60403900

CONTROL DATA CORPORATION

CDC® MAGNETIC TAPE SUBSYSTEM

7021-1/2 MAGNETIC TAPE CONTROLLER
7021-21/22 MAGNETIC TAPE CONTROLLER
7152-1 DISK/TAPE CONTROLLER
7622-1/2 MAGNETIC TAPE CONTROLLER
66X-2/3/4 MAGNETIC TAPE TRANSPORT

OPERATION AND PROGRAMMING MANUAL

ALPHABETIC LISTING OF FUNCTION CATEGORIES

60403900

CONTROL DATA CORPORATION

CDC® MAGNETIC TAPE SUBSYSTEM

7021-1/2 MAGNETIC TAPE CONTROLLER
7021-21/22 MAGNETIC TAPE CONTROLLER
7152-1 DISK/TAPE CONTROLLER
7622-1/2 MAGNETIC TAPE CONTROLLER
66X-2/3/4 MAGNETIC TAPE TRANSPORT

OPERATION AND PROGRAMMING MANUAL

New features, as well as changes, deletions, and additions to information in this manual, are indicated by bars in the margins or by a dot near the page number if the entire page is affected. A bar by the page number indicates pagination rather than content has changes.

REVISION RECORD

REVISION	DESCRIPTION
A	Manual released.
(10-26-73)	
B	This revision includes changes related to FCO 34436, pages 2-20, 3-2, 3-7, A-2, A-3, and A-4.
(3-74)	
C	This revision includes miscellaneous additions and corrections.
(4-10-74)	
D	This revision incorporates FCO 35305.
(10-15-74)	
E	This revision incorporates FCO35982. This edition obsoletes all previous editions.
(5-15-75)	
F	This revision includes changes related to Field Change Order 36584. Inside front cover, iv, 2-13,
(9-4-75)	2-24, 2-41, 2-50, 2-51, 2-52, 3-9, 3-14, Index-2, and inside back cover are revised.
G	This revision incorporates Field Change Order 36757. Inside front cover, iv, 2-8, 2-30, 2-36,
(2-6-76)	2-37, 2-38, 3-14, A-2, A-3, A-4, and inside back cover are revised. Pages 2-38.1 and 2-38.2
	are added.
H	This revision incorporates Field Change Order 37093. Pages 2-36 and A-2 are revised.
(4-19-76)	
J	Manual revised; includes Field Change Order 37442. Pages 1-1, 1-2, 2-1, 2-2, 2-22, 3-2, and
(12-13-76)	3-3 are revised.
K	Manual revised; includes Engineering Change Order 38404. Front cover and pages v, 1-1, 1-2,
(9-15-77)	2-1, 5-1 through 5-6, A-1, A-2, A-3, and A-4 are revised. Pages 1-3 and 5-7 are added.
L	Manual revised; includes Field Change Order 38303. Inside front cover, and pages iv, 1-2, 2-15,
(12-7-77)	2-16, 2-22, 2-27, 2-28, 2-50, 3-3, 3-4, 3-5, 4-1, 4-2, 5-1, 5-2, 5-3, A-2, A-4, C-2, and
	inside back cover are revised. Page 2-53 is added.
M	Manual revised; includes Field Change Order 38722. Page A-2 is revised.
(4-14-78)	
N	Manual revised; includes Engineering Change Order 38979. Pages iii, v, 2-2, 2-16, 5-4
(8-29-78)	A-1, C-2, and Index-1 are revised. Pages 2-2.1/2-2.2 and F-1 through F-77 are added.
P	Manual revised; includes Engineering Change Order 39653. Pages v and Index-2 are revised.
(1-5-79)	Page G-1 is added.
R	Manual revised; includes Field Change Order 40024 and 40025. Pages A-2, F-26, F-55, F-60,
(7-9-79	F-62, and F-69 are revised.
S	Manual revised; includes Engineering Change Order 40452, publication change only. Pages F-26,
(8-7-79)	and F-69 are revised.
T	Manual revised; includes Engineering Change Order 40730, publications change only. Page 3-3 is
(12-12-79)	revised.

Publication No.
60403900

REVISION LETTERS I, O, Q, S, X, AND Z ARE NOT USED

©1973, 1974, 1975, 1976, 1977, 1978, 1979, 1980, 1982
by Control Data Corporation
All rights reserved.
Printed in the United States of America

Address comments concerning this manual to:
Control Data Corporation
Publications and Graphics Division
4201 North Lexington Avenue
St. Paul, Minnesota 55112

or use Comment Sheet in the back of this manual.

REVISION RECORD (CONT'D)

REVISION	DESCRIPTION
U	Engineering Change Order 41203. To release microfiche at this revision level. No change to
(05-08-80)	hardcopy.
V	Manual revised; includes Engineering Change Order 43502. Pages 44C1 (1-1) and 44C2 (1-2) are
(04-21-82)	revised. Page 44C3 (1-2.1/1-2.2) is added.

Publication No.
60403900

CONTENTS

INTRODUCTION 1

This manual describes the software capabilities provided by the CONTROL DATA® Magnetic Tape Subsystem (MTS). Recommended programming sequences using MTS software are also provided in this manual.

WARNING

This equipment generates, uses and can radiate radio frequency energy and if not installed and used in accordance with the instructions manual, may cause interference to radio communications. As temporarily permitted by regulation, it has not been tested for compliance with the limits for Class A computing devices pursuant to Subpart J of Part 15 of the FCC Rules which are designed to provide reasonable protection against such interference. Operation of this equipment in a residential area is likely to cause interference in which case the user at his own expense will be required to take whatever measures may be required to correct the interference.

MTS software resides in the MTS processor's core memory. MTS software enables execution of physical tape operations required as the result of logical commands issued from a CDC® CYBER 170/70 Peripheral Processing Unit (PPU). Thus, MTS software serves as a programmable and therefore flexible interface between the PPU and the tape control hardware.

An alphabetic listing of function categories is inside the front cover and a numeric listing is inside the back cover.

MTS HARDWARE CONFIGURATION

The basic MTS hardware 1 x 8 configuration is made up of the following:

- A 7021-1 Magnetic Tape Controller (CDC CYBER 170 Series/Models 172, 173, 174, and 175; CYBER 70/Models 72, 73, and 74; and 6000 Series Computer Systems†) or a 7622-1 Magnetic Tape Controller (CDC CYBER 70/Model 76 Computer System and the CONTROL DATA 7600 Computer Systems ††). Each magnetic tape controller is made up of the following components.

  1. Magnetic tape subsystem processor (memory included)
  2. CDC CYBER 170/70 and 6000 couplers or CDC CYBER 70 and 7600 couplers
  3. Tape Control Unit (TCU)

- One to eight 66x magnetic tape transports. Table 1-1 is a summary of specific 66x transports available.

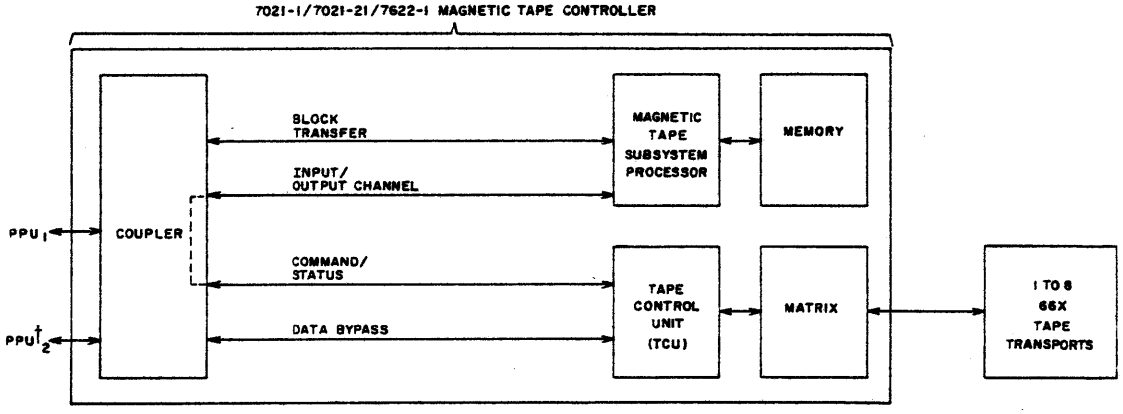
† Hereafter referred to as CDC CYBER 170/70 and 6000.
†† Hereafter referred to as CDC CYBER 76 and 7600.

Tape Transport	Number of Tracks	Recording† Method	Recording Density (cpi)		Tape Speed ips
			NRZI	PE	
667-2	7	NRZI	556/800††	-	100
667-3	7	NRZI	556/800††	-	150
667-4	7	NRZI	556/800††	-	200
669-2	9	NRZI/PE	800	1600	100
669-3	9	NRZI/PE	800	1600	150
669-4	9	NRZI/PE	800	1600	200

†NRZI = Nonreturn to zero indiscrete
PE = Phase-encoded: Tapes used for phase-encoded recording must be certified at 3200 flux reversals per inch.

††200 cpi: read only

The following diagram shows the basic 1 x 8 configuration.



An MTS 2 x 8 configuration is available. The 2 x 8 configuration consists of:

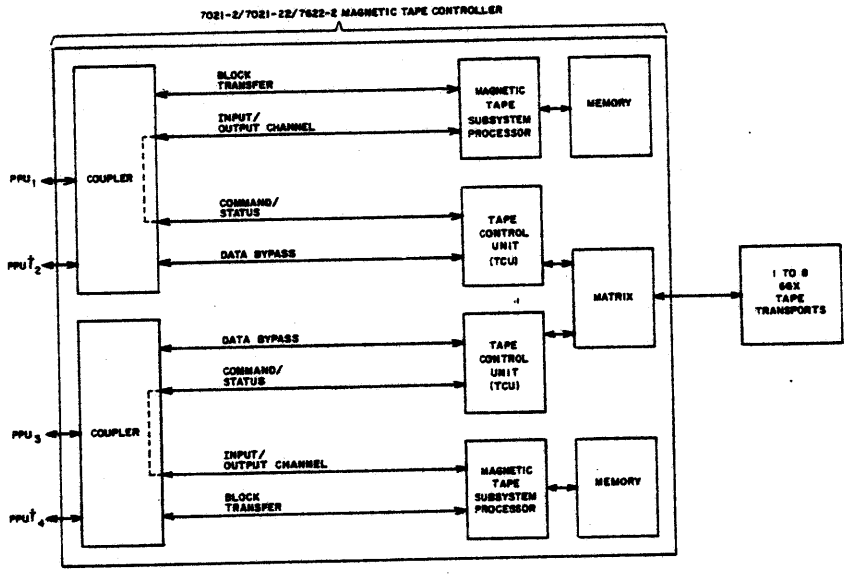
- A 7021-2/22 Magnetic Tape Controller (CDC CYBER 170/70 and 6000) or a 7622-2 Magnetic Tape Controller (CDC CYBER 70 and 7600). Each magnetic tape controller is made up of the following components.

  1. Two magnetic tape subsystem processors
  2. Two CDC CYBER 170/70 and 6000 couplers or two CDC CYBER 70 and 7600 couplers

† A second channel access is present on the 7622-1/2 Magnetic Tape Controller.

3. Two tape control units

One to eight 66x magnetic tape transports. Table 1-1 gives a description of specific 66x units.

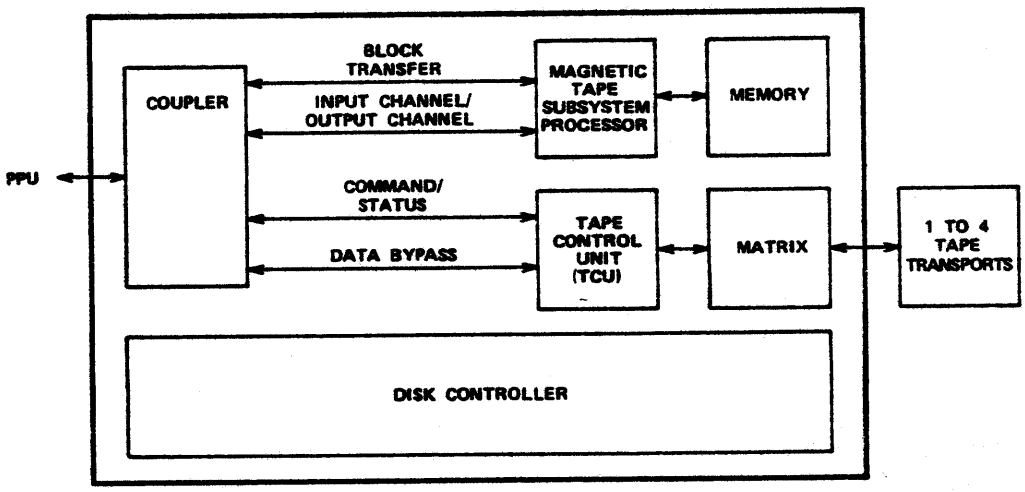


† A second channel access is present on the 7622-1/2 Magnetic Tape Controller.

A 7152-1 disk/tape 1 x 4 configuration is also available. This configuration consists of a disk controller and a tape controller in one cabinet. The tape controller portion consists of:

- A magnetic tape subsystem processor
- A coupler
- A tape control unit

This controller can control up to four 66x tape transports.



MTS FUNCTIONS 2

MTS functions are 12-bit words that are issued by the user from the PPU to initiate execution of software routines residing in MTS processor memory. For some functions, parameter words must also be issued. When executed, these routines generate tape control unit commands that perform specific tape operations.

The issuing and execution of these functions and any associated parameters involve a combination of operations and communication paths between the PPU, coupler, MTS processor, and the tape control unit. These operations and communication paths are different for the CDC CYBER 170/70 and 6000 and CDC CYBER 70 and 7600 systems.

MTS FUNCTION EXECUTION UNDER CDC CYBER 170/70 AND 6000

The issuing and execution of MTS functions using CDC CYBER 170/70 and 6000 systems involves the following operational steps.

1. A 12-bit function is issued from the PPU as the result of the execution of a send function instruction. The 7021 or 7152 controller is wired as equipment number 0 requiring that bits 11 through 9 must be zero.

2. The MTS processor indicates that the function is accepted by sending an inactive signal to the PPU within 500 microseconds after receiving the function. Acceptance of the function can exceed 500 microseconds if the MTS processor is currently executing a previous function. However, this delay in acceptance is limited to 5 seconds for the stop motion and general status functions.

   If the function is not accepted by the MTS processor, the inactive signal is not returned. The channel remains active until deactivated by the PPU. The function is not executed.

   To determine the specific cause of the error, issue the detailed status function (refer to Detailed Status function in this section and Detailed Status in section 3).

3. Parameters, if required by the function, must be sent by the PPU within 1 millisecond after the MTS processor indicates that the function was accepted. The PPU must then send an inactive pulse.

4. A general status function (refer to General Status function in this section and General Status in section 3) must be issued after the parameters are accepted to ensure that no errors in the parameters were detected by the MTS processor.

   If the parameters are valid, the alert bit in general status is not set.

   If an error in the parameters is detected, the alert bit in general status is set. The function is not executed.

