valid L pulse in preparation for transmission of a new state byte.

Microbus Interface

The microbus interface allows the M8972 to control and monitor the M8970 and M8971. The microbus interface contains decoding logic, registers and buffers, and activating/deactivating logic.

The decoding logic includes two 3:8 decoders (E11, E8), a dual 2:4 decoder (E12) and associated gates. For a microbus data transfer, the M8972 supplies an enabling signal (8MC2 SEL PMIC L) and the four least significant address bits (8MC1 A<3:0> H). The M8972 then supplies a positive pulse for 8MC1 Read Strobe H or 8MC1 Write H to specify the transfer as a read or write operation. For a write operation, the M8972 also supplies data on the microbus data lines (8MC1 D <7:0> H).

Asserting 8MC2 SEL PMIC L causes the microbus interface to decode the four least significant address bits and the read/write signals. The decoding yields a negative pulse for one of 22 signals. Seventeen of these signals select data buffers or registers, four of which are on the M8971. The remaining five of 22 are control signals. Table 2-8 lists the activating signal and selected register/buffer for all M8970/M8971 locations.

Three of the five control signals (PM6 SEL ENCLKO L, PM6 SEL CLCLKO L, and PM6 SEL CLIFO L) control the interrupt timer and are discussed in the next section — Interrupt Timer. The remaining two control signals are used to activate and deactivate the M8970. Activation/deactivation logic includes J/K flip-flop E33 and associated gates. Note that the M8972 goes through a specified sequence just before activating the M8970.

In normal operation, just before activating the M8970, the M8972 reads location 84 to make sure the M8970 is reset. The M8972 also clears the protocol diagnostic (PDIO) register and the protocol error (PERI) register on the M8971. The M8972 then writes a command code into the protocol control (PCTO) register. The command code specifies a microprogram routine address and other initial conditions for the desired operation.

To activate the M8970, the M8972 writes into location 84. The microbus interface decoder logic then provides a negative PM6 SEL ENPMO L pulse which sets E33. Signal PM6 ENBL PMIC L then goes low and two clock cycles later the clock generator asserts PM3 PCLK ENBL L. This enables four clocks used within the M8970 and causes the microbus interface to negate PM6 CLR PC L. This in turn allows the other M8970 logic sections to operate. Note that at the trailing (rising) edge of PM6 SEL ENPMO L, the FRR control logic samples PCTO bit 6 to determine the initial state of the formatter receiver ready state bit.

Table 2-8 Microbus Interface Address Selection

Microbus Address	Read Operation Decoded Signal	Selected Register of Buffer	Write Operation Decoded Signal	Selected Register or Buffer
80	PM6 SEL PSTI L	Protocol Status (PSTI)	PM6 SEL PCTO L	Protocol Control (PCTO)
81	PM6 SEL PERI L	Protocol Error (PERI)*	--	--
82	PM6 SEL CSTI L	Controller State (CSTI)*	PM6 SEL CSTO L	Formatter State (FSTO)*
83	PM6 SEL TMBI L	TMBUS/MICROBUS (TMB)	PM6 SEL TMBO L	TMBUS/MICROBUS (TMB)
84	PM6 SEL CLPMI L†	--	PM6 SEL ENPMO L†	--
85	PM6 SEL PSWI L	Port Switch (PSWI)	PM6 SEL PRTO L	Port Control (PRTO)
86	PM6 SEL PCI L	Program Counter (PCI)	PM6 SEL PDIO L	Protocol Diagnostic (PDIO)
87	PM6 SEL CRMI L	Control ROM Input (CRMI)	PM6 SEL CRMO L	Control ROM Output (CRMO)
88	PM6 SEL EDLI L	EDC Low (EDLI)	PM6 SEL CLCLKO L†	--
89	PM6 SEL EDHI L	EDC High (EDHI)	PM6 SEL ENCLKO L†	--
8A	PM6 SEL IDNI L	Identification (IDNI)	PM6 SEL LITO L	Light (LITO)*
8B	--	--	PM6 SEL CLIFO L†	--

* Located on M8971

† Control signal

Signal PM3 PCLK ENBL L stays asserted during the entire operation in all cases except for autoshift mode in a write data reception. In autoshift mode, this signal is negated if the data path FIFO is full for four or more clock cycles. However, PM6 Auto SI Mode H is high in autoshift mode, preventing the microbus interface from asserting PM6 CLR PC L. This prevents the M8970 from shutting down even though PM3 PCLK ENBL L is negated. In a write data transmission, the M8970 is taken out of autoshift mode before it is deactivated.

The M8972 reads the protocol status (PSTI) buffer, in most cases, to determine if an operation is finished or if an error has occurred. If an error has occurred, the M8972 reads the PERI register on the M8971. The M8972 then takes appropriate action based on the error.

To deactivate the M8970, the M8972 reads location 84. This causes the microbus interface decoder logic to provide a negative PM6 SEL CLPMI L pulse. At the leading (falling) edge of PM6 SEL CLPMI L, E33 strobes in a 0, negating PM6 ENBL PMIC L. Four master clock cycles later, four M8970 clocks are disabled, and the microbus interface asserts PM6 CLR PC L, resetting the M8970.

In addition to PSTI, PCTO and PDIO, the M8970 contains three addressable registers and seven addressable buffers. These registers and buffers are discussed in the following paragraphs.

The TMBUS/MICROBUS (TMB) buffer provides a bidirectional path between the M8970 MBD lines and the microbus data lines. A negative pulse for either PM6 SEL TMBO L or PM6 SEL TMBI L enables the buffer. Signal PM6 SEL TMBO L specifies the direction of transfer. For example, a negative pulse for this signal specifies a transfer from the microbus data lines to the MBD lines. Signals PM6 SEL TMBO L and PM6 SEL TMBI L are also used by the data clock control logic to produce the FIFO-input clock and FIFO-output clock during read message transmission and command message reception.

The port control (PRTO) register and port switch (PSWI) buffer are associated with STI port control and are discussed in the Port Control Logic section. The EDC low (EDLI) and EDC high (EDHI) buffers, when enabled, place the calculated EDC character on the microbus data lines. This is the case during a write data reception or read data transmission. Each EDC buffer is associated with one byte of the character.

The M8972 periodically reads the identification (INDNI) buffer to determine the state of eight individual switches in the DIP switch on the TS78 backplane. The DIP switch selects a base number between 0 and 255 which is the unit number for transport 0 (that is, the transport in the cabinet). Each individual switch is associated with a binary-weighted value. Reading a 1 for a buffer bit indicates the corresponding individual switch is off. Note that the unit number for any transport connected to the TS78 is the base number plus the TU port number (0, 1, 2, or 3) to which the transport is cabled.

The program counter (PCI) buffer, control ROM input (CRMI) buffer, and control ROM output (CRMO) register deal with the ROM controller and are used for diagnostic purposes only. The PCI buffer, when enabled, places the eight least significant bits of the ROM address onto the microbus data lines. The CRMI buffer, when enabled, places the microprogram instruction on the ROM output lines onto the microbus data lines. Note that the CRMO register has tri-state outputs.

To specify a microprogram instruction, the M8972 writes into the CRMO register. However, the M8972 must also set PDIO bit 3 to 1 (PM7 ENB CRMO H = HI). This disables the ROM and enables CRMO to place the M8972-specified microprogram instruction onto the ROM output lines (PM1 CROM <7:0> H).

The microbus interface logic inverts 8MC1 CLEAR H to produce PM6 CLEAR L. For a TS78 master reset or power-up, the M8972 asserts 8MC1 CLEAR H for several microprocessor clock cycles. The M8970 microbus interface asserts PM6 CLEAR L which in turn:

1. Clears E33 so the M8970 is deactivated and reset.
2. Clears the port control logic so both STI ports are off-line.
3. Clears the interrupt timer logic so no MCLK interrupts are issued or pending.
4. Clears the PDIO register so normal M8970 operation is specified.

Interrupt Timer

The interrupt timer logic generates a series of MCLK interrupts used by the M8972 to implement the port switch timer and command timer. This logic consists of two dual 4-bit decade counters (E13, E18) and two flip-flops (E17, E24). The two counters are wired to work as one 16-bit decade counter with its output at E13 pin 13. The counter clock is PM3 XCLK H, which has a period of about 120 ns. Flip-flop E17 produces PM6 MCLK INT L, which is the MCLK interrupt signal. The M8972 starts the timer, resets the timer, and causes negation of the interrupt signal by means of three microbus interface signals.

The number of MCLK interrupts generated is a measure of the elapsed time since the timer was started. The first MCLK interrupt is 600 us after starting the timer. Subsequent MCLK interrupts are at 1.2 ms intervals until the timer is cleared. Therefore, an interrupt count of 4 corresponds to 4.2 ms which is the port switch time.

To set up a time interval, the M8972 first writes into locations 88 and 8B to make sure the timer is reset and the interrupt signal is negated. The M8972 then initializes a software counter with the interrupt count corresponding to the desired time interval. (For example, the software counter is initialized to 4 for the port switch time interval.) The M8972 then starts the interrupt timer.

To start the interrupt timer, the M8972 writes into location 89. This causes the microbus interface to provide a negative PM6 SEL ENCLKO L pulse which sets E24. Flip-flop E24 then allows the counter to start running. After 600 us (500 clock cycles), the counter output goes high clocking a 1 into E17. This issues the first MCLK interrupt (PM6 MCLK INT L asserted), indicating to the M8972 that the M8970 interrupt timer has timed out.

After receiving the first interrupt, the M8972 causes the negation of the interrupt signal (PM6 MCLK INT L negated). To accomplish this, the M8972 writes into location 8B. The microbus interface then provides a negative PM6 SEL CLIFO L pulse resetting E17. The M8972 also decrements its software counter by 1. Note that in most cases the timer is still running.

After the first interrupt, the rising edge of the counter output appears every 1.2 ms (1000 clock cycles) causing E17 to issue interrupts at 1.2 ms intervals. The M8972 continues to cause the negation of the interrupt signal and continues to decrement the software counter until it is 0. A software counter equal to 0 indicates the desired time interval has elapsed.

When the interval has elapsed, the M8972 clears the timer and causes the negation of the interrupt signal again. To clear the timer, the M8972 writes into location 88. The microbus interface then provides a negative PM6 SEL CLCLKO L pulse which resets E24. Flip-flop E24 then resets the counter to 0.

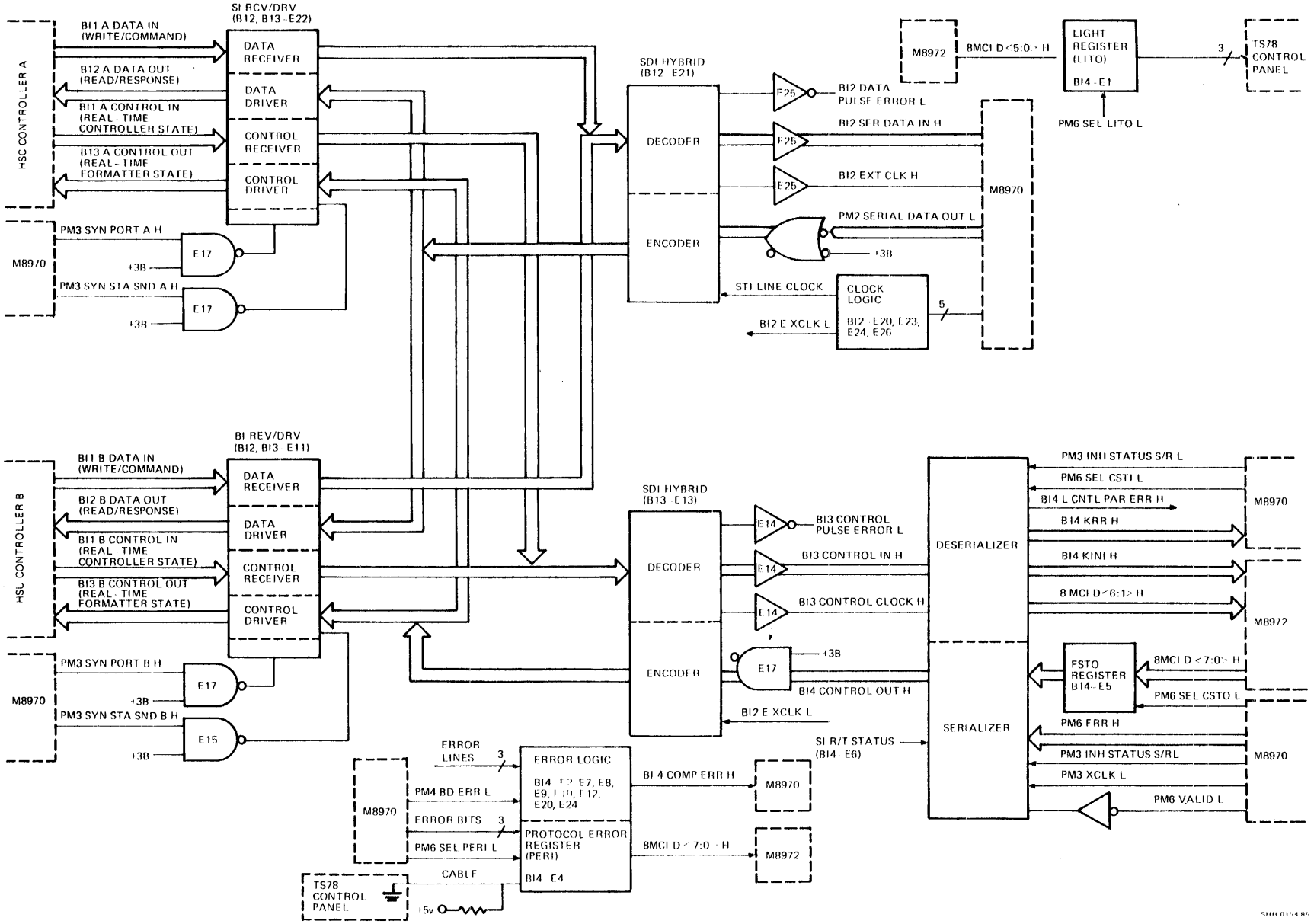
To implement the command timer, the M8972 loops several software counters. When the tape drive is on-line, the command timer is specified in seconds. Each second consists of 40, 25 ms intervals with every 25 ms interval corresponding to 22 interrupt counts. When the tape drive is available, the command timer is always a one second interval which consists of eight 125 ms intervals. The 125 ms intervals in turn correspond to 103 interrupt counts. Note that the command timer intervals are not exact since software counters must be integer values.

For a TS78 master reset or power-up, the M8972 causes the microbus interface to assert PM6 CLEAR L for several microprocessor clock cycles. Asserting PM6 CLEAR L resets E17 to make sure the interrupt signal is negated. When PM6 CLEAR L is negated, its rising edge clocks a 0 into E24, making sure the counter is reset. In effect, this cancels any pending interrupt.

2.3.3 M8971 STI Interface

The M8971 handles waveform processing for the port A and B STI bus lines. It also handles data conversions involving information on the STI real-time state lines. In addition, the M8971 includes STI error logic and four register/buffer locations in the microprocessor I/O address space. Three of these locations involve data conversion and error identification. The fourth location, the light (LITO) register, controls the three indicators on the TS78 control panel in accordance with the M8972. Figure 2-11 is a block diagram of the M8971.

Figure 2-11 M8971 Block Diagram



Five VLSI chips are the major components of the M8971: two 34-pin SI receiver/driver chips (E22, E11); two 28-pin SDI hybrid chips (E21, E13); and a 48-pin gate array (E6) designated the SI real/time status chip. The gate array has TTL input/output, but the other four VLSI chips have ECL input/output. These four chips are buffered by gates that convert between ECL and TTL levels.

The M8971 interfaces directly to the STI bus lines. (Coax cables from the HSC(s) plug into the tape drive bulkhead connector, one cable for each port used.) Two coax cables route port A and B STI bus lines from the bulkhead connector to two 8-pin, right-angle connectors (J1 and J2) on the M8971. Port A bus lines plug into J1, and port B bus lines plug into J2. Each coax cable has four differential STI bus lines. M8971 signal names are assigned to the STI bus lines at the connectors and are defined as follows.

M8971 Signal Name	STI Bus Line
BI1 A Data In +	Port A write/command data line BI1 A Data In
BI2 A Data Out +	Port A read/response data line BI2 A Data Out
BI1 A Control In +	Port A real-time controller state line BI1 A Control In
BI3 A Control Out +	Port A real-time formatter state line BI3 A Control Out
BI1 B Data In +	Port B write/command data line BI1 B Data In
BI2 B Data Out +	Port B read/response data line BI2 B Data Out
BI1 B Control In +	Port B real-time controller state line BI1 B Control In
BI3 B Control Out +	Port B real-time formatter state line BI3 B Control Out

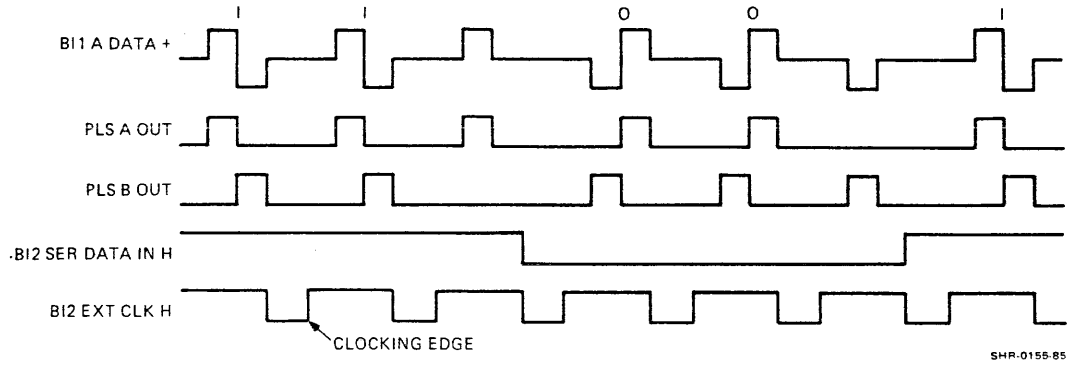
Each SI receiver/driver chip handles the STI bus lines for one port; E22 is for port A, and E11 is for port B. Each chip has two receiver circuits and two driver circuits. A receiver circuit, when enabled, converts a tri-level STI signal to bi-level signals on two output lines (PLS A Out, PLS B Out). Conversely, a driver circuit, when enabled, converts bi-level signals from two inputs (PLS A In, PLS B In) to a tri-level STI signal.

Four port control signals from the M8970 select which circuits in each chip are enabled. For each chip, one port control signal must be high to enable both receivers and one driver. A second port control signal must be high to enable the second driver. Therefore, two port control signals must be high to completely enable a port. (Refer to the paragraph Port Control Logic under Paragraph 2.3.2 for more port control information.)

Two SDI hybrids (E21 and E13) process STI information in conjunction with the SI receiver/drivers. Chip E21 processes data line information and chip E13 processes control line information. Each chip contains one decoder and one encoder. A decoder converts two bi-level signals from a receiver circuit to one NRZI output. The decoder also generates a clock signal based on the incoming pulse rate. In addition, the decoder contains circuitry to detect a pulse error (for example, a missing pulse). Conversely, an encoder combines a serial NRZI input and a clock signal to generate two bi-level signals (PLS A IN, PLS B IN). These two signals in turn are used by an enabled driver circuit(s).

Figure 2-12 shows the waveforms for a 1 and 0 on an STI line, the corresponding receiver output, and the corresponding decoder output. Encoding is not shown since it is basically the reverse of decoding.

Figure 2-12 Waveform Accessing of Received Information



STI Data Line Information Processing

Decoded write data or command message frames appear on the BI2 SER Data In H line and are routed to the M8970. The derived clock, BI2 EXT CLK H, is the basis for M8970 signals that clock data through part of the M8970. Therefore, a portion of the M8970 data path is synchronized to incoming information on the enabled port's STI write/command data line.

The M8971 receives read data or reponse message frames from the M8970 through the PM2 Serial Data Out L line. The encoder in E21 combines this serial input with the STI line clock. An enabled driver then places the encoded information on the corresponding STI read/response data line.

During a write data or command message reception, PM2 Serial Data Out L stays high and the encoder sends a stream of 0s to the data drivers. An enabled data driver places the 0s on an STI read/response data line. The HSC in turn uses the read/response clock rate to transmit command messages or data.

Flip-flop E20 and several gates form the M8971 clock logic that generates the STI line clock and BI2 E XCLK L. These two clock signals are based on the M8970 master clock (PM3 XCLK H). Signal BI2 E XCLK L is always running and is the clock signal for the encoder in E13. Most of the clock logic, in conjunction with four M8970 signals, allows inhibiting the STI line clock under certain conditions during data operations.

During a read data transmission, the M8971 shifts out data as long as the M8970 FIFO is not empty. (Flip-flop E20 samples PM1 INH SR CLK H once every clock cycle.) If the M8970 FIFO is empty for five clock cycles, the M8970 asserts PM1 INH SR CLK H, and at the leading (rising) edge of PM3 XCLK L, E20 strobes in a 1 inhibiting the STI line clock. When the clock stops, transmission stops on the enabled port's STI read/response data line. In addition, new data no longer appears on the PM2 Serial Data Out L line. After the FIFO output is ready again, the STI line clock is enabled, and data appears on the PM2 Serial Data Out L line. Therefore, data transmission on the enabled port's STI read/response data line resumes.

During a write data reception, the M8970 and M8971 go into autoshift mode (PM6 Auto SI Mode H = HI). This prevents overloading of the M8970 FIFO. If the FIFO is full for four clock cycles, the M8970 negates PM3 PCLK ENBL L which inhibits the STI line clock. The M8971 then stops sending 0s on the enabled port's STI read/response data line, and the HSC stops transmitting data. The derived decoder clock (BI2 EXT CLK H) stops and a portion of the M8970 data path is inhibited. Now data can only be retrieved from the M8970 FIFO. Two clock cycles after the FIFO input is ready again, the STI line clock is enabled. Then HSC transmission and formatter reception resume.

Control Line Information Conversion and Control

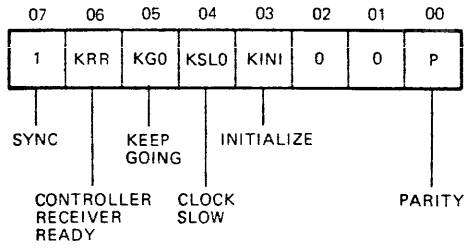
A 48-pin gate array, designated the SI real/time status chip, contains a deserializer and serializer. The deserializer performs serial-to-parallel conversion of decoded real-time controller state information. The serializer performs parallel-to-serial conversion of real-time formatter state information, which is then encoded. (The encoder clock, BI2 E XCLK L, is always running.)

Formatter state and controller state information is transmitted as a series of 16-bit state frames. A state frame consists of an all 0s preamble byte followed by a state byte. The preamble byte and bit 7 of the state byte are for synchronization. Bit 0 of the state byte is always a parity bit and bits 6—1 are the state bit locations. All six state bit locations are not used. Figure 2-13 identifies the bits in the controller state byte and formatter state byte. Note that the most significant bit is always transmitted first.

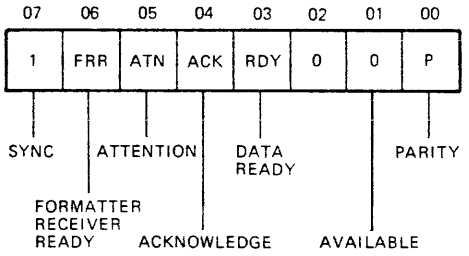
The deserializer receives the decoded real-time controller state information on the BI3 Control IN H line. The deserializer also receives the derived clock signal (BI3 Control Clock H). Every 16 clock cycles, the deserializer starts shifting in a new state frame. While the state byte is shifting in, the deserializer counts the number of 0s in the received preamble byte. It also checks the parity of the incoming state bits. When the entire state byte is present, the deserializer looks at bit 7 (sync) and bit 0 (parity). The leading (rising) edge of the next clock pulse loads bits 6—1 of the state byte into the deserializer's parallel output buffer, if the following conditions are true.

1. At least seven 0s are detected in preamble byte of state frame.
2. Bit 7 of state byte is 1.
3. Parity of bits 6—1 in state byte is correct (odd parity).
4. M8970 is not preparing for possible STI port switch (PM3 INH Status S/R L negated).
5. All decoded state-frame clock cycles are less than 240 ns wide (BI4 INH Status Load H negated).

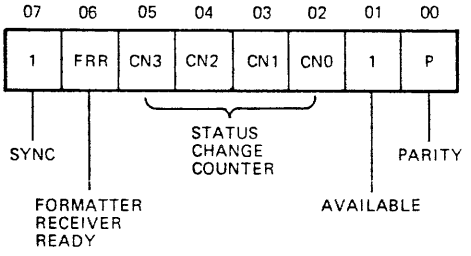
Figure 2-13 Controller/Formatter State Bit Assignments



CONTROLLER STATE BYTE



FORMATTER STATE BYTE (ON–LINE)



FORMATTER STATE BYTE (AVAILABLE)

SHR-0156-85

Assuming proper synchronization (that is, conditions 1 and 2 are true), the deserializer produces a negative BI4 RCV DLY L pulse one clock cycle after the assumed loading of the new controller state bits. The negative pulse is one clock cycle wide.

If condition 3 is not true, the deserializer asserts a parity error signal (BI4 CNTL PARR ERR H) and does not update the output buffer. The buffer is updated with succeeding error-free state bits, but the parity error signal remains asserted until the deserializer receives a clear error signal (BI4 CLR STA PE L asserted).

The contents of the output buffer appear at six totem-pole gate array outputs immediately and appear at six tri-state gate array outputs when enabled. Only two of the totem-pole outputs are used: BI4 KRR H which reflects controller receiver ready (bit 6), and BI4 KINI H which reflects initialize (bit 3). Signals BI4 KRR H and BI4 KINI H are routed to the M8970 and M8972 respectively. The tri-state outputs are for microprocessor access.

To access the controller state bits, the M8972 reads location 82. This causes the M8970 to provide a negative PM6 SEL CSTI L pulse, enabling the gate array's tri-state outputs. The controller state bits in the output buffer then appear on microbus data lines 8MC1 D <6:1> H. In effect, the deserializer's output buffer is the controller state (CSTI) buffer in the microprocessor I/O address space.

The serializer receives the intended formatter receiver ready bit from the M8970 through the PM6 FRR H line. It receives the remaining five intended state bits and two control bits from the formatter state (FSTO) register. The serializer clock is PM3 XCLK L from the M8970, and the serializer output appears on the BI4 Control Out H line.

Every 16 clock cycles, the serializer starts a new state frame. Signal BI4 CLK BFA L goes high, loading the six intended state bits and two FSTO control bits into the serializer's parallel input buffer. The serializer starts shifting out an all 0s preamble byte. While the preamble byte is shifting out, the contents of the input buffer load into a second parallel buffer. The serializer then calculates parity based on seven buffer bits (the six intended state bits and FSTO bit 0). FSTO bit 0 is a parity control bit which is 0 in normal operation. It then causes the calculation of correct (odd) parity to generate bit 0 in the state byte. Bit 7 in the state byte is always set to 1. The serializer shifts out the state byte after the all 0s preamble byte if the following conditions are true.

1. FSTO bit 7 is 1 indicating formatter is on-line or available.
2. PM6 FRR H or content of FSTO is not changing (PM6 VALID L asserted).
3. M8970 is not preparing for possible STI port switch (PM3 INH Status S/R L negated).

If one of these conditions is not true, the serializer shifts out a second all 0s byte in the state frame. This is the case when the formatter is off-line or broken (minimum integrity failure detected).

For either state, the M8972 clears FSTO bit 7, resulting in a stream of 0s at the serializer output. With the formatter broken, 0s appear on the enabled STI real-time formatter state line(s). With the formatter off-line, all drivers are disabled. Therefore, nothing appears on either port's STI real-time formatter state line. In addition, an HSC stops transmission of controller state when the formatter is off-line. (This is the case since an HSC uses the formatter-state clock rate to transmit controller state.)

The M8972 specifies formatter state bits 5—1 and two control bits through the FSTO register. To access FSTO, the M8972 writes into location 82, causing the M8970 to produce a negative PM6 SEL CSTO L pulse and a positive PM6 Valid L pulse. The serializer then shifts out all 0s. At the trailing (rising) edge of PM6 SEL CSTO L, the FSTO register strobes in the contents of the microbus data lines. The serializer then strobes in the contents of FSTO within the next 16 clock cycles [referenced to the trailing (falling) edge of PM6 Valid L].

Formatter receiver ready (FRR) logic on the M8970 controls the intended bit 6 in the state byte. (Refer to the paragraph Formatter Receiver Ready Control Logic under Paragraph 2.3.2 for a discussion of how this bit is specified.)

Error Logic and PERI Register

The M8971 error logic asserts a composite error signal (BI4 COMP ERR H) if any of six errors occurs. Three of these errors are M8970 detected. If any of these errors occur, the M8970 asserts the board error signal (PM4 BD ERR L). (Refer to the paragraph Data Clock Control Logic under Paragraph 2.3.2 for definitions of the M8970-detected errors.) Deactivating the M8970 ensures negation of the three M8970-detected error signals and the board error signal.

The remaining three errors are STI bus transmission errors detected by the M8971. These three errors are: the data line pulse error (BI4 L Data Pulse ERR H), control line pulse error (BI4 CNTL Pulse ERR H), and control line parity error (BI4 L CNTL PAR ERR H). The pulse errors are detected by the decoders and recorded by dual J-K flip-flop E8. If a pulse error occurs, the appropriate part of E8 strobes in a 1. The control line parity error is detected by the deserializer and recorded by an internal flip-flop.

The M8972 monitors the composite error signal through bit 7 in the PSTI buffer on the M8970. Reading a 1 for this bit indicates an error has occurred. If this is the case, the M8972 then reads the PERI register to identify the error. Note that the PERI register has tri-state outputs.

To access the PERI register, the M8972 reads location 81 causing the M8970 to produce a negative PM6 SEL PERI L pulse. At the leading (falling) edge of PM6 SEL PERI L, the PERI register strobes in the error status, and then (with PM6 SEL PERI L low) the PERI register places the error status on the microbus data lines. In addition, after PM6 SEL PERI L goes low, flip-flop E10 strobes in a 1 at the leading (rising) edge of the next microprocessor clock pulse. Flip-flop E10 then clears the data pulse error flop, control pulse error flop, and control parity error flop in the deserializer (BI4 CLR STA PE L asserted). Therefore, reading the PERI register also clears the three M8971-detected errors.

Shift register E2 measures the controller state clock period, which is derived from the incoming bit rate. Register E2 causes the setting of flip-flop E28 if a bit in a controller state frame does not arrive within two M8970 clock cycles (about 240 ns) after the last bit. Flip-flop E28 then asserts BI4 INH Status Load H to inhibit updating of the parallel output buffer in the deserializer. Setting E28 also clears the control pulse error flop and the control parity error flop. The trailing (rising) edge of BI4 RCV DLY L, from the deserializer, then clocks a 0 into E28. In summary, a controller state frame is considered invalid if a state frame clock cycle is too long.

To prepare for a possible port switch, the M8970 asserts PM3 INH Status S/R L for 18 clock cycles. This clears the data pulse error flop. After a port switch, an old data line pulse error is irrelevant since it may apply to a disabled receiver.

2.3.4 M8973 Extended Memory

The M8973 module contains 56K of ROM that stores 38 off-line microdiagnostics tests (that is, all off-line tests except PASS and FAIL). The M8973 also contains jumpers to specify the tape drive hardware revision level.

The memory consists of seven 8K × 8-bit chips with tri-state outputs. The M8973 also contains a socket reserved for future use by an eighth chip. The M8973 accesses the contents of the extended memory through I/O locations using three data transfers. Two transfers set up the extended memory address and the third transfer reads the memory data.

To detect I/O data transfers, decoder E1 processes: an enabling signal (8MC2 SEL D BUS L), three address bits, and two read/write control signals from the microbus. For an I/O data transfer involving the M8973, E1 produces a negative pulse for 1 of 4 signals. Three of these signals access a register or buffer, and the fourth is a control signal (Table 2-9).

Table 2-9 M8973 I/O Decoding

Microbus Address	Data Transfer Type	Decoded Signal	Selected Register or Buffer
C8	Write	MX1 WRO L	Extended Memory Address Low (EXADRL)
	Read	MX1 RD0 L*	—
C9	Write	MX1 WR1 L	Extended Memory Address High (EXADRH)
	Read	MX1 RD1 L	Hardware Revision Level (EXHRL)

* Control signal

To access the memory contents, the M8972 first writes the low byte of the extended memory address into the EXADRH. The three most significant bits in EXADRH specify the desired memory location to an 8K bank. With the address in the 0—56K (0—DFFF Hex) range, these three EXADRH bits cause decoder E4 to enable one of the memory chips. The five remaining bits in EXADRH and the eight bits in EXADRH then specify the desired location within the enabled chip.

The M8972 reads location C8 causing E1 to produce a negative MXI RD0 L pulse. This activates the tri-state output in the enabled memory chip. The enabled chip places the contents of the desired external memory location onto the microbus data lines (8MC1 D<7:0> H).

While executing the SHOW VERSION ASCII port command, the M8972 reads EXHRL buffer to determine the settings of the eight jumpers on the M8973. The jumper settings specify the tape drive hardware revision level between 0 and 256. Each jumper setting is associated with a binary weighted value. Reading a 0 for a jumper bit indicates the corresponding buffer input is jumpered to ground.

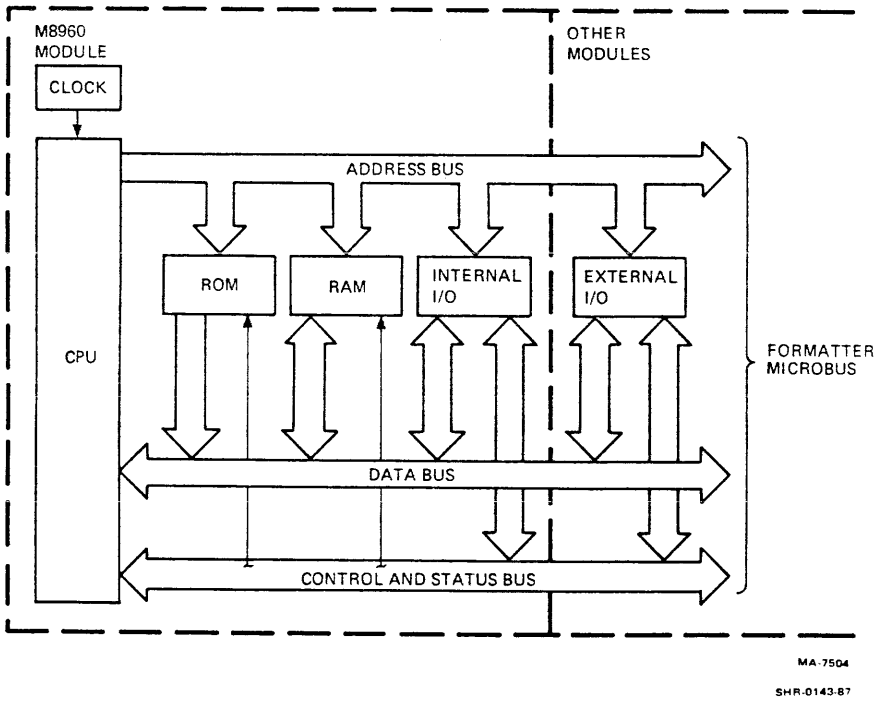
2.4 MASSBUS INTERFACE

2.4.1 M8960 Microcomputer (8MC)

The M8960 module is a completely self-contained microcomputer dedicated to supporting the formatting and data handling functions of the TM78. It contains a microprocessor CPU, 16K bytes of operational microcode stored in read-only memory (ROM), 4K (8K potentially) bytes of read/write scratchpad memory (RAM), some internal I/O devices, and I/O decoding logic. The 62-line TM78 microbus handles communication between the microcomputer and all other external I/O peripheral devices in the TM78 system.

Figure 2-14 shows the basic functional components of the microcomputer (CPU, ROM, RAM, internal I/O) and the three major components of the microbus (address, data, control/status).

Figure 2-14 M8960 Microcomputer Functional Areas



Microcomputer Architecture — The following paragraphs introduce the microcomputer architecture by describing the block diagram in Figure 2-15. It shows all major logic elements and interconnections in the microcomputer module. Included are the CPU, clock and timing, address/data paths and associated buffers/latches, storage elements, I/O decoders and internal I/O devices/registers.

NOTE
The functional block diagram in Figure 2-15 uses logical AND and OR symbols. It does not necessarily follow that a corresponding gate exists on the TM78 logic prints. The assertion of inputs A and B causing the assertion of output C may be represented on a block diagram by a single AND gate; yet, the engineering drawing may show that several circuit stages are involved in the ANDing operation.

NOTE
The signal names on the functional block diagram are the names on the engineering circuit schematics (CS prints). Where other signal names or notes are used, they are enclosed in parentheses.

The Central Processor Unit — The CPU is a single-chip 8085A microprocessor employing 8-bit parallel construction. The CPU executes instructions from the ROM/RAM storage elements at a 1.3-microsecond cycle time and can directly address up to 64K of memory or I/O area. The 8085A has an integral system controller interfaced to the microbus through the control buffer. The 8085A also has five vectored interrupt inputs and a serial data input port (for data from the maintenance panel keypad). Table 2-10 describes each functional I/O pin on the 8085A, and how it applies to the microcomputer and the TM78 formatter system.

Figure 2-15 M8960 Block Diagram

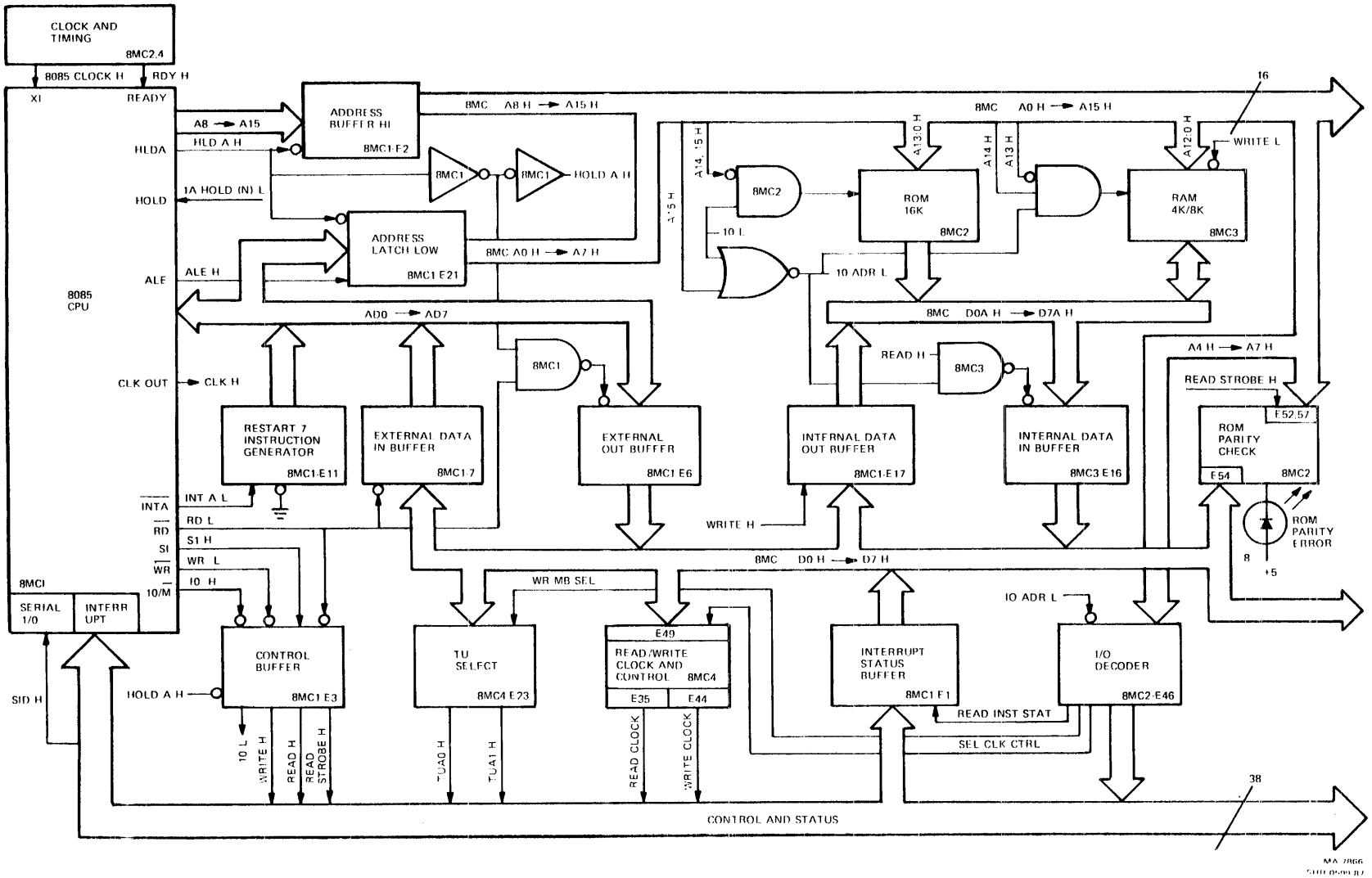


Table 2-10 Microcomputer Pins

Pin	Description
A 8-15	These are the most significant eight bits of the memory or I/O address, tri-stated during hold and reset modes.
AD 0-7	This is the multiplexed address/data bus. The least significant eight bits of the memory I/O address appear during the T1 cycle. It then becomes the I/O data bus during the T2 and T3 cycles.
ALE (ALE H)	The address latch enable signal occurs during the T1 cycle to latch the address on the AD lines to the microbus.
S1 (S1 H)	This machine cycle status bit indicates that a memory or I/O read cycle is in progress. This line becomes the microbus READ H control line that is tri-stated during hold mode.
IO/M (IO H)	The I/O or memory signal is HI during the execution of two special I/O microprocessor instructions and disables ROM and RAM memory. When LO, memory is enabled. It is tri-stated during hold mode.
RD (RD L)	The read control signal when LO indicates that the addressed memory or I/O location will be read from. It is tri-stated during hold and reset modes. This line becomes the microbus READ STROBE H control line that is tri-stated during hold mode.
WR (WR L)	The write control signal when LO indicates that the addressed memory or I/O location will be written to. It is tri-stated during hold and reset modes. This line becomes the microbus WRITE H control line that is tri-stated during hold mode.
READY (READY H)	The ready input signal is normally HI, indicating that the addressed memory or I/O register is ready to send or receive data. Ready is driven LO during a read operation from the TU port, thus extending the operation one microprocessor cycle.
HOLD (IA HOLD (N) L)	The hold input signal comes from either CAS module to indicate the host computer wants to take control of the microbus. The microprocessor releases the bus at the completion of the current transfer.
HLDA (HLD A H)	The hold acknowledged output signal indicates the microprocessor is honoring a hold request and has released control of the microbus. This signal causes the microcomputer to tri-state the address, data, and control portions of the microbus.

Table 2-10 Microcomputer Pins (Cont)

Pin	Description
INTR	The interrupt signal is raised by a TU status parity error from the currently addressed TU port. It causes the microprocessor to jump to PC address 70_8.
INTA (INTA L)	The interrupt acknowledge output signal indicates the microprocessor is honoring a request from the INTR line. It causes the restart instruction generator to place a PC vector (70_8) on the AD lines.
RST 5.5	Restart 5.5 is an interrupt raised by either a CAS parity error or a Massbus access to an illegal register. It causes the microprocessor to jump to PC address 54_8.
RST 6.5	Restart 6.5 is an interrupt raised by a Massbus versus microbus contention error in trying to access CAS at the same time. It causes the microprocessor to jump to PC address 64_8.
RST 7.5	Restart 7.5 is an interrupt raised by a TU command parity error from the currently addressed TU port. It causes the microprocessor to jump to PC address 74_8.
TRAP	Trap is an interrupt raised by a microcomputer ROM parity error or a power supply AC LO condition. It causes the microprocessor to jump to PC address 44_8.
RESET IN	Reset is a general-purpose microprocessor clear input that jams its program counter to zero and resets a hold condition. Reset is enabled by a MASSBUS INIT, TM Clear command, power supply DC LO condition, or the MASTER RESET button being depressed. If reset is caused by anything but a MASSBUS INIT, and a hold condition was in effect earlier, then the hold is automatically evoked again.
X1 (8085 CLOCK H)	X1 is the crystal clock timing input.
CLK OUT (CLK H)	Clock output is the master system clock that reflects the internal microprocessor time-state periods. The CLK H period is twice that of the X1 input.
SID (SID H)	The serial input data signal reflects information about whether a key has been pressed in the maintenance keypad matrix. The line is monitored by a special microprocessor instruction.

Clock and Timing — Timing for the microcomputer, microbus read/write cycles, and TU bus command/status transfers develops on the microcomputer module. Figure 2-15 shows the logic and the associated waveforms for this system timing.

Clock pulses are produced by a crystal oscillator running at 18.084 MHz. Pulses are output as NORMAL CLOCK H, which has a period of 55.3 ns. NORMAL CLOCK H goes through a divide-by-three counter and is output as 8085 CLOCK H, which has a period of 166 ns. 8085 CLOCK H becomes the clock input (X1) to the microprocessor chip. This chip is internally divided by two and inverted for a master system clock (CLK H) of 332 ns. CLK H is a microprocessor output that reflects the period for its internal time states.

The CLK H period is used within the microprocessor for instruction and interrupt timing; it normally continues uninterrupted as long as the ready input stays HI. However, there is one case where a modified cycle time may be advantageous: when a request for tape unit status is made to a TU port. In this case extra time must be allowed for the request to reach the distant tape unit. The tape unit then assembles its status and returns it to the microprocessor. This extra time is obtained by introducing an artificial time state (or wait state) into the read cycle from the TU port.

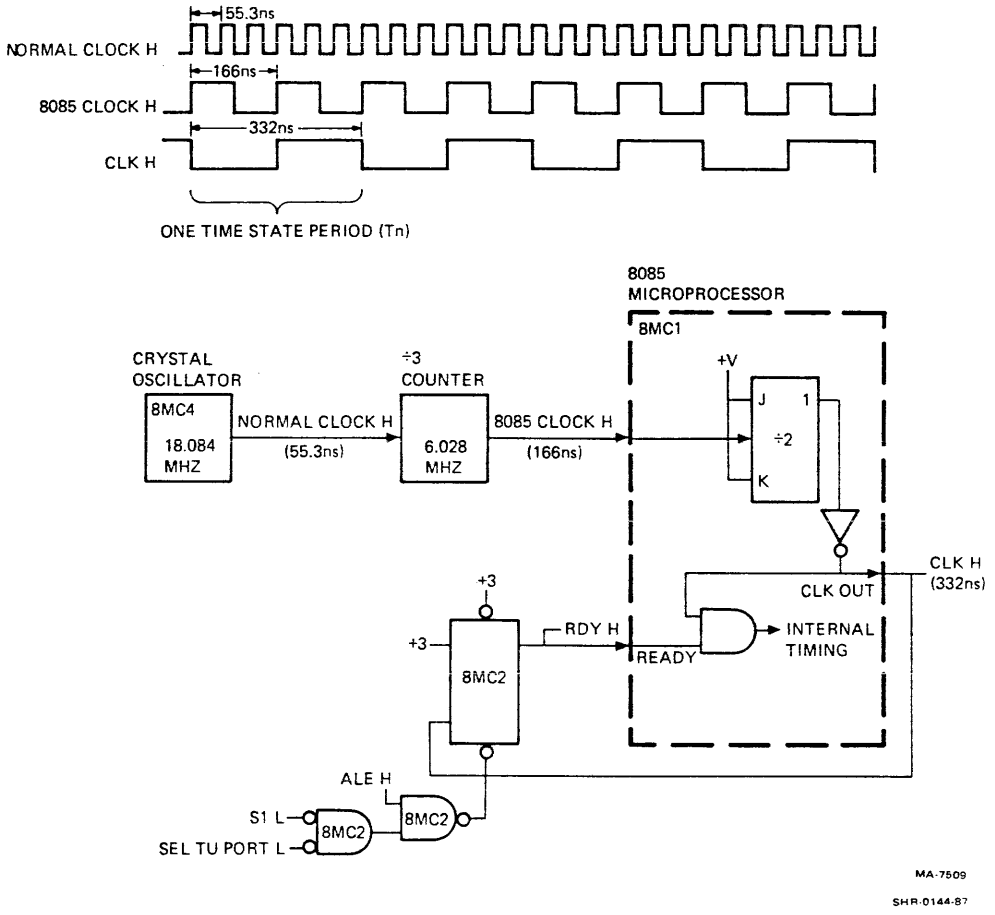
Figure 2-16 (top) shows the ready flip-flop coupled to the ready input of the microprocessor. When a TU port address is decoded (SEL TU PORT L) during a read cycle (S1 L) at address latch enable time (ALE H), the ready flip-flop is directly reset; this removes the READY enable input from the microprocessor. The microprocessor continues to produce CLK H time state pulses, but does not perform any internal processing functions. Ready is clocked set on the rising edge of CLK H following the negation of ALE. Then the microprocessor finishes the read cycle.

Address/Data Paths and Associated Buffers/Latches — Address/Data paths carry microcomputer instructions, command/status information to and from other internal peripherals, and the associated memory/peripheral addresses.

Address

The HI byte of an instruction, operand, or I/O register address is asserted by the microprocessor for the duration of an instruction fetch, input, or output cycle on the A8—A15 lines. The HI byte is buffered and becomes A8 H to A15 H as part of the microbus. The LO byte of the address is asserted on the multiplexed address/data lines AD0—AD7 at the beginning of the machine cycle (time state T1). The LO byte is latched on the falling edge of ALE H and becomes A0 H to A7 H as part of the microbus. A0 H to A15 H then address internal storage (ROM/RAM), internal I/O, and external I/O.

Figure 2-16 Microcomputing Timing



Data

During the middle and end of the machine cycle (time states T2 and T3), the multiplexed AD lines (AD0—AD7) form the microprocessor's I/O data bus. During a write cycle, data is gated from the AD lines to the microbus D lines by the external data out buffer. This buffer is disabled by the control signal RD L and is normally enabled unless a read cycle is in progress. During a read cycle, data is gated to the AD lines from the microbus D lines by the external data in buffer. This buffer is enabled by RD L.

If the microcomputer's internal storage (ROM/RAM) is the source or destination of data, then that data is carried over an intermediate data bus (D 0:7 A H). During a write cycle to RAM, data is sent to the D (N) A lines from the microbus D lines. Data is gated through the internal data-out buffer by the WRITE H control signal. During a read cycle from ROM or RAM, data is sent to the microbus D lines from the D (N) A lines. The data is gated through the internal data-in buffer by the control signal READ H. So, for a read from or write to operation involving microcomputer internal storage, data must go through three independent buses. They are the multiplexed AD lines, the microbus D lines, and the intermediate D (N) A lines.

Storage Elements — The microcomputer internal storage elements include 16K bytes of read-only memory (ROM) and 8K bytes of random access or read/write memory (RAM). Except for a small array of internal registers, the microprocessor chip does not contain any storage.

The ROM is enabled by any address in the range of 0 to 1FFFF (0 to $37,777_8$) on the A lines. The active address decoding lines are A0—A13 H, with lines A14, 15 H, or an I/O instruction in progress (IO L) acting as inhibits. ROM is physically divided into nine 2K byte by 8-bit elements; eight of these contain the operating system microcode for the microcomputer. The ninth ROM contains an odd parity bit for each of the 16K locations. For a given location within that 16K range not used by the operating system, the corresponding parity bit will be even. A read cycle from one of these locations causes a parity error (and subsequent trap interrupt) to occur. This indicates that the microcomputer branched to an illegal location. The trap interrupt brings the program counter back to a known value.

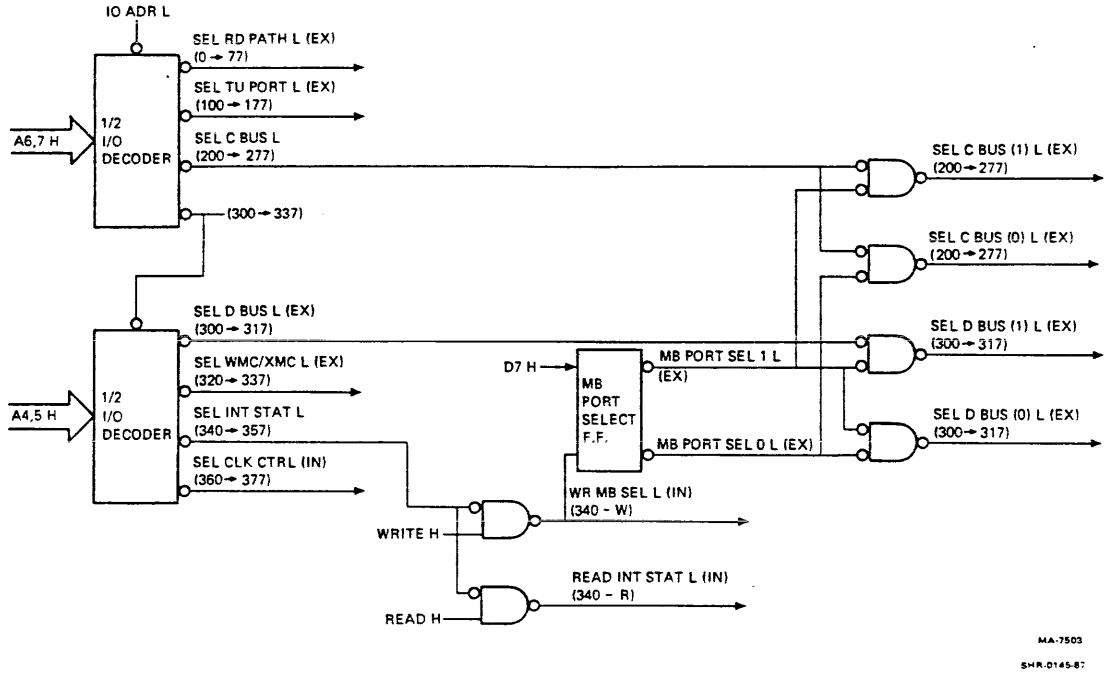
The RAM is enabled by any address in the range of 4000 to 5FFF ($40,000_8$ to $57,777_8$) on the A lines. The active address decoding lines are A0—12 H and A14 H, with lines A13, 15 H, or an I/O instruction in progress (IO L) acting as inhibits. RAM is further divided into a fixed RAM area (lower 4K bytes) and a floating RAM area (upper or lower 4K bytes). Address bit A12 H selects either the fixed RAM (A12 H unasserted) or the floating RAM (A12 H asserted). Fixed RAM is physically divided into eight 1K byte by 4-bit elements. When a fixed RAM address is specified, two of the eight RAMs are selected at the same time for an 8-bit parallel I/O. Parity for RAM is not generated or checked.

> **NOTE**
> **Floating RAM was designed into the microcomputer module's logic for future capacity and is currently unused. The circuit module is manufactured without these eight additional RAM elements.**

The Write H microbus line determines whether data will be written to the RAM elements or read from them. When Write H is asserted, data is strobed from the D (N) A lines to the addressed RAM location. When Write H is negated, data in the addressed RAM location is placed on the "D (N) A" lines.

I/O Decoding — The I/O decoder is enabled by the signal I/O ADR L. This signal is produced whenever an I/O page address is placed on the microbus A lines (A15 H asserted), or an I/O instruction is active in the microprocessor (IO L). Decoding is accomplished by address bits A4—A7 H. Figure 2-17 shows the I/O decoder logic. Where the decoder logic produces a signal sent over the microbus to select an external device, an (EX) appears after the signal name. Where the decoder produces a signal to enable an internal device or register, an (IN) appears after the signal name. Also appearing beneath the signal name will be the address or range of addresses in the I/O page which produce that signal.

Figure 2-17 I/O Decode Logic



Internal I/O Devices/Registers — There are three I/O devices internal to the microcomputer. They are the interrupt status buffer, the read/write path clock and control circuitry, and the tape unit select register.

Interrupt Status Buffer

By reading microbus address E0-R, the microprocessor receives one byte of interrupt status information. Reading address 340 causes the I/O decoder to produce the signal READ INT STAT L, which strobes certain microbus (and internal) status lines to the D lines. Refer to Appendix E, address 340_8-R, for a description of the interrupt status word.

Read/Write Path Clock and Control Circuitry

The microcontroller module provides clock pulses to the read/write path for microcontroller sequencing, and to the write path only for write data timing. One of six frequencies are programmably selectable by writing to microbus address F1-W, the clock control register. This register is selected by I/O decoder signal SEL CLK CTRL L.

Table 2-11 lists the frequency sources, which are selected individually for read and write paths.

Tape Unit/Massbus Select Register

The tape unit select register is a 2-bit field that selects one of four logical TU ports. A write to microbus address E0-W produces the signal WR MB SEL L that clocks data lines D0 and D1 H to the TU select register. The register outputs are microbus lines TU A0 H and TU A1 H, which go to both TU port modules.

WR MB SEL L also clocks the Massbus port select flip-flop set or reset, depending upon the state of data bit D7 H. When D7 = 0, Massbus port 0 is selected (MB PORT SEL 0 L); when D7 = 1, Massbus port 1 is selected (MB PORT SEL 1 L).

Table 2-11 Read/Write Path Clock Sources

Clock Frequency	Write Path	Read Path
18.084 MHz	Normal write path clock	—10% clock used for retries
21.84 MHz	Not used	+10% clock used for retries
20.0 MHz	+10% clock used for the byte assembly logic during a GCR read	Normal read path clock
15.6 MHz	—20% clock used for the byte assembly logic during a PE read	—30% clock
External Clock	Not used	Not used
Single Step	A single clock pulse provided for the read and/or write path each time the clock control register is written into; used for diagnostic purposes	

Microbus Timing — Figures 2-18, 2-19, and 2-20 are timing diagrams for microbus write/read cycles initiated by the microcomputer. Shown are the write (Figure 2-18), read (Figure 2-19), and read from TU port (Figure 2-20) cycles. Each cycle consists of three or four time states of 332 ns each, based upon the microcomputer's basic machine state. Time state T1 is always the addressing state; and time states T2, T3, and TWAIT (if applicable) are for I/O data transfers.

NOTE
These microbus timing diagrams represent cycles initiated by the microcomputer only. Cycles initiated by the host computer through the common address space module have events that occur in the same basic sequence, but across a longer time scale.

Figure 2-18 Typical Microbus Write Cycle

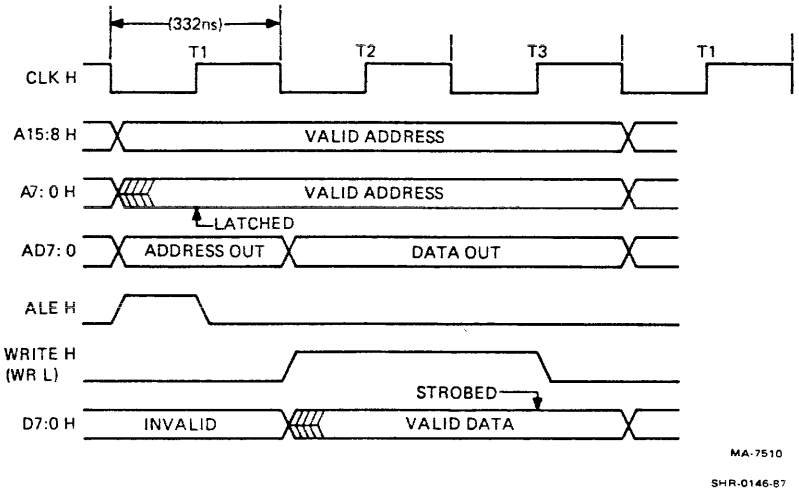
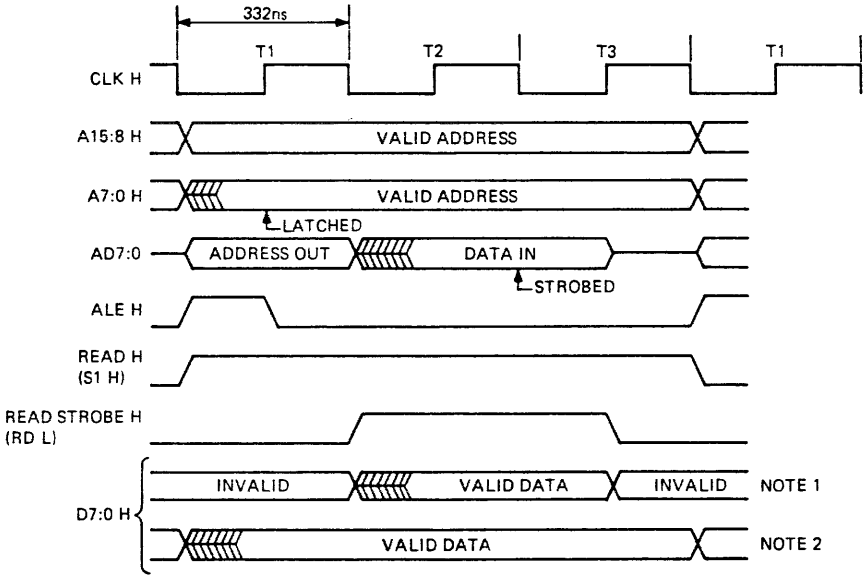


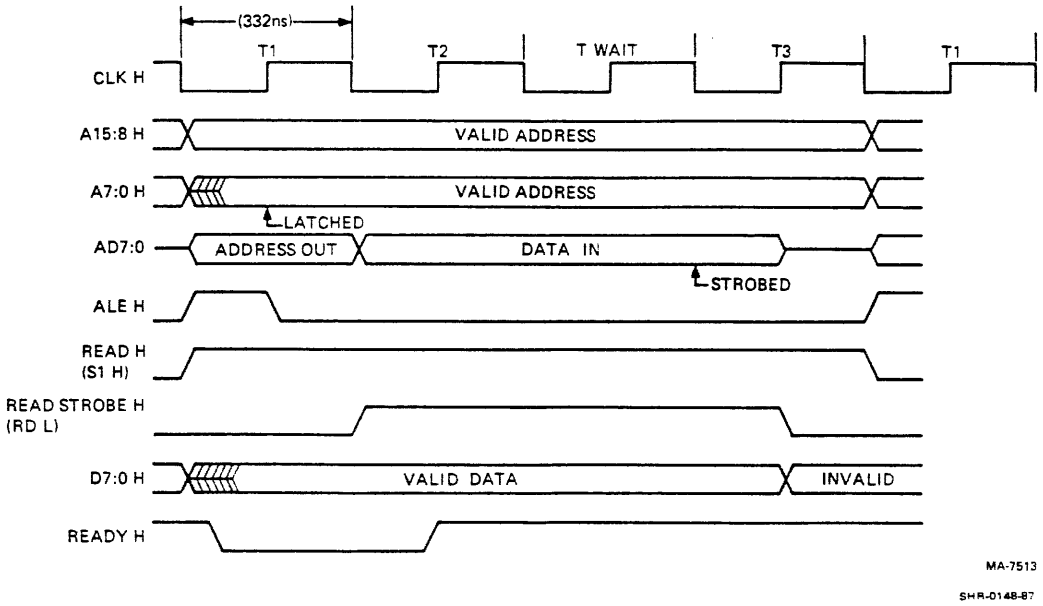
Figure 2-19 Typical Microbus Read Cycle



NOTE 1:
DATA ON D LINES DURING "READ STROBE H" TIME FOR THE FOLLOWING SOURCES:
M8953 READ PATH
M8957 CAS

NOTE 2:
DATA ON D LINES DURING "READ H" TIME FOR THE FOLLOWING SOURCES:
M8956 MB DATA
M8959 WRITE MICRO
M8960 MICROCOMPUTER (ROM/RAM)

Figure 2-20 Microbus Read Cycle from TU Port



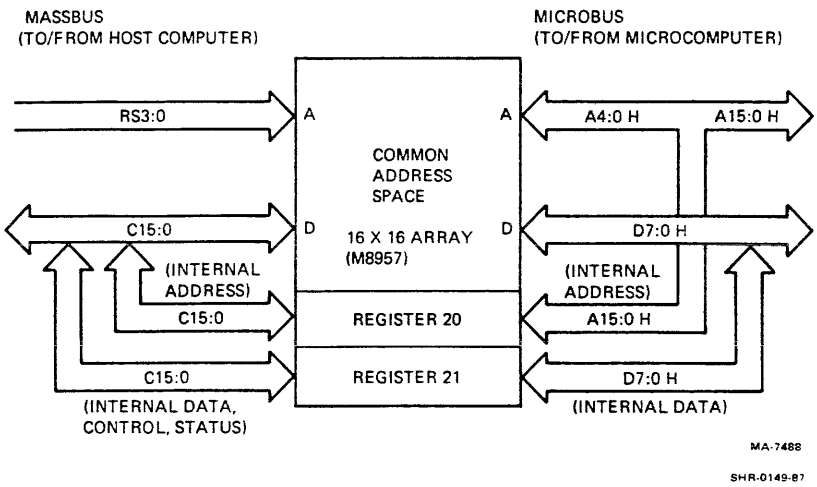
2.4.2 M8957 Common Address Space Module

The M8957 common address space (CAS) module is the element of the Massbus port that enables Massbus control bus (CBus) communication between the host computer and the TM78 formatter. It contains circuitry that stores command/status information and handles Massbus CBus protocol independently, without the aid of the microcomputer (M8960).

The major functional component of the CAS module is a 16-location by 16-bit array of random access memory (RAM) locations known as the common address space. This array is dual ported: that is, it is common to both the host computer by means of the Massbus C lines and the microcomputer by means of the internal microbus. Command information is written to CAS RAM by the host computer, then read from CAS RAM by the microcomputer during its idle loop polling routine. Similarly, the microcomputer writes status information to CAS RAM, which is then read by the host computer during its interrupt servicing routine. Thus, all command or status transfers take place by writing into and reading from the CAS RAM matrix in one direction or the other.

Figure 2-21 shows the CAS dual-ported RAM. On the Massbus side of the array, note that the RAM is addressed by register select signals <RS3:0> and data is transferred by means of the C lines <C15:0>. On the microbus side, the RAM is addressed by address lines <A4:0> and data is transferred by means of the data lines <D7:0>. The Massbus may transfer data to or from CAS RAM one word (16 bits) at a time, but the microbus may transfer only one byte (8 bits) at a time. Thus, the microcomputer must perform two individual write or read cycles to or from CAS to complete a word transfer. The microcomputer may access any single byte of CAS RAM area (1 of 32) by addressing any I/O page value in the range of 80—9F HEX.

Figure 2-21 CAS Dual-Ported RAM

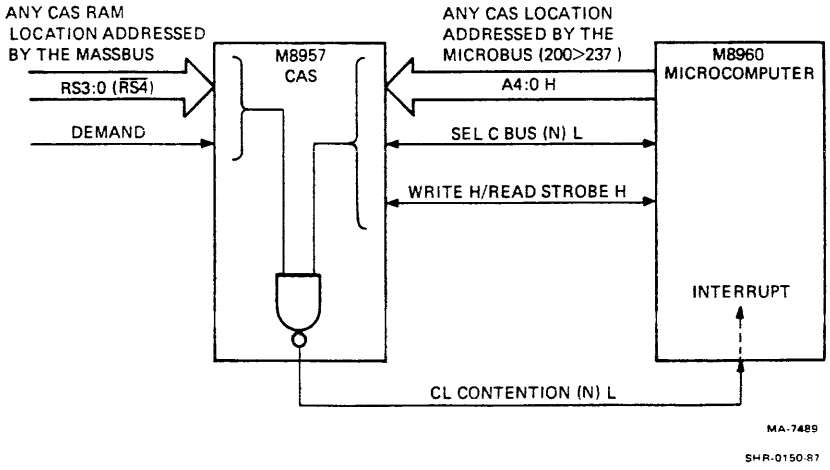


These registers are used for initializing, loading, and verifying microcode diagnostics from the host computer system. The host may also read or write practically any internal microbus address to include microcomputer ROM/RAM area and internal peripheral device locations. This gives the host computer system a powerful diagnosing capability.

Because CAS is dual ported, the host computer and the microcomputer can initiate a read/write request to a CAS address at the same time. This cannot be allowed due to the nature of the individual RAM elements and the control circuitry. In all cases, the host computer requesting the CAS access is given priority over the microcomputer's request.

If both ports attempt to access CAS at the same time, a contention error results. Figure 2-22 shows the signals from each port, which together produce the microbus signal CL CONTENTION (N) L. Contention returns to the microcomputer and interrupts the microprocessor, which then retries its CAS request.

Figure 2-22 CAS Addressing Contention Error



When a contention error occurs, the microcomputer may have just started its request cycle or may be at any point in the cycle. The microcomputer cycle in progress is not aborted midway, so that the Massbus may achieve its priority, but is allowed to continue to completion. This may result in an incorrect data transfer to or from the CAS location addressed by the microcomputer, which is the reason for the subsequent interrupt/retry operation.

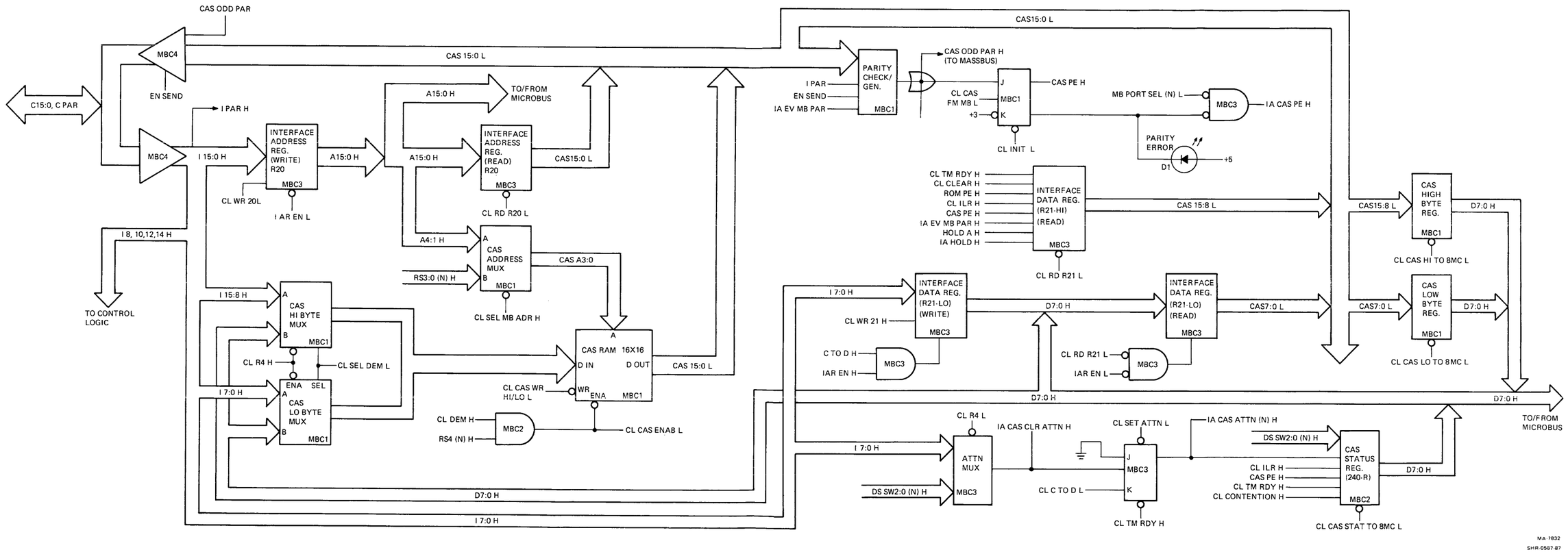
CAS Data Paths and Control Logic — The CAS module transfers most command/status information to and from the host computer and microcomputer over two internal buses, the I bus and the CAS bus. The I bus (input bus) carries information from the host computer to all device registers to include CAS RAM (registers 0 through 17) and registers 20 and 21. The CAS bus (output bus) carries information from all device registers to the host computer, and information from CAS RAM to the microcomputer. Information from the microcomputer to CAS RAM does not appear on the I bus, but rather is transmitted directly to the RAM input multiplexers by the microbus D lines.

Figure 2-23 shows the CAS data paths, and how the Massbus and microbus are interfaced, or buffered, to them. Notice that the I bus (I 15:0 H) is driven from a single source, the Massbus C line receivers. The CAS bus (CAS 15:0 L) is driven by three different sources: the interface address register (tri-state outputs), the interface data register (tri-state outputs) and the CAS RAM (open collector outputs). The following discussion of how information transfers to and from CAS covers two areas: Massbus transfers (for host computer reads and writes) and microbus transfers (for internal microcomputer reads and writes).

> **NOTE**
> **The functional block diagrams in Figures 2-23 through 2-29 use logical AND and OR symbols. It does not necessarily follow that a corresponding gate exists on the CAS module (M8957) logic prints. The assertion of inputs A and B causing the assertion of output C may be represented on a block diagram by a single AND gate; yet the engineering drawing may show that several circuit stages are involved in the ANDing operation.**

> **NOTE**
> **The signal names on the functional block diagram are the names used on the engineering circuit schematics (CS prints). Where other signal names or notes are used, they are enclosed in parentheses.**

Figure 2-23 CAS Data Paths



Massbus Write To/Read From Common Address Space — Figure 2-24 shows the control circuitry used in CAS accesses from the Massbus. Refer to this figure and Figure 2-23 for this discussion. The alphanumeric characters (prefixed MBC) in all functional logical elements found in these figures refer to the engineering drawing page number for the CAS module where the actual logic element may be found.

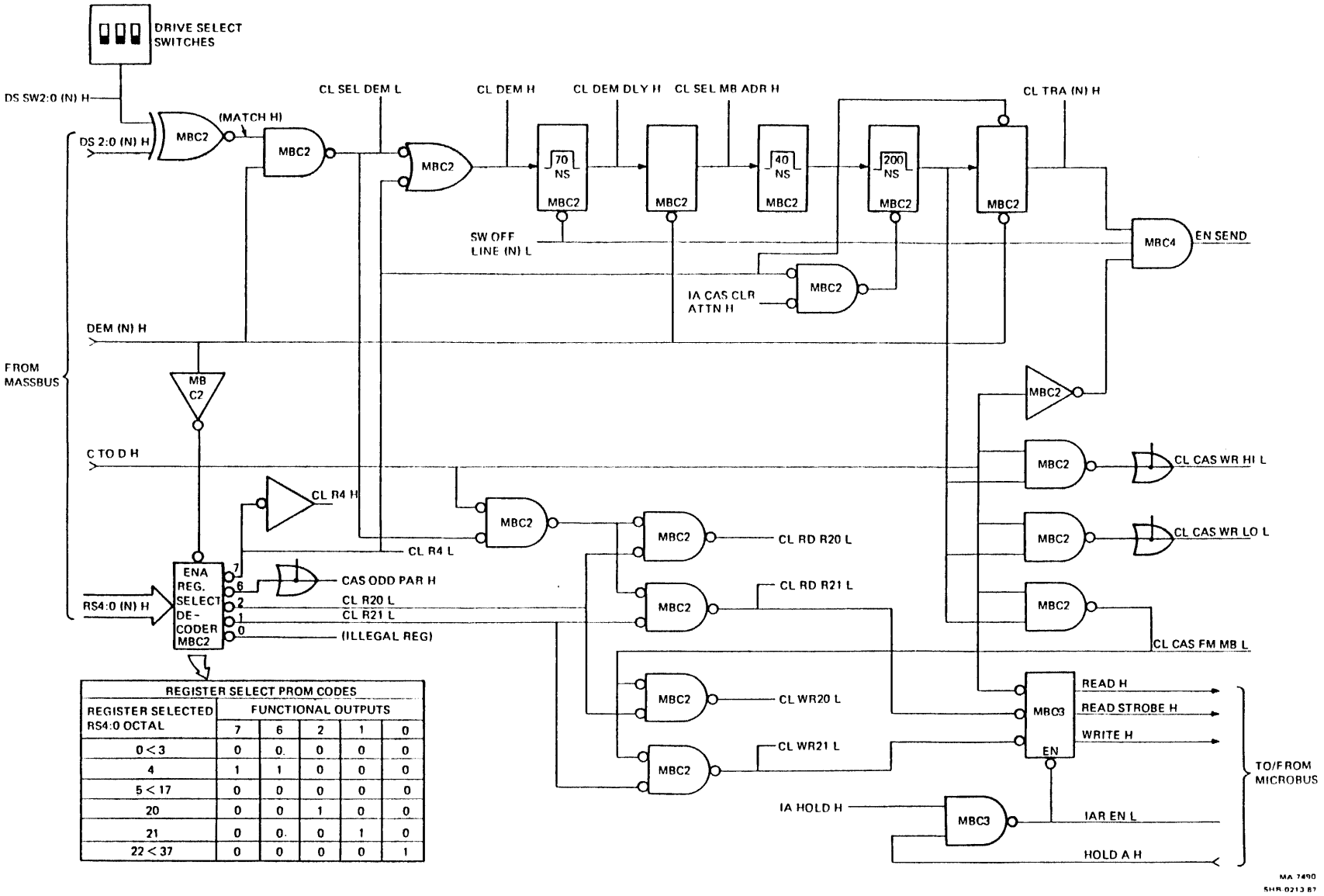
- Write

  Figure 2-25 shows the timing relationships between Massbus control signals and internal CAS control signals during a write cycle. To write into a CAS location (drive register), the controller initiates the action by selecting the TM78 (DS2:0(N)H). The controller selects a register (1 of 1810) in the TM78 (RS3:0(N)H), asserting C TO D (N)H for a controller-to-drive transfer, and places the command on the C lines (C15:0, CPAR). After a 220 ns deskew delay (275 ns for the RH20) to allow the control lines to stabilize, the controller asserts demand (DEM (N)H). The TM78 transfers the command from the C lines into the selected CAS register and then asserts transfer (TRA (N)H) to the controller. After a short delay, the controller negates demand. This causes the TM78 to negate transfer, thus completing the operation.

  When the drive select signals (DS2:0(N)H) enter the CAS module, they are compared with the code set in the drive select switches (DS SW 2:0(N)H) through an exclusive OR gate. If the two 3-bit binary codes are equal, the exclusive OR produces a HI to prime the select demand AND gate. When the controller sends demand (DEM (N)H), this AND gate is enabled and produces the signal CL SEL DEML. This signal goes to the select input of the HI/LO byte data multiplexers, which switch to their A inputs and select input data from the I bus. CL SEL DEM L also enables the demand OR gate, which produces the signal CL DEM H. This signal goes to the CAS disable AND gate; if register select bit 4 (RS4 (N)H) was false, the AND gate is disabled. This produces the signal CL CAS ENAB L, thus enabling the CAS RAM elements. If register select bit 4 was true, indicating an access to registers 20 and 21, the CAS RAM would be disabled and could not be written into or read from.

  Signal CL DEM H also goes to the 70 ns one-shot, which is enabled to start its time-out. At the end of its time-out, the 70 ns one-shot clocks the address select flip-flop set, producing the signal CL SEL MB ADR H. This signal goes to the select input of the CAS address multiplexer, which switches to its B inputs. Thus, the register select bits (RS3:0(N)H) become the CAS RAM address bits (CAS A3:0). CL SEL MB ADR H also goes to the 40 ns one-shot, which is enabled to start its time-out. This one-shot allows the CAS RAM to settle after selecting an address by means of its internal row and column decoders.

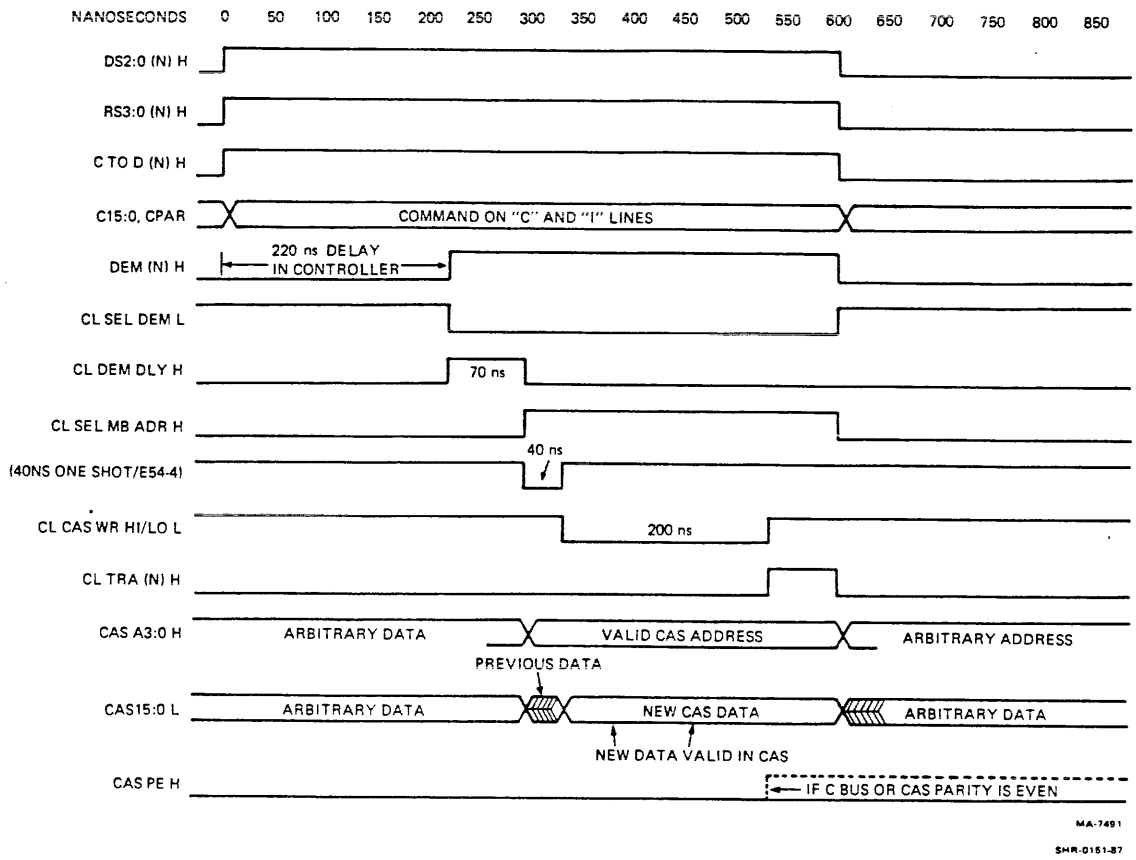
Figure 2-24 CAS Massbus Control Logic



REGISTER SELECT PROM CODES

REGISTER SELECTED RS4:0 OCTAL	FUNCTIONAL OUTPUTS				
	7	6	2	1	0
0 < 3	0	0	0	0	0
4	1	1	0	0	0
5 < 17	0	0	0	0	0
20	0	0	1	0	0
21	0	0	0	1	0
22 < 37	0	0	0	0	1

Figure 2-25 Timing for Massbus Write to CAS



When the 40 ns one-shot completes its time-out, it enables the 200 ns one-shot. On its leading edge, the output of the 200 ns one-shot enables two AND gates, which produce the signals CL CAS WR HI L and CL CAS WR LO L. These AND gates are enabled since they have been primed by the controller-to-drive (C TO D H) signal. Signals CL CAS WRITE HI/LO strobe both HI and LO bytes (16 bits) from the I bus to the selected CAS RAM location. At this time, the new data is stored in the RAM.

The data also appears at the RAM outputs and is placed on the CAS bus. Here it is sent to the parity check circuitry and compared with parity bit I PAR H Massbus control bus. The 200 ns one-shot also enables a third AND gate, which produces the signal CL CAS FM MB L. This signal is sent to the parity error flip-flop which is clocked set (even if Massbus parity was detected) on the trailing edge of the 200 ns one-shot time-out. This causes the parity error LED on the module to light and, if this Massbus port is currently selected (MB PORT SEL (N)L true), the CL CAS FM MB L signal will return IA CAS PE H over the microbus to the microcomputer.

The trailing edge of the 200 ns one-shot also clocks the transfer flip-flop set, which returns TRA to the Massbus controller. The Massbus controller "sees" TRA, performs some internal housekeeping, and then drops the device select, register select, controller-to-drive, C line, and demand Massbus signals. The negation of demand resets the transfer flip-flop and the address select flip-flop, which allows the CAS address multiplexer to select microbus addresses. This concludes the Massbus write operation.

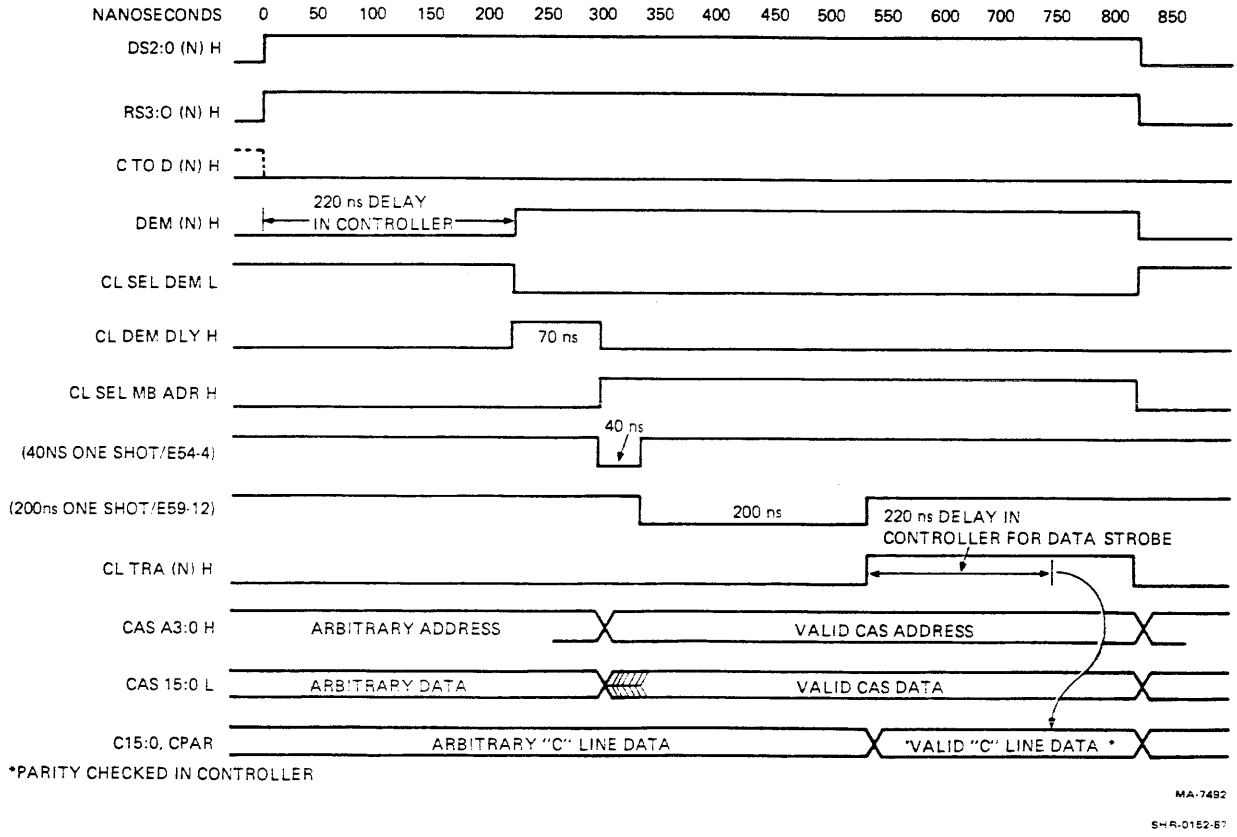
Read

Figure 2-26 shows the timing relationships between Massbus control signals and internal CAS control signals during a read cycle. To read from a CAS location (drive register), the controller initiates the action by selecting the TM78 (DS2:0(N)H) The controller selects a register (1 of 1810) in the TM78 (RS4:0(N)H), and negates C TO D H for a drive-to-controller transfer. After a 220 ns deskew delay (275 ns for the RH20), the controller asserts demand (DEM (N)H). The TM78 gates the contents of the specified register to the C lines and asserts transfer (TRA (N)H) to the controller. After another 220 ns deskew delay, the controller gates the C lines to an internal bus and negates demand. This causes the TM78 to negate transfer, which completes the operation.

The read cycle is identical to the write cycle in that the one-shot delay line starts and completes in the same manner. In the read cycle, however, the controller-to-drive signal is negated. This disables the three AND gates which ordinarily produce the CL CAS WR HI/LO L and CL CAS FM MB L signals. Thus CAS RAM is not written into, and the CAS parity error flip-flop is not strobed. CAS parity is not checked in the TM78 on a read cycle; it is assumed to be correct since no parity error was detected on a prior write cycle. Also the data may have been changed by a byte write by the microcomputer. (In any event, the parity bit is not stored in CAS RAM.) As data is read from a CAS location and placed on the CAS bus (CAS 15:0 L), the parity check/generate logic now acts as a generator and produces the odd parity bit CAS ODD PAR H. The parity bit and CAS bus go to the Massbus C line transmitters, which are enabled by the signal EN SEND; this gates the data to C 15:0, CPAR.

EN SEND is developed by an AND gate whose inputs are transfer (CL TRA (N)H), TM78 on-line (SW OFF-LINE (N)L unasserted), and C TO D H inverted. After the TM78 sends data over the C lines and 220 ns pass, the controller strobes it and performs a parity check. After another short delay, the controller negates demand and the CAS control circuitry is reset.

Figure 2-26 Timing for Massbus Read from CAS



Variations

There are two variations to these basic read/write cycles which come into play when accessing registers 4 or 20 and 21. Register 4 is the attention summary register, and registers 20 and 21 are the hardware control registers. During a Massbus write to register 4 (remember that the host must write a one bit to the attention register to clear that drive's attention bit), the data never reaches CAS RAM location 4. Before studying what occurs in the hardware to make this happen, read the following overview of the events for an attention interrupt in the TM78.

1. Something in the tape unit justifies an attention interrupt (for example, command completion, rewind completion, hardware fault, etc.).
2. The microcomputer reads the CAS status register, microbus address A0-R (240_8-R), to determine what drive number the drive select switches are set to, (drive number = N) and checks that the attention bit for that Massbus port is not set already.
3. The microcomputer writes a one to the attention register, microbus address 88-W (210_8-W) in the bit position for drive number N.
4. The microcomputer writes the interrupt code, attention address, and failure code (if applicable) to CAS register 13 (microbus address 226_8/227_8W).
5. The microcomputer sets the attention flip-flop by writing to microbus address A1-W (241_8-W). This enables the Massbus ATTN line to the controller.
6. The host is interrupted and reads the attention summary register (R4) to determine which drive(s) raised attention.
7. The host performs its interrupt servicing routine, then writes a one to bit number N of register 4, which clears the attention flip-flop. (The one bit never actually reaches CAS RAM location 4.)

Briefly, when the host computer reads and writes register 4, data is actually read from CAS RAM location 4 (step 6) but not written to it (step 7).

The register select decoder has two significant outputs when a Massbus access to register 4 takes place. The decoder is actually a 32-location PROM with 8 outputs for each location (only 5 of which are actually used). The functional PROM outputs are listed in a table below the decoder in Figure 2-24. As the table indicates, when register 4 is selected (and demand is true to select the decoder chip), outputs 7 (CL R4 L) and 6 (CAS ODD PAR H) are true (that is, they will produce a low output).

CL R4 L enables the demand OR gate, which produces the signal CL DEM H. This starts the Massbus handshake cycle. Here the cycle starts independently of the drive select code supplied by the controller. Signal CL R4 L also goes through an inverter, where it is sent to the CAS data multiplexers as CL R4 H. Here it disables the multiplexer outputs so that data cannot be written to CAS RAM address 4 from the Massbus. The data is used, however, by following the I bus down to the attention multiplexer. This multiplexer, enabled by CL R4 L, compares the I bus data bit with the code in the drive select switches. If the two are equal, the host wishes to clear the attention bit in this drive; the multiplexer produces the signal IA CAS CLR ATTN H, which clocks the attention flip-flop reset.

IA CAS CLR ATTN H provides an inhibit to one input of the 200 ns one-shot reset AND gate; the other input is primed by CL R4 L. This allows the one-shot to trigger, producing the CL CAS WR HI/LO L signals which perform a write cycle to CAS RAM location 4. Since the data inputs were inhibited, a word of zeros is written to location 4. This clears the attention bit written there earlier by the microcomputer.

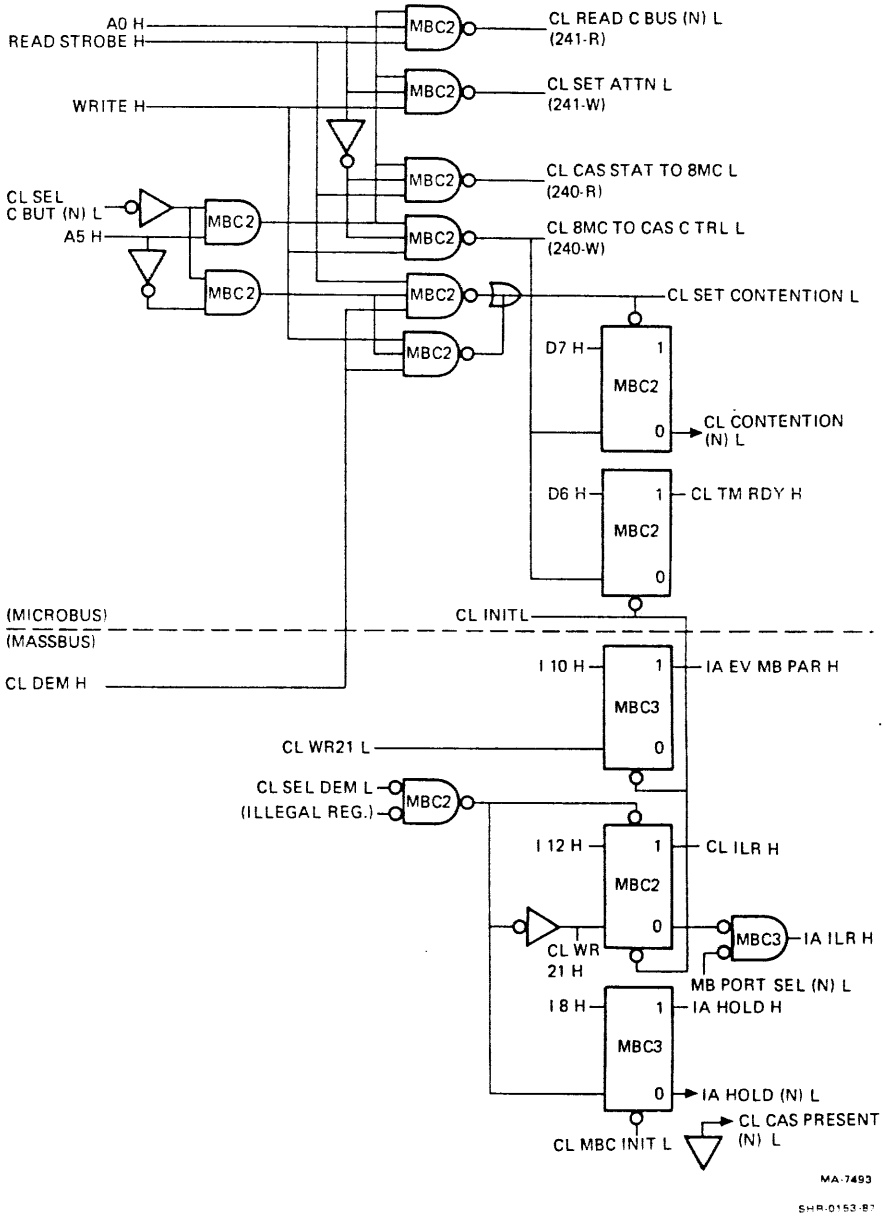
Conversely, if the host wishes to clear attention in another drive on this Massbus, the attention multiplexer does not produce IA CAS CLR ATTN H:, the attention flip-flop stays set, and the 200 ns one-shot is held reset. This prevents the CAS RAM write cycle. So the attention bit in register 4 remains there until it is recognized by the host computer. Finally, CL R4 L direct sets the transfer flip-flop, enabling a shortened read/write cycle to register 4.

The other low true signal generated by the register select decoder during an access to register 4 is CAS ODD PAR H. This output is wire ORed with the output of the CAS bus parity checker/generator. The parity line (CAS ODD PAR H) is held low during a Massbus access to register 4. During a write cycle, the host sends a one bit over the C lines to clear Attention in this drive. Since the one bit never reaches CAS RAM location 4 (a word of zeros will be written there) and the parity bit is zero, the parity checker senses wrong, or even, parity. Therefore, the register select decoder forces a Parity OK condition. During a read cycle, the TM78 may return a single bit or no bits. When the Massbus controller reads the attention summary register, all drives connected to that Massbus respond. So, any number of attention bits may be read back. Parity cannot be anticipated or generated in this case, and the Massbus controller does not check for it.

The second type of variation to the basic Massbus read/write cycle is an access to hardware control registers R20 and R21. Any host request to a device register above 17_8 causes the Massbus controller to assert the most significant register select line, RS4 (N)H. RS4 (N)H goes to the CAS enable/disable AND gate, where it is enabled; this negates the signal CL CAS ENAB L. This disables the CAS RAM so that nothing may be written to or read from it. Two other significant outputs from the register select decoder come into play when accessing these registers. Any access to register 20, the microbus address register, produces an output from CL R20 L. Similarly any access to register 21, the microbus data/control register, produces an output from CL R21 L.

While most bits in registers 20 and 21 may be written to and read from, a certain sequence of operations must be observed. If the host wishes to write to R20, or read/write to the low byte of R21, the microcomputer must be put on HOLD first. This is done by writing a one into bit position 8 of R21. This causes the CAS hold flip-flop (Figure 2-27) to set, which sends IA HOLD (N)L to the microcomputer. After finishing the current instruction, the microcomputer ceases activity; it asserts HOLD A H (hold acknowledged) back to the CAS module, thus relinquishing control of the microbus to the host computer. IA HOLD H and HOLD A H are ANDed together to produce the signal IAR EN L. It is IAR EN L that allows the host computer to take and keep control of the microbus as long as hold stays set. Hold stays set until the host writes a zero to R21 bit 8, or performs a system (Massbus) initialization.

Figure 2-27 CAS Miscellaneous Control Logic



Register 20 is the 16-bit microbus address bus. The host may read or write R20 through the interface address register (IAR). Microbus A15:0 H is gated to CAS bus 15:0 L on a Massbus read from R20 through the read IAR by the signal CL RD R20 L. This signal is produced by a combination of CL R20 L, CL SEL DEM L, and the negation of C TO D H. It is not necessary for the microcomputer to be on hold for the host to read R20.

NOTE
It is possible for the host to read R20 at a critical point in time, when the microcomputer is between microbus cycles. In this case the A lines, and subsequently the CAS lines, will be changing and the parity generator will not have enough time to produce the correct Massbus parity bit. The result is a Massbus parity error (MCPE) detected by the controller; this should be ignored and the operation repeated.

I bus 15:0 H is gated to microbus A 15:0 H on a Massbus write to R20 through the IAR by the signal CL WR20 L. The write IAR has already been enabled by the signal IAR EN L (hold). CL WR20 L is produced by a combination of CL R20 L and CL CAS FM MB L.

The low byte of register 21 is the microbus data bus, the high byte is reserved for control and status. The host computer must put the microcomputer on hold to read or write the low byte. During a Massbus read from R21, microbus D 7:0 H is gated to CAS 7:0 L, and the eight high byte bits are gated to CAS 15:8 L through the interface data register (IDR) by the signal CL RD R21 L. If the microcomputer is not in hold state, the low byte will be read as zeros. CL RD R21 L is produced by a combination of CL R21 L, CL SEL DEM L, and the negation of C TO D H.

For a Massbus write to R21, data from the I bus is gated to the appropriate destination. For the low byte, I 7:0 H is gated to microbus D 7:0 H through the IDR by the signal CL WR21 H. The write IDR (low byte) has already been enabled by IAR EN (hold) and C TO D. CL WR21 H is produced by a combination of CL R21 L and CL CAS FM MB L.

The high byte of the write IDR consists of a group of control/status flip-flops. Of the eight bits in the high byte, only four may be written to by the host computer: hold (bit 8), even Massbus parity (bit 10), illegal register (bit 12), and TM clear (bit 14). Hold has been described earlier in this section as a control element for the microcomputer.

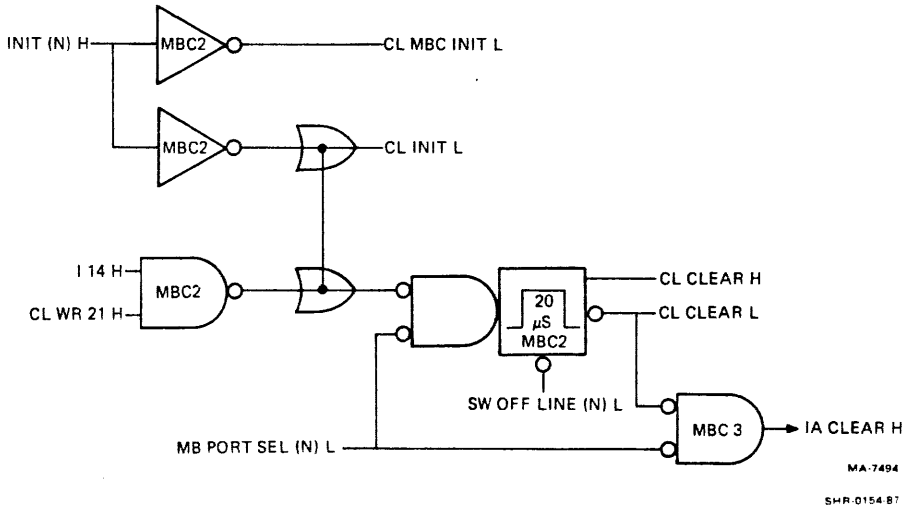
The even Massbus parity flip-flop is a control element used in standalone diagnostic mode only. It is set by asserting bit 10 to register 21 and sends the signal IA EV MB PAR H to the CAS bus parity checker/generator. There it switches the checker/generator from odd to even parity to force a parity error to occur both on Massbus reads and writes.

The illegal register flip-flop is a status bit for both the host computer and the microcomputer, and may be set in one of two ways. First, it may be clocked set by asserting bit 12 to register 21. This serves as a functional diagnostic check only. Second, it may be direct-set as a result of a Massbus access to an illegal register. If RS4:0 selects a register code equal to or greater than 228, the register select decoder produces a low true output from bit 0 ILR. This is ANDed with CL SEL DEM L and sets the ILR flip-flop. The ILR flip-flop produces the signal CL ILR H, which becomes status bit 12 when reading register 21. The flip-flop also enables an AND gate which, assuming this Massbus port is selected, produces IA ILR H over the microbus to the microcomputer and causes it to interrupt.

Signal TM Clear (Figure 2-28) is a read/write bit which does not use a holding register flip-flop. Writing bit 14 to register 21 enables an AND gate which produces the signal CL INIT L. This signal resets the even Massbus parity, illegal register and TM ready flip-flops. If this port is currently selected and not off-line, CL INIT L also fires off a 20 microsecond one-shot (CL CLEAR); produces IA CLEAR H through an AND gate. IA CLEAR H is sent to the microcomputer by means of the microbus, and resets the microprocessor chip. Signal INIT (N) H, a derivative of Massbus initialize, power supply DC LO, or the TM78 MASTER RESET pushbutton, also produces the CL INIT L along with the CL MBC INIT L. CL MBC INIT L resets only the hold flip-flop. This isolation of the two types of clear pulses prevents a TM Clear command through register 21 from resetting a microcomputer hold condition.

To write and read data through register 21, the host must be able to manipulate certain microbus control lines. This is done by means of a tri-state driver enabled by IAR EN L during hold state. The inversion of Massbus signal C TO D becomes microbus signal READ H, the decoded signal CL RD R21 L becomes microbus signal READ STROBE H, and the decoded signal CL WR21 L becomes microbus signal WRITE H. With control over these microbus lines plus access to the A and D lines, the host can write or read practically any available microbus address.

Figure 2-28 CAS Clear Logic



Microbus Write/Read From Common Address Space — Figure 2-29 shows the control circuitry used in CAS accesses from the microbus. Refer to this figure and Figure 2-23 for this discussion. The alphanumeric characters (prefixed MBC) in all functional logic elements found in these figures refer to the engineering drawing page number for the CAS module (M8957) where the actual logic element may be found.

- **Write**
  Figure 2-30 shows the timing relationships between microbus control signals and internal CAS control signals during a write cycle. The microcomputer, through the microbus, may read or write one byte (one half of a given CAS register) at any time. Since there are 16 registers, that means 1 of 32 bytes may be accessed. The 32 CAS bytes are addressed in the microcomputer's I/O page range.

Figure 2-29 CAS Microbus Control Circuitry

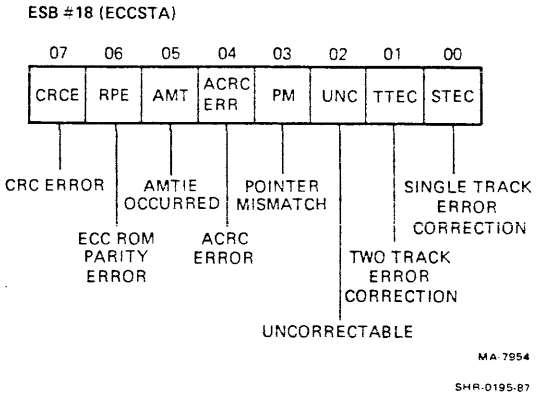
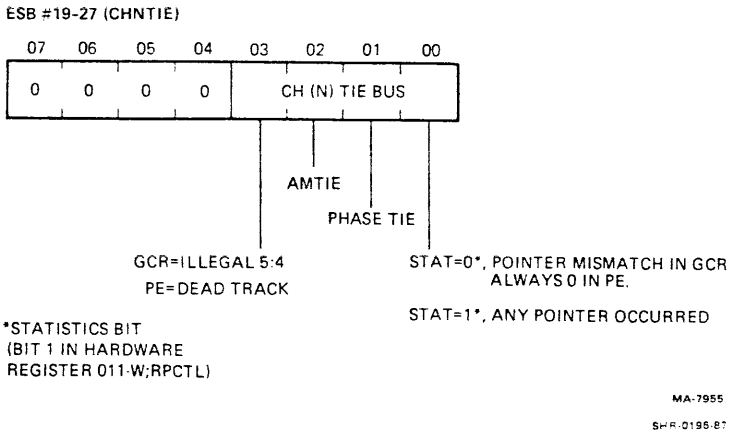


Figure 2-30 Timing for Internal Write to CAS



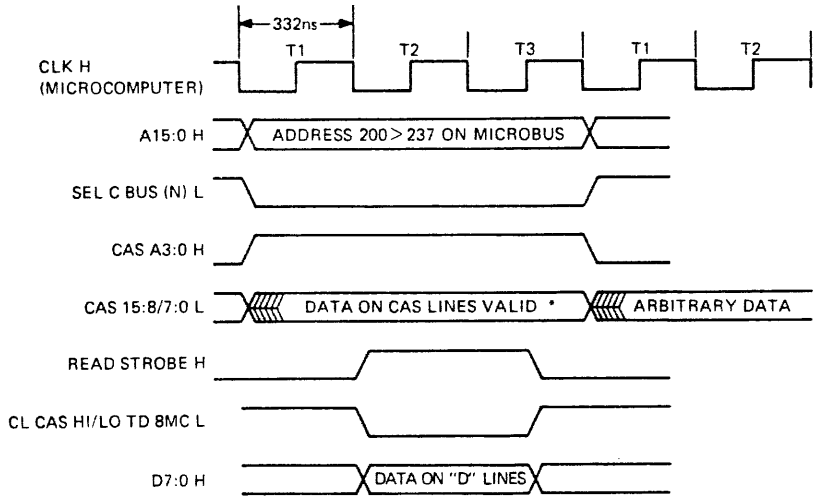
At T1 time the microcomputer asserts an address in this range on the address bus (A15:0). It is decoded as an address in the range of 2008—2778, and the microcomputer asserts the microbus signal SEL C BUS (N) L. This signal and the negation of address bit 5 (any CAS address) primes the four control gates. Address bit 0 (A0 H) determines whether data will be written to the HI or LO byte of the register word. Address bits A4:1 H are input to the A inputs of the CAS address multiplexer and become CAS A3:0 H to address 1 of 16 CAS RAM registers.

At T2 time the microcomputer asserts data on the microbus D lines (D7:0 H) with WRITE H. WRITE H enables the appropriate control gate and produces CL CAS WR HI/LO L to strobe the data to CAS RAM. Note that the microbus D lines are impressed on the B inputs of both the HI and LO byte CAS data multiplexers. This means that the same data is present at the HI and LO byte RAM inputs. However, only one of the RAM chips is strobed by the write signal, so the other RAM retains its data. Although the data is immediately felt at the RAM outputs (CAS bus), a parity check is not performed.

- Read

  Figure 2-31 shows the timing relationships between microbus control signals and internal CAS control signals during a read cycle. At T1 time the microcomputer asserts an address in the range of 80—9F HEX ($200—237_8$) on the address bus (A15:0 H). It is decoded as an address in the range of 80—9F HEX ($200—237_8$), and the microcomputer asserts the microbus signal SEL C BUS (N) L. This signal and the negation of address bit 5 (any CAS address) primes the four control gates. Address bit 0 (A0 H) determines whether data will be read from the HI or LO byte of the register word. Address bits A4:1 H are input to the A inputs of the CAS address multiplexer; they become CAS A3:0 H to address one of 16 CAS RAM registers. The CAS RAM then asserts the full word in that CAS bus location (CAS 15:0 L). This word is then available at the inputs of the CAS HI and LO byte registers.

Figure 2-31 Timing for Internal Read from CAS



* PARITY CHECK IS NOT PERFORMED

MA-7497

SHR-0175-87

At T2 time the microcomputer asserts the READ STROBE H microbus line. This enables the appropriate control gate, which produces CL CAS HI/LO TO 8MC L to strobe one of the two CAS byte registers to the microbus D lines. Then the data is gated to a location within the microcomputer. Again, as with the write cycle, data parity is not checked.

- Miscellaneous

  Two additional microbus CBus control and status registers are decoded on the CAS module. Figure 2-27 shows the logic employed for this decoding. Here CL SEL C BUS (N) L and A5 H are ANDed together to produce an output enabling four control gates. This enabling signal will be true for any address in the range of A0—9F HEX (240_8—277_8) but only two addresses in that range are actually used.

  Microbus address A0-W (240_8-R) is decoded to produce the signal CL 8MC TO CAS CTRL L. This signal strobes the CAS control register. Refer to Appendix E for a bit description of this and other registers described here. Bit 7 resets the contention flip-flop after handling the interrupt from this error. Bit 6 resets and sets the TM ready bit prior to and after a command execution.

  Microbus address A0-R (240_8-R) is decoded to produce the signal CL CAS STAT TO 8MC L. This signal strobes the CAS status register to the D lines. Microbus address A1-W (241_8-W) is decoded to produce the signal CL SET ATTN L where it direct sets the Massbus ATTN flip-flop. The microcomputer simply has to address this location to set ATTN; no data bits are necessary. Finally, microbus address A1-R (241_8-R) is decoded to produce the signal CL READ C BUS (N) L. This signal is sent to the Massbus data module, where it strobes the CBus status register to the D lines.

2.4.3 M8956 Massbus Data Module (MBD)

The M8956 Massbus data module interfaces the TM78 read/write data paths to the Massbus controller. It also contains circuitry to interface the Massbus DBus/CBus control lines to the microcomputer (M8960). During a write operation the Massbus data module receives 18-bit data inputs from the Massbus and transfers all 18 bits to the write microcontroller module (M8959). The particular Massbus controller used may in fact transfer either 16 or 18 bits across the Massbus, but the Massbus data module does not make this distinction. During a read operation the Massbus data module receives 16- or 18-bit data inputs from the write microcontroller module and transmits them to the Massbus controller.

Data is processed through the Massbus data module either as a parallel transfer (left format) or shifted two bits with an end around carry (right format). The data parity bit (DPAR) is not checked or generated on the Massbus data module, but simply passes through, going to or coming from the write microcontroller.

Data Transfer Formats and Multiplexing — Figure 2-32 shows the formats for DECsystem-10/DECSYSTEM-20 and VAX 11/780 data words and how they are placed on the Massbus for processor/Massbus transfers. Both formats place the most significant bits of the word (or halfword) in the high-numbered bit locations on the Massbus. In the VAX format, bit 0 is the LSB; in other formats, bit 0 is the MSB.

The Massbus data module assembles the 18-bit Massbus word into one of two formats, left or right. The decision of whether to format in left or right mode comes from the write microcontroller module, but the actual format assembling is accomplished by data multiplexers on the Massbus data module. Figures 2-33 (left format) and 2-34 (right format) show how the individual bits are assembled in either formatting mode.

In each figure, the upper register <17:0> represents the 18 Massbus data bits entering or leaving the Massbus data module. The lower register <17/15:0/16> represents the 18 bits going to or coming from the write microcontroller module. Each of these bits is represented by a double numeric signal name. The number to the left of the slash is the Massbus line logically connected to that line when in the left formatting mode. Similarly, the number to the right of the slash is the Massbus line logically connected to that line when in the right formatting mode.

Figure 2-32 Processor Data Word Formats on Massbus

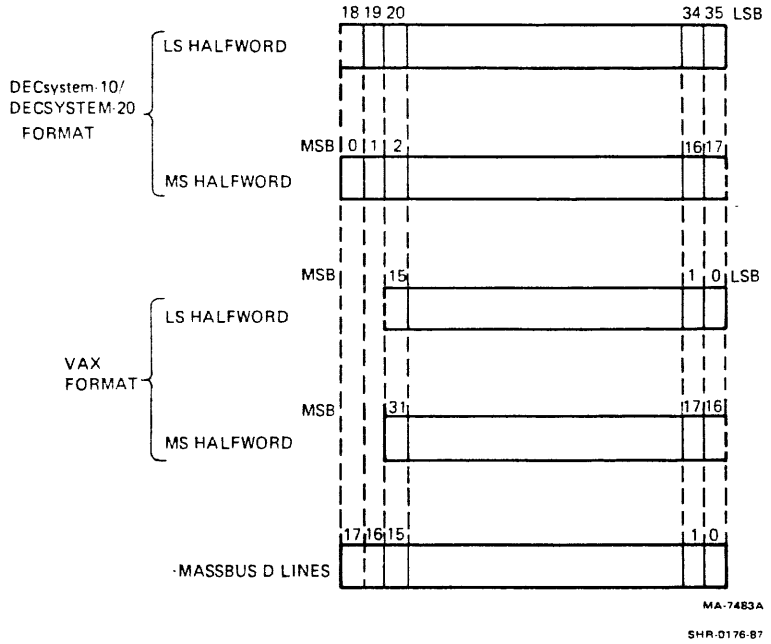


Figure 2-33 shows the left formatting mode. This mode transfers the high-order 18 bits of the DECsystem-10/DECSYSTEM-20 word (bits 0:17). Left mode also transfers extended sense data from the microcomputer, through the write microcontroller and Massbus data modules, to the host computer. As shown in this figure, the data multiplexer performs an 18-bit parallel transfer with no shift, in left mode.

Figure 2-34 shows the right formatting mode. Right mode handles all data transfers except the high-order DECsystem-10/DECSYSTEM-20 halfword and the extended sense word. This includes the low-order 18 bits of the DECsystem-10/DECSYSTEM-20 word (bits 18:35), and both 16-bit VAX halfwords (bits 31:16 and 15:0). Here the data multiplexer performs an 18-bit transfer, but the bits shift two binary places to the left with the two high-order bits appended to the right.

The reason for the left/right format multiplexing is to arrange the bits of a 36-bit DECsystem-10/DECSYSTEM-20 word into 4-bit segments (nibbles). When written to tape, these segments are compatible with existing tape formats. Paragraph 2.3.2 discusses the process of extracting 4-bit nibbles from the 18-bit multiplexed data.

Operation — Figure 2-35 shows the M8956 Massbus data functional block diagram. The module is broken down into three functional areas: Massbus data multiplexing, DBus control/status and CBus status.

Figure 2-33 Data Multiplexer (Left Format)

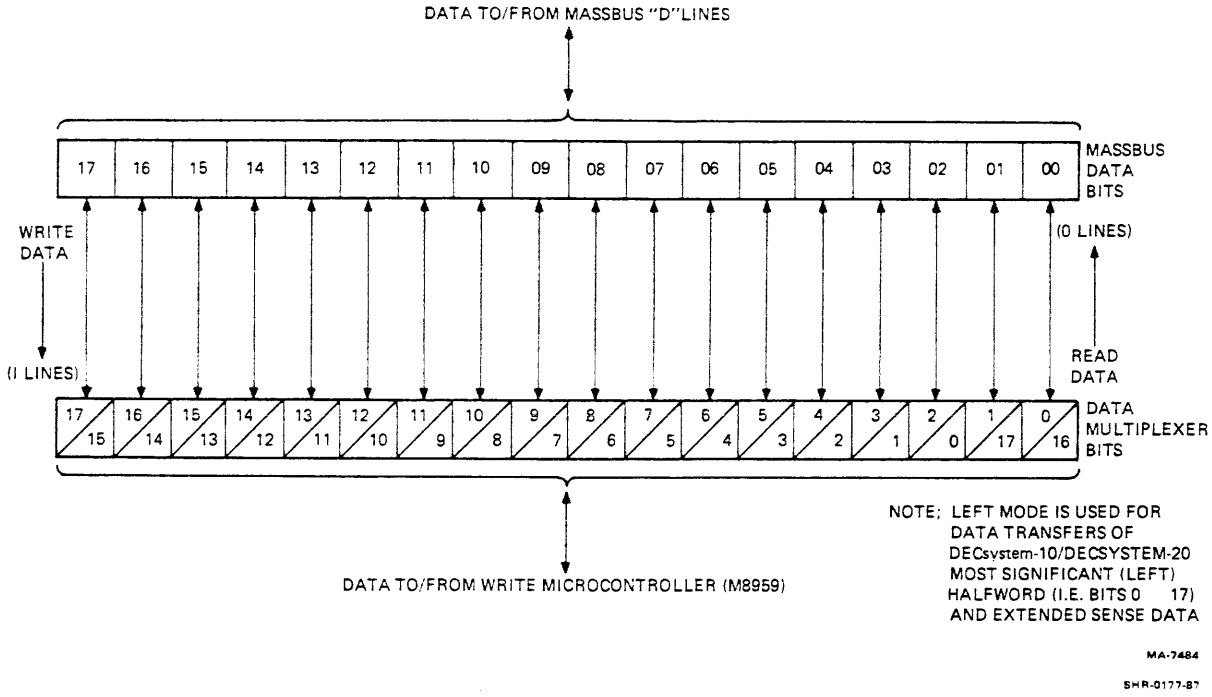


Figure 2-34 Data Multiplexer (Right Format)

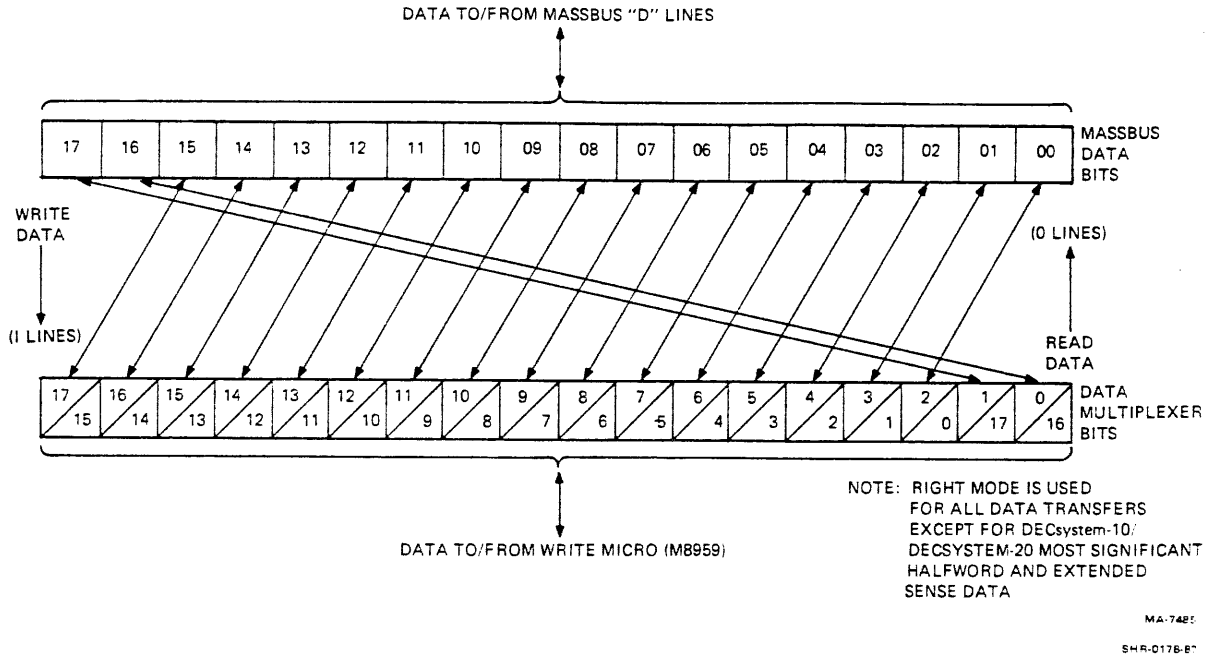
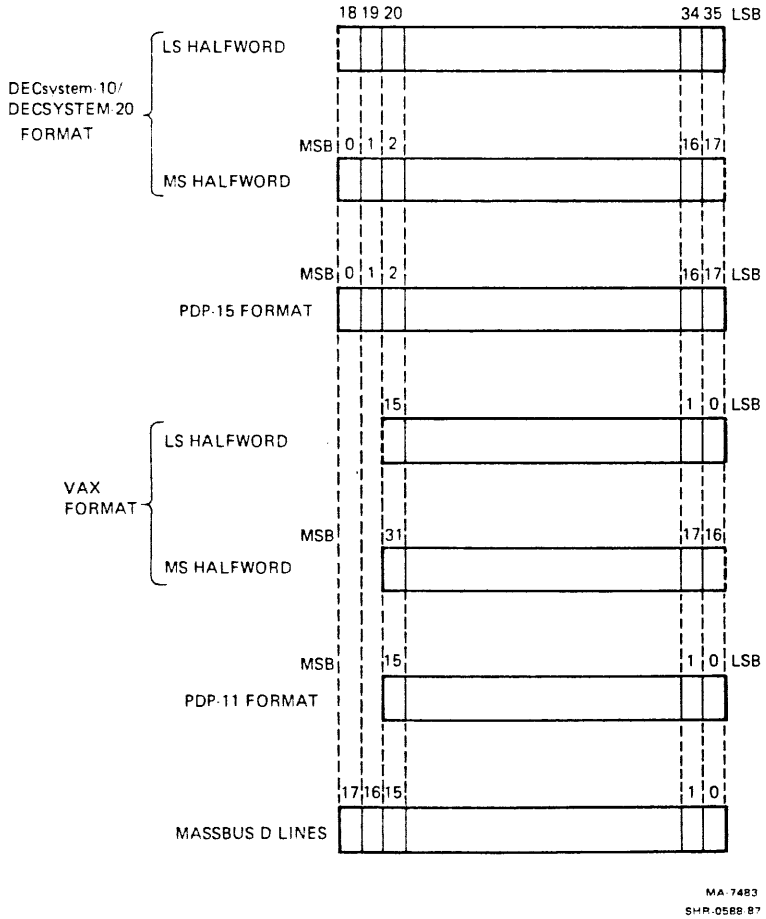


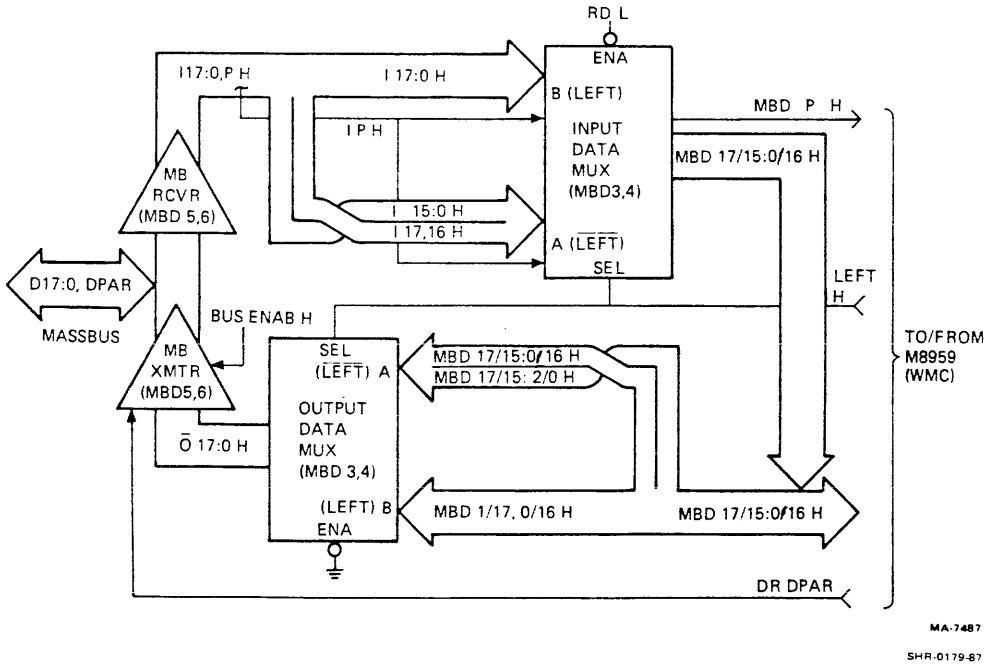
Figure 2-35 M8956 Massbus Data Functional Block Diagram



Data Multiplexing — The Massbus data module handles Massbus data through a set of I/O data multiplexers. Figure 2-36 shows the data path/multiplexing portion of the module. In a write operation, 16/18-bit Massbus data and data parity is converted to TTL levels by 75107 receivers (MBD 5, 6) and become the I17:0, I P H data bus. This bus is presented to the input data multiplexer (MBD 3, 4), where it is gated by the RD L signal out to the write microcontroller over the bidirectional multiplexed data lines MBD <17/15:0/16>H. The LEFT H signal from the write microcontroller instructs the input multiplexer to gate the I lines through in left mode (parallel) using the B inputs (LEFT H asserted) or in right mode (shifted) using the A inputs (LEFT H unasserted). In either mode, the DPAR parity line is sent through to the write microcontroller as MBD P H, where it is checked for odd parity.

In a read operation, the write microcontroller asserts 16/18-bit data on the multiplexed data lines <17/15:0/16> H along with the odd parity bit DR DPAR H. The output data multiplexer gates the multiplexed data lines to the O lines in left or right mode, again depending upon the state of the LEFT H signal. The BUS ENAB H signal gates the O lines (17:0), along with DR DPAR, to the Massbus D lines through 75113 transmitters.

Figure 2-36 Massbus Data Multiplexers



DBus Control/Status — The microcomputer (M8960) initiates and terminates a read/write data transfer operation through the Massbus data module. This is accomplished by microbus transfers to the DBus control and status registers. (Refer to Appendix E, internal address (300_8), for bit descriptions of the DBus control, (300_8-W), and DBus status, C0-R (300_8-R), registers.) The write microcontroller handles actual control and timing handshaking during the data transfer.

The DBus control register (MBD 1) asserts the OCC, EBL, and EXC Massbus data bus control lines to the Massbus controller. It asserts simulated Massbus SCLK, WCLK, and RUN test signals when running internal diagnostic routines. Finally it asserts the write enable and read enable signals; these signals when ANDed with the MB PORT SELECT signal from the microcomputer produce the bus enable (BUS ENAB H) and read (RD L) signals to the data multiplexer. In this case, write enable/bus enable refers to writing read data to the Massbus D lines. Similarly, read enable/read enables write data to be read from the Massbus.

The DBus status register (MBD 1) receives the RUN and EXC Massbus data bus control lines from the Massbus controller. It also receives bus enable, WCLK, EBL, OCC, SCLK, and SCLK OUT signals and transmits them to the microcomputer as status or confirmation of a diagnostic routine's success.

At the start of a data transfer operation, the microcomputer must enable one of the two Massbus data modules. It does this by asserting one of the MB PORT SEL (N) L microbus lines to the port to be selected (where N = 0 for Massbus A and 1 for Massbus B). Once the port is logically connected to the Massbus in this manner, the microcomputer asserts the Massbus occupied (OCC) line to the controller by writing a one into the OCC bit of the DBus command register. The controller sees OCC and responds by asserting RUN back to the formatter, where the microcomputer reads it in the DBus status register.

After this initial dialogue between the TM78 and the Massbus controller, data may be transferred. In a write operation the microcomputer enables the input data multiplexer by writing the read enable bit into the DBus control register. The write microcontroller asserts the sync clock (SCLK) Massbus line through the Massbus data module, and informs the controller that it is ready to accept a 16/18-bit data word. The Massbus controller responds by asserting the write clock (WCLK) line, which is received by the Massbus data module and transmitted to the write microcontroller as WCLK (N) H. When SCLK H goes false, the write microcontroller strobes the data from the data multiplexer. As the WCLK line goes false, the Massbus controller changes the data on the data bus lines and waits for another SCLK pulse. This handshake continues until the byte count register in the write microcontroller overflows, indicating that no further data will be written to tape. The microcomputer senses the overflow and checks for errors from the read-after-write process; then it writes the SEND EBL bit into the DBus control register which transmits end of block (EBL) to the Massbus controller. The write data transfer terminates on the trailing edge of EBL, if the Massbus controller has negated RUN. If the Massbus controller does not negate RUN, another record of the same size is written.

In a read operation, the microcomputer enables the output data multiplexer transmitters by writing the MB WR EN H bit into the DBus control register. The write microcontroller asserts 16/18-bit data from the read path onto the data multiplexer lines along with SCLK. The Massbus data module transmits the data and SCLK to the Massbus controller over the Massbus data lines. When SCLK goes false, the Massbus controller strobes the data through to its internal data channel. The write microcontroller repeats this process until the magnetic tape reaches the end of a data record. Then the microcomputer writes the SEND EBL bit into the DBus control register, which transmits end of block (EBL) to the Massbus controller. The read data transfer is terminated on the trailing edge of EBL, if the Massbus controller has negated RUN.

CBus Status — Ten Massbus CBus control lines are received by the Massbus data module and transmitted to the CAS module (M8957). They include the three drive select lines (DS2:0), five register select lines (RS4:0), controller to drive (C TO D), and demand (DEM). One Massbus CBus control line, the transfer (TRA) line, is transmitted to the controller by the Massbus data module from the CAS module. Other miscellaneous lines handled through the Massbus data module are initialize (INIT), attention (ATTN), and Massbus fail (MASS FAIL). INIT originates in the controller and is sent to CAS as INIT (N) H. This line may also be asserted to CAS by the signal SW RESTART L, which is an OR of the power supply DC LO signal and the MASTER RESET pushbutton signal. ATTN originates in the CAS and is transmitted to the controller on the Massbus line ATTN.

The MASS FAIL line originates in the controller and notifies the TM78 of a system-level power-fail condition through the Massbus data module. MASS FAIL is normally at a logic low, but is pulled up to +5 V in the Massbus data module and will go to a logic high if the Massbus cable is disconnected at either end.

The CBus status register on the Massbus data module senses some of the CBus control lines and miscellaneous lines. The CBus status register is enabled to the microbus D lines by the CL READ C BUS (N) L signal which originates on the CAS module. CL READ C BUS (N) L is decoded and generated during a microbus read cycle to internal address 1A-R. (Refer to Appendix B, address 1A-R for a definition of each CBus status register bit.) The Massbus lines read through the CBus status register are ATTN, TRA, DEM, INIT, and MASS FAIL. Other signals read through the CBus status register are LEFT, minus five volts on (5 V ON), and port off-line (SW OFF LINE (N) L). The 5 V ON signal originates from a +5 V to —5 V divider and open collector inverting driver on the Massbus data module. If the +5 V power supply fails, the +5 V OK H line goes low and is read back to the microcomputer as a zero. This line also drives a LED on the Massbus data module that lights to indicate a —5 V failure (assuming the +5 V supply is still functioning properly).

The SW OFF LINE (N) L signal originates on the backplane DIP switch and notifies the microcomputer, through the CBus status register word, that an operator has taken that Massbus port off-line relative to system commands.

2.5 FORMATTER

Paragraphs 2.5.1 through 2.5.6 describe modules common to both the TM78 and TS78 formatters.

2.5.1 M8950 Read Channel (RC)

The formatter uses nine M8950 read channel modules, one for each physical tape track. During a PE or GCR read/write operation, the module's main function is to deskew data and pass it on to the ECC (M8951) logic. During a GCR read/write operation, the module also translates the 5-bit GCR formatted data back into the original 4-bit format and detects format control characters.

Figure 2-37 is the M8950 program macro flowchart, and Figure 2-38 is the simplified block diagram. Refer to these figures while reading the following paragraphs.

Phase-Locked Clock — Converting data read from the tape into binary data requires generation of a data window for each track, synchronized with the data transitions in each track. A phase-locked clock generates the data window and keeps it synchronized with the incoming data stream. The heart of the clock (a phase-locked loop) comprises a voltage-controlled oscillator (VCO) and a phase detector.

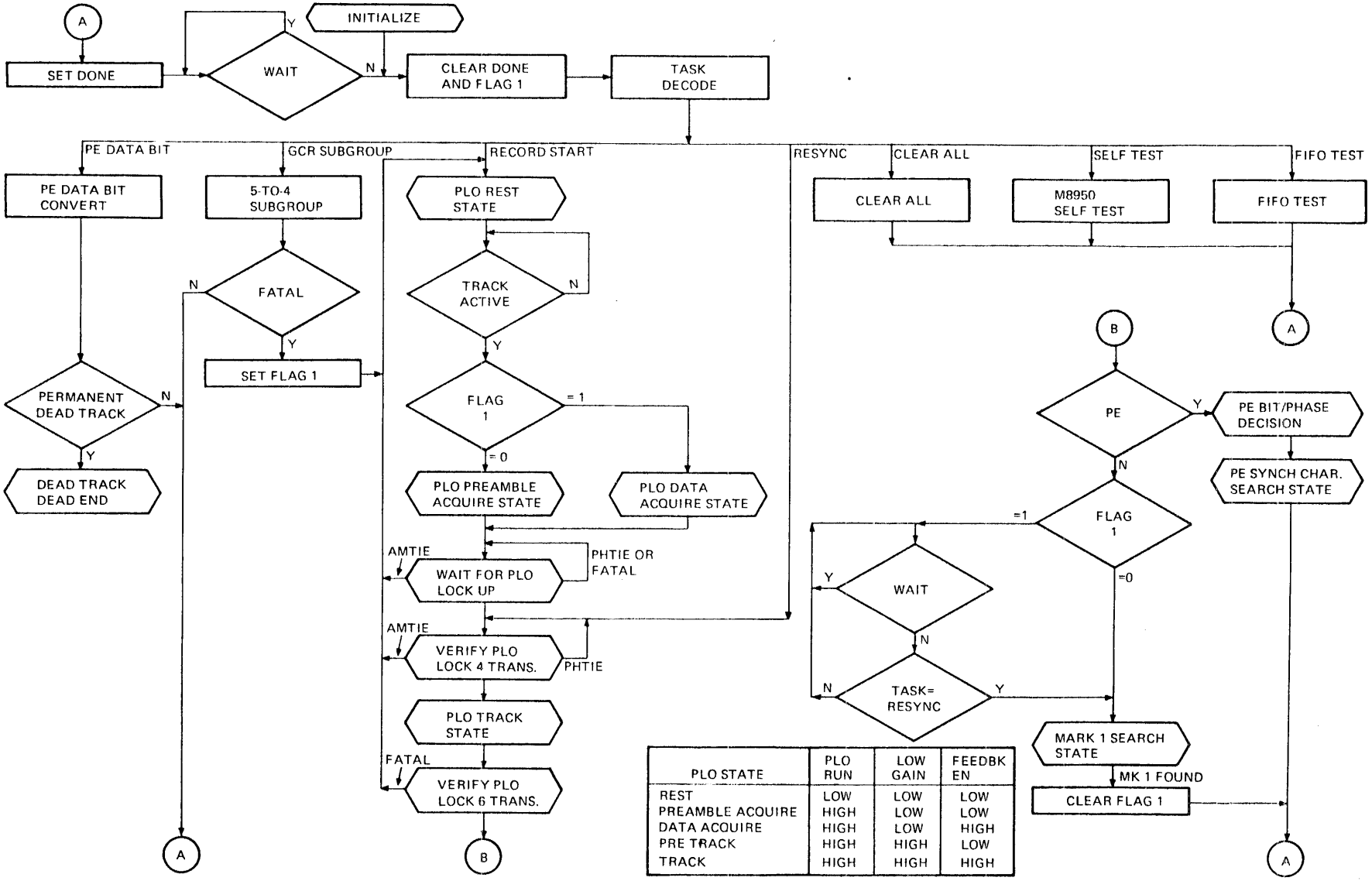
The phase-locked loop (PLL) operates in the following manner. Three control lines control all PLL action: PLO RUN, FEEDBACK EN, and LOW GAIN. A fourth control function (PE or GCR) selects the loop filter bandwidth. As shown in the table in Figure 2-37, combinations of these control signals generate five PLO states in which the PLL logic can operate.

The rest state returns the VCO control voltage to a known (nominal) level between records, in order to be ready to generate the correct VCO rate (PE or GCR). The preamble state initially captures the phase lock. The data acquire state relocks the PLL during the data (as opposed to preamble) portion of the record.

When PLO RUN asserts, the TU port read data can enter an edge detector. A flip-flop cleared by the FIFO CLK stores every data transition through the edge detector. Its output signal is then phase-compared with the ring counter output PHASE. PHASE, however, is first shifted 90 degrees so that the compared result from the phase detector (XOR) is a square wave. If PHASE is disabled by a low on FEEDBACK EN, the signal compared with the data in the phase detector is held constant. (Nothing is compared with the data.) This is useful, for example, when trying to lock up on data after an error. It makes the data rate force the PLL to run at the proper frequency.

The signal leaving the phase detector is clamped at +1.4 V and —1.4 V levels by clipping diodes. This output then passes through a switchable attenuator. At the beginning of a read operation, the logic is starting to acquire data so no attenuation is used to provide fast lockup. Once lockup is attained, LOW GAIN asserts and the ERROR signal is attenuated.

Figure 2-37 M8950 Read Channel Program Macro Flowchart

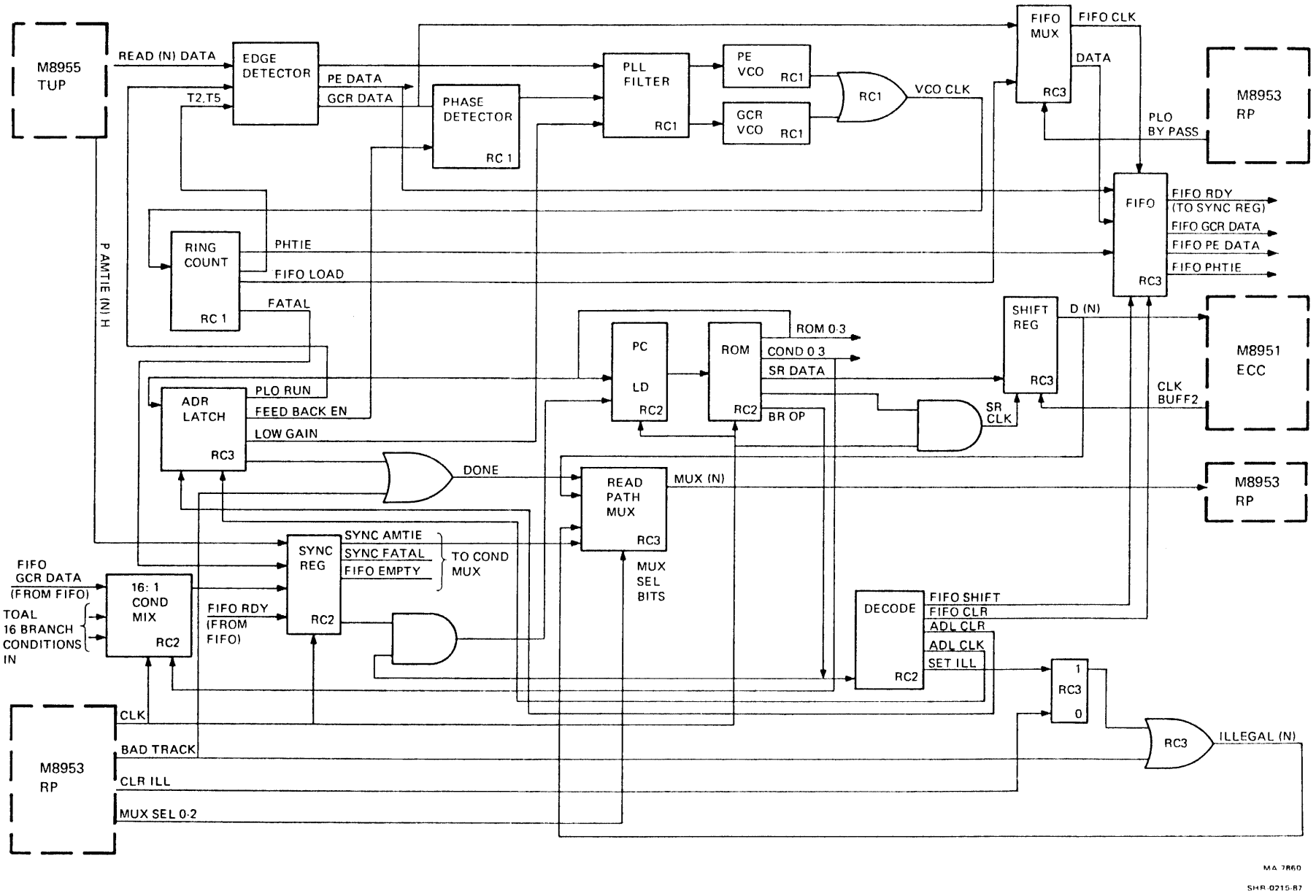


PLO STATE	PLO RUN	LOW GAIN	FEEDBK EN
REST	LOW	LOW	LOW
PREAMBLE ACQUIRE	HIGH	LOW	LOW
DATA ACQUIRE	HIGH	LOW	HIGH
PRE TRACK	HIGH	HIGH	LOW
TRACK	HIGH	HIGH	HIGH

MA 7476

SHR-0214-87

Figure 2-38 M8950 Read Channel Simplified Block Diagram



A dual-bandwidth filter (depends on PE or GCR speed) next extracts an average value from the incoming square wave. Therefore, if the square wave is higher more often than it is low, the average value is higher than if the wave were symmetrical. The filter operates as an integrator; if the average value of the square wave is higher more often than it is low, the filter output gradually increases. As the phase comparator output changes the duty cycle, the filter output voltage changes accordingly. The filter output voltage is applied to the control voltage pin of both VCOs. The enabled VCO (PE or GCR) outputs the VCO CLK signal.

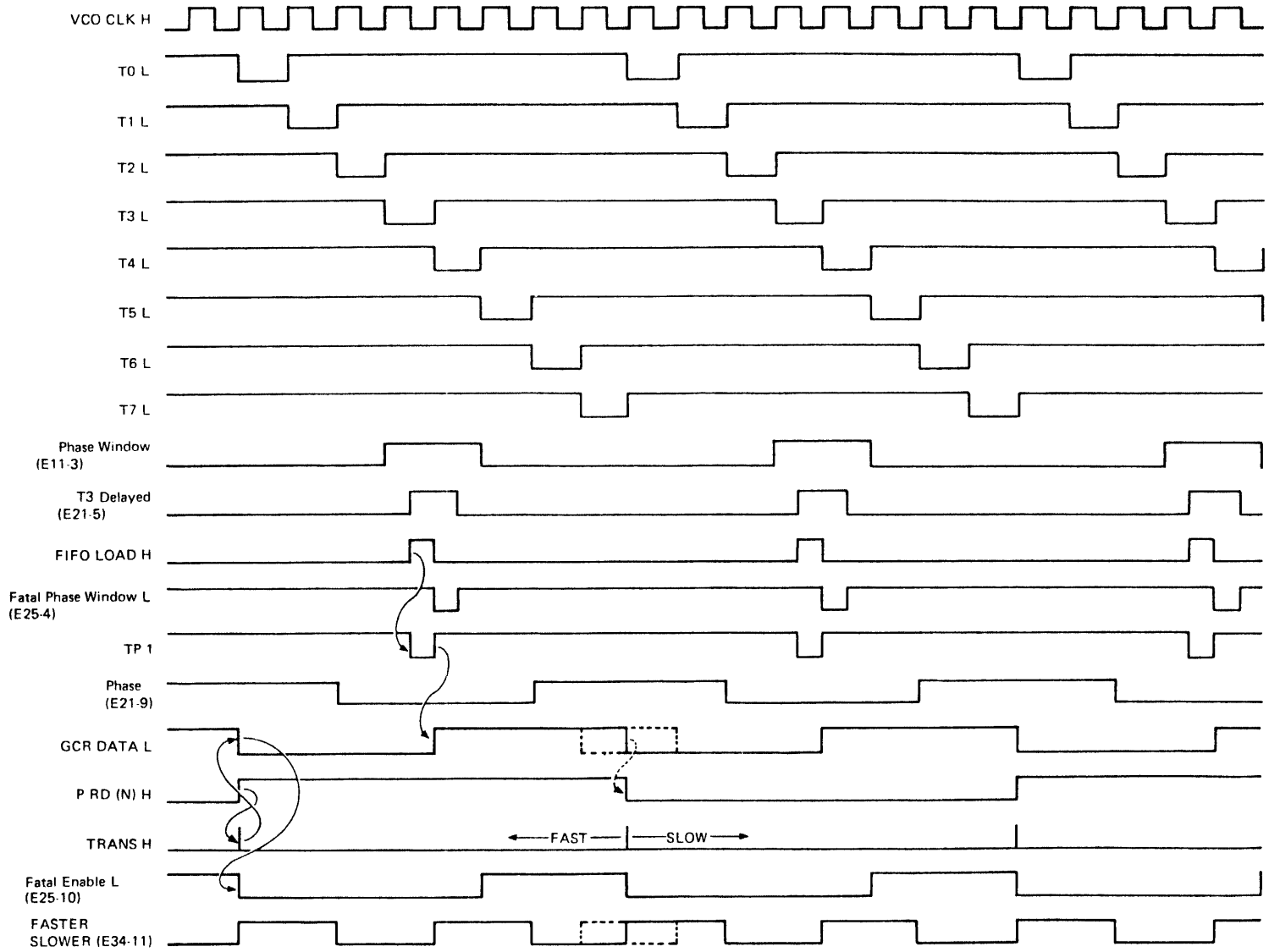
The VCO CLK signal pulses a self-correcting 8-bit ring counter. (See Figure 2-39.) Shifting at the VCO CLK rate, the resulting T times (0—7) generate the "comparing" PHASE signal compared with data in the phase detector described earlier. These T signals also generate windows that bracket the bad and very bad data areas that create PHTIE and FATAL, respectively, if the transition occurs during that time. With VCO CLK, the T signals also generate FIFO LOAD; this indicates to the FIFO when the GCR data is valid and clocks the data into the FIFO.

The PHTIE flip-flop records any unexpected transition and indicates the PLL may be having difficulty keeping up with the data. The flip-flop remains set until the microprogram recognizes it, then pulses it clear with PH CLR. FATAL is similar, except the PLL may be running too slowly. If a transition sets FATAL (window is even smaller than for PHTIE), the microprogram stops and attempts to relock the PLL. In GCR mode, an opportunity exists to recover from FATAL during the Resync burst time. Therefore, the PLL attempts to resynchronize prior to the Resync burst.

ROM Controller — A ROM (16 bits by 512 words) controller provides the programmable section of this module. A 9-bit program counter drives the ROM address lines. Certain ROM output lines (ROM 0—8) circle back to the PC inputs, providing a preset capability. The remaining ROM outputs define what is to be performed in a particular instruction, and a condition (COND 0—3) that decides whether or not to perform the next branch instruction. The four condition select lines control a multiplexer that selects one of the 16 possible branch conditions, including +3 V for an unconditional branch. The multiplexer output inputs to the sync register, which synchronizes the condition to the clock. Once synchronized, the output is gated with a branch opcode ROM bit (BROP) that dictates whether or not a branch will be performed next. If BR OP is asserted true, a new address is clocked into the PC after a one-instruction delay caused by the sync register. This addresses the ROM, and the next instruction is read. The condition for branch instructions is specified in the instruction prior to the branch.

A typical instruction is a NOP, which effectively does nothing. All ROM outputs are zeros. Assume that the next condition that appears in the assembler instruction is a reverse condition. Because a NOP is the current instruction, all the ROM outputs are zero except for the condition that corresponds to reverse. During the instruction time, the condition multiplexer selects the reverse position. At the end of the instruction time, this is clocked into the sync register; this ends the current instruction and starts the next one. If the next instruction is a branch, BR OP and the condition pulse the PC load line and cause the ROM to branch.

Figure 2-39 Read Channel PLL Timing Diagram



NOTE:
TITLES IN LOWERCASE LETTERS ARE NOT SIGNAL OUTPUTS BUT THE REPRESENTATIVE WAVEFORMS CAN BE OBSERVED AT THE ASSOCIATED PINS.

MA-7475
SHR-0216-87

An addressable latch stores read channel module states. Three of the ROM inputs select one of the eight flip-flops in the latch (routing the clock to that flip-flop), and the fourth ROM bit is the data for that latch. Among the outputs, four are flags used for monitoring internal status questions. For example, has the board asserted FATAL? Is this track still active? Is PLL still trying to lock up on the data? Another output is DONE, which asserts when the module is done with its specified function. DONE is ORed with BAD TRACK from the read path controller; this ensures that a bad module or a module with data trouble does not "hold up" the remaining read channel modules. When a module completes the task specified by the read path controller, it asserts DONE by setting that bit in the addressable latch.

If a problem exists (for example, FATAL asserts), DONE is not raised. The read path controller measures the period of every read channel operation. If an operation does not complete in approximately twice its expected time, DONE L is clocked into the read path controller's bad track register. This returns to the particular read channel module and asserts the DONE line. In this manner, the module is finished, temporarily out of the read path. It is then free to try to relock its PLL on the data without forcing the other read channel modules to wait.

GCR Data Translation — A read channel ROM controller program performs all GCR 5-to-4 bit data translations. After the GCR data is recovered by the PLL, it passes to the FIFO multiplexer and FIFO. The multiplexer (under diagnostic control) selects what goes into the FIFO data input and what the clock source should be, while the FIFO deskews the data. In a normal GCR read operation, the FIFO LOAD signal developed by the 8-bit ring counter of the PLL clocks the data into the FIFO. When the FIFO is ready to present the data at its output, FIFO RDY is asserted. It is synchronized in the sync register and inverted, becoming FIFO EMPTY. A typical instruction would be to branch to this same place on FIFO EMPTY. The program stays at this location until FIFO EMPTY is no longer asserted, indicating the FIFO is ready. Then the read channel ROM controller shifts the deskewed data out of the FIFO.

FIFO GCR DATA enters the condition multiplexer and moves on to the sync register, where it is synchronized with CLK (from the read path controller). Then it is decoded by the read channel program, which branches on each bit (one at a time).

The program performs a bit-by-bit check that decodes 1 of 32 possible codes (16 are legal). The binary code for each of these 16 legal 5-bit patterns leaves the ROM controller as SR DATA H and inputs to a shift register. It is clocked into the shift register by SR CLK.

Two 4-bit subgroups are input to the 8-bit shift register. When the shift register is full (one 8-bit group), the CLK BUFF 2 signal from the ECC module pulses the data [D(N)] serially out of the shift register. The data passes to the ECC module for error correction; here it is available to the read path controller through the read path multiplexer for diagnostics; this multiplexer can select translated data and other GCR codes (for example, Mark 2, End Mark).

2.5.2 M8951 Error Correcting Code

(ECC) — The M8951 is the error correcting code module. During a read/write PE operation, the module's main task is to provide single-track error correction. During a GCR read/write operation, the module performs single-track error correction and double-track error correction with pointers. The module has a ROM microcontroller initialized by the microcomputer and then controlled by the read path controller (M8953). This module's microcontroller implements all the error-correction algorithms.

Figure 2-40 is the M8951 simplified block diagram; refer to it while reading this section.

ROM Controller — A ROM (32 bits by 512 words) controller, similar to the read channel and read path controllers, provides the programmable section of this module. The program counter (PC) outputs drive the ROM address lines. Some of the ROM outputs handle program control (they provide a branch address on a branch instruction). The remaining ROM outputs are control lines that drive the arithmetic part of the module. An 8-input branch multiplexer inputs to a condition flip-flop (similar to the read channel's and read path controller's) and selects the desired input with a decode of COND 0—4. The BRANCH signal is gated with the condition flip-flop, which holds whether or not the selected condition was true. If the condition was true, the PC is loaded from the ROM branch address outputs on the succeeding CLK pulse.

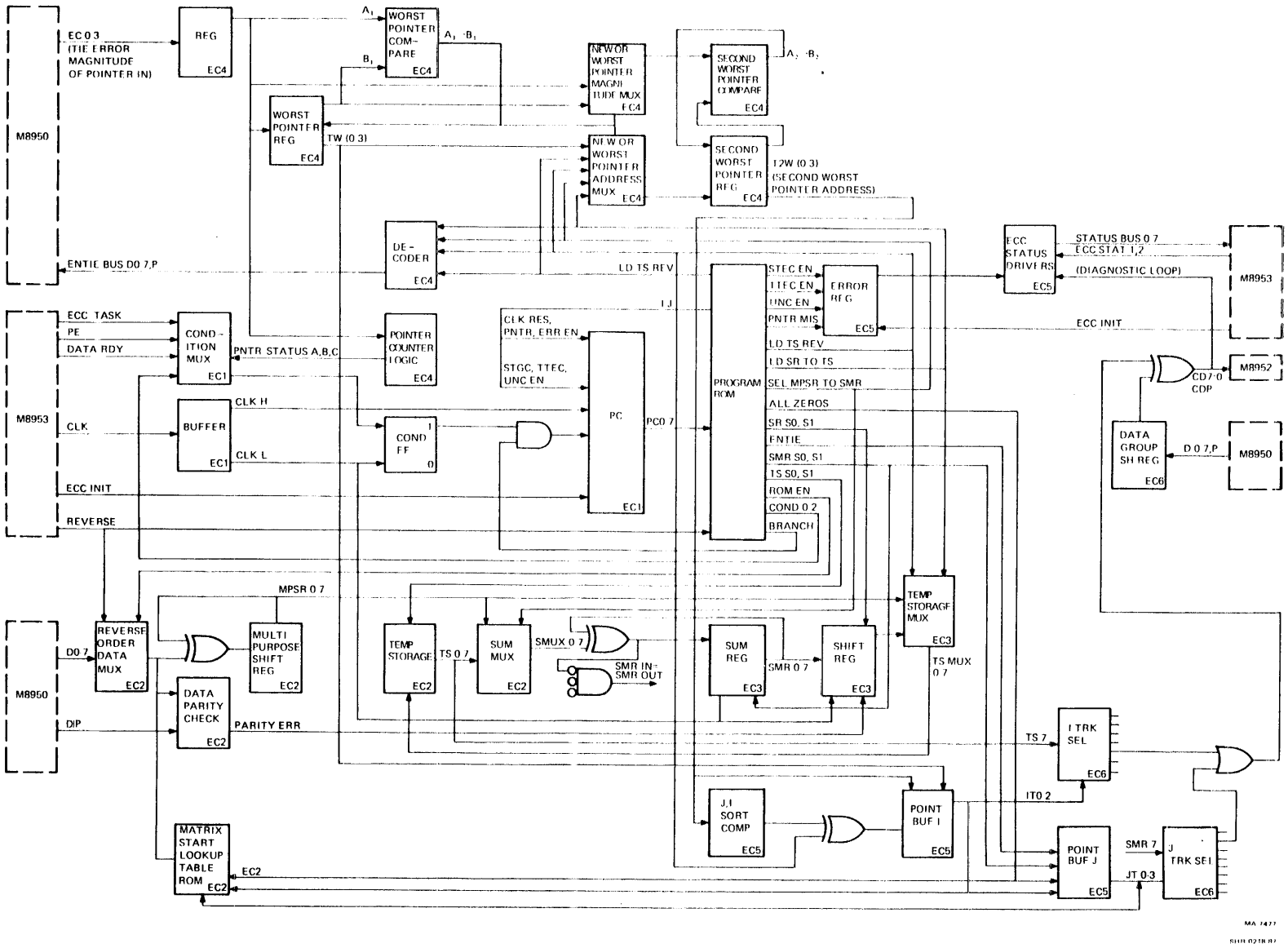
Nine bits are required to address the 512-word ROM. The PC is eight bits in length; the ninth bit is the REVERSE signal. This forces selection of either the forward or the reverse microprogram. Everything that this module's controller can branch on comes from the condition multiplexer. These conditions include pointer status (PNTR STATUS A-C), ECC TASK, PE, unconditional, SMRIN=SMROUT and DATA RDY.

Arithmetic Section — The data read from tape may contain errors, some correctable. In PE mode, there is one parity tape track, so one track can be corrected. GCR mode includes parity and an error-correction character. With the error-correction character, the correction algorithms can correct up to two tracks (in GCR mode).

In PE mode, the logic can detect an error if only one bit in a tape character is bad. The error is detected by the bad parity area in the recovered (read) data. If a pointer indicates a suspect track with the parity bit, then the parity can be corrected by changing that bit. This is just what the PE algorithm does. In addition, it flags the cases it "knows" it can not correct (for example, two equal-weight pointers). The algorithm has no way of knowing which one to correct. On a PE write operation, cases where error correction is performed are also flagged to guarantee good records by rewriting bad records. The arithmetic logic performs all algorithm calculations to determine what needs correcting.

In GCR mode, single tracks can be corrected without a pointer; or, two tracks in any data group can be corrected if pointers are provided to point to suspect tracks. The arithmetic circuitry performs all algorithm calculations for this mode also.

Figure 2-40 M8951 ECC Module Simplified Block Diagram



Data (D0—7, DP) enters the reversing order data multiplexer from the read channel modules. The multiplexer reverses the order of the tracks, depending on the direction (forward or reverse) of the operation. Then it passes the data to a vertical parity checker and an ECC checker multipurpose shift register (MPSR).

The input data is XORed with the MPSR outputs. They form a machine that performs calculations on the incoming data to help locate errors.

An error term (mathematically, a byte of information that relates to the form of the specific error) remains in the MPSR after a GCR (forward) data group is read in. The term can help determine which track to correct in single-track error-correction mode.

When data is clocked through the ECC checker and the MPSR, the parity is checked on the incoming data and placed serially into the shift register. Also at this time, data is brought into a dual 8-bit shift register. After the TIE logic receives this information, it references POINTER STATUS B to determine whether two-track or single-track error correction is required. If the status signal is high, two-track error correction is required; if low, single-track error correction is required. All zeros remain in the MPSR (zero error term) if no error correction is necessary. This indicates that no error exists (probably) in the data. However, an error could still exist on the parity channel, since parity does not pass through the ECC checker.

The MPSR contents are brought into a summing multiplexer controlled by the ROM controller. Any word from the MPSR can be compared with the summing register contents. By changing the select lines, the temporary storage register contents can be compared with the summing register contents. If the two compared registers have equal values, then SUMIN=SUMOUT asserts and the ROM controller can look at the level as one of the possible branching conditions.

Any parity errors brought in are now sitting in the shift register; if the first data byte brought in had a parity error, then the LSB in the shift register has a one in it. The shift register contents are gated through a temporary storage multiplexer to a temporary storage register. From there, the contents pass to the summing register, where comparisons can be made between the MPSR contents and the parity just brought into the summing register.

When the ROM controller asserts all zeros high, the lookup table ROM presents all zeros to the MPSR inputs; then the MPSR can be clocked with the error term in it. And, comparisons can be made between the MPSR (after a certain number of clocks) and the summing register contents, to determine which single track is in error. When a match between the MPSR and the summing register contents exists, the ROM controller "knows" how many clocks had to occur to get that match. By knowing the number, the track in error can be located.

The ROM controller searches all possible combinations for a match to represent each track's error. If it finds the match at the end, then either two-track error correction was needed or more than two tracks were in error. In either case, an uncorrectable error is detected at that point; not enough information is available to correct the data.

Once a match exists between the MPSR and the contents of the summing register that has the parity error in it, the TIE is identified and the corresponding read channel module selected. That module's flag (flip-flop) is clocked, if it does not have a pointer of its own. If the pointer given to the TIE logic is not equal to the pointer determined by this process, then the pointer must match this flag.

If the TIE logic detects two pointers, then a two-track error correction is attempted. Again, if three pointers are found, correction is not possible.

In two-track error correction mode, the pointer information determines which bits must be inverted in the two tracks selected to produce a correct data group output. The MPSR is first loaded with a starting number, based on the J-pointer in the J-pointer buffer. Matrix multiplication is performed by taking the MSB of the shift register that has the parity errors in it. The shift register MSB controls whether or not the summer multiplexer will allow the data from the MPSR to be added to the summer register contents: if a one is in the shift register MSB, then the MPSR contents are added to the summer register contents. If a zero is there, then all zeros are presented to the summer register and its contents remain unchanged.

Matrix multiplication is again performed, this time based on the difference between the I- and J-pointers. Based on this difference, a new value is read from the lookup table and loaded into the MPSR. Multiplication is performed using the same technique of the shift register contents, allowing the MPSR contents to be added to the summer register.

After completing the calculations that determine which bits need to be inverted in the two bad tracks, the contents for the J-track are in the summer register for error correction and the contents for the I-track are in the temporary storage register. When the succeeding data group is brought into the ECC module, the contents of the summer register and the temporary storage register shift out serially so that the two tracks are inverted when necessary.

When a one exists in any position, the data selected by the I- and J-track selectors is inverted and corrected.

Pointer Selector — The TIE error logic accepts the 4-bit TIE bus codes from the read channels and calculates which tracks have the two worst pointers. A decoder run by the ECC module's microcontroller enables the desired TIE bus data (ENTIE BUS D—7, P) to the read channel modules. This enables the 4-bit codes (EC 0—3) onto the TIE bus and to the ECC logic, one at a time. The ECC module's ROM controller orders the TIE bus information from the read channel modules; this gives the edge tracks priority of saving their pointers over the inner tracks. Edge track information is brought in prior to inner track information, because equal-magnitude errors do not displace the errors already in the TIE registers. And edge tracks encounter more errors than inner tracks, therefore the priority.

The codes are latched and then compared in the worst pointer comparator; this determines if the present track (code) is worse than the previous worst track being held in the worst pointer register. If so, the present track replaces the one in the worst pointer register, and this register's contents (previously the worst) go into the second worst pointer register. After all nine read channels are strobed (starting with cleared worst and second worst registers), those registers hold pointers that tell just how bad the data really is in the worst two tracks. For example, if the MSB (EC3 — error code 3) sets, it may represent an illegal 5-to-4 conversion in GCR mode and a permanent dead track in PE mode. The succeeding error code bits represent AMTIE, PHTIE, and a bit generated by the ECC module when the error-correction algorithm performs a single-track error-correction routine. In that routine, it can determine which track was in error, even without pointers. So the LSB error code asserts after a data error without a pointer is corrected.

Worst and second worst pointer registers are both 8-bit registers. Each register holds dual information; four bits tell how bad the track was, and four bits identify the bad track. So, the microcontroller can tell which two tracks are the worst and how bad they are.

While this is happening, pointer counters perform calculations used to determine which microprogram routine to go through. For example, even though two pointers may exist, three or more definite hard errors means the data cannot be corrected, even in GCR.

Four flip-flops make up an error register that holds the status of single-track error correction (STEC), two-track error correction (TTEC), uncorrectable errors (UNC), or pointer mismatch (PNTR MISMATCH). STEC and TTEC state whether single-track or two-track error correction was used. The microcomputer uses this information to decide if the record should be rewritten for a write operation. In a read operation, the record is reread if UNC sets. TTEC causes a retry on a write operation; UNC causes a retry to both read and write. PNTR MISMATCH means a single-track error-correction routine was run, but the single-track error program determined that a different track than the one indicated was in error, causing a record rewrite.

Status drivers direct status words through the read path module and on to the microcomputer. Information such as single-track error correction and two-track error correction is placed on the status bus and sent to the read path logic, where it is rebuffered and sent to the microcomputer. Two status words are available. One comprises single track, two track, uncorrectable, pointer mismatch, no ACRC error, ROM parity error, no CRC error, and AMTIE occurred in the register. The other word is corrected data (CD); this word is most useful during diagnostic testing.

The module can be observed during the testing to ensure it is functioning properly. This is done by loading data that needs correcting into the FIFOs on the read channel modules and single-stepping through the entire read path.

Two pointer buffers store the worst and second worst track numbers related to the worst and second worst pointers. The J-pointer buffer holds the lower track number (ECC-weighted order) and the I-pointer buffer holds the higher track number (ECC-weighted order). The buffer outputs are inputs to a matrix start lookup table ROM. This ROM provides a number used in the calculations to determine which bits in those tracks need correcting.

The data is stored in the data group storage logic, and the actual correction is performed there. When data shifts into this module, many things happen to it. As mentioned, the data is placed in the ECC checker, a multipurpose shift register. The data also is placed into the data group shift registers, which comprise dual 8-bit shift registers. So an entire data group shifts into these registers, as well as going other places. Calculations are performed on the data as it is shifted in. Simultaneously, the TIE logic calculates the two worst tracks.

Once a data group loads into the dual data group shift registers, calculations on the data are performed in the arithmetic section of the module; this determines which bits (on one or two tracks), if any, need correcting. That information feeds the I- and J-track selectors, whose outputs are ORed. These outputs are XORed with the data from the data group shift registers. The XORs can change the data polarity. The I- and J-track selectors guide the error bits from the temporary storage register and the summer register to the tracks indicated. Then the correction XORs invert any bit errors in those tracks.

2.5.3 M8952 Cyclic Redundancy Checker (CRC)

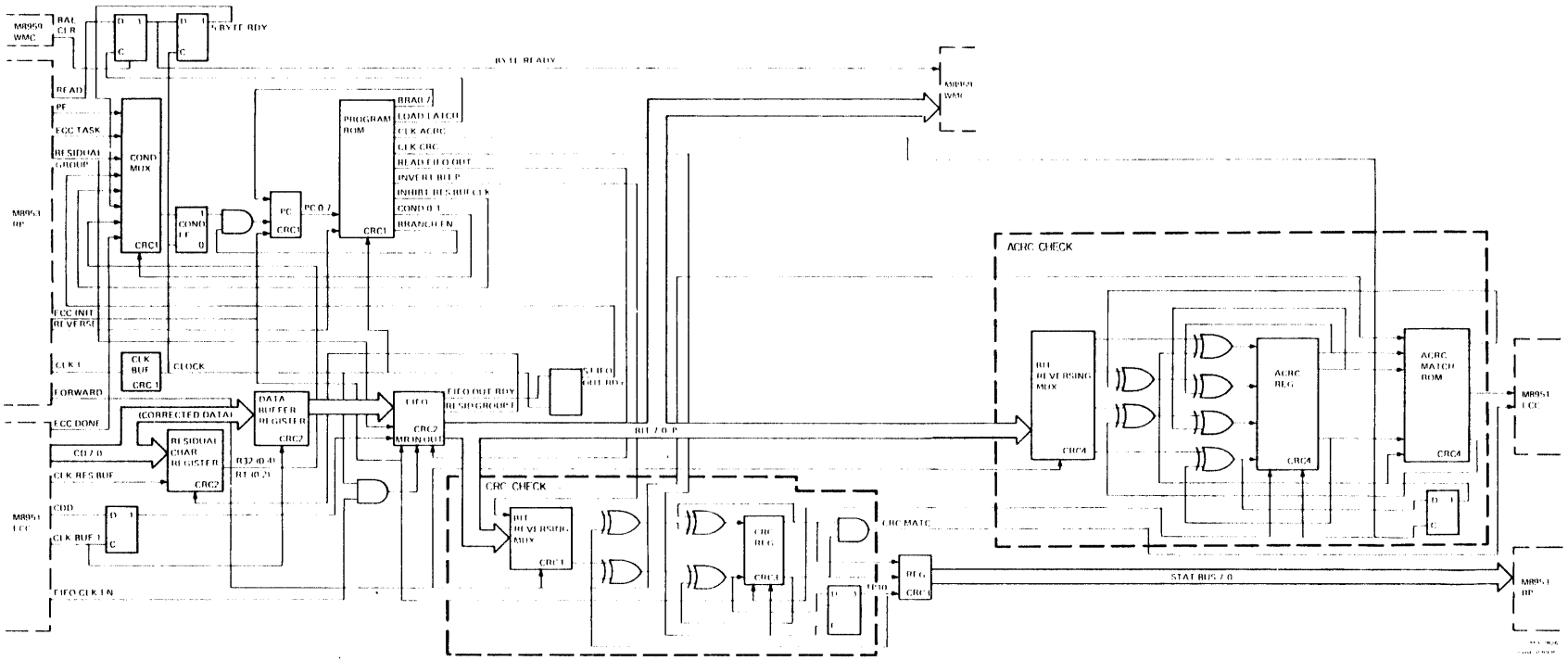
The M8952 is the cyclic redundancy checker (CRC) and auxiliary cyclic redundancy checker (ACRC) module. The M8952 is initialized by the read path controller (M8953) and thereafter controlled by the ECC (M8951) logic. The module contains a ROM controller that performs check character algorithms on both GCR and PE data in both read and write modes. Basically, this module separates the data from all other record characters. It determines how many data characters to send to the byte assembly logic (M8959). It also ensures that the error correction on a record was performed properly, by checking for matching checksum characters.

A GCR record contains CRC and ACRC characters with the data. The M8952 module computes the CRC and ACRC from the data read; then it compares these computed characters with those read. Two logic signals, CRC MATCH and ACRC MATCH, report the results of the compares. The microcomputer uses the two signals to verify the data read from the record in both read and write operations.

A PE record does not contain either of these check characters, so the M8952 cannot determine whether or not the data read is correct. Only parity checking is used when a PE record is read. However, the M8952 does compute a CRC character for all the data bytes of a PE record, and the microcomputer compares this character with the CRC character computed by the write microcontroller (WMC) module (M8959) during a write operation.

Figure 2-41 is the M8952 simplified block diagram. Refer to it while reading the following paragraphs.

Figure 2-41 M8952 CRC Module Simplified Block Diagram



ROM Controller — A ROM controller similar to the read channel, read path, and error-correction code controllers provides the programmable section of the M8952. The program counter (PC) outputs drive the ROM address lines. The most significant eight ROM output bits comprise the branch address (BRA) lines, providing an address for a branch instruction. Additional ROM outputs input to the condition multiplexer select lines. The 16-input condition multiplexer guides the desired condition to the condition flip-flop. The multiplexer inputs are selected by decoding COND 0—4. The BRANCH EN line is gated with the condition flip-flop output, which holds whether or not the selected condition was true. If the condition was true, the PC is loaded from the ROM BRA outputs on the succeeding CLK pulse.

The 512-word ROM requires nine bits to address. The PC is eight bits in length; the ninth bit is the REVERSE signal.

Data Flow — An ECC INIT pulse is generated by the read path logic prior to a read operation. It initializes both the ECC and CRC/ACRC modules, clearing the FIFOs, CRC/ACRC checkers, and PC. The FIFOs may have some useless data in them from a prior operation. However, the FIFO OUT RDY lines (not asserted) claim the FIFOs are empty at this time. So, the FIFO OUT RDY lines do not assert until new data is clocked in.

Once the ECC module receives DATA RDY from the read path controller, it begins clocking data into the CRC/ACRC module in 8-character groups, nine bits per character. For each DATA RDY pulse the ECC logic receives, eight data characters are clocked into the ECC (from the read channel), and corrected data is clocked into the CRC/ACRC (from the ECC). At the first DATA RDY of a record, the CRC logic throws away the useless eight characters it received from the ECC logic; they are not part of the record, so they are just shifted out. After the next DATA RDY pulse sends data in, the FIFO OUT RDY asserts to indicate valid data has been clocked into the FIFO. (From now on, all FIFOs are treated as one FIFO.)

The ECC has complete control over clocking corrected data bytes into the CRC module. CLK BUF 1 clocks the corrected data (CD 0—7) byte into a buffer register. From there, the bytes go into the FIFO.

The residual data group contains important information at the end of a GCR record. If six or less data bytes remain on a data record, they are contained in this data group. The seventh byte is the ACRC character (checks for data validity) and the eighth byte is the normal ECC character. If less than six data bytes remain, pad characters are required to fill the necessary six bytes. The ACRC character and the ECC character are always next, completing the 8-byte GCR format of that residual group.

The residual character serves as a data counter. Bits 0—2 of the byte serve as a modulo-7 counter, indicating how many of the residual data group bytes are data. When the record is read, only the data bytes are retrieved; all pad characters are dropped.

When tape is moving forward, the residual character is the seventh character of the CRC data group; in reverse, it is the second character. Therefore, every seventh (or second, in reverse) character of each group is potentially the residual character.

The ECC logic, therefore, provides clocks on every second (in reverse) and seventh (in forward) data byte sent into the CRC/ACRC. The ROM controller produces INHIBIT RES BUF CLK to control when the residual character register should be inhibited from being clocked. In forward, the character does not need to be inhibited because it is contained in the last group clocked in. In reverse, however, further clocking of the residual character register is inhibited after the first valid group enters the CRC logic; this is because the ROM controller realizes that the residual character is already in the register. The residual character register holds the residual character when tape is moving forward. In forward, no residual character is in the residual character register at the residual group time. Therefore, the FIFOs must store two full groups before any processing can occur. Then, once the read path controller raises the residual group flag, residual group information is clocked into the FIFOs from the ECC module.

The CRC ROM sequencer now waits for the CRC group to be brought into the FIFOs. After this, the residual character is in the residual character register.

Read data does not pass through the byte assembly logic during a write operation. However, the CRC logic must still process data in order to check the data in a read-after-write operation. So, a handshake is "faked." READ H is asserted at the data input of the BYTE RDY flip-flop. So, each time the ROM controller asserts LOAD LATCH, the CRC logic views the result of BYTE RDY; it appears that data was taken. During a normal read operation, the ROM controller loads the latch and sets the S BYTE RDY flag (by loading the latch). Then the byte assembly logic takes the data and asserts BAL CLR to the CRC logic, clearing the flag. This handshake indicates the byte assembly logic took the data.

The read path raises the residual group flag before a residual group is clocked into the FIFO. The flag is clocked into the FIFO with the residual group; the ROM controller looks at the flag as it leaves the FIFO to determine if the data is a residual group or not. If it is a residual group, the controller waits for the next data group to come in instead of processing it as a normal group right away. The last data group to be clocked in is associated with another read path flag, TASK, which indicates this is the last data group to be processed. The read path also notifies the ECC at this time that no correction is necessary on this last group.

Corrected data shifts into the CRC logic from the ECC. The CRC checks the TASK line to make sure it has received all the data before processing the residual group. The data part of the CRC group enters the FIFO almost immediately after TASK asserts (with DATA RDY).

The ROM controller knows it has part of the CRC group in the FIFO, along with the correct residual character in the residual character register. It can now check the residual character register outputs to determine what to do about the residual group. It sees the modulo-7 count (R7, 0—2) of the residual character, which tells how many characters are data and how many are pads. The data characters are then clocked into the CRC checker and ACRC checker. After every data byte, the BYTE RDY flag is set so the byte assembly logic can take the data.

Pad characters are processed just as normal data characters, except they do not enter the ACRC checker. Finally, the ACRC character arrives. It passes to both the CRC and ACRC checkers.

The ACRC checker has no control over how many items are clocked into it. Therefore, if an even number of items is clocked into it, some manipulation must be performed to generate even parity on the ECC character. The residual character register outputs (R32, 0—4) supply the necessary information. This modulo-32 count position of the residual character represents how many items were clocked into the ACRC checker, because only data and the ACRC character are clocked into it. The ROM controller can then determine if an even number of items entered. If the number is even, the ACRC character parity bit going into the ACRC checker must be inverted; INVERT BIT P asserts and inverts the bit.

If this happens, then another clocking of the ACRC character is required to go to the CRC checker without inverting the parity bit. However, if the ACRC character's parity bit does not need to be inverted (meaning an odd number of items were clocked into the ACRC checker), then both the CRC and ACRC checkers simultaneously clock the ACRC character into both checkers, just as with normal data.

After the ACRC character is clocked into both checkers, the ACRC and ECC characters shift out of the FIFO. The ROM controller then determines if a CRC pad is present by checking the modulo-7 and modulo-32 counts, which are still present in the residual character register. The main reason this data is sent to both checkers is to provide reasonable assurance that data was read correctly.

All remaining characters in the CRC group are CRC characters, the residual character and the ECC character. The CRC checker takes the first CRC character available; the remaining information is dumped from the FIFO because the operation is done. At the end of this forward GCR operation, the CRC and ACRC checkers should have matching numbers in their registers.

The first data that enters the CRC logic in a GCR reverse read operation is a CRC group in the reverse order. The ECC module then clocks the residual character register for every second frame of each data group that passes from the ECC to the CRC. The second group (the first valid group) to enter the CRC logic places a valid residual character in the residual character register. Next the ROM controller (CRC) asserts INHIBIT RES BUF CLK for the remainder of the record, because it knows that the residual character is in the residual character register at this time. This holds the residual character in the register for the remainder of the record.

In reverse, the ROM controller does not have to wait for the residual group in order to process the CRC group, because it has the residual character. So it knows what to do with the CRC group. The ECC character and residual character shift out of the FIFO, and the next character, which is a CRC character, clocks into the CRC checker. Then the four remaining CRC characters shift out of the FIFO, and the ROM sequencer checks the modulo-7 and modulo-32 counts to determine if a CRC pad is present. If a pad is present, it is clocked into the CRC checker. If one is not present, the character shifts out of the FIFO; it is not clocked into the checker. The ROM sequencer then waits for the next data group (residual) to be clocked in. The controller again determines if a CRC pad is needed. If one is needed, then data is clocked into the CRC checker, based on modulo-7 and modulo-32 counts. Then the controller dumps the remaining CRC characters out of the FIFO and waits for the next group to be clocked in; this will be the residual group.

When the residual group is received, the ECC character is again discarded. The ACRC character is also clocked into both the CRC and ACRC checkers, again inverting the parity bit if necessary. Also, the pads and data are processed just as in the forward direction. However, with tape moving in reverse, the pads are clocked into the CRC checker before the data.

Once the residual group is handled, the remainder of the record is normal data groups. The ECC character is discarded and the data is processed, seven data characters in each group.

The microcontroller knows the operation is complete when the read path asserts RESIDUAL GROUP. When this is asserted in the reverse direction, its meaning changes: the last valid data group is going into the FIFO. The logic processes the data groups until it sees the residual group flag; this indicates the operation is done. Now, both the CRC and ACRC match circuits should match.

Multiplexers at both checkers reverse the order of the data bits as the data goes from the FIFO to the checkers. The checkers can now see reverse data in the right order.

In PE mode, data is processed one byte at a time, just as though it were a GCR data group. The ECC logic takes one byte of data along with seven bytes of nonintelligent (do not care) data. The valid data byte is the byte corrected on the previous cycle. It is brought into the CRC logic along with the seven other bytes, making an 8-byte group. The first byte (valid) is clocked into the CRC and ACRC checkers; the S BYTE RDY flag asserts; and the data passes to the byte assembly logic. The remaining seven bytes are discarded; they were used only to form an 8-byte data group. The operation is the same for both PE forward and reverse. The residual group flag again asserts from the read path when the last valid data byte is brought into the CRC FIFO. Although the valid byte of data brought in is not a data group, it is treated as such; the flag is used to signify the last valid data byte in the record.