   To determine the specific cause of the error, issue the detailed status function and check the error code field in detailed status.

5. The MTS processor translates each function into one or more tape control unit commands. The MTS processor then sends these tape control unit commands to the tape control unit. Refer to the Magnetic Tape Controller Hardware Reference Manual, Publication No. 60404000, for a detailed description of tape control unit commands.

ABNORMAL END OF OPERATION

When it detects abnormal end of operation during some functions, the MTS processor continues tape motion to the next interblock gap or to loadpoint, whichever occurs first. Functions for which this activity occurs include some write recovery functions and the functions listed in the stop motion function description.

For CDC CYBER 170/70 and 6000 only, the MTS processor monitors the PPU for a stop motion function while searching for the interblock gap or loadpoint.

MTS FUNCTION EXECUTION UNDER THE CDC CYBER 70 AND 7600

The issuing and execution of MTS functions using CDC CYBER 70 and 7600 systems involve the following operational steps.

1. The 12-bit function is issued by writing a 12-bit word on the function channel.

2. The MTS processor echoes the function word back to the PPU on the status channel within 100 microseconds after receiving it.

   NOTE

   There are three exceptions: stop motion function, repeat read function, and general status function if requested before end-of-operation occurs.

3. The PPU compares the echoed 12-bit value with the function code sent to determine if an error occurred in transmitting the function.

4. After echoing the function code, the MTS processor indicates that the function word is accepted by clearing the output word flag on the function channel at least 50 microseconds before general status (refer to General Status function in this section and General Status in section 3) is returned.

   If the function is rejected, general status (the alert bit is set) is returned over the status channel before the output word flag on the function channel is cleared. The function is not executed.

To determine the specific cause of error, issue the detailed status function (refer to Detailed Status function in this section and Detailed Status in section 3). Detailed status is returned over the data channel.

5. Parameters, if required by the function, must be sent by the PPU over the data channel within 1 millisecond after the MTS processor indicates the function was accepted.

6. The MTS processor sends an end of record signal to the PPU over the data channel after accepting the last parameter.

   If an error in the parameters is detected by the MTS processor, the alert bit in general status is set and the function is not executed.

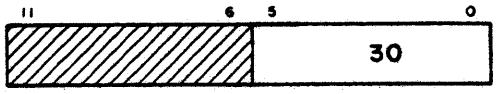
   To determine the specific cause of the error, issue a detailed status function and check the error code field in detailed status.

7. The MTS processor translates each function into one or more tape control unit commands. The MTS processor then sends these tape control unit commands to the tape control unit. Refer to the Magnetic Tape Controller Hardware Reference Manual for a detailed description of tape control unit commands.

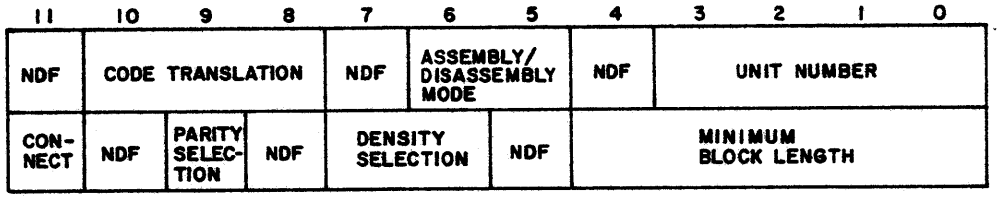
8. At the completion of most functions (refer to inside back cover for functions returning general status) general status is returned on the status channel.

FORMAT UNIT FUNCTION (FUNCTION CODE—30)

The format unit function, which requires two parameter words, is used to specify the characteristics of the data being read or written on the unit specified. The characteristics indicated are used until a format unit statement specifying new parameter values is specified.



Field	Bits	Octal Value	Value Description
	11-6	0	This field is unused.
function code	5-0	30	



Parameter Word 1

Field	Bits	Octal Value	Value Description
(ndf) new definition field	11	0	Contents of the code translation option field (bits 10 through 8) are ignored. The value for this field from the previous format unit function parameter is used. 0 (no code translation) is assumed if no previous value was specified.
		1	The code translation option field (bits 10 through 8) contains a new parameter specification.
code translation	10-8		Refer to appendix C for a detailed description of code translation tables.
		0	No code translation is to take place.
		1	Code translation table 1 is used for this unit's I/O.

Field	Bits	Octal Value	Value Description
		2	**Code translation table 2 is used for this unit's I/O.**
		3	**Code translation table 3 is used for this unit's I/O.**
		7	Refer to appendix C for a description of this option.
(ndf) new definition flag	7	0	Contents of the assembly/disassembly mode field (bits 6 and 5) are ignored. Value from previous format unit function is used. 0 (6-bit mode) is assumed if no previous value was specified.
		1	Assembly/disassembly mode (6-bit mode) field (bits 6 and 5) contains a new parameter specification.
assembly/disassembly mode	6, +5	0	6-bit mode } Refer to appendix B for a detailed description of these modes.
		1	Packed mode
		2	CPU mode
(ndf) new definition flag	4	0	Contents of the unit number field (bits 3 through 0) are ignored. The logical unit presently connected is used. If no unit is connected or the connect field (word 2, bit 11) is 1 an error code is indicated in detailed status (section 3).
		1	Unit number field (bits 3 through 0) contains a new unit number specification.
unit number	3-0	0-7	Indicates the specific logical drive number (0 through 7) to which the format unit function applies. The logical unit presently connected is not relevant.

Parameter Word 2

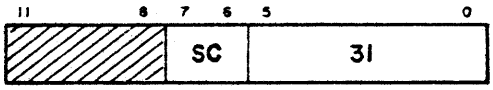
Field	Bits	Octal Value	Value Description
connect	11	0	No connect is performed. The unit specified in the unit number field (word 1, bits 3 through 0), if presently connected, remains connected.
		1	The unit specified by the unit number field (word 1, bits 3 through 0) is reserved and connected.

Field	Bits	Octal Value	Value Description
(ndf) new definition flag	10	0	Contents of the parity selection field (bit 9) is ignored. Value from previous format unit function is used. 0 is assumed for the ps field if no previous value was specified.
		1	Parity selection field (bit 9) contains a new parameter specification.
parity selection	9	0	Odd parity. Data is to be read or written in odd vertical parity.
		1	Even parity. Data is to be read or written in even vertical parity. Code translation from table 3 should be specified for all I/O done in even parity. This causes the MTS processor to generate an external code set as it loads both random access memories. Bit 2^5 is inverted if bit 2^4 is 1 and the code 12_8 is substituted for 00. Code translation tables 1 and 2 cannot be used with even parity. Even parity is intended for 7-track tape.
(ndf) new definition flag	8	0	Contents of the ds field (bits 7 and 6) are ignored. Value from previous format unit function is used. 1 is assumed for the ds field if no previous value was specified.
		1	ds field (bits 7 and 6) contain a new parameter specification.
density selection	7, +6	0	556 cpi (7-track only)
		1	800 cpi (7- or 9-track) } Density is not changed on 9-track units until load-point is reached. Therefore, density selection has no effect when reading 9-track tape.
		2	1600 cpi (9-track only) }
		3	200 cpi (7-track, read only) density status is indicated as 00 (556 cpi) after 3 is specified.
(ndf) new definition flag	5	0	Contents of the mbl field (bits 4 through 0) are ignored. Value from previous format unit function is used. 6 is assumed for the mbl field if no previous value was specified.
		1	mbl field (bits 4 through 0) contains a new parameter specification.
minimum block length	4-0	1-37	Minimum block size in frames that replaces the previous mbl value. A nonzero value enables noise recovery. Refer to section 4 for a detailed description of noise recovery.

CODE TRANSLATION TABLE TO PROCESSOR MEMORY (FUNCTION CODE-31) (SUBFUNCTION CODE-1,2,3)

The code translation table to processor memory function is issued to replace one of three code translation tables (refer to appendix C) residing in the MTS processor's memory with a new user-constructed table located in PPU memory. The specific code translation table to be replaced is indicated with the subfunction code.

The format for the code translation table to processor memory function is:



Field	Bits	Octal Values	Value Description
	11-8	0	This field is unused.
sc(subfunction code)	7, 6	1	Table 1; 256 words
		2	Table 2; 256 words
		3	Table 3; 64 words
function code	5-0	31	

UNIT RESERVE FUNCTIONS

Unit reserve functions are used to establish or remove logical connections between the PPU and a tape unit.

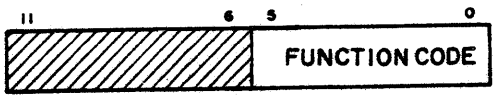
CONNECT UNIT (FUNCTION CODES-20 THROUGH 27)

The connect unit function is issued to indicate the tape unit to be used for all subsequent functions submitted. This tape unit is used exclusively until a new connect unit function is issued. The specified unit is reserved and connected. Any tape unit currently connected at the time the connect unit function is issued is disconnected but remains reserved.

A unit number of from 0 to 7 can be specified using the last three bits in the function code to indicate the desired logical unit number. That is, the 4 in function code 24 specifies logical unit number 4.

The connect unit function is rejected with an error code in detailed status (refer to section 4 and appendix A) if the unit specified is already reserved by the alternate MTS processor or no tape unit is dialed to the unit number specified.

The format for the connect unit function is:

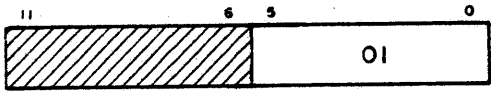


Field	Bits	Octal Value	Value Description
	11-6	0	This field is unused.
function code	5-3	2	
	0-2	0-7	Indicates the specific unit number desired.

RELEASE UNIT (FUNCTION CODE-01)

This function is issued only for a 2 x 8 configuration (section 1) to disconnect and clear the reserve status of the currently connected tape unit making the tape unit available to the alternate MTS processor in a 2 x 8 configuration.

The format for the release unit function is:

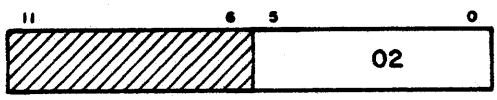


Field	Bits	Octal Value	Value Description
	11-6	0	This field is unused.
function code	5-0	01	

CLEAR ALL RESERVES (FUNCTION CODE-02)

The clear all reserves function is issued to disconnect the currently connected tape unit and to remove the unit reserve status on all reserved tape units. In a 2 x 8 configuration, the units reserved by the alternate MTS subsystem remain reserved.

The format for the clear all reserves function is:

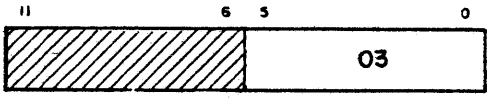


Field	Bits	Octal Value	Value Description
	11-6	0	This field is unused.
function code	5-0	02	

CLEAR OPPOSITE RESERVE (FUNCTION CODE-03)

This function is issued only for a 2 x 8 configuration (section 1) to release all tape units reserved by the alternate MTS processor making these tape units available to the subsystem to which this function was issued. This function must be used only when the alternate MTS subsystem is no longer operational.

The format for the clear opposite reserve function is:



Field	Bits	Octal Value	Value Description
	11-6	0	This field is unused.
function code	5-0	03	

UNIT MANIPULATION FUNCTIONS

Unit manipulation functions are used to initiate tape movement and positioning without the transmission of data. During execution of these functions, noise blocks are automatically passed over. The PP does not need to reissue the function.

REWIND (FUNCTION CODE-10) (SUBFUNCTION CODE-0)

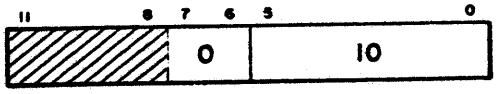
The rewind function is issued to rewind the tape on the connected unit back to loadpoint. Rewind can occur simultaneously with other tape unit operations. A write jog, which erases 1 inch of tape, is performed before status on the rewind function is returned. This occurs only if the last operation preceding the rewind was a write, write tapemark, or an erase. The write jog results in, at most, a 10-millisecond delay.

After the rewind function is issued and a write jog, if any, is executed, the PPU can connect to another tape unit to perform other tape operations.

To determine when the tape is rewound, the PPU must reconnect the unit being rewound and continue to request general status (section 3) until either the beginning of tape indicator is reached or the unit busy field in general status is 0.

If the tape to be rewound is already at loadpoint, the rewind function is accepted but has no effect.

The format for the rewind function is:



Field	Bits	Octal Value	Value Description
	11-8	0	This field is unused.
subfunction code	7, 6	0	
function code	5-0	10	

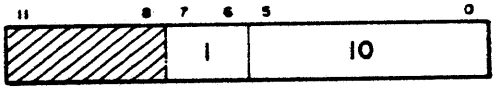
REWIND/UNLOAD (FUNCTION CODE-10) (SUBFUNCTION CODE-1)

The rewind/unload function is issued to rewind and unload the tape on the connect tape unit. Rewind/unload can occur simultaneously with other tape unit operations. A write jog, which erases 1 inch of tape, is performed before status on the rewind/unload function is returned. This occurs only if the last operation preceding the rewind/unload was a write, write file mark, or an erase. The write jog results in at most a 10-millisecond delay.

After the rewind/unload function is issued and a write jog, if any, is executed, the PPU can connect to another unit to perform other tape operations.

To determine when the tape is rewound and unloaded, the PPU must reconnect the unit being rewound and continue to request general status (section 3) until the unit busy field in general status is 0.

The format for the rewind/unload function is:



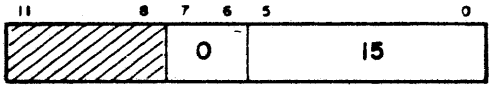
Field	Bits	Octal Value	Value Description
	11-8	0	This field is unused.
subfunction code	7, 6	1	
function code	5-0	10	

SEARCH TAPEMARK FORWARD (FUNCTION CODE-15) (SUBFUNCTION CODE-0)

The search tapemark forward function is issued to move the tape on the connected unit forward past the next tapemark into the interblock gap between the tapemark and the next block. The recognizable tapemarks and corresponding recording formats are:

Recording Format	Tapemark
7-track nonreturn to zero (NRZI)	An even parity block containing a value of 17_8.
9-track nonreturn to zero (NRZI)	An odd parity block containing a value of 13_{16}.
9-track phase encoded	A block containing between 64 and 256 flux reversals in tracks 2, 5, and 8 is written. Tracks 1, 3, 4, 6, 7, and 9 are erased.

The format of the search tapemark forward function is:



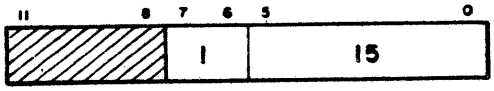
Field	Bits	Octal Value	Value Description
	11-8	0	This field is unused.
subfunction code	7, 6	0	
function code	5-0	15	

SEARCH TAPEMARK BACKWARD (FUNCTION CODE-15) (SUBFUNCTION CODE-1)

The search tapemark backward function is issued to move the tape on the connected tape unit backwards past the last tapemark into the interblock gap between the tapemark and the previous block. The recognizable tapemarks and corresponding recording formats are:

Recording Format	Tapemark
7-track nonreturn to zero (NRZI)	An even parity block containing a value of 17_8.
9-track nonreturn to zero (NRZI)	An odd parity block containing a value of 13_{16}.
9-track phase	A block containing between 64 and 256 flux reversals in tracks 2, 5, and 8. Tracks 1, 3, 4, 6, 7, and 9 are erased.