Checkers — The CRC checker is a shifting/dividing network, comprising a register that loops its outputs back through a series of XOR gates. As each data byte enters the checker, it shifts through the logic. A number is present at the output of the latch at the end of each record. The write microcontroller generates the same number during a PE write operation. Corrections are performed in the ECC logic, and the ACRC and CRC checkers ensure (by way of a match) that any corrections are correctly performed. The CRC checksum can then be placed on the status bus (STAT BUS 0—7).

The two checkers work together to provide a very tight check. Instead of the register and gates used in the CRC checker, the ACRC checker uses a ROM. The ACRC checker serves two purposes.

1. Checks the number at the end of an operation for a match. Different numbers are used for forward and reverse, so a direction signal line to the ROM is used.
2. Provides a substitute for some of the feedback gating into the checker.

Both the CRC and ACRC MATCH signals (inverted) are sent back to the ECC logic to form part of the ECC status word for the microcomputer.

2.5.4 M8953 Read Path Controller Module (RP)

The M8953 read path controller is the read logic supervisor, controlled by the microcomputer (M8960). In addition to controlling the read channel modules and microcontrol portion of both the ECC and CRC modules, the read path controller serves as a centralized status reporting area for all the read logic.

The read path controller has a microcontroller branch multiplexer and a branch condition-holding flip-flop, similar to the read channel, error correcting code, and CRC/ACRC modules. A general-purpose 10 MHz counter/timer counts a loadable count that can time a programmed interval. This may be used in controlling a record. For example, the counter can determine when 40 microseconds of record activity has elapsed and notify the microcontroller so it can determine if a true record was read.

In addition to the normal condition multiplexer inputs, a ROM also feeds the multiplexer with information such as the PE ID and GCR ID patterns of the AMTIE lines, tape mark pattern for read/write, seven or more ones, or seven or more zeros. This allows a microprogram to make decisions about the information; for example, it can decide if a majority of the read channel modules have active data, "think" they see a gap, or that the pattern seen is the same as a GCR ID burst.

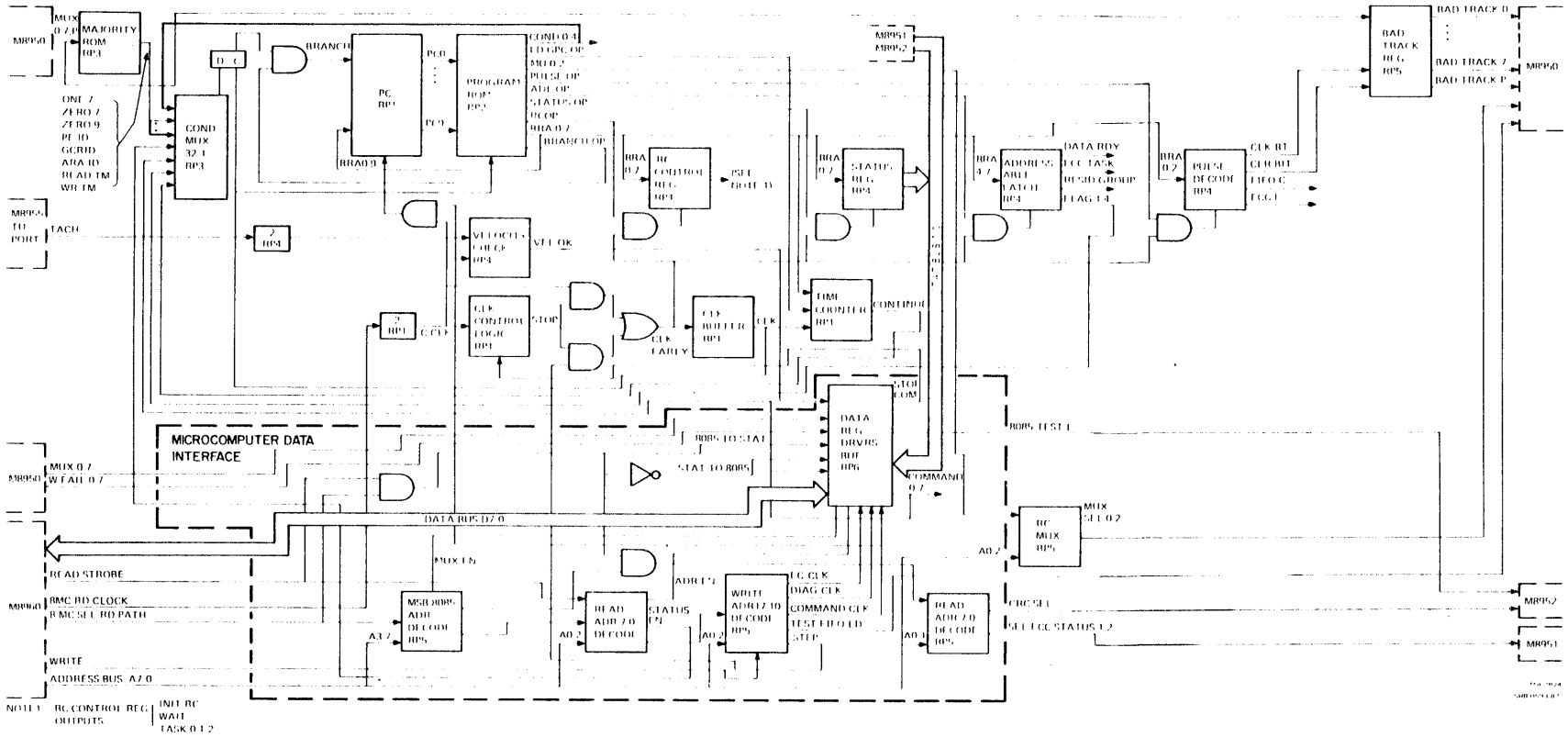
ROM controller outputs drive the read channel control, microcontroller status, addressable latch, and pulse decode registers. A velocity checker compares the 10 MHz read path clock with the TACH signal from the TU port module (M8955).

The remainder of the read path controller is basically microbus interfacing logic. An address decoder can either enable read information onto the data bus (status) or select information from the data bus (command). The 9-bit bad track register holds a one in any position relative to the read channel modules where a particular module is known to be faulty or lost. The following registers that load from the microcomputer: the command register, TIE bus register (used during diagnostics), and a general diagnostic control register (used by the microcomputer as well). The command register can loop back so that a diagnostic can check communications out to the read path and back in again.

Figure 2-42 is the M8953 simplified block diagram; refer to it while reading this section.

Timing — A 20 MHz clock, signal 8MC READ CLOCK, originates on the microcomputer module. A flip-flop divides the signal to a 10 MHz rate, creating C CLK; additional logic generates CLK EARLY. Buffers generate RP1 CLK L, which is the major clock source for the entire read path; it is distributed to the ECC, CRC, and all read channel modules. A unique diagnostic feature is built into the clock control logic. When the microcomputer sends a stop command to the diagnostic register, a flip-flop sets. It disables the 10 MHz clock and enables the STEP signal. Each time the microcomputer commands a write to microbus register C_{16} (014_8), the CLK signal occurs once if the timing logic is stopped. The diagnostics use this feature to stop the clock and step through an entire read operation, comparing the results with what is supposed to be read.

Figure 2-42 M8953 Read Path Controller Simplified Block Diagram



A general-purpose timer/counter counts the read path clock and negates the CONTINUE signal when the counter overflows; this condition is used for branching purposes. (For example, during a normal sequence of instructions, a specific wait time may be desired.) If a wait is desired, the counter is loaded with a specified number corresponding to that time. The next instruction is then to branch to the same location if CONTINUE is still asserted. When the counter times out and overflows, CONTINUE negates and the program falls through to the next instruction.

The counter is both presettable and scaled. It is also an up counter, so the 2's complement of the desired number is used. The counter counts the 10 MHz read path CLK signal. For a wait of 10 microseconds, 100 counts would be required. However, the counter is scaled; its input is actually a divided-by-4. Therefore, a 25 (7753_8) is loaded into it. After 10 ms, CONTINUE negates and the branch falls through to the next instruction.

ROM Controller — A ROM (24 bits by 1024 words) controller, similar to the read channel ROM controller, provides the programmable section of this module. The ten least significant ROM output bits comprise the branch address (BRA) lines. The remaining single-line outputs are op codes that enable certain instructions to execute. Except for BRANCH OP and LD GPC OP (load the general-purpose counter), these op code outputs are gated with CLK EARLY in order to clock a register, gate information onto the status bus, set a bit in the addressable latch, or pulse a backplane pin for scope synchronization purposes during troubleshooting.

A 5-bit (COND 0—4) condition code output selects 1 of 32 different conditions for the next instruction. This microcontroller uses look-ahead architecture, where a condition for a branch instruction is specified in the previous instruction; this allows the ROM to look up the instruction before it is actually used.

The three MU outputs (MU 0—2) go through the read channel multiplexer switch and generate the MUX SEL 0—2 signals. These input to the 8-way read path multiplexer on the read channel module, which selects AMTIE, END, POSTAMBLE, DATA, DONE, MARK 2, and ILLEGAL. The MUX SEL 0—2 lines select which read channel multiplexer input is sent back over the MUX (N) lines to the read path controller. The position of the MU 0—2 lines dictates they become part of the total condition (COND) code. For example, if the condition code is observing something generated from the AMTIE lines, the MU lines select AMTIE while the COND lines state what AMTIE function is selected (for example, seven or more ones on the AMTIE lines).

A 32-way multiplexer, controlled by the 5-bit condition (COND 0—4) code, serves as the condition multiplexer. Everything that this module's microcontroller can branch on comes from the condition multiplexer. Inputs include the COMMAND lines which, when decoded, tell the read path controller to perform (for example, a data operation, a self-test, or an ECC module self-test). If a data operation is commanded, the REVERSE, GCR, and WRITE inputs (from the write microcontroller, M8959) state the operation type.

The MUX 0—7. P lines from the read channel address a majority ROM to aid in condition branching. If those nine lines go directly into the condition multiplexer, it is not necessary to determine which read channel modules have active AMTIE lines and which do not: all that matters is knowing that a majority of the modules have active data, a majority see a gap, or all recognize the same pattern on the AMTIE lines expected from a PE or GCR ID burst. Therefore, a ROM is used to look up this information and pass it on to the condition multiplexer. The read path controller does not know what the read channel modules see on the AMTIE lines or which read channel modules are done: it only knows if a majority has ones or zeros. Certain signals from the majority ROM imply the majority condition: for example, ONE 7 means seven or more ones, while ZERO 7 means seven or more zeros. The remaining outputs assert when a named pattern is recognized (for example, PE ID, WRTM).

The condition multiplexer output becomes the data input of the condition flip-flop, which holds the condition selected on the multiplexer control lines during one instruction. At the end of that instruction period, the flip-flop is clocked (by the read path 10 MHz CLK) and is not clocked again until the end of the next instruction. The condition for the instruction is held throughout the entire instruction period. The condition flip-flop output is gated with BRANCH OP, which was asserted if a branch was commanded. BRANCH would assert and load the BRA bits into the PC.

At the end of an instruction, the specified and selected condition code is recorded in the condition flip-flop. Also, the end of this instruction signals the start of the branch instruction, when BRANCH asserts. Therefore, each branch instruction involves two instructions, one that selects the condition multiplexer position and one that contains the branch operation.

The RC control register, status register, addressable latch, and pulse decoder are all addressed by the ROM BRA lines. Each logical section is selected by its respective op code from the ROM and the CLK EARLY signal. The RC control register controls all nine read channel modules. Its outputs include signal IRC which, when buffered, becomes the read channel initialization (INIT) signal that clears the read channel PC.

The RC control register also generates WAIT, TASK 0, and TASK 2, which are inputs to the read channel condition multiplexer. A status register, as its name implies, places its outputs onto the status bus and the Microbus D lines to the microcomputer. The RC control register is used, for example, in testing the read channel modules after a read operation. The read may have been successful, or some problems may have been encountered. No matter what happened, a status code reflecting the situation is loaded into the status register and then output from the register. The outputs are enabled onto the status bus to the microcomputer.

Like the status register the addressable latch can store status information. However, the status register sends the information to the microcomputer for processing: the addressable latch provides storage for internal use at a later time. An example of this is the detection of an End Mark from most of the read channel modules. Two data group time periods must pass before the residual flag can be raised. The End Mark read is stored by decoding BRA lines 4—7 and asserting signal FLAG 3. The microprogram then "remembers" this during succeeding CRC groups until the correct time to raise the residual flag occurs.

A pulse decoder decodes BRA lines in much the same way as the addressable latch. Its selected output forces desired operations such as initializing the ECC (signal ECC I) module. Another output clears all the addressable latch outputs; this places everything on the read path controller in a known state at the beginning of an operation and ensures DATA RDY is inactive. In addition, the bad track register can be forcibly cleared and clocked (CLR/CLK BT).

A velocity checker guarantees a constant tape speed during all write operations. The TACH signal from the TU port module goes through a divide-by-two flip-flop and enters a counter network. This network also senses the 10 MHz C CLK pulses and counts how many C CLK pulses occur during the 80-microsecond flip-flop period. When the period is at 80 microseconds ± 10 percent, VEL OK asserts. If the 10-percent limits are exceeded, VEL OK is negated and sent to the microcomputer, by means of a status byte. There the operational microcode deals with the problem (for example, signals the host computer for a retry operation).

Microbus Interface — The microbus interface logic provides constant comprehensive read path status to the microcomputer. A read address decoder decodes the three least significant microbus address bits (A2:0), while an MSB address decoder decodes the most significant address bits (A5:3). The MSB address decoder is enabled by signal 8MC SEL RD PATH, a prior decode of the most significant address bits. The enable and most significant bit decode defines whether the operation occurring involves the read path; the least significant bit decode specifies the register. Additional decoders can select specific address space, or "bands" of addresses in which the command is used. Further controls from the microcomputer are the WRITE and READ STROBE signals, specifying a microbus write or read operation.

Each signal enables its respective decoder (write or read address); therefore, not only must the correct address band be selected, but also the correct operation, in order for the decoders to generate the decoded outputs.

The write address decoder outputs various clocking signals, DIAG, EC, and COMMAND CLK. These signals strobe data into those specified registers that the microcomputer can write into (for example, diagnostic command). TEST FIFO LD and STEP specify operations that do not require any data. In one of the test modes, for example, the read channel FIFO is manually loaded (with test information) by the microcomputer writing to the decoded address. TEST FIFO LD is generated and passed to the read channel, where it loads the FIFO under test conditions. STEP is a single-step clock that inputs to the clock control logic. The READ ADR 30—32 decoder is activated during a microbus read operation only; it enables either the CRC status word or one of the two ECC status registers that can be read by the microcomputer.

The data part of the microbus interface consists of command registers that can be loaded by the microcomputer, and status drivers that pass information to the microcomputer. Signals D0—D7, the microbus data bus, are the main inputs to the interface. The status bus (STAT BUS 0—7), internal to the read path, is buffered so as not to load down the data bus and to provide strong read path drive signals. STAT TO 8085 and 8085 TO STAT define the direction of transfer. If no transfer is taking place, 8085 TO STAT permits the status bus to monitor microcomputer action, but it does not become involved.

Four tri-state chips comprise the status bus drivers: each is enabled by its own address decoder. The command loop driver operates during testing only, making sure the command was received correctly. Anything that can be placed on the read channel MUX lines (for example, AMTIE signals) can be read into the read channel multiplexer driver and observed by the microcomputer, again only during testing. The remaining two drivers enable the microcomputer to monitor the status of the WFAIL lines and other important conditions: STOP, DATA RDY, PLO RUN, and VEL OK. Three registers are used to take information from the status bus. Each register is clocked by its respective address decoder signal: COMMAND CLK, EC CLK, and DIAG CLK.

For the microcomputer to write into the command register, the register address is first placed on the address lines (A7:0). Then WRITE and 8MC SEL RD PATH become active during the address decode, generating COMMAND CLK. When the data lines are valid (during the WRITE time), 8085 TO STAT asserts, passing the data lines through the buffers and placing the information at the command register inputs. COMMAND CLK catches the data pattern in the command register. To enable the microcomputer to read the command register, a similar address decode begins again; this time the READ STROBE is active instead of WRITE. STAT TO 8085 activates the command driver enabled by COMMAND EN. The COMMAND lines are placed on the status bus and buffered to the microbus data lines; the microcomputer then takes in the data.

The TIE bus register is a diagnostic register used to force the TIE bus to a particular state during testing. A diagnostic register is used to diagnose TIE bus problems. The TIE bus connects to so many modules (all read channels, ECC, RP) that it has diagnostics that ensure its operation. Data from the microcomputer is clocked into the TIE register by signal EC CLK. The diagnostic register asserts 8085 TEST, which places the TIE bus register contents onto the status bus so it can be read by the microcomputer. This enables the entire read path to be checked, even though the read channel modules are effectively disconnected from the read path.

The bad track register holds a one in the corresponding track position that indicates one of two things: either the particular read channel is not working correctly and the data is bad, or the particular module is lost and its data is not worth considering.

This register is used, typically, at the beginning of a read operation when the read channels are searching for the data preamble. When the first seven channels are done, the DONE lines activate. The read path delays for approximately 10 to 15 microseconds and then strobes the DONE lines [as MUX (N) lines from the read channels] into the bad track register. Those tracks that did not detect the end of the preamble record a one in the respective position. This information goes back to the read channel modules and is ORed with the DONE lines to make them appear done. The bad track register bits are also ORed into the ILLEGAL lines to indicate a pointer. This allows the ECC circuitry to correct the data.

2.5.5 M8955 Tape Unit Port (TUP)

The M8955 is the tape unit port (TUP) module. The formatter uses one or two TU port modules. Each TU port module provides two separate logical ports and can interface two TU78 tape transports to the subsystem. Thus, a subsystem with a full complement of four tape transports requires two TU port modules.

The TU port module acts as the data channel between the transport and read/write data paths, and as the command/status channel between the microcomputer and the transport. It also serves as one of the data looparound points for internal diagnostics. Figure 2-43 shows the I/O signals and their source/destination for a single TU port module (two TU ports).

Figure 2-44 is the TU port block diagram. Each block, or gate, of the diagram contains two alphanumeric symbols, (for example, TUP1—E14). The first symbol represents the particular drawing in the engineering print set that shows the logic represented by the block. The second symbol (if applicable) refers to the actual chip number. This block diagram details a single port (one of two on a single TU port module) and is not duplicated, since all ports are functionally identical. The port represented is port (or transport) zero or two. Refer to the block diagram while reading the following paragraphs.

Figure 2-43 TU Port I/O Signals

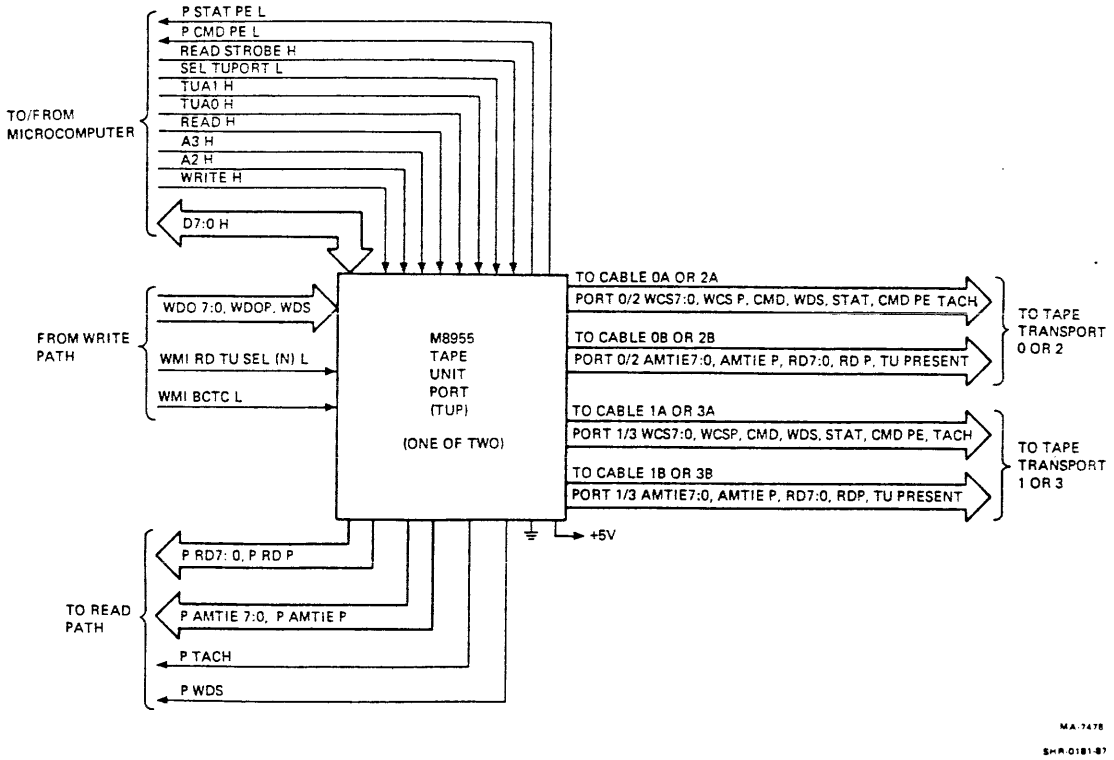
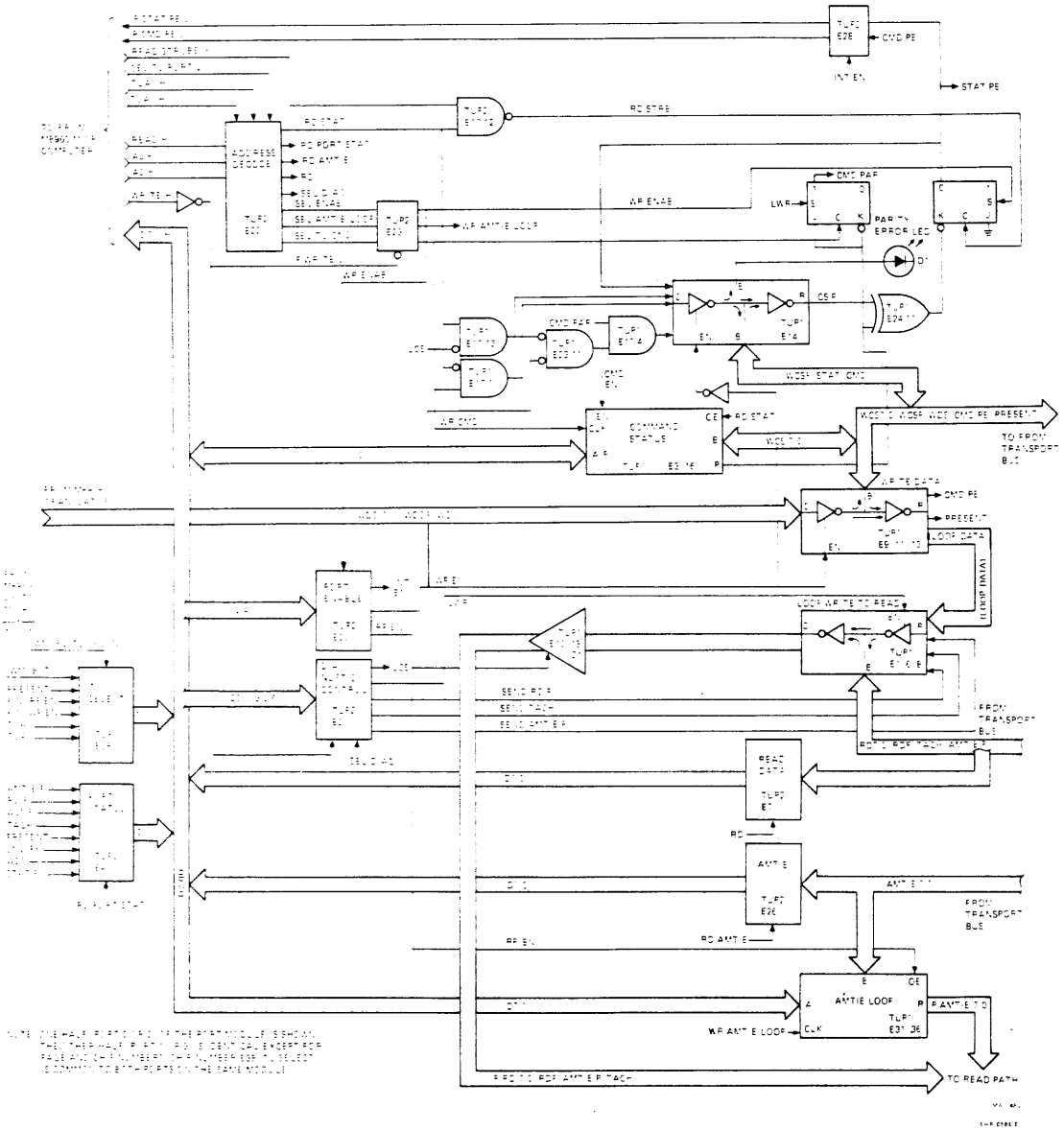


Figure 2-44 M8955 TU Port Block Diagram



Microbus Interface — The microbus connects to each of the four possible TU ports in the same manner. Figure 2-44 (top) shows all the microbus lines needed to transfer the command/status information.

- Port Addressing
  One of four TU ports is selected by microbus lines TU A1 H and TU A0 H. The proper 2-bit binary address value along with SEL TU PORT L enables the address decoder for that port. This 2-bit address field is asserted by the microcomputer and held in an internal microcomputer register. See Appendix E, internal address E0-W (340 W).

- Register Addressing
  Before a transport can perform a function, the port to which the transport is logically connected must be selected. This is achieved by addressing a particular register, through the address decoder, with microbus lines A2 H, A3 H, and READ H. One of eight possible microbus registers may be addressed and read from or written to in this manner. A ninth register (TU SELECT) may be read by the asserted lines WMI RD TU SEL (N) L, regardless of the port select status. The (N) in this signal name refers to a 0 for ports zero and one, and a 1 for ports two and three. These two lines are decoded by the write microcontroller.

  Table 2-12 lists the microbus address, associated register selected, and internal TU port signal decoded. Refer to Appendix B for a description of the individual register bits. Note that the TU port ignores the two least significant microbus address bits. Thus, address 100 could actually be any value from 100 to 103.

- Miscellaneous
  Microbus signal WRITE H and its derivative P WRITE L strobe command data from the D lines into the control and enable registers. Microbus signal READ STROBE H clocks the STAT PE flip-flop during a read tape unit status function. If a parity error occurs during the status transfer, the microcomputer is notified by means of the P STAT PE L microbus signal. The P CMD PE L microbus signal notifies the microcomputer when the tape transport detected a parity error during the transmission of a command over the TU bus.

Table 2-12 TU Port Registers

Microbus Address	Register Selected	Decoded Signal READ H	READ H
40_{16} (100_8*)	TU status byte (TSTS)	RD STAT	—
40_{16} (100_8*)	If bit 7 = 0, TU command	—	SEL TU CMD/WR CMD
	If bit 7 = 1, CMD/STA address	—	SEL TU CMD/WR CMD
44_{16} (104_8)	AMTIE (TAMT)	RD AMTIE	—
44_{16} (104_8)	AMTIE loop (TAMT)	—	SEL AMTIE LOOP/ WR AMTIE LOOP
48_{16} (110_8)	Port status (PSTAT)	RD PORT STAT	—
48_{16} (110_8)	Port diagnostic (PDIAG)	—	SEL DIAG
$4C_{16}$ (114_8)	Read data (PRDD)	RD	—
$4C_{16}$ (114_8)	Port Control (PENAB)	—	SEL ENAB/WR ENAB
$D1_{16}$ (321_8)	(Port 0/1)	TU select 0 (TU SEL 0)	WMI RD TU SEL (0) L
$D2_{16}$ (322_8)	(Port 2/3)	TU select 1 (TU SEL 1)	WMI RD TU SEL (1) L

* Microbus address 100 is actually the command/status byte and is eight registers deep (all located in the MIA module). Appendixes B and D contain descriptions of these command/status byte registers.

Special-Purpose Logic Elements — The TU port logic uses two special-purpose tri-state bus transceivers, the 2907 and 26S10 elements. You may not see, simply by following data through the logic elements on the engineering prints, just how these transceivers process the data. Where possible, Figure 2-44 (TU port block diagram) shows the function of the chip within the confines of the element. Also, Figures 2-45 and 2-46 depict the internal signal routing and gating. Figure 2-45 shows the 2907 element, and Figure 2-46 shows the 26S10 element. Refer to these figures while reading the following paragraphs.

Figure 2-45 2907 Quad Bus Transceiver

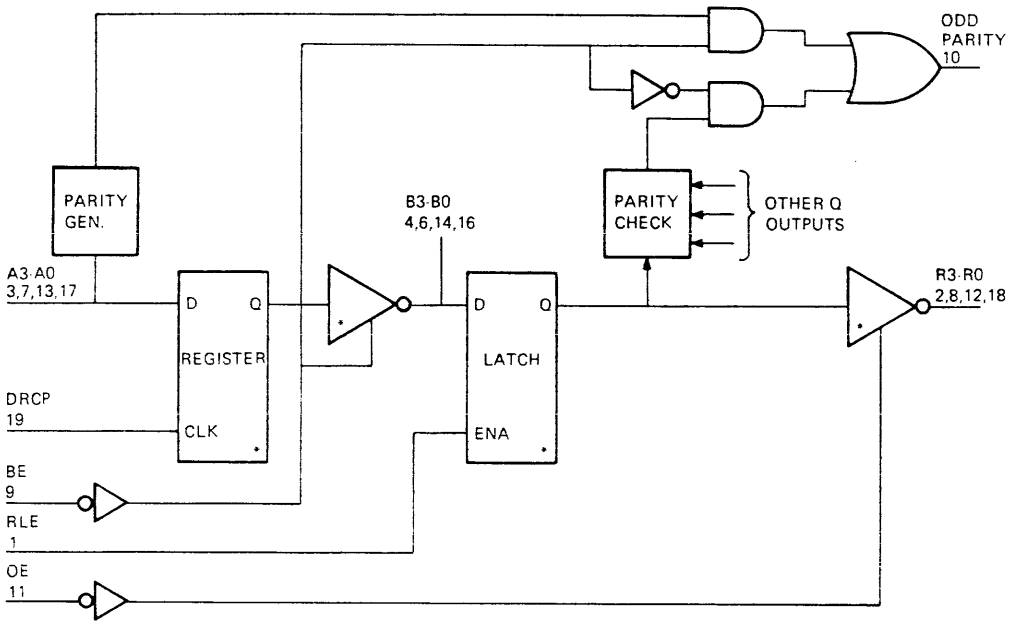
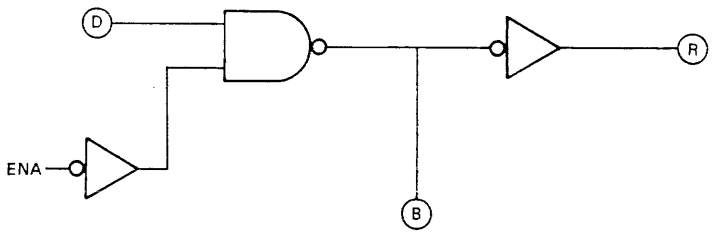


Figure 2-46 26S10 Quad Bus Transceiver



TU Port/Transport Command Processing — The tape unit command/address specifier byte (microbus address 40-W) is sent by the microcomputer over the D lines. This byte is seen at the input to the command/status transceivers (TUP1—E3, E16) and clocked into their internal buffer registers by the rising edge of WR CMD L. CMD EN L (TUP1—E23-11) is low at this time and enables the command byte to the tape transport over the WCS lines.

The command/status transceivers generate a parity bit which is clocked into the CMD PAR flip-flop (TUP1—E19-6) by WR CMD L. Then the latched parity bit is sent out to the transport over the WCS P L line through the bus transceiver (TUP1—E14). AND gate (TUP1—E17-1) through the bus transceiver (TUP1—E14) produces the TU bus signal CMD L; this strobes the command byte into the tape transport logic. The transport logic processes the command immediately after CMD L goes false.

Once the command is latched into the transport logic, a parity check is made. If the transport determines that bad parity was received (due to malfunctioning TU port logic or TU bus cable), it asserts the CMD PE L TU bus line. The CMD PE signal is inverted through the bus transceiver (TUP1—E9-3) and, assuming the port is interrupt enabled, asserts the P CMD PE L microbus line to the microcomputer. A command parity error does not cause the port parity error LED to light.

TU Port/Transport Status Processing — The TU port provides two classes of status to the microcomputer, port status and transport status.

A single byte of port status (one byte from each port) is available through the port status buffer (TUP2—E4). It is accessed by performing a microbus read of internal address 110. Refer to Appendix B for a description of the eight bits of the port status byte. When reading address 48, the signal RD PORT STAT L is produced by the address decoder (TUP2—E22) and strobes the status byte to the microbus D lines.

One of eight possible transport status bytes is accessed by performing a microbus read of internal address 40. Refer to Appendix B for a description of the transport status bytes. When reading address 40, the signal RD STAT L is produced by the address decoder (TUP2—E22). Signal RD STAT L enables the AND gate (TUP1—E17-13), and through the transceiver (TUP1—E14) it asserts the STAT L TU bus line to the tape transport.

Upon receiving STAT L, the transport strobes one of the seven transport status bytes to the WCS lines. The transport-generated parity bit WCS P is also strobed to the TU bus at this time.

RD STAT L also enables the command/status transceiver (TUP1—E3-16). This transceiver gates WCS 7:0 lines to the microbus D lines. The parity bit WCS P is inverted through a transceiver (TUP1—E14) and becomes CS P, which is one of the inputs to an XOR gate (TUP1—E24-11). The other input to this gate is the parity checking bit generated by the command/status transceivers (TUP1—E3, E16). If the transport status byte is received at the port with correct (odd) parity, the output of the XOR gate will be high. If the byte is received with incorrect (even) parity, the output of the XOR gate will be low; this primes the K input to the STAT PE flip-flop

(TUP1—E19-9). Later in the microbus read cycle, the microcomputer asserts the READ STROBE H line, which becomes RD STRB L and clocks the STAT PE flip-flop true. If the port was previously interrupt enabled, it now asserts the P STAT PE L microbus line to the microcomputer. The microcomputer then handles this error as an interrupt and retries the operation.

The output of the STAT PE flip-flop is also routed through the transceiver (TUP1—E14) and causes the port parity error LED to light.

TU Port/Transport Write Data Processing — The TU port provides a channel for write data between the write path logic and the tape transport, but does not influence write data timing, strobing, or parity generation/checking.

Prior to the transmission of write data, the microcomputer must issue certain commands to the tape transport and the TU port. The transport must be instructed to move forward and enable its write/erase logic. The microcomputer does this through internal address 40 as outlined in the TU Port/Transport Command Processing section of Paragraph 2.5.5. Then the port must be enabled (that is, the write data gate must be opened). The microcomputer accomplishes this by issuing a command to the port control byte over internal address 4C-W (114_8-W). Refer to Appendix B for a bit description of the port control byte. When the command is written to address 114, the signal WR ENAB (SEL ENAB) is produced by the address decoder (TUP2—E22). WR ENAB gates the microbus D lines to the port enable register (TUP2—E33). WR ENAB also clears the STAT PE flip-flop if it was set from a previous status parity error condition.

The port enable register (TUP2—E33) produces two signals during a write operation, WP EN and RP EN. WP EN serves three purposes when asserted. First, it enables the write data transceivers (TUP1—E9, E11, E13). Second, it disables the command/status transceivers by negating the CMD EN signal (TUP1—E23-11). Third, it disables the transceiver (TUP1—E14). RPEN enables the read data buffers (TUP1—E10, E18, E21) and the AMTIE transceivers (TUP1—E31, E36) which transmit READ, AMTIE, and TACH information to the read path. During a write operation the read path is enabled to check the read-after-write data integrity.

The translator (M8958) sends write data to the TU port in a 9-bit-wide data path (WDO 7:0 H, WDO P H) along with a strobe pulse (WDS H). The data is inverted in the TU port through the write data transceiver (TUP1—E9, E11, E13) and sent to the tape transport over the WCS lines on the TU bus (WCS 7:0 L, WCS P L, WDS L). The nine data bits are strobed to the tape transport logic by the WDS L (write data strobe) signal.

At this point, write data is in a form that allows direct recording to tape. All PE/GCR formatting and byte assembling has been done by the Massbus data module and write path.

TU Port/Transport Read Data/AMTIE Signal Processing — The TU port provides a channel for read data and AMTIE signals between the tape transport and the read path logic. It does not influence timing, strobing, or data deskewing, and does not perform parity checking.

Prior to the reception of read data/AMTIE signals, the microcomputer must issue certain commands to the tape transport and the TU port. The transport must be instructed to move forward or reverse, to disable its write/erase logic, and to enable its read logic. The microcomputer does this through internal address 100, as outlined previously. Then the port must be enabled (that is, the read data gate must be opened). The microcomputer accomplishes this by issuing a command to the port control byte over internal address 4C-W (114_8-W). Refer to Appendix B for a bit description of the port control byte. When the command is written to address 4C (114_8), the signal WR ENAB (SEL ENAB) is produced by the address decoder (TUP2—E22). WR ENAB gates the microbus D lines to the port enable register (TUP2—E33). WR ENAB also clears the STAT PE flip-flop if it was set from a previous status parity error condition.

The port enable register (TUP2—E33) produces the RP EN signal during a read operation. RP EN enables the read data buffers (TUP1—E10, E19, E21) and the AMTIE loop transceivers (TUP1—E31, E36).

The tape transport sends read data to the TM78 over the TU bus. A 9-bit-wide data path (RD 7:0 L, RDP L) is used. The data is not accompanied by a strobe pulse, since it is deskewed in the read channels. Read data from the TU bus is inverted through the loop write-to-read transceivers (TUP1—E1, E6, E8). The output of the loop write-to-read transceivers is gated through the read data buffers (TUP1—E10, E18, E21) and sent to the read path logic. Two additional TU bus lines are routed (and inverted) through these transceivers and buffers. They are the tachometer pulse (TACH L) and the AMTIE parity bit (AMTIE P L). The tachometer pulse is sent to the read path controller (M8953) and the AMTIE parity bit is sent to the parity read channel.

AMTIE signals are also sent from the tape transport to the formatter over the TU bus. Nine TU bus lines handle this transfer (AMTIE 7:0 L, AMTIE P L). The path for AMTIE P was discussed in the preceding paragraph. AMTIE 7:0 lines are inverted through the AMTIE loop transceivers (TUP1—E31, E36) and sent to the read path logic as P AMTIE 7:0 H.

TU Port/Transport Diagnostic Modes/Data Paths — The following paragraphs outline the internal (nontape-motion) diagnostic modes and data paths implemented within the subsystem through the TU port.

The microcomputer can evoke two classes of internal diagnostic modes through the TU port: those that wrap test data around within the TU port, disallowing TU bus communication; and those that wrap test data around within the tape transport logic, allowing TU bus communication. Both classes involve a special set of commands to the TU port/transport, and some unique internal microbus addresses to transfer the test data to and from the microcomputer.

Listed here are the individual diagnostic wrap modes implemented by the microcomputer through the TU port. These modes are used during the execution of internal on-line diagnostics, and host-system-initiated standalone diagnostics. The list includes the mode type, the path that test data takes (in parentheses), and a brief tabular summary of the events required to execute the data wrap.

TU Port Internal Diagnostic Data Wrap Modes — In each of the modes in this class test data is looped (wrapped) in the TU port. In each mode, certain TU bus lines connected to the TU port being diagnosed are driven. In some cases this is intentional (for example, commanding a tape transport to disable its RD/AMTIE drivers that may be confusing test data being looped in the port). However, in most cases this is simply because the TU bus connects to the TU port transceivers involved in the data wrap.

1. Loop Write Path to Microcomputer
   Data (WDO 7:0 H WCS 7:0 L D 7:0 H)
   Parity (WDO P H WCS P L CS P H)

   - Microcomputer issues WP EN port enable command (internal address 4C-W (114_8-W), bit 4 = 1).
   - Microcomputer starts write path/translator microcontrollers and provides test data to write microcontroller.
   - Translator sends data over WDO lines.
   - Microcomputer issues read status command (internal address 40-R).
   - Microcomputer reads test data over the microbus D lines for checking. The microbus P STAT PE L line is asserted if a parity error occurs.

2. Loop Write Path to Read Path
   Data and Parity (WDO 7:0 H, WDO P H P RD 7:0 H, P RD P H)

   - Microcomputer issues transport read path disable command [internal address 40-W (100_8-W), command address = 3, bit 6 = 0]. Transport will not drive RD/AMTIE TU bus lines.
   - Microcomputer issues WP EN and RP EN port enable commands (internal address 4C-W (114_8-W); bits 4, 2 = 1).
   - Microcomputer issues Loop Write-to-Read (LWR) command to port diagnostic control register [internal address 48-W (110_8-W), bit 5 = 1].
   - Microcomputer starts write path/read path microcontrollers and provides test data to write microcontroller.
   - Translator sends data over WDO lines and is looped to P RD lines by TU port.
   - Test data is routed back to the microcomputer over the microbus D lines by the read path controller for checking. By means of a status byte, the controller notifies the microcomputer if a parity error occurred.

3. Loop Microcomputer to Read Path
Data (D 7:0 H WCS 7:0 L P RD 7:0 H)
Parity (internal address 48-W (110_8-W), bit 1 P RD P H)

- Microcomputer issues transport read path disable command [internal address 40-W (100_8-W), command address = 3, bit 6 = 0]. Transport will not drive RD/AMTIE TU bus lines.
- Microcomputer issues RP EN port enable command [internal address 4C-W (114_8-W), bit 2 = 1].
- Microcomputer issues LWR and LCS commands to port diagnostic control register [internal address 48-W (110_8-W); bits 5, 6 = 1].
- Microcomputer starts read path microcontrollers.
- Microcomputer provides test data byte to TU port by writing into tape unit command byte [internal address 40-W (100_8-W)].

NOTE
Test data is not strobed to tape transport as a command because TU bus line CMD L is not asserted.

- Microcomputer provides test data parity bit by writing into the port diagnostic control register [internal address 48-W (110_8-W), bit 1 = 0/1].
- Test data is routed back to the microcomputer over the microbus D lines by the read path controller for checking. The controller notifies the microcomputer if a parity error occurred, by means of a status byte.
- The microcomputer may also read the test data back from the TU port before the data reaches the P RD lines. [It does this by reading internal address 4C-R (114_8-R)]. In this case the test data path is:

  (D 7:0 H WCS 7:0 L RD 7:0 L D 7:0 H)

  RD parity is not checked through this data path.

4. Loop Microcomputer to AMTIE Lines
Data (D 7:0 H P AMTIE 7:0 H)
Parity (internal address 48-W (110_8-W), bit 0 P AMTIE P H)

- Microcomputer issues transport read path disable command [internal address 40-W (100_8-W), command address = 3, bit 6 = 0]. Transport will not drive RD/AMTIE TU bus lines.
- Microcomputer issues RP EN port enable command [internal address 4C-W (114_8-W), bit 2 = 1].
- Microcomputer starts read path microcontrollers.

- Microcomputer provides test data byte to TU port by writing into the AMTIE loop register [internal address 44-W (104_8-W)].
- Microcomputer provides test data parity bit by writing into the port diagnostic control register [internal address 48-W (114_8-W), bit 0 = 0/1].
- AMTIE test data is routed back to the microcomputer over the microbus D lines by the read path controller for checking. By means of a status byte, the controller notifies the microcomputer if a parity error occurred.
- The microcomputer may also read the test data back from the TU port before it reaches the P AMTIE lines. It does this by reading internal address 44-R (104_8-R). In this case the test data path is:

  (D 7:0 H AMTIE 7:0 L D 7:0 H)
- AMTIE parity is not checked through this data path.

5. Loop Command to Status
   Data (D 7:0 H WCS 7:) L D 7:0 H)
   Parity (generated and checked by logic in TU port)

   - Microcomputer issues loop command-to-status (LCS) command to port diagnostic control register [internal address 48-W (114_8-W), bit 6 = 1].
   - Microcomputer sends test data to TU port by writing to tape unit command byte [internal address 40-W (100_8-W)].

   NOTE
   Test data is latched into the command/status transceivers and appears on the TU bus WCS lines, but is not strobed to the transport logic because TU bus line CMD L is not asserted.

   - Microcomputer reads and checks test data from TU port by reading tape unit status byte (internal address 40-R).
   - The TU port asserts the P STAT PE L line to the microcomputer if a parity error occurs.

TU Port External Diagnostic Data Wrap Modes — In each of the modes in this class, test data is looped (wrapped) around in the tape transport logic. In all cases the TU port drives the TU bus intentionally, with the transport producing the appropriate responses.

1. Loop Write Path to Read Path
   Data and Parity (WDO 7:0 H, WDO P H WCS 7:0 L, WCS P L RD 7:0 L, RD P L P RD 7:) H, P RD P H)

   - Microcomputer issues transport loop write to read command [internal address 40-W (100_8-W), command address = 0, bit 2 = 1].
   - Microcomputer issues WP EN and RP EN port enable commands [internal address 4C-W (114_8-W); bits 4, 2 = 1].
   - Microcomputer starts write path/read path microcontrollers and provides test data to write microcontroller.
   - Translator places data on WDO lines to port, and data is sent to transport over WCS lines. Transport loops data around and sends it back to port over RD lines. Port sends data to read path over P RD lines.
   - Test data is routed back to the microcomputer for checking over the microbus D lines by the read path controller. By means of a status byte, the controller notifies the microcomputer if a parity error occurred.

2. Loop Microcomputer to RD Lines
   Data (D 7:0 H WCS 7:0 L RD 7:0 L D 7:0 H)
   Parity (port-generated parity bit WCS P L RD P L)

   - Microcomputer issues transport loop write-to-read command [internal address 40-W (100_8-W), command address = 0, bit 2 = 1].
   - Microcomputer issues LCS command to port diagnostic control register (inhibits TU bus CMD line). [Internal address is 48-W (110_8-W), bit 6 = 1.]
   - Microcomputer sends test data to tape unit command byte [internal address 40-W (100_8-W)].
   - Port sends test data to transport over TU bus WCS lines. Transport loops data around and returns it to port over TU bus RD lines.
   - Microcomputer requests read data from port and checks it for accuracy [internal address 4C-R (114_8-R)].

3. Loop Microcomputer to AMTIE Lines
Data (D 6:0 H WCS 6:0 L AMTIE 7:0 L D 7:0 H
Parity (port-generated parity bit WCS P L AMTIE P L)

- Microcomputer issues transport loop write-to-read command [internal address 40-W (100_8-W), command address = 0, bit 2 = 1].
- Microcomputer sends partial test data to tape unit command byte [internal address 40-W (100_8-W), command address = 0, bits 1, 0 WCS 1,0 L AMTIE 7, P L].
- Transport loops test data around.
- Microcomputer reads and checks partial test data from port [internal address 44-R (104_8-R), AMTIE 7 L D 7 H].
- Microcomputer sends partial test data to tape unit command byte [internal address 40-W, command address = 2, bits 6:0 WCS 6:0 L AMTIE 6:0 L].
- Transport loops test data around.
- Microcomputer reads and checks partial test data from port [internal address 44-R (104_8-R), AMTIE 6:0 L D 6:0 H].

2.5.6 M8958 Translator (XMC)

The M8958 translator module (XMC) functions in write mode only. In GCR format it acts as a translator, converting 4-bit data subgroups from the write microcontroller module (M8959) into 5-bit storage subgroups for recording on tape. In PE format it does not translate, but rather serves as a multiplexer. The translator module also generates ID bursts, preambles, postambles, control subgroups, and Resync bursts, and controls the rate at which data is clocked out to the tape transport.

The two major functional areas of the translator module are the data translator and the translator microcontroller. Figure 2-47 shows these areas and how they tie into other parts of the formatter.

Data Translator — Figure 2-48 shows a functional view of the data translator. The translator buffers an entire data subgroup comprising eight 4-bit data nibbles plus four vertical parity bits and four possible PE done bit cells, for a total of 40 bits. During translation, the write microcontroller is busy shifting in the next data subgroup.

Data enters the translator from the write microcontroller over the write data lines (DR WD 3:0, DR BYTE PAR H). The DR WD lines carry data; the DR BYTE PAR line is multiplexed, carrying parity and done flag information. DR BYTE PAR transmits a vertical parity bit on the second, fourth, sixth, and eighth nibble transfers. It may transmit a PE mode done flag on the first, third, fifth, and seventh transfers. If the done flag is present on any odd-nibble transfer in PE mode, no further data nibbles are sent. In GCR mode the write microcontroller always sends full data subgroups and groups. Paragraph 2.5.7 details the process of forming a data nibble/subgroup and multiplexing DR BYTE PAR.

Figure 2-47 M8958 Translator Functional Area

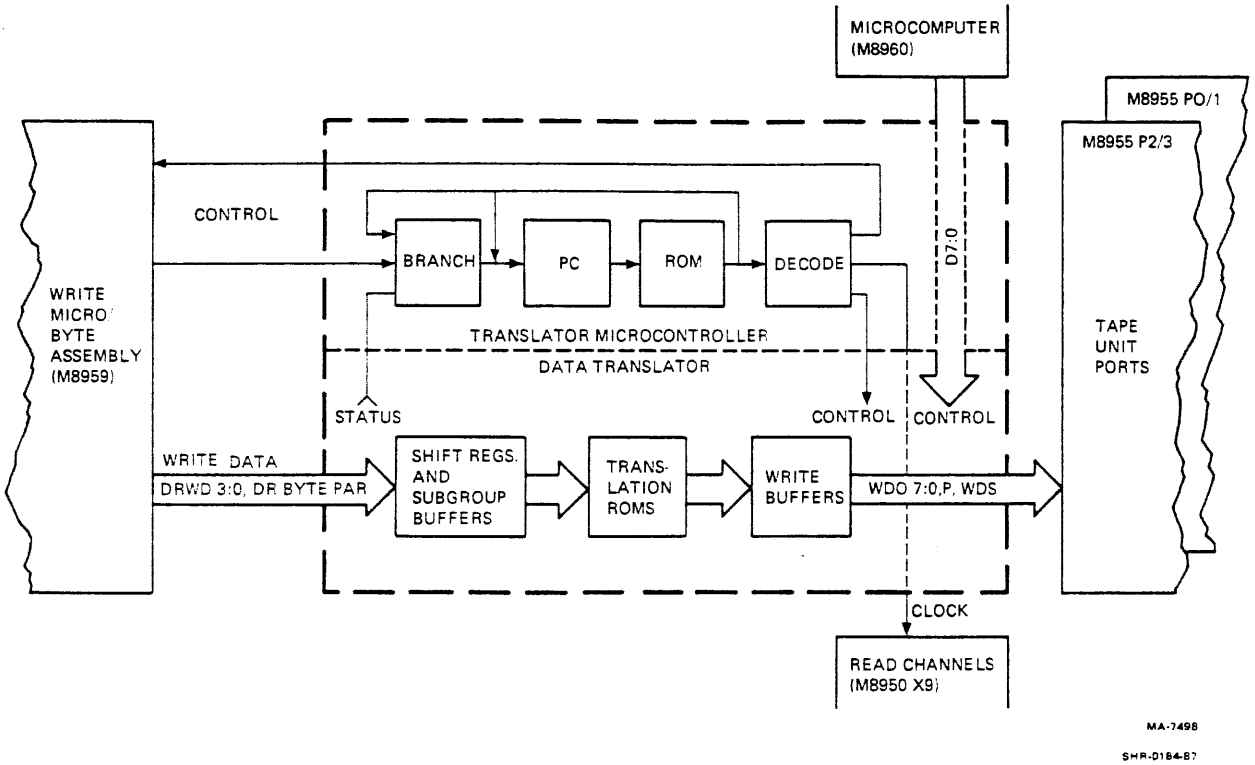
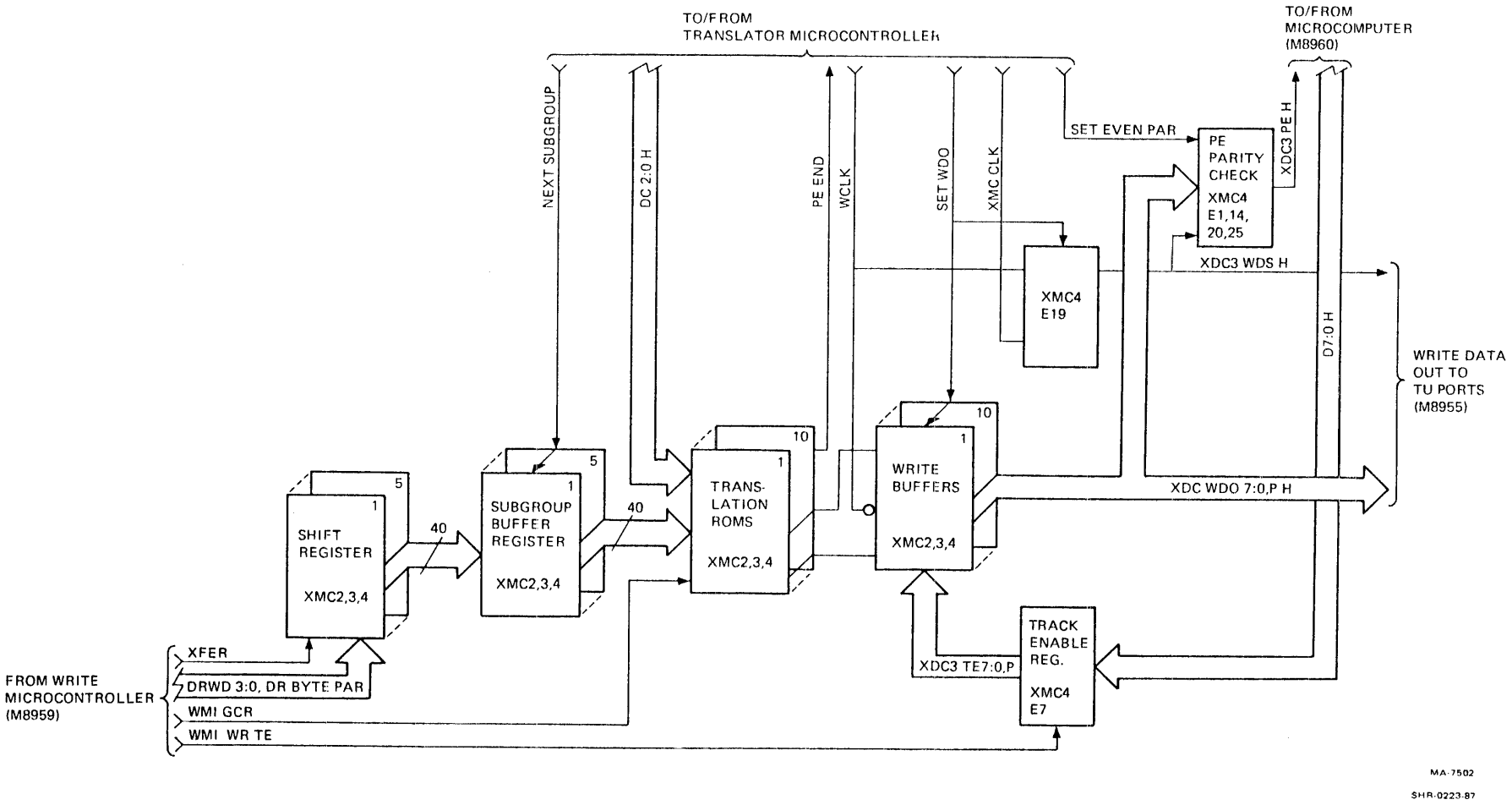


Figure 2-48 Data Translator



Each 5-bit transfer (four data plus one parity/done) is accompanied by a transfer pulse (WMC XFER H) from the write microcontroller. XFER clocks each of the five bits into the serial inputs of five independent 8-bit shift registers (XMC2—E18, E39; XMC3—E23, E40; XMC4—E17). Subsequent XFER pulses shift seven more 5-bit groups into the shift registers. The shift register outputs are presented in parallel to five subgroup buffers (XMC2—E12, E33; XMC3—E10, E41; XMC4—E16). While the translator microcontroller asserts the NEXT SUBGROUP pulse to the write microcontroller requesting another data subgroup, the present subgroup is clocked into the subgroup buffers.

The outputs of the subgroup buffers are sent to 10 translation ROMs and become part of each ROM's address input. Each buffer's output is split and interlaced to provide address components for two translation ROMs. Figure 2-49 offers a detailed illustration of this process. Additional ROM address inputs are WMI GCR H and DC 2:0 H. WMI GCR H provides the high-order address input and selects one of two tables in the ROM. When true, it selects the GCR mode translation table and when false, selects the PE mode multiplexing table. Signals DC 2:0 are the three binary outputs of the data control counter and provide the three low-order ROM address inputs.

The value of the data control counter and the current formatting mode determine the byte position of the translator output, relative to the data subgroup input. For instance, the eight nibbles held in the subgroup buffers actually represent four bytes, which in GCR mode must be translated into five storage group bytes or tape characters. The data control counter has a specific binary output that represents each of the five storage group tape characters. Similarly, in PE mode the four input bytes must be multiplexed and output as four tape characters and four possible phase transitions, so the binary output of the data control counter represents each of these eight conditions.

Figure 2-49 Translation ROM Addressing

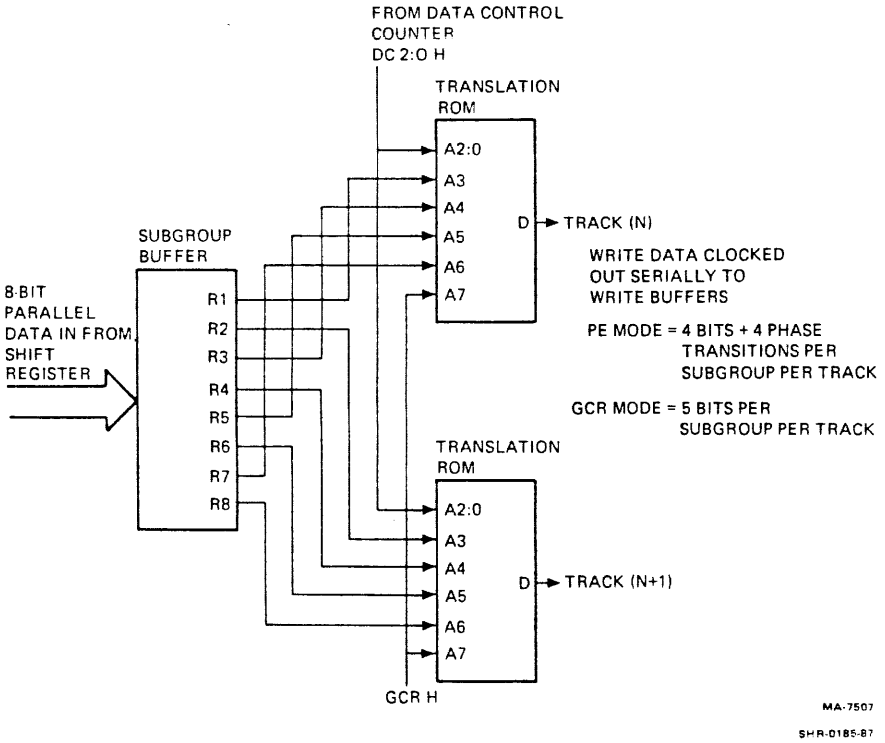


Table 2-13 shows the data control counter outputs and what they represent in each formatting mode. The word "Flip" in the table refers to the effect on the translator write buffers when the translator microcontroller issues a write clock (WCK) instruction. They are flipped, or changed to the opposite state, to provide the necessary phase transition.

Table 2-13 Data Control Counter Description

DC2:0 Value	PE Function	GCR Function
0	Multiplex byte A	Flip
1	Flip	No change
2	Multiplex byte B	0
3	Flip	Translate storage group byte A
4	Multiplex byte C	Translate storage group byte B
5	Flip	Translate storage group byte C
6	Multiplex byte D	Translate storage group byte D
7	Flip	Translate storage group byte E

Each of the 10 translation ROMs contains the same identical look-up tables and has a 256-location by 4-bit output. Outputs 0 and 1 of the first nine ROMs (data and parity) drive the nine write buffer flip-flops. Output 2 of the tenth ROM is the PE END flag that goes to the translator microcontroller. Output 3 is not used. The nine write buffer flip-flops provide the write data output signals (WDO 7:0, P H) to both TU port modules (M8955). The write buffers may be manipulated in one of three ways. First, a write clock instruction (WCLK) will clock data in from the translation ROMs. Second, the translator microcontroller may issue a Set Write Data Output (SET WDO) instruction followed by a WCLK instruction, in order to create the first all one character of a PE preamble. And third, combinations of write buffer bits may be held reset by the track enable register (TE 7:0, P H). This is done during the creation of ID bursts when eight of nine tracks must be held reset, or the creation of tape marks when three of nine tracks must be held reset. The track enable register (XMC4—E7) is selected by the signal WMI WR TE L decoded on the write microcontroller when the microcomputer issues a command to microbus address D2-W.

Accompanying a data character on the WDO lines is a write data strobe pulse (XDC3 WDS H/L) produced at XMC4—E19. This pulse latches the character into the selected tape transport. The frequency of write data strobe pulses is based upon the current recording mode and is controlled by a defined loop of instructions in the translator microprogram. In GCR mode, a loop of eight microinstructions (110 ns per microinstruction) creates a data rate of 9042 fr/in (flux reversals per inch). In PE mode, a loop of 23 microinstructions per phase transition (46 per bit cell) creates a data rate of 3200 fr/in. These numbers are based upon writing data to a transport moving tape at 125 in/sec.

The WDO lines also input to a vertical parity checker (WMC4—E1, E14, E20, E25) enabled in PE mode only as a write path integrity check. It is not used in GCR mode because the four vertical parity bits of a data subgroup translate into five storage bits that will not match a vertical parity scheme. A parity error is returned to the microcomputer's interrupt status buffer as XDC3 PE H.

Translator Microcontroller — A ROM controller (8 bits by 512 words) provides the programmable section of the translator module and controls the data translation and write data clocking rate. The controller is directed by commands from the microcomputer both directly and indirectly through the write microcontroller module. Figure 2-50 shows a simplified functional flow for this microcontroller's microprogram. Note that the microprogram circulates in a self-testing idle loop if not commanded to enter into a write operation (WMI EN XMC = 0).

Figure 2-51 shows the functional components of the translator microcontroller. The program ROM (XMC1—E38) is addressed with eight program counter bits (A7:0) and the high-order page bit (P1 H). This ROM provides an 8-bit-wide instruction word (I7:0). The program counter (XMC1—E32, 44) increments its address output by one at each XMC CLOCK pulse unless instructed to branch. ROM output I7, when false, strobes the branch condition multiplexer (XMC1—E35). Its output, BRANCH, loads the program counter with the balance of the ROM outputs (I6:0) and the low-order page bit (P 0 H), thus forming a new program address. If ROM output I7 = 1, then I6 is the odd parity bit. ROM parity is checked at XMC1—E25, E26, and if wrong is reported to the microcomputer's interrupt status buffer by means of the ROM PE H line. The parity error is latched, causing an LED on the translator to light.

The BRANCH output of the condition multiplexer is true if the input condition selected by the branch condition counter (XMC1—E37) is present. The microcontroller uses a special branch select (BRS) instruction to load the branch condition counter with low-order ROM bits I3:0. Bits I2:0 select one of eight conditions at the condition multiplexer input. If bit I3 is true, the XMC DONE flag returns to the microcomputer in a status word by means of the write microcontroller. This is the translator's way of signaling the completion of a PE/GCR postamble. Once loaded, the branch condition counter may be incremented by XMC CLOCKs, thus selecting different conditions to branch on. This allows the microcontroller to issue a series of branching instructions sequentially without interleaving them with BRS instructions.

ROM outputs are also sent to the ROM instruction decoders. The decoders provide the signals needed to direct the data translator's operation and provide synchronization with the write microcontroller. The write clock instruction provides a pulse to the XMC WCLK line. This is a data strobe to the nine read channel modules (M8950) during loop write-to-read internal diagnostics, when the phase lock loops are not used.

Figure 2-50 Translator Microprogram Flowchart (Sheet 1 of 4, Entry)

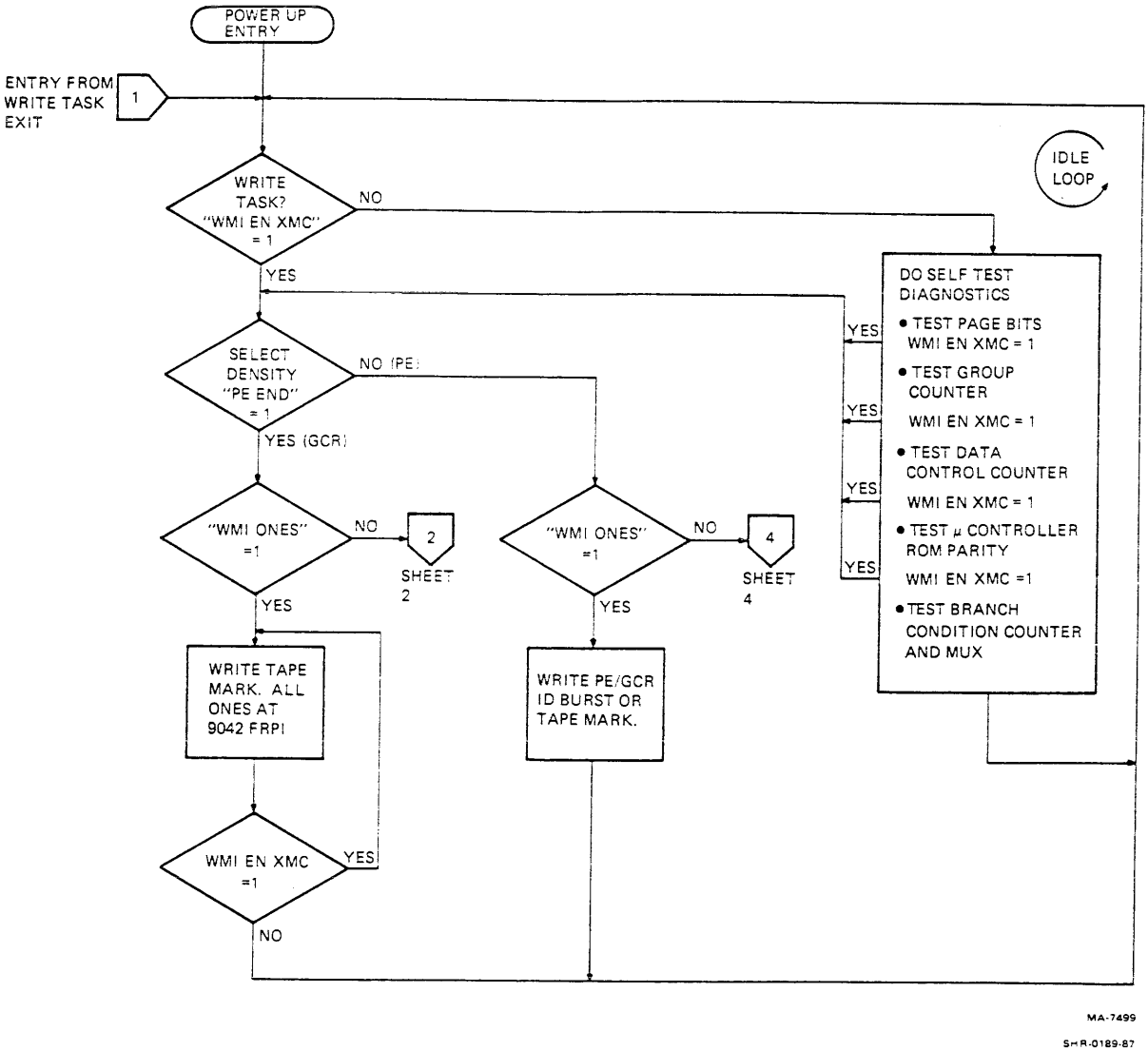


Figure 2-50 Translator Microprogram Flowchart (Sheet 2 of 4, GCR Write)

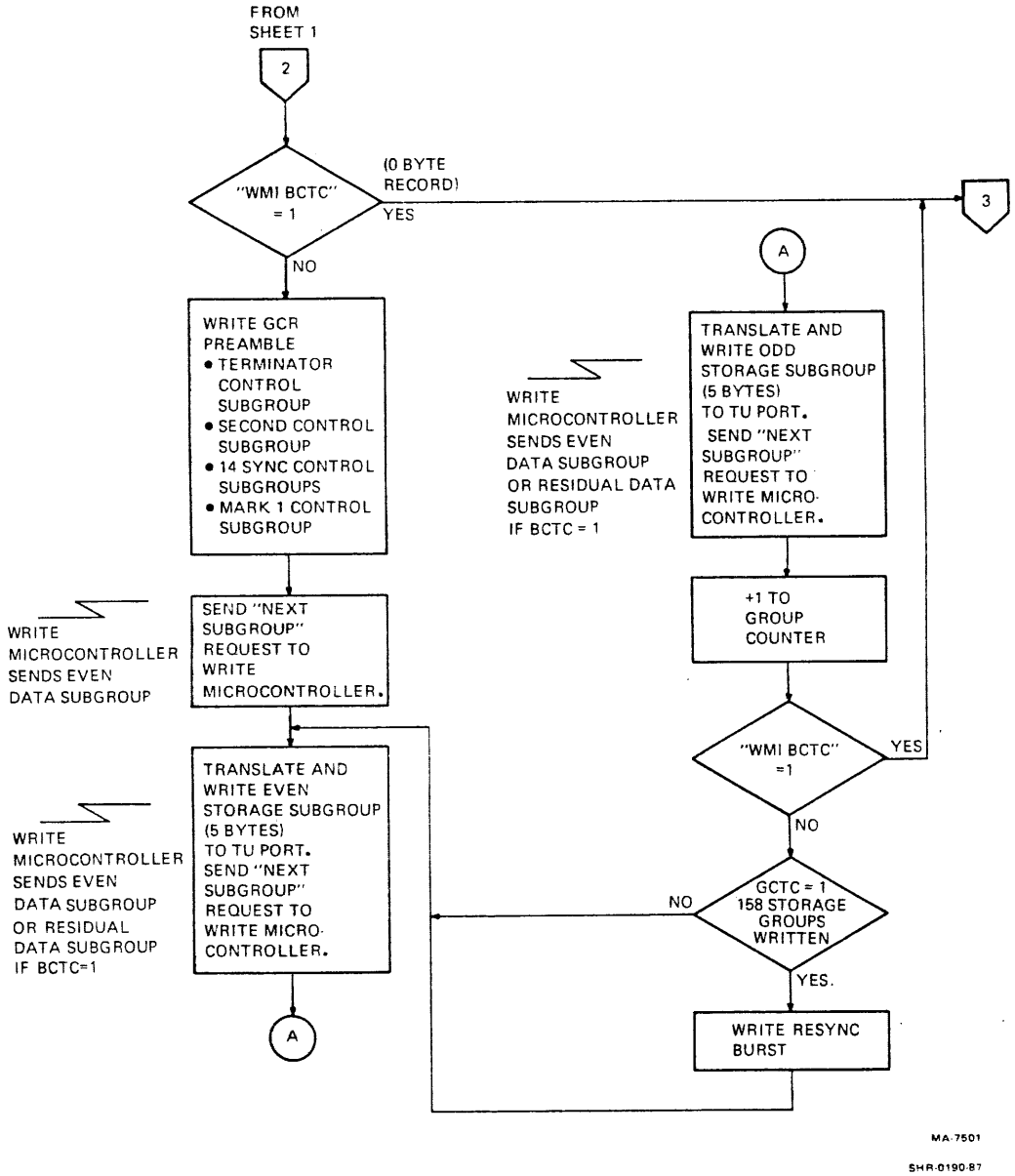


Figure 2-50 Translator Microprogram Flowchart (Sheet 3 of 4. GCR Write)

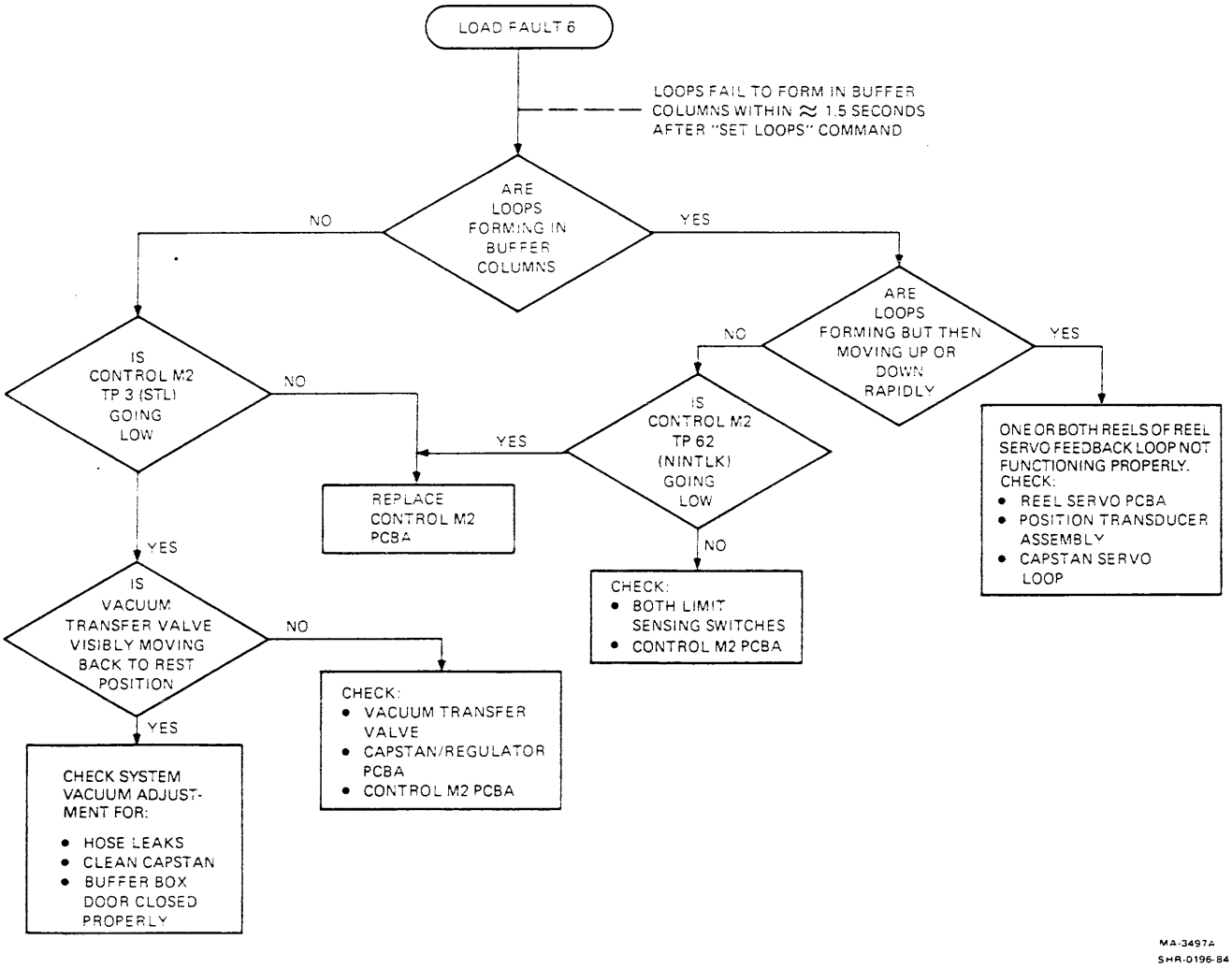


Figure 2-50 Translator Microprogram Flowchart (Sheet 4 of 4, PE Write)

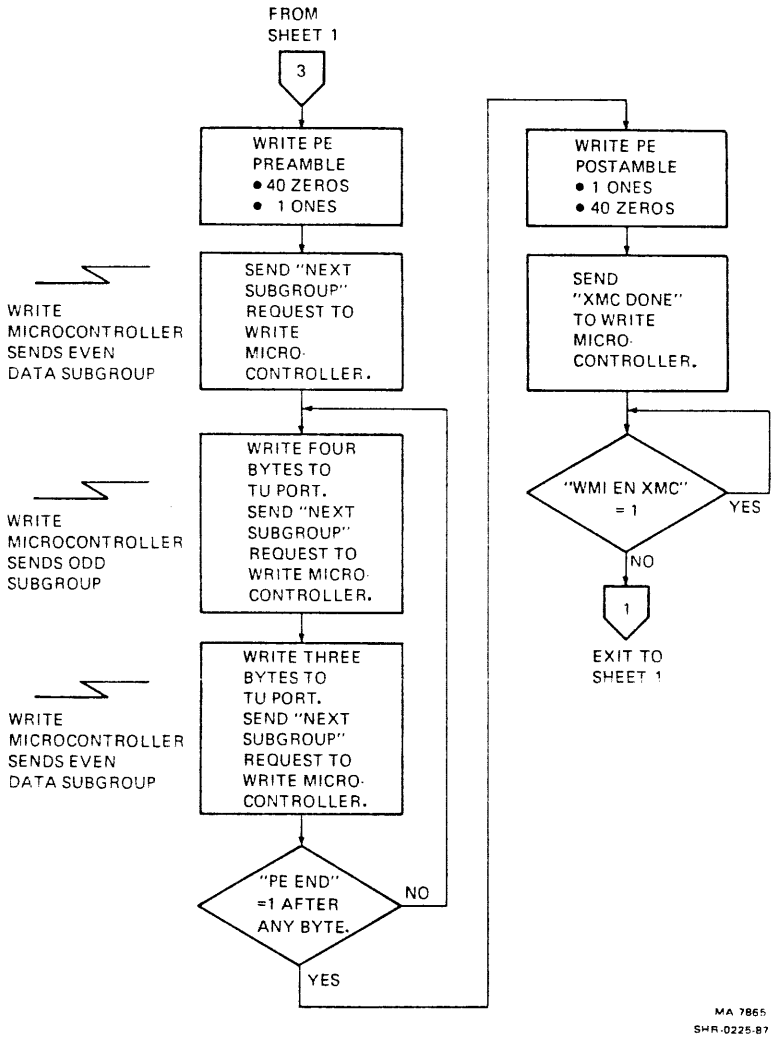
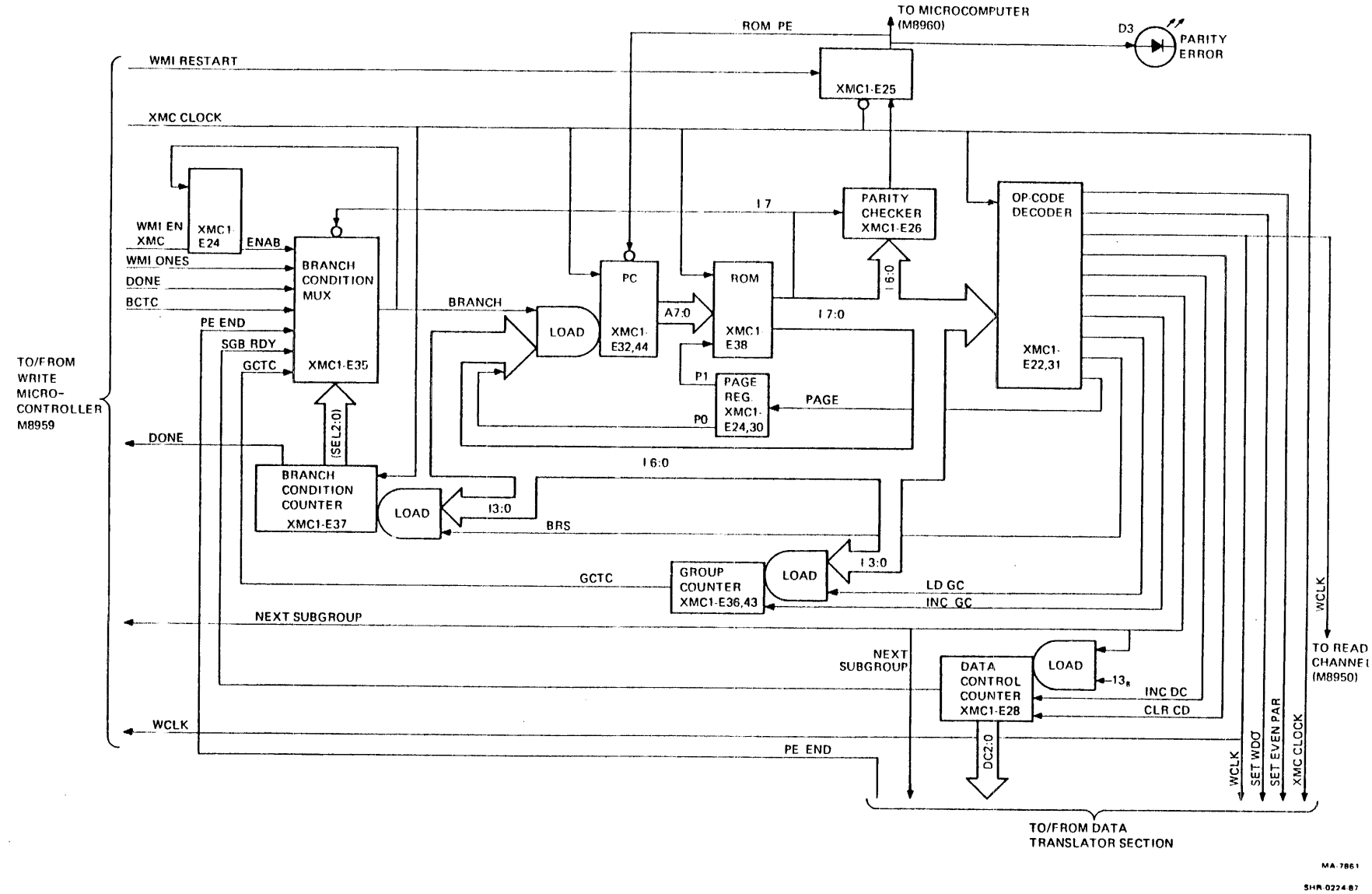


Figure 2-51 Translation Microcontroller



Two other counters are used in the microcontroller, the data control counter and the group counter. The data control counter controls the sequence of translation or multiplexing. The group counter handles a variety of operations. In GCR mode, it keeps track of the number of data groups translated and signals for a Resync burst, (if a count of 15810 is reached) by asserting group counter terminal count (GCTC) to the branch condition multiplexer. In PE mode, it counts the number of bytes multiplexed in an odd subgroup and asserts GCTC after a count of three. The fourth byte is an ECC character, which is dropped in PE mode. The group counter also serves as a general-purpose counter for the following tasks.

- Counts the number of zeros to be written in a PE preamble or postamble.
- Maintains a count for controlling flux reversal rate when writing PE (1600 fr/in) or GCR (3200 fr/in) ID bursts.
- Counts the number of one bits to be written in GCR control subgroups.

2.5.7 M8959 Write Microcontroller (WMC)

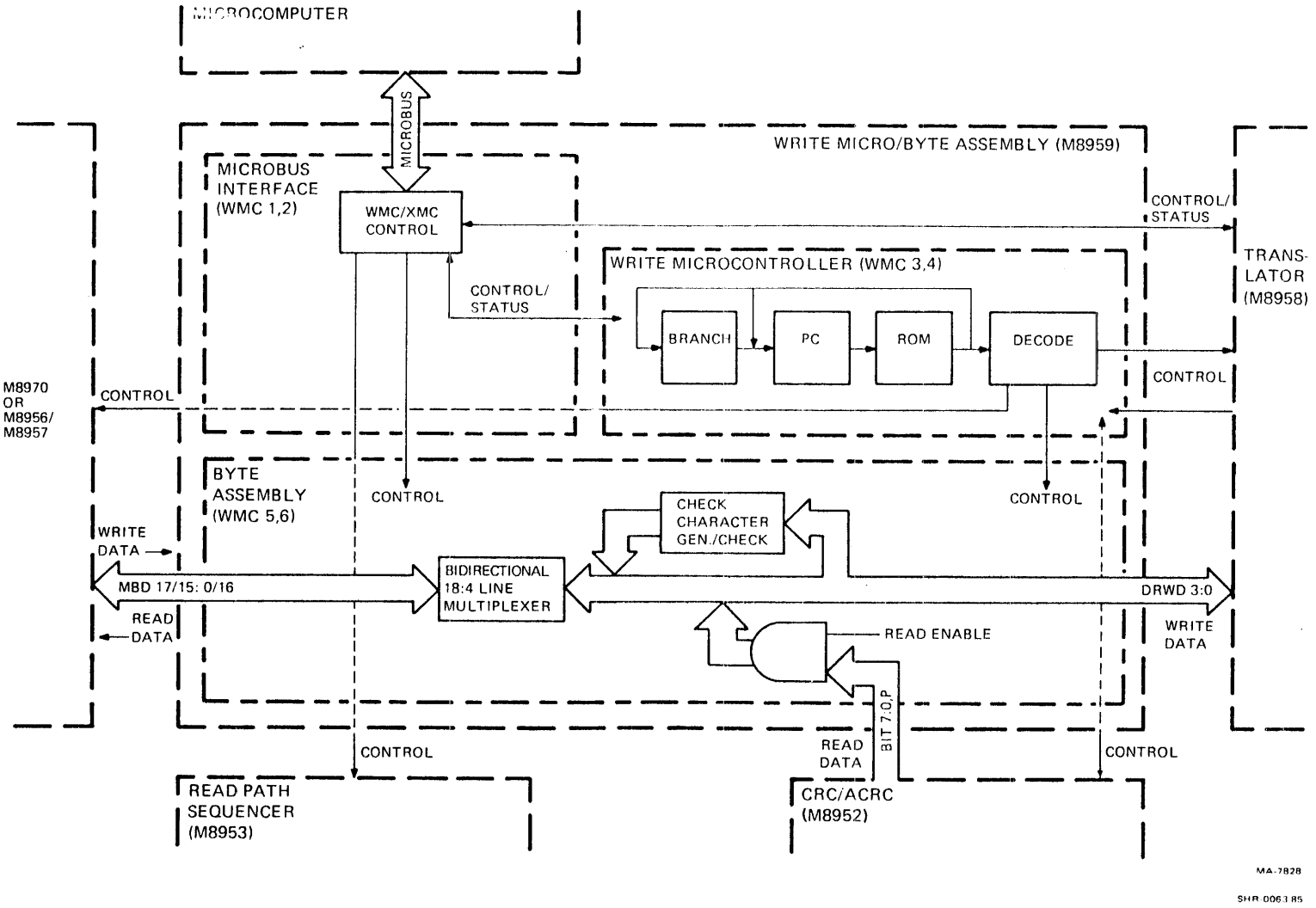
The M8959 write microcontroller (WMC) module functions in both write and read mode as the byte assembly logic. In write mode, it unpacks 16/18-bit Massbus data words received from the Massbus data module into 4-bit portions called nibbles. In the STI bus, it unpacks STI bus bytes received from the STI bus data module (4-bit nibbles). The nibbles are sent to the translator module for translation. In read mode, it receives 9-bit data characters from the read path and packs them into 16/18-bit Massbus data words or STI bus data bytes. In either mode, the byte assembly logic performs its pack/unpack function as determined by the format code specified by the host computer.

Figure 2-52 shows the three major functional areas of the write microcontroller module. They are the byte assembly logic, the write microcontroller, and the microbus interface. In addition to performing the pack/unpack functions, the byte assembly logic generates the various error-correction and check characters required for the selected format. The byte assembly logic also checks parity on data transfers from the Massbus/STIbus or read path, and generates parity on data transfers to the Massbus/STIbus or translator (M8958).

The write microcontroller section is a microprogrammed ROM controller, similar to the type used in the read path components (M8950 through M8953). It is under control of the system microcomputer and controls the byte assembly process in read and write mode. The write microcontroller works with the translator microcontroller to perform a synchronous handshaking operation during a write data transfer.

The remainder of the write microcontroller module is the microbus interfacing logic. The interfacing logic allows the microcomputer to send commands to the write microcontroller and the translator microcontroller. It also returns write path status from both microcontrollers to the microcomputer. In addition, the interfacing logic allows diagnostic data to be injected into or retrieved from the write data path, and loads values determined by the microcomputer into format control counters.

Figure 2-52 M8959 Write Microcontroller Functional Areas



Data Paths — Figure 2-53 shows the data paths portion of the write microcontroller module. Five data buses are used, three external to the module and two internal. The external buses are:

- Eighteen bidirectional Massbus data lines (MBD 17/15:0/16 H) that carry write/read data between the Massbus data module(s) and the write microcontroller module.
- Nine bit lines (CRC BIT 7:0, P H) that carry read data from the CRC/ACRC module to the write microcontroller module.
- Five write data lines (DR WD 3:0 H, DR BYTE PAR H) that carry write data and ECC, CRC, and ACRC characters from the write microcontroller module to the translator module.