The format for the search tapemark backward function is:



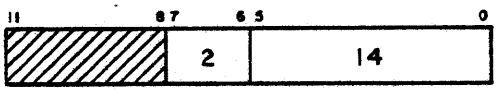
Field	Bits	Octal Value	Value Description
	11-8	0	This field is unused.
subfunction code	7, 6	1	
function code	5-0	15	

CONTROLLED FORESPACE FINDGAP (FUNCTION CODE-14) (SUBFUNCTION CODE-2)

The controlled forespace findgap function is issued to move the tape forward a specified number of frames, as if the number of frames specified were being written. Nothing is erased or written on the tape during this forespace. This initial tape motion is independent of any detected flux reversals. Then, findgap continues tape motion until 0.10 inch of erased tape is found.

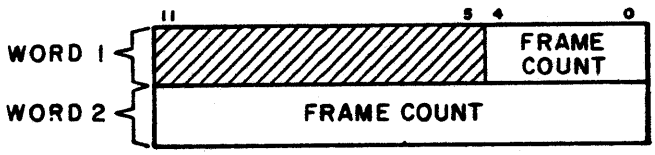
The number of frames the tape is to be moved forward is specified with a two-word parameter. If this frame count parameter is zero, the tape moves to the position where the next block would be written and then findgap searches for 0.10 inch of erased tape and comes to a stop (for example, the next block is searched through until the next interblock gap is reached, causing a stop).

The format for the controlled forespace findgap function is:



Field	Bits	Octal Value	Value Description
	11-8	0	This field is unused.
subfunction code	7-6	2	
function code	5-0	14	

The format for the two word frame count parameter is:



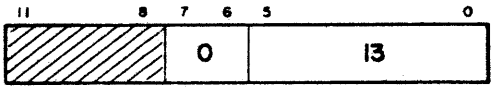
Field	Word	Bits	Value Description
frame count	1	4-0	The upper five bits of the number of frames the tape is to be moved forward (excluding findgap search).
frame count	2	11-0	The lower twelve bits (starting at 0) of the number of frames the tape is to be moved forward (excluding findgap search).

FORESPACE (FUNCTION CODE-13) (SUBFUNCTION CODE-0)

The forespace function is issued to move the tape on the connected tape unit forward one block or tapemark. If the tape is more than 10 feet past the end of tape marker, the function is not executed.

If this function is issued while the tape is within a block, the tape is moved into the next interblock gap.

The format for the forespace function is:

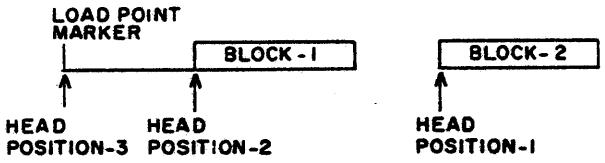


Field	Bits	Octal Value	Value Description
	11-8	0	This field is unused.
subfunction code	7, 6	0	
function code	5-0	13	

BACKSPACE (FUNCTION CODE-13) (SUBFUNCTION CODE-1)

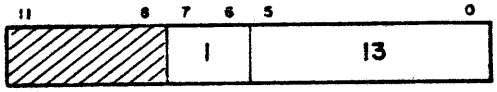
The backspace function is issued to move the tape on the connected tape unit back one block or tapemark. An error code is returned in detailed status if loadpoint is encountered. If this function is issued while the tape is within a block, the tape is moved back into the previous interblock gap.

Backspacing back into the loadpoint marker involves the following three head positions.



Issuing a backspace from head position 1 results in a new head position 2, not 3. To position head at loadpoint (head position 3) requires issuing another backspace function.

The format for the backspace function is:



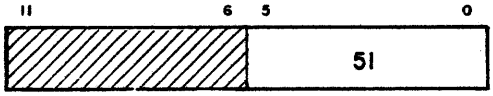
Field	Bits	Octal Value	Value Description
	11-8	0	This field is unused.
subfunction code	7, 6	1	
function code	5-0	13	

WRITE TAPEMARK (FUNCTION CODE-51)

The write tapemark function is issued to erase approximately 6 inches of tape and write one of three possible tapemarks depending on the specific recording format being used. These three possible tapemarks and their associated recording formats are:

Tapemark	Recording Format
An even parity single frame block containing a value of 17_8.	7-track nonreturn to zero (NRZI)
An odd parity single frame block containing a value of 23_8. No cyclic redundancy character is written.	9-track nonreturn to zero (NRZI)
A block containing 82 flux reversals in tracks 2, 5, and 8 is written. Tracks 1, 3, 4, 6, 7, and 9 are erased.	9-track phase

The format for the write tapemark function is:

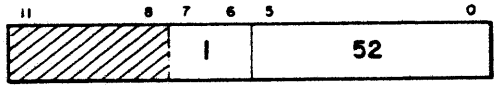


Field	Bits	Octal Value	Value Description
	11-6	0	This field is unused.
function code	5-0	51	

ERASE TO END OF TAPE (FUNCTION CODE-52) (SUBFUNCTION CODE-1)

The erase to end of tape function is issued to erase the tape until the end of tape marker is detected.

The format for the erase to end of tape function is:



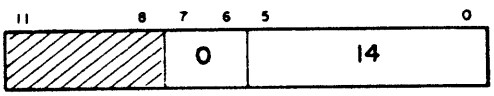
Field	Bits	Octal Value	Value Description
	11-8	0	This field is unused.
subfunction code	7-6	1	
function code	5-0	52	

CONTROLLED FORESPACE (FUNCTION CODE-14) (SUBFUNCTION CODE-0)

The controlled forespace function is issued to move the tape forward a specified number of frames, as if the number of frames specified were being written. Nothing is erased or written on the tape during this forespace. Tape motion is independent of any detected flux reversals.

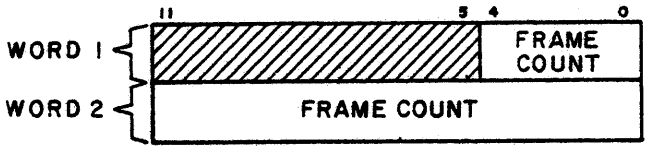
The number of frames the tape is to be moved forward is specified with a two word parameter. If this frame count parameter is zero, the tape moves to the position where the next block would be written and comes to a stop.

The format for the controlled forespace function is:



Field	Bits	Octal Value	Value Description
	11-8	0	This field is unused.
subfunction code	7-6	0	
function code	5-0	14	

The format for the two word frame count parameter is:



Field	Word	Bits	Value Description
frame count	1	4-0	The upper five bits of the number of frames the tape is to be moved forward.
frame count	2	11-0	The lower twelve bits (starting at 0) of the number of frames the tape is to be moved forward.

CONTROLLED BACKSPACE (FUNCTION CODE-14) (SUBFUNCTION CODE-1)

NOTE

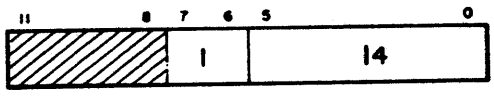
This function is intended primarily for diagnostic use, since tape speed tolerances may produce inacurrate positioning.

The controlled backspace function is issued to move the tape backward a specified number of frames from the end of the last block. Nothing is erased or written on the tape during this backspace. Tape motion is independent of any detected flux reversals.

A forward motion operation must precede a controlled backspace. A write jog is not performed before starting a controlled backspace.

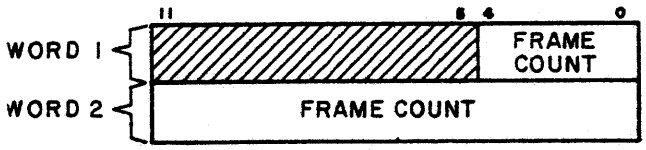
The number of frames the tape is to be moved backwards is specified with a two word parameter. If this frame count parameter is zero, the tape moves to the end of the last block and comes to a stop.

The format for the controlled backspace function is:



Field	Bits	Octal Value	Value Description
	11-8	0	This field is unused.
subfunction	7, 6	1	
function code	5-0	14	

The format for the two word frame count parameter is:



Field	Word	Bits	Value Description
frame count	1	4-0	Specified with the upper five bits of the number of frames the tape is to be moved backwards.
frame count	2	11-0	Specified with the lower twelve bits (starting at 0) of the number of frames the tape is to be moved backwards.

CONTROLLED BACKSPACE FINDGAP (FUNCTION CODE-14) (SUBFUNCTION CODE-3)

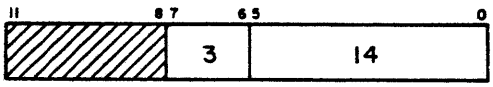
The controlled backspace findgap function moves the tape backward a specified number of frames from the end of the last block and then continues tape motion until 0.10 inch of erased tape is found. Nothing is erased or written and tape motion is independent of detected flux reversals.

A forward motion operation must precede a controlled backspace findgap. A write jog is not performed before starting a controlled backspace findgap.

A two-word frame count parameter specifies the number of frames to be passed over before the gap search begins. To prevent overshooting the gap (due to tape speed tolerances) the frame count supplied with the function should be at least six percent less than the actual number of frames preceeding the gap. (Right-shifting a number four bit positions is the same as multiplying the number by 6.25 percent.)

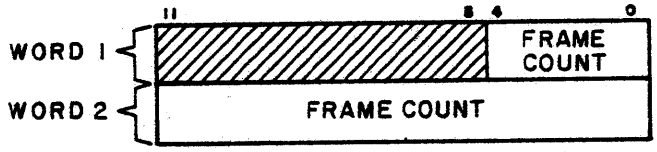
If the frame count parameter is zero, the tape moves to the end of the last block and findgap searches for 0.10 inch of erased tape and stops. For example, the last block is searched through until the previous interblock gap is found, causing a stop.

The format for the controlled backspace findgap function is:



Field	Bits	Octal Value	Value Description
	11-8	0	This field is unused.
subfunction	7, 6	3	
function code	5-0	14	

The format for the two word frame count parameter is:



Field	Word	Bits	Value Description
frame count	1	4-0	Specified with the upper five bits of the number of frames the tape is to be moved backwards (excluding findgap search).
frame count	2	11-0	Specified with the lower twelve bits (starting at 0) of the number of frames the tape is to be moved backwards (excluding findgap search).

STOP MOTION (FUNCTION CODE-11)

The stop motion function is issued to halt current tape motion during positioning requests or during read requests.

The MTS processor does not look for, or detect a stop motion function until data is detected. Therefore, it is impossible to stop tape motion in the gap before reaching the block.

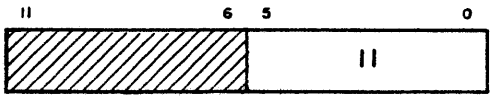
If, during execution of data transmission functions, a read is terminated by the PPU signalling end of data before the end of block is detected, tape motion continues to the next interblock gap. However, use of the stop motion function will stop the tape before the interblock gap is reached. This capability is useful when reading gapless tape.

Execution of the stop motion function terminates the following functions.

- Backspace
- Forespace
- Long Backspace
- Long Forespace
- Search Tapemark Forward
- Search Tapemark Backward
- Read Forward †
- Read Backward †
- Read Backward With Odd Length Parity †
- Reread Forward †
- Reread Backward †
- Reread Backward With Odd Length Parity †
- Repeat Read †

† For CDC CYBER 70 and 7600, the PPU cannot issue a stop motion function while data is being transferred. The PPU must stop reading data and wait up to 500 microseconds for an end-of-record signal sent by the processor before issuing the stop motion function.

The format for the stop motion function is:



Field	Bits	Octal Value	Value Description
	11-6	0	This field is unused.
function code	5-0	11	

READ FUNCTIONS

Read functions are used for initiating normal read operation.

READ FORWARD (FUNCTION CODE-40) (SUBFUNCTION CODE-0)

The read forward function is issued to perform the following sequence of operations.

1. Reads one block of data forward on the connected tape unit.
2. Assembles the data into 12-bit words.
3. Transfers the data to the PPU.

The last format unit function issued for the tape unit predefines the parity, density, minimum block length, assembly/disassembly mode, and code translation of the data being read.

Even if reading is terminated by the PPU before the end of block is reached, the full length of the block read is indicated in the frame count field of detailed status. Tape motion continues to the next interblock gap. The odd count field in general status (section 3) is 1 if there is an odd number of frames transmitted.

If the actual block length is valid (greater than or equal to the minimum block length specified), noise recovery is not initiated. This is the case even if the PPU accepts fewer frames than the minimum number defined for the tape unit.

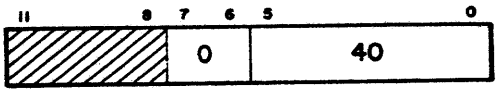
Reading stops, and an inactive (CDC CYBER 170/70 and 6000) or end-of-record (CDC CYBER 70 and 7600) signal is returned to the PPU if the end-of-block is reached while reading is in progress. Tape motion continues to the next interblock gap, even if the block extends past the end of tape marker.

MTS hardware inserts zeros to complete the last word if the number of frames read is not a multiple of 12 bits.

If the last read function issued for this unit detected a cyclic redundancy check (CRC) correctable error (limited to one track) in detailed status (refer to Detailed Status, section 3) the data on the track is automatically corrected during the next read operation

(refer to appendix D). The alert field of general status is set if the correction is necessary. If the correction is successful the parity error field of detailed status is zero.

The format for the read forward function is:



Field	Bits	Octal Value	Value Description
	11-8	9	This field is unused.
subfunction code	7, 6	0	
function code	5-0	40	

READ BACKWARD (FUNCTION CODE-40) (SUBFUNCTION CODE-1)

The read backward function is issued to perform the following sequence of operations.

1. Reads one block of data backwards on the connected tape unit.
2. Assembles the data into 12-bit words.
3. Transfers the data to the PPU.

The last format unit function issued for the tape unit predefines the parity, density, minimum block length, assembly/disassembly mode (modes 1 and 2, appendix B, are not supported), and code translation of the data.

Even if reading is terminated by the PPU before the end of block is reached, the full length of the block read is indicated in the frame count field of detailed status (refer to Detailed Status, section 3). Tape motion continues to the next interblock gap. The odd count field in general status is set if an odd number of frames is transmitted.

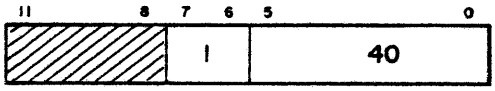
If the actual block length is valid (greater than or equal to the minimum block length specified), noise recovery is not initiated. This is the case even if the PPU accepts fewer frames than the minimum defined for the unit.

Reading stops and an inactive (CDC CYBER 170/70 and 6000) or end-of-record (CDC CYBER 70 and 7600) signal is returned to the PPU if the end-of-block is reached while reading is in progress.

MTS hardware inserts zeros to complete the last word if the number of frames read is not a multiple of 12 bits.

If the last read function issued for this tape unit detected a cyclic redundancy check (CRC) correctable error (limited to one track) in detailed status (refer to Detailed Status, section 3), the data on the track is automatically corrected during the next read operation (refer to appendix D). The alert field of general status (refer to General Status, section 3) is set if the correction is necessary. If the correction is successful, the parity error field of detailed status is zero.

The format for the read backward function is:



Field	Bits	Octal Value	Value Description
	11-8	0	This field is unused.
subfunction code	7, 6	1	
function code	5-0	40	

WRITE FUNCTIONS

Write functions are used for initiating normal write operations.

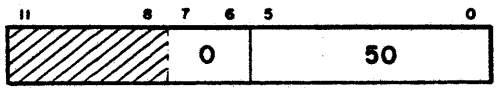
WRITE (FUNCTION-50) (SUBFUNCTION CODE-0)

The write function is issued to write one block of data in a forward direction on the connected unit. Writing is terminated when an inactive (CDC CYBER 170/70 and 6000) or end-of-record (CDC CYBER 70 and 7600) signal is received from the channel, indicating the PPU has stopped sending data.

Data is transferred from the PPU to the coupler in 12-bit bytes. From the coupler it is transmitted to the connected unit in 6-bit frames for 7-track tape or 8-bit frames for 9-track tape.

The vertical parity, assembly/disassembly mode, and code translation to be used for the data being written is predefined by the last format unit function issued for the connected tape unit.

The format for the write function is:



Field	Bits	Octal Values	Value Description
	11-8	0	This field is unused.
subfunction code	7, 6	0	
function code	5-0	50	

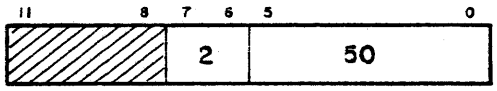
WRITE ODD LENGTH (FUNCTION CODE-50) (SUBFUNCTION CODE-2)

The write odd length function is issued to write one block of data, consisting of an odd number of frames, in a forward direction on the connected tape unit. Writing is terminated when an inactivate (CDC CYBER 170/70 and 6000) or an end-of-record (CDC CYBER 70 and 7600) signal is received from the coupler indicating the PPU has stopped sending data.

Data is transferred from the PPU to the coupler in 12-bit bytes. From the coupler it is transmitted to the connected unit in 6-bit frames for 7-track tape or 8-bit frames for 9-track tape.

The vertical parity, assembly/disassembly mode, and code translation to be used for the data being written is predefined by the last format unit function issued for the connected unit.

The format for the write odd length function is:



Field	Bits	Octal Value	Value Description
	11-8	0	This field is unused.
subfunction code	7, 6	2	
function code	5-0	50	

STATUS FUNCTIONS

Status functions are used to initiate transmission of specific types of information that can indicate if execution of the previous function was successful and/or present hardware conditions.

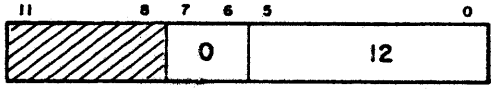
GENERAL STATUS (FUNCTION CODE-12) (SUBFUNCTION CODE-0)

The general status function is issued to return general status to PPU memory over the data channel (CDC CYBER 170/70 and 6000) or the status channel (CDC CYBER 70 and 7600). The description and format for general status is in section 3.

For CDC CYBER 70 and 7600, general status consists of a single 12-bit byte. For CDC CYBER 170/70 and 6000, general status consists of a general status byte and a block identification code byte. The CDC CYBER 170/70 and 6000 PPU must deactivate the channel after receiving either one or two general status bytes.

If end-of-operation has occurred, the MTS processor returns general status within 20 microseconds after accepting the function. If more than 20 microseconds elapse without the status word being returned, a nondata read is being performed and an end-of-operation has not occurred (no word will be returned). In this case, the general status function must be reissued.

The format for the general status function is:

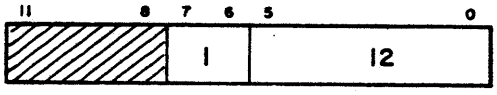


Field	Bits	Octal Value	Value Description
	11-8	0	This field is unused.
subfunction code	7, 6	0	
function code	5-0	12	

DETAILED STATUS (FUNCTION CODE-12) (SUBFUNCTION CODE-1)

The detailed status function is issued to return eight 12-bit bytes of detailed status information to PPU memory over the data channel. The description and format for detailed status is in section 3.

The format for the detailed status function is:



Field	Bits	Octal Value	Value Description
	11-8	0	This field is unused.
subfunction code	7, 6	1	
function code	5-0	12	

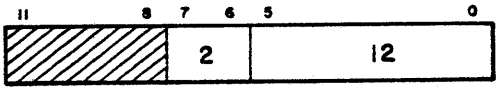
CUMULATIVE STATUS (FUNCTION CODE-12) (SUBFUNCTION CODE-2)

The cumulative status function is issued to return eight bytes of cumulative status information regarding the performance of the currently connected unit over the data channel to PPU memory.

The tally fields located in the MTS processor's memory are cleared after the status information is sent to the PPU.

The description and format for general status is described in section 3.

The format for the cumulative status function is:

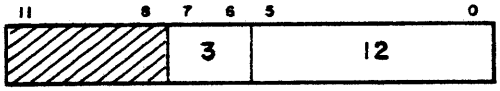


Field	Bits	Octal Value	Value Description
	11-8	0	This field is unused.
subfunction code	7, 6	2	
function code	5-0	12	

UNITS READY STATUS (FUNCTION CODE-12) (SUBFUNCTION CODE-3)

The units ready status function is issued to return two 12-bit bytes of status, containing a bit map of all ready units, over the data channel to PPU memory. Refer to section 3 for a detailed description of unit ready status.

The format for the units ready status function is:



Field	Bits	Octal Value	Value Description
	11-8	0	This field is unused.
subfunction code	7, 6	3	
function code	5-0	12	

NONMOTION READ RECOVERY FUNCTIONS

Nonmotion read recovery functions are used to temporarily modify read characteristics during read error recovery. The effect of any of these functions is removed following completion of the next motion function.

SELECT NOMINAL READ CLIP (FUNCTION CODE-06) (SUBFUNCTION CODE-0)

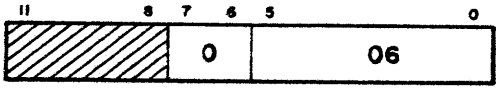
The select nominal read clip function is issued to set the read clip (read threshold) in the tape unit to a specific percentage of the maximum amplitude. Signals falling below the specified read clip are ignored by the hardware.

The nominal read clip percentages set by issuing this function for either read or read after write operations using either NRZI or phase encoded formatted tape are as follows:

Operation	NRZI	Phase Encoded
Read	20 percent	16 percent
Read-after-write	35 percent	28 percent

The function is not required to restore nominal clip. Nominal read clip is automatically restored at the completion of all motion functions.

The format for the select nominal read clip function is:



Field	Bits	Octal Value	Value Description
	11-8	0	This field is unused.
subfunction code	7, 6	0	
function code	5-0	06	

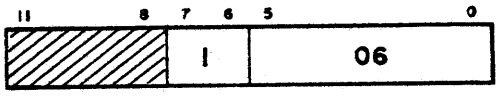
SELECT HIGH READ CLIP (FUNCTION CODE-06) (SUBFUNCTION CODE-1)

The select high clip function is issued to set the read clip (read threshold) in the unit to a specific percentage of the maximum amplitude. Signals falling below the specified read clip are ignored by the hardware.

The high read clip percentages set by issuing this function for either read or read after write operations using either NRZI or phase-encoded formatted tape are as follows:

Operation	NRZI	Phase Encoded
Read	27 percent	22 percent
Read-after-write	35 percent	28 percent

The format for the select high read clip function is:



Field	Bits	Octal Value	Value Description
	11-8	0	This field is unused.
subfunction code	7, 6	1	
function code	5-0	06	

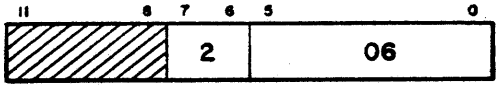
SELECT LOW READ CLIP (FUNCTION CODE-06) (SUBFUNCTION CODE-2)

The select low read clip function is issued to set the read clip (read threshold) in the unit to a specific percentage of the maximum amplitude. Signals falling below the clip level are ignored by the hardware.

The low read clip percentages set by issuing this function for either read or read after write operations using either NRZI or phase encoded formatted tape are as follows:

Operations	NRZI	Phase Encoded
Read	14 percent	0 percent
Read-after-write	35 percent	28 percent

The format for the select low read clip function is:



Field	Bits	Octal Value	Value Description
	11-8	0	This field is unused.
subfunction code	7, 6	2	
function code	5-0	06	

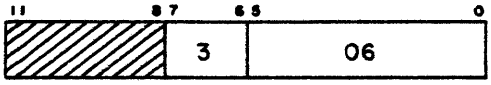
SELECT HYPER READ CLIP (FUNCTION-06) (SUBFUNCTION CODE-3)

The select hyper read clip function is issued to set the read clip (read threshold) in the unit to a specific percentage of the maximum amplitude. Signals falling below the clip level are ignored by the hardware.

The hyper read clip percentages set by issuing this function for either read or read after write operations using either NRZI or phase encoded formatted tape are as follows:

Operations	NRZI	Phase Encoded
Read	35 percent	28 percent
Read-after-write	35 percent	28 percent

The format for the select hyper read clip function is:



Field	Bits	Octal Value	Value Description
	11-8	0	This field is unused.
subfunction code	7, 6	3	
function code	5-0	6	

NOMINAL READ SPROCKET DELAY (FUNCTION CODE-07) (SUBFUNCTION CODE-0)

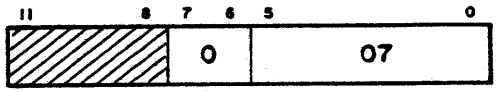
The nominal read sprocket delay function is issued to specify the time lag between the read sprocket (occurrence of first flux change in any track) and the instant all tracks are sampled to define the character.

This function is not required to be issued to restore nominal read sprocket delay. Nominal read sprocket delay is automatically restored at the completion of all motion functions.

This time specification, set by issuing this function, is a percentage of the frame time for either a read or read after write operation as follows:

Operation	Percentage of Frame Time
Read	50 percent
Read-after-write	25 percent

The format for the nominal read sprocket delay function is:



Field	Bits	Octal Value	Value Description
	11-8	0	This field is unused.
subfunction code	7, 6	0	
function code	5-0	07	

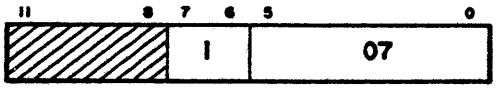
INCREASE READ SPROCKET DELAY (FUNCTION CODE-07) (SUBFUNCTION CODE-1)

The increase read sprocket delay function is issued to specify the time lag between the read sprocket (occurrence of first flux change in any track) and the instant all tracks are sampled to define the character.

This time specification, set by issuing this function, is a percentage of the frame time for either a read or read after write operation as follows:

Operation	Percentage of Frame Time
Read	52 percent
Read-after-write	25 percent

The format for the increase read sprocket delay function is:



Field	Bits	Octal Value	Value Description
	11-8	0	This field is unused.
subfunction code	7, 6	1	
function code	5-0	07	

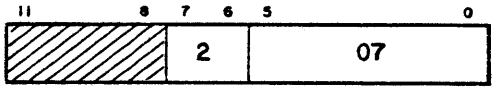
DECREASE READ SPROCKET DELAY (FUNCTION CODE-07) (SUBFUNCTION CODE-2)

The decrease read sprocket delay function is issued to specify the time lag between the read sprocket (occurrence of first flux change in any track) and the instant all tracks are sampled to define the character.

This time specification, set by issuing this function, is a percentage of the frame time for either a read or read after write operation as follows:

Operation	Percentage of Frame Time
Read	48 percent
Read-after-write	25 percent

The format for the decrease read sprocket delay function is:



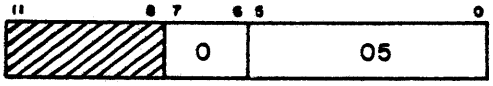
Field	Bits	Octal Value	Value Description
	11-8	0	This field is unused.
subfunction code	7, 6	2	
function code	5-0	07	

OPPOSITE PARITY MODE (FUNCTION CODE-05) (SUBFUNCTION CODE-0)

The opposite parity mode function is issued to invert the value last specified by a format unit function for a 7-track tape in an attempt to perform an error free read on the tape. The mode is restored to its previous value after the next function involving tape movement terminates.

For 9-track tapes, the function is accepted but the mode is not changed.

The format for the opposite parity mode function is:



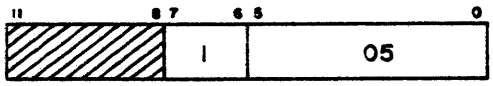
Field	Bits	Octal Value	Value Description
	11-8	0	This field is unused.
subfunction code	7, 6	0	
function code	5-0	05	

OPPOSITE DENSITY (FUNCTION CODE-05) (SUBFUNCTION CODE-1)

The opposite density function is issued to temporarily change the density of a 7-track unit from either 556 cpi to 800 cpi or from 800 cpi to 556 cpi, depending on the density last specified by a format unit function.

If a density of 200 cpi was last specified with a format unit function or if the unit is 9-track, the function is accepted but the density is not changed.

The format for the opposite density function is:



Field	Bits	Octal Value	Value Description
	11-8	0	This field is unused.
subfunction code	7, 6	1	
function code	5-0	05	

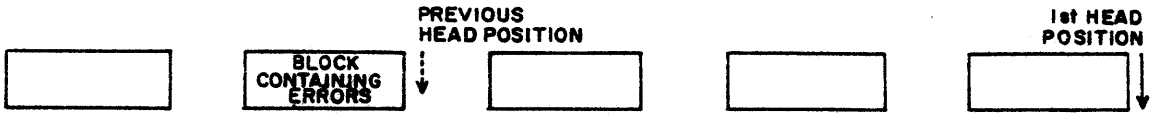
READ ERROR RECOVERY FUNCTIONS

Read error recovery functions are used to implement error recovery procedures required following a read operation that resulted in errors. The controller automatically passes over noise blocks during non-data motion functions. The PP does not need to reissue the function. Refer to section 4 for specific read error recovery procedures.

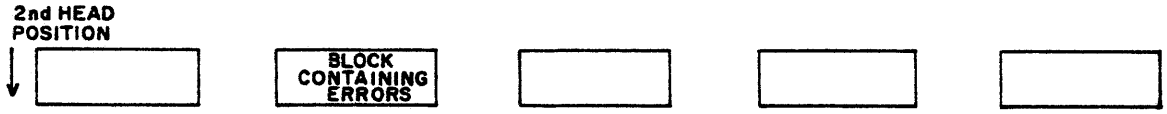
LONG FORESPACE (FUNCTION CODE-13) (SUBFUNCTION CODE-2)

The long forespace function is issued to reposition the tape for read recovery of a backward read. The function is intended to be used in association with and preceding the reread backward function. The tape is repositioned according to the following operational steps.