The internal buses are:

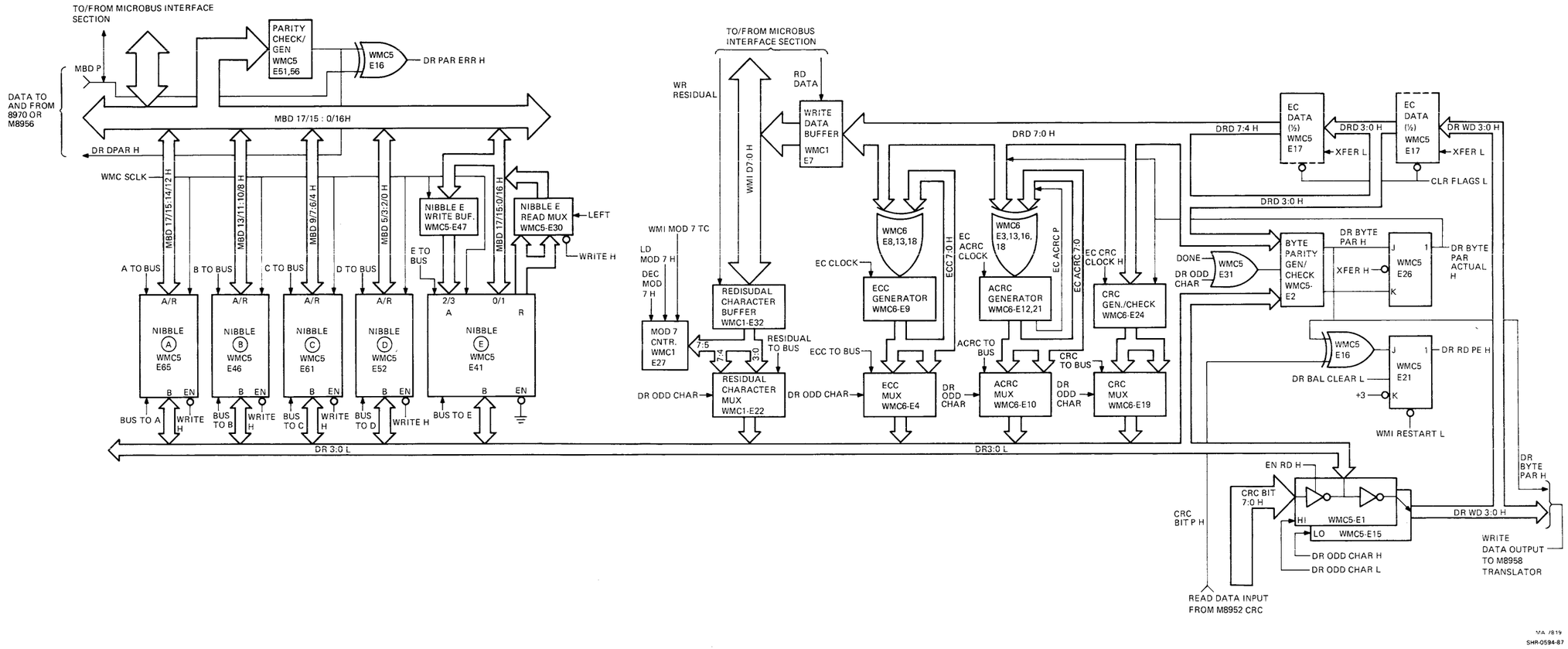
- Four bidirectional tri-state data register lines (DR 3:0 L) that carry write/read/formatting information on a nibble basis.
- Eight data register delayed lines (DRD 7:0 H) that carry write data on a byte basis to the generators/checkers for the ECC, CRC, and ACRC characters. The DRD lines also carry write/read data back to the microbus interfacing logic to be checked in diagnostic mode.

During a data write operation, the write microcontroller asserts the signal SCLK to request a word of data from the M8956 (Massbus) and the M8970 (STIbus) data bus interface modules. In response, the system sends a data word on the data lines which are strobed into the byte assembly logic on the trailing edge of SCLK.

During a data write operation, the write microcontroller asserts the signal SCLK to request a word of data (Massbus). In response, the host computer sends 16/18-bit data over the Massbus D lines. The data passes through the left/right multiplexer in the Massbus data module. The multiplexed data is placed on the MBD lines and is strobed into the byte assembly logic on the trailing edge of SCLK. The parity checker at WMC5—E51, E56 checks the MBD lines and compares them with the Massbus parity line MBD P at WMC5—E16. If wrong parity (even) was detected, the Massbus parity error bit returns to the microcomputer in an error status byte.

Now the data is captured in an array of logic elements, which together perform an 18-line to 4-line multiplexing operation (WMC5—E41, E46, E47, E52, E61, E65). The multiplexer is organized so the write microcontroller can manipulate data in five 4-bit nibbles (nibbles A through E). Nibbles A through D are used for all formatting modes; nibble E is used only for 36-bit word length manipulation. Depending upon data format, the write microcontroller asserts the signals A—E TO BUS, causing one nibble at a time to be gated to the DR lines. Data on the DR lines is inverted and sent out on the DR WD lines to the translator and the check character logic.

Figure 2-53 Write Microcontroller Byte Assembly Logic



VA /819
SHR-0594-87

When the write microcontroller gates a nibble to the DR/DR WD lines, it also generates a transfer pulse (XFER). Data is clocked on the trailing edge of the XFER pulse into the translator and the first half of the EC data register (WMC5—E17). The output of the EC data register (first half) provides one half of the DRD bus (DRD 3:0 H). DRD 3:0 H and DR 3:0 L are the inputs for the byte parity generator (WMC5—E2). A byte parity bit (odd parity) is generated for every two nibbles transferred (DRD 3:0 being the previous nibble and DR 3:0 being the present nibble) and sent to the translator over the multiplexed DR BYTE PAR H line.

The byte parity generator is enabled in one of two ways, either by the DR ODD CHAR signal or the DONE signal through OR gate WMC5—E31. DR ODD CHAR goes true on every other nibble transfer and enables the parity generator after every eight bits sent in PE or GCR mode. The DR BYTE PAR line may also transmit the WMC DONE status bit to the translator in PE mode, during the time slice when a parity bit is not asserted. This is accomplished by the write microcontroller asserting DONE at the completion of a record (byte counter overflow), and clearing the DR and DRD lines to provide no data inputs to the byte parity generator. No true inputs to the parity generator cause it to produce an output indicating done or completion of all data transfers to the translator for this record. DR BYTE PAR H also goes to the J-input of the actual parity flip-flop. Its output, DR BYTE PARITY ACTUAL, will be correct on all even-nibble transfers and incorrect on all odd-nibble transfers. This is of no consequence, however, because it is not examined on odd transfers. DR BYTE PARITY ACTUAL becomes the parity input to the CRC and ACRC character generators.

On the second XFER pulse and all XFER pulses thereafter, data shifts from the first half to the second half of the EC data register. This allows a full byte of data to appear on the DRD lines every other XFER pulse. DRD 7:0 is presented to the inputs of the check character logic. The check character logic consists of an error correcting character (ECC) generator, an auxiliary cyclic redundancy character (ACRC) generator, and a cyclic redundancy character (CRC) generator/checker.

When writing in either PE or GCR mode, data transfers from the byte assembly logic to the translator on a subgroup basis. Because there are four bytes per subgroup, eight nibbles with four parity bits are sent with eight XFER pulses. In GCR mode, this is always the case. In PE mode, a partial subgroup may be sent if the byte counter overflows somewhere in the middle of the subgroup. In either mode, seven data bytes plus one ECC byte are sent; this fulfills the requirements of GCR mode. In PE mode, the translator ignores the eighth byte (ECC) and simply requests the next subgroup.

Each of the three check character generators is clocked once per data byte transferred. They are not clocked at the time the check character (ECC/ACRC/CRC) nibbles are gated to the DR/DRWD/DRD lines.

The write microcontroller gates the ECC character to the DR lines by supplying two pulses to the ECC TO BUS line. The first pulse gates ECC 7:4 to DR 3:0 through multiplexer WMC6—E4. The second pulse causes the DR ODD CHAR line to change state, and ECC 3:0 is gated to DR 3:0. The other two check characters are gated to the DR bus in much the same manner. In GCR mode, the ACRC character is gated out of WMC6—E10 with two pulses to the ACRC TO BUS line, as the third byte in subgroup B of the residual data group. The CRC character is gated out of WMC6—E19 with two pulses to the CRC TO BUS line. This is done five or six times in GCR mode, when writing the CRC data group. It is done once in PE mode, even though the PE format does not use a CRC character.

During a PE data write operation the CRC generator builds a CRC character. This character verifies that a good record was written on tape. While the translator generates postamble information, the byte assembly logic transfers the CRC character to the EC data register. Again, similar to the ECC character, the translator ignores the CRC character on its data input lines.

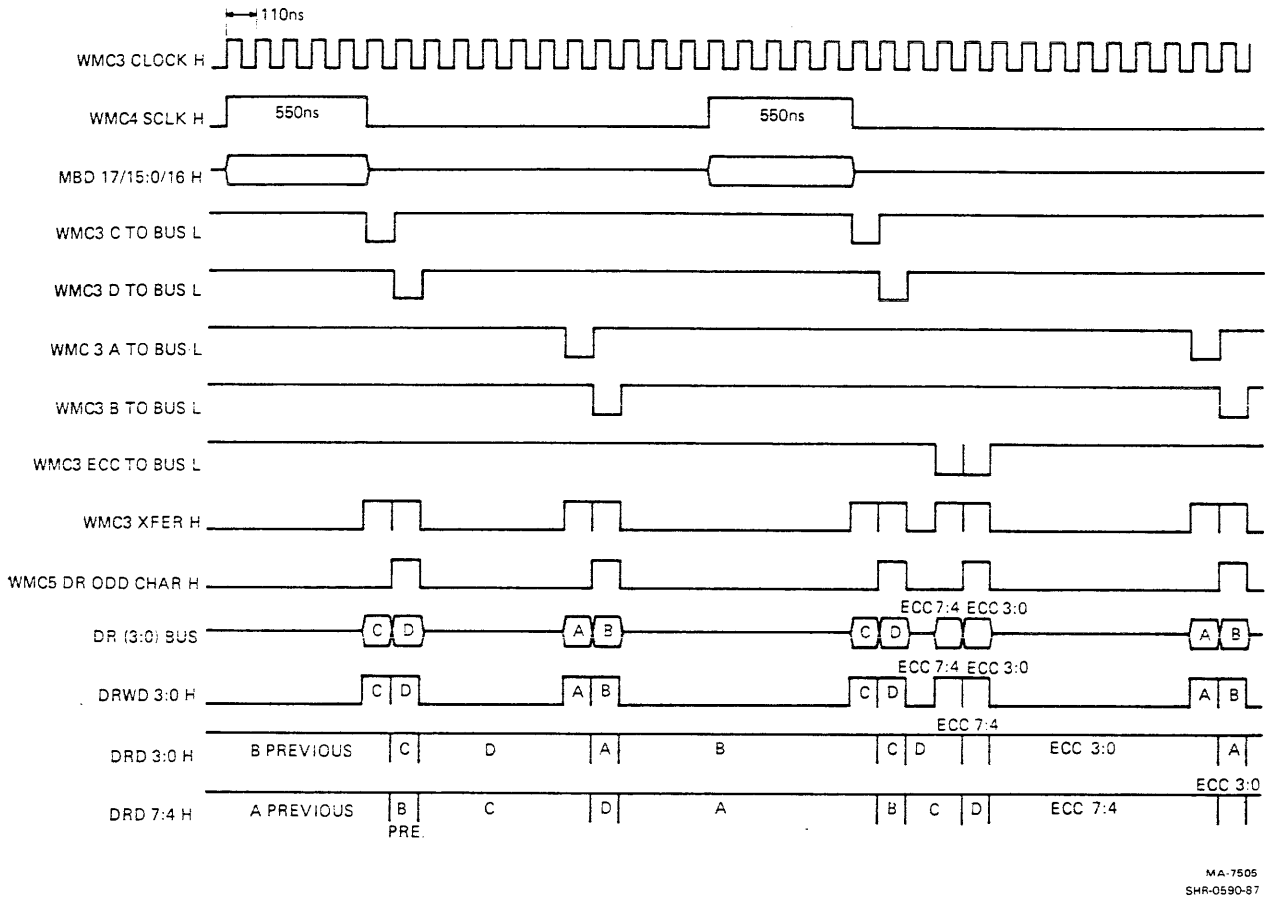
When a record completes, the microcomputer requests the CRC character. The microbus interface section generates the decoded signal RD DATA; the CRC character on the DRD lines is gated through the write data buffer (WMC1—E7) to the internal data lines (WMI D 7:0 H) and then to the microbus D lines. The microcomputer also requests the CRC character from the read path when the read-after-write function is complete, and then compares the two. If a match does not result, the record just written on tape is suspect.

In GCR format, the seventh byte of the CRC data group is the residual character. Prior to the start of an actual write to tape, the microcomputer calculates the two parts of the residual character (modulo-7 and modulo-32 counts) based upon the byte count specified by the host computer. The microcomputer sends the residual character over the microbus to the write microcontroller module, and it is gated to the residual character buffer (WMC1—E32) by the decoded signal WR RESIDUAL. When the moment comes to write the residual character to tape, it is gated to the DR lines with two RESIDUAL TO BUS pulses through multiplexer (WMC1-E22).

The modulo-7 count comprises the three high-order bits of the residual character. This is a binary value that represents the number of data bytes to be written in byte positions one through six of the residual data group in GCR mode. When the write microcontroller goes through its initial setup operations for a write function, it loads bits 7:5 of the residual character into the modulo-7 counter (WMC1—E22) with the signal LD MOD 7 H. During the residual group, the counter is decremented by the signal DEC MOD 7 H each time a residual data byte is sent to the translator. When the counter underflows, it produces the signal WMI MOD 7 TC. This notifies the write microcontroller that it must gate out pad (zero-filled) characters or the ACRC character.

Figure 2-54 shows the timing relationships for a typical write data flow through the byte assembly logic. The example illustrates subgroup B of an odd data group in PDP-11 normal formatting mode. As mentioned, the ECC character is gated out of the byte assembly logic regardless of the selected density format; so this diagram may apply to PE as well as GCR mode.

Figure 2-54 Timing for Typical Write Data Flow through Byte Assembly Logic in PDP-11 Normal Mode



The byte assembly logic handles a read operation in much the same manner as a write operation. The significant differences are that data moves in the opposite direction and the check character generators are not used.

Prior to the start of a read data transfer, the write microcontroller enables the receiver portion of the transceiver pair WMC5—E1.15 with the signal EN RD H. This allows the read data on the bit lines to be placed on the DR lines. To start, the CRC microcontroller asserts BYTE RDY H. This informs the write microcontroller that a byte of data is on the bit lines (CRC BIT 7:0, P H). The byte is gated to the DR lines in two 4-bit nibbles. CRC BIT 7:4 or 3:0 are gated depending upon the state of the DR ODD CHAR flip-flop. The nibbles are gated to the bidirectional 18-line to 4-line multiplexers by the signals BUS TO (N), and appear immediately on the MBD lines. The write microcontroller informs the CRC microcontroller that it has accepted the data byte, by asserting the signal WMC5 DR BAL CLEAR L.

The BUS TO (N) signals are accompanied by a XFER pulse, which loads data into the EC data register (WMC5—E17). The nibble output of the EC data register first half is sent to the byte parity checker (WMC5—E2) over the DRD 3:0 H lines. The DR lines provide the second nibble to the byte parity checker, which is then enabled by the DR ODD CHAR signal. The odd output of the parity checker (DR BYTE PAR H) is XORed with the read parity bit (CRC BIT P H) at WMC5—E16. If the parity bits do not match, an error indicator is latched into the read parity flip-flop by the DR BAL CLEAR signal. Then the parity error returns to the microcomputer in a status byte.

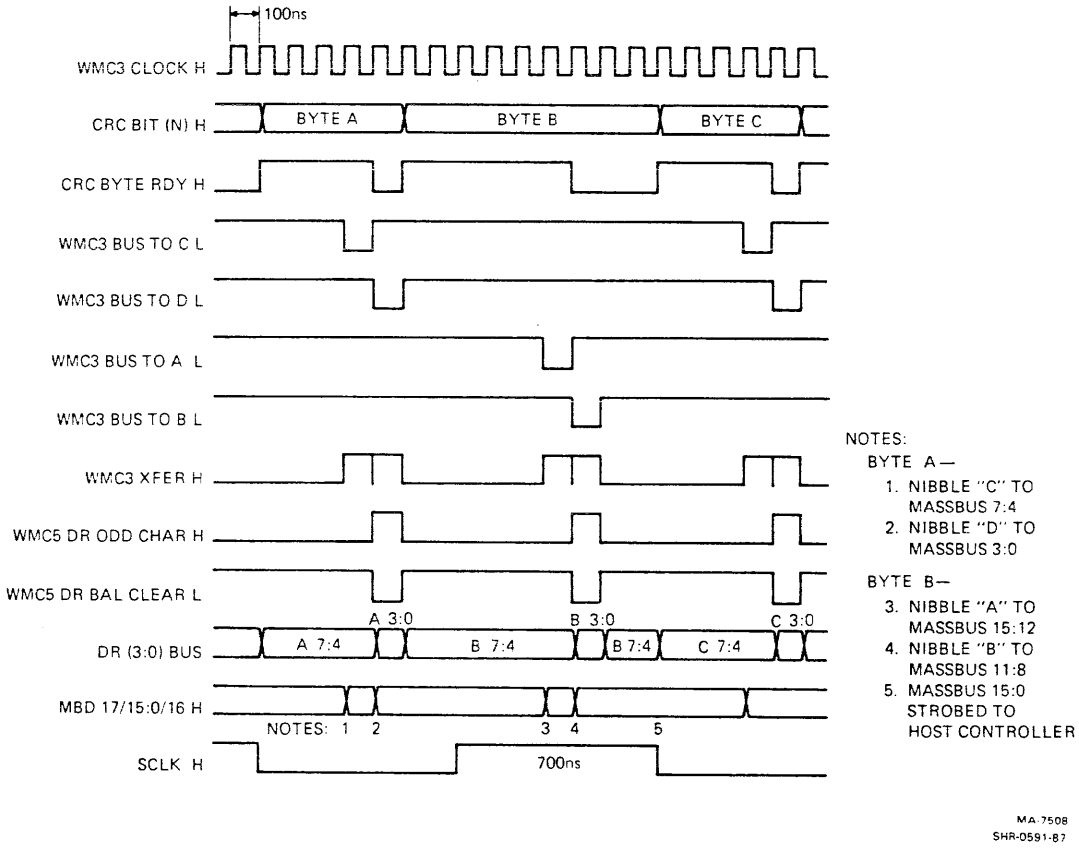
For the Massbus operation, this handshake continues until data is loaded into all five, or as few as two, of the multiplexer latches, depending upon the formatting mode. Massbus data parity is generated at WMC5—E51, E56 and sent to the Massbus data module on the DR DPAR H line. The write microcontroller asserts SCLK to the Massbus controller when the required number of nibbles is assembled. Read data on the Massbus D lines is strobed on the trailing edge of SCLK to the Massbus controller.

For the STI bus operation, this handshake continues until data is loaded into two of the multiplexer latches. The write microcontroller asserts SCLK to the M8970 when the required number of nibbles (2) is assembled. Read data on the MBD lines is strobed on the trailing edge of SCLK to the M8970.

Because read data is clocked through the EC data register, it may be viewed by the microcomputer by strobing the DRD lines to the WMI D lines and then to the microbus. This is done in low-speed single-step mode when running wraparound diagnostics.

Figure 2-55 shows the timing relationships for a typical read data flow through the byte assembly logic. Shown is subgroup B of the odd data group in PDP-11 normal mode during read forward.

Figure 2-55 Timing for Typical Read Data Flow through Byte Assembly Logic in PDP-11 Normal Mode



ROM Controller and Clocks — A ROM controller (8 bits by 1024 words) provides the programmable section of the write microcontroller module and is directed by commands from the microcomputer. Figure 2-56 shows a simplified functional flow for this controller's microprogram. Note that the microprogram circulates in a self-testing idle loop, if not commanded to perform a subsystem read/write operation.

Figure 2-56 Write Microcontroller Flow

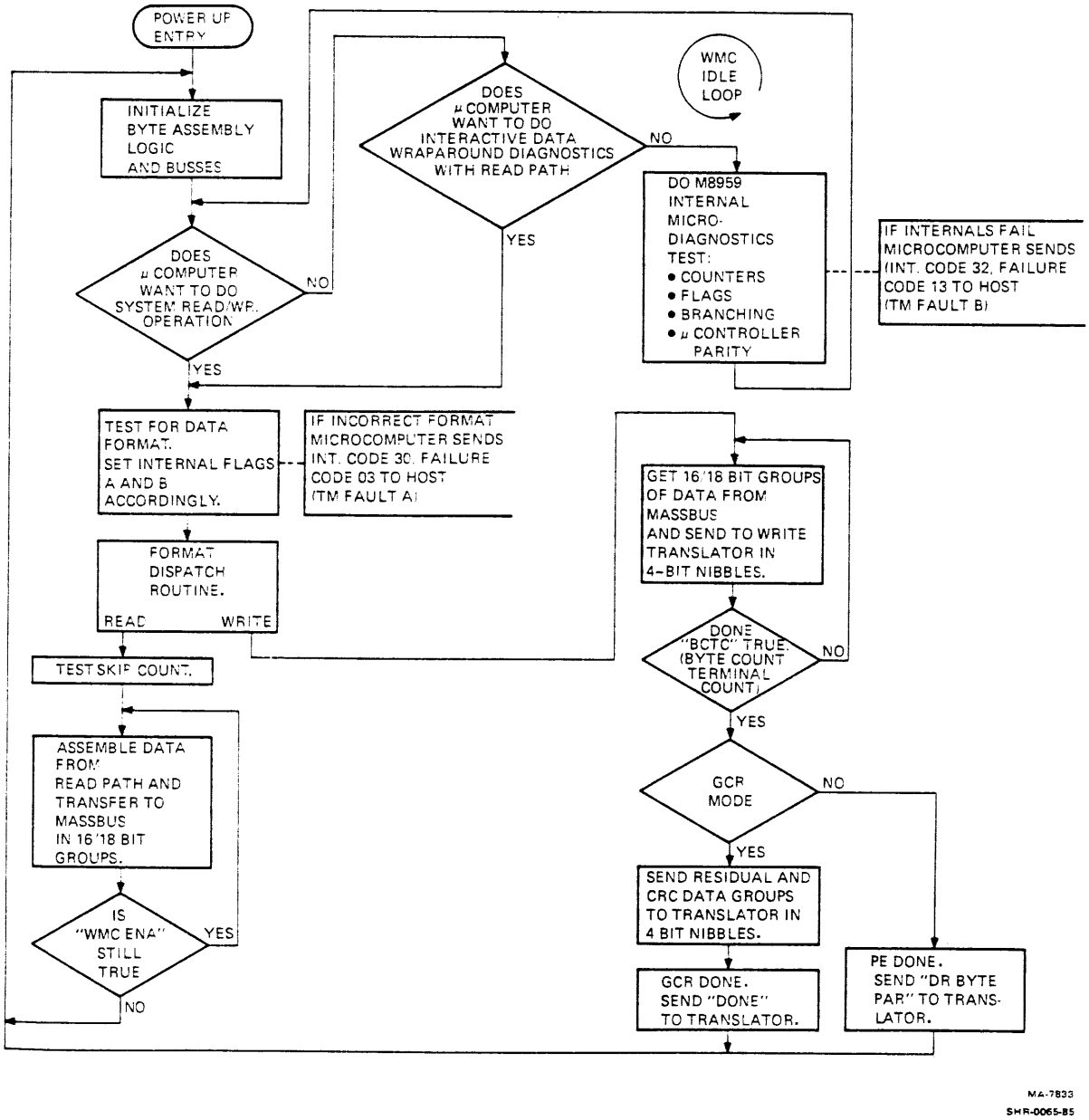


Figure 2-57 shows the ROM controller and clocking logic. Write clock pulses enter the write microcontroller module from the microcomputer. They are divided by two at WMC3—E66 to become WMC/XMC CLOCK pulses. The XMC clocks are sent to the translator microcontroller. The WMC clocks strobe the program counter and ROM to provide the basic microinstruction timing. The program ROM is addressed with 10 program counter bits (A9:0) and provides an 8-bit-wide instruction word (I7:0). The program counter increments its address output by one at each clock pulse, unless instructed to branch. ROM output I7, when true, strobes the branch gate (WMC4—E55). Its output loads the program counter with the balance of the ROM outputs (I6:0) and the page register bits (P2:0). If ROM output I7 = 0, then I6 is the odd parity bit. ROM parity is checked at WMC4—E59; and if wrong, it is reported to the microcomputer in a status word by means of the ROM PE line. The parity error is also latched and causes an LED on the module to light.

The microcontroller uses a special branch select (BRS) instruction that selects 1 of 16 conditions to examine when encountering a branch instruction. The selected condition is presented to the output of the branch condition multiplexer (WMC4—E53) and sent to the branch gate. If the condition is true, the branch gate is inhibited and the program counter will not branch. Conversely, if the condition is false, the program counter will branch.

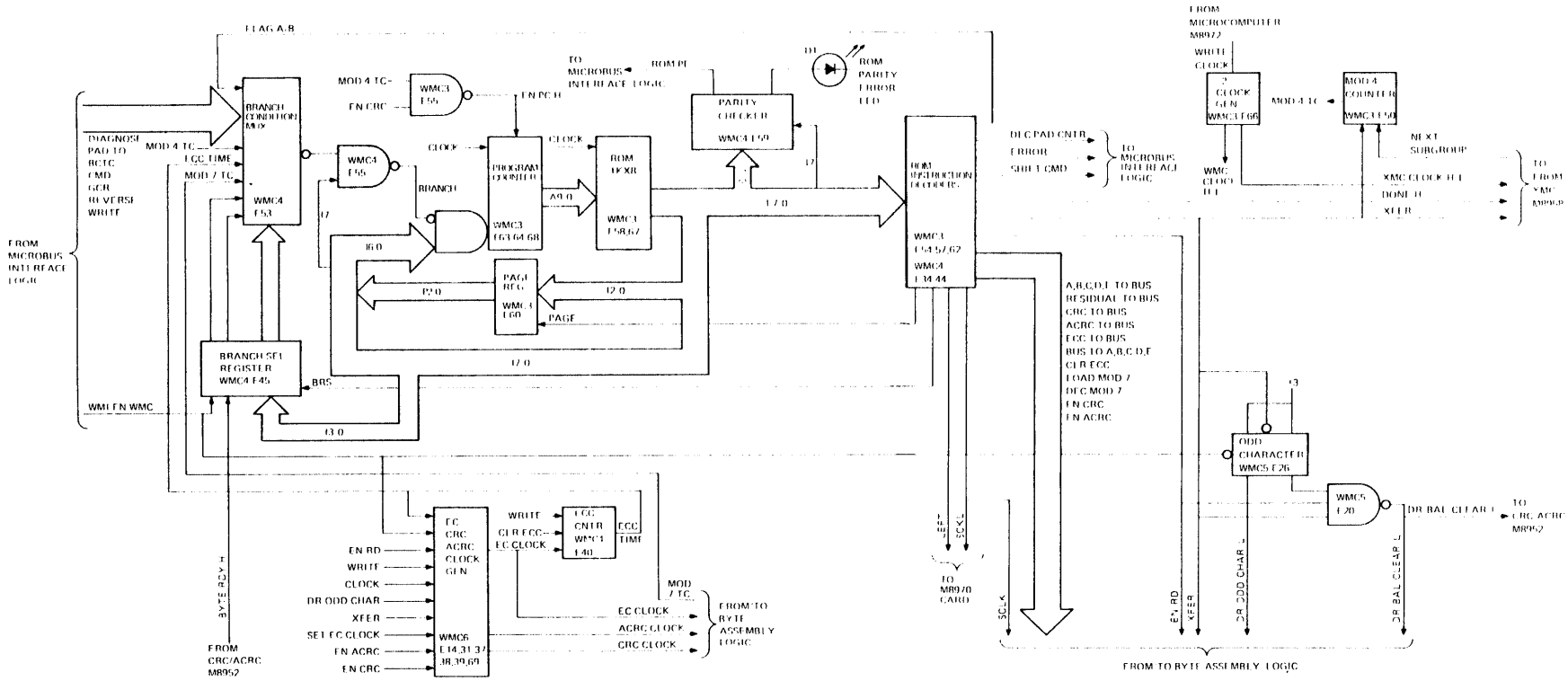
ROM outputs are also sent to the ROM instruction decoders. The decoders provide the signals needed to direct the operation of the byte assembly logic and BUS data transfer timing, and provide synchronization with the translator and CRC modules.

The write microcontroller is synchronized with the translator microcontroller during a write operation through the XFER and NEXT SUBGROUP lines. The translator asserts the NEXT SUBGROUP pulse, which loads zeros into the modulo-4 counter (WMC3—E50). The modulo-4 counter is incremented each time the write microcontroller asserts XFER (with an accompanying data nibble from the byte assembly logic). On the eighth XFER pulse, the modulo-4 counter produces its terminal count output (MOD 4 TC), which enables the program counter disable gate at WMC3—E55. This disables the program counter and stops the write microcontroller program until the translator is ready for another subgroup.

The write microcontroller is synchronized with the CRC microcontroller during a read operation through the BYTE READY and BAL CLEAR lines. The CRC microcontroller asserts the BYTE READY signal to the branching logic, indicating that it has presented a byte of data to the byte assembly logic. The byte is processed, then the write microcontroller asserts the BAL CLEAR pulse to the CRC microcontroller, signifying that it has accepted the byte and is ready for another.

The ECC/CRC/ACRC clock generator produces three types of clock pulses for the check character generators. One EC CLOCK pulse is produced for each data byte (alternate nibble) sent to the translator. However, no EC CLOCK pulse exists for the ECC byte. This means that seven EC CLOCK pulses are produced for each data group transferred to the translator. The EC CLOCK pulse decrements the byte counter during read/write operations. EC CLOCK also increments the ECC counter (WMC4—E40). The ECC counter is configured as a modulo-7 counter; it produces the ECC TIME signal to the branching logic and the EC CLOCK generator after seven data bytes are transferred. Similarly, CRC and ACRC CLOCK pulses are produced in the same sequence, but are not generated during a read operation.

Figure 2-57 Write Microcontroller



Microbus Interface — The microbus interface logic allows the microcomputer to send commands to and receive status from the entire write path. It also provides a path to send and receive diagnostic data. Refer to Figure 2-58 for a functional diagram of the microbus interface.

An address decoder (WMC2—E25, E43) enabled by the signal SEL WMC/XMC decodes the four least significant address bits (A3:0). Further microcomputer controls are the read/write signals specifying the direction of data transfer on the microbus D lines. Tri-state read/write buffers isolate the D lines from the internal data bus WMI D7:0 H. Outputs from the address decoder select the data buffers, various registers and buffers connected to the internal data bus, and functions external to the write microcontroller module.

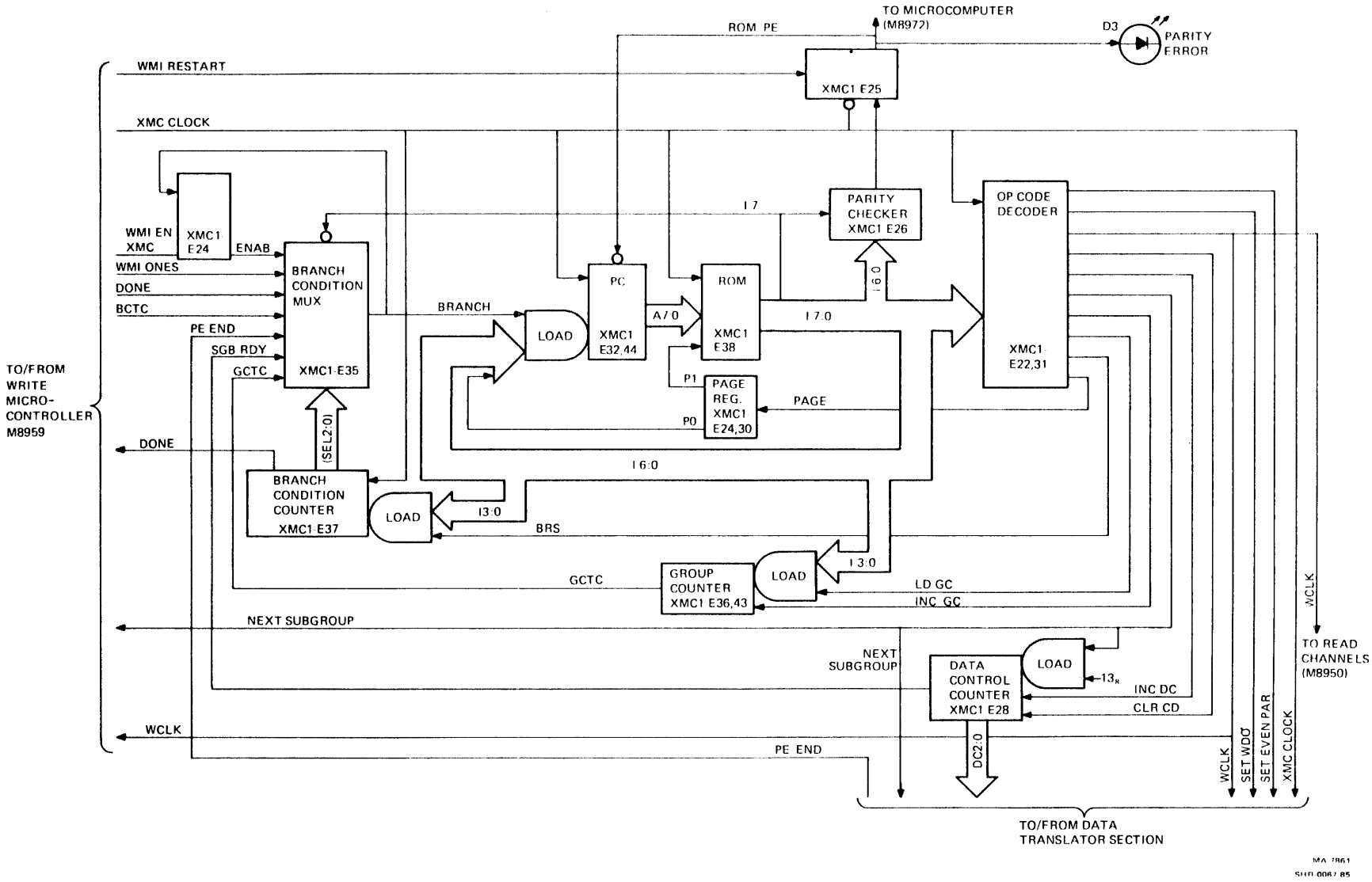
Signal WMI RD TU SEL (N) L selects one of two TU select buffers on the TU port module. Signal WMI WR TE selects the track enable register in the translator logic. Signal WMI RD status enables the WMC status buffer (WMC1—E28) to the data lines. The status buffer serves largely for diagnostic routines; it sends the translator DONE signal to the microcomputer, signifying that the postamble is written.

Signal WMI WR CMD enables the data lines to the data control register (WMC1—E23). The data control register is a parallel-in, serial-out device, which is connected to recirculate its contents. This register holds the skip count and format information; it is shifted out serially by the SHIFT CMD pulse from the ROM controller. Signal WMI SEL CTRS enables the counter chip (WMC1—E36). The counter chip contains three independant binary counters that may be loaded and examined by the data bus. The microcomputer loads the byte counter at the beginning of a record and is decremented by the EC CLOCK pulse. When the byte counter reaches zero, it sends terminal count (WMI BCTC) to the ROM controller. The microcomputer loads the pad counter at the beginning of a record and is decremented by the DEC PAD CTR pulse during the residual data group. When the pad counter reaches zero, it sends terminal count (WMI PAD TC) to the ROM controller. The Ecode counter is not loaded, but rather starts out at the beginning of a record in a reset state. If any error conditions are encountered while writing a record, the ROM controller decrements this counter with the ERROR pulse. When the record completes, the microcomputer requests the value in the Ecode counter and begins appropriate action.

Signal WMI WR CTRL allows a command on the data lines to be written into the WMC control register (WMC1—E42). Control signals go to the ROM controller, translator and read path controller. Signal WMI RD DATA goes to the byte assembly logic, where it enables the write data buffer (DRD lines to data lines). Signal WMI WR RESIDUAL goes to the byte assembly logic, where it enables the data lines to the residual character buffer.

Finally, signal WMI SEL DDR enables the diagnostic data register/error status chip (WMC2—E35). The DDR chip allows test data to be sent from the microcomputer to the 18 MBD lines. Data is then processed through the write and read paths; it appears back on the DRD lines, where it may be sent back to the microcomputer for comparison. The DDR serves as the passageway for extended sense data from the microcomputer RAM area through the bus data module to the host computer. Four error status bits may be read by the microcomputer through the error status portion of the chip. They are the bus parity error (DR MB PE), read data parity error (DR RD PE), ROM controller parity error (ROM PE), and the error instruction pulse (ERROR).

Figure 2-58 M8959 Microbus Interface



2.6 FORMATTER POWER SUPPLY SYSTEM

The following paragraphs provide mechanical and functional descriptions of the power supply. Detailed electrical interconnection diagrams of the power supply and power distribution system are included.

2.6.1 Mechanical Description

The formatter power supply consists of the following elements.

- One H7423 power supply chassis
  or
- One H7422 power supply chassis

  Two H7441 regulators

  One H7476 regulator

  One H7490 regulator

  One 54-14192 AC/DC LO module

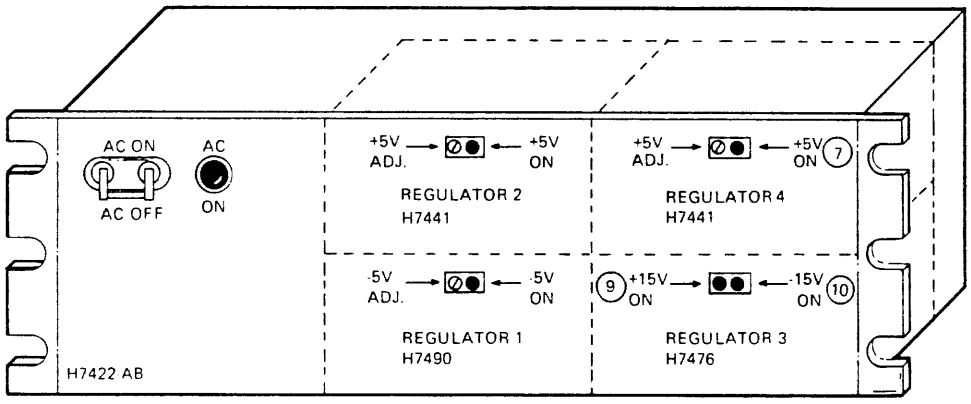
NOTE
Earlier models of the TU78/TA78 transports may contain the H7422-AB power supply. Later TU78/TA78 and all TU79/TA79 transports contain the H7423 power supply.

(All regulators and the AC/DC LO module are housed within the H7422 power supply chassis. Two module carriers hold the regulators captive within the chassis. There are two regulators per carrier. To service the regulators, remove the power supply front panel and slide the carriers forward for removal. This can be done without removing the supply from the equipment rack.)

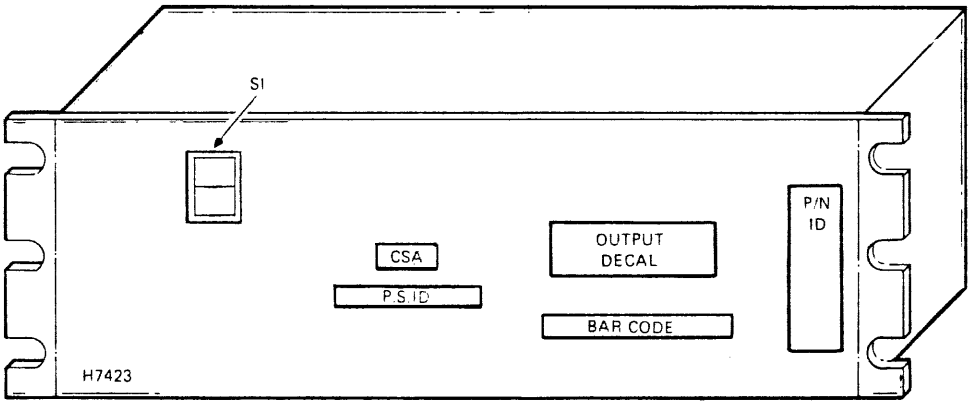
The H7422 measures 13.3 cm (5.25 in) high by 48.3 cm (19 in) wide by 27.9 cm (11 in) deep and mounts in a rack with 48.3 cm (19 in) centers. The H7422 weighs 20.4 Kg (45 lbs) including the regulator assemblies. Cooling is provided by an internal fan: it forces air laterally through the supply, past the transformer, and across the regulator heat sink fins.

Figure 2-59a shows the H7422 power supply (front view). Figure 2-59b shows the H7423 power supply. The dotted lines on Figure 2-59 indicate the relative position of the four regulator assemblies within the supply chassis of the H7422-AB. The H7423 has only one module. Cutouts in the front plate provide access to the regulator voltage adjustments and reveal the power on indicators. There are six power indicators: one for ac power on, one for each +5 V regulator, one for the 5 V regulator, one for the +15 V regulator, and one for the —15 V regulator. There are three voltage adjustments, one for each +5 V regulator and one for the —5 V regulator.

Figure 2-59 TS78 Power Supply



a. H7422 POWER SUPPLY



b. H7423 POWER SUPPLY

SHR-0596-87

2.6.2 Output Power Specifications

Table 2-14 lists the modules used in the power supply and their output power/signal specifications. Also listed are the jack numbers through which the power/signals are available.

Table 2-14 TM78 Power Supply Output Power Characteristics

Regulator	Voltage and Regulation	Output (max)*	Peak-to-Peak Ripple (max)	Jack No.
H7490 (No. 1)	—5 Vdc ± 160 mV	5 A	10 mV	J1
H7441 (No. 2)	+5 Vdc ± 50 mV	32 A	100 mV	J4
H7476 (No. 3)	+15 Vdc ± 750 mV	2 A	100 mV	J2
	—15 Vdc ± 750 mV	1 A	50 mV	J2
H7441 (No. 4)	+5 Vdc ± 50 mV	32 A	100 mV	J3
54-14192	AC LO/DC LO signals	N/A	N/A	J6

* Although the modules can provide the load current shown, the total output power of the 70-17121 must not exceed 300 W.

2.6.3 Power/Signal Distribution

Figure 2-60 shows the formatter power supply cabling. Alternating current from the 874-E power controller is bused to the power supply over a 2.7 m (9 ft) power cord and enters the rear of the chassis. All regulated voltages and status signals leave the power supply at the rear of the chassis through five jacks, J1—J4 and J6. Plugs on the formatter power harness mate with the jacks and carry the power/signals to the logic backplane.

The power harness terminates at the backplane with 19 vertically oriented solder connections, TP1—TP19, extending down the left side (viewed from the front). These solder connections interface with voltage planes sandwiched within the construction of the printed circuit backplane. Then the planes distribute voltage to the various formatter module connectors (slots 1—16, rows A—D).

Figure 2-61 and Table 2-15 depict the separate interconnections involved in the power/signal distribution and show which jack/pin numbers connect to each backplane test point.

Figure 2-60 TS78 Power Distribution

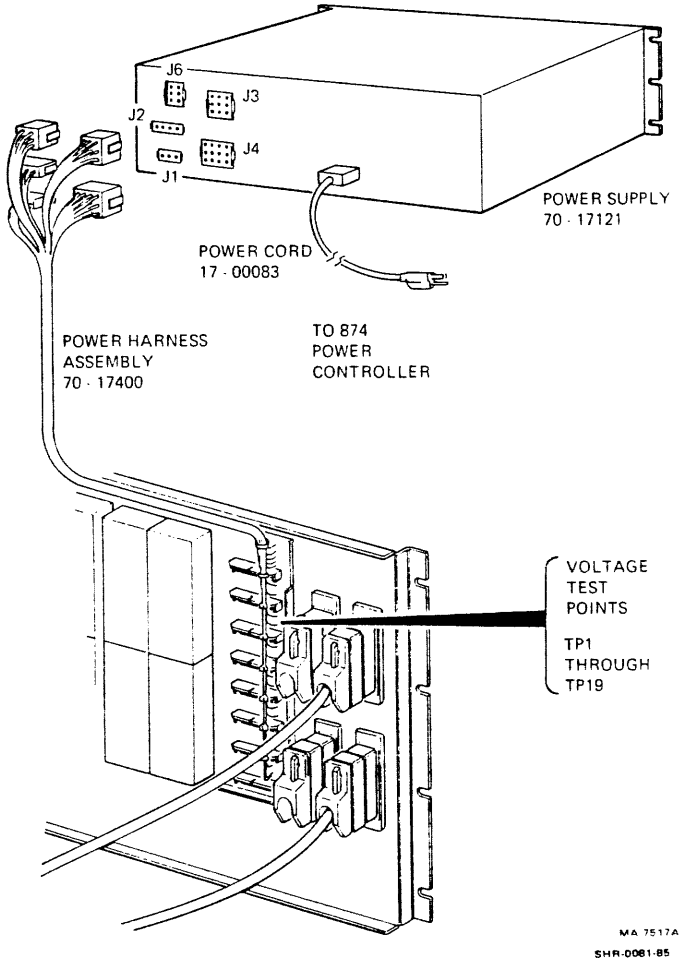


Figure 2-61 Backplane/Power Supply Interconnection

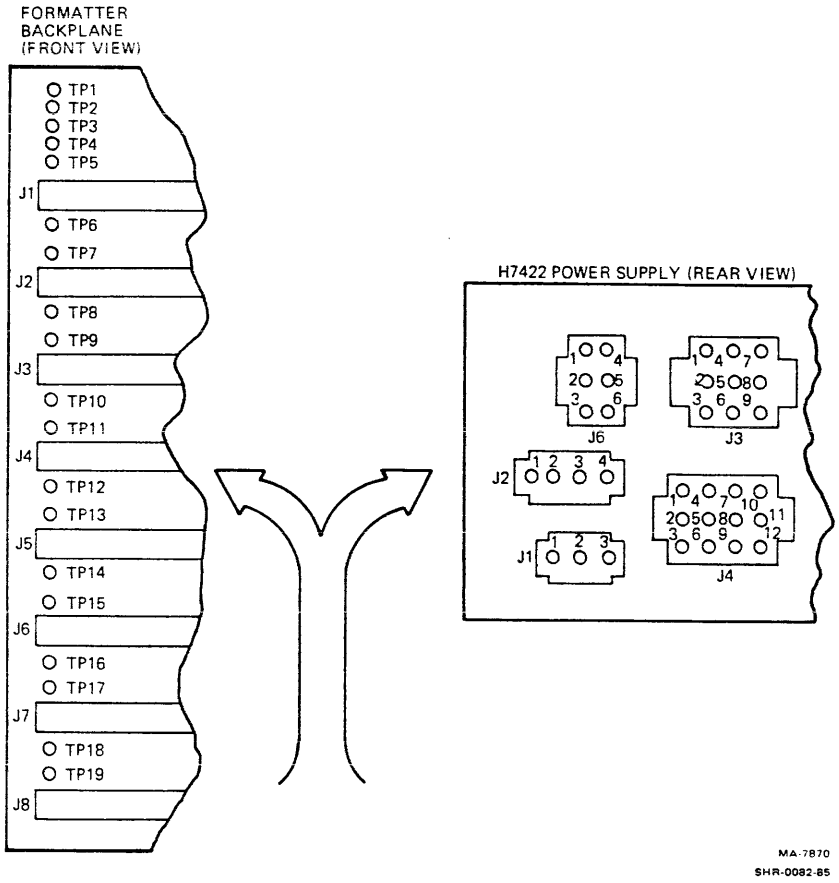


Table 2-15 Backplane/Power Supply Interconnections

Backplane Test Point	Voltage/Signal	Power Supply Jack/Pin No.
1	+15 V	J2-4
2	—15 V	J2-3
3	Ground	J2-1/J6-3
4	AC LO	J6-4
5	—5 V	J1-1
6	DC LO	J6-3
7	Ground	J1-3
8	+5 V (Regulator 4)	J3-1
9	Ground	J3-4
10	+5 V (Regulator 4)	J3-2
11	Ground	J3-8
12	+5 V (Regulator 4)	J3-5
13	Ground	J3-3
14	+5 V (Regulator 2)	J4-1
15	Ground	J4-4
16	+5 V (Regulator 2)	J4-2
17	Ground	J4-8
18	+5 V (Regulator 2)	J4-5
19	Ground	J4-3

2.6.4 Power Supply Functional Description

The following paragraphs describe the formatter power supply, including the chassis, regulators, and AC/DC LO module. Refer to Figure 2-62, the H7422-AB power supply functional block diagram, to see the relationships among these components.

2.6.4.1 H7423 Power Supply - The H7423 measures 13.3 cm (5.25 in) high by 48.3 cm (19 in) wide by 27.9 cm (11 in) deep and mounts in a rack with 48.3 cm (19 in) centers. It weighs 20.4 Kg (45 lbs). Cooling is provided by an internal fan; it forces air laterally through the supply, past the transformer, and across the regulator heat sink fins.

Refer to the bottom half of Table 2-14 for the H7423 power supply characteristics.

2.6.4.2 H7422 Power Supply Chassis - A nominal 240 Vac is supplied to the chassis through a 2.7 m (9 ft) detachable power cord. The voltage is filtered and applied to a terminal board (TB1) through a 10 A circuit breaker (CB1). The terminal board distributes line voltage to the AC ON indicator (I1) and the primary windings of transformer assembly T1. The two transformer primary windings connect in series with the addition of a jumper wire between terminals 2 and 3 of TB1. One half of the line voltage comes from this formed center tap in the primary and delivered to the 120 Vac cooling fan assembly.

The transformer steps down the line voltage and supplies it to the regulators through four secondary windings. Figure 2-62 shows the nominal voltage output for each secondary. Voltage from each secondary is brought out to jack J5. An internal wiring harness distributes this voltage from mating plug P5 to regulator plugs P1—P4. The same wiring harness conducts the regulated dc voltages from plugs P1—P4 to jacks J1—J4. Jacks J1—J4 make the regulated voltages available externally at the rear of the chassis. Here another wiring harness conducts the regulated voltages to the formatter.

H7441 Regulator Assembly - Two H7441 regulators are used in positions 2 and 4 of the 70-17121 power supply. Each provides +5 Vdc output. Figure 2-63 shows a block diagram of the H7441 regulator.

The H7441 is a switching regulator using a fixed frequency, variable pulse width modulator for control. The ac input (from the H7422 chassis) is rectified and filtered within the H7441; the switching regulator stage converts this raw dc into the desired 5 Vdc output.

AC, in the range of 20 V—30 V range enters the regulator through J1 pins 6 and 7. It is rectified and filtered to produce raw dc in the range of 25 V—40 V. The raw dc goes to a 12 V linear regulator through a thermal cutout switch. The linear regulator provides supply voltage to other H7441 components. The raw dc also provides a feed forward voltage (an anticipatory type of feedback) to a current source within the pulse width modulator (PWM) section. The amount of current produced by the current source is directly proportional to the level of raw dc voltage. Thus the duty cycle of the PWM signal is proportional, in part, to the feed forward signal that reflects the amplitude of the incoming voltage.

Figure 2-62 TS78 Power Supply Block Diagram

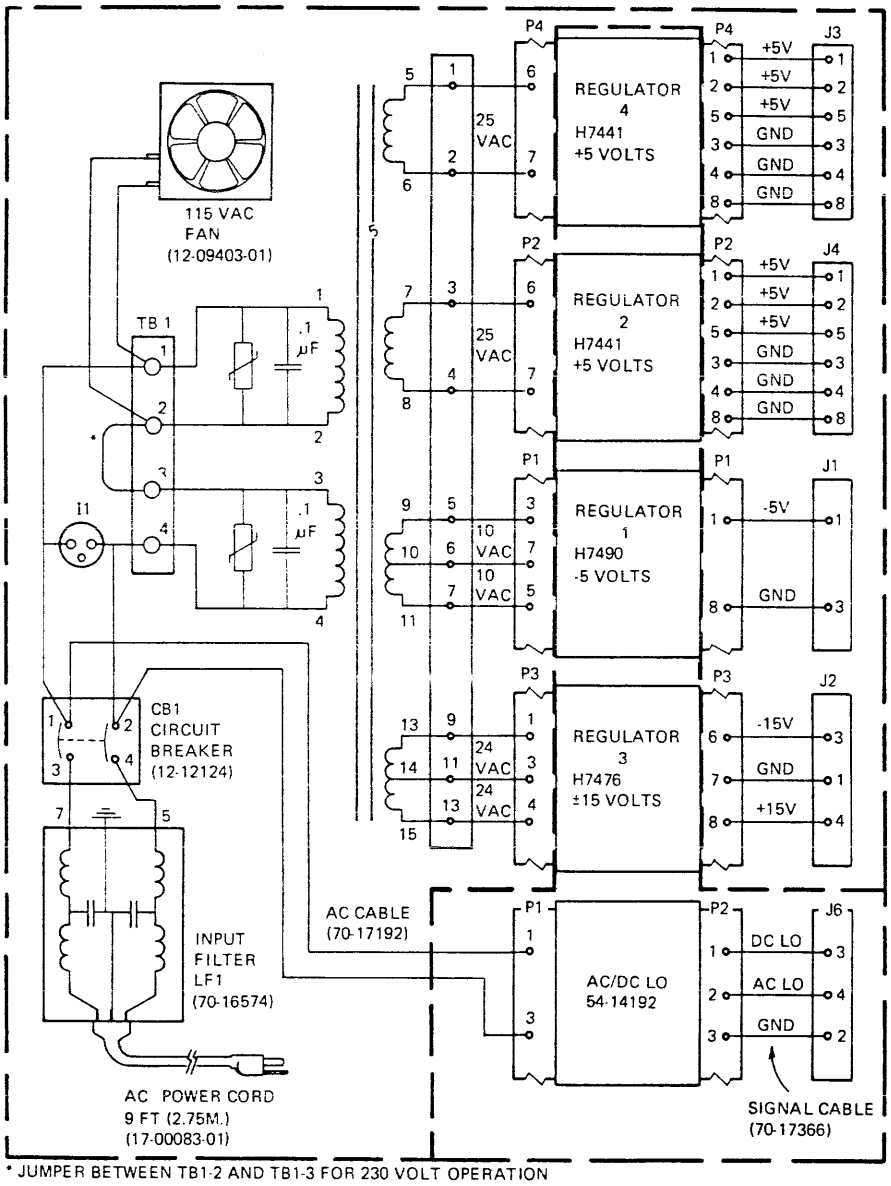
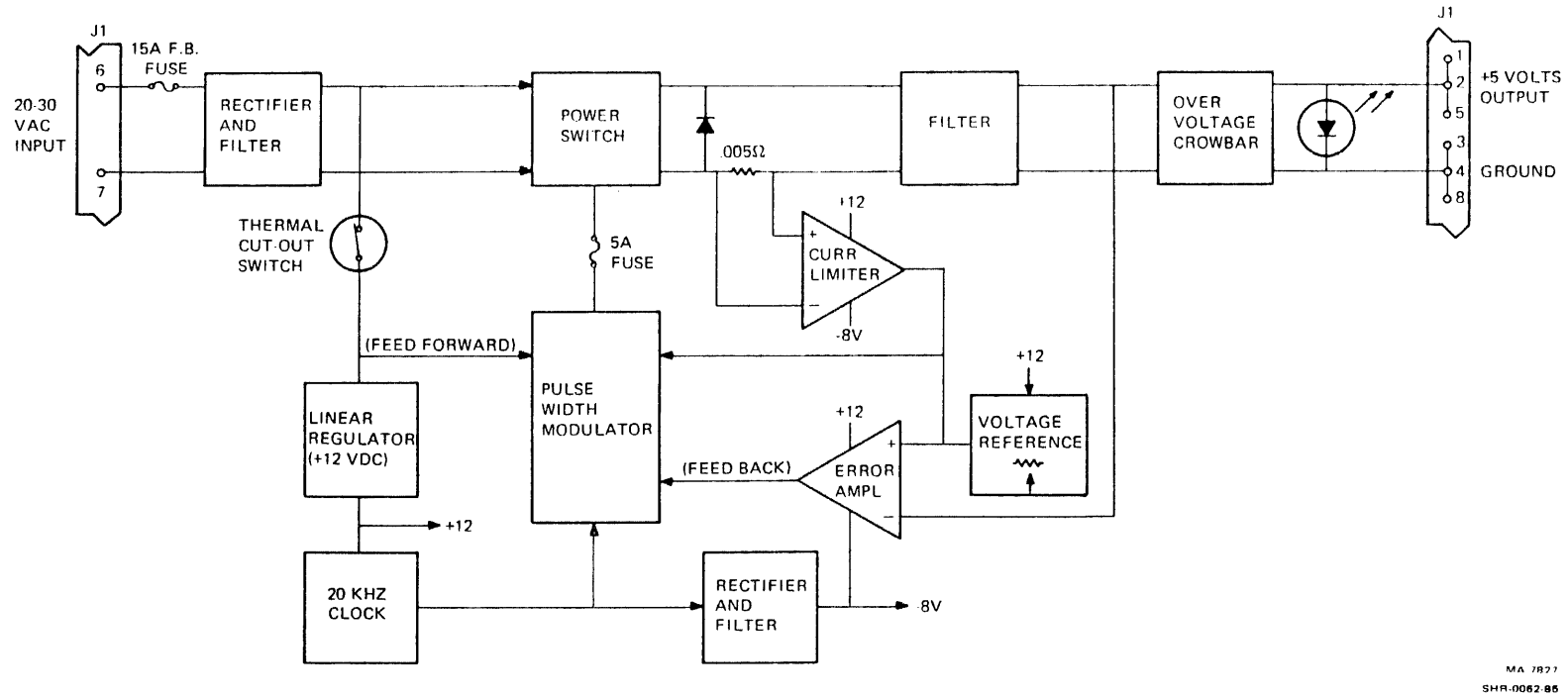


Figure 2-63 H7441 +5 Volt Regulator Block Diagram



A clock generates a 20 kHz nominal signal and provides the fundamental switching frequency for the PWM. The clock signal is also rectified and filtered to produce an 8 Vdc source for the operational amplifiers (op amps). The output of the PWM is a fixed-frequency (20 kHz) square wave with a duty cycle (the ratio of positive on state to negative off state) based upon the feed forward and feedback signals. This pulse width modulated square wave goes to the power switch, which interrupts or chops the raw dc going to the filter. So the duty cycle of the PWM square wave determines the length of time raw dc is conducted through the power switch for each clock cycle.

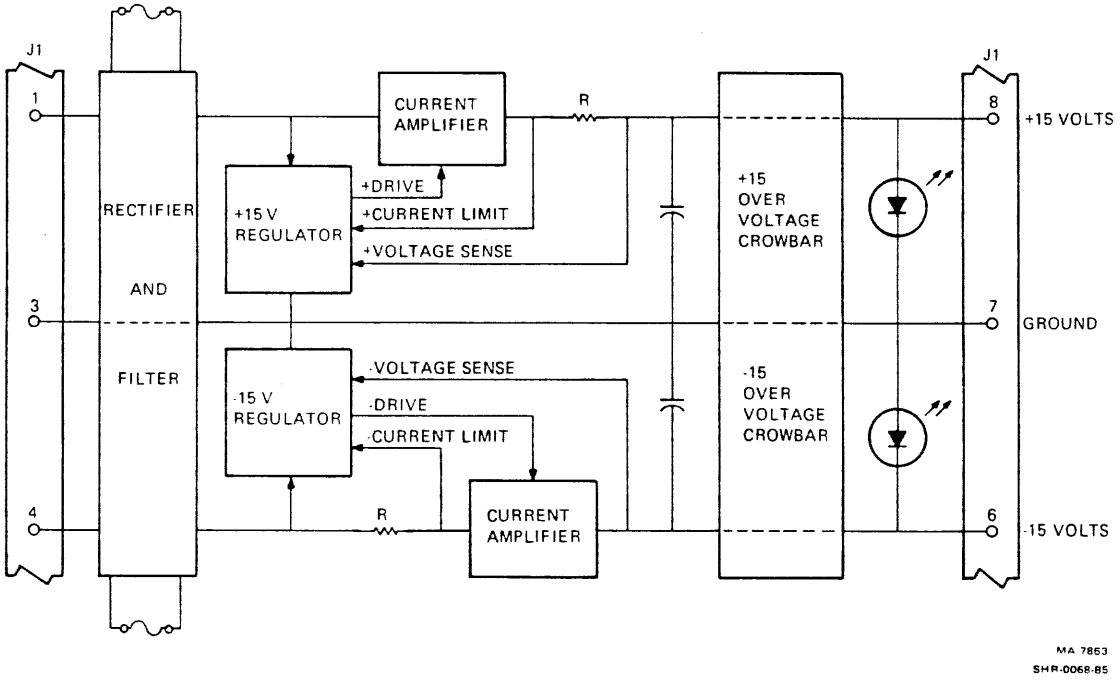
The chopped raw dc goes through the filter, where it is averaged to produce the desired +5 V output. The filter also removes the 20 kHz high frequency component. A diode at the output of the power switch provides a current path through which energy stored in the filter inductor can safely dissipate without damaging the power switch during its off time. A 0.005 ohm resistor in series with the negative line to the filter provides a voltage drop proportional to the amount of current being drawn by a load on the regulator. If the current rises beyond the allowable amount, a current-limiting op amp is turned on. The current limiter output resets the PWM and disables the feedback error amplifier.

Voltage at the filter output is sampled and compared to an adjustable reference by an error amplifier. The error amplifier output is fed back to the PWM which adjusts the square wave output accordingly. Filtered voltage passes through an over voltage crowbar network and exits the regulator through J1 pins 1, 2, 5 (+5 V), 3, 4, and 8 (ground). If the filtered voltage rises above 5.5 V due to component failure or improper adjustment, the crowbar triggers and causes +5 V to be pulled down to ground. This is a protective measure for the load device. An LED (visible through the regulator assembly cutout) will light, indicating the presence of +5 V.

H7476 Regulator Assembly - One H7476 regulator is used in position 3 of the power supply. This dual series pass linear regulator provides +15 V at 2 A, and ±15 V at 1 A. The H7476 includes a tracking regulator integrated circuit (IC) that allows the +15 V output to follow the ±15 V output. This means that should the ±15 V output change, the +15 V output will change an equal amount in the opposite direction. This balanced output, relative to ground, is necessary since it supplies voltage to the formatter sensitive phase-locked loop (PLL) read channel circuitry. Over current limiting and over voltage crowbar protection is included. Figure 2-64 shows a block diagram of the H7476 regulator.

AC in the 18 V to 26 V range enters the regulator through J1 pins 1 and 4, with pin 3 connected to the transformer center tap. The ac is rectified and filtered to produce plus and minus raw dc voltages that are fused and input to the series pass circuitry. The series pass circuitry is in a cascaded darlington configuration, and is split between the voltage regulator IC and the associated current amplifier. Voltage tracking between the two supplies is performed within the regulator IC. Current drawn by the load is reflected across a small resistance (R) and fed back to the regulator IC. The regulator IC reduces the drive current if the sample current exceeds the allowable tolerance. The voltage output to the load is fed back to the regulator IC and compared to an internal reference to maintain the correct value.

Figure 2-64 H7476 ± 15 Volt Regulator Block Diagram

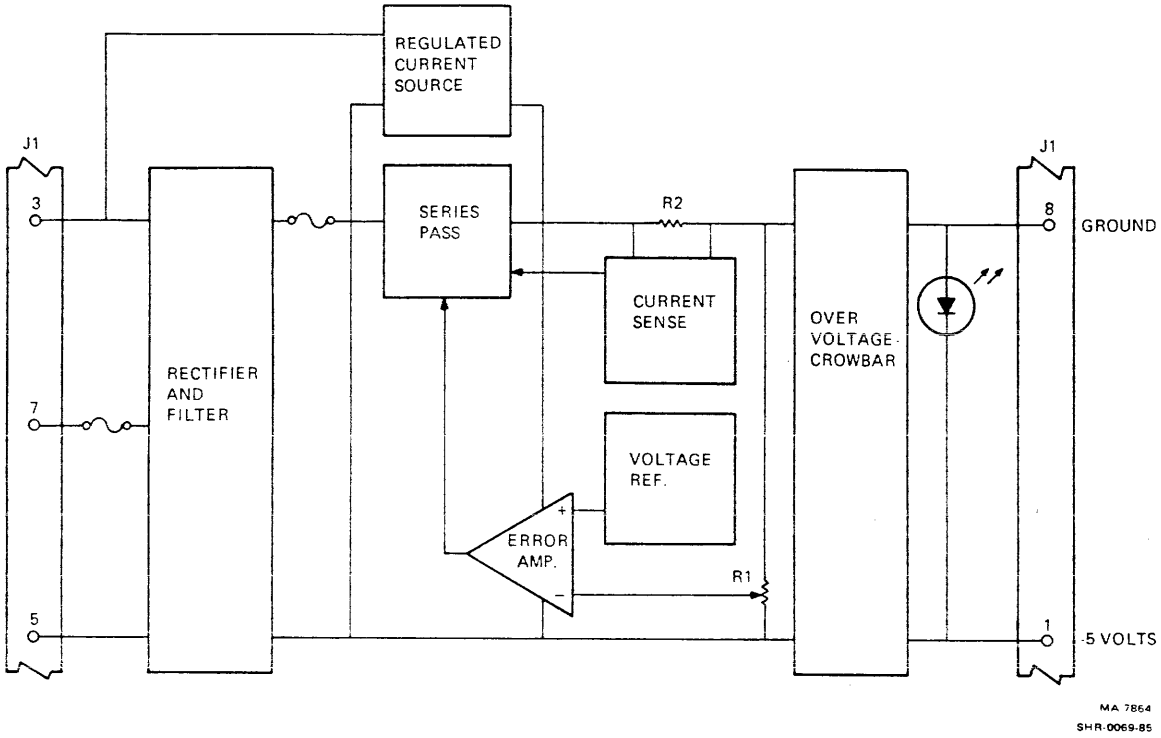


Regulated voltage passes through an over voltage crowbar network and exits the regulator through J1 pins 8 (+15 V), 6 (—15 V) and 7 (ground). If the voltage rises above 20 V, the crowbar triggers and causes the output voltage to be pulled down to ground. Two LEDs visible through a cutout in the regulator assembly will light, showing +15 V and —15 V on.

2.6.4.3 H7490 Regulator Assembly - One H7490 regulator is used in position 1 of the power supply. It is a series pass linear regulator and provides 5 Vdc output. Over current foldback and over voltage crowbar protection circuitry is included. Figure 2-65 shows a block diagram of the H7490 regulator.

AC in the range of 15 V to 21 V enters the regulator through J1 pins 3 and 5, with pin 7 connected to the transformer center tap. The ac rectified and filtered to produce raw dc that is fused and input to the series pass stage. A regulated current source is driven by one leg of the transformer and provides power to the feedback error amplifier.

Figure 2-65 H7490 —5 Volt Regulator Block Diagram



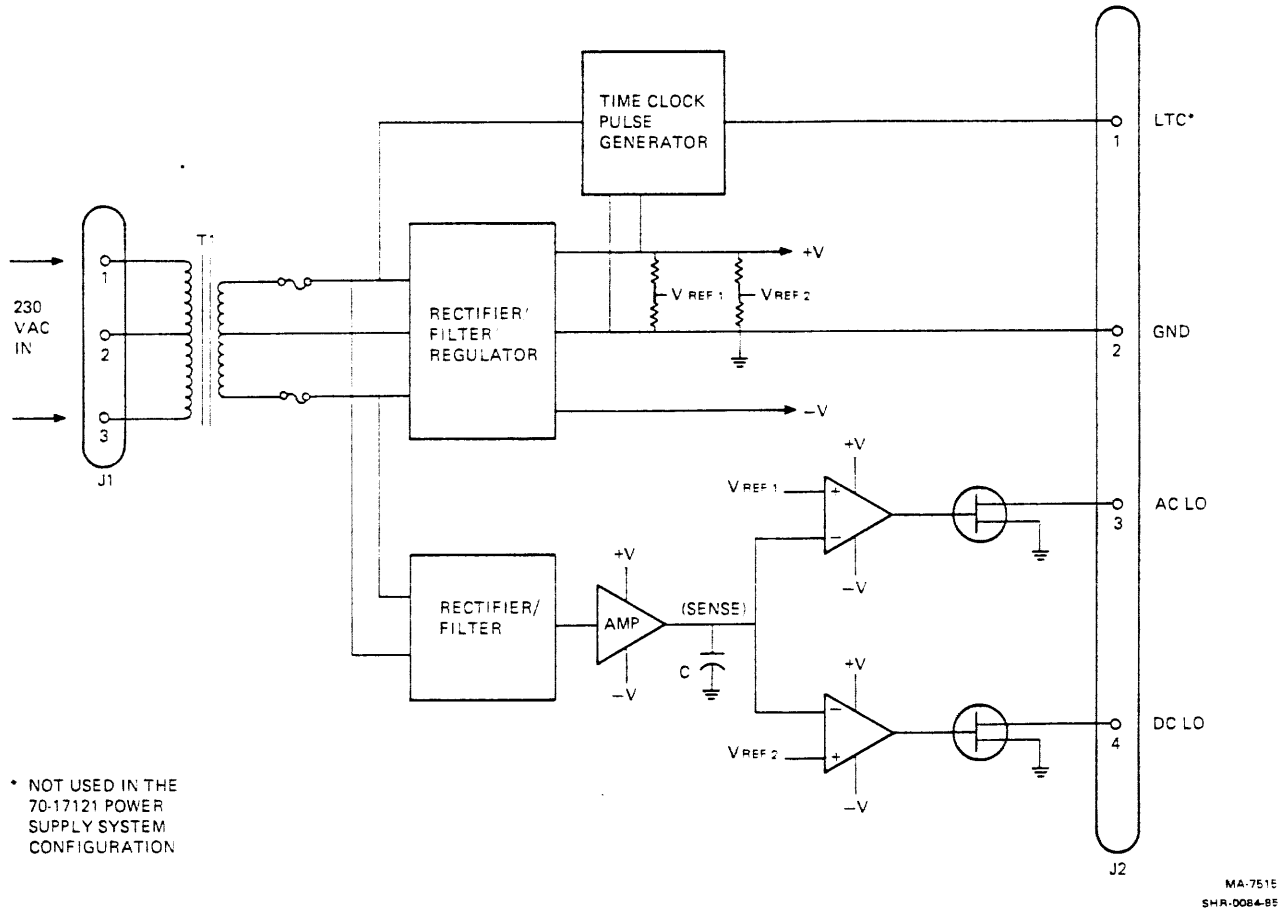
The output of the series pass stage is sampled through R1 and compared to a reference voltage by the error amplifier. The output of the error amplifier provides the feedback input to the series pass stage, thus completing the loop. Resistor R2 samples current drawn by the load. If the current rises beyond the allowable amount, a current sensing stage reduces the flow through the series pass stage.

Regulated voltage passes through an over voltage crowbar network and exits the regulator through J1 pins 1 (5 V) and 8 (ground). If the voltage rises above 6.5 V, the crowbar triggers and causes the output voltage to be pulled down to ground. An LED visible through a cutout in the regulator assembly will light, showing 5 V on.

2.6.4.4 54-14192 AC/DC LO Module - The 54-14192 module provides the AC LO and DC LO status signals to the formatter. It also provides the LTC time clock signal. LTC is not used in the power supply configuration, however.

Refer to Figure 2-66, the AC/DC LO module simplified block diagram. Line voltage enters the module through J1 pins 1 and 3 and is stepped down through transformer T1. The stepped-down ac voltage is rectified, filtered, and regulated to produce internal positive and negative dc supply voltages, +V and —V. These supply voltages provide power to the op amp/comparator logic. The +V supply also powers the time clock generator and is resistor-divided to produce two reference voltages, Vref1 and Vref2.

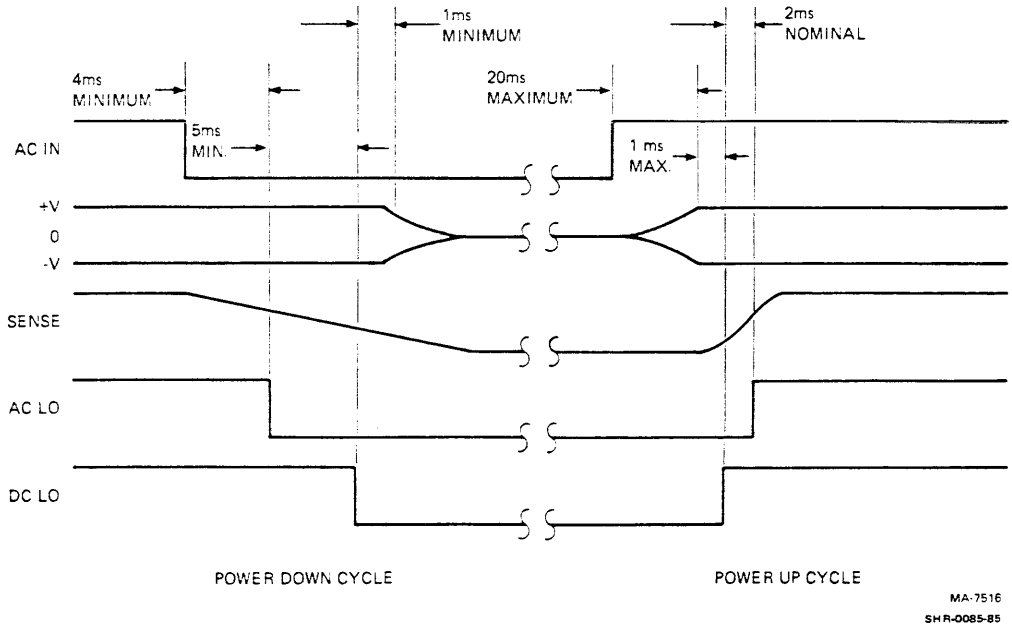
Figure 2-66 AC/DC LO Module Simplified Block Diagram



The stepped-down ac voltage is also rectified and filtered to provide a low-level signal, which indicates the amplitude of the line voltage. This low-level signal is amplified and applied across a capacitor (C) to provide a ramped "sense" voltage. This sense voltage is input to the comparators, where it is compared with Vref1 and Vref2. The reference voltages are chosen such that if the ac input (and thus the sense voltage) starts to decrease due to a power failure or shutdown, the AC LO comparator turns on before the DC LO comparator. As power is restored, the DC LO comparator turns off before the AC LO comparator. Each comparator drives the gate of a field effect transfer (FET), which floats or grounds the AC LO or DC LO signal accordingly. Refer to Figure 2-67 for AC LO and DC LO timing during power up and power down.

NOTE
AC LO and DC LO indicate the status of the ac line voltage. A failure of the H7441, H7476, or H7490 regulators will not generate an AC LO or DC LO signal.

Figure 2-67 AC/DC LO Power Up, Power Down Cycle



2.6.5 54-14174 Maintenance Panel

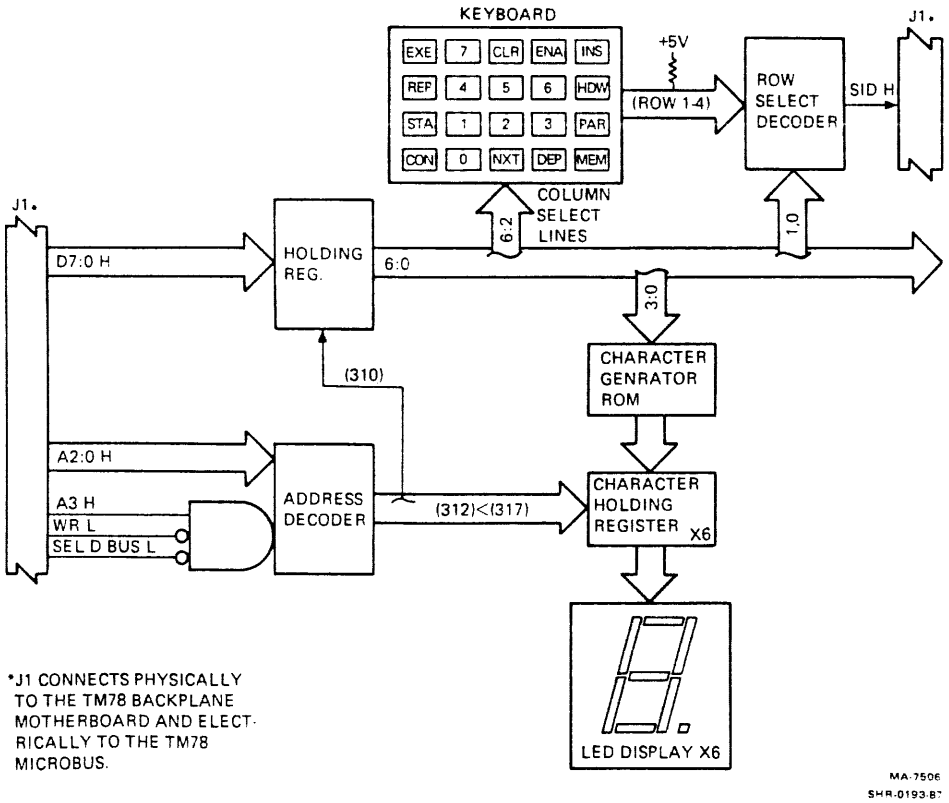
The TM78 maintenance panel provides maintenance personnel access to the TM78 system operational microcode functions, hardware registers, and diagnostics. Many of its functions can operate while the TM78 is on-line and performing tape operations for a host CPU. Additional maintenance functions are provided when the TM78 is switched off-line from the host CPU.

The panel comprises a 20-key keypad and a 6-digit LED display (Figure 2-68). A dual-ported TM78 is considered off-line only when both Massbus interfaces are switched off-line.

The keypad has both numeric and control keys. A command is initiated by pressing the desired control key. Numeric arguments are entered by pressing a sequence of numeric keys before pressing the control key. The display indicates a number being entered and presents status information resulting from a command.

When the TM78 is powered on or reset the keypad is disabled and the display is blank. This is the normal TM78 operating mode. While the keypad is disabled, the operational microcode looks at only one key, the enable (ENA) key. Pressing any key other than ENA causes nothing to happen. Pressing the ENA key enables the keypad and causes the microcode to begin looking at all the keys. Each key is examined once per pass through the microcomputer's idle loop. If the ENA key is pressed while the keypad is enabled, the keypad again becomes disabled. Using the keypad causes a small degradation in response to commands issued by the host CPU. Therefore, for improved system throughput, the keypad should always be disabled when not in use.

Figure 2-68 TM78 Maintenance Panel Block Diagram



Operation

Figure 2-68 shows the block diagram of the TM78 maintenance panel logic. Refer to it while reading the following paragraphs. Refer also to Appendix B, in the range of C8W—CFW (310_8-W—317_8-W).

The maintenance panel connects to the TM78 microbus and is a write-only peripheral to the microcomputer. The address decoder responds to all microbus addresses in the maintenance panel range and outputs a single line for each address. Address C8 (310_8-W) goes to the holding register, where it latches the data on the microbus D lines. Address C9 (311_8) is not used. The remaining six addresses (312_8-W—317_8-W) latch a preselected character into the character holding registers. This is displayed in the LEDs.

The data accompanying a write-to-address C8 (310) may select a key to be examined or a character to be displayed. Data lines 6:2 independently select one of five vertical key columns with a low true enable. Data lines 1 and 0 are a 2-bit binary field that enables the row select decoder to examine one of four horizontal key rows. The four keyboard row outputs are normally held to a logic HI. Thus, if the key under examination is pressed, for example the ENA key (examined with a 3C (074_8), the low on column select line 6 pulls row 1 output low and the row select decoder output SID H goes low. Now the microcomputer examines the SID line; if low, it "presumes" that someone is pressing the ENA key.

Data lines 3:0 specify a character code and input to the character generator ROM. The ROM produces an output that enables the elements of the 7-segment LED to produce the desired character. The character ROM output is latched in one of six holding registers, depending upon the LED selected. Microbus address CA (312_8-W) corresponds to LED A (the least significant digit), and address CF (317_8-W) corresponds to LED F (the most significant digit).

CHAPTER 3
TAPE DRIVE, THEORY OF OPERATION

3.1 GENERAL

This chapter describes basic functioning of the magnetic tape transport and detailed circuit operation of the printed circuit board assemblies (PCBAs). Paragraph 3.2 identifies and briefly describes seven functional areas in the tape drive. Subsequent paragraphs discuss each functional area separately, with associated PCBA(s) discussed in detail. The text is supported by schematics, simplified detail diagrams, and other conventional illustrations.

3.2 FUNCTIONAL DESCRIPTION

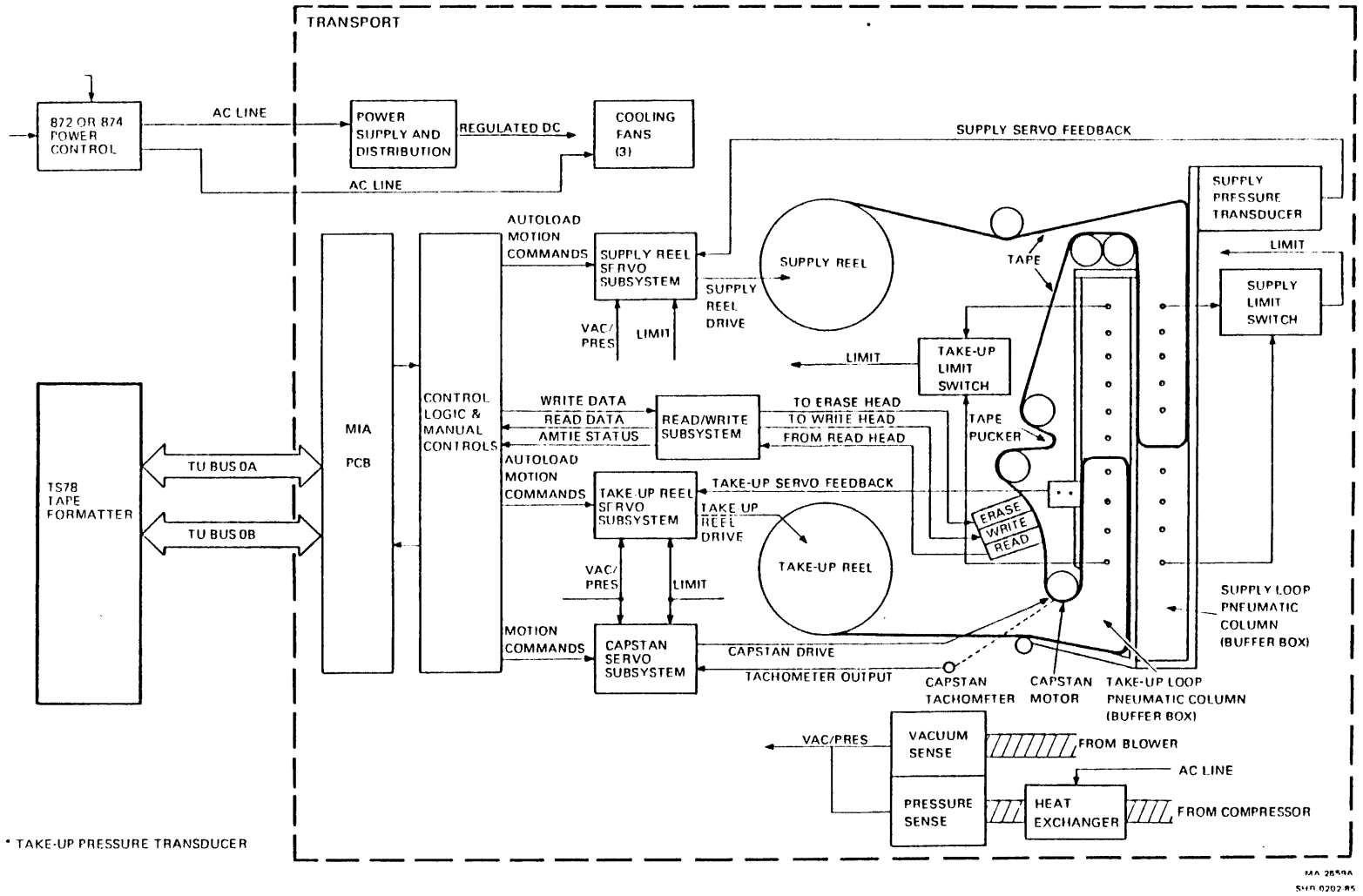
The basic transport contains the following seven functional areas. (Refer to Figure 3-1.)

- Capstan servo subsystem
- Reel servo subsystem (2)
- Pneumatic subsystem
- Read/write subsystem
- Control logic and manual controls
- MIA interface
- Power supply and distribution

3.2.1 Capstan Servo Subsystem

The capstan servo subsystem controls the speed and direction of tape movement past the read/write heads. The subsystem is a velocity servo that receives command signals from the control logic, specifying forward, reverse, or rewind motion. The capstan motor responds with the appropriate velocity. The capstan tachometer generates a feedback signal proportional to speed. This signal is summed with the basic command signal to maintain the correct capstan velocity at all times.

Figure 3-1 Tape Drive Block Diagram



3.2.2 Reel Servo Subsystems

The reel servo subsystems control the speed of the tape reels to maintain optimum tape tension between the supply and take-up reels. The supply reel and take-up reel servos are similar but separate subsystems. The path followed by the tape (in either direction) between the supply and take-up reels contains two tape loops in the buffer box (supply loop and take-up loop). The separately formed loops are maintained by a vacuum in conjunction with automatically controlled reel motor speeds. In effect, the reel servos function to feed tape into and remove tape from the buffer box at the rate needed to maintain the correct loops.

3.2.3 Pneumatic Subsystem

Signals developed within the pneumatic subsystem initiate servo operation. The subsystem senses that the tape loop position has changed as a result of forward or reverse tape motion. Air is drawn from the closed ends of the two buffer boxes, creating a vacuum and causing the tape loop to form in each box. The differential between the positive pressure inside the loop and the relatively negative pressure at the closed end of the buffer box (outside the loop) maintains the proper tension on the tape during the tape-loaded state.

There is a separate chamber behind each buffer box, connected to the box by a series of holes. These holes are spaced and arranged in such a way that, if the loop becomes larger, more holes are exposed to the positive (atmospheric) pressure inside the loop, and fewer to the lower pressure (vacuum) area outside the loop. This causes pressure in the chamber to rise. Conversely, if the loop becomes smaller, the pressure in the chamber decreases.

Pressure transducers are connected to the supply and take-up chambers. They interpret pressure variations to provide the supply and take-up servo feedback signals. The pressure-sensitive signals feed back to the reel servos to adjust the velocity of the reel motors for the proper loop in the two buffer boxes. The uppermost and lowermost holes in each buffer column are limit ports. These connect to supply and take-up limit switches that feed back to both the supply and take-up servos. If the tape crosses a limit port in either the supply or take-up columns, a disabling signal couples back to the servos, stopping both reel motors before tape damage occurs.

A pneumatic interlock shuts down the capstan servo and reel servos if a pneumatic failure is detected. The pneumatic subsystem contains a blower, which creates the vacuum for the tape columns, and a compressor, which generates pressure for the tape path bearings. Sensing devices monitor vacuum and pressure. If either is lost, the sensing device sends a VAC/PRES signal to the three servo subsystems, stopping the servo motors before tape damage occurs.

3.2.4 Read/Write Subsystem

The read/write subsystem processes data and transfers it to and from the magnetic tape. The read function processes data picked up from the tape by the read heads. It translates information from the recorded PE or GCR format to digital data acceptable to the external controlling circuits. The function includes the read-after-write capability that permits the formatter to verify execution of write command while writing is in progress. The write function prepares incoming data for recording in PE or GCR format and writes the information on the tape in the selected format. The read/write subsystem also develops amplitude track in error (AMTIE) status signals. It sends nine AMTIE signals to the formatter, one for each track. The signals become active when the strength of the analog signal read from tape falls below a specified threshold. During a write function, this causes the operation to be retried. During a read function, AMTIE signals help the formatter develop pointers for the error correction process.

3.2.5 Control Logic and Manual Controls

The control logic and manual control circuit interfaces other tape subsystems. The control logic transfers read/write data to and from the read/write subsystem. It also transfers the operational commands to the capstan servo. During the autoload sequence, logic circuits control the sequence steps by issuing appropriate commands to the reel servos. The control logic monitors and controls timing of the autoload sequence steps and other operational sequences, such as rewind. It processes commands generated by the manual controls and applies them to the appropriate subsystem. Control logic also senses transport status (for example, transport selected, on-line, EOT, BOT, etc.), and modifies signals to the read/write and servo subsystems. Logic lights the appropriate control panel indicators to indicate transport status. Transport status is sent to the formatter by means of the MIA interface module.

3.2.6 MIA Interface

The MIA module couples control/status, read/write, and AMTIE signals from the tape unit bus (TU bus) to the transport. The MIA adapts signals on the TU bus to meet the transport needs, and vice versa. This includes multiplexing, de-multiplexing, signal gating, latching, and timing.

3.2.7 Power Supply and Distribution

Power supply functions include ac rectification, filtering, dc regulation, and distribution of power to the various subsystems.

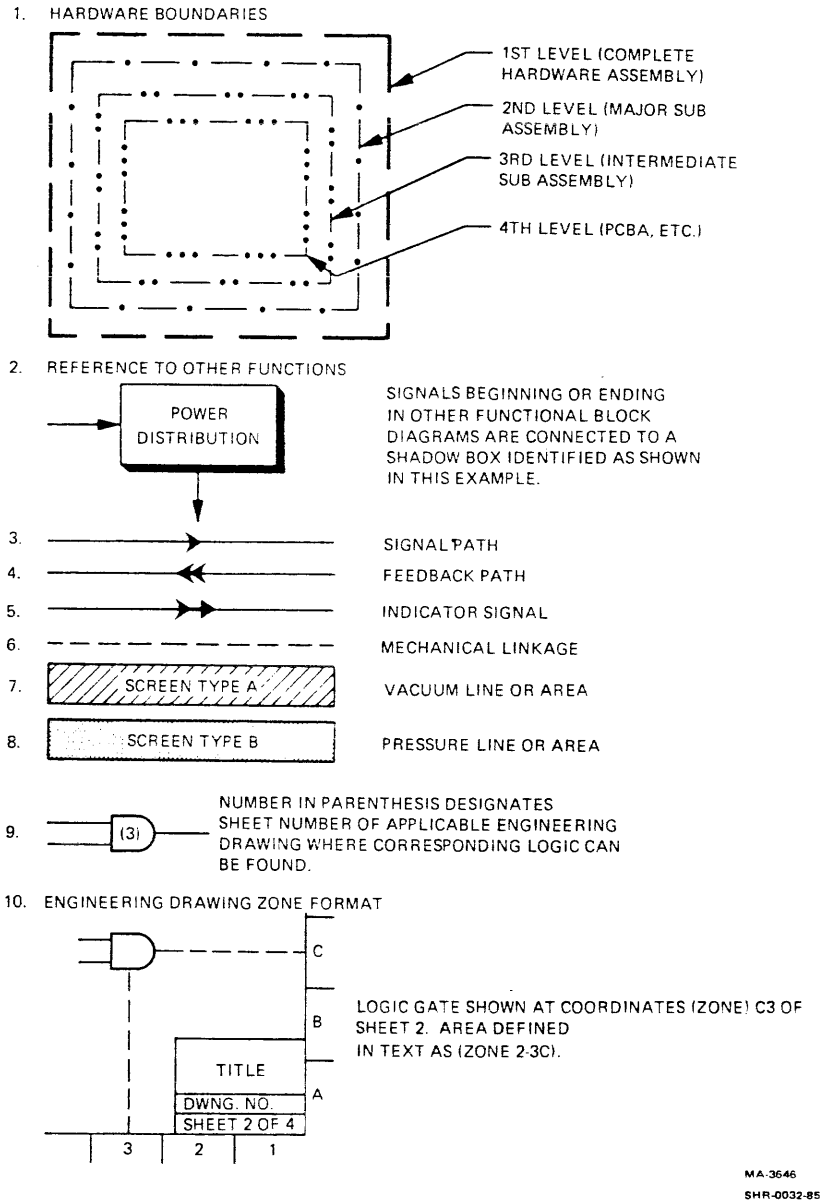
3.3 THEORY OF OPERATION SYMBOLOGY

Specific symbols and mnemonic standards are used in the text and illustrations in this chapter. An exception is the paragraph on the multiple interface adapter (MIA) interface module. The following explanations refer to other tape transport areas.

3.3.1 Functional Block Diagram Symbology

Symbols used in the functional block diagrams are shown in Figure 3-2. Parts of signal paths confined to hardware assemblies are placed within boundaries that indicate the subassembly level. One level of hardware (such as a PCBA) is included within boundaries of the next level assembly (such as the card cage).

Figure 3-2 Functional Block Diagram Symbology



The block diagrams show the purpose of the modules, cards, assemblies, or components involved in the overall operation. Active components in each block are shown to link the block diagram to the schematic. Interconnecting wires, plugs, jacks, terminal boards, adjustments, controls, meters, and test points are shown. Signal lines are coded by special arrowheads and identified by signal flags. The weight of the signal lines show the significance of the signal in the discussion.

Hardware references are printed in the upper left corner of the area representing that hardware. Controls and control nomenclature visible when the equipment is mounted and operating under normal conditions (dust covers, doors closed, etc.) are considered front panel controls. The diagram shows front panel controls in a line art window in the function and hardware on which they are located.

Signals generated in one function, and used in another, are interfaced by terminating the signal in a shadow box area representing the other function using the signal. Since all functions are in hardware, all pins of plugs and jacks through which the signal passes are shown. This provides an easy method of tracing signals from one functional block diagram to another. Signal flags (mnemonic terms) help the user find the desired signal. Mnemonic terms and abbreviations are defined in the glossary.

The text provides a description of the functional block's operation within the overall function. The text is written to establish how one portion of the function interplays with other portions of the function.

3.3.2 Schematic Diagram Symbology

Interface voltage levels between the MIA and the transport (at interconnect D1) are as follows.

Low (true) = 0 V
High (false) = +3 V

A true signal from the MIA is 0 V (nominal) at the input to the transport's receiver circuits. Similarly, a true signal from the transport to the MIA is 0 V (nominal) at the output of the transport's driver circuits. Therefore, at interface low = true, and a mnemonic term with a prefix I for interface is always interpreted as low = true. This applies whether the interface signal is an input or output.

At other points in a circuit, a true signal may be low. For example, ISIGNAL applied to a NOR gate or inverter produces SIG, which is high = true. If SIG is similarly inverted, it produces NSIG, which is low = true. As far as voltage levels are concerned, NSIG is the same as ISIG. ISIG is known to be low = true without the N because the I indicates an interface signal, which is always low = true.

An interface signal that appears to be irregular in regard to voltage level is the high/low density select signal. This is because the signal is essentially true in either high or low state, depending on whether high or low density is selected. To understand the logic more easily, consider this to be a high density select signal. It is true (high density mode is selected) when the voltage level is low. If the voltage level is high, the high density signal is false and the system operates in low density mode.

Standard symbols are used in all schematics, logic drawings, and so on. Mnemonic terms are defined in Appendix A. The alphabetic character I is used as the first letter of a mnemonic term to indicate an interface signal. When needed for clarity, a D (driver) or an R (receiver) is added at the end of the expression to indicate output and input signals, respectively.

An N is used in the beginning of a mnemonic term to indicate a NOT (low voltage level) state. In some documentation this is expressed by the overline or bar symbology (for example, NSIGNAL A = NOT SIGNAL A = SIGNAL A). The N pertains only to the voltage level; it does not imply that the signal is logically false.

3.4 CABLING/INTERCONNECTIONS

3.4.1 Input/Output Cabling and Connectors

All input/output signals between the tape formatter and the tape transport are through the multiple interface adapter (MIA) PCBA (Figure 3-3). The MIA receives the TU bus from the formatter and plugs into J1, J2, and J3 on the interconnect D1 PCBA. The transport interface signals are also coupled to two other sets of connectors on interconnect D1. These are J101, J102, and J103 on the top edge of the interconnect D1 PCBA, and J201, J202, J203, and J204 on back of the interconnect D1 PCBA (Figure 3-4). Both sets of connectors are not used, but can serve as test points for the transport interface signals.

Figure 3-3 TU78 Interface Connections Control Logic Block Diagram

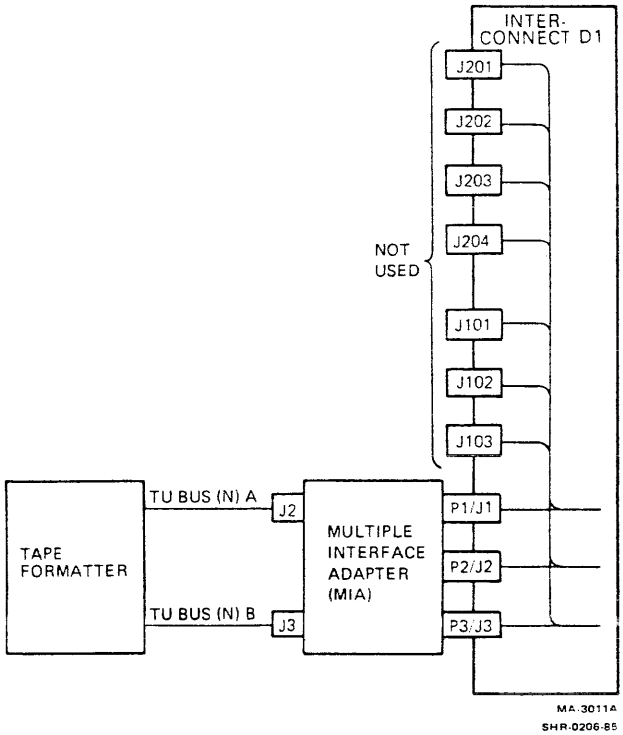
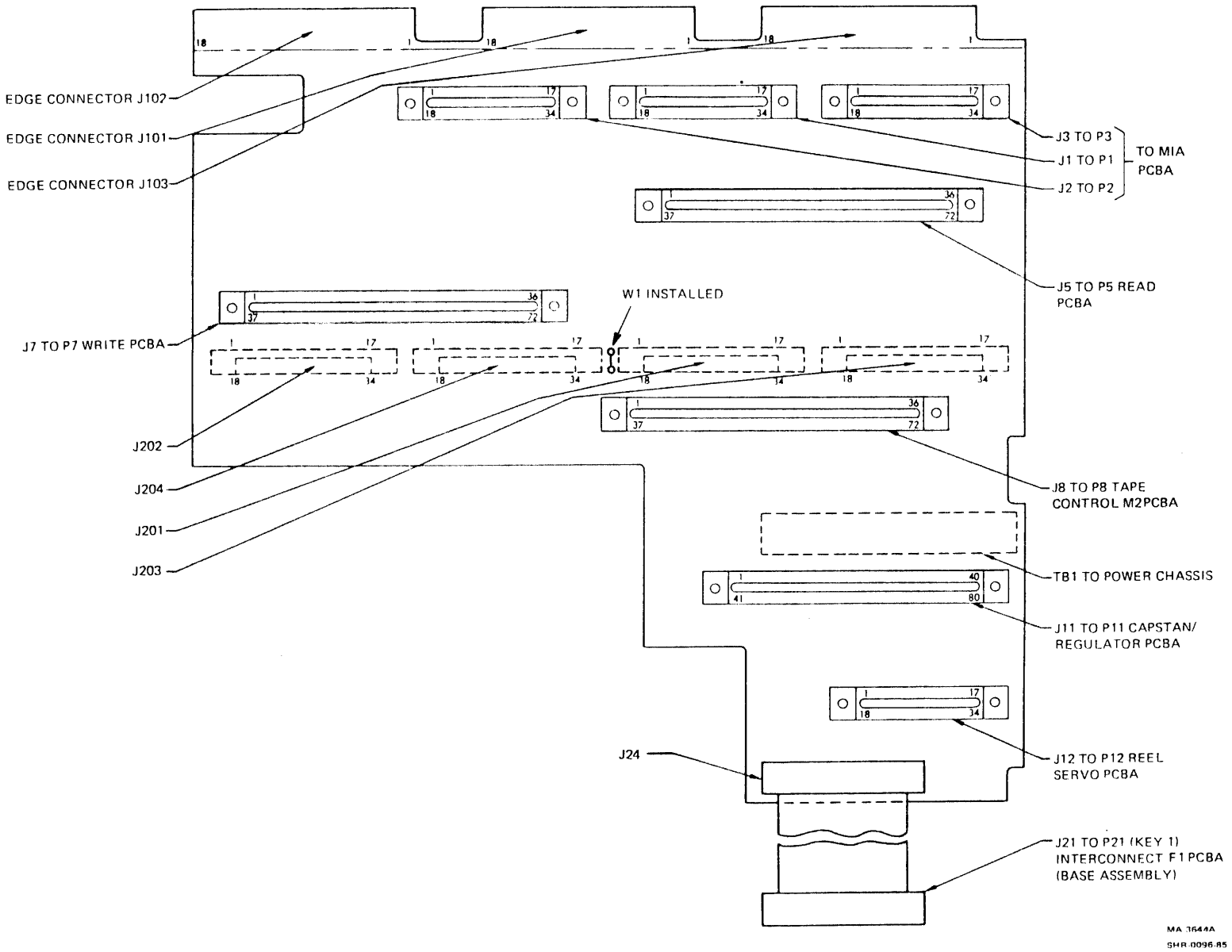


Figure 3-4 Interconnect D1 PCBA, Front View



3.4.2 Interconnect D1 PCBA

The interconnect D1 PCBA is the vertical board that provides the connections among the various logic boards in the card cage. The interconnect D1 PCBA is shown in Figure 3-4. The interconnect D1 PCBA also connects to the interconnect F1 PCBA, by means of a cable, through jack J24 (Figures 3-4 and 3-5).

3.4.3 Interconnect F1 PCBA

The interconnect F1 PCBA, mounted on the back of the transport base assembly, provides connections between the card cage logic circuits and the controls, sensors, and indicators on the base assembly. The interconnect F1 PCBA connects to the interconnect D1 PCBA, by means of a cable, through jack J21 (Figure 3-5). Terminal boards 1 through 4 allow connections to base assembly components.

3.4.4 Internal Interconnections

Connections between various functional PCBAs (control M2, write, and so on) are provided by the interconnect D1 PCBA, into which the other boards are plugged (Figure 3-4). Conductors on the interconnect D1 PCBA connect logic signals, distribute power, and route commands. The few logic connections made by cable are shown in Figure 3-5. Primary and secondary power circuit cables are described in Paragraph 3.5.

3.5 POWER SUPPLY AND DISTRIBUTION

The transport derives all power from a single ac input (shown in simplified form in Figure 3-6). Primary ac power is used for the motor which drives the blower and compressor. A transformer T1 secondary winding output provides 8.5 Vac to the power reset (NPORST) circuits. Other secondary winding outputs are rectified and delivered to the capstan/regulator PCBA, where they are regulated. Reel servo unregulated 36 Vdc power is supplied to the reel servo PCBA. Power used for the read, write, and erase heads is regulated on the GCR/PE preamp 1 PCBA.

The following paragraphs describe in detail the power supply and distribution circuits for a transport operating on 210 V 60 Hz power. For equipment using voltages and frequencies other than 210 V 60 Hz, you must determine that the input receptacle is properly connected to the primary winding taps. These connections are detailed further in the following paragraphs. For clarity, the subject matter is divided into the following topics.

- Primary power connection and controls
- Blower/compressor motor power
- AC power
- Unregulated dc power supplies
- DC power regulation
- Regulated power distribution
- Power reset enable, and master reset pulse generation.

Figure 3-5 Circuit Card Interconnections

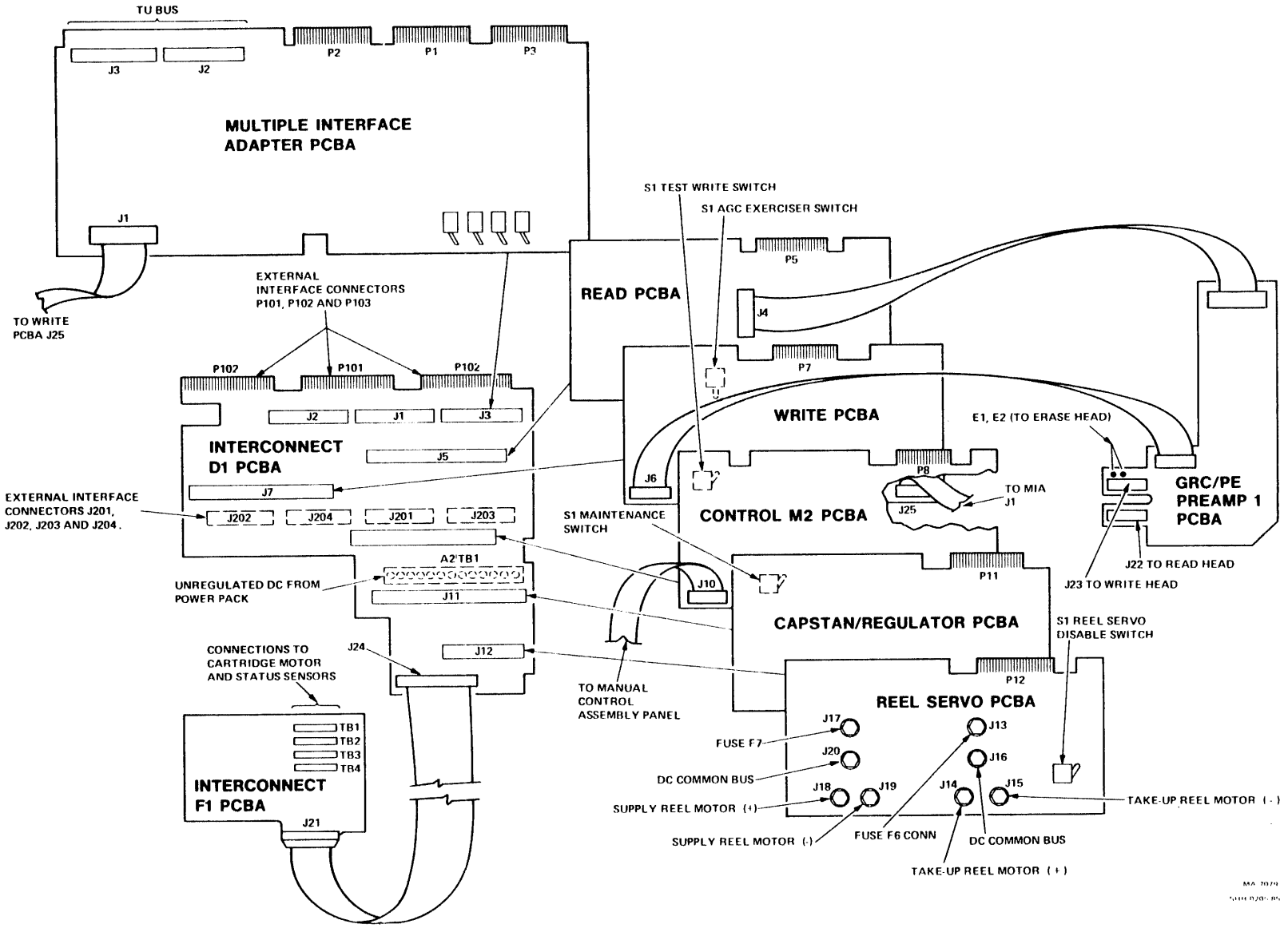
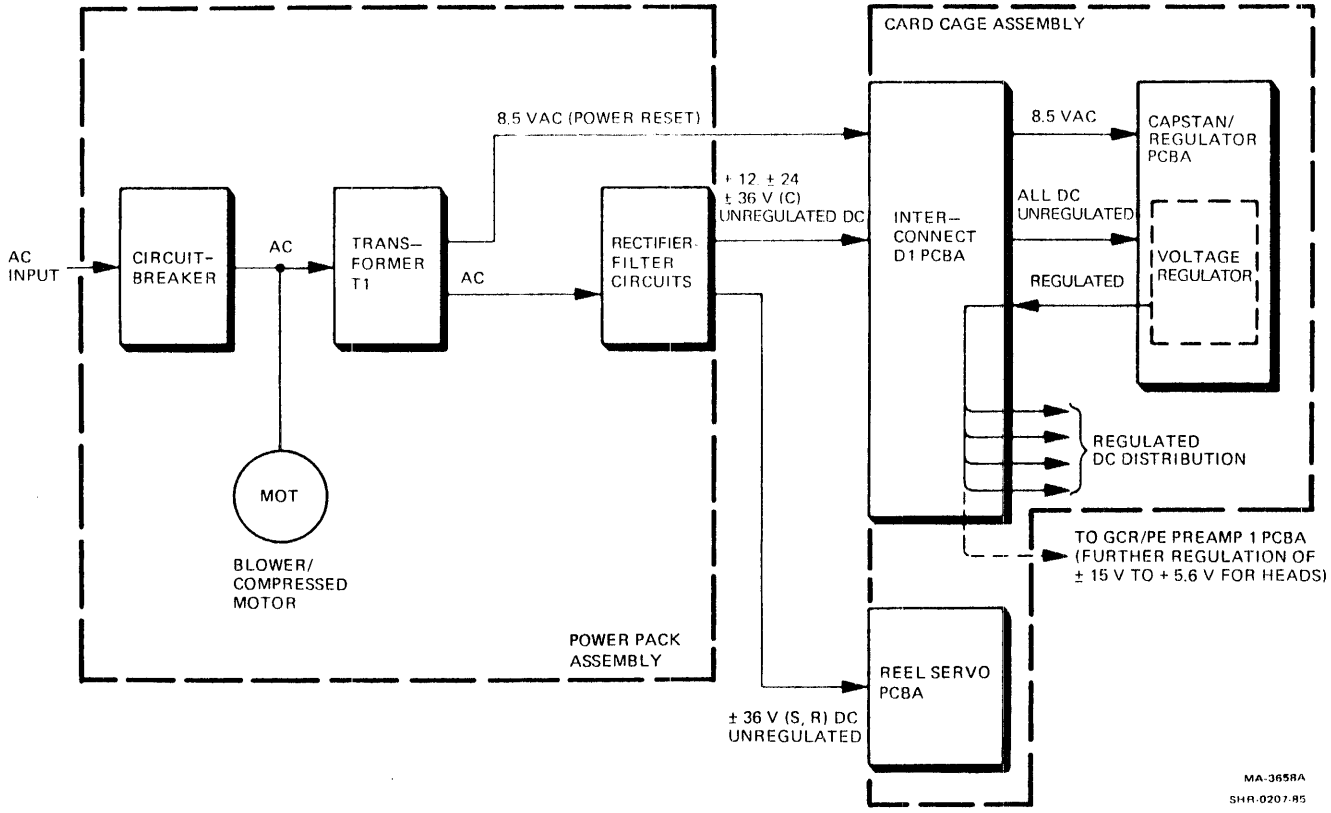


Figure 3-6 Power Supply Simplified Block Diagram



3.5.1 Primary Power Connections and Controls

The transport derives all necessary power from a single ac input. The input voltage is optional between 190 and 250 Vac, in 10 V increments, but the input must correspond to the connections between power circuit breaker CB1 and the transformer's primary winding taps. These taps are available at power supply terminal board A1TB1 (Figure 3-7). The transport can also be prepared for either 50 or 60 Hz power. This requires a different ac motor pulley and a different blower drive belt. Figure 3-7 shows that ac power is available to the transport when circuit breaker CB1 is closed.

3.5.2 Blower/Compressor Motor Power

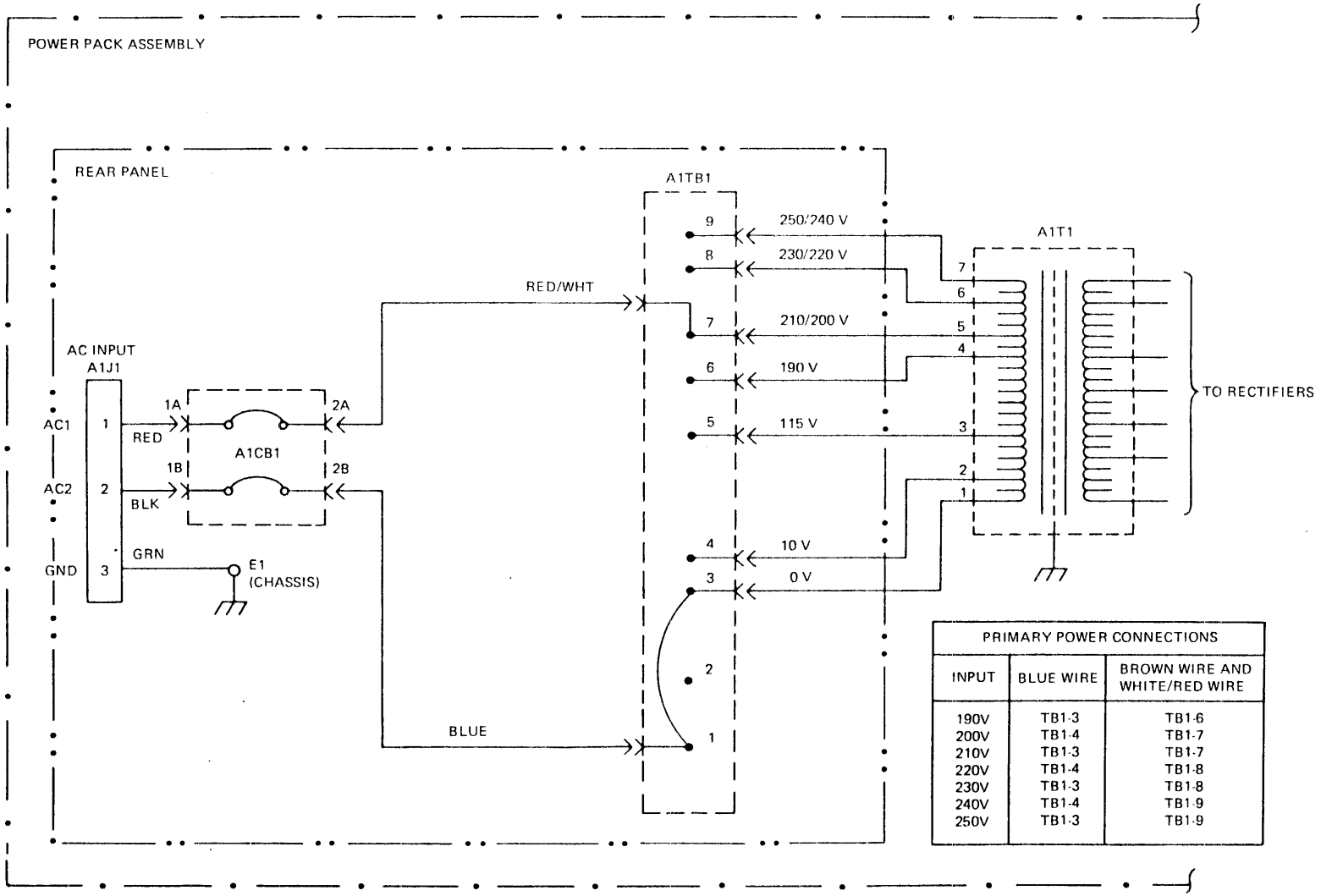
The blower/compressor ac motor operates directly on ac input shown in Figure 3-8. It is turned on automatically when solid-state switch S3 closes the circuit between A1TB1-1 and A1TB1-2. Switch S3 is controlled by +12 Vdc at S3-3 and a low (0 Vdc) pneumatic return PNU RET at S3-4. Rectifier CR2 supplies +12 Vdc when the transformer T1 secondary winding is energized. S3-4 goes low during load operation and stays low unless interlock is lost, or air or vacuum pressure is inadequate. The motor has an internal thermal switch which interrupts power to the motor windings when the temperature is too high. The switch contacts close when the motor temperature returns to normal.

3.5.3 AC Power

AC power is used in the transport to operate the blower/compressor drive motor and to initiate the power-on reset process (NPORST) by means of power supply rectifiers and power-on reset logic (Paragraph 3.5.7). The voltage applied to the motor is 210 V, as long as the input taps on the transformer match the applied voltage.

The power-on reset requirement is 8.5 Vac derived directly from the transformer T1 secondary winding. The connection to the secondary tap is physically made at the input to rectifier CR2, which is connected to A2TB1-10. A2TB1 is at the rear of the card cage and provides direct inputs to vertically mounted interconnect D1 PCBA, to which it is attached. The 8.5 Vac is delivered to pin 40 of J11 to generate the power-on reset signal (NPORST) in the capstan/regulator PCBA.

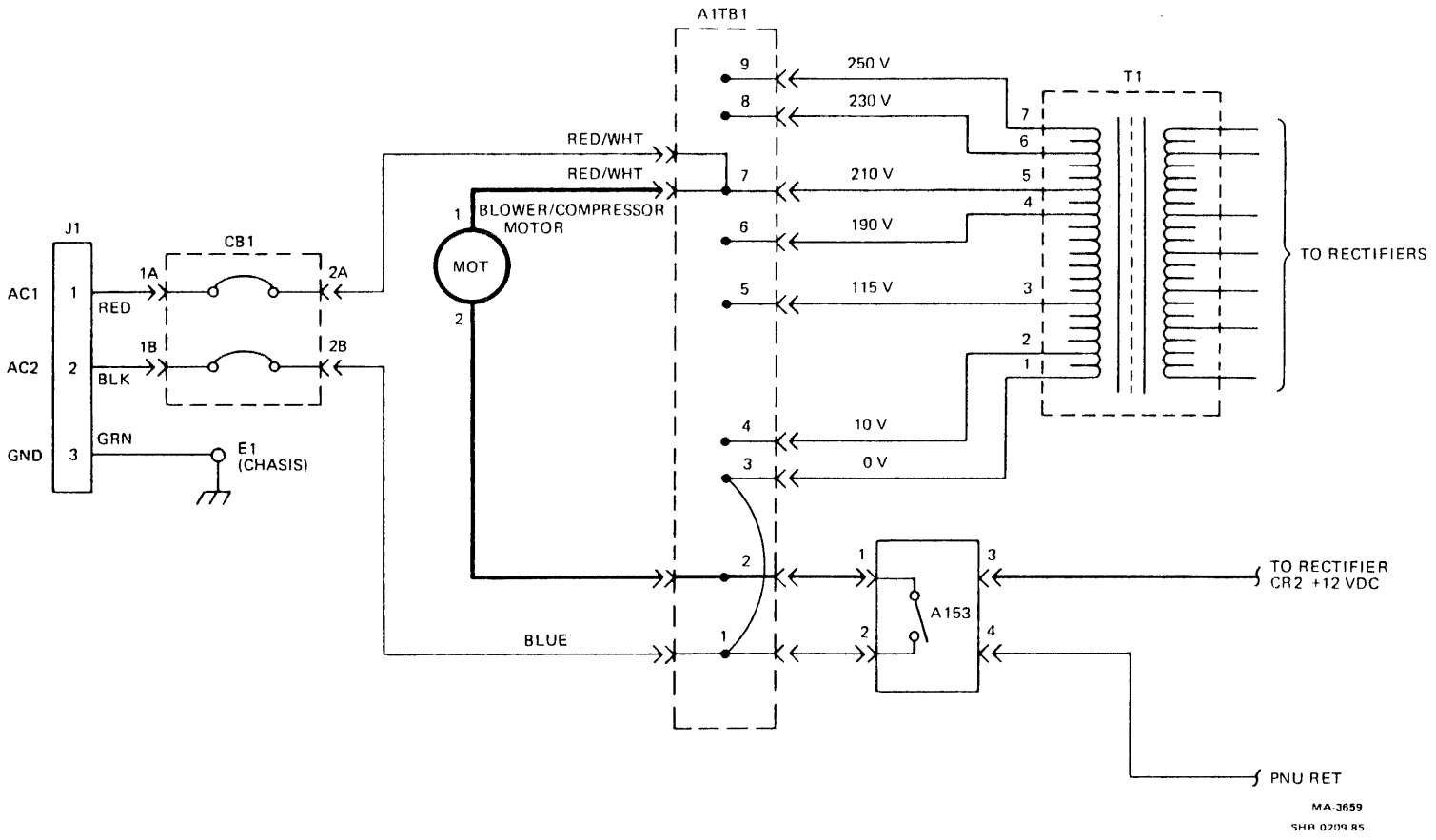
Figure 3-7 Primary Power Hookup and Control



PRIMARY POWER CONNECTIONS

INPUT	BLUE WIRE	BROWN WIRE AND WHITE/RED WIRE
190V	TB1-3	TB1-6
200V	TB1-4	TB1-7
210V	TB1-3	TB1-7
220V	TB1-4	TB1-8
230V	TB1-3	TB1-8
240V	TB1-4	TB1-9
250V	TB1-3	TB1-9

Figure 3-8 Blower/Compressor Motor Power and Control



3.5.4 Unregulated DC Power Supplies

Power for the dc circuits is derived from various taps of the transformer T1 secondary winding (Figure 3-9). These ac voltages are rectified by CR1, CR2, and CR3 to produce nominal +12 Vdc, ±24 Vdc, and ±36 Vdc unregulated voltage.

Actual rectifier outputs are listed as follows.

Nominal	Rectifier Output
+12	9.0 to 11.0 Vdc at 10 A, 2 V peak-to-peak maximum ripple at 10 A
+24	21.5 to 24.5 Vdc at 5 A, 2 V peak-to-peak maximum ripple at 5 A
—24	21.5 to 24.5 Vdc at 5 A, 2 V peak-to-peak maximum ripple at 5 A
+36	33.0 to 37.0 Vdc at 15 A, 43 Vdc maximum at no load, 2 V peak-to-peak maximum ripple at 15 A
—36	33.0 to 37.0 Vdc at 15 A, 43 Vdc maximum at no load, 2 V peak-to-peak maximum ripple at 15 A

Rectifier outputs are routed through the respective fuses to terminal A2TB1, which is attached to the rear of interconnect D1 PCBA (Figure 3-9). They are connected through interconnect D1 PCBA conductors to the capstan/regulator PCBA for regulation and further distribution. However, the PNU RET line is routed to the control M2 PCBA.

Supply reel and take-up reel +36 Vdc unregulated power is routed directly from the fuse to the reel servo PCBA. It does not go through the interconnect D1 PCBA. Similarly, —36 Vdc unregulated power is also routed to the reel servo PCBA.

All dc return lines terminate at the dc common bus mounted on the top of the capacitors. The bus is connected to the center tap of the T1 secondary winding.

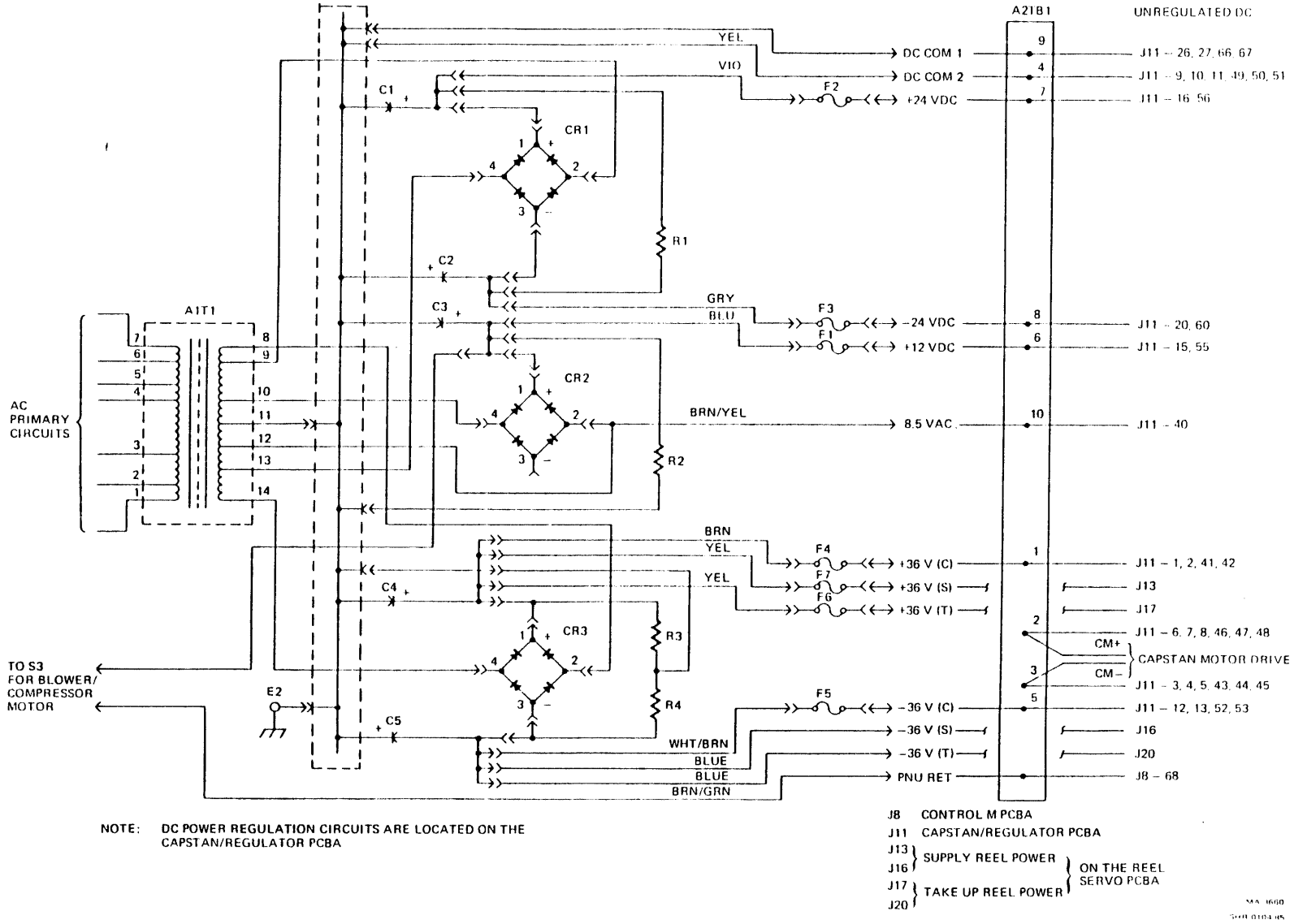
3.5.5 DC Power Regulation

The +12 and 24 Vdc unregulated power outputs are later regulated to provide the required +5, —15, and +5/—6 Vdc power (Figure 3-10). This is described in the following paragraphs.

3.5.5.1 +5 Vdc Regulator - The +12 Vdc unregulated voltage is received by the +5 Vdc regulator on the capstan/regulator PCBA. The +5 Vdc regulator uses a type LM305-IC to control the series power transistor and has an 8 A full load capacity. TP11 (on capstan/regulator PCBA) is used to monitor the regulated +5 Vdc output.

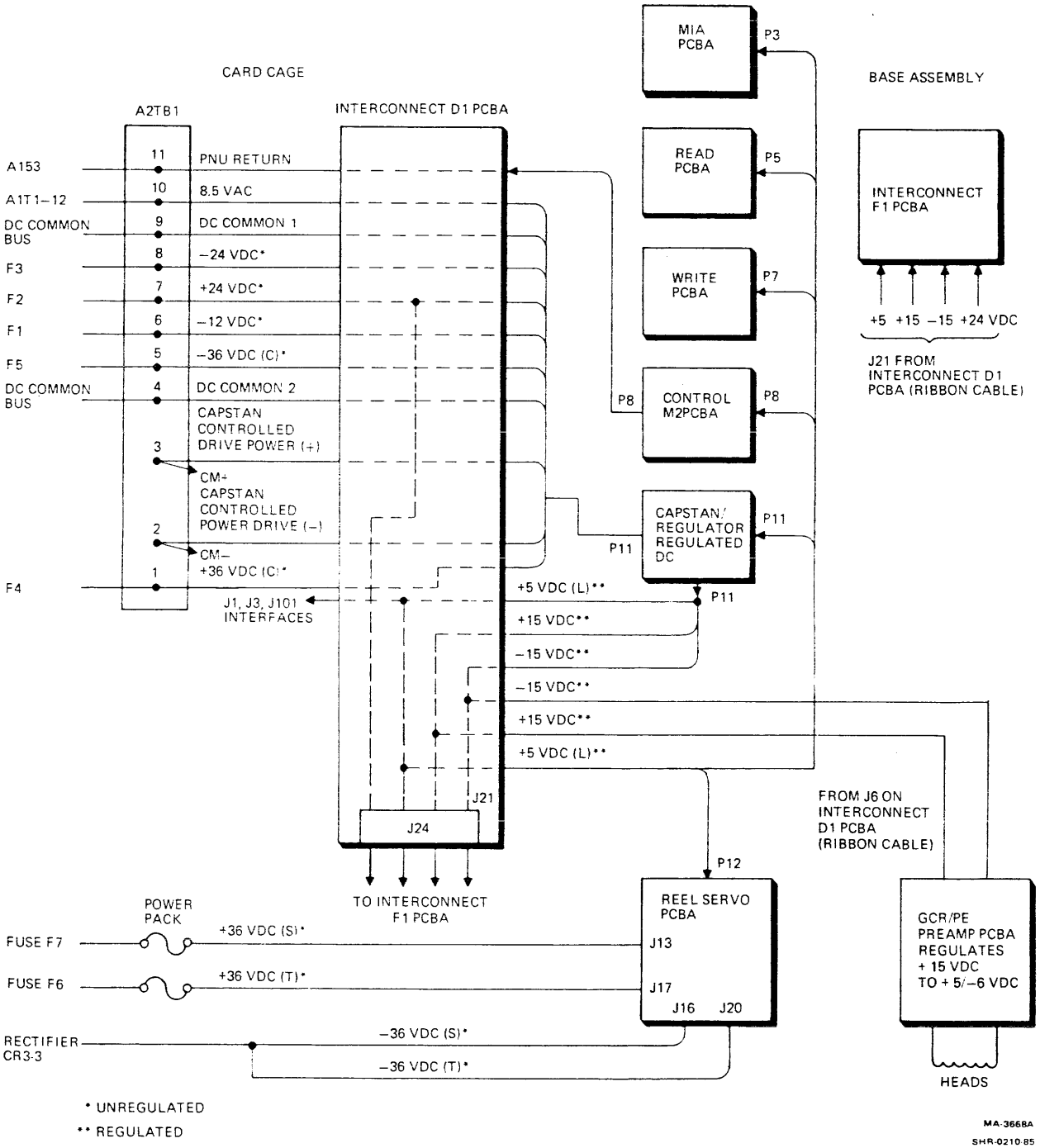
The +5 Vdc is used primarily in the logic and lamp drive circuitry. Current is limited in the +5 Vdc supply by current foldback techniques. This provides over-current protection.

Figure 3-9 Unregulated Power Distribution Circuit



NOTE: DC POWER REGULATION CIRCUITS ARE LOCATED ON THE CAPSTAN/REGULATOR PCBA

Figure 3-10 DC Power Regulation and Description



Over-voltage protection is provided on the +5 Vdc lines. A voltage over 6.2 Vdc fires an SCR and crowbars the output voltage to zero. Primary power must be removed for a short period of time to reset the crowbar circuitry.

Separate +5 Vdc regulation is provided on the reel servo PCBA for its internal logic circuits and amplifier circuits, +5 Vdc (S) and +5 Vdc (A) respectively.

The MIA interface logic uses +5 Vdc (L) from the transport by means of P3-11, 12, 13, 28, 29, and 30. Write, read, and control M2 modules are made compatible with MIA logic by obtaining their terminator voltage +5 Vdc (T) from the +5 Vdc (L) source by means of jumper W1 on the interconnect D1 PCBA. W1 is used to connect the +5 Vdc (L) at J11-22, 23, 62, and 63 to J5-7, 43; J7-21, 57; and J8-37. After the interconnection the signal source name changes to +5 Vdc (T).

3.5.5.2 15 Vdc - The 15 Vdc regulator on the capstan/regulator PCBA receives the 24 Vdc unregulated voltage and converts it into 15 Vdc. The 15 Vdc regulator uses a type LM325 IC to control the series power transistors and has a 2.5 A full-load capacity. TP15 is used to monitor regulated +15 Vdc output, and TP18 to monitor regulated 15 Vdc.

3.5.5.3 +5/6 Vdc - The +5 Vdc and 6 Vdc used in preamp circuitry are derived from the +15 Vdc and —15 Vdc by voltage regulators U13 and U14 on the GCR/PE preamp 1 PCBA. The GCR/PE preamp 1 is mounted on the back of the transport drive base assembly behind the read/write head area. The regulated 15 Vdc used for this purpose is supplied through the read PCBA.

3.5.5.4 +/—36 Vdc (C) - Unregulated +36 Vdc (C) for the capstan motor is connected from fuse F4/F13 to A2TB1 on the interconnect D1 PCBA, through the PCBA conductors to J11 and the capstan/regulator PCBA. Unregulated —36 Vdc (C) is routed from fuse F5/14 in the same way.

> **NOTE**
> **A2TB1-2 and A2TB1-3 serve as connecting points for controlled capstan motor drive power (negative and positive, respectively). These are part of the capstan circuits and not part of the power distribution system.**

3.5.5.5 +/—36 Vdc (S) - Unregulated +36 Vdc (S) for the supply reel motor is connected from fuse F7/9 directly to J13 on the reel servo PCBA. Interconnect D1 PCBA circuits are not used. Unregulated 36 Vdc (S) is connected from rectifier CR3-3 and F11, without fusing, to J16 on the reel servo PCBA.

3.5.5.6 +/—36 Vdc (T) - Unregulated +36 Vdc (T) for the take-up reel motor is connected directly from fuse F6 to J17 on the reel servo PCBA. Unregulated —36 Vdc (T) is connected from CR3-3, without fusing, directly to J20 on the reel servo PCBA.

3.5.6 Regulated Power Distribution

All 5 Vdc and 15 Vdc power is regulated on the capstan/regulator PCBA and distributed to other boards by means of J11 and conductors on the interconnect D1 PCBA. Internal connections of the interconnect D1 PCBA are listed in Appendix F. Regulated power distribution is shown in Figure 3-10.

The reel servo PCBA contains voltage regulators on the same board for the +5 Vdc (L) and +5 Vdc (A) used in the logic and amplifier circuits, respectively. Similarly, the +5/—6 Vdc used by the preamp circuitry is regulated internally on the GCR/PE preamp 1 PCBA.

3.5.7 Power Reset (NPORST), Enable (ENBL, NENBLE), and Master Reset Pulse (NMRSTP) Generation

Restoring power after an interruption initiates pulses that reset or preset various flip-flops, and so on, before starting operation. The following unregulated power at inputs to the capstan/regulator PCBA is involved.

+24 Vdc input at P11-16, 56
—24 Vdc input at P11-20, 60
+12 Vdc input at P11-15, 55
8.5 Vac input at P11-40

3.6 INTERFACING AND THE MULTIPLE INTERFACE ADAPTER (MIA) PCBA

Paragraphs 3.6.1 through 3.6.5 describe the theory of operation of the multiple interface adapter (MIA).

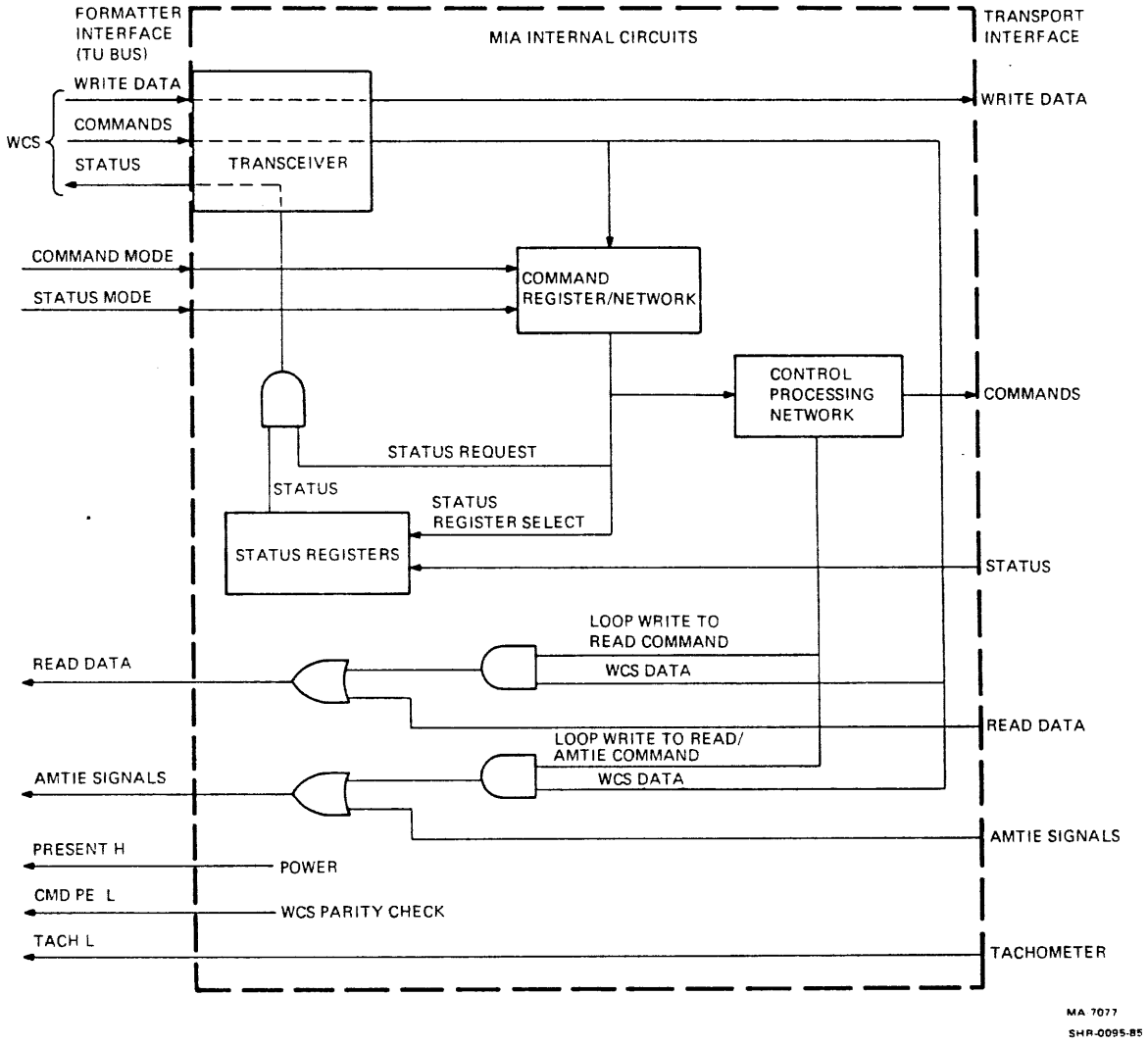
3.6.1 General

The multiple interface adapter (MIA) interfaces the formatter and transport circuitry. The MIA translates signals between the formatter interface (TU bus) and the transport interface. Figure 3-11 shows a basic functional block diagram of the MIA. The formatter interface (TU bus) is shown entering at the left and the transport interface is shown entering at the right.

3.6.2 Physical Description

The TU bus connects to the MIA through two jacks, J2 and J3. The MIA plugs into J1, J2, and J3 on the interconnect D1 PCBA. A separate cable connects the MIA to the transport's write PCBA (see Figure 3-5). Three sets of eight miniature DIP switches are mounted on the MIA. Two of these sets are used to set a hardware serial number, which is used as a signature in status reporting. The third set is used to create a special off-line test data pattern for maintenance purposes. Four toggle switches, used in manual (maintenance) mode, set the density, tape direction, and read or write. Two LEDs indicate manual mode and wrong parity on the incoming command word.

Figure 3-11 MIA Basic Block Diagram



3.6.3 Interface Connections

This section summarizes the interface connections (formatter and transport) for the MIA. There is a brief description with each signal name.

3.6.3.1 Formatter to MIA Interface (TU Bus) Signals - Table 3-1 lists the TU bus signals. A signal line is true when low (<1.0 V) and false when high (>2.8 V) except for the present line. The present line is true when high.

Table 3-1 TU Bus Signal Summary

Signal Name	Function
WCS 7:0 L	Eight write/command/status lines are bi-directional and multiplexed for the three different functions. Information on the WCS lines is determined by the assertion of the WDS, CMD or STAT control lines. The WCS lines are asserted by the formatter during a write data or command transfer and asserted by the tape transport during a status transfer.
WCS P L	A single bi-directional, multiplexed WCS line. It carries the odd parity bit for command/status transfers, and the parity channel for a write data transfer. WCS P is asserted by the formatter during command or write data transfers and by the tape transport during status transfers.
WDS L	Write data strobe indicates that the WCS lines contain data to be written on tape. WDS is asserted by the formatter.
CMD L	Command indicates that the WCS lines contain a command byte. CMD is asserted by the formatter.
STAT L	Status enables the tape unit to place a status byte on the WCS lines. STAT is asserted by the formatter.
CMD PE L	The command parity error line indicates that a CMD/STAT address or command byte has been received by the tape transport with even parity. CMD PE is asserted by the tape transport. This line remains asserted until a clear command is received with odd parity. While CMD PE L is set, motion and density signals are not released to the tape transport.
TACH L	Tachometer is a line that reflects the digital output of the tape transports capstan servo motor tachometer.
PRESENT H	Present is asserted by the tape transport when it is connected to the TU bus and has power applied.

Table 3-1 TU Bus Signal Summary (Cont)

Signal Name	Function
RD 7:0 L, RD P L	The nine read data lines are asserted by the tape transport when read enabled. They can also be asserted during the tape transport loop-write-to-read diagnostic mode.
AMTIE 7:0 L, AMTIE P L	The nine amplitude track in error signals can be asserted by the tape transport when read enabled. They indicate, independently, that the read back amplitude, on the track for which they are named, has not come up to a predetermined threshold value. They can also be asserted during the tape transport loop-write-to-read diagnostic mode.

The WDS, CMD and STAT lines are mutually exclusive. That is, only one of the three (indicating one of three types of information on the WCS lines) can be logically true at any time.

3.6.3.2 Transport to MIA Interface Signals - Table 3-2 lists the transport interface signals. A signal line is true when low (<0.8 V) and false when high (>3.0 V) unless indicated otherwise.

Table 3-2 Transport to MIA Interface Signal Summary

Signal Name	Description
Signals From the Transport to the MIA	
ISLT0	The signal, when low, indicates the operator panel port select switch is in the 0 position.
ISLT1	Same as ISLT0 except switch position 1.
ISLT2	Same as ISLT0 except switch position 2.
IONL	This level is low when the on-line flip-flop is set. When low, the transport is under remote control; when high, the transport is under local control.
ILDP	This level is low when the transport is ready and on-line and tape is at rest with the BOT tab under the photosensor. The signal goes high after the tab leaves the photosensor area.
IFPT	This level is low when tape is loaded and under tension in the vacuum column, and the supply reel has the write-enable ring removed.

Table 3-2 Transport to MIA Interface Signal Summary (Cont)

Signal Name	Description
IEOT	This level, when low and the transport is ready and on-line, indicates the EOT reflective tab is positioned under the photosensor.
IRWD	This level is low when the transport is engaged in any rewind operation.
IRDY	This level is low when the transport is ready to accept any external command, that is, when: 1. Tape is under tension in the vacuum column. 2. A load or rewind command has completed. 3. There is no unload command in progress. 4. The transport is on-line.
IDDI	This level is low when the PE density mode is selected.
ITACH	This pulse (1 us) is triggered by the positive and negative transitions of the capstan optical tachometer. The tape movement between alternate pulses is 0.254 mm (0.01 in).
IARA ERR	This signal is generated on the read PCBA and sent through the write PCBA to the MIA by means of cable. It is effective in write mode only and notifies the system that the automatic read amplifier burst expected at the transport for recording has failed to record properly within the allotted time.
WRT STAT H	Write status is indicated by a high = true WRT STAT signal to the MIA by means of cable from the write PCBA.
MOTION H	The MOTION high = true signal, by means of cable from the write PCBA to the MIA, indicates the tape transport is in motion.
WRT BIN H	WRT BIN indicates that the tape transport is writing binary data. It is routed from the write PCBA to the MIA by means of cable. This information is arbitrarily derived from the WD4 channel.

Table 3-2 Transport to MIA Interface Signal Summary (Cont)

Signal Name	Description
NPORST	NPORST initiates resetting and presetting of circuits when power is applied.
IAMTIE 7:0, P	An AMTIE signal is generated for each of the nine tracks on the basis of the amplitude of the data as received from the GCR/PE preamp 1 PCBA. An unsatisfactory channel output causes the Write PCBA AMTIE circuits to issue a low = true IAMTIE signal to the MIA by means of cable. IAMTIEP is not generated as a parity bit for the other eight channels, but reflects the amplitude status of the IAMTIEP track.
IRD 7:0, P	The nine IRD channels provide the MIA with data read from the tape.
Signals From the MIA to the Transport	
IDDS	This level, when low, conditions the read electronics to operate in the 63 c/mm (1600 cpi) density mode. In addition, the data density line (IDDI) to the controller/formatter goes low and the 1600 indicator lights. Conversely, when IDDS is high, the read electronics operate in the 246 c/mm (6250 cpi) density mode, the 6250 lights, indicator, and IDDI goes high.
ISFC	This level, when low and the transport is ready and on-line, causes tape to move forward. When the level goes high, tape motion ceases.
ISRC	This level, when low and the transport is ready and on-line, causes tape to move in reverse. When the level goes high, tape motion ceases. An SRC is terminated upon encountering the BOT tab, or ignored if given when tape is at load point.
ISWS	This level is low for a minimum of 20 us after the front edge of an ISFC when the write mode of operation is required. The front edge of the delayed ISFC samples the ISWS signal and sets the write/read flip-flop in the transport to the write state. If read mode is required, the ISWS signal is high for a minimum period of 20 us after the front edge of an ISFC (or ISRC), in which case the write/read flip-flop is set to the read state.

Table 3-2 Transport to MIA Interface Signal Summary (Cont)

Signal Name	Description
IRWU	This level resets the on-line flip-flop and initiates a rewind operation. When rewind completes, an unload sequence runs automatically.
IRWC	This level, if the transport is selected ready, and on-line, causes tape to move in reverse, moving at rewind speed unless a low tape (LT) condition is sensed. In this case, movement is at reverse speed until tape stops at BOT. An IRWC is ignored if tape is already at BOT.
IRTH2	Not used
NLTH STDTH	NLTH and STDTH form a field that establishes the threshold of the read electronics for the nine AMTIE signals as follows:

NLTH	STDTH	Threshold	Mode
low	low	10%	IRG check
low	high	10%	IRG check
high	low	25%	write
high	high	20%	read

IWINH	This level, when low, prevents the write head from being energized, while allowing the erase head to operate normally.
IWD7:0, IWDP	These write data signals, when low at IWDS time, result in a flux reversal being recorded on the corresponding tape track.
IWDS	This is the trailing edge of the IWDS signal. It strobes the write data lines to the write PCBA circuits.
NTSTR	Not used

For ISLT0, 1, 2 to reflect their true levels, MIA command register 3, bits 3, 4, and 5 must be loaded with 1s by the formatter.

3.6.4 Command/Status Register Descriptions

The formatter sends commands to the MIA PCBA to control the tape transport and get status information. It sends commands over the WCS lines. Because of bidirectional and multiplexed traffic on these lines, command and status information is handled by specifically addressable registers. Instantaneous status information is constantly updated at the registers and available upon request by the formatter.

3.6.5 Circuit Description

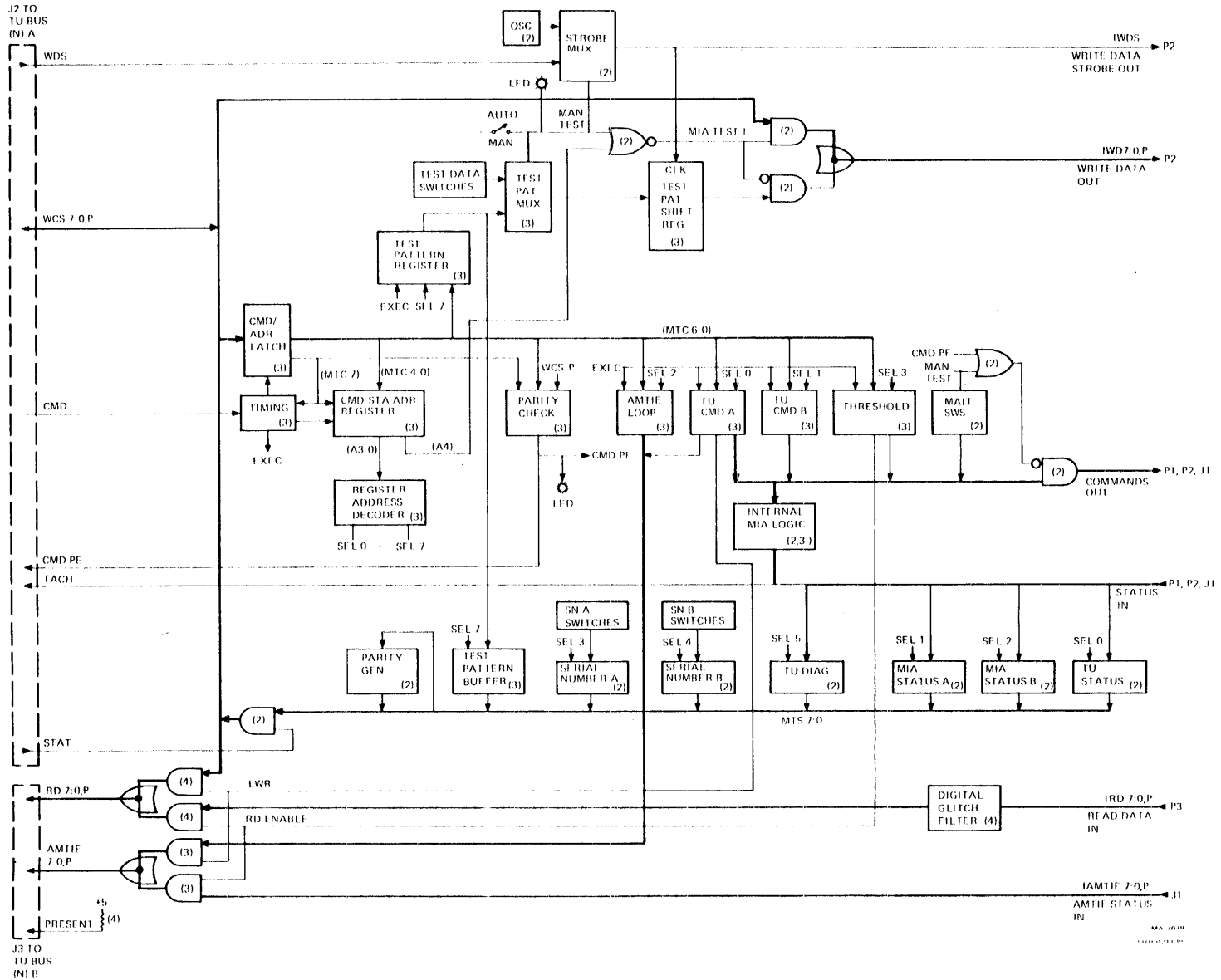
The MIA is essentially a switching and routing device for signals between the TS78 formatter and the tape transport, in which the MIA is installed. Generally, the MIA performs the following three groups of functions.

1. It accepts commands and write data from the formatter and relays them to the transport in a form compatible with transport circuits.
2. It accepts read data, amplitude track in error (AMTIE) signals, and status indications from the transport and relays them to the formatter in a form compatible with formatter circuits.
3. It participates in certain system-checking functions, such as looping WCS signals from the formatter into the read data lines or, through registers, into the AMTIE lines back to the formatter. These functions are called loop-write-to-read (LWR).

With reference to the transport interface, each signal uses a dedicated path. However, in dealing with the formatter interface (TU bus), some of the signals (that is, write data and most of the command and status indications) use the same conductors on a time-sharing basis. The MIA's basic function is multiplexing/demultiplexing signals on the write/command/status lines.

Figure 3-12 is a detailed functional view of the MIA PCBA. The various signal paths are described in the following paragraphs.

Figure 3-12 MIA PCBA Detailed Block Diagram



3.6.5.1 Commands and Command Timing - Two types of command formats may be sent over the WCS lines by the formatter, as shown in Appendix C. They are the command/status address specifier byte and the actual command byte. The address specifier byte always has bit 7 = 1 and the command byte always has bit 7 = 0. Both bytes are strobed to the MIA by the CMD L line. Figure 3-13 shows the MIA logic associated with command timing, and Figure 3-14 shows the timing associated with both types of command formats.

Figure 3-13 Command Timing Logic

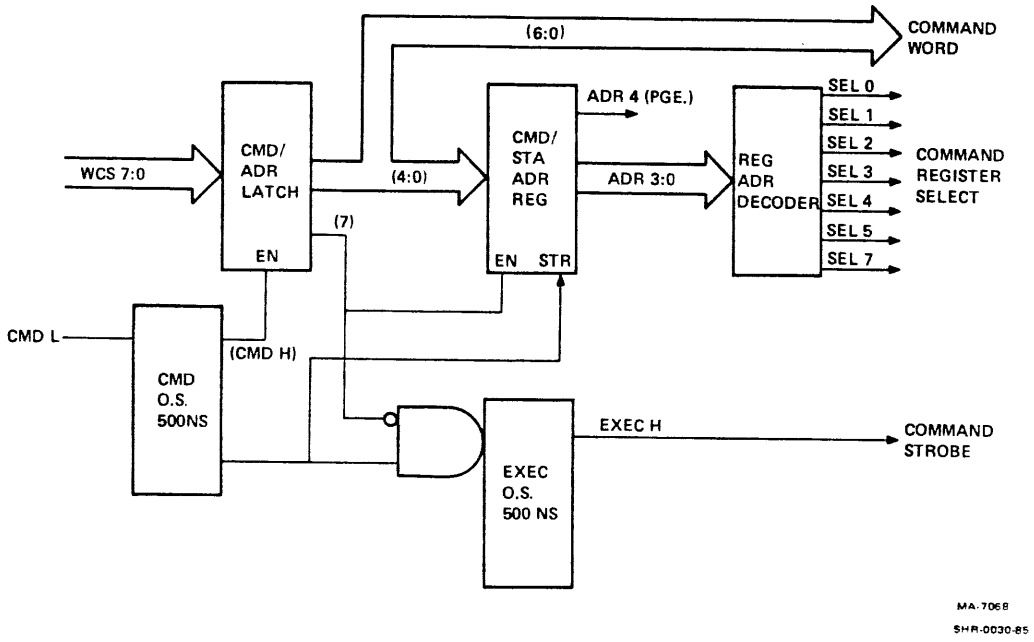
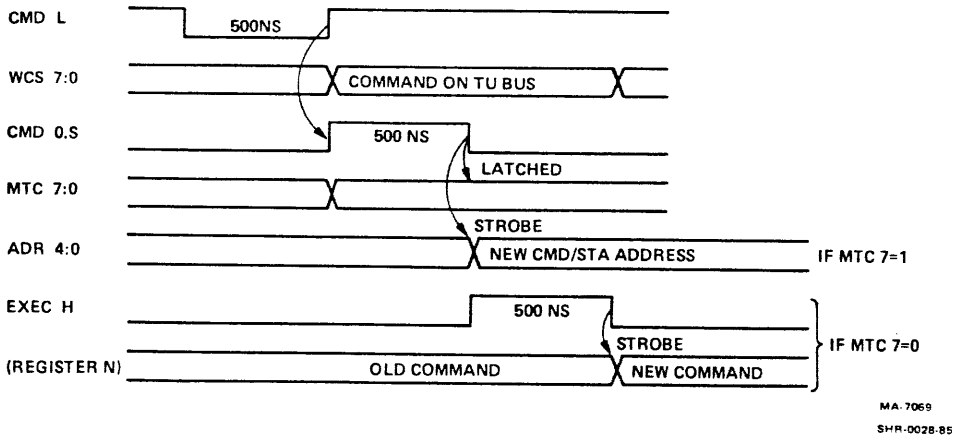


Figure 3-14 Command Timing



The formatter places the command/status address specifier byte on the WCS lines during the trailing (rising) edge of CMD L. The rising edge of CMD L also triggers the 500 ns CMD one shot. When the CMD one shot times out, its Q output latches the byte into the CMD/ADR latch, whose outputs MTC 7:0 now become stable. The Q output strobes MTC 4:0 to the CMD/STA ADR register which is enabled by CMD 7=1. The CMD/STA ADR register now holds the lower five bits of the command/status address specifier byte. The four low bits of its output (ADR 3:0) input to the MIA register address decoder, which asserts a select line (SEL 0-5, 7) to the associated command and status register pair. The next step in the process is a formatter initiated command write into the addressed command register, or a status request from the addressed status register. The high bit output of the CMD/STA ADR register (ADR 4) is not an address bit, but rather the pattern generator enable (PGE) bit. The PGE bit enables the test data pattern, written before in command register 7, to be placed on the transport interface IWD lines.

When the formatter writes a command byte to the MIA, the byte is latched into the CMD/ADR latch as described in the previous paragraph. However, because bit 7 = 0, the CMD/STA ADR register input is disabled and retains its previous contents. Signal MTC 7 enables one input of the execute (EXEC) one-shot. On the trailing edge of the CMD one-shot, the EXEC one-shot is fired. The one-shot output (EXEC H) strobes MTC 6:0 into the addressed command register on the trailing edge of its 500 ns time-out. The output of the command register is routed to the transport interface or internally within the MIA, depending on the register.

Command parity is checked in both command byte formats by inputting MTC 7:0 and WCSP into a parity checker. The checker is strobed on the trailing edge of the CMD one shot. If wrong (even) parity was received at the MIA, the CMD PE TU bus line is asserted back to the formatter, and the command is not gated to the transport interface. A command parity error also causes the CMD PE LED on the MIA to light.

3.6.5.2 Status - The formatter can examine the contents of any status register in the MIA, after it selects the desired register with an address specifier command (refer to Paragraph 3.6.5.1). Transport status, from the transport interface, is input to certain status registers while others receive inputs from sources on the MIA. The formatter requests status by asserting the TU bus STAT L line. Signal STAT L clocks the transport interface lines to the registers and enables the selected register outputs from the MTS lines to the WCS lines. The trailing edge of STAT L clocks the status byte into the formatter circuitry. Status parity is generated by examining MTS 7:0 and asserting WCS P as required to maintain odd parity on the WCS lines.

While the tachometer signal (TACH) is input to the TU DIAG register, it is also transferred from the transport interface to the TU bus with no gating. This is done so the formatter may examine it in real time during a data transfer operation, to guarantee constant tape velocity and ramp times within specification.

3.6.5.3 Write Data - Once the formatter has commanded the transport to move tape, and enabled the write circuitry, it asserts nine bits of write data on WCS 7:0, P, along with the WDS strobe pulse. The data is gated through the MIA to the transport interface IWD 7:0, P lines, and the strobe pulse is gated to the IWDS line. Another source of write data is from within the MIA PCBA in test mode. This source is discussed in greater detail in Paragraph 3.6.5.1.

3.6.5.4 Read Data - Data from the transport's read electronics enters the MIA from the transport interface as IRD 7:0, P. The read data passes through nine independent digital filters that condition the waveforms, and is gated to the TU bus RD 7:0, P lines. Before the data transfer, the formatter specifies a read enable command by setting bit 6 of the threshold command register.

3.6.5.5 Amplitude Track In Error (AMTIE) Bits - AMTIE status information enters the MIA card from the transport interface by means of cable from the write PCBA) as I AMTIE 7:0, P. AMTIE status is gated out to the TU bus as AMTIE 7:0, P, assuming the MIA is read enabled. Read enable is set and AMTIE bits are generated, in read and write mode, for both PE and GCR formats.

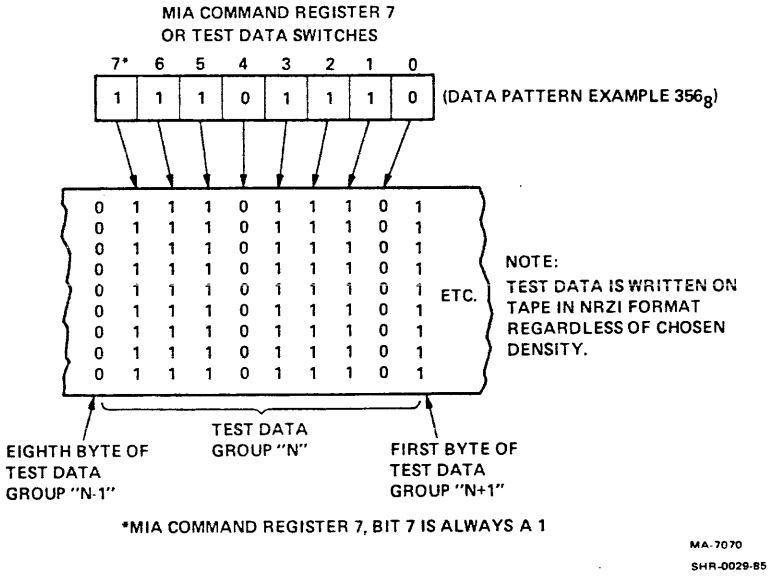
3.6.5.6 Diagnostic Loops - The formatter can configure MIA circuitry in loop write-to-read diagnostic mode to check the TU bus and portions of the MIA circuitry, independent of the transport. It does this by resetting the read enable bit (bit 6 of the Threshold command register), and setting the loop write-to-read (LWR) bit (bit 2 of the TU CMD A register). In this diagnostic loop mode, data can be sent from the formatter, looped around in the MIA, and sent back to the formatter in two ways. First, a data byte plus parity is sent out on the WCS lines and immediately looped around to the RD lines. In this case the WCS data is not accompanied by an enabling pulse on the CMD or STAT TU bus lines. Second, the formatter can loop data around from the WCS lines to the AMTIE lines. It does this by writing a full nine bits into the AMTIE loop command register (bits 6:0) and the TU CMD A register (bits P, 7). After writing data into these registers, the formatter can read it back on the nine AMTIE lines.

3.6.5.7 Switches and Indicators - The MIA PCBA has three sets of eight DIP switches, four toggle switches, and two light emitting diodes (LEDs). Two of the DIP switches set the four digit transport serial number in BCD format. The third DIP switch is used to create a write data pattern in off-line (manual) mode. One of the toggle switches functionally removes the transport from the formatter (auto), and places it in off-line test mode (manual). When the transport is in manual mode the remaining three toggle switches set tape direction (rev-off-fwd), data mode (read-write), and density (1600—6250). The two LEDs indicate manual mode and command byte parity error.

3.6.5.8 Test Pattern Generation - Test data patterns can be created and written to the transport over the transport IWD lines in any of three ways. First, the formatter can write test data through the MIA in the conventional manner any data is written to tape. That is, strobing data from the TU bus WCS lines to the IWD lines. Second, the formatter may write a pattern into MIA command register 7, the test pattern register. Then the pattern is shifted out to the IWD lines by formatter-generated WDS strobe pulses. And third, a maintenance engineer places the transport in manual mode and creates a test data pattern in the appropriate on-board DIP switches. In this case, data is strobed to tape by an oscillator on the MIA PCBA pulsing the transport interface IWDS line.

The way the second and third methods actually write the test data is much different than the first, however. When register 7 or the switches are used, the pattern in the register or switches is not the pattern written to tape for each character. Rather, each bit of the 8-bit test pattern byte corresponds to a full 9-bit character on tape, for a total of eight tape characters. If a given bit in the test pattern is a 1, the corresponding 9-bit character is written to tape as a flux change in all tracks. Similarly, if the test pattern bit happens to be a 0, the tape character has no flux changes in all tracks. In this method of writing test data, parity and recording format on tape is not observed. The pattern of eight bytes is written repetitively on tape until commanded to stop. Figure 3-15 shows an example of the test data tape format.

Figure 3-15 Test Data Tape Format



3.7 SYSTEM CONTROL

The following paragraphs describe the control signal generation and routing.

3.7.1 Control System Overview

Inputs and outputs from the host system are connected by means of the MIA, which plugs into J1, J2, and J3 on the interconnect D1 PCBA. The control logic circuitry manages transport operation through commands received from the host system and/or manual commands entered by means of switches on the transport. Figure 3-16 is a simplified block diagram of the control logic. Figure 3-17 shows the general routing of control signals throughout the transport assembly.

Most of the control circuits are mounted on the control M2 PCBA, which is plugged into connector J8 on the vertical interconnect D1 PCBA. The control M2 PCBA circuits are described in detail in Paragraph 3.7.6. Paragraphs 3.7.2 to 3.7.5 describe the control system in general.

Figure 3-16 Control Logic Block Diagram

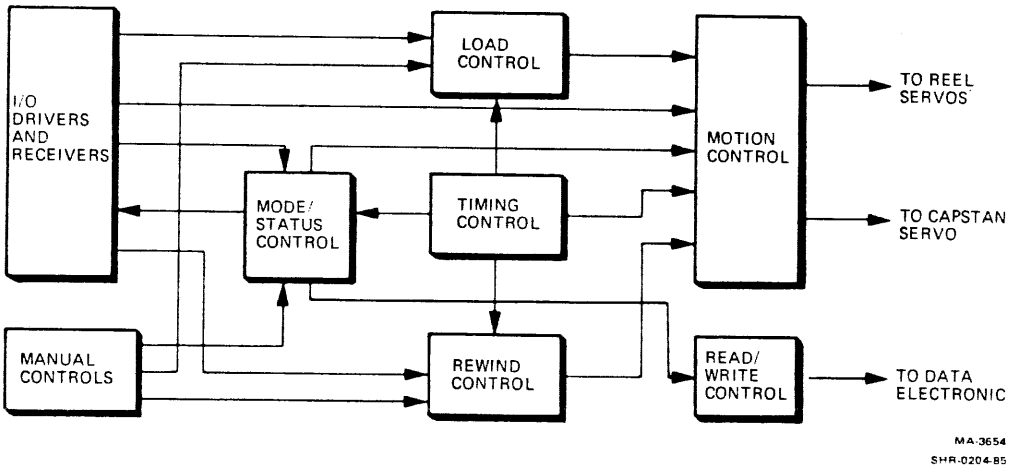
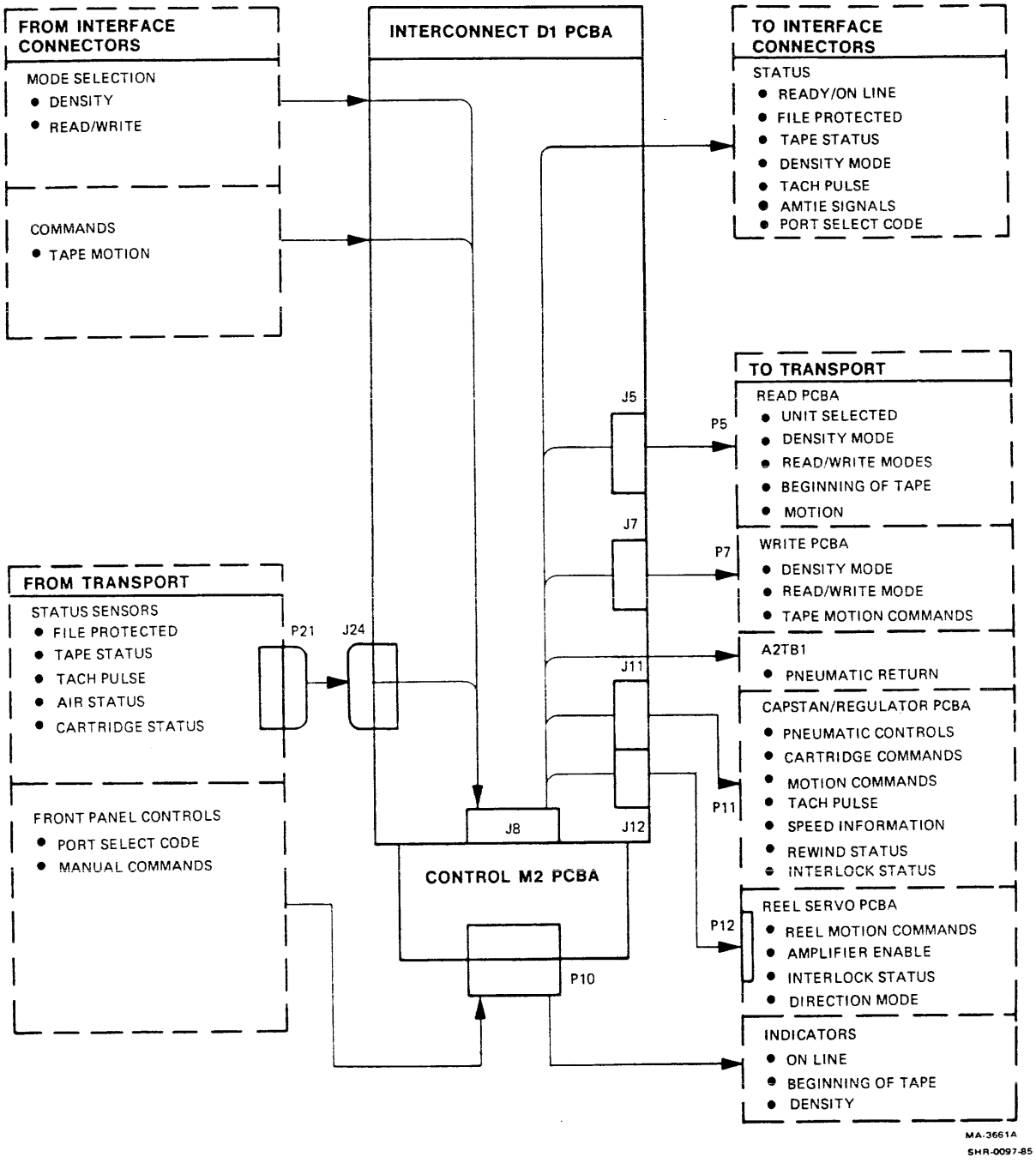


Figure 3-17 Control Signal General Routing



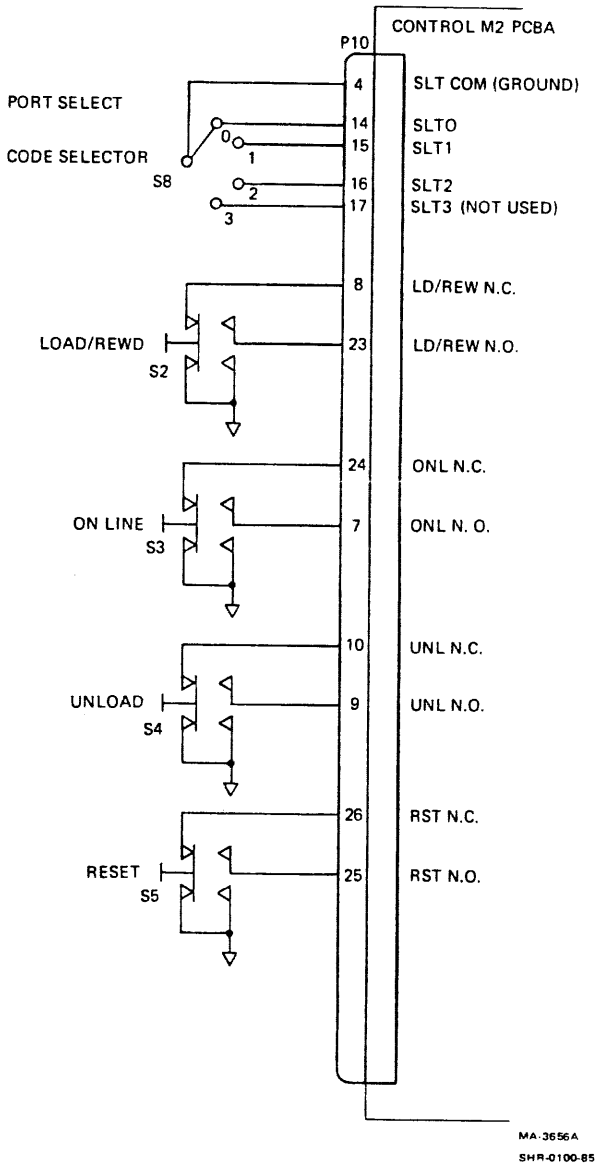
3.7.2 Manual Controls

Manual controls include the mode select thumbwheel, LOAD/REW, ON LINE, UNLOAD, and RESET switches. Figure 3-18 shows manual switching. The power circuit breaker switch, on the rear panel, is discussed in the power supply and distribution text.

In addition to the operating switches mentioned, the following switches are for maintenance purposes.

Tape motion switch (on control M2 PCBA)
Reel servo disabling switch (on reel servo PCBA)
Test write switch (on the write PCBA)
AGC exerciser switch (on the read PCBA)

Figure 3-18 Manual Control Switch



The thumbwheel switch (S8), next to the POWER indicator, sets the mode select code so the formatter may determine in which mode (normal or test) to place the transport. Positions 0, 1, 2, and 3 are used. Contacts 0, 1, and 2 are connected to the MIA module by means of the control M2 module. SLT COM on P10-4 is connected to ground. The switch grounds one of the four select lines which connect to the MIA module by means of the control M2 module. The mode selected by the switch determines which select line is grounded. SLT COM is grounded by jumper W3 on the control M2 module. Jumper W3 also asserts SLT and SLTA, which enables the various transport circuits involved in host system operation.

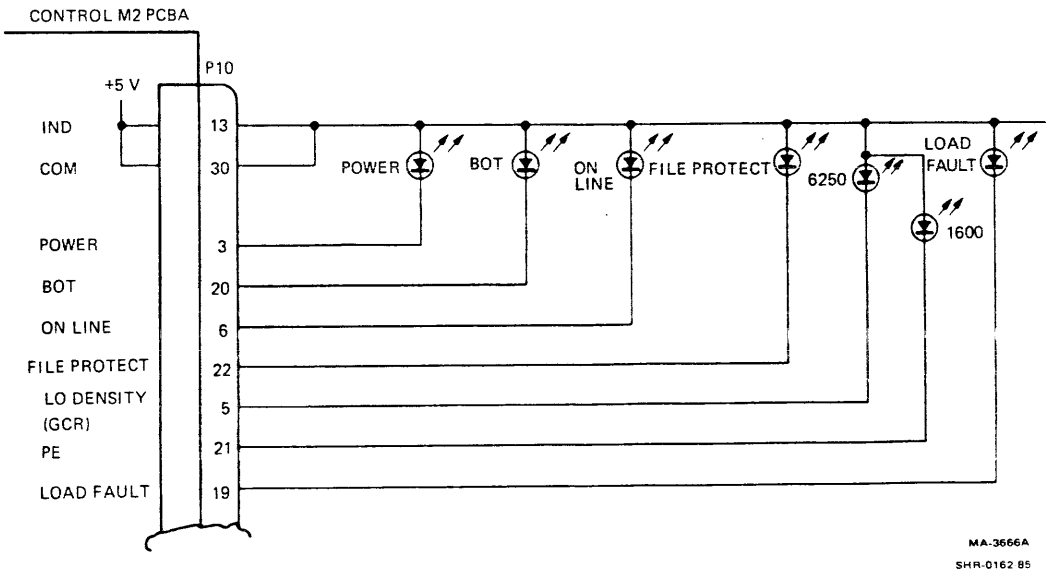
LOAD/REW, ON LINE, UNLOAD, and RESET switches S2 through S5 (Figure 3-18) are momentary pushbutton devices which normally hold one of their two contacts at low (nominal 0 V) level. When pressed, they remove signal ground from the contact and momentarily connect ground to the other contact. Circuits operated by these switch commands are on the control M2 PCBA, described in Paragraph 3.7.6.

3.7.3 LED Status Indicators

Panel LED indicators include power, BOT (beginning of tape), on-line, file protect, load fault, 6250, and 1600. All indicators are connected to system circuitry through connector P10, which is mounted on the control M2 PCBA (Figure 3-19). The BOT indicator lights when the BOT marker is sensed and the tape is in the beginning-of-tape position. On-line indicates that the transport has been switched to the host system. (It is not an active part of the host system, however, until it is selected by the system and the transport is in ready status.) File protect (FPT) lights to indicate that the write-enable ring is not in place and the file is protected against writing. When lit, the Load fault (LDF) indicator cautions the operator that the tape failed to load. The 6250 indicator lights when the system is set for high density (GCR) operation; the 1600 indicator lights when the system is set for low density (PE) operation.

The indicator common (IND COM) provides +5 V to the indicators whenever transport power is on. Any of the LEDs conduct and emit light when the controlling input to the LED is low = true.

Figure 3-19 Front Panel Indicator Connections



3.7.4 Control System Inputs/Outputs

Figure 3-17 shows general routing for inputs and outputs to the control system. Interface connectors are mounted directly on the interconnect D1 PCBA, which routes command/status signals between the MIA interface module and the control M2 PCBA edge connector, P8.

Manual control switches and associated LED indicators are connected directly to J10 on the control M2 PCBA. Status sensor information is connected through interconnect F1 PCBA J21 to J24 on interconnect D1 PCBA. It is then routed to the control M2 PCBA. Outputs from the control M2 PCBA to various other card cage PCBAs (read, write, capstan, reel, servo, etc.) are all routed by means of the interconnect D1 PCBA.

3.7.5 General Modes of Operation

There are two types of transport operational modes: off-line modes and on-line modes.

Off-line operation includes tape loading, unloading, and some maintenance/test procedures. Front panel control switches are used to select and run off-line operations. In addition to these controls, maintenance test procedures involve the use of a manual switch (on the control M2 PCBA) to control tape motion, and a similar switch (on the reel servo PCBA) to disable reel motion. Other manual switches are on the MIA interface board.

On-line operation essentially includes read/write and motion procedures under host system control. Initiation of on-line operation requires the following conditions: tape has been loaded; all interlocks have been made; and the transport has provided the selected, ready, and on-line (SRO) signals.

1. A rewind command is initiated.
2. A high speed reverse signal is sent to the capstan servo.
3. Tape is wound onto the supply reel at high speed until low tape is sensed.
4. The tape slows to synchronous reverse speed.
5. Tape continues to wind onto the supply reel until it reaches the BOT tab, where it stops.

An unload command may be initiated from the control panel (off-line) or the interface (on-line). If the tape is not at BOT, a rewind is executed; then the unload operation is executed. If the tape is at BOT, only the unload operation is executed.

Jumper W3 on the control M2 module asserts SLT and provides SLTA to the output logic. The output logic provides NSLTA to the read function through J8-23. Jumper W3 also grounds SLT COM on J10-4. SLT COM routes to the select switch on the front panel control unit. The select switch connects ground to one of the ISLT lines, which routes to the MIA module through J10 and P8 of the control M2 module and the interconnect D1 PCBA.

3.7.6 Control M2 PCBA

The control M2 PCBA integrates, develops, and applies commands received from the host system, manual control switches, or MIA test switches.

> **NOTE**
> **The transport must be on-line to use MIA test switches.**

External commands are delivered from the interface to the control M2 PCBA by means of the interconnect D1 PCBA, connector J8. Manual commands from the control panel are connected through J10 on the control M2 PCBA.

Internal signals used in the control process include feedbacks and status signals. These arrive at the control M2 PCBA by means of the interconnect F1 PCBA (J24) and J10 on the interconnect D1 PCBA. Timing is provided by a 1 MHz oscillator and a battery of frequency dividers, which are part of the control M2 PCBA.

Inputs and outputs are detailed in Figure 3-20. The following text describes control M2 PCBA circuits that develop input signals into control outputs for other transport circuits and the host system. Refer to Paragraph 3.2 for a functional description of the entire control subsystem external to the control M2 PCBA.

3.7.6.1 Control System Timing - Control M2 PCBA clocks are derived from the 1 MHz oscillator circuits associated with crystal Y1 (Schematic Number 106875, zone 2-6H). The 1 MHz prime frequency is referred to as clock A (CLKA), which may be monitored at TP20.

The oscillator output is cascaded through a series of decade dividers and one final flip-flop to provide the required additional frequencies. The available frequencies, including oscillator output, are listed in Table 3-3. Load/unload sequence counter outputs are described in Paragraph 3.9.2.