1. The tape is moved back three blocks of data so that the block in error is passed under the tape cleaners.

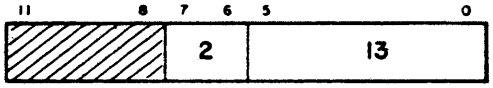


2. The tape is then moved forward five blocks so that the read head is positioned after the block immediately following the block in error.



Positioning is automatically adjusted if the loadpoint marker is detected during execution of the long forespace function.

The format for the long forespace function is:



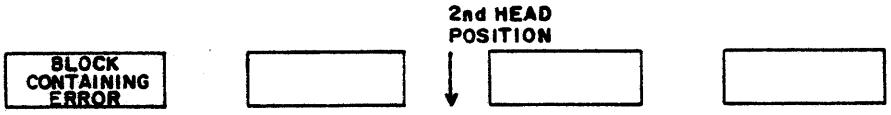
Field	Bits	Octal Value	Value Description
	11-8	0	This field is unused.
subfunction code	7, 6	2	
function	5-0	13	

LONG BACKSPACE (FUNCTION CODE-13) (SUBFUNCTION CODE-3)

The long backspace function is issued to reposition the tape for read recovery of a forward read. The function is intended to be used in association with, and preceding the reread forward function. The tape is repositioned according to the following operational steps.

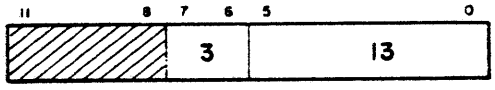
1. The tape is moved back over four blocks of data so that the block in error is passed under the tape cleaners.

2. The tape is then moved forward two blocks so that the read head is positioned in front of the block immediately preceding the block in error.



Positioning is automatically adjusted if the loadpoint marker is encountered during execution of the long backspace function.

The format for the long backspace function is:

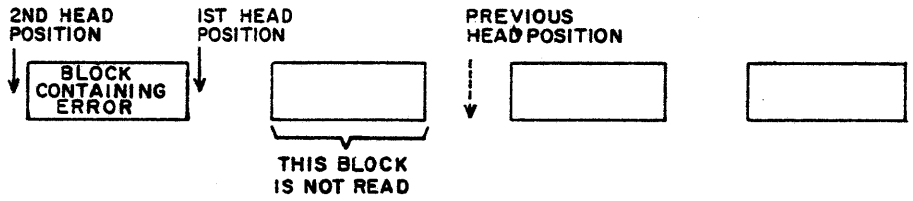


Field	Bits	Octal Value	Value Description
	11-8	0	This field is unused.
subfunction code	7, 6	3	
function code	5-0	13	

REREAD FORWARD (FUNCTION CODE-41) (SUBFUNCTION CODE-0)

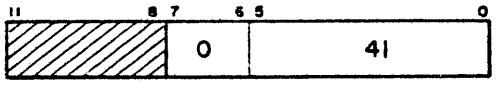
The reread forward function is issued to move the tape forward from the position set by execution of the previous long backspace function to enable a read recovery operation. The forward movement of the tape as the result of the reread function involves the following operational steps.

1. The tape is moved forward one block.
2. The forward movement continues nonstop into the next block where reading of the tape begins.



If the tape is initially positioned at loadpoint for execution of the reread function, the first block is read. No forward movement (step 1) is performed.

The format for the reread forward function is:



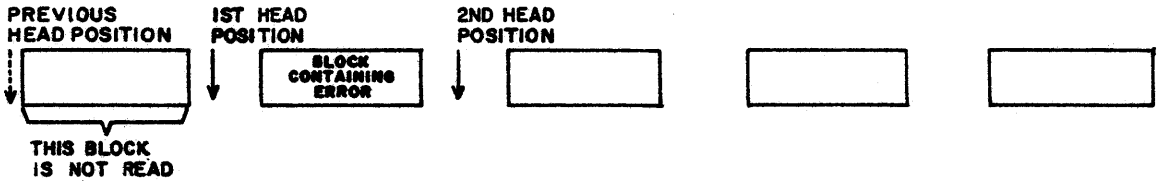
Field	Bits	Octal Value	Value Description
	11-8	0	This field is unused.
subfunction code	7, 6	0	
function code	5-0	41	

REREAD BACKWARD (FUNCTION CODE-41) (SUBFUNCTION CODE-1)

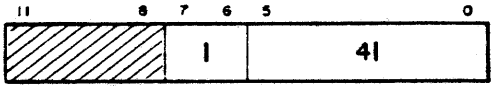
The reread backward function is issued to move the tape backwards from the position set by execution of the previous long forespace function, enabling a read recovery operation. The backward movement of the tape as the result of the reread function involves the following operational steps.

1. The tape is moved backwards one block.

2. The backward movement continues nonstop to the next block where reading of the tape begins.



The format for the reread backward function is:



Field	Bits	Octal Value	Value Description
	11-8	0	This field is unused.
subfunction code	7, 6	1	
function code	5-0	41	

READ BACKWARD WITH ODD LENGTH PARITY (FUNCTION CODE - 40) (SUBFUNCTION CODE-3)

The read backward with odd length parity function is issued to perform the following sequence of operations.

1. Selection of the correct (even) vertical parity test on the cyclic redundancy check character (CRCC) for an odd-length block.
2. Reads one block of data in a backwards direction on the connected unit.
3. Assembles the data into 12-bit words.
4. Transfers the data to the PPU.

The last format unit function issued for the tape unit predefines the parity, assembly/disassembly mode (modes 1 and 2 are not supported), and code translation of the data.

Even if reading is terminated by the PPU before the end-of-block is reached, the full length of the block being read is indicated in the frame count field of detailed status. Tape motion continues to the next interblock gap. The odd count field in general status is set if an odd number of frames is transmitted.

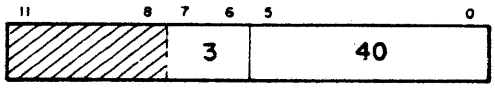
If the actual block length is valid, noise recovery is not initiated. This is the case even if the PPU accepts fewer frames than the minimum defined for the unit.

Tape motion stops and an inactive (CDC CYBER 170/70 and 6000 only) or end of record (CDC CYBER 70 and 7600 only) signal is returned to the PPU if the end of block is reached while reading is in progress.

MTS hardware inserts zeros to complete the last word if the number of frames read is not a multiple of 12 bits.

If the last read function issued for this unit detected a cyclical redundancy check (CRC) correctable error (limited to one track) in detailed status (refer to Detailed Status, section 3) the data on the track is automatically corrected during the next read operation (appendix D). The alert field of general status (refer to General Status, section 3) is set if the correction is necessary. If the correction is successful, the parity error field of detailed status is zero.

The format for the read backward with odd length parity function is:



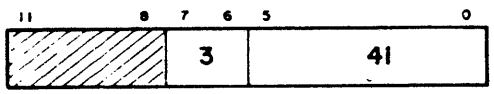
Field	Bits	Octal Value	Value Description
	11-8	0	This field is unused.
subfunction code	7, 6	3	
function code	5-0	40	

REREAD BACKWARD WITH ODD LENGTH PARITY (FUNCTION CODE-41) (SUBFUNCTION CODE-3)

The reread backward function is issued to move the tape backwards from the position set by execution of the previous long forespace function enabling a read recovery operation. Execution of the reread backward with odd length parity function involves the following operational steps.

1. Selection of the correct (even) vertical parity test on the cyclic redundancy check character (CRCC) for an odd-length block.
2. The tape is moved backwards one block.
3. The backward movement continues nonstop to the next block where reading of the tape begins.

The format for the reread backward with odd length parity function is:

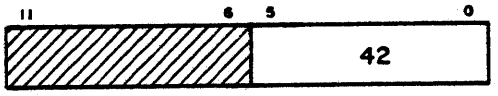


Field	Bits	Octal Value	Value Description
	11-8	0	This field is unused.
subfunction code	7, 6	3	
function code	5-0	41	

REPEAT READ (FUNCTION CODE-42)

The repeat read function is issued to repeat the current read operation as requested by the MTS processor when noise was encountered.

The format for the repeat read function is:



Field	Bits	Octal Values	Value Description
	11-6	0	This field is unused.
function code	5-0	42	

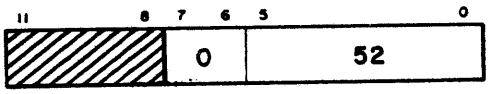
WRITE ERROR RECOVERY FUNCTIONS

Write recovery functions are used to implement error recovery procedures required following a write operation in which errors were detected. Refer to section 4 for specific write error recovery procedures.

ERASE (FUNCTION CODE-52) (SUBFUNCTION CODE-0)

The erase is issued to erase approximately 6 inches of tape.

The format for the erase function is:

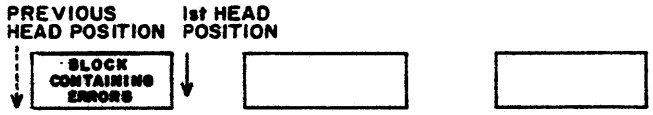


Field	Bits	Octal Value	Value Description
	11-8	0	This field is unused.
subfunction code	7, 6	0	
function code	5-0	52	

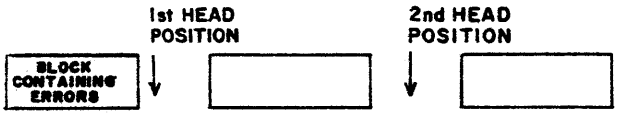
WRITE REPOSITION USING BACKSPACE (FUNCTION CODE-17) (SUBFUNCTION CODE-0)

The write reposition using backspace function followed by a two-word parameter is issued to perform the following operational steps.

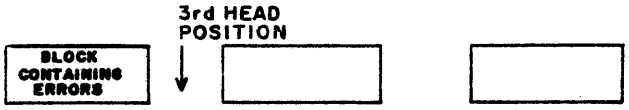
1. The tape is moved back over the block containing errors.



2. The tape is moved back over the previous block.



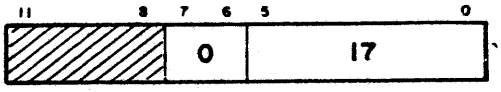
3. The tape is moved forward one block to realign tape azimuth and to turn on the erase current after the end of the previous block is detected. If the loadpoint marker is detected during the backspace, this forward movement is not performed.



This function must be preceded by forward motion of the tape.

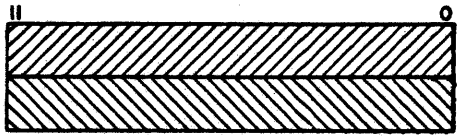
No write jog is performed before the tape is moved backward.

The format for the write reposition function is:



Field	Bits	Octal Value	Value Description
	11-8	0	This field is unused.
subfunction code	7, 6	0	
function code	5-0	17	

The format for the two word parameter is:



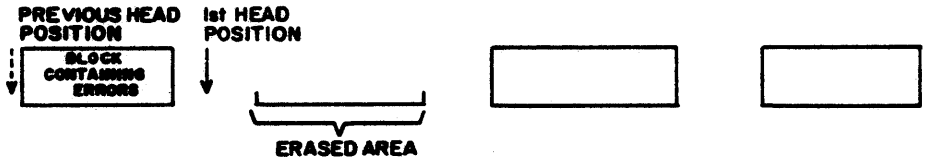
Word	Bits	Value Description
1	11-0	Any value.
2	11-0	Any value.

Although controlware ignores the parameter supplied with a 17/0 function, the two-word parameter must still be included to ensure compatability with previous operating system tape drivers.

WRITE REPOSITION TO ERASE USING BACKSPACE (FUNCTION CODE-17) (SUBFUNCTION CODE-1)

The write reposition to erase using backspace function followed by a two-word parameter is issued to perform the following operational steps.

1. The tape is moved back over the block containing errors.

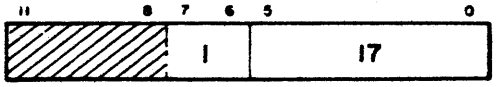


2. The tape is moved backwards an additional inch to ensure that the erase head is moved into the interblock gap.

This function must be preceded by forward motion of the tape.

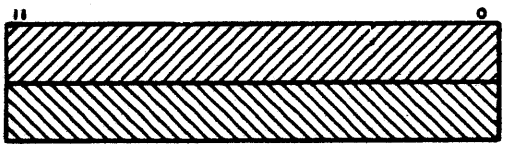
No write jog is performed before the tape is moved backward.

The format for the write reposition to erase using backspace function is:



Field	Bits	Octal Value	Value Description
	11-8	0	This field is unused.
subfunction code	7, 6	1	
function code	5-0	17	

The format for the two-word parameter is:



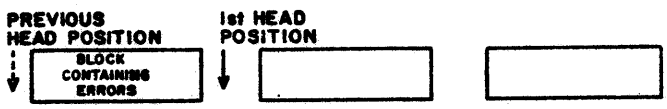
Word	Bits	Value Description
1	11-0	Any value.
2	11-0	Any value.

Although controlware ignores the parameter supplied with a 17/1 function, the two-word parameter must still be included to ensure compatibility with previous operating system tape drivers.

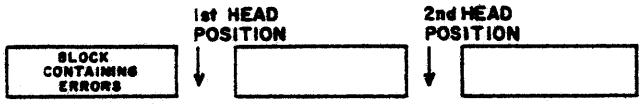
WRITE REPOSITION USING CONTROLLED BACKSPACE (FUNCTION CODE-17) (SUBFUNCTION CODE-2)

This function and its associated byte count parameter perform the following steps.

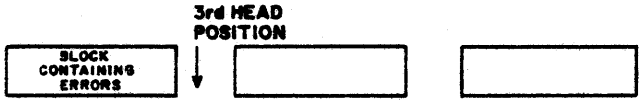
1. The tape is moved backward 90 percent of the length specified by the byte count parameter. Motion is continued until an erased area of 0.10 inch is detected.



2. The tape is moved back over the previous block.



3. The tape is moved forward one block to realign tape azimuth and to turn on the erase current after the end of the previous block is detected. If the loadpoint marker is detected during the backspace, this forward movement is not performed.

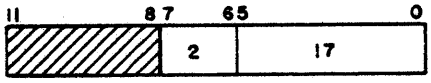


This function must be preceded by forward motion of the tape.

No write jog is performed before the tape is moved backward.

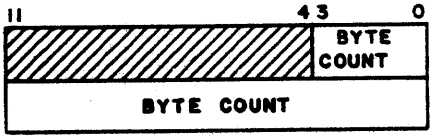
When it detects flux changes within 0.35 inch after step 2 begins, controlware sets the alert bit in general status, sets the error code field in detailed status to 14_8 (short read access), and terminates the write reposition function. (Error code 14_8 accommodates the rare case where write permit status from the tape unit occurs before the 1st HEAD POSITION is achieved in step 1.) The operating system should declare an irrecoverable write error condition unless the operating system determines that additional recovery steps will provide correct positioning.