Figure 3-20 Control M2 PCBA Inputs and Outputs

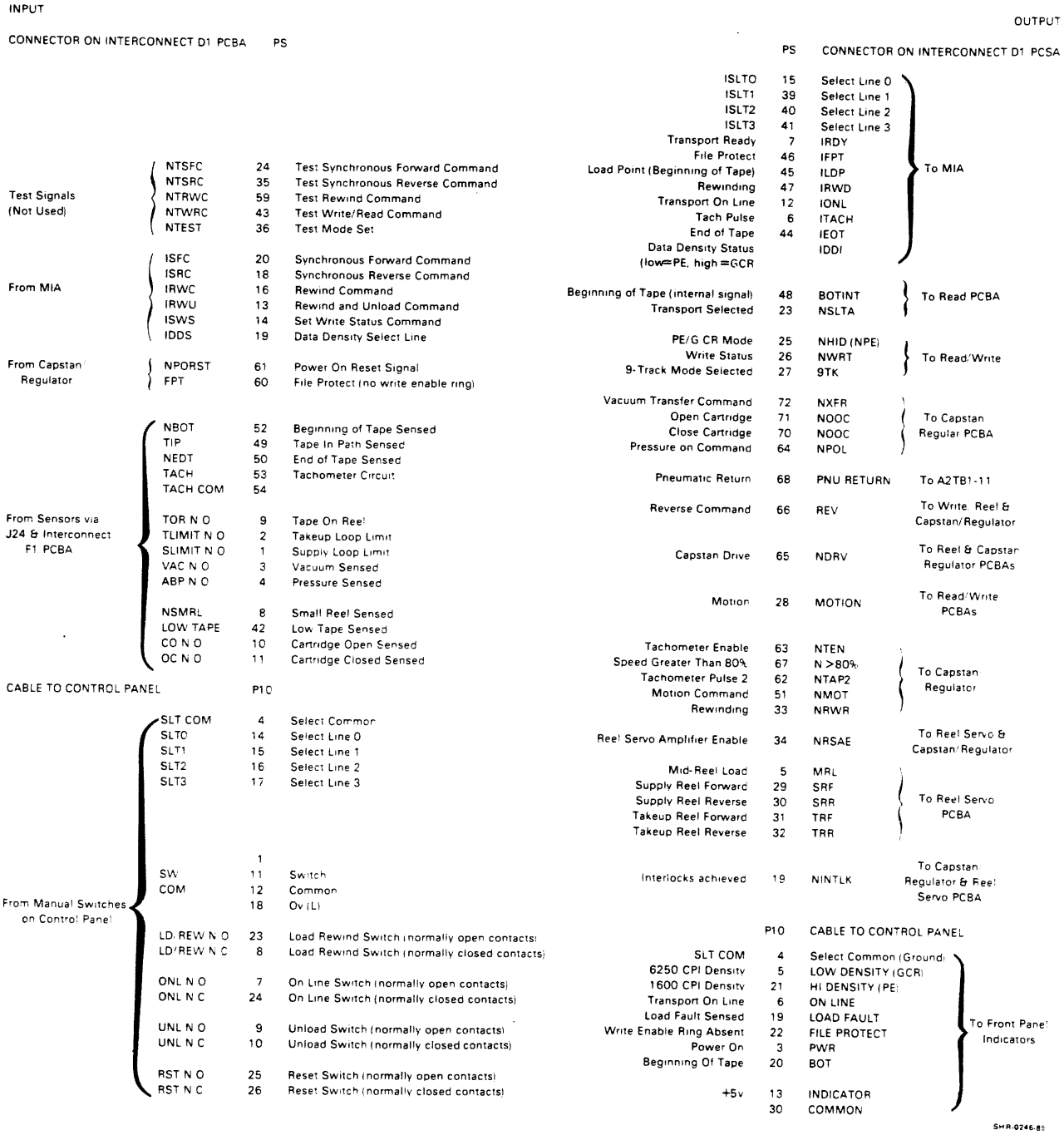


Table 3-3 Basic Timing Frequencies

Name	Frequency	Test Point	Distribution	Primary Use	Associated Signals
CLKA	1 MHz	TP20	U12-3, 11	Tachometer pulse generation	TAPEN, TACHP, NTAP2, NTEN, NRWR, N>80%
			U52-3,11 U53-3,11	GO pulse generation	NGOP
CLKB	100 KHz	TP15	U35-11, U25-11	Master reset generation	NMRSTP
			U35-3, U34-3, 11 U61-3	Load reset pulse, load pulse and rewind pulse generation	NLRSTP, NLDP NREWP
CLKC	10 KHz	TP39	U225-12,	Stop pulse generation	NDRV
CLKD	100 Hz	TP17		Not used	
CLKE	10 Hz	TP19	U63-9	Load fault indicator control	LOAD FAULT (LDFS)
			U54-13	GO pulse, stop pulse generation	NGOP, NDRU, NMOT
			U114-11	Load fault signal	LDFS
				Vacuum control	NXFR
				Reel motion	SRF, SRR, TRF, TRR
			U183-5	Thread signal	THDS
			U216-11, U226-11	Set tape loops command	STL
			U115-9	Load fault 6	LDF6
CLKF	1 Hz		U184-11,3	Interlock circuits	INTLK1, DINTLK, NDINTLK,
INTLKPI					
			U101-14	Load fault 0	LDF0
CLKG	0.5 Hz		U64-13, U74-1	Forward/ reverse motion	NGOP, MOT

3.7.6.2 Normal/Test Mode Selection - The common contact of the mode select switch (SLT COM) is grounded by jumper W3 on the control M2 PCBA. This ground eventually generates the NSLTA control signal. NSLTA is applied to other transport circuits through interconnect D1 PCBA.

3.7.6.3 Modes of Operation - The control M2 PCBA processes internal and external commands to establish the required modes of operation. Control M2 PCBA circuits which process various commands and the operational task operations (load, and so on) are discussed later in this chapter. The major modes are determined by the purposes for which the transport is to be used, and are as follows.

Read Only, Forward
Read Only, Reverse
Write and Read Forward
Low Recording Density (PE)
High Recording Density (GCR)
Load Tape
Rewind Tape
Unload Tape

Some of these modes are effective simultaneously and with certain transport states that define conditions required to enable the major modes of operation. The transport states are as follows.

Reset/preset*
Interlock*
9-track PE/GCR
Beginning of tape interlock
Unit selected

3.7.6.4 Reset/Preset State - The reset/preset state clears and resets the transport's circuits for a new operation. It is initiated automatically when power is applied after an interruption. The operator can initiate it at other times by momentarily pressing RESET.

3.7.6.5 Interlock State - The interlock state circuits are designed to inhibit and/or enable various modes on the basis of monitored air and tape conditions.

---

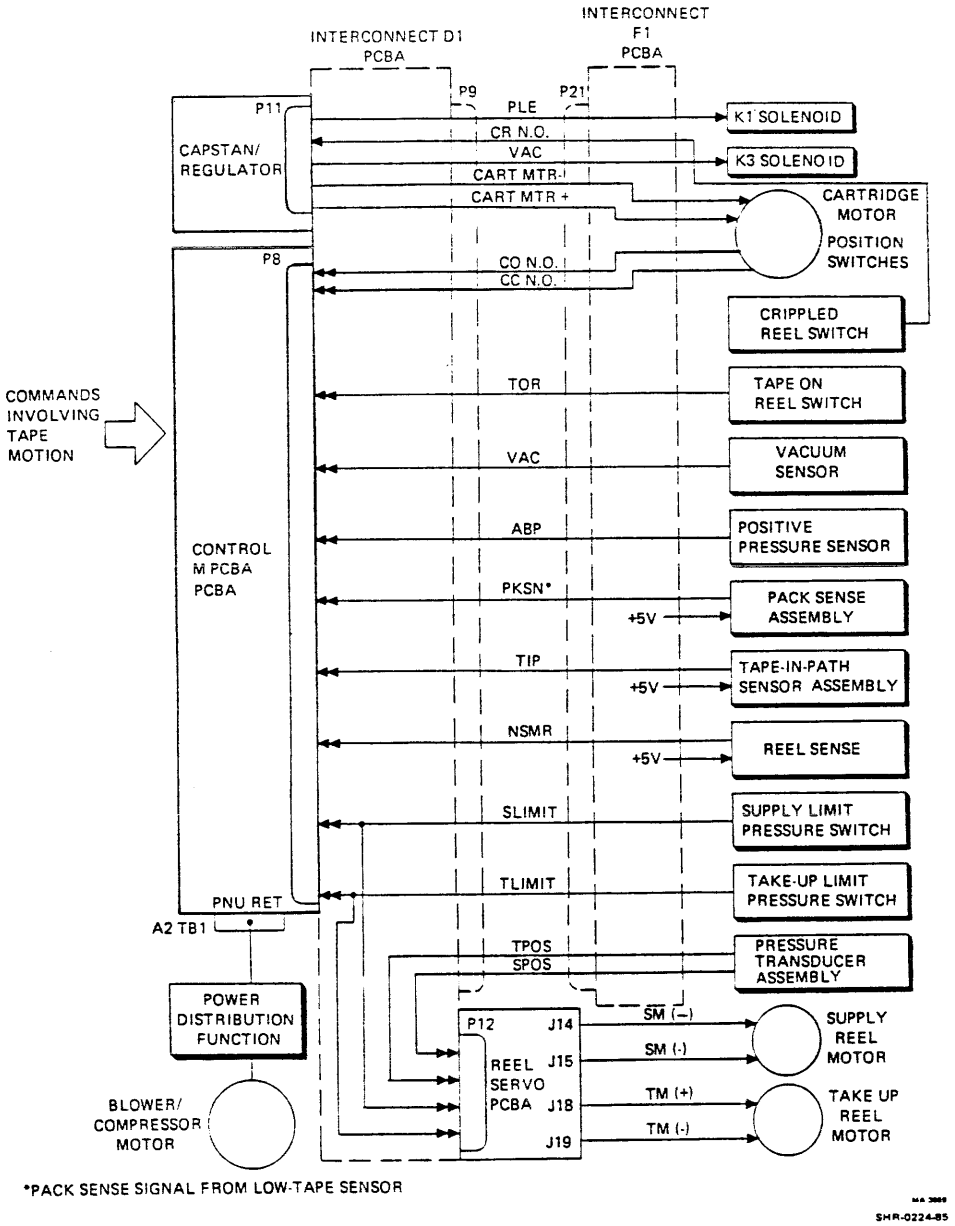
* These states are discussed in the following paragraphs.

3.8 AIR LOAD/CONTROL

The air load/control subsystem includes air and electronic provisions which control and monitor the tape path between the reels (Figure 3-21).

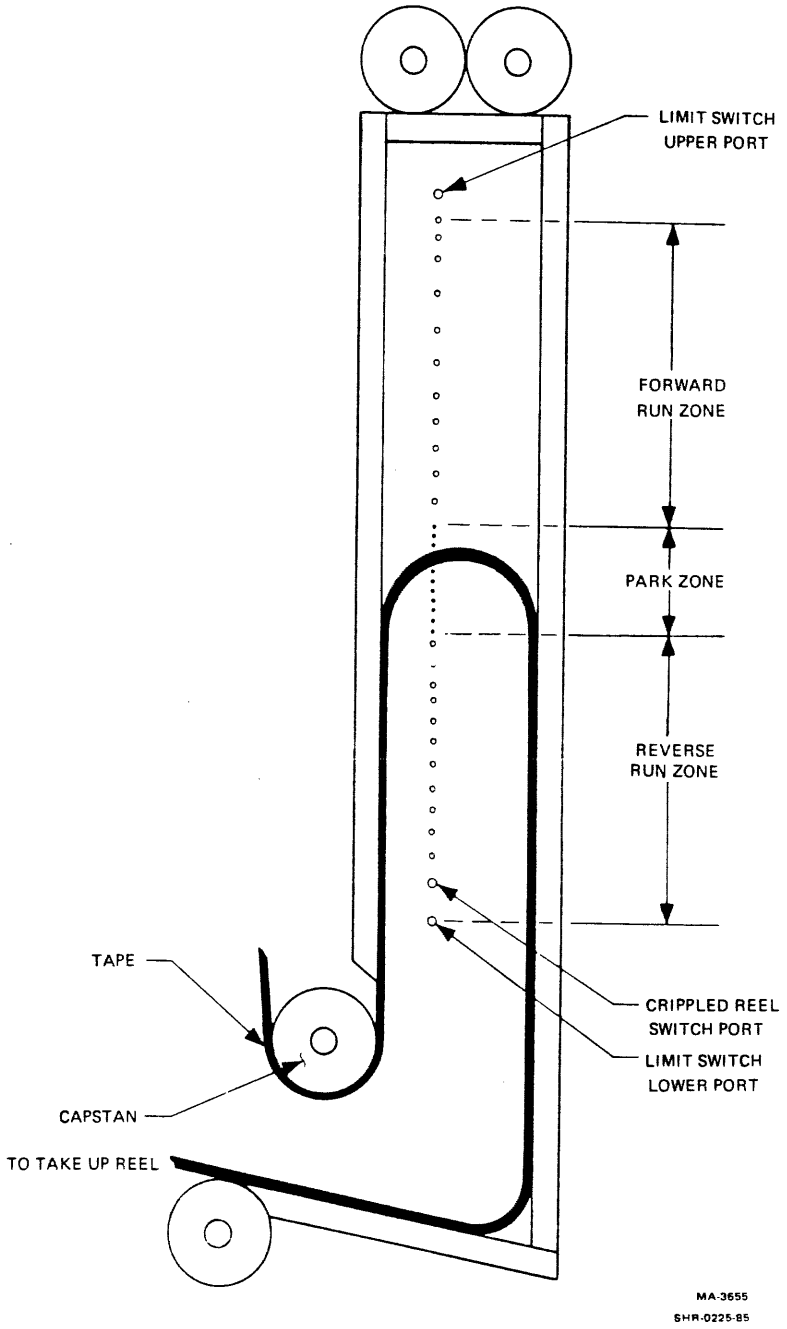
The capstan servo subsystem controls the speed at which the tape passes the read and write heads. The speed of the supply and take-up reels varies according to the quantity of tape on the reels. For example, when the supply reel is full and the take-up reel nearly empty, the take-up reel must rotate much faster than the supply reel to transfer the tape past the read/write head assembly at a constant 3.2 m/s (125 in/s) rate. Each of the reels is driven by a separate motor controlled by a servo system that adjusts the speed as needed to maintain proper tape transfer. The air load/control subsystem provides tape loop status signals to the reel servo circuits, which use the information to regulate power applied to the reel motors.

Figure 3-21 Air Load/Control Function



The input port of the blower is used as the source of vacuum (minus atmosphere) pressure that forms the tape loops. Vacuum is applied to the buffer boxes, loop pocket, corner pockets, and capstan. Energizing solenoid K3, on the vacuum valve, switches vacuum to the take-up reel hub to make threading easier. A typical tape loop formation is shown in Figure 3-22.

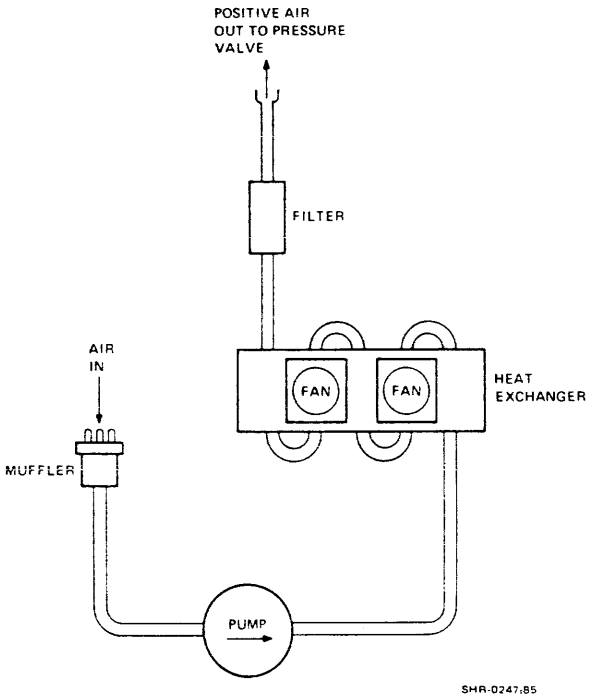
Figure 3-22 Tape Loop in Takeup Buffer Box



Pressure (positive atmosphere) is developed by a pump in the power pack/pneumatic assembly.

Air is drawn through an acoustic muffler by the pump and cooled in the forced air heat exchanger. The cooled air is filtered and delivered to the pressure valve on the base assembly. Figure 3-23 shows a schematic of the positive air supply system.

Figure 3-23 Positive Air Supply System



3.9 TAPE LOAD, UNLOAD, AND REWIND OPERATION

3.9.1 Load/Unload Sequences

3.9.1.1 Tape Load - During a load operation, the following series of events occurs (Figure 3-24).

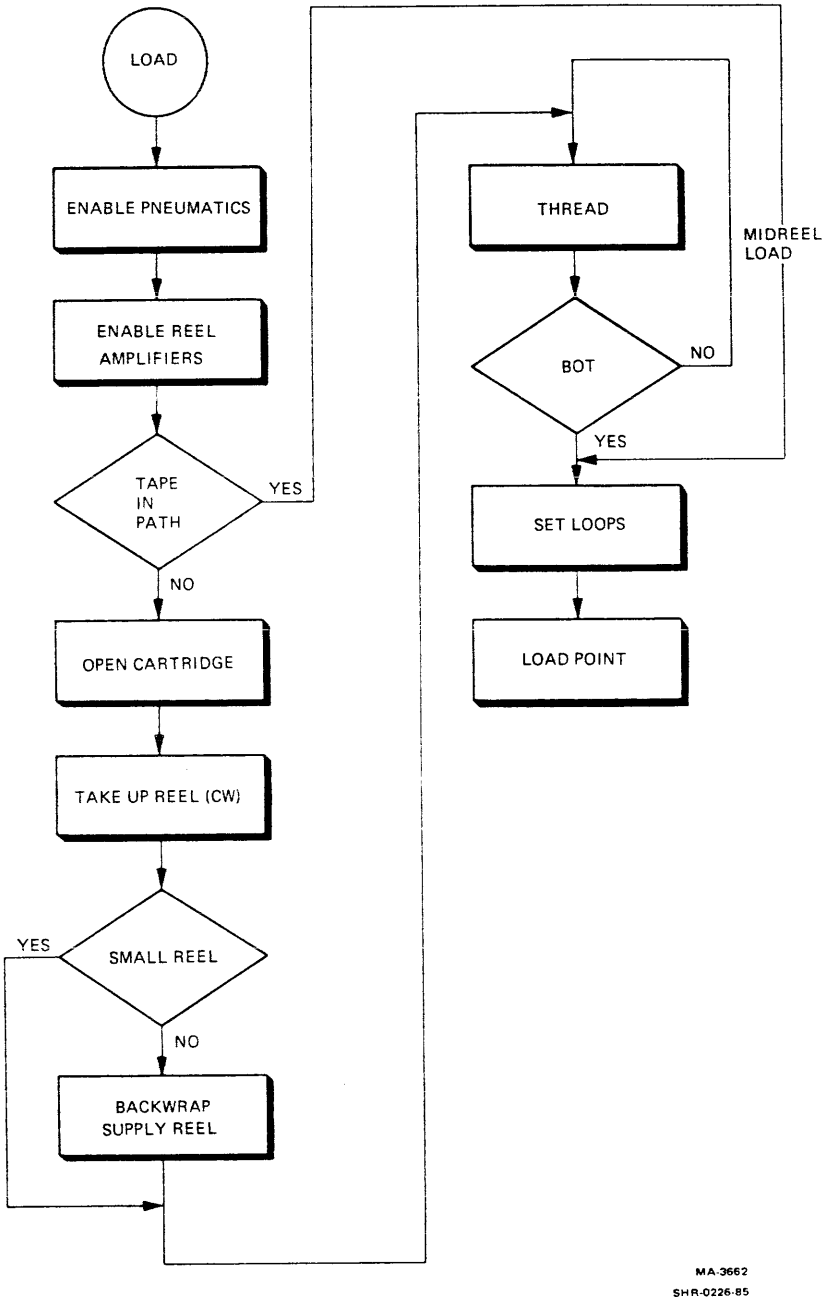
1. When the LOAD/REW button is pressed, the blower (vacuum) and compressor activate and the reel servos are enabled.
2. If tape is in the tape path at this time, a midreel load situation is assumed, and the unit sets loops and starts towards BOT.
3. If tape was not in the tape path, the cartridge motor opens the cartridge, the take-up reel turns clockwise (CW), and a normal load is initiated.
4. With the transport front door closed, the supply reel backwraps for about two turns before starting the threading process.

   NOTE
   If the diameter of the outside turn of tape is too small, the operator must manually place the tape in the tape path and load it with the transport front door open. The outer turn of tape must be between 1.59 and 0.64 cm (5/8 and 1/4 in) from the outer edge of the reel to accomplish autoload. Also, 216 and 178 mm (8-1/2 and 7 in) reels require the operator to place the leader over thread block 1.

5. After backwrapping, the supply reel starts forward, and tape threads through the tape path.
6. When beginning of tape (BOT) is detected, vacuum is applied to the buffer boxes, and the reel motors turn so that tape loops are set in the buffer boxes and interlock is made.
7. The transport is then ready for controller commands and data operation.

When the tape is satisfactorily loaded, the transport is ready to make data runs, and so on, as commanded by the controller. When a motion command arrives at the interface, the system control sends a drive (NDRV) and direction (REV, true or false) signal to the capstan servo subsystem. This initiates tape movement across the read/write head assembly. Pressure transducers in the air load/control subsystem sense tape loop position, and feed signals to the summing logic in the reel servo subsystem. The output of the summing logic is applied to the power amplifiers that drive the reel motors.

Figure 3-24 Load Sequence

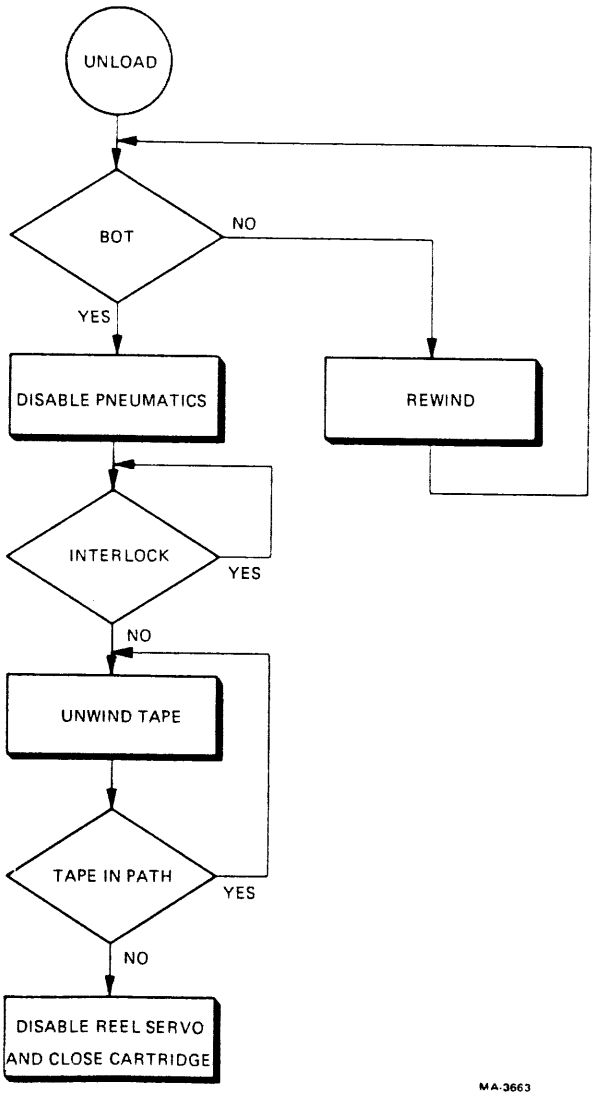


3.9.1.2 Tape Unload - The same auxiliary circuits used to load tape into the vacuum chambers (load operation) are used during an unload operation (Figure 3-25). When the UNLOAD button is pressed while the unit is at midtape, the following sequence is initiated.

1. Tape rewinds to BOT.
2. Blower (vacuum) motor is turned off, causing the pneumatic interlock to be broken.
3. Tape is wound onto the supply reel.
4. Reel motors are stopped.
5. Cartridge is closed.

The unload procedure with tape at BOT is the same but without rewind operation.

Figure 3-25 Unload Sequence



MA-3663
SHR-0101-85

3.9.2 Load/Unload/Rewind Circuit Operation

The tape loading procedure involves all steps that result in tape being installed and brought to the beginning-of-tape (BOT) position. The steps include motion in either or both directions, and may include cartridge and pneumatic system control. If a tape reel is installed on the supply reel spindle, but sensors determine the tape has not been threaded through the guides to the take-up reel, normal load conditions exist. Pressing the LOAD/REW button initiates the automatic threading process, which connects the tape to the take-up reel, forms the loops, and stops at BOT. The tape is then ready for a read or write operation.

If tape is sensed between the supply and take-up reels when the LOAD/REW button is pressed, a situation called a midreel load condition exists, and rewind mode is initiated. Rewind mode also terminates at BOT.

Note that when the UNLOAD switch is pressed, rewind mode is also initiated if the tape is not at BOT. Unload mode causes tape to be completely wound on the supply reel. Unload mode then causes the supply reel cartridge to close.

Figures 3-24 and 3-25 are a flowchart for reel-motion logic by manual control. The flowchart includes motions commanded by the LOAD/REW and UNLOAD switches.

3.10 REEL SERVOS

The following paragraphs describe the reel servo system.

3.10.1 Reel Servo Overview

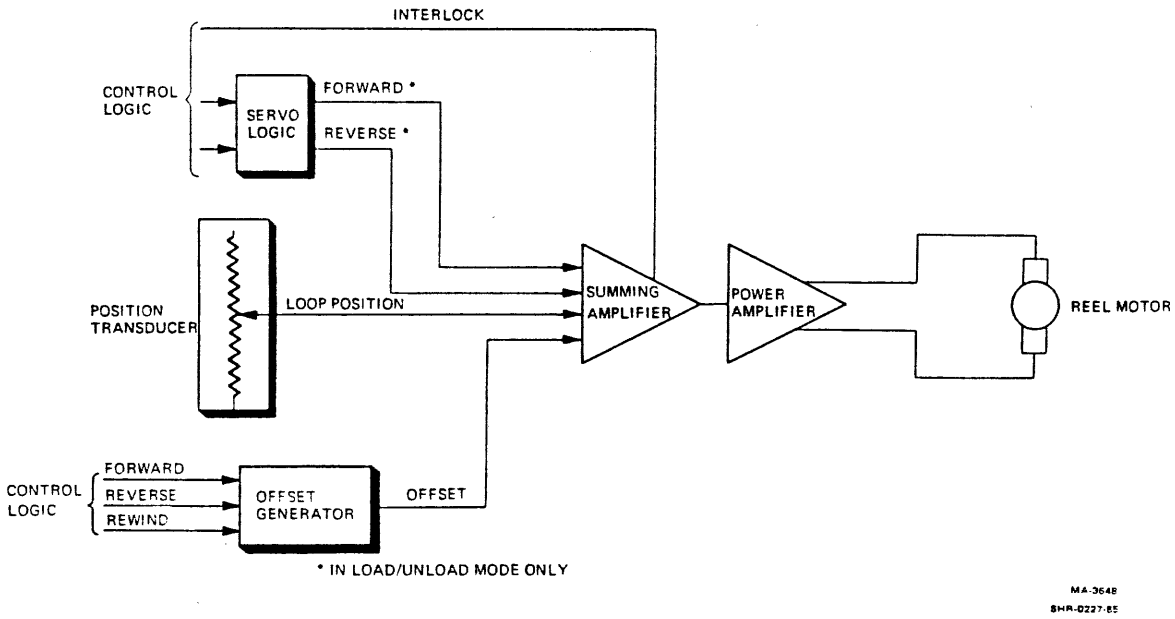
The reel servo subsystem controls tape reel speed as required to maintain optimum tension on the tape between the supply and take-up reels. (The tension is interpreted from the positions of the tape loops.)

Figure 3-26 shows the basic signal flow as applied to each reel servo. The servo loop monitors the power interlock signal and disables the reel servos if the interlock is broken. While loading and unloading tape, the reel motors are controlled by forward and reverse signals from the control logic. When tape is loaded, servo loop control is assumed by the loop position signal from the position transducer. During normal run operations, a loop offset generator generates a signal which offsets the tape loops. The direction of the offset is determined by the direction and type of motion (forward, reverse, or rewind). During rewind, a larger offset is used to allow for the higher tape speed.

If the supply reel is properly loaded, the air system is in operation, and interlock circuit status is acceptable, reel motion begins. The reel rotation speed is subject to feedback information from the air load/control subsystem. This information includes the tape loop position transducer output for the appropriate servo (supply or take-up) and an offset input. The offset input pertains to adjusting reel motion to achieve a different loop configuration for different tape motion directions.

The supply and take-up reel servo networks are identical except for the polarity of the position transducer outputs. The crippled reel signal in the take-up reel servo system is applied to the capstan servo circuitry.

Figure 3-26 Reel Servo Block Diagram



During load and unload operations, at which times the tape path has not stabilized, the reel speed is totally a function of voltage specified by the control electronics. Load and unload sequences are described in Paragraph 3.9. Tables 3-4 and 3-5 show the reel servo inputs and outputs respectively.

Table 3-4 Reel Servo PCBA Inputs

Input	Connector	From	Purpose
+36 V(S)	J13	F7/F9	Supply reel motor power.
—36 V(S)	J16	C5(—)/F11	Supply reel motor power return.
+36 V(T)	J17	F6/F10	Take-up reel motor power.
—36 V(T)	J20	C5(—)/F12	Take-up reel motor power return.
REWR	J12-13	J11-78	Rewind ramp signal from capstan/regulator ramp generator.
REV	J12-33	J8-66	Reverse command from control M PCBA.
NRSAE	J12-11	J8-34	Reel servo enable signal.
NPORST	J12-9,26	J11-37,77	Power-on reset signal.
NTINTLK	J12-31	J2-10	Take-up loop interlock (TLIMIT signal).

Table 3-4 Reel Servo PCBA Inputs (Cont)

Input	Connector	From	Purpose
NSINTLK	J12-30	J21-6	Supply loop interlock (SLIMIT signal).
TRF	J12-15	J8-31	Take-up reel forward command.
TRR	J12-14	J8-32	Take-up reel reverse command.
SRF	J12-17	J8-29	Supply reel forward command.
SRR	J12-16	J8-30	Supply reel reverse command.
MRL	J12-28	J8-5	Midreel load signal.
SPOS	J12-20	J21-37	Supply loop position signal.
TPOS	J12-3	J21-21	Take-up loop position signal.
NDRV	J12-27	J8-66	Drive reel command.
+15 V(L)	J12-6,23	J11-21,61	Regulated dc from capstan/ regulator PCBA.
0 V(L)	J12-4,5, 21,22	J11-26,27, 66,77	DC ground.
−15 V(L)	J12-7,24	J11-14,54	Regulated dc from capstan/ regulator PCBA.
+15 V(R)	J12-32	Interconnect	Power for position transducer.
0 V(R)	J12-34	F1 PCBA	

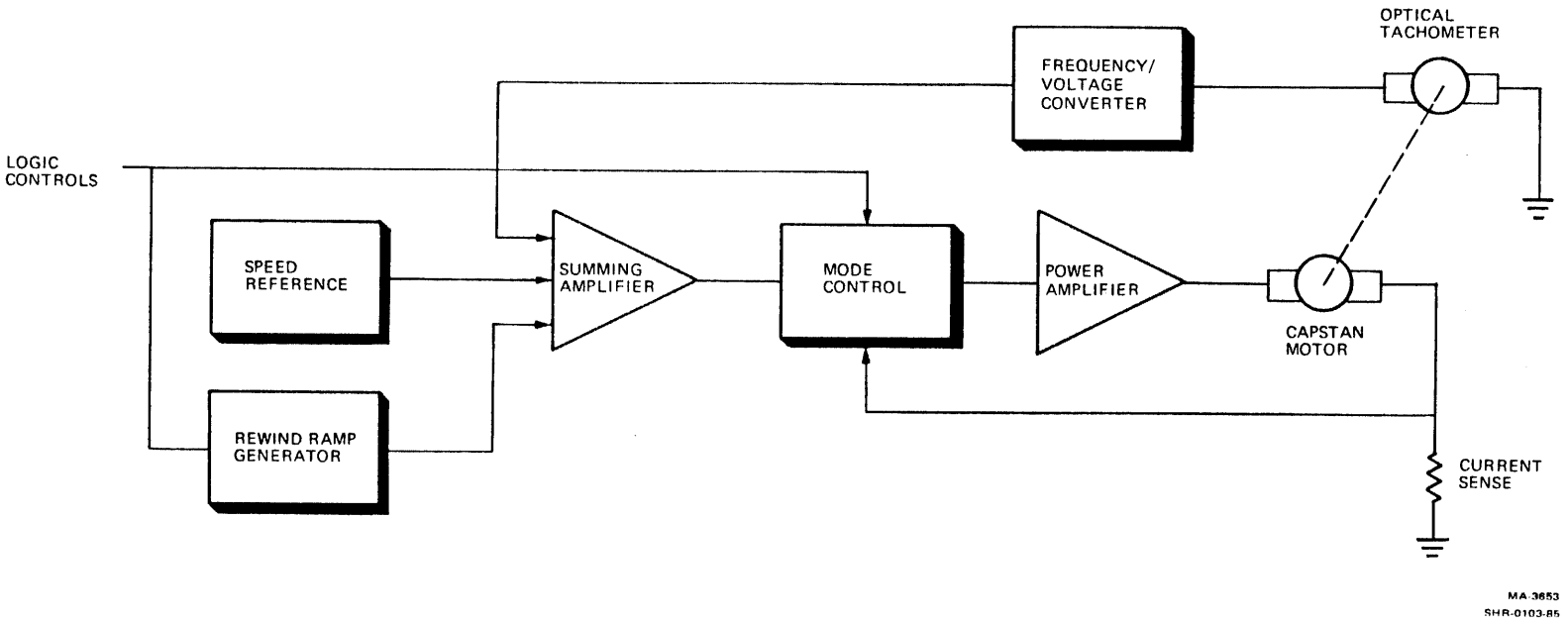
Table 3-5 Reel Servo PCBA Outputs

Output	Connector	To	Purpose
SM(+)	J14	Supply reel motor	Drive power.
SM(−)	J15		
TM(+)	J18	Take-up reel motor	Drive power.
TM(−)	J19		

3.11 CAPSTAN SERVO

The capstan servo is a velocity management system. It acts as the tape mover that pulls tape across the magnetic head assembly for data recording or reproduction. The capstan servo consists of the functional blocks shown in Figure 3-27.

Figure 3-27 Capstan Servo Block Diagram



The heart of the servo is the summing amplifier, which receives current signals from three sources, sums them, and forces the power amplifier to the proper voltage. The power amplifier applies this voltage to the capstan motor, which responds with the appropriate speed. The capstan tachometer is shaft-coupled to the capstan motor and produces a frequency output proportional to the speed of the capstan motor. This frequency is converted to voltage, which is the tachometer feedback required for constant velocity operation.

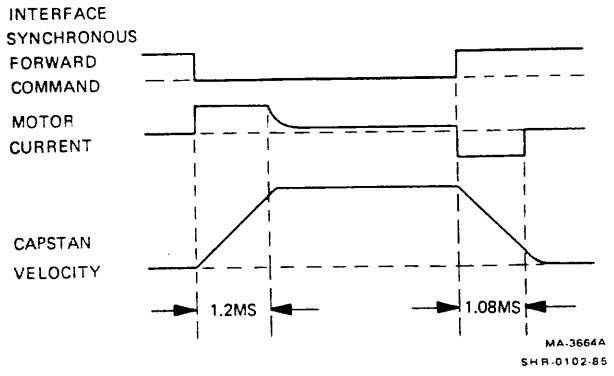
The primary inputs to the capstan servo are the logic control signals. These signals initiate either a positive or negative ramp for forward and reverse operation, or a long rewind ramp used to accelerate the capstan motor to rewind speed. The ramp slopes and final velocities for forward and reverse are adjusted to achieve the desired start/stop characteristics.

Figure 3-28 shows typical capstan servo waveforms. The following sequence of events describes normal capstan servo operation.

1. With power applied and tape loaded interlock made, the capstan power amplifier is enabled.
2. Upon receipt of an ISFC command, capstan drive current is applied in the forward direction. The magnitude of the current is constant and determines the constant rate of acceleration.
3. As the capstan approaches synchronous speed, the difference between the commanded speed and the actual speed decreases, as does the capstan drive speed.
4. A small error signal determined by the loop gain is required to overcome running losses.
5. When ISFC is terminated, the polarity of the drive current is reversed. This reverse current is maintained at a constant level for a fixed time.
6. At the end of the reverse current pulse the capstan motor drive voltage is brought to, and held at, ground level to produce a dynamic braking effect.

NOTE
The synchronous reverse mode (ISRC) procedure is the same except the direction and drive current is reversed.

Figure 3-28 Capstan Servo Waveform



Each time power is applied to the unit, the power distribution block generates a reset signal (NPORST) to the velocity decoder on the control M2 PCBA and the capstan control logic on the capstan/regulator PCBA. After tape is loaded into the vacuum column, the interlock signal (NINTLK) from the air load/control function goes low and enables the capstan amplifier.

The system control function, upon receipt of a motion command, supplies the following signals to the capstan mode control logic: drive (NDRV), direction (REV), motion (NMOT), and speed greater than 80 percent (N>80%). (N indicates low = true.)

The power amplifier provides drive to the capstan motor by means of J11 and TB1-3 on the interconnect D1 PCBA.

Tachometer pulses, from the optical sensor mounted on the capstan shaft, are amplified on interconnect F1 PCBA. The signal is routed through interconnect D1 PCBA to control M2 PCBA where it is squared and sent to the velocity decoder and the interface (TACHP). The velocity decoder generates a 20-microsecond pulse for each tachometer pulse input. This is called NTAP2 and is sent to the capstan/regulator PCBA. There it is converted by the tachometer frequency/voltage converter to the feedback analog (ANALOG TACH) signal for the capstan summing amplifier.

3.12 DATA PATHS

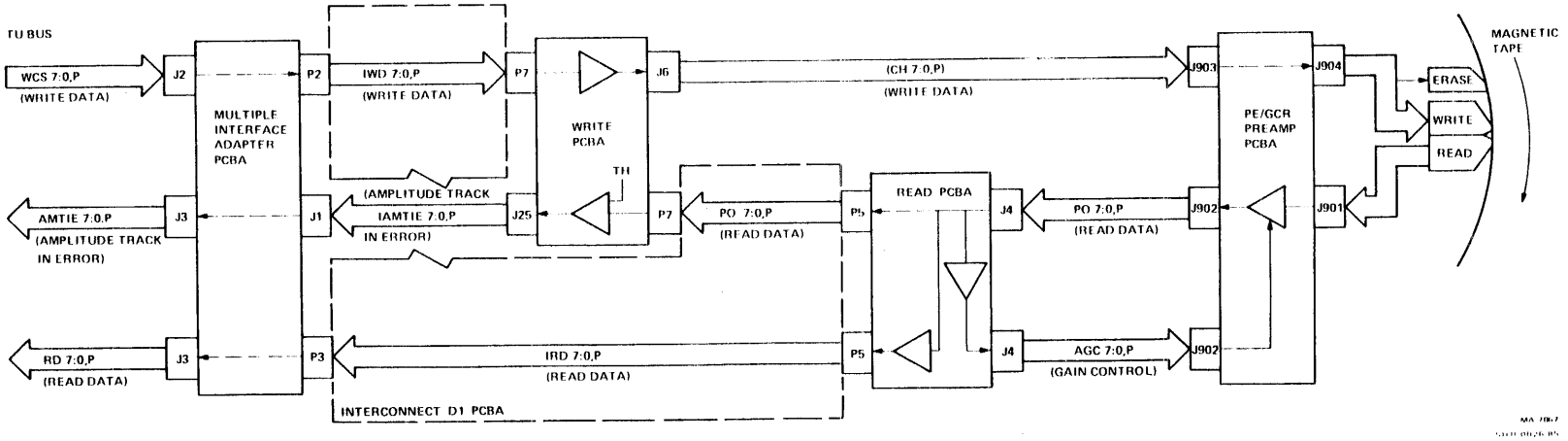
This section outlines the data paths used in write operations, and traces the flow of data onto tape and back through the read recovery chain. For an understanding of the group code recording and phase encoded tape formats, refer to Chapter 1.

3.12.1 Introduction

Figure 3-29 shows an overview of the read/write data paths. Write data originates in the formatter and is placed on the TU bus. The MIA PCBA gates the write data through to the write PCBA where it is conditioned before going through the preamp PCBA to the write head.

Data picked up by the read head is amplified in the preamp PCBA and input to the read PCBA. The read PCBA further amplifies the read data and passes it on to the MIA PCBA where it is gated to the formatter over the TU bus. The read PCBA also establishes a gain control voltage for each track; the voltage is fed back to the preamp PCBA. Read data from the preamp PCBA passes through the read PCBA and is input to the write PCBA. The write PCBA generates an AMTIE signal if the read data amplitude falls below a predetermined threshold level. AMTIEs are gated through the MIA PCBA and sent to the formatter over the TU bus.

Figure 3-29 Data Paths



3.12.2 Write Function

The write function records formatted digital information on tape. The recording format and density are compatible with ANSI and IBM, 9-track PE or GCR. Figure 3-30 is a simplified block diagram of the write function.

Assume power is applied, and a reel with a write-enable ring is installed on the supply reel. Switch S8, on the base assembly, closes and supplies write power (WP1 N.O.) by means of P21-29 and 32 to the capstan/regulator PCBA (J11-69) and to the write PCBA (J7-24 and 60), where it is designated as WRT PWR. If the write-enable ring is not installed on the reel, WP1 N.O. is not applied to the circuitry.

The density signal (NPE) from the control M PCBA (J8-25) is input to the write control logic on the write PCBA at P7-61. The density command is received from the formatter by means of the MIA PCBA. Signal NPE is used to select either PE or GCR format (high = GCR, low = PE).

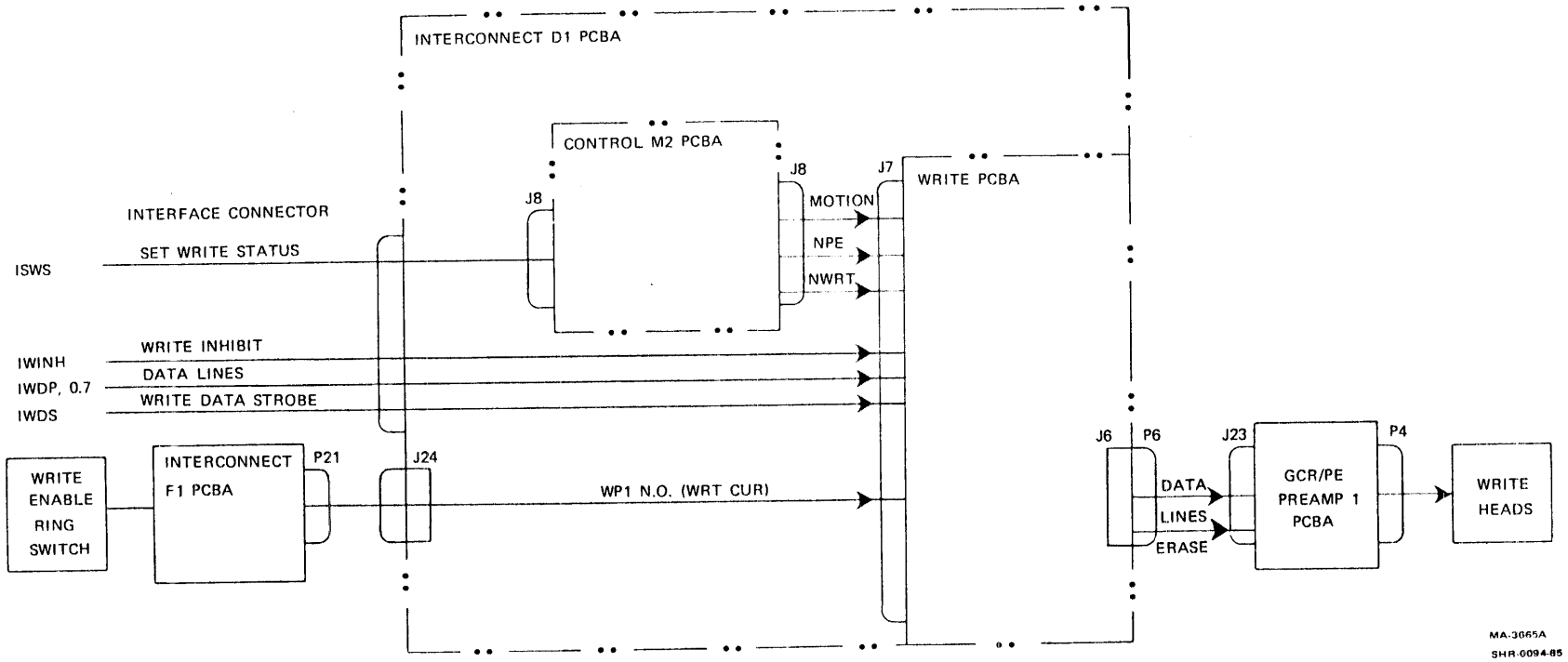
When a load command is detected, the air load/control function provides a low reel servo enable signal (NRSAE) to the capstan/regulator PCBA through J8-34. The WP1 N.O. signal from S8 (P21-29 and 32) and NRSAE from the air load/control function (J8-34) are applied to the write driver on the capstan/regulator PCBA. The write driver provides a low file protect signal (FPT) to the system control function via J11-19, and a low holding path feedback write lockout signal (W.P. SOL RET) to WLO solenoid, K2 via J11-33 and 73.

Signal W.P. SOL RET is coupled through the interconnect F1 PCBA (P21-27 and 28) and energizes the WLO solenoid K2, which holds S8 closed and maintains WP1 N.O. If WP1 N.O. is not present, the write driver provides a high FPT signal to the system control function (J11-19), and a high W.P. SOL RET to the WLO solenoid (J11-33 and 73). The high W.P. SOL RET does not energize the WLO solenoid and no WP1 N.O. is available. TP51 (capstan/regulator PCBA) is used to monitor the file protect signal and TP58 (Capstan/Regulator PCBA) is used to monitor the write lockout feedback signal.

If the transport is selected, ready, and on-line, and the system control function detects both interface write and interface motion commands, the system control function provides a high motion signal (MOTION) through J8-28 and a low write signal (NWRT) through J8-26 to write control logic jacks J7-26 and J7-25, respectively.

The interface input data lines (IWDP, IWD0-IWD7) are routed to the write data buffers through J2 of the MIA PCBA. The MIA PCBA also provides the write data strobe (IWDS) to the write strobe logic through J7-47.

Figure 3-30 Write Function Block Diagram



3.12.3 Read Function

The read function recovers digital information from magnetic tape in either the forward or reverse direction. The format is compatible with ANSI and IBM, 9-track GCR or PE. Figure 3-31 is a simplified block diagram of the read function.

The density signal (NPE) from the control M2 PCBA (J8-25) is applied to the read control logic on the read PCBA. The density command is received from the formatter by means of the MIA PCBA. Signal NPE is used to select either PE or GCR format (high = GCR, low = PE).

When the system control function detects a read command and an interface motion command, the control M2 PCBA provides a high motion signal (MOTION) through J8-28 and a high write signal (NWRT) through J8-26 to the read PCBA. NWRT is used to select either a read or write operation (high = read, low = write).

Because of the read-after-write feature, the transport reads in both read and write modes. Selection of the read-only function is essentially a suppression of the write mode.

Data retrieved by the read heads is preamplified on the preamp PCBA and routed directly to J4 on the read PCBA. There it is processed and presented to the interface in binary form as required by the formatter. The read data is also sent to the write PCBA as pointers (P07:O0, POP) for development of the AMTIE signals. Here the read data is examined for signal amplitude against a reference threshold.

3.12.4 Write PCBA

The write PCBA processes data received from the MIA PCBA, controls the write/erase heads, and checks the read data to provide the amplitude track-in-error (AMTIE) status signals.

The write PCBA plugs into J7 on the interconnect D1 PCBA. Figure 3-32 is a block diagram of circuits on the write PCBA. Write PCBA interface signals are listed in Tables 3-6 and 3-7.

Figure 3-31 Read Function Block Diagram

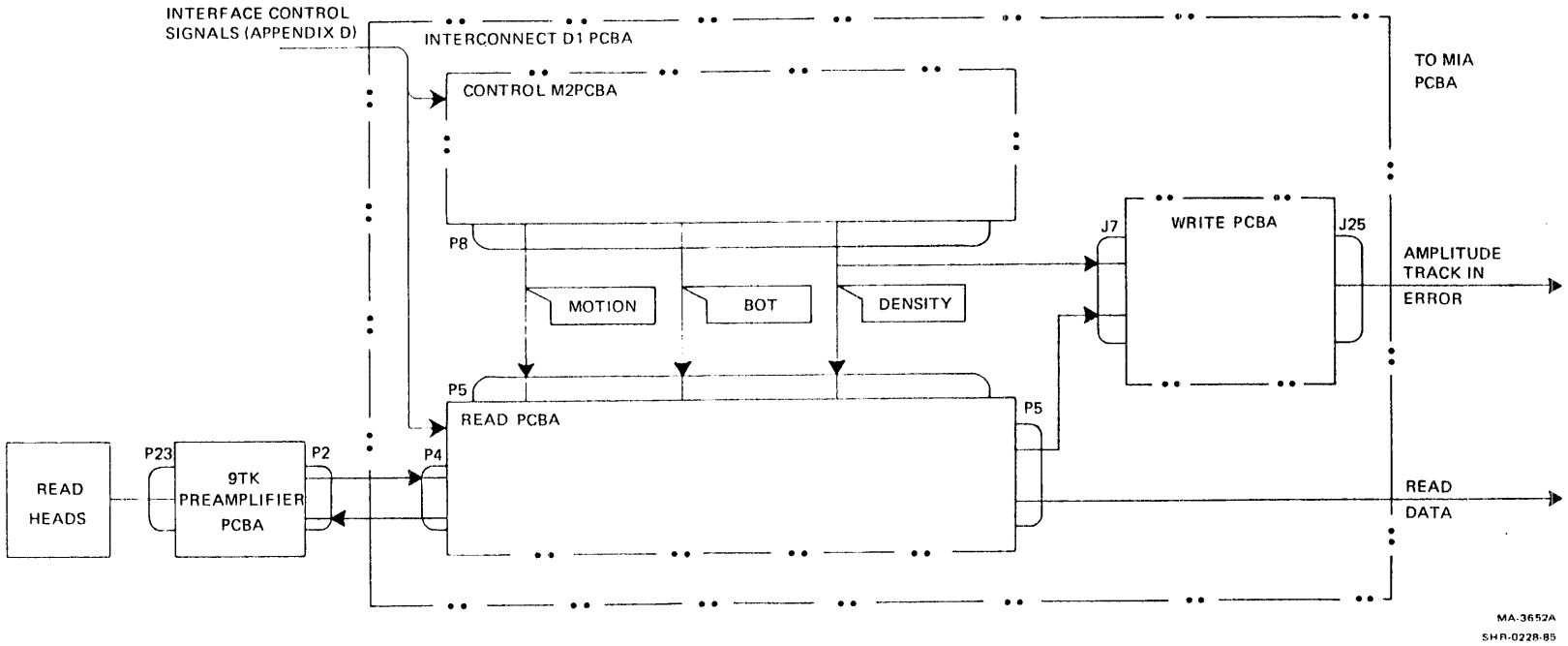


Figure 3-32 Write PCBA Block Diagram

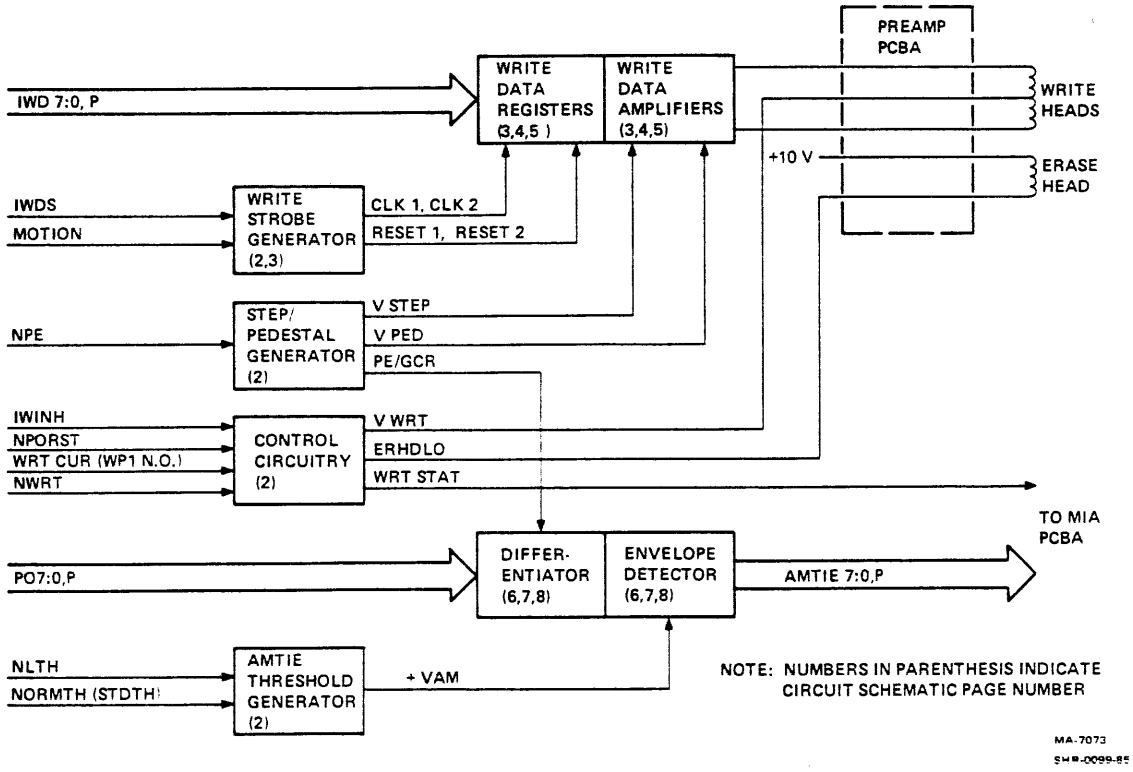


Table 3-6 Write PCBA Inputs

Mnemonic	Connection	From	Purpose
WP1 N.O./			
WRT CUR	J7-60,24	P21-29,32	Write power
NPE	J7-61	J8-25	Density select
NWRT	J7-25	J8-26	Write command
MOTION	J7-26	J8-28	Specifies tape is rolling
IWINH	J25-4, J7-49	MIA J1-4	Write inhibit
NPORST	J7-11	J11-33,77	Power-on reset
NARA	J7-32	J5-17	ARA burst error
NTEST	J7-56	J1-22	Test mode (not used)
NTSTR	J7-20	J3-19	Test data strobe (not used)
IWDS	J7-47	J2-34	Write data strobe
IWD0	J7-44	J2-24	Write data
IWD1	J7-23	J2-23	
IWD2	J7-42	J2-22	
IWD3	J7-41	J2-21	
IWD4	J7-40	J2-20	
IWD5	J7-39	J2-19	
IWD6	J7-38	J2-18	
IWD7	J7-37	J2-1	
IWDP	J7-45	J2-25	
PO0	J7-65	J5-50	Read data bits used
PO1	J7-66	J5-51	as pointers—used for
PO2	J7-67	J5-52	developing AMTIE signals
PO3	J7-68	J5-53	
PO4	J7-69	J5-54	
PO5	J7-70	J5-55	
PO6	J7-71	J5-56	
PO7	J7-72	J5-57	
POP	J7-64	J5-49	
+5 V (T)	J7-21,57	INT. D1	Termination voltage
		W1-2, W2-2	(not used)
+5 V (L)	J7-18,19,	J11-22,23	Logic power
	54,55	62,63	
+15 V	J7-22,58	J11-21,61	Power for erase head amplifier
—15 V	J7-23,59	J11-14,54	Power for erase head amplifier
NLTH	J25-5	MIA J1-5	Establishes AMTIE threshold
STDTH	J25-15	MIA J1-15	Establishes AMTIE threshold

Table 3-7 Write PCBA Outputs

Mnemonic	Connection	To	Purpose
NARA			
(I ARA ERR)	J25-11	MIA J1-11	ARA burst error
MOTION	J25-16	MIA J1-16	Specifies tape is rolling
WRT STAT	J25-14	MIA J1-14	Transport in write mode
WRT BIN	J25-9	MIA J1-9	Status of write data bit 4
(CH0)	J6-8,20	J23-8,17	Write head current
(CH1)	J6-7,19	J23-7,16	
(CH2)	J6-6,18	J23-6,15	
(CH3)	J6-4,16	J23-4,13	
(CH4)	J6-10,22	J23-10,19	
(CH5)	J6-2,14	J23-2,11	
(CH6)	J6-9,21	J23-9,18	
(CH7)	J6-3,15	J23-3,12	
(CHP)	J6-5,17	J23-5,14	
VWRT	J6-1,23	J23-1,20	Write head center tap
ERHD LO	J6-24	E2	Erase head current
WRT REG	J6-11	E1	Erase head current
AMTIE 0	J25-18	MIA J1-18	Amplitude track-in-error status
AMTIE 1	J25-19	MIA J1-19	
AMTIE 2	J25-20	MIA J1-20	
AMTIE 3	J25-12	MI J1-12	
AMTIE 4	J25-3	MIA J1-3	
AMTIE 5	J25-2	MIA J1-2	
AMTIE 6	J25-13	MIA J1-13	
AMTIE 7	J25-1	MIA J1-1	
AMTIE P	J25-17	MIA J1-17	

3.12.4.1 Write Circuitry - The write circuitry comprising the current source, step delay, edge detector, write buffer, and amplifier is shown in simplified form in Figure 3-33.

3.12.4.2 Amplitude Track In Error (AMTIE) Circuits - Amplitude track in error (AMTIE) signals are generated on the write PCBA from read data. They notify the error checking and correction circuits in the formatter of marginal data. Figure 3-34 shows a simplified diagram of the AMTIE generator.

Figure 3-33 Write Amplifier Circuitry

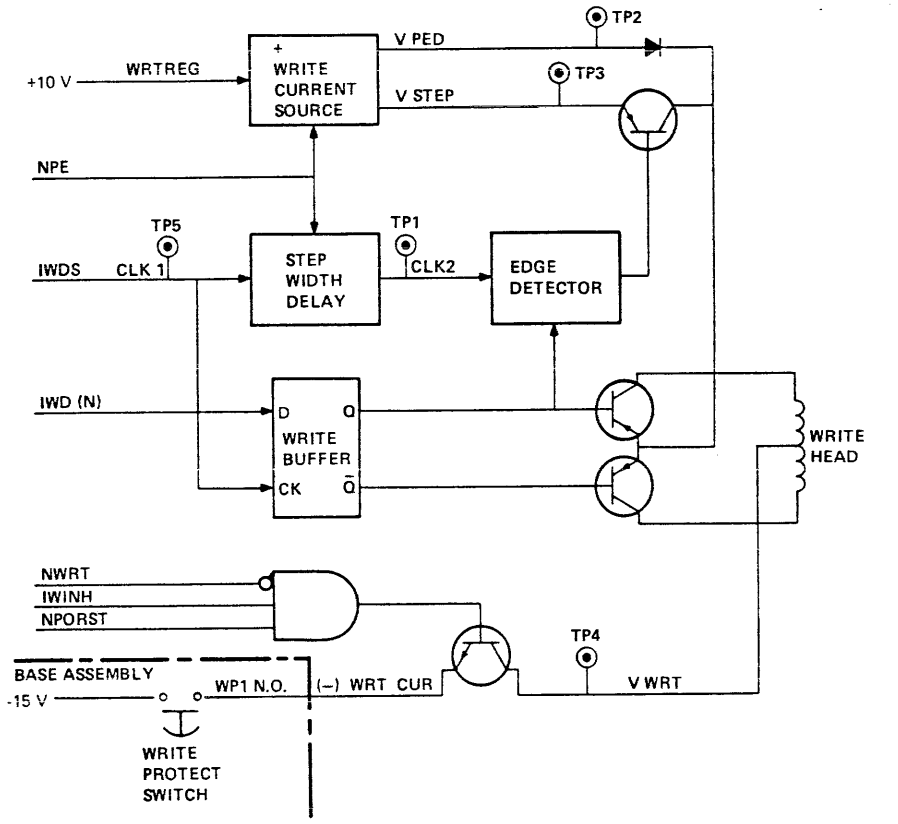
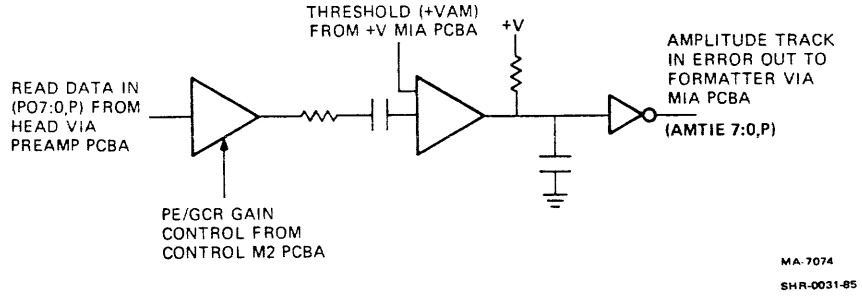


Figure 3-34 AMTIE Generator



3.12.5 Redesigned Write PCBA

All TU78 transports contained the early write PCBA. Many of these TU78 transports were upgraded by FCO TU78-I032 or FCO TA78-J004 (EQ-1455-01). The upgraded TU78 transports and all TU79/TA79 transports contain the new read/write circuitry. The main feature of this circuitry is to provide improved performance. The module contains only one adjustment, and without additional adjustment, is capable of operating reliably over:

1. The entire range of head parameters
2. The allowable range of head wear
3. The range of specified media.

NOTE
The terms "step" and "pedestal" in this section refer to the shape of the write current waveform. The step can be generally thought of as the level of current necessary to saturate the tape (to 105%). The pedestal is a reduced level induced to minimize peak-shift. When driven by a controlled write driver, the step allows operation over a much wider dynamic range of head and tape parameters. Step-and-pedestal is an industry standard technique.

Figure 3-35 is a block diagram of major circuits on the write PCBA.

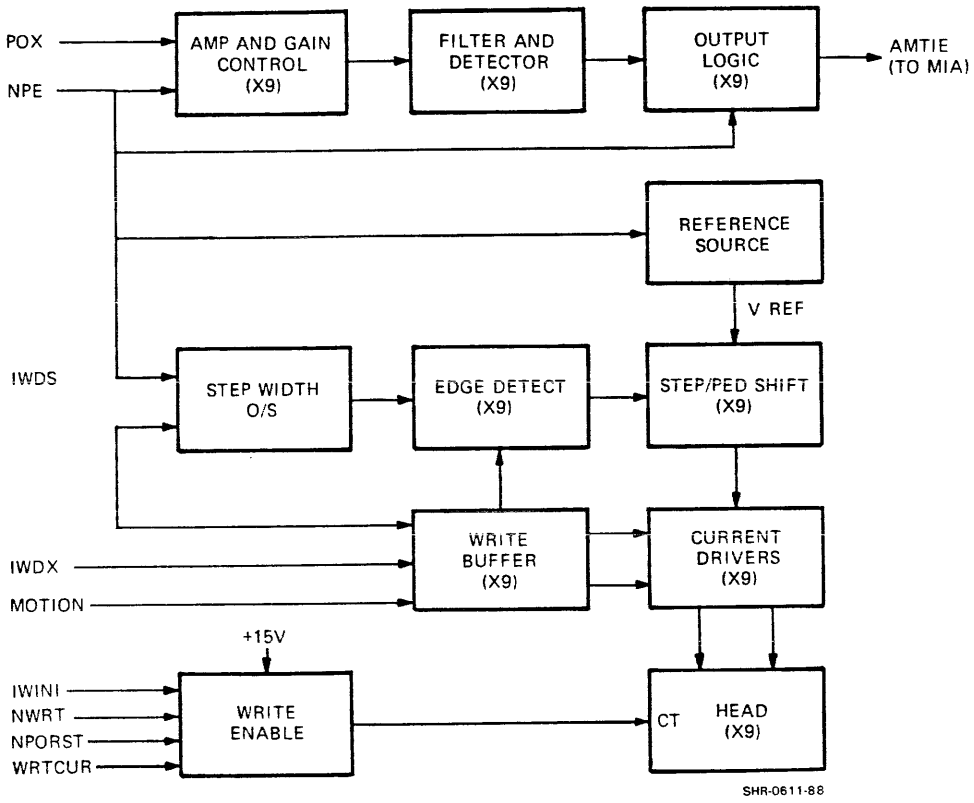
3.12.5.1 Interconnections - Associated interface signals are listed in Tables 3-1 and 3-2.

The write PCBA plugs into J7 on the interconnect D1 PCBA (Figure 3-10). A cable from J1 of the write PCBA to J903 of the read-preamp module connects the write driver outputs to the head. A small portion of the read preamp is used here to:

1. Allow write cable termination to be located near the head
2. Provide a place for the cable connector (J903) to be mounted
3. Provide a convenient mechanical method of mating to the head connector.

A second cable connected from J2 of the write PCBA to J1 of the MIA module contains the AMTIE status signals.

Figure 3-35 Write PCBA Functional Block Diagram



3.12.5.2 Control Inputs - The major control input components are:

1. Inverters U6, U17, U18, and U28
2. Transistors Q28 to Q30, Q32, Q33, and Q37
3. Head CT enabling transistor Q39.

Various signals are brought onto the module to enable (and disable) writing on tape. These signals are:

1. IWINH For data security erase — erase on, write off
2. NWRT Logical write enable — from host
3. NPORST Power-on reset
4. WRTCUR Write protect ring sensor.

In addition to controlling the logical enabling of writing, the above signals provide a fail-safe mechanism to ensure that file-protected tapes do not become corrupted. To enable writing, power is applied to or removed from the head drive circuit, signal VWRT, by the head enabling transistor.

3.12.5.3 Input Data - The major input data components are:

1. Receivers/inverters U2 to U6
2. D-flop latches (1/2 ea) U8 to U16

Input data is received nine bits at a time (one for each track), each bit being clocked into its corresponding input buffer by IWDS. The input buffers are initially cleared by a reset signal derived from motion information.

3.12.5.4 Reference Voltage - The major reference voltage components are:

1. Op-amp U30, power transistor Q31
2. Gain selection Q35, Q36, and Q38
3. Adjustment R103, zener VR1.

The reference voltage is the basis for determining the actual current being driven to the head. There is one reference generator circuit per module, but nine divider circuits, as described in Paragraph 3.12.5.6. Reference voltage is generated by applying a zener voltage to an adjustable-gain op amp. This circuit contains the only adjustment on the module.

Write current adjustment is done in the more critical GCR mode. To obtain the different head currents required for PE, the amplifier gain is changed by a fixed amount. From any given GCR current setting, PE current levels will "track" proportionally.

3.12.5.5 Step-Pedestal Generation - Major step-pedestal generation components are:

One-shots U29
Control logic U1, U25, and U27
D-flops (1/2 ea) U8 to U16
Summing XOR U20, U23, and U25
Reference divider resistors R9, R12, R13, R16, R17, R20, R21, R25, R28, R29, R32, R33, R36, R37, R40, R41, and R44
Reference shifting transistors Q1 to Q9.

While data is clocked, IWDS initiates a one-shot that is used to generate the "step" in the current waveform described earlier. The width of the one-shot determines the width of the step. There are two one-shots per module — one for GCR and one for PE. GCR width is preset to 400 nanoseconds, and PE is preset to 1.25 microseconds. They are not adjustable.

The current level during the "step" portion of the waveform is obtained by applying the full reference voltage to the current source/head driver transistors. After time-out of the one-shot, the current is reduced to its "pedestal" level by altering the reference voltage applied to the current source/head driver transistors. This shifting is done individually for each track, and uses a voltage divider technique. The step-to-pedestal ratio has been set to a fixed ratio of 8:5.7. This ratio remains the same, even though the overall reference changes for GCR and PE.

3.12.5.6 Current Source/Head Driver - The major current source/head driver components are:

Buffers U19, U21, U22, U24, and U26
Coupling resistors R10, R11, R14, R15, R18, R19, R22, R23, R26, R27, R30, R31, R34, R35, R38, R39, R42, and R43
Rise-time limiting caps C20 to C28, C32 to C40
Current source/head driver transistors Q10 to Q27
Current source resistors R51 to R66.

The function of this circuit is to drive a controlled current to the head. *Controlled* write current waveshape is essential for reliable operation over a wide range of head and media parameters and conditions. The reference and step-pedestal circuits mentioned previously generate a controlled waveshape in terms of voltage, which now must be driven to the head in terms of current.

The write head contains nine windings (nine tracks), all with a common center tap. The center tap is connected to +15 V by the enabling circuits mentioned earlier. Current through the head originates at the center-tap and is pulled alternately from each end of the winding (through the cable) by the current source/head driver transistors. These transistors are driven alternately by the open collector inverters connected to the data buffer flop. Pulling current through one end creates north polarization; pulling current through the other end creates south. The rise times for these currents has been purposely limited to approximately 200 nanoseconds to minimize feedthrough or crossfeed into read circuits.

The following two features minimize distortion due to wavelength:

1. The maximum allowable cable length is four feet.
2. Termination resistors are placed at the head end of the cable, not on the module.

3.12.5.7 Erase Circuit - The major erase circuit components are:

Current limiting resistors R114 and R115
Blocking diode CR 26

The erase circuit is quite simple. One end of the erase head is connected to +15 V by a current-limiting resistor. The other end of the erase head connects to —15 V by the control circuits mentioned earlier.

3.12.5.8 AMTIE Circuits - The major AMTIE circuit components are:

Three-terminal supply regulators Q41 and Q101
Amplifier/differentiator U31 to U39
Control transistors Q42 to U59
Threshold comparators U40 to U48
Detector filters C146, C148 to C152, C159 to C164, C167 to C172, R250 to R255, R276 to R287
Filter selector U50, U51, U53, U54, U59, U60, and U61
Output driver U56 and U58

The AMTIE (amplitude track in error) circuits are not part of the write system. They are located on the write PCBA only because of available space.

The AMTIE circuits notify the error checking (and correction) circuits in the formatter of marginal low amplitude data. Input signals come from the read PCBA but are essentially the same as preamp output signals.

A selectable gain amplifier-differentiator normalizes between GCR and PE read signals. This normalized signal is then checked against a threshold voltage. Threshold voltages are generated by logically selecting various zener diodes depending on the present mode of operation, reading or writing. A special low threshold is also available for a last-ditch effort to recover bad read data.

An envelope detector (a filter, selectable for GCR and PE) is then used to indicate missing or low-amplitude data. The output is an open-collector logic signal that is cabled to the MIA module.

3.12.6 Read PCBA

The read PCBA accepts signals from the read heads by means of the GCR/PE preamp 1 PCBA and decodes the signals to provide the formatter with read data in usable form. It also provides the preamp with a feedback gain control voltage. Finally, it routes the read signals to the write PCBA for use in generation of AMTIE signals. Table 3-8 lists inputs and Table 3-9 lists outputs for the read PCBA.

Table 3-8 Read PCBA Inputs

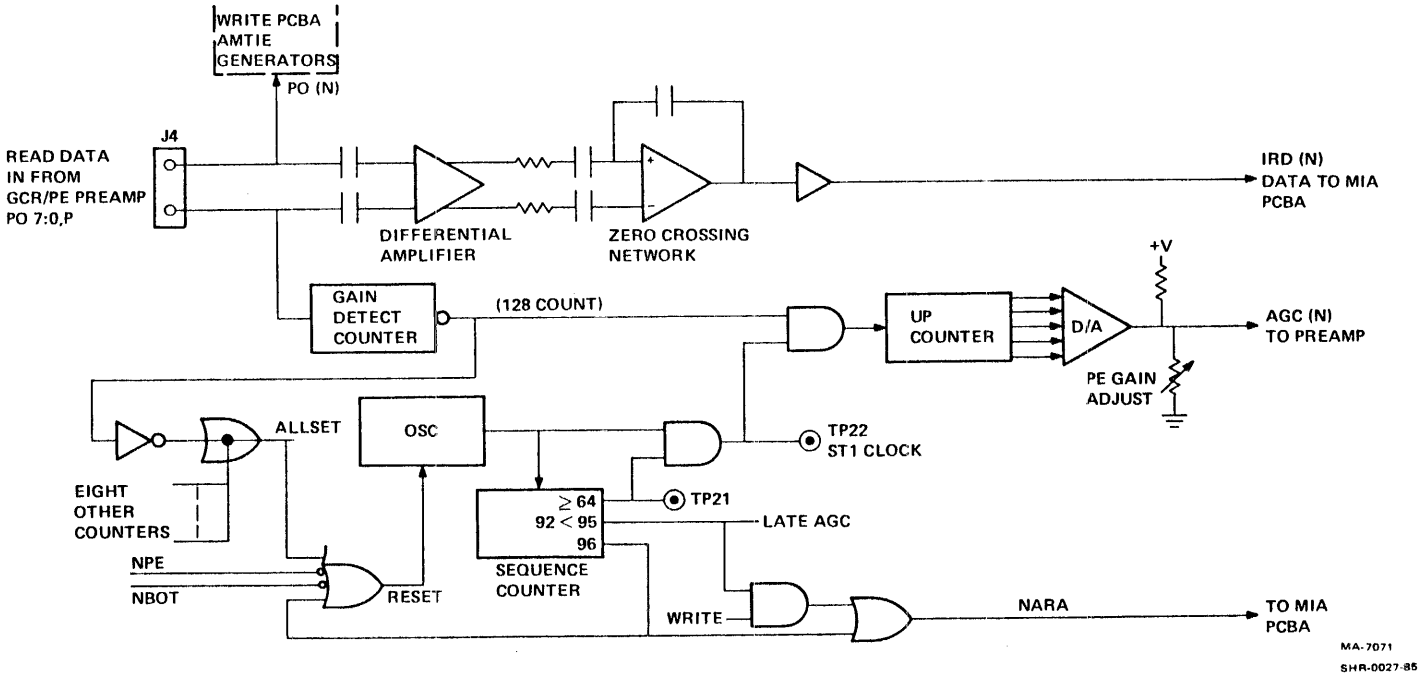
Mnemonic	Connection	From	Purpose
NPE	J5-10	J8-25	PE/GCR density select
NWRT	J5-47	J8-26	Write/read mode
BOT	J5-12	J8-48	Beginning of tape
PO0	J4-6,22	GCR/PE	Analog read data
PO1	J4-7,24	Preamp 1	
PO2	J4-9,25	PCBA, J902	
PO3	J4-42,28		
PO4	J4-3,19		
PO5	J4-31,32		
PO6	J4-4,21		
PO7	J4-13,30		
POP	J4-10,27		
+15	J5-5,41	J11-21,61	Regulated precision voltage
—15	J5-6,42	J11-14,54	Regulated precision voltage

Table 3-9 Read PCBA Outputs

Mnemonic	Connection	To	Purpose
+15	J4-16,33	GCR/PE preamp 1	Regulated voltage
—15	J4-15	PCBA, J902	
NARA			
(IARA ERR)	J5-17	J1-11	ARA burst error
Late AGC	J5-37		not used
AGC0	J4-5	GCR/PE preamp 1	Automatic gain voltage
AGC1	J4-23	PCBA, J902	
AGC2	J4-8		
AGC3	J4-11		
AGC4	J4-2		
AGC5	J4-14		
AGC6	J4-20		
AGC7	J4-29		
AGCP	J4-26		
PO0	J5-50	J7-65	Read data for AMTIES
PO1	J5-51	J7-66	
PO2	J5-52	J7-67	
PO3	J5-53	J7-68	
PO4	J5-54	J7-69	
PO5	J5-55	J7-70	
PO6	J5-56	J7-71	
PO7	J5-57	J7-72	
POP	J5-49	J7-64	
IRD0	J5-70	J3-32	Read data output
IRD1	J5-69	J3-31	
IRD2	J5-68	J3-27	
IRD3	J5-67	J3-26	
IRD4	J5-62	J3-21	
IRD5	J5-61	J3-20	
IRD6	J5-59	J3-18	
IRD7	J5-58	J3-1	
IRDP	J5-72	J3-34	

The read PCBA plugs into J5 on the interconnect D1 PCBA. A simplified functional block diagram of the read PCBA is shown in Figure 3-36.

Figure 3-36 Read PCBA Functional Block Diagram



APPENDIX A
GLOSSARY OF TERMS AND MNEMONICS

A.1 GENERAL NOTES

1. I, prefixed to a mnemonic term, designates an interface I/O signal. If the term contains a suffix R (receiver), the signal is an input (with respect to the tape transport). Similarly, a suffix D (driver) implies an output signal. Low = true for all I/O signals.

   NOTE
   Do not confuse the above terms with specialized uses of I, R, and D, such as I for inverted in NRZI, R for read, and D for data.

2. N, prefixed to a mnemonic term, has a meaning similar to a logic bar or not symbol. N implies that the true signal identified by the remainder of the term is electronically low (low = true) at the critical point in the circuit identified by the term.
3. Status signals are true if the condition monitored is true; for example, LOAD FAULT is true if the tape is improperly loaded.
4. Mnemonics for signals derived from circuits controlled by switches (automatic as well as manual) may include N.O. (normally open) or N.C. (normally closed).
5. D, prefixed to a term, means the signal has been delayed with reference to the signal identified by the remainder of the term.

Term or Mnemonic	Definition
8MC	8085 microcomputer (M8960).
9TK	Nine track (always high).
9TK N.O.	Nine track normally open (not used).
A	Address lines or address on address lines.
ABP	Air bearing pressure.
ACRC	Auxiliary cyclic redundancy check.
Address	A specific memory or peripheral register location.
AGC	Automatic gain control.
AGC0—AGC7	Automatic gain control voltage for read Amplifier 0—7.
AGCP	Automatic gain control voltage for the parity channel read amplifier.
AMTIE	Amplitude track in error.
AMTIE P	Amplitude track in error interface signal for the parity read channel (low = true).
AMTIE0—AMTIE7	Amplitude track in error interface signals for read channels 0—7 (low = true).
ANSI	American National Standards Institute.
ARA	Automatic read amplification.
Architecture	The internal configuration of a processor, system, or subsystem.
ASCII	American Standard Code for Information Interchange.
Asynchronous	A mode of operation scheduled by ready and done signals rather than by time intervals.
ATTN	Attention.
BAL	Byte assembly logic (M8959).
BCD	Binary coded decimal.
BCTC	Byte count terminal count.
Bit	A binary number, either 0 or 1.
Bit/In	Bits per inch (also known as characters per inch).
BKW	Backwrap.
Block	A record (sometimes considered to be multiple records within a file).
BOT	Beginning of tape.

Term or Mnemonic	Definition
BPI	Bits per inch.
Buffer	A temporary storage area or an isolating circuit used to avoid distortion of the input signal by the driven circuit or transmission line.
Bus	A group of signal lines that convey information from source to destination. There may be many destinations, but only one source at a given time.
Byte	A word consisting of eight bits.
C TO D	Controller To Drive.
C0—C4	Load/unload sequence count 0 through 4 (high = true).
CART MTR	Cartridge motor supply voltage.
CART N.O.	Cartridge normally open.
CART PRESS N.O.	Cartridge pressure.
CART SOL RET	Cartridge solenoid return.
CAS	Common address space (M8957).
CC N.O.	Cartridge closed (low = true).
Char/In	Characters per inch (also known as bits per inch).
Character	A single letter, numeral, or symbol used to represent information.
CLK	Clock.
CLKA	Clock A (1 MHz).
CLKB	Clock B (100 kHz).
CLKC	Clock C (10 kHz).
CLKD	Clock D (100 Hz) not used.
CLKE	Clock E (10 Hz).
CLKF	Clock F (1 Hz).
CLKG	Clock G (0.5 Hz).
CLR	Clear.
CM	Capstan motor voltage.
CO N.O.	Cartridge open (low = true).
Command	A word, byte, or portion of a byte that causes a predefined operation to be performed.
CPAR	(Massbus) control bus parity error.

Term or Mnemonic	Definition
CPU	Central processor unit (this is the 8085 microprocessor chip relative to the TS78 controller).
CR N.O.	Crippled reel normally open (low = true).
CRC	Cyclic redundancy check.
CRCC	Cyclic redundancy check character.
D	Data lines or data on data lines.
Data Group	In GCR mode, seven data characters plus an ECC character combined as a group prior to translation. Consists of two data subgroups.
DDI	Data density indicator signal.
DDR	Diagnostic data register.
Decoder	A device that detects a specific address and produces a corresponding single output.
DEM	Demand.
Density	The nominal distribution per unit length of recorded information, usually expressed in bits or characters per inch.
DINTLK	Delayed interlock pulse.
DIP	Dual in-line package.
DMA	Direct memory access. A process whereby data is transferred rapidly into RAM by the host computer without microcomputer control.
DONL	Delayed on-line signal.
Down-Line Load	See DMA.
DPAR	(Massbus) data bus parity error.
Drive	The mass storage tape subsystem (sometimes used to refer to a tape transport).
drive	Combination of a formatter and one transport.
DRV	Drive.
DS	Drive select.
DSE	Data security erase.
DT	Drive type.
EBL	End of block.

Term or Mnemonic	Definition
EC	Error correction.
ECC	Error correcting code (M8951).
ECC Character	A special error correcting code character on tape, used for error detection and correction.
Ecode	Error code. This code is developed in the write microcontroller and sent to the microcomputer on demand.
EOT	End of tape.
ERHDLO	Erase head low.
EXC	Exception.
F	Frequency.
F/V	Frequency-to-voltage converter.
Fc/In	Flux changes per inch.
FF	Flip-flop.
FIFO	See Shift Register.
File	A logical collection of data treated as a unit. A file may consist of one or more records or blocks on tape.
File Gap	A special extended, erased portion on tape just before the file or tape mark record.
FMK	File mark (also tape mark).
Format	The entire set of unique parameters used to define the recording mode.
FPT	File protected (high = true).
Fr/In	Flux reversals per inch.
FWD	Forward.
GCR	Group-coded recording.
GCTC	Group count terminal count.
Group	See Data Group or Storage Group.
Group-Coded Recording	A recording technique that collects groups of characters and translates or encodes them before putting them on tape.
HHT	Handheld terminal.
HI DEN N.O.	High density normally open (not used).
HLD A	Hold acknowledged.

Term or Mnemonic	Definition
Hold	A state in which the microcomputer may be placed during a host-to-RAM DMA transfer.
HSC	Hierarchical storage controller.
I/O	Input/output.
I/O Page	That portion of memory in which specific locations are associated directly with subsystem peripheral device registers.
IBG	Interblock gap (see Interrecord Gap).
IBOT	Beginning of tape interface output (low = true).
IC	Integrated circuit.
ICMD L	Command strobe (low = true).
ICMD PE L	Command parity error (low = true).
ID	Identification.
IDB	Identification burst.
IDDI	Data density indicator signal output (low = PE, high = GCR).
IDDS	Data density select input.
IDDSE	Data density select enable (always low).
IEOT	End of tape output signal (low = true).
IFPT	File protect signal at interface (low = true).
ILDP	Load point (low = true when tape is at load point).
ILEXER	In-line exerciser.
ILR	Illegal register.
ILTAPE	In-line tape diagnostic.
In/Sec	Inches per second.
INIT	Initialize.
Initialize	To set certain logic elements to a known state, or to set program variables to their starting values before issuing a subsystem command.
Interrecord Gap	A blank space deliberately erased between data records on tape.
Interrupt	A hardware signal that causes the microprocessor to jump to a specific subroutine.

Term or Mnemonic	Definition
Interrupt Service Routine	The routine or subroutine executed when an interrupt occurs.
INTLKP1	Pulsed interlock (occurs after INTLK before DINTLK).
IONL	On-line signal at interface (low = true).
IPS	Inches per second.
IRD0—IRD7	Read data bits 0 through 7 output, respectively. Interface signal (low = true).
IRDP	Read data parity output. Interface signal (low = true).
IRDY	Tape transport ready output. Interface signal (low = true).
IRG	Interrecord gap.
IRTH2	Read threshold level 2 (not used).
IRWC	Rewind command input. Interface signal (low = true).
IRWD	Rewinding signal output. Interface signal (low = true).
IRWU	Rewind and unload command input. Interface signal (low = true).
ISFC	Synchronous forward command input. Commands tape forward motion for either reading or writing (low = true).
ISLT0	Select port A input. Interface signal (low = true).
ISLT1	Select port B input. Interface signal (low = true).
ISLT2	Select ports A/B input. Interface signal (low = true).
ISLT3	Select maintenance input. Interface signal (not used).
ISRC	Synchronous reverse command input. Commands tape motion in reverse at reading speed..
ISTAT L	Status strobe (low = true).
ISWS	Set write status. Interface signal (low = true).
ITACH	Tachometer output. Interface signal (low = true).
IWCS0—IWCS7	Write/command/status bits 0—7.
IWCSP	Write/command/status parity bit.

Term or Mnemonic	Definition
IWD0—IWD7	Write data bits 0 through 7, respectively. Interface signal (low = true).
IWDP	Write data parity. Interface signal (low = true).
IWDS	Write data strobe. Interface signal (low = true).
IWINH	Write inhibit (low = true).
Latch	A circuit that, when triggered, stores whatever appears on its inputs and holds it. When not triggered, the output reflects the input.
LATE AGC	Late automatic gain control setup (not used).
LD/REW	Load/rewind.
LDF0	Load fault zero.
LDFS	Load fault status.
LDS	Load status.
LED	Light emitting diode.
LO DENSITY	Low density (for 6250 indicator).
LOAD FAULT	Load fault warning signal.
LSB	Least significant bit or least significant byte.
LSI	Large scale integration.
LWR	Loop write to read.
MB	Massbus.
MBC	Massbus control (M8957).
MBD	Massbus data (M8956).
MCPE	Massbus control bus parity error.
MIA	Multiplexer interface adapter.
Microcomputer	A class of computer having all major central processor functions, memory, and control circuitry self-contained on a single printed circuit board. A microcomputer usually employs a microprocessor chip.
Microcontroller	A semi-intelligent set of logic that performs simple control functions directed by a microprogram.
Microprocessor	A single LSI circuit that performs the functions of a CPU.

Term or Mnemonic	Definition
Microprogram	A program implemented in microcode for a microprocessor or microcontroller. A microprogram may reside in ROM or RAM.
MM I/O	Memory-mapped I/O.
Modulo	A mathematical operation that yields the remainder function of a division.
Monitor	In an operating system, the master control program that checks, controls, or verifies the operation of a computer system. The set of routines that allocates resources, performs I/O, and, in general, controls the operation of user and system programs, schedules and operation.
MOT	Motion.
MRL	Mid-reel load.
µs	Microsecond.
ms	Millisecond.
MSB	Most significant bit or most significant byte.
MUX	Multiplexer.
N>80%	Capstan speed is greater than 80% (low = true).
NAE	Reel servo amplifier enable.
NAOK	Air okay (low = true).
NARA	Automatic read amplification burst error (low = true).
NBOT	Beginning of tape (low = true).
NC0—4	Load/unload sequence count 0 through 4 (low = true).
NCCC	Cartridge closed command (low = true).
NCOC	Cartridge open command (low = true).
NDINTLK	Delayed interlock signal (low = true).
NDRV	Capstan/reel servo drive (low = true).
NED	Nonexistent drive.
NEF	Nonexecutable function.
NEOT	End of tape (low = true).

Term or Mnemonic	Definition
NGOP	Go pulse (low = true).
Nibble	The low-order or high-order 4-bit half of one byte.
NLDC	Load clock (low = true).
NLDFS	Load fault status (low = true).
NLDP	Load pulse (low = true).
NLDS	Load status (low = true).
NLRSTP	Load reset pulse (low = true).
NLTH	Low AMTIE threshold.
NLTP	Low tape pulse (low = true).
NMOT	Motion (low = true).
NMRL	Mid-reel load (low = true).
NMRSTP	Master reset pulse (low = true).
NORMTH	Normal (standard) AMTIE threshold.
NPE	Phase encoded (low = true).
NPOL	Pressure on-low (low = true).
NPORST	Power reset, reset signal generated when power is turned on (low = true).
NREWP	Rewind pulse (low = true).
NRSAE	Reel servo amplifier enable (low = true).
NRST1	Reset 1 (low = true).
NRTY	Retry (repeat load attempt) (low = true).
NRWC	Rewind command (low = true).
NRWR	Rewind ramp (low = true).
NRWS	Rewind status (low = true).
NRWT	Write signal (low = true).
NRZI	Non-return to zero, inverted.
NSFC	Synchronous forward command to move tape in forward direction at reading and writing speed (low = true).
NSMRL	Small reel sense signal (low = true).
NSRC	Synchronous reverse command for reverse motion at reading speed (low = true).

Term or Mnemonic	Definition
NTAP2	Tach pulse 2 (squared and set at 20 us pulses) (low = tach mark sensed true).
NTEN	Tach enable (low = true).
NTEST(A)	Test signal (low = true) (not used).
NTHD	Thread command.
NTIP	Tape in path (low = true).
NTSTR	Test write strobe (low = true) (not used).
NULC	Unload command (low = true).
NULRW	Rewind and unload command (low = true).
NUNLC2	Unload command 2 (low = true).
NXFR	Transfer, energize vacuum transfer solenoid (low = true).
OCC	Occupied.
ONL	On-line.
OPI	Operation incomplete.
P N.O.	Pressure normally open.
Pad Character	A null or zero character in the residual data group.
Page	A set of 32K contiguous byte locations.
Parity Bit	A binary digit appended to a group of bits to make the sum of all the bits always odd (odd parity) or always even (even parity). The parity bit is used to verify data integrity.
PCB	Printed circuit board.
PCBA	Printed circuit board assembly.
PE	Phase encoded.
peripheral	Any device responding to an address in the I/O page.
PKSN	Pack sense.
PLS	Pulse.
PNU	Pneumatic (blower system on) command.
PNU RETURN	Pneumatic system return signal.
PO0—PO7	Preamplifier output analog data for read channels 0—7.

Term or Mnemonic	Definition
Polling	A process in which the microcomputer interrogates certain command or status registers one at a time to determine if service is required.
POP	Preamplifier output analog data for the parity channel.
Port	A place through which input/output data is channeled.
Postamble	Groups of special characters recorded on tape at the end of a data record for synchronization.
Preamble	Groups of special characters recorded on tape at the beginning of a data record for synchronization.
PRES SOL RET	Pressure solenoid return.
Priority	The order in which the microcomputer satisfies simultaneous interrupt requests.
Program	A set of sequential instructions that a microprocessor or microcontroller follows.
Program Counter	A special register in the microprocessor or microcontrollers that points to the location in memory of an instruction to be executed.
PROM	Programmable read only memory.
Protocol	The set of system or subsystem rules that define handshaking on a bus.
PRST	Preset.
PSOL	Pressure solenoid command.
PWR	Power okay.
R/W	Read/write.
RAM	Random access memory (see Read/Write Memory).
RC	Read channel (M8950).
RD	Read data.
RD0—RD7	Read data bit 0 through 7, respectively. (high = true).
RDP	Read data parity bit.
RDY	Ready.
Read-Only Memory	A memory in which the contents are not intended to be altered during normal operation.

Term or Mnemonic	Definition
Read/Write Memory	A RAM in which each cell may be selected by applying appropriate electrical input signals. The binary state of the cell may be either (1) sensed at appropriate output terminals, or (2) changed in response to other similar electrical input signals.
REC	Record.
Record	A group of multiple characters on tape, taken in succession. Records are separated on tape by interrecord gaps.
Register	A device that stores data.
Resident	Present in the subsystem or its memory.
REV	Reverse.
REW	Rewind.
RMC	Read (path) microcontroller.
ROM	See Read-Only Memory.
RP	Read path (M8953).
RS	Register select.
RST	Reset.
RWD	Rewind.
RWND	Rewind.
RWS	Rewind status.
Scratchpad	An area of RAM memory set aside for short and often-repeated calculations, or temporary storage.
SEL	Select.
SFBRK	Supply reel forward brake.
Shift Register	A memory in which data enters serially and shifts to successive storage locations. Data may be read when it has sequentially shifted to the output (also known as a FIFO).
Skew	The deviation of bits within a tape character from the intended or ideal placement, which is perpendicular to the reference edge.
SLIMIT	Supply tape — loop limit.
SLT A	Transport selected.

Term or Mnemonic	Definition
SLT COM	Select common (always low).
SM	Supply reel motor voltage.
SN	Serial number.
SPOS	Supply (loop) position.
SRBRK	Supply reel reverse brake.
SRF	Supply reel forward.
SRR	Supply reel reverse.
Stack	Storage in RAM memory for data during subroutines or interrupts.
STDTH	Standard (normal) threshold.
STI	Standard tape interface (TA79).
STL	Set loops.
Storage Group	Ten tape characters created from the 8-character data group via the record code value translation.
Subgroup	One half of a data or storage group.
Subroutine	A program within a program that performs a specific, often-used function.
Synchronous	Simultaneous performance of a sequence of operations controlled by an external clock.
TACH	Tachometer.
TACHP	Tachometer output pulse to interface.
Tape Mark	A special control record on tape which serves to separate files.
TAPEN	Tachometer pulse enable.
TC	Terminal count.
TCU	Tape control unit.
TFBRK	Take-up reel forward brake.
THDS	Thread status.
TIP	Tape in path.
TLIMIT	Take-up tape — loop limit.
TM	Tape mark (also known as file mark).

Term or Mnemonic	Definition
TOR	Tape on reel.
Total Character Skew	The deviation, during reading, from time coincidence of the bits within a recorded character; the sum of static skew and dynamic skew.
TPOS	Take-up (loop) position.
TRA	Transfer.
Trap	An urgent microprocessor interrupt used to flag a power failure or system monitor parity error.
TRBRK	Take-up reel reverse brake.
TRF	Take-up reel forward.
Tri-State	The property of a bus signal line to be placed in a dissociated (high-impedance) state, as well as the common HI or LO state.
TRR	Take-up reel reverse.
TSTDAT	Test data (not used).
TU BUS	Tape unit bus.
TUP	Tape unit port (M8955).
TWDS	Test write data strobe.
UNL	Unload.
VAC	Vacuum applied.
VAC SOL RET	Vacuum solenoid return.
VCO	Voltage-controlled oscillator.
Vector	A specific address loaded into the microprocessor's program counter to force it to start processing at that address.
VPED	Pedestal voltage.
VSTEP	Step voltage.
VWRT	Write voltage.
WCS	Writable control store; RAM storage area that may be overwritten with microprogram information.
WD	Write data.
WDS	Write data strobe.
WMC	Write microcontroller (M8959).

Term or Mnemonic	Definition
Word	Any group of bits indicating a single number or expression.
WP SOL RET	Write protect solenoid return.
WP1NO	Write protect solenoid set.
WRT	Write or write signal.
WRT BIN	Write binary track 4.
WRT CUR	Write current.
WRT REG	Regulated write voltage.
WRT STAT	Write status (high = true).
XMC	Translator microcontroller (M8958).

APPENDIX B
STI BUS AND MASSBUS REGISTER DESCRIPTIONS

B.1 INTRODUCTION

Section B.2 of this appendix shows register descriptions for STI tape subsystems. Section B.3 shows Massbus register descriptions.

B.2 STI BUS REGISTER DESCRIPTIONS

This appendix lists the bytes in the extended drive and extended formatter status areas. These bytes are accessible via the HSC and the ASCII port. The HSC may request the bytes in any area as a group. In the listing, the byte number identifies the order in which the TA78/TA79 transmits these bytes to the HSC. Via the ASCII port, these bytes may be accessed individually with the parameter addresses listed.

Hardware registers are the source for many extended drive status bytes. The microbus addresses listed are the 8085 I/O locations of these registers. Bytes that do not have a microbus address listed come from RAM locations in the M8972 module. This is the case for all bytes in the extended formatter status area. Additional front panel error codes are in Table B-1.

Table B-1 TS78 Control Panel Error Code

Indicator Display			Fault	Associated Microdiag-	Possible Cause	Related
FAULT	A	B	Number	nostic Chain	General Area	Modules
Off	Off	On	1	401	Miscellaneous	—
Off	On	Off	2	402	Read	M8950 (9) M8953
Off	On	On	3	403	Write	M8959 M8958
On	Off	Off	4	404	TU port	M8955 (2)
On	Off	On	5	405	Error correction	M8951
On	On	Off	6	406	STI communication	M8970 M8971
On	On	On	7	407	Microcomputer	M8972

NOTES

1. On = indicator blinking
   Off = indicator always off
2. Fault number 0 is never used.

B.2.1 Extended Drive Status

Byte number 1

Parameter address	24
Name	— (Diagnostic mode status byte 1)
Microbus address	—
Register location	—

Bit	Name	Description
7:0	—	With TA79 in diagnostic mode, byte 1 is one of two status bytes for classifying soft errors. One of 6 codes, defined below, may appear in this byte. In normal operation, byte 1 is a 0 (see note below byte 4).

Partial Diagnostic Mode Soft Error Status

Code	Meaning
00	No soft error (see note 1)
07	Other (write) (see note 2)
0E	Status (read) (see note 3)
16	Status (read reverse) (see note 3)
2D	Transmission (read)
35	Transmission (read reverse)

NOTES

1. Bytes 1 and 2 must contain 0s to indicate no soft error.
2. Other (write) — write retry succeeded in same physical location that previous write retry failed.
3. Status (read and read reverse) — successful operation after retries.

Byte number 2

Parameter address	25
Name	— (Diagnostic mode status byte 2)
Microbus address	—
Register location	—

Bit	Name	Description
7:0	—	With TA79 in diagnostic mode, byte 2 is one of two status bytes for classifying soft errors. One of 8 codes, defined below, may appear in this byte. In normal operation, byte 2 is a 0 (see note below byte 4).

Partial Diagnostic Mode Soft Error Status

Code	Meaning
00	No soft error (see note 1)
01	Double track correction (read)
02	Double track correction (read reverse)
03	Single track correction (read)
04	Single track correction (read reverse)
08	Media
0D	Other (read). See note 2.
15	Other (read reverse). See note 2.

NOTES

1. Bytes 1 and 2 must contain 0s to indicate no soft error.
2. Other (read and read reverse) — successful operation without error correction, but with interesting occurrence (AMTIE or PHTIE flag).

Byte number 3

Parameter address	26
Name	— (Diagnostic mode status byte 3)
Microbus address	—
Register location	—

Bit	Name	Description
7:0	—	This byte is reserved for future use and is always a 0.

Byte number 4

Parameter address	27
Name	— (Diagnostic mode status byte 4)
Microbus address	—
Register location	—

Bit	Name	Description
7:0	—	This byte is reserved for future use and is always a 0.

NOTE
When a non-zero value appears in byte 1 or 2, the TA79 does not update bytes 5 and above in an extended drive status area. The TA79 then transmits only the first 4 bytes of the requested extended drive status area to the HSC.

Byte number 5

Parameter address	28
Name	OPSAV
Microbus address	—
Register location	—

Bit	Name	Description
7:0	—	Last STI level 2 command to begin execution before error detection. The table below identifies each available code and its meaning.

Command Code	Meaning
41	Clear drive errors
42	Clear formatter errors
44	Disconnect
47	Get formatter characteristics
48	Get summary status
4B	Initiate rewind
4D	Position tape
4E	Set unit execution mode
50	Set byte count
53	Write tape mark
55	Data error recovery
C0	Change controller flags
C3	Diagnose
C5	Get extended drive status
C6	Get formatter error log
C9	Get unit characteristics
CA	Erase gap
CC	Online
CF	Read memory
D1	Set unit characteristics
D2	Topology
D4	Write memory

Byte number 6

Parameter address	29
Name	ERRNUM (Error number low byte)
Microbus address	—
Register location	—

Bit	Name	Description
7:0	—	Eight least significant bits of 10-bit identification number. (Refer to Paragraph B.3.1)

Byte number 7

Parameter address	2A
Name	ERRN1 (Error number high byte)
Microbus address	—
Register location	—

Bit	Name	Description
7	Critical error	True if error is critical: transport should be removed from service. This bit is forced to 0 for error that occurs while operational code is running.
6	Data available	True if data generated by diagnostic code is available. This bit is forced to 0 for error that occurs while operational code is running.
Bit	**Name**	**Description**
5:3	Fault number	Indicates one of six areas in the formatter that may have caused a potentially fatal error.
2	—	When 1, indicates operational error. When 0, indicates diagnostic error.
1:0	—	Two most significant bits of 10-bit error identification number. (Refer to Paragraph B.3.1)

Byte number 8

Parameter address	2B
Name	RPFAIL (RMC write fail bits)
Microbus address	00-R
Register location(s)	M8950s (RC3) via M8953 (RP6)

Bit	Name	Description
7:0	Write fail 7:0	These are write fail bits from read channels 7—0. Write fail is an OR of certain error conditions, namely: Illegal 5-to-4 translation AMTIE PHTIE Pointer mismatch

Byte number 9

Parameter address	2C
Name	RPATH (Read path status)
Microbus address	01-R
Register location(s)	M8953 (RP6)

Bit	Name	Description
7	Velocity ok	True if velocity is within 10% of 125 in/sec or jam velocity ok bit is set (RPCTL, 09-W bit 2).
6	Status valid	Indicates M8953 status is valid (set after 0.1 inch gap found).
5	Preamble error	True if preamble had AMTIE.
4	Data ready	Goes false when M8953 asserts data ready to M8951.
3	Beginning of preamble	True when read path senses that the preamble has just passed the read head.
2	Clock stopped	True if read path clock is stopped.
1	Statistics select	True if TIE bus statistics are being sent to TIE bus.
0	Write fail P	Indicates illegal 5 to 4, AMTIE, PHTIE, or pointer mismatch for parity bit.

Byte number A

Parameter address	2D
Name	RSTAT (RMC status byte)
Microbus address	02-R
Register location(s)	M8950s via M8953 (RP4)

Bit	Name	Description
7:0	RMC status 7:0	This byte contains the read path microcontroller status codes. The following table identifies each code and its meaning.

Read Path Microcontroller Status Codes

Status	Meaning
Status from ECC self-test command	
41	ECC controller passed self-test.
42	ECC controller failed self-test.
Status from an M8953 self-test	
43	Read path passed self-test.
44	Read path failed self-test.
Status from an M8950 self-test command	
46	Read channel tests all passed.
Status from a clear all test command for velocity testing of drive by microcode	
1	First tach pulse
81	Last tach pulse (eleventh; ten spaces)
Status from a sample density command	
88	NOT CAPABLE found.
89	GCR ID found.
8A	PE ID found.
Status from a write test of IBG, PE ID, GCR ID, ARA ID, or ARA burst	
90	Bad status (write test).
Status from a tape mark test command	
92	Good tape mark found on tape status.

Read Path Microcontroller Status Codes

Status resulting from a non-BOT command (read or write FWD or REV, GCR or PE)

98	ARA ID found (not record or tape mark).
99	Tape mark found.
9C	Preamble end not found.
9D	Read path fault 1, too many M8950s have been fataled to continue record processing.
9E	Read path fault 2, 7 or more M8950 modules found illegal 5-to-4 translations.
A1	Unexpected IBG in data; probably creased tape (seven or more AMTIES active).
B1	Postamble long.
B2	Postamble short.
FF	OK.

Byte number B

Parameter address	2E
Name	RCMLP (RMC command loop back)
Microbus address	03-R
Register location(s)	M8953 (RP6)

Bit	Name	Description
7:0	Command 7:0	Contains the last command sent to the M8953 via RCMD (address 0B-W). The table below identifies each code and its meaning.

Command Code	Task
00	NOP
01	Interblock read
02	Test PE ID burst
03	Test GCR ID burst
04	Test ARA ID burst
05	Test tape mark
06	Test ARA burst
07	Normal non-BOT read
08	Run RMC self-test
09	Test unknown ID burst
0A	Run read channel micro's test
0B	Diagnostic read command
0C	Run read channel self-test
0D	Run clear all RMC test program
0E	Run ECC self-test program
0F	Find gap

Byte number C

Parameter address	2F
Name	RAMT (Read channel AMTIE status)
Microbus address	10-R
Register location(s)	Tape transport via M8955, M8950, and M8953

Bit	Name	Description
7:0	AMTIE 7:0	This byte reflects the state of the AMTIE lines for channels 7 through 0.

Byte number D

Parameter address	30
Name	RDON (Read channel done status)
Microbus address	11-R
Register location(s)	M8950s via M8953

Bit	Name	Description
7:0	Done 7:0	A given bit is false if the associated M8950 has completed its assigned task.

Byte number E

Parameter address	31
Name	RILL (Read channel illegal status)
Microbus address	12-R
Register location(s)	M8950 via M8953

Bit	Name	Description
7:0	Illegal 7:0	A given bit is true if the associated M8950 indicates the following. GCR Data contained an illegal 5-bit code which was either an error or a tape format control character. PE Data error occurred such as no bit time phase transition.

Byte number F

Parameter address	32
Name	RMK2 (Read channel mark 2 status)
Microbus address	13-R
Register location(s)	M8950s via M8953

Bit	Name	Description
7:0	7:0 Mark 2	When true, indicates that the associated M8950 has detected a Mark 2 tape format control character. If the corresponding illegal 5-to-4 bit is set (RILL 7:0), a Mark 2 was detected during 5-to-4 conversion.

Byte number 10

Parameter address	33
Name	REND (Read channel end status)
Microbus address	14-R
Register location(s)	M8950s via M8953

Bit	Name	Description
7:0	END 7:0	End Mark for read channels 0 — 7.

Byte number 11

Parameter address	34
Name	RPSTA (Read channel parity status)
Microbus address	15-R
Register location(s)	M8950 parity module

Bit	Name	Description
7	Corrected data P	Contains ECC module corrected output data for parity bit.
6	Data P	Contains M8950 data output for parity bit.
5	Post P	Indicates PE postamble for parity bit.
4	End P	Indicates End Mark for parity bit.
3	Mark 2 P	Indicates Mark 2 for parity bit.
2	Illegal P	Indicates illegal 5-to-4 for parity bit.
1	Not done P	Parity M8950 not done.
0	AMTIE P	Weak amplitude on parity bit.

Byte number 12

Parameter address	35
Name	RPOSTN (Read channel PE postamble detect)
Microbus address	16-R
Register location(s)	M8950s via M8953

Bit	Name	Description
7:0	Postamble 7:0	In PE, contains the inverted data for the byte prior to the one currently on the data lines (RC3 D(N) H). Used for PE POSTAMBLE detection. Not used in GCR.

Byte number 13

Parameter address	36
Name	RDATA (Read channel data)
Microbus address	17-R
Register location(s)	M8950s via M8953

Bit	Name	Description
7:0	Data 7:0	Contains data output from read channels 7—0 to ECC module.

Byte number 14

Parameter address	37
Name	CRCWRD (CRC byte)
Microbus address	18-R
Register location(s)	M8952 (CRC3)

Bit	Name	Description
7:0	CRC 7:0	These are CRC checker output bits.

Byte number 15

Parameter address	38
Name	ECCOR (Corrected data)
Microbus address	19-R
Register location(s)	M8951 (EC5)

Bit	Name	Description
7:0	Corrected data 7:0	Contains corrected data output from ECC module to CRC module.

Byte number 16

Parameter address	39
Name	ECCSTA (ECC status byte)
Microbus address	1A-R
Register location(s)	M8951 (EC5)

Bit	Name	Description
7	CRC error	Indicates CRC did not check.
6	ECC ROM parity error	Indicates M8951 program parity error.
5	AMTIE occurred	Indicates AMTIE occurred during data portion of record.
4	ACRC error	Indicates ACRC did not check.
3	Pointer mismatch	Indicates the track in error did not have a pointer.
2	Uncorrectable	Indicates ECC could not correct data error.
1	Two-track error	Indicates two-track error correction was performed on data.
0	Single-track error	Indicates single-track error correction was performed on data.

Byte number 17

Parameter address	3A
Name	— (Read channels 0 and 1 TIE)
Microbus address(es)	20-R and 21-R
Register location(s)	M8950s channels 0 and 1

Bit	Name	Description
7	CH1 TIE bus 3	Indicates illegal 5-to-4 in GCR, and dead track in PE. (Bad track register is ORed into this bit.)
6	CH1 TIE bus 2	AMTIE
5	CH1 TIE bus 1	PHTIE
4	CH1 TIE bus 0	If statistics = 0, indicates pointer mismatch in GCR and 0 in PE. If statistics = 1, indicates any pointer occurred (GCR and PE).

Bit	Name	Description
3	CH0 TIE bus 3	Indicates illegal 5-to-4 in GCR, and dead track in PE. (Bad track register is ORed into this bit.)
2	CH0 TIE bus 2	AMTIE
1	CH0 TIE bus 1	PHTIE
0	CH0 TIE bus 0	If statistics = 0, indicates pointer mismatch in GCR and 0 in PE. If statistics = 1, indicates any pointer occurred (GCR and PE).

Byte number 18

Parameter address	3B
Name	— (Read channels 2 and 3 TIE)
Microbus address(es)	22-R and 23-R
Register location(s)	M8950s channels 2 and 3

Bit	Name	Description
7	CH3 TIE bus 3	Indicates illegal 5-to-4 in GCR, and dead track in PE. (Bad track register is ORed into this bit.)
6	CH3 TIE bus 2	AMTIE
5	CH3 TIE bus 1	PHTIE
4	CH3 TIE bus 0	If statistics = 0, indicates pointer mismatch in GCR and 0 in PE. If statistics = 1, indicates any pointer occurred (GCR and PE).
3	CH2 TIE bus 3	Indicates illegal 5-to-4 in GCR, and dead track in PE. (Bad track register is ORed into this bit.)
2	CH2 TIE bus 2	AMTIE
1	CH2 TIE bus 1	PHTIE
0	CH2 TIE bus 0	If statistics = 0, indicates pointer mismatch in GCR and 0 in PE. If statistics = 1, indicates any pointer occurred (GCR and PE).

Byte number 19

Parameter address	3C
Name	— (Read channels 4 and 5 TIE)
Microbus address(es)	24-R and 25-R
Register location(s)	M8950s channels 4 and 5

Bit	Name	Description
7	CH5 TIE bus 3	Indicates illegal 5-to-4 in GCR, and dead track in PE. (Bad track register is ORed into this bit.)
6	CH5 TIE bus 2	AMTIE
5	CH5 TIE bus 1	PHTIE
4	CH5 TIE bus 0	If statistics = 0, indicates pointer mismatch in GCR and 0 in PE. If statistics = 1, indicates any pointer occurred (GCR and PE).
3	CH4 TIE bus 3	Indicates illegal 5-to-4 in GCR, and dead track in PE. (Bad track register is ORed into this bit.)
2	CH4 TIE bus 2	AMTIE
1	CH4 TIE bus 1	PHTIE
0	CH4 TIE bus 0	If statistics = 0, indicates pointer mismatch in GCR and 0 in PE. If statistics = 1, indicates any pointer occurred (GCR and PE).

Byte number 1A

Parameter address	3D
Name	— (Read channels 6 and 7 TIE)
Microbus address(es)	26-R and 27-R
Register location(s)	M8950s channels 6 and 7

Bit	Name	Description
7	CH7 TIE bus 3	Indicates illegal 5-to-4 in GCR, and dead track in PE. (Bad track register is ORed into this bit.)
6	CH7 TIE bus 2	AMTIE
5	CH7 TIE bus 1	PHTIE

Bit	Name	Description
4	CH7 TIE bus 0	If statistics = 0, indicates pointer mismatch in GCR and 0 in PE. If statistics = 1, indicates any pointer occurred (GCR and PE).
3	CH6 TIE bus 3	Indicates illegal 5-to-4 in GCR, and dead track in PE. (Bad track register is ORed into this bit.)
2	CH6 TIE bus 2	AMTIE
1	CH6 TIE bus 1	PHTIE
0	CH6 TIE bus 0	If statistics = 0, indicates pointer mismatch in GCR and 0 in PE. If statistics = 1, indicates any pointer occurred (GCR and PE).

Byte number 1B

Parameter address	3E
Name	— (Read channel P TIE and TIE bus)
Microbus address(es)	28-R and 30-R
Register location(s)	M8950 channel P and M8953

Bit	Name	Description
7:4	TIE bus	Reads TIE bus without enabling any read channel modules. If jam TIE bus (address 09-W bit 7) is set, then the value written into address 0A-W is read.
3	CHP TIE bus 3	Indicates illegal 5-to-4 in GCR, and dead track in PE. (Bad track register is ORed into this bit.)
2	CHP TIE bus 2	AMTIE
1	CHP TIE bus 1	PHTIE
0	CHP TIE bus 0	If statistics = 0, indicates pointer mismatch in GCR and 0 in PE. If statistics = 1, indicates any pointer occurred (GCR and PE).

Byte number 1C

Parameter address	3F
Name	TAMT (AMTIE byte)
Microbus address	44-R
Register location(s)	M8955 (TUP 2, 4)

Bit	Name	Description
7:0	AMTIE 7:0	This byte reflects information on TU bus lines AMTIE 7:0 L.

Byte number 1D

Parameter address	40
Name	PSTAT (Tape unit port status)
Microbus address	48-R
Register location(s)	M8955 (TUP 2, 4)

Bit	Name	Description
7	Status parity error	Indicates a parity error has occurred when reading a TU79 register over the WCS lines.
6	Write data	Reflects the state of the TU bus strobe line WDS.
5	Command parity error	Indicates the TU79 has detected a parity error on the WCS lines during a TS78 command or command/status address write.
4	Tape unit present	Indicates that a TU79 transport is physically cabled to this TU port.
3	Tachometer	Reflects the state of the TU bus line TACH.
2	WCS parity	Reflects the state of the TU bus line WCS P.
1	Read parity	Reflects the state of the TU bus line RD P.
0	AMTIE parity	Reflects the state of the TU bus line AMTIE P.

Byte number 1E

Parameter address	41
Name	PRDD (TU port read data)
Microbus address	4C-R
Register location(s)	M8955 (TUP 2, 4)

Bit	Name	Description
7:0	Read data 7:0	This byte reflects information on TU bus lines RD 7:0 L.

Byte number 1F

Parameter address	42
Name	PERI (STI bus errors)
Microbus address	81
Register location(s)	M8971

Bit	Name	Description
7	Level 1 protocol error	Indicates that after first frame of level 2 message, a received hamming code is not recognized as valid continue or end code.
6	CROM parity error	Indicates M8970 program parity error.
5	Data late	Indicates one of three occurrences. An attempt was made to retrieve data from the PM FIFO that was not available. An attempt was made to store data in the PM FIFO before the input was ready. Or in autoshift mode, new data loaded into the hold register before the PM FIFO could accept the previous data.
4	Control parity error	Indicates parity error or synchronization error detected on real-time controller-state line. A parity error applies to the status bits. A synchronization error applies to the pattern of eight 0s and a 1 before the status bits.
3	Data pulse error	Indicates missing pulse on write/command data line. That is, two positive pulses appear with no negative pulse in between, or two negative pulses appear with no positive pulse in between.

Bit	Name	Description
2	Control pulse error	Indicates missing pulse on real-time controller-state line. This error is analogous to a data pulse error.
1	Cable not present	Indicates that there is no cable present between the TS78 logic gate and the TS78 control panel.
0	—	Not used

Byte number 20

Parameter address	43
Name	WMCSTA (Write microcontroller status)
Microbus address	D0-R
Register location(s)	M8959 (WMC1)

Bit	Name	Description
7	XMC not done	If 0, translator microcontroller has finished writing postamble and waiting to be serviced. If 1, translator is busy.
6	WMC not done	If 0, write microcontroller has finished writing PE data or GCR data groups. (Used in diagnostic single-step mode; normally appears for 110 ns.) If 1, write microcontroller busy or idle.
5	Transfer	Shifts one nibble on WMC DR bus into translator.
4	ECC to WMC DR bus	Gates HI or LO nibble of ECC character to WMC DR bus.
3	CRC to WMC DR bus	Gates HI or LO nibble of CRC character to WMC DR bus.
2	ACRC to WMC DR bus	Gates HI or LO nibble of ACRC character to WMC DR bus.
1	Residual to WMC DR bus	Gates HI or LO nibble of residual character to WMC DR bus.
0	XMC WCLK	Reflects true state of translator write clock pulse (110 ns at normal speed).

Byte number 21

Parameter address	44
Name	TUSEL0 (Tape unit port 0/1 select)
Microbus address	D1-R
Register location(s)	M8955 (port 0/1 slot 08)

Bit	Name	Description
7	Single tape unit port	If 0, both TUP modules are installed in backplane slots 08 and 09. If 1, only one TUP module is installed in backplane slot 08.
6	Byte count terminal count	If 0, write microcontroller count terminal count (BCTC) is asserted. If 1, BCTC is unasserted.
5	TU port 0 read path enable	When true, indicates that the read path for tape unit port 0 is enabled.
4	TU port 1 read path enable	When true, indicates that the read path for tape unit port 1 is enabled.
3	TU port 0 write path enable	When true, indicates that the write path for tape unit port 0 is enabled.
2	TU port 1 write path enable	When true, indicates that the write path for tape unit port 1 is enabled.
1, 0	Tape unit select 1.0	A 2-bit binary field that reflects the tape unit port currently selected by the TUSEL register (E0-W bits 1:0).

Byte number 22

Parameter address	45
Name	TUSEL1 (Tape unit port 2/3 select)
Microbus address	D2-R
Register location(s)	M8955 (port 2/3, slot 09)

Bit	Name	Description
7, 6	—	Not used.
5	TU port 2 read path enable	When true, indicates that the read path for tape unit port 2 is enabled.
4	TU port 3 read path enable	When true, indicates that the read path for tape unit port 3 is enabled.
3	TU port 2 write path enable	When true, indicates that the write path for tape unit port 2 is enabled.
2	TU port 3 write path enable	When true, indicates that the write path for tape unit port 3 is enabled.
1, 0	Tape unit select 1:0	A 2-bit binary field that reflects the tape unit port currently selected by the TUSEL register (E0-W bits 1:0).

NOTE
This byte is valid only when two TU port modules are present in the system (TUSEL0, bit 7 = 0.)

Byte number 23

Parameter address	46
Name	WRTDAT (write data)
Microbus address	D3-R
Register location(s)	M8959 (WMC 1)

Bit	Name	Description
7:0	DR D7:0 H	Looks at write or read data on the intermediate DRD bus.

Byte number 24

Parameter address	47
Name	— (Byte counter lo)
Microbus address	D4-R (first read)
Register location(s)	M8959 (WMC 1)

Bit	Name	Description
7:0	Byte count 7:0	Low byte of the write micro/byte assembly byte counter word. The first read of location D4 yields this byte.

Byte number 25

Parameter address	48
Name	— (Byte counter hi)
Microbus address	D4-R (second read)
Register location(s)	M8959 (WMC 1)

Bit	Name	Description
7:0	Byte count 15:8	High byte of the write micro/byte assembly byte counter word. The second read of location D4 yields this byte.

Byte number 26

Parameter address	49
Name	— (Pad counter lo)
Microbus address	D5-R (first read)
Register location(s)	M8959 (WMC1)

Bit	Name	Description
7:0	Pad count 7:0	Low byte of the write micro/byte assembly pad character counter word. The first read of location D5 yields this byte.

Byte number 27

Parameter address	4A
Name	— (Pad counter hi)
Microbus address	D5-R (second read)
Register location(s)	M8959 (WMC1)

Bit	Name	Description
7:0	Pad count 15:8	High byte of the write micro/byte assembly pad character counter word. The second read of location D5 yields this byte.

Byte number 28

Parameter address	4B
Name	— (Error code counter lo)
Microbus address	D6-R (first read)
Register location(s)	M8959 (WMC1)

Bit	Name	Description
7:0	Error code 7:0	Low byte of write micro/byte assembly error code count. The first read of location D6 yields this byte. (See byte 29 for Ecode counter values.)

Byte number 29

Parameter address	4C
Name	— (Error code counter hi)
Microbus address	D6-R (second read)
Register location(s)	M8959 (WMC1)

Bit	Name	Description
7:0	Error code 15:8	High byte of write micro/byte assembly error code count. The second read of location D6 yields this byte. Ecode counter is normally set to zero at the start of an operation. Ecode counter values when read at the completion of an operation:

Bit Value	Error Indication
0	Operation completed successfully.
—1	Read skip count >4 in 0 core dump, >3 in 10 compatible, or >8 in 10 high-density compatible.
—2	Read skip count >1 in 11 normal or 15 normal.
—3	Format code >6.
—4	WMC self-test diagnostic error.
—5	Read overrun or write fault.

Byte number 2A

Parameter address	4D
Name	WMCERR (Write microcontroller errors)
Microbus address	DA-R
Register location(s)	M8959 (WMC2)

Bit	Name	Description
7	—	Not used
6	DR MBD parity error	Bad (even) parity was received in data at TS78 input port.
5	Error	Write microcontroller diagnostic overall error status bit.
Bit	**Name**	**Description**
4	WMC ROM parity error	A write microcontroller ROM parity error was detected or the WMC internal micro-diagnostic stopped at an error halt.
3	DR read parity error	A parity error was detected on a read data byte sent from the CRC/ACRC to WMC module over the CRC bus.
2:0	—	Not used

Byte number 2B

Parameter address	4E
Name	INTSTA (Interrupt status)
Microbus address	E0-R
Register location(s)	M8972 (8MC1)

Bit	Name	Description
7	Interrupt timer (MCLK)	When clear, indicates the M8970 interrupt timer has timed out. First interrupt is 600 us after timer is started and subsequent interrupts are at 1.2 ms intervals until timer is reset.
6	USART receiver ready (RX)	When clear, indicates ASCII port USART has received a byte which is available in the receiver buffer.
5	Power ok	When set, indicates that the TS78 power supply has not signaled an AC LO condition.
4	Write data register parity	Reflects the state of the DR BYTE PAR H line. Serves as the parity bit for the WRTDAT register (D3-R), or as a PE write operation done flag.
3	KINI	Reflects the state of the KINI bit on the real-time controller-state line. When set, indicates controller request for STI bus initialization.
2	USART transmitter ready (TX)	When clear, indicates the ASCII port USART is ready to accept a new character in the transmitter buffer.
1	PE write parity error	When set, indicates that a vertical parity error has been detected at the output of the translator during a phase-encoded write operation.
Bit	**Name**	**Description**
0	Translator ROM parity error	When set, indicates that the translator microcontroller ROM has experienced a parity error or has branched to an error halt during its internal diagnostic routines.

Byte number 2C

Parameter address	4F
Name	TADR00 (TU79 status)
Microbus address	40-R (MIA register 0)
Register location(s)	M8955/MIA (TU79)

Bit	Name	Description
7	Ready	The tape unit is on-line, not rewinding, not loading or unloading, and all pneumatic interlocks are made.
6	Ready on	Ready has undergone a transition to the set state. Cleared by a CLR TU command.
5	On-line	The tape unit has been placed on-line; if ready, it will respond to motion commands issued remotely from the formatter.
4	Rewinding	The tape unit is currently rewinding.
3	PES	The tape unit is currently set to phase-encoded recording format. If reset, the tape unit is set to group-coded recording format.
2	BOT	The BOT marker on the tape is positioned at the BOT sensor and ready is set.
1	EOT	The EOT marker on the tape is or was positioned at the EOT sensor during a forward operation with ready set. EOT status clears when: • The EOT marker is positioned at the EOT sensor during a reverse operation with ready set • The tape unit is commanded to rewind tape • A CLR EOT command is written into the TU CMD B location.
0	File protect	The tape reel currently loaded does not have the write enable ring in place.

Byte number 2D

Parameter address	50
Name	TADR01 (MIA status A)
Microbus address	40-R (MIA register 1)
Register location(s)	M8955/MIA (TU79)

Bit	Name	Description
7	Manual test	The manual test switch is in the MAN TEST position.
6	Forward	The forward command bit has been written with a 1, or a DSE command has been written into the TU CMD B location.
5	Reverse	Indicates the state of the reverse command bit.
4	Write	Indicates the state of the write command bit.
3	Write inhibit	Indicates the state of the write inhibit command bit.
2	LWR	Indicates the state of the loop write-to-read command bit.
1	MOT	Indicates that the capstan servo motor is turning.
0	DSE	Indicates that a DSE command has been written into the TU CMD B location. Clears when EOT sets, or when a CLR TU command is written into TU CMD B.

Byte number 2E

Parameter address	51
Name	TADR02 (MIA status B)
Microbus address	40-R (MIA register 2)
Register location(s)	M8955/MIA (TU79)

Bit	Name	Description
7	CMD PE	A command or CMD/STA address has been received by the tape unit with even parity, or a TEST command has been written into TU CMD B. Clears when a CLR TU command is received, whether parity is correct (odd) or not. This bit drives the TU bus CMD PE L line.
6	PEC	A SET PE command has been sent to TU CMD B. Assuming proper hardware operation, this bit is the same as PES in TU STATUS.
5	ARA error	During a GCR operation, from 24.1 cm (9.5 in) after BOT, the read amplifier gains failed to achieve the required value. Valid only if ready is set.
4,3	Speed 1:0	This 2-bit field indicates the speed of a tape unit; TU79 = 2 (125 in/sec).
2,1,0	Mode select 2,1,0	Indicates the position of the multiposition mode select switch on the tape unit operator panel. The bits are meaningful only when the mode select bits in the threshold command location have been written to ones.

Bit Position

2	1	0	Meaning
1	1	0	Position 0 (Normal)
1	0	1	Position 1 (Maintenance)
0	1	1	Position 2 (Maintenance)
1	1	1	Position 3 (Maintenance)

Byte number 2F

Parameter address	52
Name	TADR03 (Serial number A)
Microbus address	40-R (MIA register 3)
Register location(s)	M8955/MIA (TU79)

Bit	Name	Description
7:4	SN TH 3:0	Most significant (thousands) BCD digit of the tape unit's serial number
3:0	SN HU 3:0	The hundreds BCD digit of the tape unit's serial number

Byte number 30

Parameter address	53
Name	TADR04 (Serial number B)
Microbus address	40-R (MIA register 4)
Register location(s)	M8955/MIA (TU79)

Bit	Name	Description
7:4	SN tens 3:0	The tens BCD digit of the tape unit's serial number.
3:0	SN ones 3:0	Least significant (ones) BCD digit of the tape unit's serial number.

Byte number 31

Parameter address	54
Name	TADR05 (Tape unit diagnostics)
Microbus address	40-R (MIA register 5)
Register location(s)	M8955/MIA (TU79)

Bit	Name	Description
7	Write bit 4	Indicates the state (polarity) of the write driver for physical track 4 head.
6	Write	There is no current flowing through the write or erase heads.
5	Read enable	Indicates the state of the read enable bit in the THRESHOLD command location.
4	EOT DET	The tape's EOT tab is positioned at the EOT sensor. Not valid unless ready = 1.
3	TACH	The state of the output from the capstan motor digital tachometer.

Bit	Name	Description
2	—	Zero
1,0	AMTIE THR 1:0	Indicates the value of the AMTIE threshold field in the THRESHOLD command location.

Byte number —

Parameter address 55
Name — (Spare location)
Microbus address —
Register location(s) —

Bit	Name	Description
7:0	—	This location is valid only for temporary data storage when using the ASCII port. Contents of this location are not returned to the HSC for error logging.

Byte number —
Parameter address 56
Name — (Spare location)
Microbus address —
Register location(s) —

Bit	Name	Description
7:0	—	This location is valid only for temporary data storage when using the ASCII port. Contents of this location are not returned to the HSC for error logging.

B.2.2 Extended Formatter Status

Byte number 1

Parameter address 57
Name ERRNUM (Error number low byte)

Bit	Description
7:0	Eight least significant bits of 10-bit error identification number.

Byte Number 2

Parameter address	58
Name	ERRN1 (Error number high byte)

Bit	Description
7	Critical error. When 1, indicates error is critical; formatter should be removed from service. This bit is forced to 0 for error that occurs while operational code is running.
6	Data available. When 1. indicates data generated by diagnostic code is available. This bit is forced to 0 for error that occurs while operational code is running.
5:3	Fault number. Indicates one of six areas in the formatter that may have caused a potentially fatal error.
2	When 1, indicates operational error. When 0, indicates diagnostic error.
1:0	Two most significant bits of 10-bit error identification number.

Byte number 3

Parameter address	59
Name	OPSAV

Bit	Description
7:0	Last received level 2 opcode. Listed below are the available command codes and their meaning.

Command Code	Meaning
41	Clear drive errors
42	Clear formatter errors
44	Disconnect
47	Get formatter characteristics
48	Get summary status
4B	Initiate rewind
4D	Position tape
4E	Set unit execution mode

Command Code	Meaning
50	Set byte count
53	Write tape mark
55	Data error recovery
C0	Change controller flags
C3	Diagnose
C5	Get extended drive status
C6	Get formatter error log
C9	Get unit characteristics
CA	Erase gap
CC	Online
CF	Read memory
D1	Set unit characteristics
D2	Topology
D4	Write memory

Byte number 4

Parameter address	5A
Name	CONNST (Connection state byte)

Bit	Description
7:5	Not used
4	Broken. When 1, indicates minimum integrity failure. Formatter cannot communicate via STI bus because of internal hardware problem.
3	Off-line. When 1, indicates both formatter ports are off-line with respect to controller. (No STI communication since PORT SELECT A and B are in disabled position.)
2	Topology. When 1, indicates formatter entered topology mode. With one formatter port on-line, second port simulates available state for up to 1 second.
1	Available. When 1, indicates one or both formatter ports are in available state relative to controller. (PORT SELECT A and/or B in enabled position but controller has not placed a port on-line.) Controller can access available port but access is limited to very few commands such as retrieving status.
0	On-line. When 1, indicates a formatter port is on-line with respect to controller (ready to receive any command or data). If PORT SELECT A and B are in enabled position, other port is unavailable except if formatter is in topology mode.

Byte number 5

Parameter address	5B
Name	SUMMOD (Summary mode byte 1)

Bit	Description
7	Formatter attention. When 1, indicates potentially significant status change in formatter; not caused by reception, validation or execution of controller command.
6	Transport 3 attention. Analogous to bit 7.
5	Transport 2 attention. Analogous to bit 7.
4	Transport 1 attention. Analogous to bit 7.
3	Transport 0 attention. Analogous to bit 7.
2	When 1, indicates formatter entered topology mode. (Refer to CONNST bit 2 for further definition.)
1	When 1, indicates PORT SELECT A and/or B is in enable (in) position
0	When 1, indicates diagnostic request. Formatter requests controller to execute diagnostic identified in formatter memory region FFFD.

Byte number 6

Parameter address	5C
Name	SUMERR (Summary error byte)

Bit	Description
7	Formatter error. When 1, indicates formatter error detected while running operational code.
6	Transmission error. When 1, indicates detection of STI bus error that falls into one of the following categories.

- Received hamming code of suspected level 1 message is not recognized, or received unit number is invalid
- Received level 1 command is in improper context (for example, formatter is not on-line for write command).
- Level 1 protocol error; received hamming code is not recognized as continue or end, after first frame of level 2 message.
- Checksum error; calculated checksum does not agree with received checksum.
- Level 0 error; pulse or parity error on real-time controller-state line, or pulse error on write/command data line.

Bit	Description
5	Level 2 protocol error. When 1, indicates detection of STI bus programming error relating to received message byte(s) in level 2 command. For example, operand has wrong parity or wrong number of message bytes received for specified command).
4	Diagnostic failed. When 1, indicates formatter error detected while running in-line, off-line or subsystem diagnostic. Diagnostic error report is available in formatter memory region 3432.
3:0	Not used.

Byte number 7

Parameter address	5D
Name	SUMMD2 (Summary mode byte 2)

Bit	Description
7:4	Not used
3	When 1, indicates new error log information is available in the extended formatter status area
2:0	Retry bits RP, RT and FD. These bits indicate formatter progress in an error recovery sequence instructing the controller what recovery step to do next. Retry codes are listed below.

Bit Position 2	1	0	
RP	RT	FD	Meaning
0	0	X	No error
0	1	0	Ready for data transfer in same direction
0	1	1	Ready for data transfer in opposite direction
1	0	0	Retried transfer succeeded
1	0	1	Retried transfer failed
1	1	0	Ready to position for retry in same direction
1	1	1	Ready to position for retry in opposite direction

Byte number 8

Parameter address	5E
Name	CONBYT (Controller flag byte)

Bit	Description
7	When 0, indicates normal operation. When 1, indicates formatter only responds to commands from controller diagnostic. The controller uses this mode to execute diagnostics that cannot keep the formatter on-line.
6:0	Not used.

Byte number 9

Parameter address	5F
Name	PRTOSV (Port control byte)

Bit	Description
7:4	Not used.
3	When 1, indicates formatter transmission enabled on port A real-time formatter-state line.
2	When 1, indicates formatter transmission enabled on port B real-time formatter-state line.
1	When 1, indicates formatter reception and transmission enabled on port A data lines, and formatter reception enabled on port A real-time controller-state line.
0	When 1, indicates formatter reception and transmission enabled on port B data lines, and formatter reception enabled on port B real-time controller-state line.

Byte number A

Parameter address	60
Name	FSTOSV (Formatter state byte)

Bit	Description
7	When 0, forces transmission of 0s on enabled port(s) real-time formatter-state line. When 1, allows transmission of state bits on enabled port(s) real-time formatter-state line. Bits 6—1 in FSTOSV are the state bits. Bit 7 is a 1 when formatter is on-line or available, and a 0 when formatter is off-line. (Refer to CONNST for definition of formatter connection states.)
6	Formatter receiver ready. When 1, indicates formatter is ready to receive a new message frame or data. Reception of a valid sync character causes the formatter to clear this bit to 0.
5	Attention. When 1, indicates potentially significant status change not caused by reception, validation or execution of controller command. In most cases, a status change is caused by user intervention such as opening a transport door or pressing the LOAD/REW button. Source of the attention condition is identified in SUMMOD. (See note below.)
4	Acknowledge. When 0, indicates data transfer error occurred (operation aborted) or exception occurred (tape mark, EOT or BOT detected during data transfer). See note below.
3	Data ready. In read operation, leading edge (transition to 1) indicates tape drive is starting sequence to transmit sync character and then data. Falling edge (transition to 0) indicates formatter has transmitted all of the data and part of the byte count. In write operation, leading edge (transition to 1) indicates tape drive is starting write sequence in autoshift mode. Falling edge (transition to 0) indicates data received and stored with no error, and EDC received and verified. In position tape operation, toggling of data ready bit indicates progress is being made. Formatter complements data ready for every gap detected (every record or tape mark passed). Acceptable progress is a toggling rate of at least twice for every long time-out period. Data ready is a 0 at beginning and end of operation. (See note below.)

Bit	Description
2	Not used. (See note below.)

NOTE
Definitions above for bits 5—2 are valid only when formatter is on-line. With formatter in available state, bits 5—2 represent a 4-bit event counter with bit 5 being the most significant bit (MSB). The counter records the number of changes in connection state of the transport(s). (The three transport connection states are on-line, available, and off-line.)

1	Available. When 0, indicates formatter is on-line. When 1, indicates formatter is available or off-line. (Refer to CONNST for definition of formatter connection states.)
0	Parity. When 0, allows normal operation: calculate odd parity based on bits 6-1 in FSTOSV to yield bit 0 (parity bit) on enabled real-time formatter-state line(s). When 1, forces parity error: calculate even parity yielding a parity bit in bit 0 that is inverted with respect to its correct value.

Byte number B

Parameter address	61
Name	PSWISV (Port switch byte)

Bit	Description
7	Not used.
6	When 1, indicates FAULT is being pressed.
5	When 1, indicates PORT SELECT A is in enabled (in) position.
4	When 1, indicates PORT SELECT B is in enabled (in) position.
3:0	Not used.

Byte number C

Parameter address	62
Name	LASSTA

Bit	Description
7:0	Last checkpoint passed in operational code. (Refer to Paragraph B.3.2)

Byte number D

Parameter address	63
Name	ERRFLG (Error retry flag byte)

Bit	Description
7:6	Not used.
5	When 1, indicates initial command was a write.
4	When 1, indicates initial command moved tape in reverse direction.
3	When 1, indicates initial command moved tape in opposite direction.
2	Most-significant-bit (MSB) of unit number for transport that executed initial command.
1	Least-significant-bit (LSB) of unit number for transport that executed initial command.
0	Not used.

Byte number E

Parameter address	64
Name	RETCNT (Retry counter)

Bit	Description
7:0	Number of retry requests during error recovery sequence.

Byte number F

Parameter address	65
Name	KICNT (STI bus initialization counter)

Bit	Description
7:0	Number of initializations since TA79 master reset or power-up.

Byte number 10

Parameter address	66
Name	—

Bit	Description
7:0	Contents of memory location specified in stack pointer (SP) after underflow error.

Byte number 11

Parameter address	67
Name	—

Bit	Description
7:0	Contents of next location (SP+1) after underflow error.

Byte number 12

Parameter address	68
Name	—

Bit	Description
7:0	Contents of next location (SP+2) after underflow error.

Byte number 13

Parameter address	69
Name	—

Bit	Description
7:0	Contents of next location (SP+3) after underflow error.

NOTE
Bytes 10-13 are only valid if an underflow error has occurred. Otherwise, these four bytes are cleared to 0.

Byte number —

Parameter address	6A
Name	— (Spare location)

Bit	Description
7:0	This location is valid only for temporary data storage when using the ASCII port. Contents of this location are not returned to the HSC for error logging.

PROTOCOL CONTROL REGISTER

Register name	PCTO
Microbus address	(200-W)
Register location(s)	M8970

Bit	Name	Description
7	Autoshift	When 1, places M8970 and M8971 in autoshift mode. This bit is set by the M8972 during write data reception to prevent the M8970 FIFO from overflowing. If the FIFO is full during autoshift mode, the M8971 inhibits the STI line clock, the HSC inhibits data transmission and a portion of the M8970 data path halts.
6	Formatter receiver ready	This bit specifies the state of the formatter receiver ready (FRR) state bit for the start of the next M8970 operation (for example, bit 6 = 1 yields FRR = 1). Once the operation is in progress, other conditions control FRR. Changing bit 6 then has no effect.
5	Transfer direction	This bit specifies the direction of information transfer through the M8970 data path. When 1, the M8970 transfers data or message bytes from the M8971 to the M8959. When 0, the M8970 transfers data or message bytes from the M8959 to the M8971.
4	Master flag 1	The M8972 sets this bit during an M8970 response transmission to indicate that the checksum has been loaded into the FIFO. When 1, this bit causes the M8970 to transmit the end hamming code after the checksum byte and then stop the operation.
3	Master flag 0	The M8972 uses this bit as a control flag to affect M8970 operations in progress. The meaning of the flag depends on the operation. The operations and corresponding meanings are listed below. Port switching — When 1, causes the M8970 to lengthen the positive port transition pulse to more than 18 clock cycles. The M8972 keeps this bit as a 0. Write data reception — Set when received data has been stored as a record on tape. Causes M8970 to end operation.

Bit	Name	Description
		Read data transmission — Set when all but 3 bits of the EDC character are shifted out. Causes M8970 to complete transmission operation. Command message reception — Cleared if error detected during level 1 validation. Causes M8970 to continue operation but without storing message bytes in FIFO. Response message transmission — Set when last message byte has been loaded. If message bytes are already shifted out, causes M8970 to shift out extra 0s waiting for the checksum. The M8972 then clears this bit when the checksum is loaded. Completion/Diagnostic echo — Cleared to send frame with diagnostic echo hamming code. Set to send frame with completion hamming code. This routine used only with this bit as a 0. STI test pattern transmission — Set causes M8970 to transmit 0116 test pattern. Clear causes M8970 to transmit AA16 test pattern.
2:0	Program counter 8:6	These are the three most significant address bits applied to the M8970 control ROM. These bits select 1 of 8 microprogram routines as follows.

Bit Value	Selected Routine
0	Not used
1	Port switching
2	Command message reception
3	Completion/diagnostic echo
4	Response message transmission
5	Read data transmission
6	Write data reception
7	STI test pattern transmission

PROTOCOL STATUS REGISTER

Register name	PSTI
Microbus address	(200-R)
Register location(s)	M8970

Bit	Name	Description
7	Composite error	When 1, indicates that the M8970 or M8971 has detected an error. PERI bits 7-2 identify sources for the composite error.
6	Done	When 1, indicates an M8970 operation is finished.
5	In progress	When 1, indicates an M8970 operation is in progress. For reception operations, this bit is set only after a sync character is received. For transmission operations, this bit is set only after shifting out a sync character. Once set, this bit is not cleared until the operation is finished.
4	Program counter 0	This bit is the least significant address bit applied to the M8970 program ROM. However, at the end of command message reception, this bit has special meaning. A 1 indicates that the M8970 FIFO contains a level 1 command or an error has occurred (for example, first received frame did not have a start hamming code). A 0 indicates the FIFO contains a level 2 message.
3	PCLK enabled	When 0, indicates all M8970 clocks are running. For transmission operations, a 1 for this bit indicates M8970 data path and ROM controller clocks are halted. For write data reception in autoshift mode, a 1 for this bit indicates the M8970 FIFO is full stopping the STI line clock.
2	Serial data output	Reflects the serial output of the M8970 which is applied to the M8971. This bit is forced to 1 during reception operations.
1	FIFO input ready	When 1, indicates M8970 FIFO input is ready to accept new data. A full FIFO forces this bit to 0.
0	FIFO output ready	When 1, indicates data is available for retrieval at the M8970 FIFO output. An empty FIFO forces this bit to 0.

PROTOCOL ERROR REGISTER

Register name	PERI
Microbus address	(201-R)
Register location(s)	M8971

Bit	Name	Description
7	Level 1 protocol error	Indicates that after first frame of level 2 message, a received hamming code is not recognized as valid continue or end code.
6	CROM parity error	Indicates M8970 program parity error.
5	Data late	Indicates one of three occurrences. An attempt was made to retrieve data from the PM FIFO that was not available. An attempt was made to store data in the PM FIFO before the input was ready. Or in autoshift mode, new data loaded into the hold register before the PM FIFO could accept the previous data.
4	Control parity	Indicates state bit parity error detected on real-time controller state line.
3	Data pulse error	Indicates missing pulse on write/command data line. That is, two positive pulses appear with no negative pulse in between, or two negative pulses appear with no positive pulse in between.
2	Control pulse error	Indicates missing pulse on real-time controller state line. This error is analogous to a data pulse error.
1	Cable not present	Indicates that there is no cable present between the TS78 logic gate and the TS78 control panel.
0	—	Not used

FORMATTER STATE REGISTER

Register name	FSTO
Microbus address	(202-W)
Register location(s)	M8971

Bit	Name	Description
7	Enable sending state	When 0, forces transmission of 0s on enabled STI real-time formatter state line(s). When 1, allows transmission of state bits on enabled STI real-time formatter state line(s). Bit 7 is a 1 when formatter is on-line or available, and a 0 when formatter is off-line.
6	—	Not used.
5	Attention/counter 3	Setting this bit to 1, sets formatter state bit 5 (designated the attention bit when the formatter is on-line). When on-line, the formatter sets bit 5 when a potentially significant status change occurs that is not caused by reception, validation or execution of a controller command. In most cases, a status change is caused by user intervention such as opening a transport door or pressing the LOAD/REW button. (See note below.)
4	Acknowledge/counter 2	Setting this bit to 1, sets formatter state bit 4 (designated the acknowledge bit when the formatter is on-line). When on-line, the formatter sets bit 4 when a data transfer error occurs (operation aborted) or exception occurs (tape mark, EOT or BOT detected during data transfer). (See note below.)
3	Data ready/counter 1	Setting this bit to 1, sets formatter state bit 3 (designated the data ready bit when the formatter is on-line). In read operation, leading edge (transition to 1) occurs when tape drive is starting sequence to transmit sync character and then data. Falling edge (transition to 0) occurs when formatter has transmitted all of the data and part of the byte count. In write operation, leading edge (transition to 1) occurs when tape drive is starting write sequence in autoshift mode. Falling edge (transition to 0) occurs when data received and stored with no error, and EDC received and verified.

Bit	Name	Description
		In position tape operation, toggling of data ready bit indicates progress is being made. Formatter complements data ready for every gap detected (that is, every record or tape mark passed). Acceptable progress is a toggling rate of at least twice for every long time-out period. Data ready is a 0 at beginning and end of operation. (See note below.)
2	Counter 0	Setting this bit to 1, sets formatter state bit 2. This bit is always a 0 when the formatter is on-line. (See note below.)

NOTE
With formatter in available state, bits 5-2 represent a 4-bit event counter with bit 5 being the most-significant-bit (MSB). The counter records the number of changes in connection state of the transport(s). (The three transport connection states are on-line, available and off-line.)

1	Available	Setting this bit to 1, sets the available state bit (bit 1). This bit is a 1 when the formatter is available or off-line, and a 0 when the formatter is on-line.
0	Parity	When 0, allows normal operation; calculate odd parity based on the six state bit locations to yield bit 0 (parity bit) on enabled STI real-time formatter state line(s). When 1, forces parity error; calculate even parity yielding a parity bit in bit 0 that is inverted with respect to its correct value.

CONTROLLER STATE REGISTER

Register name	CSTI
Microbus address	(202-R)
Register location(s)	M8971

Bit	Name	Description
7	Deserializer receiver ready	When 0, indicates that the last received controller state frame had a state-frame clock cycle more than 240 ns long.
6	Controller receiver ready	When 1, indicates controller is ready to receive a new message frame or data. Reception of a valid sync character causes the controller to clear this bit to 0.
5	Keep going	When 1, indicates controller is ready to issue the same command that is now in progress. The formatter ignores this bit unless a write data reception is in progress and is successful (that is, acknowledge = 1). A 1 for the keep going bit toward the end of a successful write data reception causes the TA78 to keep the tape moving for 2.4 ms after writing a record. This yields a 0.3 in interblock gap in anticipation of the next write command. If the correct command does not arrive within 2.4 ms, the TA78 stops the tape and records an error condition.
4	Clock slow	When 1, indicates the controller cannot process STI information with clock rates faster than 8.8 MHz. The TA78 ignores this bit.

Bit	Name	Description
3	Initialization	A leading edge (transition to 1) for this bit causes the formatter to perform an STI bus initialization which consists of the following sequence. 1. The formatter momentarily disables both STI ports to acknowledge receipt of the initialization signal. The formatter then enables the port A and B real-time formatter state lines and transmits the sync pattern with all formatter state bits equal to 0. However, all formatter transmission on a STI read/response data line remains halted. 2. The formatter increments the software initialization counter (KINICT) by 1. The formatter then cancels the command timer. 3. The TA78 halts any tape motion except if a DSE or rewind operation is in progress. These two operations are allowed to continue until completion. For a write operation, the TA78 halts tape motion immediately and sets the position lost (PL) bit in the drive error (DE) byte. For a read, read reverse or position operation, tape motion continues until a gap is detected. The TA78 then halts the tape and updates the gap count. 4. If the formatter is on-line, the M8972 resets all the formatter microcontroller modules. 5. The formatter updates the extended drive status area for any connected transport. 6. The formatter runs the minimum integrity microdiagnostic chain. 7. The formatter returns to the appropriate idle state reflecting the mode it was in before initialization. (The same STI ports are enabled and the same formatter state bits are transmitted.)
2:1	—	Not used.
0	Reload	When 1, indicates the M8971 serializer's parallel input buffer is ready to accept a new byte. Writing into the FSTO register clears this bit to 0.

TMBUS/MICROBUS REGISTER

Register name	TMB
Microbus address	(203-R/W)
Register location(s)	M8970

Bit	Name	Description
7:0	—	This register provides access to the M8970 FIFO, by means of the MBD lines between the M8970 and M8959. Any write to this register momentarily places the supplied microbus data onto the MBD lines and loads it into the FIFO. Any read from this register retrieves the data from the MBD lines which reflects the M8970 FIFO output. Reading this register also shifts out the FIFO data allowing the next data byte to appear at the FIFO output and the MBD lines.

NOTE
To load or retrieve from the M8970 FIFO, you must first select the direction of information transfer through the M8970, using PCTO (80-W) bit 5. For loading the FIFO, reset PCTO bit 5 to 0. For retrieving from the FIFO, set PCTO bit 5 to 1.

NOTE
For loading the FIFO, you must also disable the M8959 MBD drivers. To do this, set WMCCTL (D3-W) bit 3 to 1 and also set bits 7, 4, 3, 1 and 0 to 1 in the DDRCTL register (DA-W).

ENABLE PROTOCOL MICROCONTROLLER REGISTER

Register name	ENPMO
Microbus address	(204-W)
Register location(s)	M8970

Bit	Name	Description
7:0	—	Not used. Any write to this register activates the M8970 allowing the ROM microcontroller, data path and other logic sections to function. The M8970 then starts executing a specified microprogram routine for an operation. Writing into this register also causes the M8970 to sample PCTO (80-W) bit 6 to determine the initial state of the formatter receiver ready state bit.

NOTE
Before activating the M8970, write a command code into the PCTO register to specify the microprogram routine address and other initial conditions for the desired operation.

CLEAR PROTOCOL MICROCONTROLLER REGISTER

Register name	CLPMI
Microbus address	(204-R)
Register location(s)	M8970

Bit	Name	Description
7:0	—	Not used. Any read to this register resets the M8970. This inhibits any operation in progress, clears any detected error condition, and resets the formatter receiver ready state bit to 0. However, the master clock and therefore the STI line clock remain running.

PORT CONTROL REGISTER

Register name	PRTO
Microbus address	(205-W)
Register location(s)	M8970

Bit	Name	Description
7:4	—	Not used.
3	Send state port A	When 1, enables formatter transmission on the port A real-time formatter state line.
2	Send state port B	When 1, enables formatter transmission on the port B real-time formatter state line.
1	Enable port A	When 1, enables formatter reception and transmission on the port A data lines, and enables formatter reception on the port A real-time controller state line.
0	Enable port B	When 1, enables formatter reception and transmission on the port B data lines, and enables formatter reception on the port B real-time controller state line.

NOTE
This register only affects the formatter STI ports if the M8970 runs the port switch routine. Therefore, after loading the PRTO register, the M8972 supplies the port switch routine address and activates the M8970.

PORT SWITCH REGISTER

Register name	PSWI
Microbus address	(205-R)
Register location(s)	M8970

Bit	Name	Description
7	SCLK	Reflects the state of the write microcontroller SCLK line.
6	Fault	When 0, indicates FAULT is being pressed.
5	Port A	When 0, indicates PORT SELECT A is in the enabled (in) position.
4	Port B	When 0, indicates PORT SELECT B is in the enabled (in) position.
3	Send state port A	When 1, indicates formatter transmission is enabled on the port A real-time formatter state line.
2	Send state port B	When 1, indicates formatter transmission is enabled on the port A real-time formatter state line.
1	Enable port A	When 1, indicates formatter reception and transmission are enabled on the port A data lines, and formatter reception is enabled on the port A real-time controller state line.
0	Enable port B	When 1, indicates formatter reception and transmission are enabled on the port B data lines, and formatter reception is enabled on the port B real-time controller state line.

NOTE
Bits 3-0 in this register reflect the STI ports currently enabled. These bits should match bits 3-0 in the PRTO register, after the M8970 successfully runs the port switch routine.

PROTOCOL DIAGNOSTIC REGISTER

Register name	PDIO
Microbus address	(206-W)
Register location(s)	M8970

Bit	Name	Description
7	—	Not used.
6	Force pulse error	When 1, causes control and data pulse error conditions; pulse inputs are held low for both M8971 decoders. A properly working M8971 module should detect the errors setting PERI bits 3 and 2.
5	Force data late	When 1, causes a data late condition; M8970 FIFO-input clock is held low inhibiting FIFO data storage. A properly working M8970 module should detect the error setting PERI bit 5.
4	Select alternate pattern	When 1, allows M8970 to generate diagnostic patterns for transmission on enabled STI read/response data line.
3	Enable CRMO	When 1, disables M8970 control ROM and enables CRMO register to place its contents on the ROM output lines.
2	Force SI	Setting this bit to 1 causes the M8970 FIFO-input clock to go high strobing one byte into the FIFO. (A data late appears if the FIFO input is not ready or the FIFO is full.)

NOTE
Setting PDIO bit 5 overrides the effects of this bit (PDIO bit 2).

1	Single step	When 1, allows single step operation of the M8970 clock circuitry.
0	Clock step	When set and reset (toggled), this bit produces a single M8970 master clock pulse. (All other M8970 clocks and the STI line clock are based on the master clock.)

PROGRAM COUNTER REGISTER

Register name	PCI
Microbus address	(206-R)
Register location(s)	M8970

Bit	Name	Description
7:0	—	Eight least significant bits of the M8970 control ROM address.

CONTROL ROM OUTPUT REGISTER

Register name	CRMO
Microbus address	(207-W)
Register location(s)	M8970

Bit	Name	Description
7:0	—	If enabled by PDIO bit 3, these bits replace the microprogram instruction bits on the M8970 control ROM output lines.

CONTROL ROM INPUT REGISTER

Register name	CRMI
Microbus address	(207-R)
Register location(s)	M8970

Bit	Name	Description
7:0	—	This register contains the microprogram instruction on the M8970 control ROM output lines. (But if PDIO bit 3 is set, this register reflects the contents of the CRMO register.)

CLEAR INTERRUPT TIMER REGISTER

Register name	CLCLKO
Microbus address	(210-W)
Register location(s)	M8970

Bit	Name	Description
7:0	—	Not used. Any write to this register resets the interrupt timer on the M8970. The M8972 writes into this register when a desired time interval (for example, port switch time) has elapsed.

EDC LOW REGISTER

Register name	EDLI
Microbus address	(210-R)
Register location(s)	M8970

Bit	Name	Description
7:0	—	Low byte of error detection code (EDC) character generated by M8970.

NOTE
The EDC character is valid only after STI transfer of a record with an even number of data bytes or an odd number of data bytes plus an all 0s filler byte.

ENABLE INTERRUPT TIMER REGISTER

Register name	ENCLKO
Microbus address	(211-W)
Register location(s)	M8970

Bit	Name	Description
7:0	—	Not used. Any write to this register starts the interrupt timer on the M8970. The timer then generates an MCLK interrupt signal 600 us later followed by subsequent MCLK interrupts every 1.2 ms until the timer is reset.

NOTE
For the M8970 to continually generate new MCLK interrupts, the M8972 must write into the CLIFO register (8B-W) after receiving every interrupt.

EDC HIGH REGISTER

Register name	EDHI
Microbus address	(211-R)
Register location(s)	M8970

Bit	Name	Description
7:0	—	High byte of error detection code (EDC) character generated by M8970.

NOTE
The EDC character is valid only after STI transfer of a record with an even number of data bytes or an odd number of data bytes plus an all 0s filler byte.

LIGHT REGISTER

Register name	LITO
Microbus address	(212-W)
Register location(s)	M8971

Bit	Name	Description
7:3	—	Not used.
2	Fault	Setting this bit turns on the FAULT indicator.
1	Port select A	Setting this bit turns on the PORT SELECT A indicator.
0	Port select B	Setting this bit turns on the PORT SELECT B indicator.

IDENTIFICATION REGISTER

Register name	IDNI
Microbus address	(212-R)
Register location(s)	M8970

Bit	Name	Description
7:0	—	Binary base number set by 8 individual switches in the DIP switchpack on the TS78 backplane. The base number is the unit number for transport 0 (that is, the transport in the TA78 cabinet). The unit number for any transport connected to the TS78 is the base number plus the TU port number (0, 1, 2 or 3) that the transport is cabled to.

CLEAR INTERRUPT SIGNAL REGISTER

Register name	CLIFO
Microbus address	(213-W)
Register location(s)	M8970

Bit	Name	Description
7:0	—	Not used. Any write to this register causes the M8970 to negate the MCLK interrupt signal. The M8972 writes into this register after receiving every MCLK interrupt until the desired time interval has elapsed.

USART COMMAND REGISTER

Register name	USCMD
Microbus address	(301-W)
Register location(s)	M8972

Bit	Name	Description
7	EH	1 = Enter hunt mode, not valid for asynchronous mode
6	IR	1 = Internal reset (same action as hardware reset line)
5	RTS	1 = Assert RTS output (low)
4	ER	1 = Reset error bits
3	SBR	1 = Send break character (continuous low)
2	REN	1 = Receive enable
1	DTR	1 = Assert DTR output (low)
0	TEN	1 = Transmitter enabled

NOTE
Setting bit 6 (IR) to a 1 allows the next byte to be written into the USDAT register.

USART DATA REGISTER

Register name	USDAT
Microbus address	(300-R/W)
Register location(s)	M8972

Bit	Name	Description
	Write transmitter, read transmitter	USART data register (bidirectional)

USART MODE REGISTER

Register name USDAT
Microbus address (301-W)
Register location(s) M8972

Bit	Name	Description
7, 6	S2, S1	Sets the number of stop bits
5	EP	1 = even parity (if enabled)
4	PEN	1 = parity enabled
3, 2	L2, L1	Sets the character length
1, 0	B2, B1	Sets the clock division factor

USART STATUS REGISTER

Register name USSTAT
Microbus address (301-R)
Register location(s) M8972

Bit	Name	Description
7	DSR	Follows the DSR input
6	SYN	Sync detect. Not used for asynchronous mode
5	FE	Framing error
4	OE	Overrun error
3	PE	Parity error
2	TXE	Transmitter empty; 1 = transmitter empty, not outputting anything
1	RXR	Receiver ready: 1 = character present in the receive data buffer. Reflects the RXRDY output
0	TXR	Transmitter ready; 1 = transmitter data buffer empty

EXTERNAL MEMORY ADDRESS LOW BYTE

Register name EXADRL (External memory register)
Microbus address (310-W)
Register location(s) M8973

Bit	Name	Description
7:0	External memory address Low byte	Write only. Memory is accessed by writing 16 bit address into EXADRL and EXADRH.

EXTERNAL MEMORY DATA

Register name	EXDAT
Microbus address	(310-R)
Register location(s)	M8973

Bit	Name	Description
15:0	External memory data	Data is read from this register.

EXTERNAL MEMORY ADDRESS HIGH BYTE

Register name	EXADRH
Microbus address	(311-W)
Register location(s)	M8973

Bit	Name	Description
15:8	External memory address High byte	Write only. Memory is accessed by writing 16 bit address into EXADRL and EXADRH.

EXTERNAL HARDWARE REVISION LEVEL

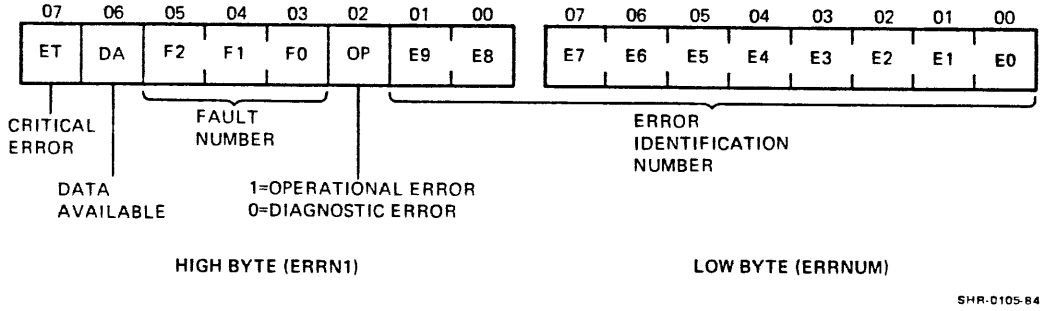
Register name	EXHRL
Microbus address	(311-R)
Register location(s)	M8973

Bit	Name	Description
7:0	Hardware revision level	The hardware revision level is read from this register.

B.2.3 Error Number

A 16-bit error number, shown in Figure B-1, reports both operational and diagnostic errors. Operational errors are errors that occur while the operational code is running. Diagnostic errors are errors that occur while an in-line diagnostic is running (in normal mode) or while an off-line or subsystem diagnostic is running. A hex number of C or 4 in the least-significant-digit (LSD) of the high byte (ERRN1) indicates an operational error which is identified in Table B-1. Any other LSD for ERRN1 indicates a diagnostic error which is identified in a standard error report including a summary description.

Figure B-1 Error Number Bytes



B.3 MASSBUS REGISTER DESCRIPTIONS

These register descriptions show the internal TM78 microbus hardware registers that can be accessed by the maintenance keypad/display (by means of the M8960 Microcomputer) or the host CPU (by means of Massbus registers R20 and R21).

The list is in I/O address numerical order. All addresses given are hexadecimal and are in the range of 0—FF (0—377). The octal equivalent of this address is shown in parentheses. They represent microcomputer addresses for data transfer to/from the 8085 accumulator. Except where noted, all locations may also be accessed by the host CPU. You can do this by setting the operand address in Massbus register R20 (R40) equal to 80nn, where nn = 0—FF (100nnn, where nnn = 0—377). Following the hex address are the letters R and/or W. This indicates whether the register is read only, write only, or read/write.

Table B-2 is a cross reference between the register's name and its internal address.

Table B-2 Register and Internal Address

Register Name	Address	Register Name	Address
BYTCNT	324-R/W	PDIAG	110-W
CH0TIE	40-R	PDIO	206-W
CH1TIE	41-R	PENAB	76-W
CH2TIE	42-R	PERI	201-R
CH3TIE	43-R	PRTO	205-W
CH4TIE	44-R	PRDD	114-R
CH5TIE	45-R	PSTAT	110-R
CH6TIE	46-R	PSTI	200-R
CH7TIE	47-R	PSWI	205-R
CHPTIE	50-R	RAMT	20-R
CLCLKO	210-W	RCMD	13-W
CLIFO	213-W	RCMLP	3-R
CLKCTL	360-W	RDATA	27-R
CLPMI	204-R	RDON	21-R
CNTCTL	327-W	REND	24-R
CRCWRD	30-R	RESCHR	321-W
CRMI	207-R	RFIFOL	10-W
CRMO	207-W	RILL	22-R
CSTI	202-R	RINST	14-W
DATACTL	320-W	RMK2	23-R
DDRA	330-R/W	RPATH	1-R
DDRB	331-R/W	RPCTL	11-W
DDRC	332-W	RPFAIL	0-R
DDRCTL	333-W	RPOSTN	26-R
ECCOR	31-R	RPTRIG	15-W
ECCSTA	32-R	RPSTA	25-R
EDHI	211-R	RSTAT	2-R
EDLI	210-R	RTIEB	12-W

Table B-2 Register and Internal Address (cont)

Register Name	Address	Register Name	Address
ENCLKO	211-W	RTIER	60-R
ENPMO	204-W	TAMT	104-R/W
ERRCNT	326-R/W	TCMP	100-W
EXADRH	311-W	TMB	203-R/W
EXADRL	310-W	TRKENA	322-W
EXDAT	310-R	TSTS	100-R
EXHRL	311-R	TUSEL0	321-R
FSTO	202-W	TUSEL1	322-R
IDNI	212-R	USCMD	301-W
INTSTA	340-R	USDAT	300-R/W
LITO	138-W	USMODE	301-W
PADCNT	325-R/W	USSTAT	301-R
PCI	206-R	WMCCTL	323-W
PCTO	200-W	WMCERR	332-R
		WMCSTA	320-R
		WRTDAT	323-R

NOTES:
R = read only register
W = write only register
R/W = read/write register

RMC WRITE FAIL BITS 7:0

Register name	RPFAIL
Microbus address	00-R (0-R)
Register location(s)	M8950s (RC3) by means of M8953 (RP6)

Bit	Name	Description
7:0	Write fail 7:0	These are write fail bits from read channels 7—0. Write fail is an OR of certain error conditions, namely: Illegal 5-to-4 translation, AMTIE, PHTIE, Pointer mismatch

READ PATH STATUS

Register name	RPATH
Microbus address	01-R (1-R)
Register location(s)	M8953 (RPG)

Bit	Name	Description
7	Velocity ok	True if velocity is within 10% of 125 in/sec or jam velocity ok bit is set (RPCTL, 011-W bit 2).
6	Status valid	Indicates M8953 status is valid (set after 0.1 inch gap found).
5	Preamble error	True if preamble had AMTIE.
4	Data ready	Goes false when M8953 asserts data ready to M8951.
3	Beginning of preamble	True when read path senses that the preamble has just passed the read head.
2	Clock stopped	True if read path clock is stopped.
1	Statistics select	True if TIE bus statistics are being sent to TIE bus.
0	Write fail P	Indicates illegal 5 to 4, AMTIE, PHTIE, or pointer mismatch for parity bit.

RMC STATUS WORD

Register name	RSTAT
Microbus address	02-R
Register location(s)	M8950s by means of M8953 (RP4)

Bit	Name	Description
7:0	RMC status 7:0	Read path microcontroller status codes return in this register. The following table identifies each code and its meaning.

Read Path Microcontroller Status Words

Status		Meaning
Status from ECC self-test command		
41	(100)	ECC controller passed self-test.
42	(102)	ECC controller failed self-test.
Status from an M8953 self-test		
43	(103)	Read path passed self-test.
44	(104)	Read path failed self-test.
Status from an M8950 self-test command		
46	(106)	Read channel tests all passed.
Status from a clear all test command for velocity testing of drive by microcode		
1	(1)	First tach pulse
81	(201)	Last tach pulse (eleventh)
Status from a sample density command		
88	(210)	NOT CAPABLE found.
89	(211)	GCR ID found.
8A	(212)	PE ID found.

Read Path Microcontroller Status Words

Status		Meaning
Status from a write test of IBG, PE ID, GCR ID, ARA ID, or ARA burst		
90	(220)	Bad status (write test).
Status from a tape mark test command		
92	(222)	Good tape mark found on tape status.
Status from a non-BOT command (read or write FWD or REV, GCR or PE)		
98	(230)	ARA ID found (not record or tape mark).
99	(231)	Tape mark found.
9C	(234)	Preamble end not found.
9D	(235)	Read path fault 1, too many M8950s have been fataled to continue record processing.
9E	(236)	Read path fault 2, 7 or more M8950 modules found illegal 5-to-4 translations.
A1	(241)	Unexpected IBG in data: probably creased tape (seven or more AMTIES active).
B1	(261)	Postamble long.
B2	(262)	Postamble short.
FF	(377)	OK

RMC COMMANDS LOOP BACK REGISTER

Register name	RCMLP
Microbus address	03-R (3-R)
Register location(s)	M8953 (RP6)

Bit	Name	Description
7:0	Command 7:0	Contains the last command sent to the M8953 by means of RCMD (address 0B-W).

TEST FIFO LOAD

Register name	RFIFOL
Microbus address	08-W(10-W)
Register locations	M8953 (RP5)

Bit	Name	Description
7:0	—	Not used. Any write to this register generates a test FIFO load pulse to the nine read channel modules. Used in PLO bypass modes 2 and 3 only.

READ PATH DIAGNOSTIC CONTROL REGISTER

Register name	RPCTL
Microbus address	09-W (11-W)
Register location(s)	M8953 (RP6)

Bit	Name	Description
7	Jam TIE bus	Used for diagnostics. Causes the TIE bus jam register to be used in place of the TIE data from the M8950 modules.
6	RC test data	Test data for read channel FIFOs.
5:4	PLO bypass	
	1:0	

Bit Value	FIFO Data	FIFO Clock
0	PLO	PLO
1	PLO	XMC W CLOCK
2	TU PORT	TEST FIFO LD
3	RC TEST	TEST FIFO LD DATA

3	PLO disable	Disables all M8950 PLOs.
2	Jam velocity ok	Forces velocity ok.
1	Statistics select	Causes statistic register in M8950s to be applied to TIE bus.
0	Stop clock	Stops the read path clock so that it can be single stepped. Also allows addresses 10—17 to be read.

RMC TIE BUS JAM REGISTER

Register name	RTIEB
Microbus address	0A-W (12-W)
Register location(s)	M8953

Bit	Name	Description
7:4	—	Not used.
3:0	TIE jam 3:0	These bits may be used instead of TIE bus data from M8950s. Jam TIE bus bit 7 must be set in RPCTL register (address 09-W).

READ MICROCONTROLLER COMMAND WORD

Register name	RCMD
Microbus address	0B-W (13-W)
Register location(s)	M8953 (RP6)

Bit	Name	Description
7	Run continuous	When set, allows RMC command codes 01-06 to run continuously.
6:0	RMC command 6:0	This register sets the read path task. Any write causes the M8953 program counter to be initialized, then interpret the task. The table below lists the tasks with the associated command code.

Command Code		Task
00	(00)	NOP
01	(01)	Interblock read
02	(02)	Test PE ID burst
03	(03)	Test GCR ID burst
04	(04)	Test ARA ID burst
05	(05)	Test tape mark
06	(06)	Test ARA burst
07	(07)	Normal non-BOT read
08	(10)	Run RMC self-test
09	(11)	Test unknown ID burst
0A	(12)	Run read channel micro's test
0B	(13)	Diagnostic read command
0C	(14)	Run read channel self-test
0D	(15)	Run clear all RMC test program
0E	(16)	Run ECC self-test program
0F	(17)	Find gap

RMC SINGLE-STEP CLOCK WORD

Register name	RINST
Microbus address	0C-W (14-W)
Register location	M8953 (RP5)

Bit	Name	Description
7:0	—	Not used.

Any write to this register causes the read microcontroller to advance one microinstruction.

Note that the read path normal clock must be stopped, and RPCTL (09-W) bit 0 = 1.

READ PATH TRIGGER

Register name	RPTRIG
Microbus address	0D-W (15-W)
Register location(s)	M8953 (RP5)

Bit	Name	Description
7:0	—	Not used. Any write to this register produces a low-going scope trigger pulse at pin C10K1.

READ CHANNEL AMTIE STATUS*

Register name	RAMT
Microbus address	10-R (20-R)
Register location(s)	Tape transport by means of M8955, M8950, and M8953

Bit	Name	Description
7:0	AMTIE 7:0	This register reflects the state of the AMTIE lines for channels 7 through 0.

* The read clock must be stopped before reading this register.

READ CHANNEL DONE STATUS*

Register name	RDON
Microbus address	11-R (21-R)
Register location(s)	M8950s by means of M8953

Bit	Name	Description
7:0	7:0	A given bit is false if the associated M8950 has completed its assigned task.

* The read clock must be stopped before reading this register.

READ CHANNEL ILLEGAL STATUS*

Register name	RILL
Microbus address	12-R (22-R)
Register location(s)	M8950 by means of M8953

Bit	Name	Description
7:0	7:0	A given bit is true if the associated M8950 indicates the following. GCR Data contained an illegal 5-bit code which was either an error or a tape format control character. PE Data error occurred such as no bit time phase transition.

* The read clock must be stopped before reading this register.

READ CHANNEL MARK 2 STATUS*

Register name	RMK2
Microbus address	13-R (23-R)
Register location(s)	M8950s by means of M8953

Bit	Name	Description
7:0	7:0 Mark 2	When true. indicates that the associated M8950 has detected a Mark 2 tape format control character. If the corresponding illegal 5-to-4 bit is set (RILL 7:0), a Mark 2 was detected during 5-to-4 conversion.

* The read clock must be stopped before reading this register.

READ CHANNEL END STATUS

Register name	REND
Microbus address	14-R (24-R)
Register location(s)	M8950s by means of M8953

Bit	Name	Description
7:0	END 7:0	End Mark for read channels 0—7.

READ CHANNEL PARITY STATUS*

Register name	RPSTA
Microbus address	15-R (25-R)
Register location(s)	M8950 parity module

Bit	Name	Description
7	Corrected data P	Contains ECC module corrected output data for parity bit.
6	Data P	Contains M8950 data output for parity bit.
5	Post P	Indicates PE postamble for parity bit.
4	End P	Indicates End Mark for parity bit.
3	Mark 2 P	Indicates Mark 2 for parity bit.
2	Illegal P	Indicates illegal 5-to-4 for parity bit.
1	Not done P	Parity M8950 not done.
0	AMTIE P	Weak amplitude on parity bit.

* The read clock must be stopped before reading this register.

READ CHANNEL PE POSTAMBLE DETECT REGISTER*

Register name	RPOSTN
Microbus address	16-R (26-R)
Register location(s)	M8950s by means of M8953

Bit	Name	Description
7:0	Postamble 7:0	In PE, contains the inverted data for the byte prior to the one currently on the data lines (RC3 D(N) H). Used for PE POSTAMBLE detection. Not used in GCR.

* The read clock must be stopped before reading this register.

READ CHANNEL DATA REGISTER*

Register name	RDATA
Microbus address	17-R (27-R)
Register location(s)	M8950s by means of M8953

Bit	Name	Description
7:0	Data 7:0	Contains data output from read channels 7—0 to ECC module.

* The read clock must be stopped before reading this register.

CRC WORD

Register name	CRCWRD
Microbus address	18-R (30-R)
Register location(s)	M8952 (CRC3)

Bit	Name	Description
7:0	CRC 7:0	These are CRC checker output bits.

CORRECTED DATA REGISTER

Register name	ECCOR
Microbus address	19-R (31-R)
Register location(s)	M8951 (EC5)

Bit	Name	Description
7:0	Corrected data 7:0	Contains corrected data output from ECC module to CRC module.

ECC STATUS WORD

Register name	ECCSTA
Microbus address	1A-R (32-R)
Register location(s)	M8951 (EC5)

Bit	Name	Description
7	CRC error	Indicates CRC did not check.
6	ECC ROM parity error	Indicates M8951 program parity error.
5	AMTIE occurred	Indicates AMTIE occurred during data portion of record.
4	ACRC error	Indicates ACRC did not check.
3	Pointer mismatch	Indicates the track in error did not have a pointer.
2	Uncorrectable	Indicates ECC could not correct data error.
1	Two-track error correction	Indicates two-track error correction was performed on data.
0	Single-track error correction	Indicates single-track error correction was performed on data.

READ CHANNEL 0 TIE BUS REGISTER

Register name	CH0TIE
Microbus address	20-R (40-R)
Register location(s)	M8950 channel 0

Bit	Name	Description
7:0	0	Zero
3	TIE bus 3	Indicates illegal 5-to-4 in GCR, and dead track in PE. (Bad track register is ORed into this bit.)
2	TIE bus 2	AMTIE
1	TIE bus 1	PHTIE
0	TIE bus 0	If statistics = 0, indicates pointer mismatch in GCR and 0 in PE. If statistics = 1, indicates any pointer occurred (GCR and PE).

READ CHANNEL 1 TIE BUS REGISTER

Register name	CH1TIE
Microbus address	21-R (41-R)
Register location(s)	M8950 channel 1

Bit	Name	Description
7:4	0	Zero
3	TIE bus 3	Indicates illegal 5-to-4 in GCR, and dead track in PE. (Bad track register is ORed into this bit.)
2	TIE bus 2	AMTIE
1	TIE bus 1	PHTIE
0	TIE bus 0	If statistics = 0, indicates pointer mismatch in GCR and 0 in PE. If statistics = 1, indicates any pointer occurred (GCR and PE).

READ CHANNEL 2 TIE BUS REGISTER

Register name	CH2TIE
Microbus address	22-R (42-R)
Register location(s)	M8950 channel 2

Bit	Name	Description
7:4	0	Zero
3	TIE bus 3	Indicates illegal 5-to-4 in GCR, and dead track in PE. (Bad track register is ORed into this bit.)
2	TIE bus 2	AMTIE
1	TIE bus 1	PHTIE
0	TIE bus 0	If statistics = 0, indicates pointer mismatch in GCR and 0 in PE. If statistics = 1, indicates any pointer occurred (GCR and PE).

READ CHANNEL 3 TIE BUS REGISTER

Register name	CH3TIE
Microbus address	23-R (43-R)
Register location(s)	M8950 channel 3

Bit	Name	Description
7:4	0	Zero
3	TIE bus 3	Indicates illegal 5-to-4 in GCR, and dead track in PE. (Bad track register is ORed into this bit.)
2	TIE bus 2	AMTIE
1	TIE bus 1	PHTIE
0	TIE bus 0	If statistics = 0, indicates pointer mismatch in GCR and 0 in PE. If statistics = 1, indicates any pointer occurred (GCR and PE).

READ CHANNEL 4 TIE BUS REGISTER

Register name	CH4TIE
Microbus address	24-R (44-R)
Register location(s)	M8950 channel 4

Bit	Name	Description
7:4	0	Zero
3	TIE bus 3	Indicates illegal 5-to-4 in GCR, and dead track in PE. (Bad track register is ORed into this bit.)
2	TIE bus 2	AMTIE
1	TIE bus 1	PHTIE
0	TIE bus 0	If statistics = 0, indicates pointer mismatch in GCR and 0 in PE. If statistics = 1, indicates any pointer occurred (GCR and PE).

READ CHANNEL 5 TIE BUS REGISTER

Register name	CH5TIE
Microbus address	25-R (45-R)
Register location(s)	M8950 channel 5

Bit	Name	Description
7:4	0	Zero
3	TIE bus 3	Indicates illegal 5-to-4 in GCR, and dead track in PE. (Bad track register is ORed into this bit.)
2	TIE bus 2	AMTIE
1	TIE bus 1	PHTIE
0	TIE bus 0	If statistics = 0, indicates pointer mismatch in GCR and 0 in PE. If statistics = 1, indicates any pointer occurred (GCR and PE).

READ CHANNEL 6 TIE BUS REGISTER

Register name	CH6TIE
Microbus address	26-R (46-R)
Register location(s)	M8950 channel 6

Bit	Name	Description
7:4	0	Zero
3	TIE bus 3	Indicates illegal 5-to-4 in GCR, and dead track in PE. (Bad track register is ORed into this bit.)
2	TIE bus 2	AMTIE
1	TIE bus 1	PHTIE
0	TIE bus 0	If statistics = 0, indicates pointer mismatch in GCR and 0 in PE. If statistics = 1, indicates any pointer occurred (GCR and PE).

READ CHANNEL 7 TIE BUS REGISTER

Register name	CH7TIE
Microbus address	27-R (47-R)
Register location(s)	M8950 channel 7

Bit	Name	Description
7:4	0	Zero
3	TIE bus 3	Indicates illegal 5-to-4 in GCR, and dead track in PE. (Bad track register is ORed into this bit.)
2	TIE bus 2	AMTIE
1	TIE bus 1	PHTIE
0	TIE bus 0	If statistics = 0, indicates pointer mismatch in GCR and 0 in PE. If statistics = 1, indicates any pointer occurred (GCR and PE).

READ CHANNEL P TIE BUS REGISTER

Register name	CHPTIE
Microbus address	28-R (50-R)
Register location(s)	M8950 channel P

Bit	Name	Description
7:4	0	Zero
3	TIE bus 3	Indicates illegal 5-to-4 in GCR, and dead track in PE. (Bad track register is ORed into this bit.)
2	TIE bus 2	AMTIE
1	TIE bus 1	PHTIE
0	TIE bus 0	If statistics = 0, indicates pointer mismatch in GCR, and 0 in PE. If statistics = 1, indicates any pointer occurred (GCR and PE).

READ CHANNEL TIE BUS REGISTER

Register name	RTIER
Microbus address	30-R (60-R)
Register location(s)	M8953

Bit	Name	Description
7:4	0	Zero
3:0	TIE bus	Reads TIE bus without enabling any read channel modules. If jam TIE bus (address 09-W bit 7) is set, then the value written into address 0A-W is read.

NOTE
Before performing I/O write/read operations to microbus addresses 40 — 4C (TU port/MIA registers), you must first select the desired TU port. This is done by writing the select address in MBSEL (E0-W) bits 1:0.

TU CMD/STAT ADDRESS TAPE UNIT COMMAND

Register name	TCMP
Microbus address	40-W (100-W)
Register location(s)	M8955/MIA (TU78)

Bit	Name	Description
—	—	Refer to Appendix C.

TAPE UNIT STATUS

Register name	TSTS
Microbus address	40-R (100-R)
Register location(s)	M8955/M1A (TU78)

Bit	Name	Description
—	—	Refer to Appendix C.

AMTIE LOOP

Register name	TAMT
Microbus address	44-W(104-W)
Register location(s)	M8955 (TUP 1, 3)

Bit	Name	Description
7:0	AMTIE loop data 7:0	Data sent to this register is latched and placed on TU BUS AMTIE 7:0 L.

NOTE
AMTIE P L must be asserted by means of PDIAG (110-W) bit 0.

NOTE
If PENAB (4C-W) bit 2 (RP EN) is asserted, data is placed on TUP P AMTIE 7:0 H.