The format for the write reposition using controlled backspace function is:



Field	Bits	Octal Value	Value Description
	11-8	0	This field is unused.
subfunction code	7,6	2	
function code	5-0	17	

The format for the byte count parameter is:

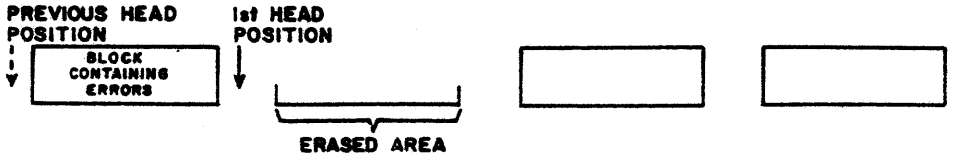


Field	Word	Bits	Value Description
byte count	1	3-0	The upper order bits that indicate the number of 12-bit bytes in the erroneous block.
byte count	2	11-0	The lower order bits (starting at 0) that indicate the number of 12-bit bytes in the erroneous block.

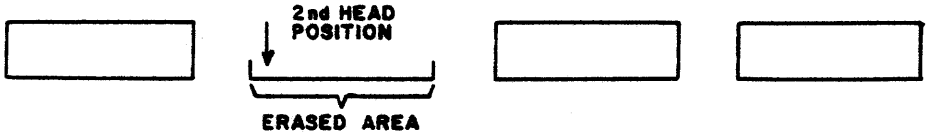
WRITE REPOSITION TO ERASE USING CONTROLLED BACKSPACE (FUNCTION CODE-17) (SUBFUNCTION CODE-3)

This function and its associated byte count parameter perform the following steps.

1. The tape is moved backward 90 percent of the length specified by the byte count parameter. Motion continues until the controller detects an erased area of 0.10 inch.



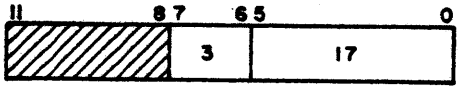
2. The tape is moved backward an additional inch to ensure that the erase head is moved into the interblock gap.



This function must be preceded by forward motion of the tape.

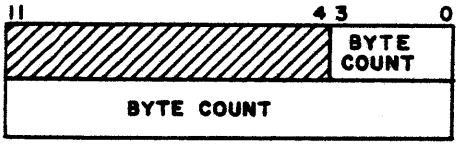
No write jog is performed before the tape is moved backward.

The format for the write reposition to erase using controlled backspace function is:



Field	Bits	Octal Value	Value Description
	11-8	0	This field is unused.
subfunction code	7, 6	3	
function code	5-0	17	

The format for the byte count paremeter is:

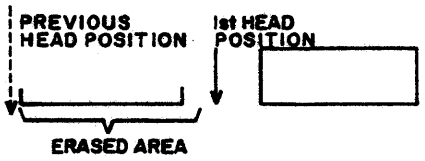


Field	Word	Bits	Value Description
byte count	1	3-0	The upper-order bits that indicate the number of 12-bit bytes contained in the erroneous block.
byte count	2	11-0	The lower-order bits that indicate the number of 12-bit bytes contained in the erroneous block.

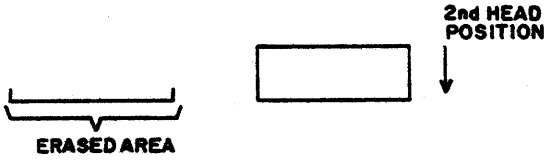
ERASE REPOSITION (FUNCTION CODE-16) (SUBFUNCTION CODE -0)

The erase reposition function is issued to perform the following operational steps.

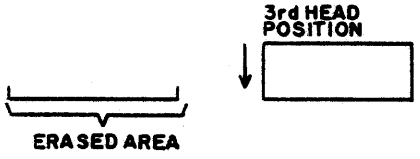
1. The tape is moved backwards 5.4 inches.



2. The tape is moved backwards over the previous block.



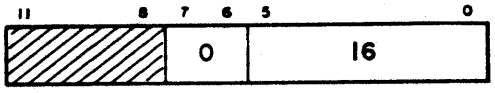
3. The tape is moved forward one block to realign tape azimuth and to turn the erase current on as soon as the last block is detected. This step is not performed if the loadpoint marker was detected during execution of step 2.



This function must be preceded by forward movement of the tape.

No write jog is performed before the tape is moved backward.

The format for the erase reposition function is:

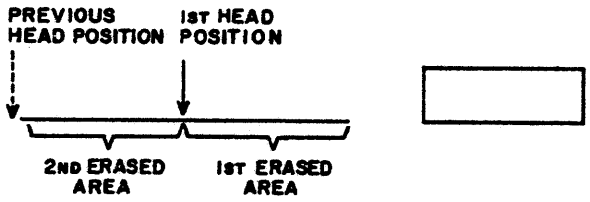


Field	Bits	Octal Value	Value Description
	11-8	0	This field is unused.
subfunction code	7, 6	0	
function code	5-0	16	

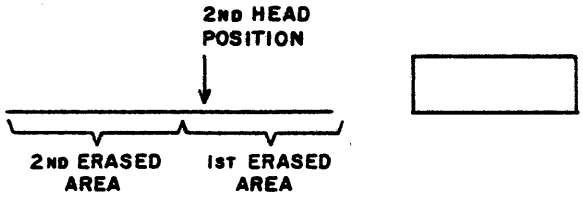
ERASE REPOSITION TO ERASE (FUNCTION CODE-16) (SUBFUNCTION CODE-1)

The erase reposition to erase function is issued to perform the following operational steps.

1. The tape is moved backwards 5.4 inches.



2. The tape is moved backwards an additional one inch to ensure that the erase head is moved into a previously erased area.

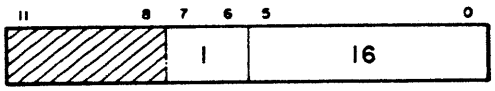


This function should be issued only if the last two functions issued were erase functions.

This function must be preceded by forward movement on the tape.

No write jog is performed before the tape is moved backwards.

The format for the erase reposition to erase function is:



Field	Bits	Octal Value	Value Description
	11-8	0	This field is unused.
subfunction code	7, 6	1	
function code	5-0	16	

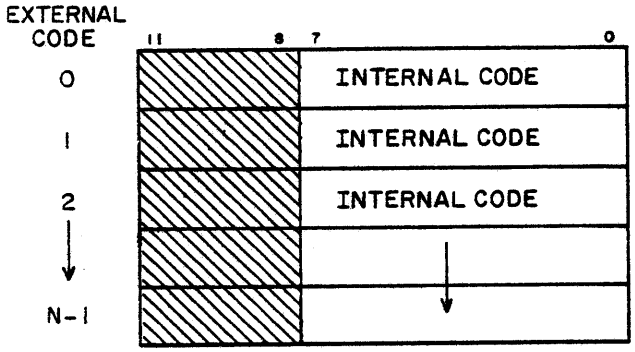
DIAGNOSTIC FUNCTIONS

Diagnostic functions are used to provide direct access to subsystem hardware for PPU resident diagnostics software. The format TCU or units ready status function is both a diagnostic function and a function used by the operating system.

LOAD READ RAM (FUNCTION CODE-32) (SUBFUNCTION CODE-1)

The load read RAM function is issued to load the read random access memory residing in the tape control unit. This is accomplished by transferring over the data channel a user defined code translation table consisting of up to 256 12-bit bytes from PPU memory through the MTS processor table 2 memory to the random access memory.

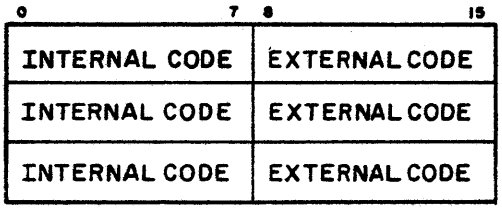
The table resides initially in the PPU memory in the following format.



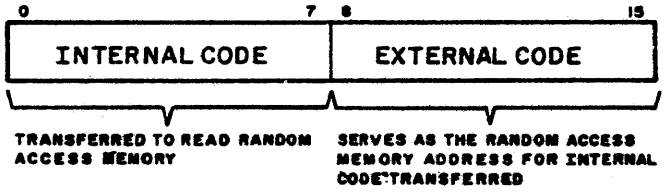
external code — Indicates an ordinal value corresponding to the position of a specific internal code.

internal code — Indicates the internal code value corresponding to a specific external code ordinal.

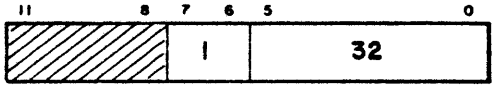
The table when transferred and located in MTS memory is formatted as follows:



The read random access memory is loaded by transferring the internal code stored in bits 0 through 7 from each word of the code translation table in MTS memory. Bits 8 through 15 of each word in the table serve as the random access memory address for the internal codes transferred. Up to 256 entries are loaded.



The format for the load read RAM function is:

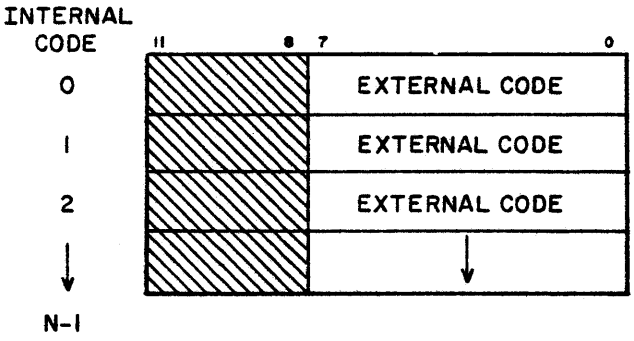


Field	Bits	Octal Value	Value Description
	11-8	0	This field is unused.
subfunction code	7, 6	1	
function code	5-0	32	

LOAD WRITE RAM (FUNCTION CODE-32) (SUBFUNCTION CODE-2)

The load write RAM function is issued to load the write random access memory residing in the tape control unit. This is accomplished by transferring over the data channel a user defined code translation table consisting of up to 256 bytes from PPU memory through the MTS processor table 2 memory to the random access memory.

The table resides initially in the PPU memory in the following format.

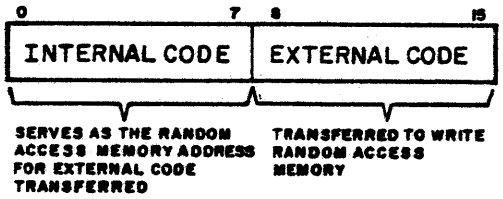


internal code	Indicates an ordinal value corresponding to the position of a specific external code.
external code	Indicates the external code value corresponding to a specific internal code ordinal.

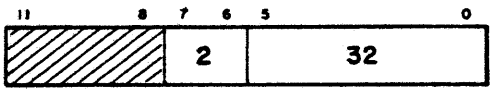
The table when transferred and located in MTS memory is formatted as follows:

0 – 7	8 – 15
INTERNAL CODE	EXTERNAL CODE
INTERNAL CODE	EXTERNAL CODE
INTERNAL CODE	EXTERNAL CODE

Loading the write random access memory involves transferring the external code in bits 8 through 15 from each word of the code translation table in MTS memory. Bits 0 through 7 of each word in the table serve as the random access memory address for the external codes transferred. Up to 256 entries are loaded.



The format for the load write RAM function is:

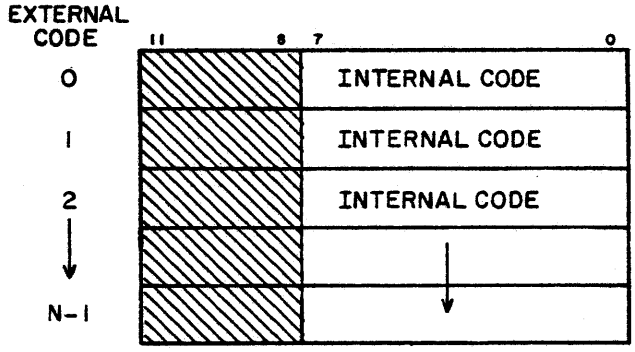


Field	Bits	Octal Value	Value Description
	11-8	0	This field is unused.
subfunction code	7, 6	2	
function code	5-0	32	

LOAD READ/WRITE RAM (FUNCTION CODE-32) (SUBFUNCTION CODE-3)

The load read/write RAM function is issued to load both the read and write random access memories residing in the tape control unit. This is accomplished by transferring over the data channel a user defined code translation table consisting of up to 256 12-bit bytes from PPU memory through the MTS processor table 2 memory to the random access memory.

The table resides initially in the PPU memory in the following format.



external code — Indicates an ordinal value corresponding to the position of a specific internal code to be loaded into the read random access memory and the external code value to be loaded into the write random access memory corresponding to a specific internal code ordinal.

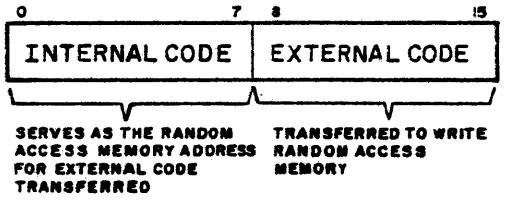
internal code — Indicates the internal code value corresponding to a specific external code ordinal to be loaded into the read random access memory and an ordinal value corresponding to the position of a specific external code value to be loaded into the write random access memory.

The table when transferred and located in MTS memory is formatted as follows:

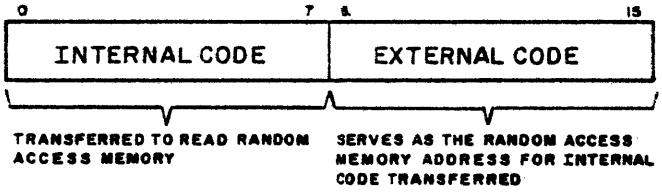
0 … 7	8 … 15
INTERNAL CODE	EXTERNAL CODE
INTERNAL CODE	EXTERNAL CODE
INTERNAL CODE	EXTERNAL CODE

As the code translation table is moved into both random access memories, each 16-bit word of the table in MTS memory is used to simultaneously load one 8-bit entry in both the write and read random access memories. Up to 256 entries are loaded into both the write and read random access memories.

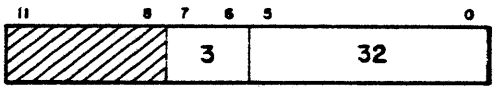
Loading the write portion of random access memory involves transferring the external code stored in bits 8 through 15 from each word of the code translation table in MTS memory. Bits 0 through 7 of each word in the table serves as the random access memory address for the external codes transferred.



The read random access memory is loaded by transferring the internal code stored in bits 0 through 7 from each word of the code translation table in MTS memory. Bits 8 through 15 of each word in the table serve as the random access memory address for the internal codes transferred.