AMTIE WORD

Register name	TAMT
Microbus address	44-R (104-R)
Register location(s)	M8955 (TUP 2, 4)

Bit	Name	Description
7:0	AMTIE 7:0	This register reflects information on TU bus line AMTIE 7:0 L.

TU PORT DIAGNOSTIC REGISTER

Register name	PDIAG
Microbus address	48-W(110-W)
Register location(s)	M8955 (TUP 2, 4)

Bit	Name	Description
7	—	Not used.
6	Loop command to status	Setting this bit inhibits TU bus lines TU CMD L and TU STAT L so that MIA registers are not strobed during a diagnostic write/read to the TU port CMD/STAT registers.
5	Loop write-to-read	Setting this bit allows information on the TU bus WCS lines (TU WCS 7:0, P) and bits 3, 1, 0 of this register to be placed on TU bus lines TU RD 7:0; RD P, AMTIE P, TACH.

NOTE
If PENAB (4C-W) bits 4 (WP EN) and 2 (RP EN) are set, loop write-to-read allows data from the write path (WDO 7:0, P) to be sent to the read path (P RD 7:0,P).

4	—	Not used.

NOTE
Bits 3, 1, and 0 may be set in the PDIAG register, but will not serve any function unless bit 5 (LWR) is set.

3	Send TACH	Setting this bit asserts the TU bus TACH line and simulates the output of the TU78 capstan servo tachometer.
2	—	Not used.
1	Send RD P	Setting this bit asserts the TU bus line RD P.
0	Send AMTIE P	Setting this bit asserts the TU bus line AMTIE P.

TAPE UNIT PORT STATUS WORD

Register name	PSTAT
Microbus address	48-R (110-R)
Register location(s)	M8955 (TUP 2, 4)

Bit	Name	Description
7	Status parity error	Indicates a parity error has occurred when reading a TU78 register over the WCS lines.
6	Write data strobe	Reflects the state of the TU bus line WDS.
5	Command parity error	Indicates the TU78 has detected a parity error on the WCS lines during a TS78 command or command/status address write.
4	Tape unit present	Indicates that a TU78 transport is physically cabled to this TU port.
3	Tachometer	Reflects the state of the TU bus line TACH.
2	WCS parity	Reflects the state of the TU bus line WCS P.
1	Read parity	Reflects the state of the TU bus line RD P.
0	AMTIE parity	Reflects the state of the TU bus line AMTIE P.

PORT ENABLE CONTROL REGISTER

Register name	PENAB
Microbus address	4C-W (114-W)
Register location(s)	M8955 (TUP 2, 4)

Bit	Name	Description
7	Interrupt enable	When set, allows the STAT PE/CMD PE bits (PSTAT 7, 5) to interrupt the microcomputer program. When reset, disallows the hardware interrupt but the status bits may be read by means of the PSTAT register (48-R bits 7, 5).
6, 5	—	Not used.
4	Write path enable	When set, enables write data to be placed on the WCS lines and disables the command/status functions. When reset, enables the WCS lines to be used for command/status information and inhibits write data.
3	—	Not used.
2	Read path	When set, allows TU bus signals enable RD 7:0, P; AMTIE 7:0, P; and TACH to be gated to the read path.
1, 0	—	Not used.

PORT READ DATA WORD

Register name	PRDD
Microbus address	4C-R (114-R)
Register location(s)	M8955 (TUP 2, 4)

Bit	Name	Description
7:0	Read data 7:0	This register reflects information on TU bus line RD 7:0 L.

NOTE
Before performing I/O write/read operations to microbus addresses 80 — C0 (Massbus port registers), you must first select the desired Massbus port. This is done by writing the select address in MBSEL (E0-W) bit 7.

BYTE ASSEMBLY LOGIC DATA CONTROL REGISTER

Register name	DATACTL
Microbus address	(320-W)
Register location(s)	M8959 (WMC1)

Bit	Name	Description
7	Pad/CRC	If 0, GCR CRC data group byte 1 will be CRC character. If 1, GCR CRC data group byte 1 will be pad character (all zeros).
6:4	Format 2:0	Bit Value — Format: 0 — 11 normal 1 — 15 normal 2 — 10 compatible 3 — 10 core dump 4 — 10 high-density compatible 5 — Image (Skip count is ignored.) 6 — 10 high-density dump 7 — Illegal (Ecode 3)

Bit	Name	Description		
3:0	Skip Count 3:0	Bit Value		Skip Count
		0	(00)	No skip
		1	(01)	Skip 1 byte
		2	(02)	Skip 2 bytes
		3	(03)	Skip 3 bytes
		4	(04)	Skip 4 bytes
		5	(05)	Skip 5 bytes
		6	(06)	Skip 6 bytes
		7	(07)	Skip 7 bytes
		8	(10)	Skip 8 bytes
		9-F	(11—17)	Illegal (Ecode 1)

WRITE MICROCONTROLLER STATUS

Register name	WMCSTA
Microbus address	(320-R)
Register location(s)	M8959 (WMC1)

Bit	Name	Description
7	XMC not done	If 0, translator microcontroller is done writing postamble and waiting to be serviced. If 1, translator is busy.
6	WMC not done	If 0, write microcontroller is done writing PE data or GCR data groups. (Used in diagnostic single-step mode; normally appears for 110 ns.) If 1, write microcontroller busy or idle.
5	Transfer	Shifts one nibble on WMC DR bus into translator.
4	ECC to WMC DR bus	Gates HI or LO nibble of ECC character to WMC DR bus.
3	CRC to WMC DR bus	Gates HI or LO nibble of CRC character to WMC DR bus.
2	ACRC to WMC DR bus	Gates HI or LO nibble of ACRC character to WMC DR bus.
1	Residual to WMC DR bus	Gates HI or LO nibble of residual character to WMC DR bus.
0	XMC WCLK	Reflects true state of translator write clock pulse (110 ns at normal speed).

RESIDUAL CHARACTER

Register name	RESCHR
Microbus address	(321-W)
Register location(s)	M8959 (WMC1)

Bit	Name	Description
7:5	Modulo-7 count	This binary count represents the number of data bytes to be written in the GCR residual data group. (Legal counts are 0-6.)
4:0	Modulo-32 count	This binary count represents the low-order five bits of the byte count, minus 1.

TAPE UNIT PORT 0/1 SELECT

Register name	TUSEL 0
Microbus address	(321-R)
Register location(s)	M8955 (port 0/1 slot 08)

Bit	Name	Description
7	Single tape unit port	If 0, both TUP modules are installed in backplane slots 08/09. If 1, only one TUP module is installed in backplane slot 08.
6	Byte count terminal count	If 0, write microcontroller count terminal count (BCTC) is asserted. If 1, BCTC is unasserted.
5	TU port 0 read path enable	When true, indicates that the read path for tape unit port 0 is enabled.
4	TU port 1 read path enable	When true, indicates that the read path for tape unit port 1 is enabled.
3	TU port 0 write path enable	When true, indicates that the write path for tape unit port 0 is enabled.
2	TU port 1 write path enable	When true, indicates that the write path for tape unit port 1 is enabled.
1, 0	Tape unit select 1:0	A 2-bit binary field that reflects the tape unit port currently selected by the MBSEL register (E0-W bits 1:0).

TRACK ENABLE

Register name	TRKENA
Microbus address	(322-W)
Register location(s)	M8958 (XMCH)

NOTE
This register enables or disables the nine write buffers independently and must be loaded in two sequential write operations.

Bit	Name	Description
7:1	Track enable 7:1	When true, TE 7:1 enables the write buffers for binary tape tracks 7:1.
0	Track enable P or Track enable 0	Every write to this register loads 7:0 and causes the previous contents of 0 to load into P.

TAPE UNIT PORT 2/3 SELECT

Register name	TUSEL 1
Microbus address	(322-R)
Register location(s)	M8955 (port 2/3, slot 09)

Bit	Name	Description
7, 6	—	Not used.
5	TU port 2 read path enable	When true, indicates that the read path for tape unit port 2 is enabled.
4	TU port 3 read path enable	When true, indicates that the read path for tape unit port 3 is enabled.
3	TU port 2 write path enable	When true, indicates that the write path for tape unit port 2 is enabled.
2	TU port 3 write path enable	When true, indicates that the write path for tape unit port 3 is enabled.
1, 0	Tape unit select 1:0	A 2-bit binary field that reflects the tape unit port currently selected by the MBSEL register (E0-W bits 1:0).

NOTE
This register exists only when two TU port modules are present in the system (that is, TUSEL0, bit 7 = 0).

WMC CONTROL

Register name	WMCCTL
Microbus address	(323-W)
Register location(s)	M8959 (WMC 1)

Bit	Name	Description
7	Enable write microcontroller	When true, causes the write microcontroller to examine the rest of the bits in this register and start transferring data to or from the Massbus.
6	Enable translator microcontroller	When true, causes the translator microcontroller to examine the rest of the bits in this register and start sending data to the TU port.
5	Ones	When true, informs the translator microcontroller that an ID burst or tape mark will be written.
4	GCR/PE	When true, informs the write translator and read path microcontrollers that a GCR mode write or read operation is in progress. When false, a PE mode write or read operation is in progress.
3	Write	When true, informs the write and read path microcontrollers that a write operation is in progress. When false, a read operation is in progress.
2	Reverse	When true, informs the write and read path microcontrollers that a read reverse operation is in progress. When false, (and bit 3 = 0) a read forward operation is in progress.
1	Diagnose	When true, informs the write microcontroller that the microcomputer wants to test the WMC byte/pad/Ecode counters.
0	Restart	When true, resets the write and translator microcontroller program counters to zero and clears any parity error conditions in those microcontrollers.

WRITE DATA

Register name	WRTDAT
Microbus address	(323-R)
Register location(s)	M8959 (WMC 1)

Bit	Name	Description
7:0	DR D7:0 H	Looks at write or read data on the intermediate DRD bus.

BYTE COUNTER HI/LO*

Register name	BYTCNT
Microbus address	(324-R/W)
Register location(s)	M8959 (WMC 1)

Bit	Name	Description
7:0	Byte count 7:0, 15:8	Write micro/byte assembly byte counter word. First write sets 7:0. Second write sets 15:8. First read accesses 7:0. Second read accesses 15:8.

PAD COUNTER HI/LO*

Register name	PADCNT
Microbus address	(325-R/W)
Register location(s)	M8959 (WMC 1)

Bit	Name	Description
7:0	Pad count 7:0, 15:8	Write micro/byte assembly pad character counter word. First write sets 7:0. Second write sets 15:8. First read accesses 7:0. Second read accesses 15:8.

* A double read/write operation must be performed to this register with CNTCTL 5, 4 = 3. This register cannot be written to by means of Massbus register 21.

ERROR CODE (ECODE) COUNTER HI/LO*

Register name	ERRCNT
Microbus address	(326-R/W)
Register location(s)	M8959 (WMC 1)

Bit	Name	Description
7:0	Error code 7:0, 15:8	Write micro/byte assembly error code count. First write sets 7:0. Second write sets 15:8. First read accesses 7:0. Second read accesses 15:8. Normally set to zeros at the start of an operation. Ecode counter values when read at the completion of an operation:

Bit Value	Error Indication
(000)	Operation completed successfully.
(377)	Read skip count >4 in 0 core dump. >3 in 10 compatible, or >8 in 10 high-density compatible.
(376)	Read skip count >1 in 11 normal or 15 normal.
(375)	Format code >6.
(374)	WMC self-test diagnostic error.
(373)	Read overrun or write fault.

* A double read/write operation must be performed to this register with CNTCTL 5, 4 = 3. This register cannot be written to by means of Massbus register 21.

WMC BYTE, PAD, ECODE COUNTER CONTROL

Register name CNTCTL
Microbus address (327-W)
Register location(s) M8959 (WMC 1)

Bit	Name	Description
7, 6	Counter select 1, 0	Selects one of three counters to read from or write into.

Bit Value	Selected Counter
0	Byte counter HI/LO
1	Pad counter HI/LO
2	Error code counter HI/LO
3	Illegal

Bit	Name	Description
5, 4	Read/write 1, 0	Selects read/write sequence.

Bit Value	Sequence
0—2	Not used.
3	Read/write least significant byte first, then most significant byte.

Bit	Name	Description
3:1	Mode 2:0	Selects counting mode.

Bit Value	Mode Selected
0	Terminal count stops counter.
1—7	Not used.

Bit	Name	Description
0	Binary/BCD	If 0, counters operate in 16-bit binary mode. If 1, counters operate in 4-decade BCD mode. (Not used.)

APPENDIX C
TM78 TAPE UNIT COMMAND/STATUS ADDRESSES AND BIT DESCRIPTIONS

The TM78 formatter controls the tape unit by writing into the TU command locations and senses the tape unit status by reading the TU status locations. The formatter does this by writing and reading internal address 100, which results in an 8-bit parallel transfer across the TU bus WCS (write/command/status) lines. The tape unit logic contains a maximum of 16 command and 16 status locations. However all 32 locations are not necessarily used.

The address of the 16 command or status locations is determined by a special form of write command that has bit 7 = 1. All other commands have bit 7 = 0, using the other seven bits for command information. Once a command/status address is specified, it remains in effect until the formatter sends a new address to the tape unit. Figure C-1 shows the format of the three types of command/status bytes. Figure C-2 is a summary of the tape unit command/status locations. Tables C-1 (command) and C-2 (status) provide a detailed description of the command/status locations.

Figure C-1 Tape Unit Command/Status Byte Format

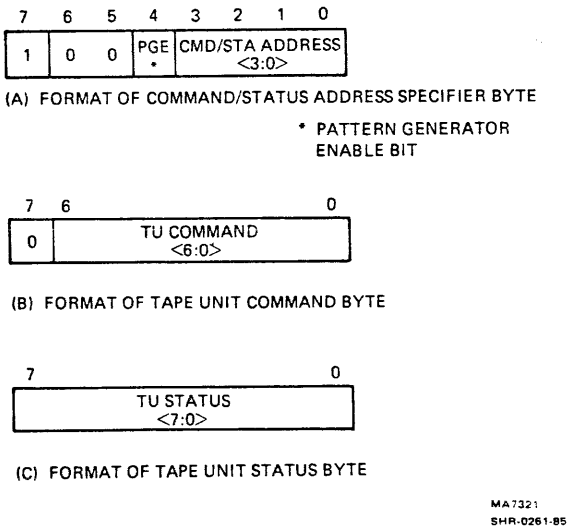


Figure C-2 TU Command/Status Summary

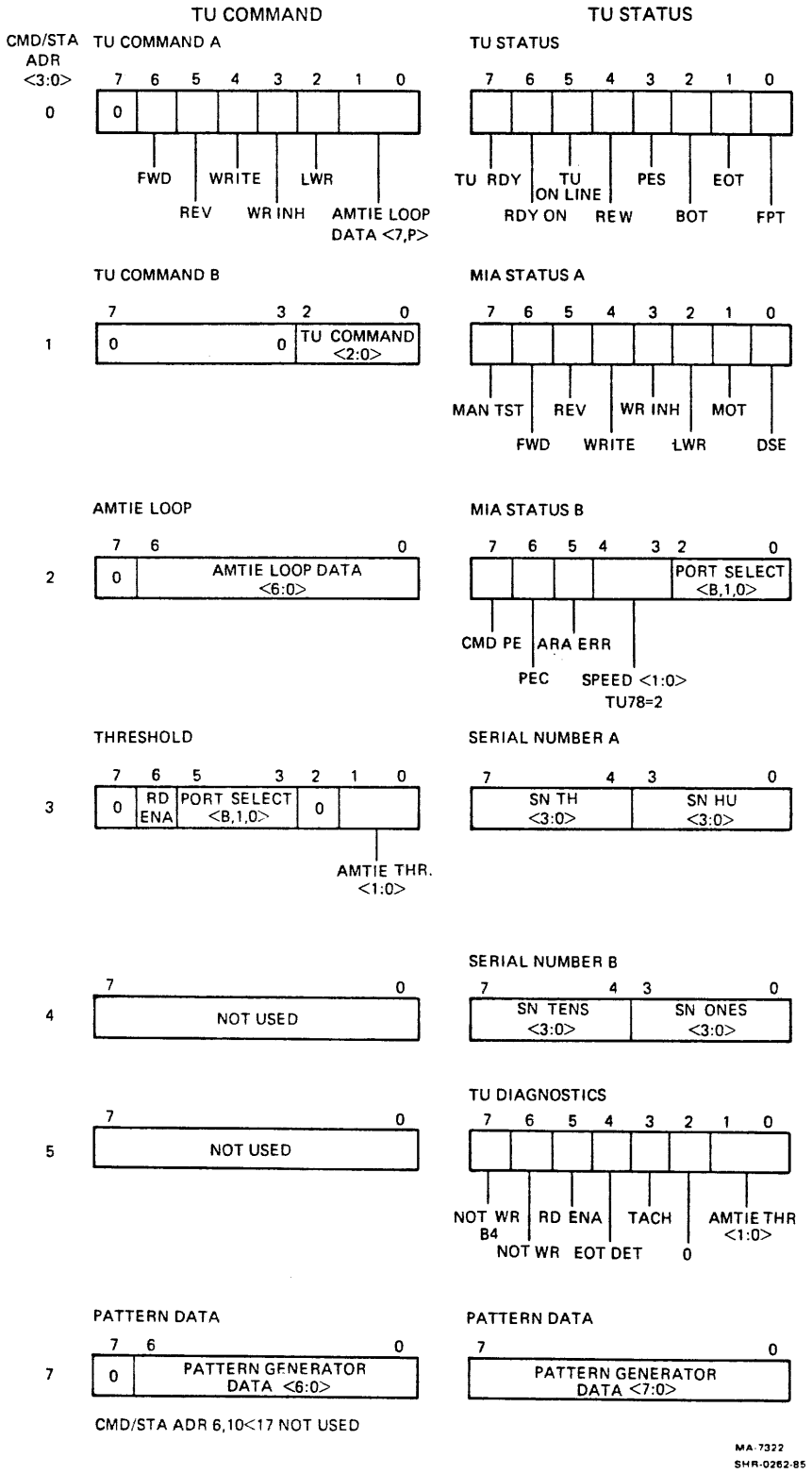


Table C-1 TU Command Register Descriptions

Bits	Name	Description
TU CMD/STA ADR = 0 (TU COMMAND A)		
7		— Zero
6		Forward Commands the tape unit (if ready and on-line) to move tape in the forward direction.
5		Reverse Commands the tape unit (if ready and on-line) to move tape in the reverse direction. The tape stops upon detection of BOT.
4		Write The tape unit (if on-line and not file-protected or rewinding) is enabled to erase tape. If not write-inhibited, then the tape may also be written.
3		Write Inhibit Prevents the tape unit's write head from being energized, while allowing the erase head to operate normally. Meaningful only when write = 1.
2		LWR Loop write-to-read. The tape unit drives the nine read data (RD) lines from the corresponding nine WCS lines, and the nine AMTIE lines from the corresponding nine AMTIE loop bits.
1.0		AMTIE Loop These bits drive AMTIE (amplitude track in error) bus lines <7,P> respectively when (transport) LWR is set.
TU CMD/STA ADR = 1 (TU COMMAND B)		
7:3	—	Zero
2:0	TU command 2:0	Provides tape unit command function as follows.

Octal Value	Function	Description
0	CLR TU	Clears FWD, REV, WRITE, WR INH, LWR, DSE, CMD PE, and RDY ON.
1	SET PE	Sets TU to phase-encoded recording format.
2	SET GCR	Sets TU to group-coded recording format.

Table C-1 TU Command Register Descriptions (Cont)

Bits	Name	Description

TU CMD/STA ADR = 1 (TU COMMAND B)

Octal Value	Function	Description
3	CLR EOT	Clears EOT status.
4	REWIND	If ready and on-line, the tape is rewound to BOT. RDY = 0 until the rewind sequence has completed.
5	UNLOAD	If ready and on-line, the transport is switched off-line; the tape is rewound and removed from the tape path.
6	DSE	Causes FWD to set until EOT sets or a CLR TU command occurs.
7	TEST	Sets RDY ON and CMD PE, causing the CMD PE TU bus line to assert.

TU CMD/STA ADR = 2 (AMTIE LOOP)

Bits	Name	Description
7	—	Zero
6:0	AMTIE loop 6:0	These bits drive AMTIE bus lines 6:0 respectively when LWR (transport) is set.

TU CMD/STA ADR = 3 (THRESHOLD)

7	—	Zero
6	Read enable	Enables tape unit read/AMTIE signals to TU bus RD/AMTIE lines.
5:3	Port select <B,1,0>	The complement of these bits is logically ORed with the tape unit's port select switch. The bits are normally set to allow the position of the port select switch to be read via status address 2 MIA STATUS B.

Table C-1 TU Command Register Descriptions (Cont)

Bits	Name	Description
2	High-read threshold	When set, tapes are read using the high-read amplifier threshold. If reset, the read electronics operate in the low-read threshold mode.
TU CMD/STA ADR = 3 (THRESHOLD)		
1,0	AMTIE threshold 1:0	This 2-bit field establishes the threshold of the TU read electronics for the nine AMTIE signals.

Octal Value	Threshold	Mode
0/1	10%	IBG check
2	25%	Write
3	20%	Read

TU CMD/STA ADR = 7 (PATTERN DATA)		
7	—	Zero
6:0	Pattern generator data	Represents data field intended to be written to tape for maintenance data purposes. This register is actually eight bits wide. On every write to this register, bit 7 is set to a 1. The loaded pattern is generated on all nine channels with bit 0 first, then bit 1, until all have been sent.

Table C-2 TU Status Register Descriptions

Bits	Name	Description
TU CMD/STA ADR = 0 (TU STATUS)		
7	Ready	The tape unit is on-line, not rewinding, not loading or unloading, and all pneumatic interlocks are made.
6	Ready on	Ready has undergone a transition to the set state. Cleared by a CLR TU command.
5	On-line	The tape unit has been placed on-line; if ready, it responds to motion commands issued remotely from the formatter.
4	Rewinding	The tape unit is currently rewinding.
3	PES	The tape unit is currently set to phase-encoded recording format. If reset, the tape unit is set to group-coded recording format.
2	BOT	The BOT marker on the tape is positioned at the BOT sensor and ready is set.
TU CMD/STA ADR = 0 (TU STATUS)		
Bits	Name	Description
1	EOT	The EOT marker on the tape is, or was, positioned at the EOT sensor during a forward operation with ready set. EOT status clears when: • The EOT marker is positioned at the EOT sensor during a reverse operation with ready set • The tape unit is commanded to rewind tape • A CLR EOT command is written into the TU CMD B location.
0	File protect	The tape reel currently loaded does not have the write enable ring in place.

Table C-2 TU Command Register Descriptions (Cont)

Bits	Name	Description
TU CMD/STA ADR = 1 (MIA STATUS A)		
7	Manual test	The manual test switch is in the MAN TEST position.
6	Forward	The forward command bit has been written with a 1, or a DSE command has been written into the TU CMD B location.
5	Reverse	Indicates the state of the reverse command bit.
4	Write	Indicates the state of the write command bit.
3	Write Inhibit	Indicates the state of the write inhibit command bit.
2	LWR	Indicates the state of the loop write-to-read command bit.
1	MOT	Indicates that the capstan servo motor is turning.
0	DSE	Indicates that a DSE command has been written into the TU CMD B location. Clears when EOT sets, or when a CLR TU command is written into TU CMD B.
TU CMD/STA ADR = 2 (MIA STATUS B)		
7	CMD PE	A command or CMD/STA address has been received by the tape unit with even parity, or a TEST command has been written into TU CMD B. Clears when a CLR TU command is received, whether parity is correct (odd) or not. This bit drives the TU bus CMD PE L line.
6	PEC	A SET PE command has been sent to TU CMD B. Assuming proper hardware operation, this bit is the same as PES in TU STATUS.
5	ARA error	During a GCR operation, from 24.1 cm (9.5 in) after BOT, the read amplifier gains failed to achieve the required value. Valid only if ready is set.
4,3	Speed 1:0	This 2-bit field indicates the speed of a tape unit; TU79 = 2 (125 ips).

Table C-2 TU Command Register Descriptions (Cont)

Bits	Name	Description
2,1,0	Port Select B,1,0	Indicates the position of the multiposition port select switch on the tape unit operator panel. The bits are meaningful only when the port select bits in the threshold command location have been written to ones.

Octal Values

B	1	0	Meaning
1	1	0	Position 0 (MASSBUS A)
1	0	1	Position 1 (MASSBUS B)
0	1	1	Position 2 or both (MASSBUS A or B)
1	1	1	Any other position (Select neither MASSBUS.)

TU CMD/STA ADR = 3 (SERIAL NUMBER A)

7:4	SN TH 3:0	Most significant (thousands) BCD digit of the tape unit's serial number.
3:0	SN HU 3:0	The hundreds BCD digit of the tape unit's serial number.

TU CMD/STA ADR = 4 (SERIAL NUMBER B)

7:4	SN tens 3:0	The tens BCD digit of the tape unit's serial number.
3:0	SN ones 3:0	Least significant (ones) BCD digit of the tape unit's serial number.

Table C-2 TU Status Register Descriptions (Cont)

Bits	Name	Description
TU CMD/STA ADR = 5 (TU DIAGNOSTICS)		
7	Not write bit 4	Indicates the state (polarity) of the write driver for physical track 4 head.
6	Not write	There is no current flowing through the write or erase heads (same as MIA STATUS A, bit 7).
5	Read enable	Indicates the state of the read enable bit in the THRESHOLD command location.
4	EOT DET	The tape's EOT tab is positioned at the EOT sensor. Not valid unless ready = 1.
3	TACH	The state of the output from the capstan motor digital tachometer.
2	High-read threshold	Indicates the state of the high-read threshold bit in the THRESHOLD command location.
1,0	AMTIE THR 1:0	Indicates the value of the AMTIE threshold field in the THRESHOLD command location.
TU CMD/STA ADR = 7 (PATTERN DATA)		
7:0	Pattern generator data	This data field is currently in the PATTERN GENERATOR command location.

APPENDIX D
TM78 EXTENDED SENSE COMMAND (73) DATA BYTES

Bytes	Description
1	Command code being executed on last error
2	Interrupt code from last error
3	Failure code from last error
4	Hardware register 0; read path write fail bits
5	Hardware register 1; read path diagnostic bits
6	Hardware register 2; read path status
7	Hardware register 3; read path command loop
8	Hardware register 20; AMTIES (CH 7:0). Immediate status of the AMTIE lines from the tape drive. This is a dynamic read of the M8950 done bits by the M8953 read path microcontroller.
9	Hardware register 21; RC DONE (CH 7:0). This bit will be false if the associated M8950 has completed its assigned task.
10	Hardware register 22; (CH 7:0) GCR illegal 5-4. PE data error such as no bit time phase transition.
11	Hardware register 23; mark 2 (CH 7:0 — RMK2)
12	Hardware register 24; end mark (CH 7:0 — REND) for read channels — location: M8950(s).
13	Hardware register 25; RC PAR bits
14	Hardware register 26; read channel PE postamble detect register (CH 7:0): location — M8950(s). In PE, contains the inverted data for the byte prior to the one currently on the data lines (RC30H). Used for PE postamble detection. Not used in GCR.
15	Hardware register 27; R data (CH 7:0). Data output from M8950 to ECC — M8951.
16	Hardware register 30; CRC word. CRC checker output bits — location: M8952.

Bytes	Description
17	Hardware register 31; ECCOR. Corrected data (CH 7:0) output from M8951 to M8952.
18	Hardware register 32; ECC status — M8951
19	Hardware register 40; channel 0 TIE bus
20	Hardware register 41; channel 1 TIE bus
21	Hardware register 42; channel 2 TIE bus
22	Hardware register 43; channel 3 TIE bus
23	Hardware register 44; channel 4 TIE bus
24	Hardware register 45; channel 5 TIE bus
25	Hardware register 46; channel 6 TIE bus
26	Hardware register 47; channel 7 TIE bus
27	Hardware register 50; channel P TIE bus
28	Hardware register 60; RTIER. Read channel bypass for diagnostic purposes.
29	Hardware register 104; TAMT. Diagnostic AMTIE loop register in M8955 module.
30	Hardware register 110; PORT status
31	Hardware register 114; PRDD
32	Hardware register 240; CAS status
33	Hardware register 241; CBUSSTA status
34	Hardware register 300; DBUSSTA status
35	Hardware register 320; WMC status. See Appendix C for bit descriptions.
36	Hardware register 321; TU select 0 (M8955-slot 8). See Appendix C for bit descriptions.
37	Hardware register 322; TU select 1 (M8955-slot 9). See Appendix C for bit descriptions.
38	Hardware register 323; WRTDAT. Write microcontroller data-M8959 module.
39	Hardware register 324; BYTCNT <7:0>. Write byte count lo word.
40	Hardware register 324; BYTCNT <15:8>. Write byte count hi word.
41	Hardware register 325; PADCNT <7:0>. Write microbyte assembly pad character counter lo word.
42	Hardware register 325; PADCNT <15:8>. Write microbyte assembly pad character counter hi word.

Bytes	Description
43	Hardware register 326; ERRCNT <7:0>. Hi word-bits 8-15. Write micro-byte assembly error code.
44	Hardware register 326; ERRCNTH <7:0>. Word-bits 0-17. See Appendix C for bit codes.
45	Hardware register 330; DDRA. Diagnostic data register A used to test bus between M8956 and M8959.
46	Hardware register 331; DDRB. Diagnostic data register B used to test bus between M8956 and M8959.
47	Hardware register 332; WMCERR.
48	Hardware register 340; INSTA (interrupt status byte on M8960).
49	MIA register 0; TU79 status
50	MIA register 1; MIA status A
51	MIA register 2; MIA status B
52	MIA register 3; SNTH. Serial number: thousands and hundreds digits (BCD).
53	MIA register 4; SNTU. Serial number: tens and units digits (BDC).
54	MIA register 5; TU diag
55	Retry counter (RETCNT). This byte is the count of retry interrupt requests given for the tape unit. When this count is zero, the tape unit is not in a retry sequence.
56	Retry control bits (RETCNT+1). This byte is used by the microcode to control error recovery. It is meaningful only when the retry counter (byte 55) is not zero. Bit 5 — set when initial command moved in the reverse direction. Bit 6 — set when initial command was a read Bit 7 — set when last retry requested was in opposite direction of initial command
57	TU software status (TUX). This byte contains information about the tape drive. Bit 0 — set when a data security erase command is in progress Bit 1 — set when a rewind command is in progress Bit 2 — set when tape unit exists and power is on Bit 3 — set when a non-data transfer command issued from a MASSBUS port is in progress Bit 4 — set when tape was last moved in the reverse direction

Bytes	Description
	Bit 5 — set when last tape operation involved writing on tape
	Bit 6 — set when last record seen was a tape mark
	Bit 7 — set when last Massbus command came from port B
58	Transfer control word (XFRCTL). This byte contains control information used by data transfer commands.
	Bit 0-2 — write clock select
	Bit 3-5 — read clock select
	Bit 6 — PLO bypass
	Bit 7 — low read threshold
59	Retry suppress and format control (XRETRY). This byte contains the contents of the left half of the Massbus register (CAS register 2) which contains the SER (Suppress Internal Error Repositioning) bit, format, and byte count.
60	Keypad enable flag (ENAON). This byte is not zero when the keypad is enabled.

APPENDIX E
TM78 INTERNAL I/O REGISTERS

This appendix describes the internal TM78 microbus hardware registers that can be accessed by the maintenance keypad/display (via the M8960 Microcomputer) or the host CPU (via Massbus registers R20 and R21).

The list is in I/O address numerical order. All addresses given are in the range of $0—377_8$ and represent microcomputer addresses for data transfer to/from the 8085 accumulator. Except where noted, all locations may also be accessed by the host CPU. You can do this by setting the operand address in Massbus register R20 equal to 100nnn (where nnn = $0—377_8$). Following the octal address is the letter R or W. This indicates whether the register is read only or write only.

The following table is a cross reference between the register's name and its internal address.

Register Name	Octal Address	Register Name	Octal Address
BYTCNT	324-R	DBUSCTL	300-W
BYTCNT	324-W	DBUSSTA	300-R
CAS	200-R/W	DDR A	330-R
CAS	237-R/W	DDR A	330-W
CASCTL	240-W	DDR B	331-R
CASSTA	240-R	DDR B	331-W
CBUSSTA	241-R	DDR C	332-W
CHOTIE	040-R	DDRCTL	333-W
CH1TIE	041-R	ECCOR	031-R
CH2TIE	042-R	ECCSTA	032-R
CH3TIE	043-R	ERRCNT	326-R
CH4TIE	044-R	ERRCNT	326-W
CH5TIE	045-R	INSTA	340-R
CH6TIE	046-R	KEYBRD	310-W
CH7TIE	047-R	LDLEDA	312-W
CHPTIE	050-R	LDLEDB	313-W
CLKCTL	360-W	LDLEDC	314-W
CNTCTL	327-W	LDLEDD	315-W
CRCWRD	030-R	LDLEDE	316-W
DATACTL	320-W	LDLEDF	317-W

Register Name	Octal Address	Register Name	Octal Address
MBSEL	340-W	RPCTL	011-W
PADCNT	325-R	RPFAIL	000-R
PADCNT	325-W	RPOSTN	026-R
PDIAG	110-W	RPSTA	025-R
PENAB	114-W	RSTAT	002-R
PRDD	114-R	RTIEB	012-W
PSTAT	110-R	RTIER	060-R
RAMT	020-R	SETATA	241-W
RCMD	013-W	TAMT	104-R
RCMLP	003-R	TAMT	104-W
RDATA	027-R	TCMP	100-W
RDON	021-R	TRKENA	322-W
REND	024-R	TSTS	100-R
RESCHR	321-W	TUSELO	321-R
RFIFOL	010-W	TUSEL1	322-R
RILL	022-R	WMCCTL	323-W
RINST	014-W	WMCERR	332-R
RMK2	023-R	WMCSTA	320-R
RPATH	001-R	WRTDAT	323-R

RMC WRITE FAIL BITS 7:0

Register name: RPFAIL
Microbus address: 000-R
Register location(s): M8950s (RC3) via M8953 (RP6)

Bit	Name	Description
7:0	Write fail 7:0	These are write fail bits from read channels 7—0. Write fail is an OR of certain error conditions, namely: Illegal 5-to-4 translation AMTIE PHTIE Pointer mismatch

READ PATH STATUS

Register name	RPATH
Microbus address	001-R
Register location(s)	M8953 (RPG)

Bit	Name	Description
7	Velocity ok	True if velocity is within 10% of 125 in/sec or jam velocity ok bit is set (RPCTL, 011-W bit 2).
6	Status valid	Indicates M8953 status is valid (set after 0.1 inch gap found).
5	Preamble error	True if preamble had AMTIE.
4	Data ready	Goes false when M8953 asserts data ready to M8951.
3	Beginning of preamble	True when read path senses that the preamble has just passed the read head.
2	Clock stopped	True if read path clock is stopped.
1	Statistics select	True if TIE bus statistics are being sent to TIE bus.
0	Write fail P	Indicates illegal 5 to 4, AMTIE, PHTIE, or pointer mismatch for parity bit.

RMC STATUS WORD

Register name	RSTAT
Microbus address	002-R
Register location(s)	M8950s via M8953 (RP4)

Bit	Name	Description
7:0	RMC status 7:0	Read path microcontroller status codes return in this register. The following table identifies each code and its meaning.

Read Path Microcontroller Status Words

Status	Meaning
Status from ECC self-test command	
101	ECC controller passed self-test.
102	ECC controller failed self-test.
Status from an M8953 self-test	
103	Read path passed self-test.
104	Read path failed self-test.

Status	Meaning
Status from an M8950 self-test command	
106	Read channel tests all passed.
Status from a clear all test command for velocity testing of drive by microcode	
1	First tach pulse
201	Last tach pulse (eleventh)
Status from a sample density command	
210	NOT CAPABLE found.
211	GCR ID found.
212	PE ID found.
Status from a write test of IBG, PE ID, GCR ID, ARA ID, or ARA burst	
220	Bad status (write test).
Status from a tape mark test command	
222	Good tape mark found on tape status.
Status from a non-BOT command (read or write FWD or REV, GCR or PE)	
230	ARA ID found (not record or tape mark).
231	Tape mark found.
234	Preamble end not found.
235	Read path fault 1, too many M8950s have been fataled to continue record processing.
236	Read path fault 2, 7 or more M8950 modules found illegal 5-to-4 translations.
241	Unexpected IBG in data; probably creased tape (seven or more AMTIES active).
261	Postamble long.
262	Postamble short.
377	OK

RMC COMMANDS LOOP BACK REGISTER

Register name	RCMLP
Microbus address	003-R
Register location(s)	M8953 (RP6)

Bit	Name	Description
7:0	Command 7:0	Contains the last command sent to the M8953 via RCMD (address 013-W).

TEST FIFO LOAD

Register name	RFIFOL
Microbus address	010-W
Register locations	M8953 (RP5)

Bit	Name	Description
7:0	—	Not used. Any write to this register generates a test FIFO load pulse to the nine read channel modules. Used in PLO bypass modes 2 and 3 only.

READ PATH DIAGNOSTIC CONTROL REGISTER

Register name	RPCTL
Microbus address	011-W
Register location(s)	M8953 (RP6)

Bit	Name	Description
7	Jam TIE bus	Used for diagnostics. Causes the TIE bus jam register to be used in place of the TIE data from the M8950 modules.
6	RC test data	Test data for read channel FIFOs.
5:4	PLO bypass	
	1:0	

Bit Value	FIFO Data	FIFO Clock
0	PLO	PLO
1	PLO	XMC W CLOCK
2	TU PORT	TEST FIFO LD
3	RC TEST	TEST FIFO LD DATA

3	PLO disable	Disables all M8950 PLOs.
2	Jam velocity ok	Forces velocity ok.
1	Statistics select	Causes statistic register in M8950s to be applied to TIE bus.
0	Stop clock	Stops the read path clock so that it can be single stepped. Also allows addresses 020 — 027 to be read.

RMC TIE BUS JAM REGISTER

Register name	RTIEB
Microbus address	012-W
Register location(s)	M8953

Bit	Name	Description
7:4	—	Not used.
3:0	TIE jam 3:0	These bits may be used instead of TIE bus data from M8950s. Jam TIE bus bit 7 must be set in RPCTL register (address 011-W).

READ MICROCONTROLLER COMMAND WORD

Register name	RCMD
Microbus address	013-W
Register location(s)	M8953 (RP6)

Bit	Name	Description
7	Run continuous	When set, allows RMC command codes 01-06 to run continuously.
6:0	RMC command 6:0	This register sets the read path task. Any write causes the M8953 program counter to be initialized, then interpret the task. The table below lists the tasks with the associated command code.

Command Code	Task
00	NOP
01	Interblock read
02	Test PE ID burst
03	Test GCR ID burst
04	Test ARA ID burst
05	Test tape mark
06	Test ARA burst
07	Normal non-BOT read
10	Run RMC self-test
11	Test unknown ID burst
12	Run read channel micro's test
13	Diagnostic read command
14	Run read channel self-test
15	Run clear all RMC test program
16	Run ECC self-test program
17	Find gap

RMC SINGLE-STEP CLOCK WORD

Register name	RINST
Microbus address	014-W
Register location	M8953 (RP5)

Bit	Name	Description
7:0	—	Not used. Any write to this register causes the read microcontroller to advance one microinstruction. Note that the read path normal clock must be stopped, and RPCTL (011-W) bit 0 = 1

READ PATH TRIGGER

Register name	RPTRIG
Microbus address	015-W
Register location(s)	M8953 (RP5)

Bit	Name	Description
7:0	—	Not used. Any write to this register produces a low-going scope trigger pulse at pin C10K1.

READ CHANNEL AMTIE STATUS*

Register name	RAMT
Microbus address	020-R
Register location(s)	Tape transport via M8955, M8950, and M8953

Bit	Name	Description
7:0	AMTIE 7:0	This register reflects the state of the AMTIE lines for channels 7 through 0.

* The read clock must be stopped before reading this register.

READ CHANNEL DONE STATUS*

Register name	RDON
Microbus address	021-R
Register location(s)	M8950s via M8953

Bit	Name	Description
7:0	7:0	A given bit is false if the associated M8950 has completed its assigned task.

READ CHANNEL ILLEGAL STATUS*

Register name	RILL
Microbus address	022-R
Register location(s)	M8950 via M8953

Bit	Name	Description
7:0	7:0	A given bit is true if the associated M8950 indicates the following. GCR Data contained an illegal 5-bit code which was either an error or a tape format control character. PE Data error occured such as no bit time phase transition.

READ CHANNEL MARK 2 STATUS*

Register name	RMK2
Microbus address	023-R
Register location(s)	M8950s via M8953

Bit	Name	Description
7:0	7:0 Mark 2	When true, indicates that the associated M8950 has detected a Mark 2 tape format control character. If the corresponding illegal 5-to-4 bit is set (RILL 7:0), a Mark 2 was detected during 5-to-4 conversion.

* The read clock must be stopped before reading this register.

READ CHANNEL END STATUS

Register name	REND
Microbus address	024-R
Register location(s)	M8950s via M8953

Bit	Name	Description
7:0	END 7:0	End Mark for read channels 0 — 7.

READ CHANNEL PARITY STATUS*

Register name	RPSTA
Microbus address	025-R
Register location(s)	M8950 parity module

Bit	Name	Description
7	Corrected data P	Contains ECC module corrected output data for parity bit.
6	Data P	Contains M8950 data output for parity bit.
5	Post P	Indicates PE postamble for parity bit.
4	End P	Indicates End Mark for parity bit.
3	Mark 2 P	Indicates Mark 2 for parity bit.
2	Illegal P	Indicates illegal 5-to-4 for parity bit.
1	Not done P	Parity M8950 not done.
0	AMTIE P	Weak amplitude on parity bit.

READ CHANNEL PE POSTAMBLE DETECT REGISTER*

Register name	RPOSTN
Microbus address	026-R
Register location(s)	M8950s via M8953

Bit	Name	Description
7:0	Postamble 7:0	In PE, contains the inverted data for the byte prior to the one currently on the data lines (RC3 D(N) H). Used for PE POSTAMBLE detection. Not used in GCR.

* The read clock must be stopped before reading this register.

READ CHANNEL DATA REGISTER*

Register name	RDATA
Microbus address	027-R
Register location(s)	M8950s via M8953

Bit	Name	Description
7:0	Data 7:0	Contains data output from read channels 7—0 to ECC module.

CRC WORD

Register name	CRCWRD
Microbus address	030-R
Register location(s)	M8952 (CRC3)

Bit	Name	Description
7:0	CRC 7:0	These are CRC checker output bits.

CORRECTED DATA REGISTER

Register name	ECCOR
Microbus address	031-R
Register location(s)	M8951 (EC5)

Bit	Name	Description
7:0	Corrected data 7:0	Contains corrected data output from ECC module to CRC module.

ECC STATUS WORD

Register name	ECCSTA
Microbus address	032-R
Register location(s)	M8951 (EC5)

Bit	Name	Description
7	CRC error	Indicates CRC did not check.
6	ECC ROM parity error	Indicates M8951 program parity error.
5	AMTIE occurred	Indicates AMTIE occurred during data portion of record.
4	ACRC error	Indicates ACRC did not check.
3	Pointer mismatch	Indicates the track in error did not have a pointer.
2	Uncorrectable	Indicates ECC could not correct data error.

* The read clock must be stopped before reading this register.

1	Two-track error correction	Indicates two-track error correction was performed on data.
0	Single-track error correction	Indicates single-track error correction was performed on data.

READ CHANNEL 0 TIE BUS REGISTER

Register name	CH0TIE
Microbus address	040-R
Register location(s)	M8950 channel 0

Bit	Name	Description
7:0	0	Zero
3	TIE bus 3	Indicates illegal 5-to-4 in GCR, and dead track in PE. (Bad track register is ORed into this bit.)
2	TIE bus 2	AMTIE
1	TIE bus 1	PHTIE
0	TIE bus 0	If statistics = 0, indicates pointer mismatch in GCR and 0 in PE. If statistics = 1, indicates any pointer occurred (GCR and PE).

READ CHANNEL 1 TIE BUS REGISTER

Register name	CH1TIE
Microbus address	041-R
Register location(s)	M8950 channel 1

Bit	Name	Description
7:4	0	Zero
3	TIE bus 3	Indicates illegal 5-to-4 in GCR, and dead track in PE. (Bad track register is ORed into this bit.)
2	TIE bus 2	AMTIE
1	TIE bus 1	PHTIE
0	TIE bus 0	If statistics = 0, indicates pointer mismatch in GCR and 0 in PE. If statistics = 1, indicates any pointer occurred (GCR and PE).

READ CHANNEL 2 TIE BUS REGISTER

Register name	CH2TIE
Microbus address	042-R
Register location(s)	M8950 channel 2

Bit	Name	Description
7:4	0	Zero
3	TIE bus 3	Indicates illegal 5-to-4 in GCR, and dead track in PE. (Bad track register is ORed into this bit.)
2	TIE bus 2	AMTIE
1	TIE bus 1	PHTIE
0	TIE bus 0	If statistics = 0, indicates pointer mismatch in GCR and 0 in PE. If statistics = 1, indicates any pointer occurred (GCR and PE).

READ CHANNEL 3 TIE BUS REGISTER

Register name	CH3TIE
Microbus address	043-R
Register location(s)	M8950 channel 3

Bit	Name	Description
7:4	0	Zero
3	TIE bus 3	Indicates illegal 5-to-4 in GCR, and dead track in PE. (Bad track register is ORed into this bit.)
2	TIE bus 2	AMTIE
1	TIE bus 1	PHTIE
0	TIE bus 0	If statistics = 0, indicates pointer mismatch in GCR and 0 in PE. If statistics = 1, indicates any pointer occurred (GCR and PE).

READ CHANNEL 4 TIE BUS REGISTER

Register name	CH4TIE
Microbus address	044-R
Register location(s)	M8950 channel 4

Bit	Name	Description
7:4	0	Zero
3	TIE bus 3	Indicates illegal 5-to-4 in GCR, and dead track in PE. (Bad track register is ORed into this bit.)
2	TIE bus 2	AMTIE
1	TIE bus 1	PHTIE
0	TIE bus 0	If statistics = 0, indicates pointer mismatch in GCR and 0 in PE. If statistics = 1, indicates any pointer occurred (GCR and PE).

READ CHANNEL 5 TIE BUS REGISTER

Register name	CH5TIE
Microbus address	045-R
Register location(s)	M8950 channel 5

Bit	Name	Description
7:4	0	Zero
3	TIE bus 3	Indicates illegal 5-to-4 in GCR, and dead track in PE. (Bad track register is ORed into this bit.)
2	TIE bus 2	AMTIE
1	TIE bus 1	PHTIE
0	TIE bus 0	If statistics = 0, indicates pointer mismatch in GCR and 0 in PE. If statistics = 1, indicates any pointer occurred (GCR and PE).

READ CHANNEL 6 TIE BUS REGISTER

Register name	CH6TIE
Microbus address	046-R
Register location(s)	M8950 channel 6

Bit	Name	Description
7:4	0	Zero
3	TIE bus 3	Indicates illegal 5-to-4 in GCR, and dead track in PE. (Bad track register is ORed into this bit.)
2	TIE bus 2	AMTIE
1	TIE bus 1	PHTIE
0	TIE bus 0	If statistics = 0, indicates pointer mismatch in GCR and 0 in PE. If statistics = 1, indicates any pointer occurred (GCR and PE).

READ CHANNEL 7 TIE BUS REGISTER

Register name	CH7TIE
Microbus address	047-R
Register location(s)	M8950 channel 7

Bit	Name	Description
7:4	0	Zero
3	TIE bus 3	Indicates illegal 5-to-4 in GCR, and dead track in PE. (Bad track register is ORed into this bit.)
2	TIE bus 2	AMTIE
1	TIE bus 1	PHTIE
0	TIE bus 0	If statistics = 0, indicates pointer mismatch in GCR and 0 in PE. If statistics = 1, indicates any pointer occurred (GCR and PE).

READ CHANNEL P TIE BUS REGISTER

Register name	CHPTIE
Microbus address	050-R
Register location(s)	M8950 channel P

Bit	Name	Description
7:4	0	Zero
3	TIE bus 3	Indicates illegal 5-to-4 in GCR, and dead track in PE. (Bad track register is ORed into this bit.)
2	TIE bus 2	AMTIE
1	TIE bus 1	PHTIE
0	TIE bus 0	If statistics = 0, indicates pointer mismatch in GCR, and 0 in PE. If statistics = 1, indicates any pointer occurred (GCR and PE).

READ CHANNEL TIE BUS REGISTER

Register name	RTIER
Microbus address	060-R
Register location(s)	M8953

Bit	Name	Description
7:4	0	Zero
3:0	TIE bus	Reads TIE bus without enabling any read channel modules. If jam TIE bus (address 011-W bit 7) is set, then the value written into address 012-W is read.

NOTE
Before performing I/O write/read operations to microbus addresses 100 — 114 (TU port/MIA registers), you must first select the desired TU port. This is done by writing the select address in MBSEL (340-W) bits 1:0.

TU CMD/STAT ADDRESS TAPE UNIT COMMAND

Register name	TCMP
Microbus address	100-W
Register location(s)	M8955/MIA (TU79)

Bit	Name	Description
—	—	Refer to Appendix C.

TAPE UNIT STATUS

Register name	TSTS
Microbus address	100-R
Register location(s)	M8955/MIA (TU79)

Bit	Name	Description
—	—	Refer to Appendix C.

AMTIE LOOP

Register name	TAMT
Microbus address	104-W
Register location(s)	M8955 (TUP 1, 3)

Bit	Name	Description
7:0	AMTIE loop data 7:0	Data sent to this register is latched and placed on TU BUS AMTIE 7:0 L.

NOTE
AMTIE P L must be asserted via PDIAG (110-W) bit 0.

NOTE
If PENAB (114-W) bit 2 (RP EN) is asserted, data is placed on TUP P AMTIE 7:0 H.

AMTIE WORD

Register name	TAMT
Microbus address	104-R
Register location(s)	M8955 (TUP 2, 4)

Bit	Name	Description
7:0	AMTIE 7:0	This register reflects information on TU bus line AMTIE 7:0 L.

TU PORT DIAGNOSTIC REGISTER

Register name	PDIAG
Microbus address	110-W
Register location(s)	M8955 (TUP 2, 4)

Bit	Name	Description
7	—	Not used.
6	Loop command to status	Setting this bit inhibits TU bus lines TU CMD L and TU STAT L so that MIA registers are not strobed during a diagnostic write/read to the TU port CMD/STAT registers.
Bit	Name	Description
5	Loop write-to-read	Setting this bit allows information on the TU bus WCS lines (TU WCS 7:0, P) and bits 3, 1, 0 of this register to be placed on TU bus lines TU RD 7:0; RD P, AMTIE P, TACH.

NOTE
If PENAB (114-W) bits 4 (WP EN) and 2 (RP EN) are set, loop write-to-read allows data from the write path (WDO 7:0, P) to be sent to the read path (P RD 7:0,P).

4	—	Not used.

NOTE
Bits 3, 1, and 0 may be set in the PDIAG register, but will not serve any function unless bit 5 (LWR) is set.

3	Send TACH	Setting this bit asserts the TU bus TACH line and simulates the output of the TU79 capstan servo tachometer.
2	—	Not used.
1	Send RD P	Setting this bit asserts the TU bus line RD P.
0	Send AMTIE P	Setting this bit asserts the TU bus line AMTIE P.

TAPE UNIT PORT STATUS WORD

Register name	PSTAT
Microbus address	110-R
Register location(s)	M8955 (TUP 2, 4)

Bit	Name	Description
7	Status parity error	Indicates a parity error has occurred when reading a TU79 register over the WCS lines.
6	Write data strobe	Reflects the state of the TU bus line WDS.
5	Command parity error	Indicates the TU79 has detected a parity error on the WCS lines during a TM78 command or command/status address write.
4	Tape unit present	Indicates that a TU79 transport is physically cabled to this TU port.
3	Tachometer	Reflects the state of the TU bus line TACH.
2	WCS parity	Reflects the state of the TU bus line WCS P.
1	Read parity	Reflects the state of the TU bus line RD P.
0	AMTIE parity	Reflects the state of the TU bus line AMTIE P.

PORT ENABLE CONTROL REGISTER

Register name	PENAB
Microbus address	114-W
Register location(s)	M8955 (TUP 2, 4)

Bit	Name	Description
7	Interrupt enable	When set, allows the STAT PE/CMD PE bits (PSTAT 7, 5) to interrupt the microcomputer program. When reset, disallows the hardware interrupt but the status bits may be read via the PSTAT register (110-R bits 7, 5).
6, 5	—	Not used.
4	Write path enable	When set, enables write data to be placed on the WCS lines and disables the command/status functions. When reset, enables the WCS lines to be used for command/status information and inhibits write data.
3	—	Not used.
2	Read path enable	When set, allows TU bus signals RD 7:0, P; AMTIE 7:0, P; and TACH to be gated to the read path.
1, 0	—	Not used.

PORT READ DATA WORD

Register name	PRDD
Microbus address	114-R
Register location(s)	M8955 (TUP 2, 4)

Bit	Name	Description
7:0	Read data 7:0	This register reflects information on TU bus line RD 7:0 L.

NOTE
Before performing I/O write/read operations to microbus addresses 200 — 300 (Massbus port registers), you must first select the desired Massbus port. This is done by writing the select address in MBSEL (340-W) bit 7.

COMMON ADDRESS SPACE

Register name	CAS
Microbus address	200 — 237-R/W
Register location(s)	M8957 (MBC 1)

Bit	Name	Description
—	—	Refer to Chapter 2 for a description of CAS contents and Figure 2-9 for an illustration of CAS layout. Massbus register 0 LO byte corresponds to microbus address 200, register 0 HI byte to address 201, and so on to register 17 HI byte corresponding to address 237.

NOTE
CAS registers cannot be accessed via Massbus register R21.

NOTE
Reading or writing these registers may cause a contention error. In that event the read or write should be retried.

CAS CONTROL

Register name	CASCTL
Microbus address	240-W
Register location(s)	M8957 (MBC2)

Bit	Name	Description
7	Contention	Sets the contention error bit as a diagnostic check. Clears the contention error for a retry. Causes an internal microcomputer interrupt to Restart 6.5.
6	TM ready	Sets the TM78 ready bit to the selected Massbus port.
5:0	—	Not used.

CAS STATUS

Register name	CASSTA
Microbus address	240-R
Register location(s)	M8957 (MBC2)

Bit	Name	Description
7	Contention	This is a CAS contention error. Both the host computer and the microcomputer tried to access the common address space at the same time. Causes an internal microcomputer interrupt to Restart 6.5.
6	TM ready	Indicates this Massbus port is ready to accept a command from the host computer. Used by the microcomputer to detect when the host computer initiates a reset (by seeing this bit clear).
5	CAS parity error	Indicates a Massbus control bus parity error has been detected on a write to the TM78. Causes an internal microcomputer interrupt to Restart 5.5.
4	Illegal register	Indicates the host computer attempted to access an illegal Massbus register in the range of 22_8 — 37_8. Causes an internal microcomputer interrupt to Restart 5.5.
3	Attention	Indicates the Massbus attention line for this port is active. The host computer has not completed the ATTN interrupt service routine yet.
2:0	Drive select 2:0	This is an octal drive number, set in the DS switches for the selected Massbus port. The DS switches are located on the backplane.

SET ATTENTION

Register name	SETATA
Microbus address	241-W
Register location(s)	M8957 (MBC 3)

Bit	Name	Description
7:0	Attention	Any write to this register sets ATTN active to the host computer.

CBUS STATUS

Register name	CBUSSTA
Microbus address	241-R
Register location(s)	M8956 (MBD1)

Bit	Name	Description
7	Attention	Reflects the status of the Massbus attention line.
6	Transfer	Reflects the status of the Massbus transfer line.
5	Demand	Reflects the status of the Massbus demand line.
4	Initialize	Reflects the status of the Massbus initialize line.
3	MASS FAIL	Reflects the status of the Massbus fail line. When HI indicates a controller power failure or no Massbus connected to this port.
2	Left	Reflects the status of the write microcontroller LEFT line.
1	Minus 5 V ok	When true, indicates that the —5 V power supply for the Massbus receivers is operating.
0	On-line	When true, indicates that the selected Massbus port is on-line to the host computer. This is the state of the on-line switch located on the backplane.

DBUS CONTROL

Register name	DBUSCTL
Microbus address	300-W
Register location(s)	M8956 (MBD1)

Bit	Name	Description
7	Massbus write enable	When true, enables the Massbus port's data bus (D17:0, P) and SCLK transmitters. Set on subsystem READ DATA or EXSNS commands.*

* Bits 7 and 1 cannot be set at the same time!

Bit	Name	Description
6	Write clock	Asserts the Massbus WCLK line. (Sets the MBD I/O data multiplexers to left mode.)
5	Occupied	Asserts the Massbus OCC line.
4	End of block	Asserts the Massbus EBL line.
3	Exception	Asserts the Massbus EXC line.
2	Run	Asserts the Massbus RUN line.
1	Read enable	Enables the selected Massbus port's read data path and SCLK transmitters. Set on subsystem write data commands.*
0	Sync clock	Asserts the Massbus SCLK line.

DBUS STATUS

Register name	DBUSSTA
Microbus address	300-R
Register location(s)	M8956 (MBD1)

Bit	Name	Description
7	Massbus write enable	When true, indicates that the selected Massbus port's data bus (D17:0, P) and SCLK transmitters are enabled.
6	Write clock	Reflects the status of the Massbus WCLK line.
5	Occupied	Reflects the status of the Massbus OCC line.
4	End of block	Reflects the status of the Massbus EBL line.
3	Exception	Reflects the status of the Massbus EXC line.
2	Run	Reflects the status of the Massbus RUN line.
1	Sync clock out	Reflects the status of DBUSCTL bit 0 or the status of the write microcontroller SCLK line.
0	Sync clock	Reflects the status of the Massbus SCLK line.

* Bits 7 and 1 cannot be set at the same time!

KEYBOARD/DISPLAY SELECT REGISTER*

Register name	KEYBRD
Microbus address	310-W
Register location(s)	Maintenance panel

Bit	Name	Description
Keyboard		
7	—	Not used.
6:0	Key select 6:0	This 7-bit field selects 1 of 20 keys to be examined by the microcomputer. A low on the microbus SID line indicates a depressed key. The field is broken into two areas, column select (6:2) and row select (1:0). A zero in any column bit position selects one of five 4-key columns. The 2-bit row select area is decoded to select one of four 5-key rows. The following table shows the codes used to select an individual key.

Value	Key Selected
074	EXE (Execute)
075	REP (Repeat)
076	STA (Start)
077	CON (Continue)
134	7
135	4
136	1
137	0
154	CLR (Clear)
155	5
156	2
157	NXT (Next)
164	ENA (Enable)
165	6
166	3
167	DEP (Deposit)
170	INS (Instruction)
171	HDW (Hardware)
172	PAR (Parameter)
173	MEM (Memory)

* This register cannot be written into via Massbus registers 20 and 21.

Bit	Name	Description
Display		
7:4	—	Not used.
3:0	Character select	This 4-bit field selects 1 of 16 characters to be displayed in the 7-part LEDs. This register selects the character first; then microbus addresses 312 — 317 select one of six LEDs to display that character. The following table shows the codes used to select a character.

Value	Character Selected
00	0
01	1
02	2
03	3
04	4
05	5
06	6
07	7
10	8
11	— (minus)
12	E
13	H
14	I
15	L
16	P
17	(blank)

LOAD LED A*

Register name	LDLEDA
Microbus address	312-W
Register location(s)	Maintenance panel

Bit	Name	Description
7:0	—	Not used.
		Any write to this register causes LED A (the most significant digit) to display the character previously selected in the KEYBRD register (address 310-W).

* This register cannot be written into via Massbus registers 20 and 21.

LOAD LED B*

Register name	LDLEDB
Microbus address	313-W
Register location(s)	Maintenance panel

Bit	Name	Description
7:0	—	Not used.
		Any write to this register causes LED B to display the character previously selected in the KEYBRD register (address 310-W).

LOAD LED C*

Register name	LDLEDC
Microbus address	314-W
Register location(s)	Maintenance panel

Bit	Name	Description
7:0	—	Not used.
		Any write to this register causes LED C to display the character previously selected in the KEYBRD register (address 310-W).

LOAD LED D*

Register name	LDLEDD
Microbus address	315-W
Register location(s)	Maintenance panel

Bit	Name	Description
7:0	—	Not used.
		Any write to this register causes LED D to display the character previously selected in the KEYBRD register (address 310-W).

* This register cannot be written into via Massbus registers 20 and 21.

LOAD LED E*

Register name	LDLEDE
Microbus address	316-W
Register location(s)	Maintenance panel

Bit	Name	Description
7:0	—	Not used. Any write to this register causes LED E to display the character previously selected in the KEYBRD register (address 310-W).

LOAD LED F*

Register name	LDLEDF
Microbus address	317-W
Register location(s)	Maintenance panel

Bit	Name	Description
7:0	—	Not used. Any write to this register causes LED F (the least significant digit) to display the character previously selected in the KEYBRD register (address 310-W).

BYTE ASSEMBLY LOGIC DATA CONTROL REGISTER

Register name	DATACTL
Microbus address	320-W
Register location(s)	M8959 (WMC1)

Bit	Name	Description
7	Pad/CRC	If 0, GCR CRC data group byte 1 will be CRC character. If 1, GCR CRC data group byte 1 will be pad character (all zeros).

* This register cannot be written into via Massbus registers 20 and 21.

Bit	Name		
		Bit Value	Format
6:4	Format 2:0	0	11 normal
		1	15 normal
		2	10 compatible
		3	10 core dump
		4	10 high-density compatible
		5	Image (Skip count is ignored.)
		6	10 high-density dump
		7	Illegal (Ecode —3)
		Bit Value	Skip Count
3:0	SKIP COUNT 3:0	00	No skip
		01	Skip 1 byte
		02	Skip 2 bytes
		03	Skip 3 bytes
		04	Skip 4 bytes
		05	Skip 5 bytes
		06	Skip 6 bytes
		07	Skip 7 bytes
		10	Skip 8 bytes
		11—17	Illegal (Ecode —1)

WRITE MICROCONTROLLER STATUS

Register name	WMCSTA
Microbus address	320-R
Register location(s)	M8959 (WMC1)

Bit	Name	Description
7	XMC not done	If 0, translator microcontroller is done writing postamble and waiting to be serviced. If 1, translator is busy.
6	WMC not done	If 0, write microcontroller is done writing PE data or GCR data groups. (Used in diagnostic single-step mode; normally appears for 110 ns.) If 1, write microcontroller busy or idle.
5	Transfer	Shifts one nibble on WMC DR bus into translator.
4	ECC to WMC DR bus	Gates HI or LO nibble of ECC character to WMC DR bus.

Bit	Name	Description
3	CRC to WMC DR bus	Gates HI or LO nibble of CRC character to WMC DR bus.
2	ACRC to WMC DR bus	Gates HI or LO nibble of ACRC character to WMC DR bus.
1	Residual to WMC DR bus	Gates HI or LO nibble of residual character to WMC DR bus.
0	XMC WCLK	Reflects true state of translator write clock pulse (110 ns at normal speed).

RESIDUAL CHARACTER

Register name	RESCHR
Microbus address	321-W
Register location(s)	M8959 (WMC1)

Bit	Name	Description
7:5	Modulo-7 count	This binary count represents the number of data bytes to be written in the GCR residual data group. (Legal counts are 0 — 6).
4:0	Modulo-32 count	This binary count represents the low-order five bits of the byte count, minus 1.

TAPE UNIT PORT 0/1 SELECT

Register name	TUSEL 0
Microbus address	321-R
Register location(s)	M8955 (port 0/1 slot 08)

Bit	Name	Description
7	Single tape unit port	If 0, both TUP modules are installed in backplane slots 08/09. If 1, only one TUP module is installed in backplane slot 08.
6	Byte count terminal count	If 0, write microcontroller count terminal count (BCTC) is asserted. If 1, BCTC is unasserted.
5	TU port 0 read path enable	When true, indicates that the read path for tape unit port 0 is enabled.
4	TU port 1 read path enable	When true, indicates that the read path for tape unit port 1 is enabled.

Bit	Name	Description
3	TU port 0 write path enable	When true, indicates that the write path for tape unit port 0 is enabled.
2	TU port 1 write path enable	When true, indicates that the write path for tape unit port 1 is enabled.
1, 0	Tape unit select 1:0	A 2-bit binary field that reflects the tape unit port currently selected by the MBSEL register (340-W bits 1:0).

TRACK ENABLE

Register name	TRKENA
Microbus address	322-W
Register location(s)	M8958 (XMCH)

NOTE
This register enables or disables the nine write buffers independently and must be loaded in two sequential write operations.

Bit	Name	Description
7:1	Track enable 7:1	When true, TE 7:1 enables the write buffers for binary tape tracks 7:1.
0	Track enable P or Track enable 0	Every write to this register loads 7:0 and causes the previous contents of 0 to load into P.

TAPE UNIT PORT 2/3 SELECT

Register name	TUSEL 1
Microbus address	322-R
Register location(s)	M8955 (port 2/3, slot 09)

Bit	Name	Description
7, 6	—	Not used.
5	TU port 2 read path enable	When true, indicates that the read path for tape unit port 2 is enabled.
4	TU port 3 read path enable	When true, indicates that the read path for tape unit port 3 is enabled.
3	TU port 2 write path enable	When true, indicates that the write path for tape unit port 2 is enabled.

Bit	Name	Description
2	TU port 3 write path enable	When true, indicates that the write path for tape unit port 3 is enabled.
1, 0	Tape unit select 1:0	A 2-bit binary field that reflects the tape unit port currently selected by the MBSEL register (340-W bits 1:0).

NOTE
This register exists only when two TU port modules are present in the system (i.e., TUSEL0, bit 7 = 0.)

WMC CONTROL

Register name	WMCCTL
Microbus address	323-W
Register location(s)	M8959 (WMC 1)

Bit	Name	Description
7	Enable write microcontroller	When true, causes the write microcontroller to examine the rest of the bits in this register and start transferring data to or from the Massbus.
6	Enable translator microcontroller	When true, causes the translator microcontroller to examine the rest of the bits in this register and start sending data to the TU port.
5	Ones	When true, informs the translator microcontroller that an ID burst or tape mark will be written.
4	GCR/PE	When true, informs the write translator and read path microcontrollers that a GCR mode write or read operation is in progress. When false, a PE mode write or read operation is in progress.
3	Write	When true, informs the write and read path microcontrollers that a write operation is in progress. When false, a read operation is in progress.
2	Reverse	When true, informs the write and read path microcontrollers that a read reverse operation is in progress. When false, (and bit 3 = 0) a read forward operation is in progress.

Bit	Name	Description
1	Diagnose	When true, informs the write microcontroller that the microcomputer wants to test the WMC byte/pad/Ecode counters.
0	Restart	When true, resets the write and translator microcontroller program counters to zero and clears any parity error conditions in those microcontrollers.

WRITE DATA

Register name	WRTDAT
Microbus address	323-R
Register location(s)	M8959 (WMC 1)

Bit	Name	Description
7:0	DR D7:0 H	Looks at write or read data on the intermediate DRD bus.

BYTE COUNTER HI/LO*

Register name	BYTCNT
Microbus address	324-R/W
Register location(s)	M8959 (WMC 1)

Bit	Name	Description
7:0	Byte count 7:0, 15:8	Write micro/byte assembly byte counter word. First write sets 7:0. Second write sets 15:8. First read accesses 7:0. Second read accesses 15:8.

PAD COUNTER HI/LO*

Register name	PADCNT
Microbus address	325-R/W
Register location(s)	M8959 (WMC 1)

* A double read/write operation must be performed to this register with CNTCTL 5, 4 = 3. This register cannot be written to via Massbus register 21.

Bit	Name	Description
7:0	Pad count 7:0, 15:8	Write micro/byte assembly pad character counter word. First write sets 7:0. Second write sets 15:8. First read accesses 7:0. Second read accesses 15:8.

ERROR CODE (ECODE) COUNTER HI/LO*

Register name	ERRCNT
Microbus address	326-R/W
Register location(s)	M8959 (WMC 1)

Bit	Name	Description
7:0	Error code 7:0, 15:8	Write micro/byte assembly error code count. First write sets 7:0. Second write sets 15:8. First read accesses 7:0. Second read accesses 15:8. Normally set to zeros at the start of an operation. Ecode counter values when read at the completion of an operation:

Bit Value	Error Indication
000	Operation completed successfully.
377	Read skip count >4 in 0 core dump, >3 in 10 compatible, or >8 in 10 high-density compatible.
376	Read skip count >1 in 11 normal or 15 normal.
375	Format code >6.
374	WMC self-test diagnostic error.
373	Read overrun or write fault.

* A double read/write operation must be performed to this register with CNTCTL 5, 4 = 3. This register cannot be written to via Massbus register 21.

WMC BYTE, PAD, ECODE COUNTER CONTROL

Register name	CNTCTL
Microbus address	327-W
Register location(s)	M8959 (WMC 1)

Bit	Name	Description
7, 6	Counter select 1, 0	Selects one of three counters to read from or write into.

Bit Value	Selected Counter
0	Byte counter HI/LO
1	Pad counter HI/LO
2	Error code counter HI/LO
3	Illegal

Bit	Name	Description
5, 4	Read/write 1, 0	Selects read/write sequence.

Bit Value	Sequence
0—2	Not used.
3	Read/write least significant byte first, then most significant byte.

Bit	Name	Description
3:1	Mode 2:0	Selects counting mode.

Bit Value	Mode Selected
0	Terminal count stops counter.
1—7	Not used.

Bit	Name	Description
0	Binary/BCD	If 0, counters operate in 16-bit binary mode If 1, counters operate in 4-decade BCD mode. (Not used.)

DIAGNOSTIC DATA REGISTER A*

Register name	DDR A
Microbus address	330-R/W
Register location(s)	M8959 (WMC 2)

Bit	Name	Description
7:0	Massbus data 7/5 : 0/16	This register reads from and writes to a portion of the 18-bit-wide MBD bus between the M8956 and M8959 modules as follows. **Read** If WMC left bit = 1, Massbus D7:0 transfers to microbus D7:0. If WMC left bit = 0, Massbus D5:0, 17, 16 transfers to microbus D7:0. **Write** If WMC left bit = 0 only, microbus D7:0 transfers to Massbus D5:0, 17, 16.

DIAGNOSTIC DATA REGISTER B*

Register name	DDR B
Microbus address	331-R/W
Register location(s)	M8959 (WMC 2)

Bit	Name	Description
7:0	Massbus data 15/13 : 8/6	This register reads from and writes to a portion of the 18-bit-wide MBD bus between the M8956 and M8959 modules as follows. **Read** If WMC left bit = 1, Massbus D15:8 transfers to microbus D7:0. If WMC left bit = 1, Massbus D13:6 transfers to microbus D7:0. **Write** If WMC left bit = 0 only, microbus D7:0 transfers to Massbus 13:6.

* This register cannot be written to via Massbus register 21.

DIAGNOSTIC DATA REGISTER C*

Register name	DDR C
Microbus address	332-W
Register location(s)	M8959 (WMC 2)

Bit	Name	Description
		This register writes to a portion of the 18-bit-wide MBD bus between the M8956 and M8959 modules.
7:3	—	Not used.
2:0	Massbus data P, 17/15, 16/14	If WMC left bit = 0 only, microbus 2:0 transfers to MBD P, 17/15, and 16/14 respectively.

WRITE MICROCONTROLLER ERRORS

Register name	WMCERR
Microbus address	332-R
Register location(s)	M8959 (WMC 2)

Bit	Name	Description
		This register reads from a portion of the 18-bit-wide MBD bus between the M8956 and M8959 modules. It also reads certain WMC error status bits.
7	DR no write clock	Host Massbus controller failed to respond to SCLK in time to transfer next data byte. (Currently not used.)
6	DR Massbus parity error	Bad (even) Massbus parity was received at TM78 data input port.
5	Error	Write microcontroller diagnostic overall error status bit.
4	WMC ROM parity error	A write microcontroller ROM parity error was detected or the WMC internal microdiagnostic stopped at an error halt.
3	DR read parity error	A parity error was detected on a read data byte sent from the CRC/ACRC to WMC module over the CRC bus.

* This register cannot be written to via Massbus register 21.

Bit	Name	Description
2:0	Massbus data P, 17/15, 16/14	If WMC left bit = 1, Massbus D P, 17, 16, transfers to microbus D2:0. If WMC left bit = 0, Massbus D P, 15, 14 transfers to microbus D2:0.

DIAGNOSTIC DATA REGISTER CONTROL

Register name	DDRCTL
Microbus address	333-W
Register location(s)	M8959 (WMC 2)

Bit	Name	Description
7	Mode set flag	Indicates that mode bits 6, 5, 2 are to be written into. Always set to a 1.
6, 5	Group A mode select	Selects the operating mode for DDR A 7:0 and DDR C 7:4. Always set to 0 (mode 0).
4	DDR A in/out	Determines the direction of data transfer for DDR A. If 0, data from microbus D7:0 transfers to MBD 7/5 : 0/16. If 1, data from MBD 7/5 : 0/16 transfers to microbus D7:0.
3	DDR C (7:4) in/out	Determines the direction of data transfer for DDR C 7:4. Always set to a 1, indicating WMC error status to microbus D7:4
2	Group B mode select	Selects the operating mode for DDR B and DDR C 3:0. Always set to a 0 (mode 0).
1	DDR B in/out	Determines the direction of data transfer for DDR B. If 0, data from microbus D7:0 transfers to MBD 15/13 : 8/6. If 1, data from MBD 15/13 : 8/6 transfers to microbus D7:0.
0	DDR C (3:0) in/out	Determines the direction of data transfer for DDR C 3:0. If 0, data from microbus D2:0 transfers to MBD P, 17/15, 16/14. If 1, data and error status from DR RD PE; MBD P, 17/15, 16/14 transfers to microbus D3:0.

MASSBUS/TAPE UNIT PORT SELECT REGISTER

Register name	MBSEL
Microbus address	340-W
Register location(s)	M8960 (8MC2, 4)

Bit	Name	Description
7	Massbus port select	If 0, Massbus port A selected. If 1, Massbus port B selected.
6:2	—	Not used.
1,0	Tape unit select 1:0	This 2-bit binary field selects one of four possible TU ports and/or tape transports.

INTERRUPT STATUS REGISTER

Register name	INTSTA
Microbus address	340-R
Register location(s)	M8960 (8 MC1)

Bit	Name	Description
7	Massbus port select	Reflects the Massbus port currently selected by MBSEL (340-W, bit 7). If 0, port A selected. If 1, port B selected.
6	Microcomputer ROM parity error	When set, indicates that the microcomputer ROM has experienced a parity error. (May indicate a program branch to an unused address.)
5	Power ok	When set, indicates that the TM78 power supply has not signaled an AC LO condition.
4	Write data register parity	Reflects the state of the DR BYTE PAR H line. Serves as the parity bit for the WRTDAT register (323-R), or as a PE write operation done flag.
3	Massbus A out	When set, indicates that the logic for Massbus Port A is not present in the TM78.
2	Massbus B out	When set, indicates that the logic for Massbus port B is not present in the TM78.

Bit	Name	Description
1	PE write parity error	When set, indicates that a vertical parity error has been detected at the output of the translator during a phase-encoded write operation.
0	Translator ROM parity error	When set, indicates that the translator microcontroller ROM has experienced a parity error or has branched to an error halt during its internal diagnostic routines.

CLOCK CONTROL REGISTER

Register name	CLKCTL
Microbus address	360-W
Register location(s)	M8960 (8 MC4)

Bit	Name	Description
7	—	Not used.
6	Clock step	When set and reset, this bit produces a single clock pulse for the read or write path when the clock select field for that path is 5.
5:3	Read clock select 2:0	Bit Value / Function 0 —10% clock 1 +10% clock 2 Normal clock 3 —30% clock 4 External clock 5 Single step 6 Not used 7 Not used
2:0	Write clock select 2:0	2:0 / Function 0 Normal clock (write) 1 +20% clock 2 +10% clock (GCR read) 3 —20% clock (PE read) 4 External clock 5 Single step 6 Not used 7 Not used

APPENDIX F
INTERFACE WIRE LIST AND TRANSPORT INTERCONNECTIONS

Table F-1 Major Plug/Jack Reference Designations

Item	Location	Description/Purpose
MIA J1	On MIA PCBA	Connects to write PCBA J25 by means of cable
MIA J2	On MIA PCBA	Connector for TU bus A
MIA J3	On MIA PCBA	Connector for TU bus B
J1—J3	Interconnect D1 PCBA	Connectors for P1—P3 on MIA PCBA
J4	On read PCBA	For ribbon cable to preamp PCBA on base assembly
J5	On interconnect D1 PCBA	Main connector for read PCBA P5
J6	On write PCBA	For ribbon cable to preamp PCBA
J7	On interconnect D1 PCBA	Main connector for write PCBA P7
J8	On interconnect D1 PCBA	Main connector for control M2 PCBA
J10	On control M2 PCBA	For ribbon cable to manual controls on transport front panel
J11	On interconnect D1 PCBA	Main connector for capstan/regulator PCBA
J12	On interconnect D1 PCBA	Main connector for reel servo PCBA
J13	On reel servo PCBA	For +36 V to unregulated dc supply
J14	On reel servo PCBA	For supply reel motor drive power (+)
J15	On reel servo PCBA	For supply reel motor drive power ( )
J16	On reel servo PCBA	For 36 V to unregulated dc supply
J17	On reel servo PCBA	For +36 V from unregulated dc supply
J18	On reel servo PCBA	For take-up reel drive power (+)
J19	On reel servo PCBA	For take-up reel drive power ( )
J20	On reel servo PCBA	For 36 V from unregulated dc supply

Table F-1 Major Plug/Jack Reference Designations (Cont)

Item	Location	Description/Purpose
J21	Accommodates P21 of cable from J24 in card cage to interconnect F1 PCBA on base assembly	For connecting card cage circuits to base assembly sensor and control circuits
J22	On GCR/PE preamp 1 PCBA attached to rear of base assembly	Read head input to preamp PCBA
J23	On GCR/PE preamp 1 PCBA	Output from GCR/PE preamp 1 to write heads
J24	On interconnect D1 PCBA	For cable to interconnect F1 PCBA
J25	On write PCBA	Connects to MIA PCBA

Table F-2 TU Bus Interface Signals

Pin	TU Bus A MIA J2	TU Bus B MIA J3
1		
2	GND	GND
3	WCS0	AMTIE0
4	GND	GND
5	WCS1	AMTIE1
6	GND	GND
7	WCS2	AMTIE2
8	GND	GND
9	WCS3	AMTIE3
10	GND	GND
11	WCS4	AMTIE4
12	GND	GND
13	WCS5	AMTIE5
14	GND	GND
15	WCS6	AMTIE6
16	GND	GND
17	WCS7	AMTIE7
18	GND	GND
19	WCSP	AMTIEP
20	GND	GND
21	CMD	RDP
22	GND	GND
23	WDS	RD7
24	GND	GND
25	STAT	RD6

Table F-2 TU Bus Interface Signals (Cont)

Pin	TU Bus A MIA J2	TU Bus B MIA J3
26	GND	GND
27	CMD PE	RD5
28	GND	GND
29	TACH	RD4
30	GND	GND
31		RD3
32	GND	GND
33		RD2
34	GND	GND
35		RD1
36	GND	GND
37		RD0
38	GND	GND
39		PRESENT H
40	GND	GND

Table F-3 AMTIE Cable Signals

MIA J1/ Write J25 Pin	Signal
1	IAMTIE7
2	IAMTIE5
3	IAMTIE4
4	IWINH
5	NLTH
6	Ground
7	Ground
8	Ground
9	WRT BIN H
10	
11	IARA ERR
12	IAMTIE3
13	IAMTIE6
14	WRT STAT H
15	STDTH
16	MOTION H
17	IAMTIEP
18	IAMTIE0
19	IAMTIE1
20	IAMTIE2

Table F-4 MIA PCBA to Transport Interface

Connector J1/P1		Connector J2/P2		Connector J3/P3		Connector J3/P3	
Pin	Signal	Pin	Signal	Pin	Signal	Pin	Signal
1	NPORST	1	IWD7	1	IRD7	1	IRD7
2	IEOT	2	Ground	2	Ground	2	Ground
3	ILDP	3	Ground	3	Ground	3	Ground
4	IFPT	4	Ground	4	Ground	4	Ground
5	IRWD	5	Ground	5	Ground	5	Ground
6	IONL	6	Ground	6	Ground	6	Ground
7	IRWU	7	Ground	7	Ground	7	Ground
8	ISWS	8	Ground	8	Ground	8	Ground
9	ISLT0	9	Ground	9	Ground	9	Ground
10	IRWC	10	Ground	10	Ground	10	Ground
11	IDDI	11	Ground	11	+5	11	+5
12	ISRC	12	Ground	12	+5	12	+5
13	IDDS	13	Ground	13	+5	13	+5
14	IFSC	14	Ground	14	Ground	14	Ground
15	Not used	15	Ground	15	Ground	15	Ground
16	Not used	16	Ground	16	Ground	16	Ground
17	Not used	17	Ground	17	Ground	17	Ground
18	Not used	18	IWD6	18	IRD6	18	IRD6
19	IRDY	19	IWD5	19	NTSTR	19	NTSTR
20	Not used	20	IWD4	20	IRD5	20	IRD5
21	Not used	21	IWD3	21	IRD4	21	IRD4
22	Not used	22	IWD222	22	Not used	22	Not used
23	Ground	23	IWD1	23	Not used	23	Not used
24	Ground	24	IWD0	24	Not used	24	Not used
25	Ground	25	IWDP	25	Not used	25	Not used
26	Ground	26	ITACH	26	IRD3	26	IRD3
27	Ground	27	Not used	27	IRD2	27	IRD2
28	Ground	28	Not used	28	+5	28	+5
29	Ground	29	Not used	29	+5	29	+5
30	Ground	30	Not used	30	+5	30	+5
31	Ground	31	ISLT2	31	IRD1	31	IRD1
32	Ground	32	Not used	32	IRD0	32	IRD0
33	Ground	33	ISLT1	33	Not used	33	Not used
34	Ground	34	IWDS	34	IRDP	34	IRDP

Table F-5 Power Distribution on Interconnect D1 PCBA

L to evel Signal	From	Termination To	Termination To	Termination To
+12 Vdc	A2TB1-6	J11-15.55		
+24 Vdc	A2TB1-7	J11-16.56	P21-22.23.24	J24-27.28.29
24 Vdc	A2TB1-8	J11-20.60		
+36 V(C)	A2TB1-1	J11-1.2.41.42		
DC COM 2	A2TB1-4	J11-9.10. 11.49.50.51		
36 V(C)	A2TB1-5	J11-12.13.52.53		
DC COM 1	A2TB1-9	J11-26.27.66.67		
8.5 Vac	A2TB1-10	J11-40		
CM(+)	A2TB1-3	J11-6.7.8. 46.47.48		
CM( )	A2TB1-2	J11-3.4.5. 43.44.45		
0 V(L)	J11-26. 27.66.77	P21-11.12.14 J24-37.39.40 J8-22.58	J5-3.4.39.40 J12-4.5.21.22	J7-15.16.17. 51.52.53
0 V(1)	J11-26. 27.66.77	J1-23 to 34 J5-26 to 36.38	J2-2 to 17 J7-1 to 10	J3-2 to 10. 14 to 17 J8-38
		J101-2 to 14.16.17	J102-1 to 18	J103-A.B.C.D. J to V
		J201-20 to 34 J204-2 to 17	J202-2 to 17	J203-2 to 10. 14 to 17
+5 V(L)	J11-22.23. 62.63	P21-2.4 J24-47.49 J5-8.9.44.45 J12-1.2.18.19	J7-18.19. 54.55 J101-S	J3-11.12.13. 28.29.30 J8-21.57 W1-1
+5 V(1)	J203-11.12. 13.28.29.30	J201-18	W2-1	
+5 V(T)	W1-2.W2-2	J204-1.18	J7-21.57	J8-37
+15 V	J11-12.61	J5-5.41 J12-6.23	J7-22.58 P21-39	J8-56 J24-12
15 V	J11-14.54	J5-6.42 J12-7.24	J7-23.59 P21-33.36	J8-55 J24-15.18

Table F-6 Deck Interface Ribbon to Interconnect D1 PCBA

Level Signal	From	To	Termination To	To	To
TACH	P21-44	J24-7	J8-53		
TACH COM	P21-46	J24-5	J8-54		
SPARE 2	P21-49	J24-2		J11-68	
S.POS	P21-37	J24-14			J12-20
T.POS	P21-21	J24-30			J12-3
0 V(R)	P21-19	J24-32			J12-34
SPARE	P21-1	J24-50			
S.LIMIT	P21-6	J24-45	J8-1		J12-30
T.LIMIT	P21-10	J24-41	J8-2		J12-31
CR N.O.	P21-8	J24-43		J11-80	
NEOT	P21-50	J24-1	J8-50		
TIP	P21-45	J24-6	J8-49		
NBOT	P21-42	J24-9	J8-52		
PKSN	P21-30	J24-21	J8-42		
NSMR	P21-3	J24-48	J8-8		
CART N.O.	P21-43	J24-8		J11-79	
CO	P21-41	J24-10	J8-10		
CC	P21-40	J24-11	J8-11		
CART M+	P21-31,34	J24-20,17		J11-30,70	
CART M	P21-25,26	J24-25,26		J11-31,71	
C.SOLRET	P21-17,20	J24-31,34		J11-34,74	
SPARE 1	P21-5,7	J24-44,46		J11-36,76	
VAC	P21-48	J24-3	J8-3		
VAC.SOL.RET.	P21-15,18	J24-33,36		J11-32,72	
WP2	P21-35,38	J24-13,16			
WP1 N.O.*	P21-29,32	J24-19,22		J11-69	J7-60,24
W.P.SOL.RET.	P21-27,28	J24-23,24		J11-33,73	
P.SOL.RET.	P21-13,16	J24-33,38		J11-35,75	
ABPNO	P21-9	J24-42	J8-4		
TOR	P21-47	J24-4	J8-9		

* Also WRT CUR

Table F-7 Internal Control Signals on Interconnect D1 PCBA

Level Signal	From	To	Termination To	To	To	To
NPE	J8-25	J5-10	J7-61			J103-7
NWRT	J8-26	J5-47	J7-25			
MOTION	J8-28	J5-48	J7-26			
FPT	J8-60			J11-19		
BOT(INT)	J8-48	J5-12				
NMOT	J8-51			J11-24		
SRF	J8-29				J12-17	
SRR	J8-30				J12-16	
TRF	J8-31				J12-15	
TRR	J8-32				J12-14	
MRL	J8-5				J12-28	
REWR				J11-78	J12-13	
REV	J8-66			J11-18	J12-33	
NDRV	J8-65			J11-64	J12-27	
N>80%	J8-67			J11-25		
NTAP2	J8-62			J11-57		
NTEN	J8-63			J11-17		
NINTLK	J8-69			J11-65	J12-8.25	
NRWR	J8-33			J11-38		
NRSAE	J8-34			J11-58	J12-11	
NCCC	J8-70			J11-29		
NCOC	J8-71			J11-28		
NXFR	J8-72			J11-39		
NPOL	J8-64			J11-59		
(SPARE A)						
NPORST	J11-37.77	J1-1 J12-9.26	J7-11 J201-1	J8-61		
PNU RET	J8-68	A2TB1-11				

Table F-8 Interface Control Signals on Interconnect D1 PCBA

Level Signal	From	To	Termination To	To
ISFC	J101-C	J1-14	J201-14	J8-20
IDDS	J101-D	J1-13	J201-13	J8-19
ISRC	J101-E	J1-12	J201-12	J8-18
IDDI	J101-F	J1-11	J201-11	J8-17
IRWC	J101-H	J1-10	J201-10	J8-16
ISWS	J101-K	J1-8	J201-8	J8-14
IRWU	J101-L	J1-7	J201-7	J8-13
IONL	J101-M	J1-6	J201-6	J8-12
IRWD	J101-N	J1-5	J201-5	J8-47
IFPT	J101-P	J1-4	J201-4	J8-46
ILDP	J101-R	J1-3	J201-3	J8-45
IRDY	J101-T	J1-19	J201-19	J8-7
IEOT	J101-U	J1-2	J201-2	J8-44
ISLT0	J101-J	J1-9	J201-9	J8-14
ISLT1	J102-B	J2-33	J202-33	J8-15
ISLT2	J102-D	J2-31	J202-31	J8-16
ITACH	J102-K	J2-26	J202-26	J8-6

Table F-9 Interface Read Signals on Interconnect D1 PCBA

Level Signal	From	To	Termination To	To
IRDP	J103-1	J3-34	J5-72	J203-34
IRD0	J103-3	J3-32	J5-70	J203-32
IRD1	J103-4	J3-31	J5-69	J203-31
IRD2	J103-8	J3-27	J5-68	J203-27
IRD3	J103-9	J3-26	J5-67	J203-26
IRD4	J103-14	J3-21	J5-62	J203-21
IRD5	J103-15	J3-20	J5-61	J203-20
IRD6	J103-17	J3-18	J5-59	J203-18
IRD7	J103-18	J3-1	J5-58	J203-1

Table F-10 Interface Write Signals on Interconnect D1 PCBA

Level Signal	From	To	Termination To	To
IWDS	J102-A	J2-34	J7-47	J202-34
IWDP	J102-L	J2-25	J7-45	J202-25
IWD0	J102-M	J2-24	J7-44	J202-24
IWD1	J102-N	J2-23	J7-43	J202-23
IWD2	J102-P	J2-22	J7-42	J202-22
IWD4	J102-R	J2-21	J7-41	J202-21
IWD3	J102-S	J2-20	J7-40	J202-20
IWD5	J102-T	J2-19	J7-39	J202-19
IWD6	J102-U	J2-18	J7-38	J202-18
IWD7	J102-V	J2-1	J7-37	J201-1

Index

Digital Equipment Corporation • Shrewsbury, MA 01545