The format for the load read/write RAM function is:

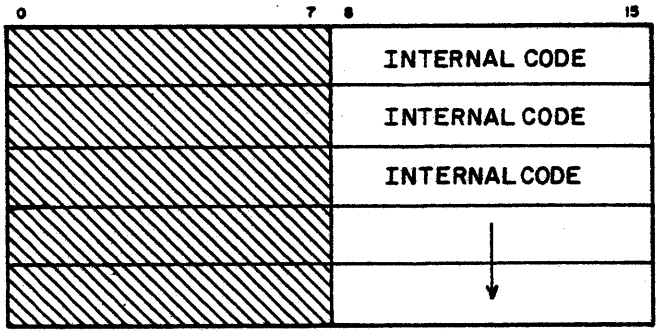


Field	Bits	Octal Value	Value Description
	11-8	0	This field is unused.
subfunction code	7, 6	3	
function code	5-0	32	

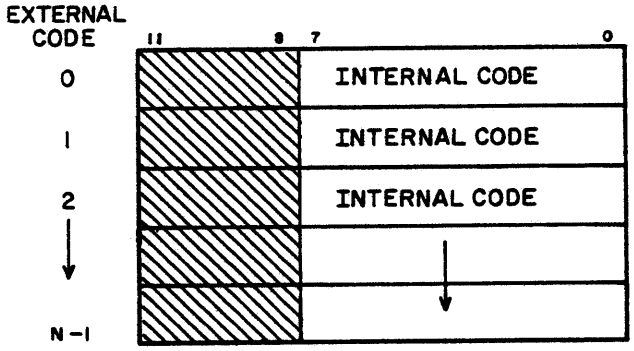
COPY READ RAM (FUNCTION CODE-33) (SUBFUNCTION CODE-1)

The copy read RAM is issued to copy the contents of the read random access memory residing in the tape control unit through MTS processor table 2 memory to PPU memory. The read random access memory is made up of 256 8-bit entries. The format of each entry is as follows:

The table, when transferred and located in MTS memory, is formatted as follows:

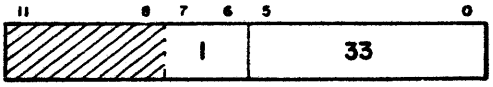


The table is copied into PPU memory in the following format.



external code	Indicates an ordinal value corresponding to the position of a specific internal code.
internal code	Indicates the internal code values corresponding to specific external code ordinals.

The format for the copy read RAM function is:



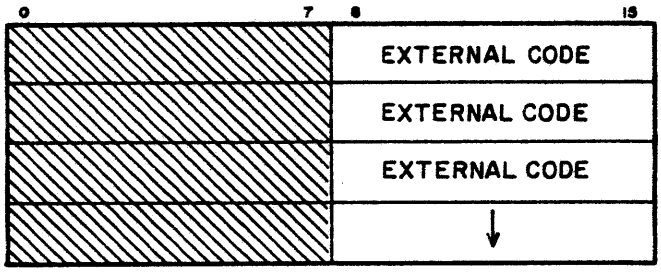
Field	Bits	Octal Value	Value Description
	11-8	0	This field is unused.
subfunction code	7, 6	1	
function code	5-0	33	

COPY WRITE RAM (FUNCTION CODE-33) (SUBFUNCTION CODE-2)

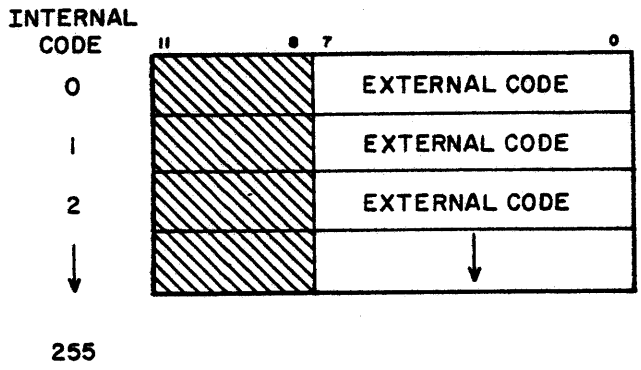
The copy write RAM function is issued to copy the contents of the write random access memory residing in the tape control unit back through MTS processor table 2 memory to the PPU memory.

The write random access memory is made up of 256 8-bit entries. The format of each entry is as follows:

The table when transferred and located in MTS memory is formatted as follows:



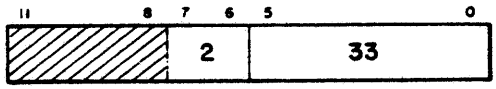
The table is copied to the PPU memory in the following format.



internal code — Indicates an ordinal value corresponding to the position of a specific external code.

external code — Indicates the external code value corresponding to a specific internal code ordinal.

The format of the copy write RAM function is:



Field	Bits	Octal Value	Value Description
	11-8	0	This field is unused.
subfunction code	7, 6	2	
function code	5-0	33	

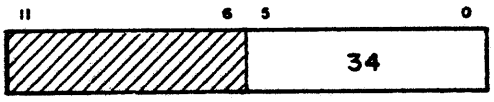
FORMAT TCU OR UNITS READY STATUS (FUNCTION CODE-34)

If a function code 34 is requested, then a one word parameter must also be issued to indicate the specific status desired. If bit 8 of the parameter word is zero, then a format TCU status is processed (as described in the next paragraph). If bit 8 of the parameter word is one, then a format units ready status is processed (as described on the next page).

FORMAT TCU STATUS (PARAMETER WORD BIT 8 EQUAL ZERO)

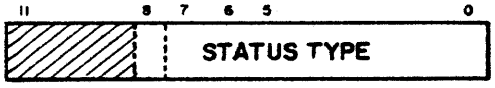
The format TCU or units ready status function is issued to specify which tape control unit or coupler status is to be returned to the PPU on all subsequent copy TCU functions. A one word parameter must be issued to indicate the specific status desired. A format TCU or units ready status function must be issued each time a different TCU status is required.

The format for the format TCU or units ready status function is:



Field	Bits	Octal Value	Value Description
	11-6	0	This field is unused.
function code	5-0	34	

The format for the parameter word is:

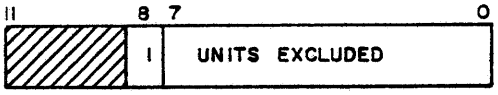


Field	Bits	Decimal Value	Value Description
status type	7-0	0	Indicates that coupler status is requested.
		1 through 20	Indicates a request to return a specific TCU status corresponding to the number specified. Refer to the Magnetic Tape Controller Hardware Reference Manual, Publication No. 60404000 for a detailed description of available TCU status.
	8	0	Indicates that format TCU status is requested.

FORMAT UNITS READY STATUS (PARAMETER WORD BIT 8 EQUAL ONE)

The format TCU or units ready status function is issued to specify which tape unit or units (if any) are to be excluded during all subsequent units ready status functions. A one word parameter must be issued to indicate the unit or units to be excluded. A format TCU or units ready status function must be issued each time a different set of units is required. A default value of no units excluded exsists when the MTS software is loaded in the MTS processor.

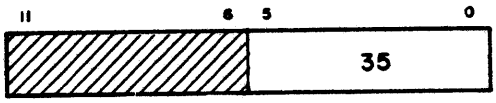
The format for the parameter word is:



Field	Bits	Description
units excluded	7-0	Bits 7 through 0 correspond to unit numbers 7 through 0. If bits 7 through 0 are all zeros then no units are to be excluded.
	8	If the bit is set, indicates that format units ready status is requested.

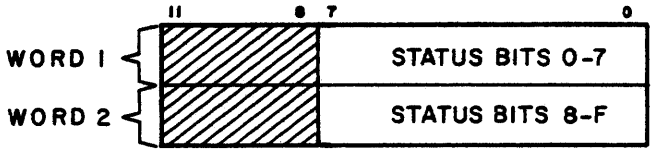
COPY TCU STATUS (FUNCTION CODE-35)

The copy TCU status function is issued to request that unedited tape control unit or coupler status be returned to PPU memory. A specific status indicated by the last format TCU status function is copied and returned to the PPU in two 12-bit words. Refer to the Magnetic Tape Controller Hardware Reference Manual for a detailed description of the status bits.



Field	Bits	Octal Value	Value Description
	11-6	0	This field is unused.
function code	5-0	35	

The following two words of status are returned.



The 16 bits of TCU status are returned in bits 7 through 0 of each word.

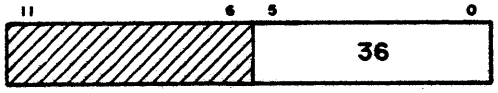
SEND TCU COMMAND (FUNCTION CODE-36)

The send TCU command function is issued to send a specific tape control unit command to the tape control unit. Refer to the Magnetic Tape Controller Hardware Reference Manual for a detailed description of tape control unit commands.

A three word parameter must be also issued to indicate the specific tape control unit command desired.

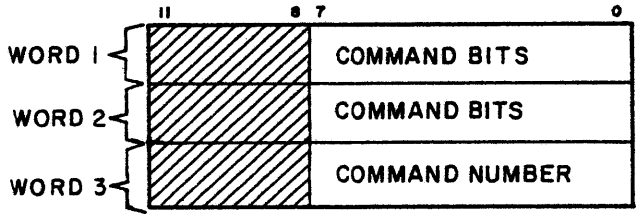
Indiscriminate use of this function can affect subsequent MTS operations, or in a 2x8 configuration, disrupt current and/or subsequent operations on the alternate MTS processor.

The format for the send TCU command function is:



Field	Bits	Octal Value	Value Description
	11-6	0	This field is unused.
function code	5-0	36	

The format for the three word parameter is:

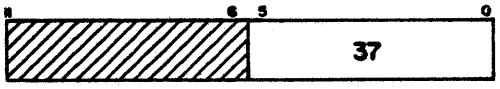


Word	Field	Bits	Decimal Value	Bit Description
1	command bits	7-0		Indicates the bit settings for TCU command bits 0 through 7.
2	command bits	7-0		Indicates the bit settings for TCU command bits 8 through F.
3	command number	7-0	1-13	Indicates the specific TCU command to be issued.

SET 25 PERCENT READ SPROCKET DELAY (FUNCTION CODE-37)

This diagnostic function selects a quarter-cell read-after-write sprocket delay window for the next read function (read functions include read forward, read backward with odd length parity, reread forward, reread backward, reread backward with odd length parity, backspace, forespace, search tapemark forward or backward, and long backspace or forespace). Selecting the quarter-cell window imposes a tighter performance standard on the tape transport while checking NRZI read capability.

The format for the set 25 percent read sprocket delay function is:



Field	Bits	Octal Value	Value Description
	11-6	0	This field is unused.
function code	5-0	37	

STATUS 3

The MTS subsystem provides four types of status information that can be returned to PPU memory. This status information can indicate whether a function executed successfully and/or indicate specific hardware conditions and operations. The four types of status are called by specific function codes (section 2) issued from the PPU†. These four types of status and the corresponding functions required to initiate transmission of each type of status to the PPU are as follows:

Status	Function
General†	General Status (Function Code - 12) (Subfunction Code - 0)
Detailed	Detailed Status (Function Code - 12) (Subfunction Code - 1)
Cumulative	Cumulative Status (Function Code - 12) (Subfunction Code - 2)
Units Ready	Units Ready Status (Function Code - 12) (Subfunction Code - 3)

† MTS under CDC CYBER 70 and 7600 automatically returns general status for certain functions without requiring the general status function to be sent.

GENERAL STATUS

General status consists either of one 12-bit word (CDC CYBER 70 and 7600) or two 12-bit words (CDC CYBER 170/70 and 6000). The first 12-bit word provides basic information concerning the present condition of hardware as well as read or write operations. The second word provides a block identification (ID) code and an even vertical parity bit. A 9-bit cyclic redundancy code is generated from the entire block last read or written. The code is direction-dependent and is invalid if the detailed status parity error bit (bit 9 in word 1) is set. Its only purpose is to help the PPU locate the last good block during read or write recovery.

Recovery functions that generate a block ID code include backspace, forespace, long backspace, long forespace, read forward, read backward, read backward with odd length parity, repeat read, reread forward, reread backward, reread backward with odd length parity, write reposition using backspace, write reposition using controlled backspace, erase reposition, write, write odd length, and write tapemark (for phase density, the second word of general status is zero immediately after the write tapemark function completes). If the tape is moved over one or more blocks as the last step of a multiple positioning function, then the block ID code from the last block moved over is returned (for example, step 3 of the write reposition using backspace function moves the tape forward one block at the end of the function).

Note that several functions return a zero block ID code rather than the code from the last block read or written (for example, the second word of general status is zero after a connect unit, release unit, rewind or rewind/unload function).

If more than 20 microseconds elapse before the MTS processor returns general status, a nondata read is being performed and an end-of-operation has not yet occurred. In this case, the general status function must be reissued.

CYBER 170/70 AND 6000

The CDC CYBER 170/70 and 6000 system requires that a general status function be issued before the MTS processor returns general status. An inactive signal is returned to the PPU when the MTS processor is ready to transmit the status word. A second inactive signal is sent by the MTS processor after the PPU accepts the status word. General status must be requested after each motion function to determine the end-of-operation.

CYBER 70/7600

General status is transmitted from the MTS processor over the status channel to the CDC CYBER 70/7600 PPU at the following times.

1. When a general status function is issued.
2. Prior to sending a resume after receiving an erroneous function, to indicate that the function was rejected.
3. When any of the functions listed on the inside back cover of this manual as receiving general status have completed execution.
4. When a short block was detected during a read, and noise recovery was specified.

The format for general status is:

	11	10	9	8	7	6	5	4	3	2	1	0
WORD 1	ALERT	COUPLER STATUS	NO UNIT	NOISE	WRITE RING	UNIT TYPE	ODD COUNT	TAPE-MARK	END OF TAPE	LOAD POINT	UNIT BUSY	UNIT READY
WORD 2	BLOCK IDENTIFICATION CODE										EVEN VERTICAL PARITY	

Word 1 Format

Field	Bit	Value	Value Description
alert	11	1	Indicates that an error was detected or a condition was encountered that must be written on the operating systems engineering file. Detailed status should be requested if the alert bit is set.
coupler status	10	1	This field indicates status originated in coupler . For a CDC CYBER 70 and 7600 systems the field is specified in conjunction with bits 3, 2, 1, or 0 to indicate one of the following conditions. Bit 3: A memory parity error occurred in MTS processor memory. Bit 2: A local autoload is in progress. Bit 1: A remote autoload is in progress. Bit 0: A deadman timeout occurred. If coupler status sets and bits 3 through 0 do not set (general status word contains 2000_8), then the deadman timer expires and bit 0 clears by the time general status is read by the PPU. If general status word contains either 2000_8 or 2001_8, send a record flag on the PPU function channel to clear the static resume and prepare the coupler for receipt of another function. (The record flag is sent because the deadman timer is expired.) If coupler is reserved by the other channel access, the record flag should have no effect (the unreserved PPU cannot clear the reservation).
no units	9	1	Indicates that no unit is currently connected.
noise	8	1	Indicates that a block was read that was shorter than the minimum block length specified. A repeat read function must be sent. This bit is not set if the block was a file mark, if the loadpoint mark was reached, or if noise recovery is not enabled.
write ring	7	1	Indicates that the write ring is in the tape reel.
unit type	6	0	Indicates 7-track unit.
		1	Indicates 9-track unit.
odd count	5	1	Indicates that an odd number of frames was read. Refer to appendix B to see how add count can be used in association with the number of 12-bit bytes transmitted to determine how many fill bits were included in the last byte.
tape mark	4	1	Indicates that a tapemark was successfully read or written.
end of tape	3	1	Indicates that the tape is positioned at or past the end of tape marker. Less than 10 feet of uasble recording surface remains.
loadpoint	2	1	Indicates that the tape is positioned at the loadpoint marker.
unit busy	1	1	Indicates that tape is in motion.
unit ready	0	1	Indicates that the tape unit is loaded and ready for operation. This bit is not set during a rewind or rewind/unload operation.

Word 2 Format

Field	Bit	Value	Value Description
block identification code	11-3		Indicates the 9-bit cycle redundancy code generated from the entire block last read or written. A forward read generates the same code (if parity error is not set) that was generated when the block was written; a backward read yields a different code.
even vertical parity	1	1	Indicates that the currently selected vertical parity mode is even.

DETAILED STATUS

Detailed status must be requested, by issuing a detailed status function, if the alert bit in general status is set. The MTS processor returns detailed status to PPU memory over the data channel in eight 12-bit bytes. The format for detailed status residing in PPU memory is:

	11	10	9	8	7	6	5	4	3	2	1	0
WORD 1	LOST DATA	ERASE ERROR	PARITY ERROR	NOISE IN GAP	NOISE FAULT	LATE DATA	ERROR CODE					
WORD 2	FRAME COUNT ERROR-1	FRAME COUNT ERROR-2	FRAME COUNT ERROR-3	SINGLE FRAME ERROR	LRCC PARITY ERROR	VERTICAL PARITY ERROR	LONGITUDINAL PARITY ERROR	CRCC PARITY ERROR	FALSE EOP	SKEW ERROR	VELOCITY ERROR	POSTAMBLE ERROR
WORD 3	SEVERE START ERROR	ODD FRAME COUNT	POSTAMBLE DETECTED	FOUR FRAME SKEW	OPPOSITE CHANNEL DOWN	SINGLE FRAME SKEW	ILLEGAL CHARACTER	LOOP FAULT	AIR PRESSURE FAULT	ERASE CURRENT FAULT	LOAD FAULT	ABNORMAL TEMPERATURE
WORD 4	TRACK IN ERROR	CRC DETECTED ERROR	MULTITRACK ERROR	TRACKS CORRECTED								
WORD 5	TAPE DIRECTION	UNIT'S SPEED			UNIT'S DENSITY		ACCESS ERROR	WRITE REPLY	UNIT'S PHYSICAL ADDRESS			
WORD 6				LARGEST NOISE SLOCK LENGTH					TAPE POSITION			
WORDS 7 AND 8	FRAME COUNT											

Word 1 Format

Field	Bit	Value	Value Description
lost data†	11	1	Indicates one of the following: 1. During a read, an end-of-record signal was not received before the next frame and all data registers were full. 2. During a write, an end-of-record signal was not received, and data was not available for writing the next frame.

Field	Bit	Value	Value Description
erase error†	10	1	Indicates that unerased flux changes were detected at a low read clip setting: 1. During an erase operation. 2. During the erase portion of a write tapemark operation. 3. In the second half of an interblock gap during a write data operation.
parity error†	9	1	Indicates an error was detected, requiring that the block be reread or rewritten. Subsequent fields in detailed status cause this field to be set.
noise in gap†	8	1	Indicates unerased flux changes were detected in the interblock gap prior to the current operation.
noise fault†	7	1	Indicates that unerased flux changes are detected either: 1. At a low read clip setting following the last write operation. 2. At a normal clip setting following the last read operation.
late data†	6	1	Indicates that data was not available to be written at write access time and within the next 0.4 inch of tape. A gap is continued up to 34 milliseconds or until data is transmitted.
error code†	5-0	nonzero	Indicates a fatal error code was detected during processing. Refer to appendix A for a description of the specific error codes appearing in this field.

†A 1 in this field causes the alert field in general status to be set.

Word 2 Format

Field	Bit	Value	Value Description
frame count error-1†	11	1	Indicates that too many frames were written before the first frame was read. This may be caused by a slow start time in the tape unit.
frame count error-2†	10	1	Indicates that more frames were read than were written. This was probably caused by unerased noise.
frame count error-3†	9	1	Indicates that fewer frames were read than were written. This was probably due to one or more of the following: 1. Missing oxide 2. Wrinkled tape 3. Dirty heads
single frame error†	8	1	(7-track NRZI only) Indicates that a frame containing all zeros (including the parity track) was read. Data will be at least one frame short.
LRCC parity error†	7	1	(9-track NRZI only) Indicates that the longitudinal redundancy check character had even vertical parity. If track correction (appendix D) was successful, the parity error bit is not set.
vertical parity error†	6	1	Indicates that one or more frames have incorrect vertical parity. If track correction (appendix D) was successful, the parity error bit is not set.
longitudinal parity error†	5	1	(NRZI only) Indicates that one or more tracks had odd longitudinal parity.
CRCC parity error†	4	1	(9-track NRZI only) Indicates one of the following conditions associated with a specific operation was detected. 1. The cyclic redundancy check character vertical parity was incorrect during a forward read or write. 2. The cyclic redundancy check character parity was even when executing a read backward function. 3. The cyclic redundancy check character parity was odd when executing a read backward with odd length parity function. The parity bit is not set for a read operation.

†A 1 causes the parity error field (bit 9 in word 1) to be set.

Field	Bit	Value	Value Description
false EOP†	3	1	Indicates that unexpected frames were detected before: 1. The longitudinal check character. 2. The postamble
skew error†	2	1	Indicates that excessive phase mode skew occurred. 1. More than 4 frames of skew occurred during a read. 2. More than 2 frames of skew occurred during a write.
velocity error†	1	1	Indicates that the velocity of the tape varied more than 7 percent after reaching operating speed. The parity bit is not set for a read operation. This bit should be ignored for read operations. An accurate tally of actual velocity errors during read is reported by cumulative status as defined in this section.
postamble error†	0	1	Indicates that a missing or defective postamble was detected as a result of: 1. Part of the block not being read. 2. A defective postamble being written.

†A 1 causes the parity error field (bit 9 in word 1) to be set.

Word 3 Format

Field	Bit	Value	Value Description
severe start error†	11	1	During a write, the interblock gap was lengthened by over 0.20 inch because the unit was slow reaching full velocity.
odd frame count	10	1	Indicates one of the following: NRZI - An odd number of frames was read or written. PHASE - An even number of frames was read or written.
postamble detected	9	1	Indicates that a postamble was detected during a phase read or write.
four frame skew†	8	1	Indicates that more than four frames of skew occurred during a phase read or a phase read after write. Causes the parity error field (bit 9 in word 1) to be set.
opposite channel down	7	1	Indicates that the opposite channel in a 2 x 8 configuration is inoperable. This field is always set in a 1 x 8 configuration.
single frame skew	6	1	Indicates that more than one frame of skew was detected during a phase read or phase read after a write.
illegal character†	5	1	Indicates that a 1 was detected in bit 6 of one or more translated characters read from tape. If converting to a 6-bit internal code, bit 6 is never set. Therefore, 8-bit external codes not included in the 64-character subset can be detected by loading the corresponding read RAM locations with an internal code containing a 1 in bit 6.
loop fault†	4	1	Indicates the unit lost a tape loop. The unit requires operator attention.
air pressure fault†	3	1	Indicates one of the following: 1. An abnormal air pressure for the tape guides that could result in tape damage. The unit automatically stops. 2. The tape cartridge failed to unlatch during a loading attempt. 3. Abnormal air pressure on supply reel hub. The READY indicator goes out and the unit stops.
erase current fault†	2	1	Indicates current in the erase head is abnormal. Defective erase operations can result from this condition. Unit is still operational, but requires immediate attention.
load fault†	1	1	Indicates that the unit failed to load.
abnormal temperature†	0	1	Indicates the temperature in unit is within 5°F of automatic power cutoff. Unit is still operable, but requires immediate attention.

†A 1 in this field causes the alert field in general status to be set.

Word 4 Format

Field	Bit	Value	Value Description
track in error†	11	1	Indicates that correction was attempted to the track or tracks indicated in the track corrected field (bits 8 through 0) of this word. If the multitrack error field (bit 9) of this word is zero, the data was successfully corrected. If set during a write, the parity error field (bit 9 in word 1) is set.
CRC detected error†	10	1	Indicates that the cyclic redundancy check detected an error reading or writing 9-track NRZI tape or while writing 7-track tape. Causes the parity error field (bit 9 in word 1) to be set.
multitrack error†	9	1	Indicates that more than one track was in error during a read operation. Causes the parity error field (bit 9 in word 1) to be set.
tracks corrected†	8-0	nonzero value	Indicates that data correction was attempted on the tracks identified by the corresponding bit numbers that are set. That is: Bit 8 Parity track Bit 7 Track 2^7 . . . . . . Bit 0 Track 2^0

† A 1 bit causes the alert field in general status to be set.

Word 5 Format

Field	Bit	Octal Value	Value Description
tape direction	11	0	Indicates the current or last direction of tape motion was forward.
		1	Indicates the current or last direction of tape was backward.
unit's speed	10-8	1	Indicates a tape speed of 100 inches per second.
		2	Indicates a tape speed of 150 inches per second.
		4	Indicates a tape speed of 200 inches per second.
unit's density	7, 6	0	Indicates a density of 200 or 556 characters per inch.
		1	Indicates a density of 800 characters per inch.
		2	Indicates a density of 1600 characters per inch.
access error	5	1	Indicates that either write permit status occurred (when writing) or the first frame was detected (when reading) before the tape reached full speed. The alert bit in general status is set for write operations if the interblock gap increased more than 0.2 inch.
write reply	4	1	Indicates that the unit's write and erase currents are on, due to the last operation being a write.
unit's physical address	3-0		Indicates the unit's cable connector address in the tape control unit. This number can be used to identify physical tape units regardless of the logical unit settings.

Word 6 Format

Field	Bit	Quantity	Value Description
largest noise block length†	8-4	frames	Indicates, in number of frames, the longest block read that was shorter than the minimum block length specified in the format unit function. Except for a repeat read, this field is cleared when any motion function is issued.
tape position	3-0	blocks	Indicates, using two's complement, the number of blocks passed over during the last operation. Positive numbers indicate forward motion, negative numbers indicate backward motion. For example: 00 Indicates tape was not moved or the function is not one of the functions listed below that clears this field when executed. 01 Tape was moved one block forward 17 Tape was moved one block backwards This field is cleared at the start of all motion functions and incremented (decremented if motion is backward) by the following functions. • Long forespace • Long backspace • Erase reposition • Write reposition • Write reposition to erase • Reread (first block only)

Words 7 and 8 Format

24-bit frame count field indicating for:

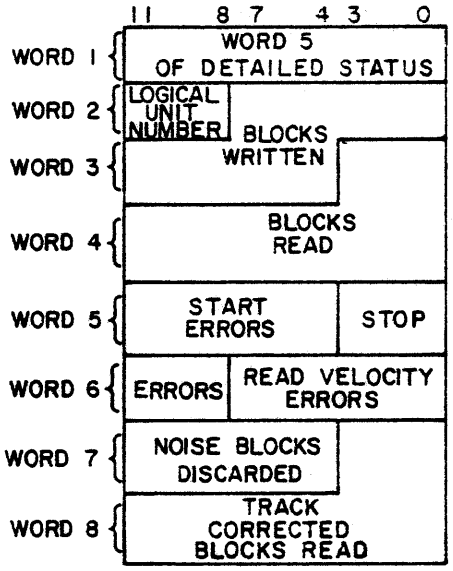
Read Operations	The number of frames in the block last read or passed over.
Write Operations	A negative (modulo 512) or positive nonzero value is indicated if bits 10 and 9 of word 2 indicate a 2 or 3 frame count error. The frame count is zero for a successful write.

† A 1 bit in this field causes the alert field in general status to be set.

CUMULATIVE STATUS

Cumulative status consists of a tally of occurrences involving marginal tape unit performance. This tally is maintained in processor memory for each tape unit being used. Cumulative status for a tape unit is cleared from MTS processor memory and transferred to PPU memory by executing the cumulative status function (refer to section 2) whenever the job is through with the tape. Cumulative status is intended to be transferred by the user from PPU memory to the PPU engineering file. Overflow in any 8-bit field cumulative status is indicated by leaving a tally at its maximum value with all bits set. †

The format of cumulative status after its returned to PPU memory is:



Word 1 Format

Word 1 is returned with bits 11 through 0 of the fifth word of detailed status (refer to Detailed Status) identifying the tape unit.

Word 2 Format

Field	Bits	Description
logical unit number	11-8	Indicates the logical unit number of the last connected drive.
blocks written	7-0	Indicates the upper 8 bits of the number of blocks written.

† This does not apply to the blocks written, blocks read, and track corrected blocks read fields.

Word 3 Format

Field	Bits	Description
blocks written	11-4	Indicates the lower 8 bits of the number of blocks written.
blocks read	3-0	Indicates the upper 4 bits of the number of blocks read forward and backward.

Word 4

Indicates the lower 12 bits of the number of blocks read forward and backward. A block is counted regardless of whether data is transmitted. That is, a search file mark function increments the count by the number of blocks it must space over to reach the file mark, including one for the file mark.

Word 5 Format

Field	Bits	Description
start errors	11-4	Indicates the number of blocks written where the tape velocity was low at write permit time but where the velocity reached full speed within 0.2 inch.
stop error	3-0	Indicates the upper 4 bits of the number of blocks written where write permit occurred too soon after starting tape motion.

Word 6 Format

Field	Bits	Description
stop errors	11-8	Indicates the lower 4 bits of the number of blocks written where write permit occurred too soon after starting tape motion.
read velocity errors	7-0	Indicates the number of blocks read where tape velocity attained full speed, but dropped during the read.

Word 7 Format

Field	Bits	Description
noise blocks discarded	11-4	Indicates the number of blocks read which were shorter than the minimum block length specified in the format unit functions (refer to section 2).
	3-0	Indicates the upper 4 bits of the number of track corrected blocks read.

Word 8

track corrected blocks read	11-0	Indicates the lower 12 bits of the number of track corrected blocks read.

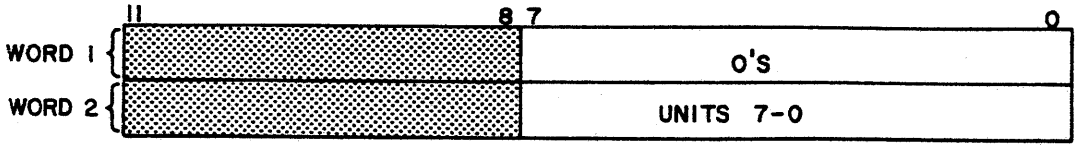
UNIT READY STATUS

Unit ready status provides the PPU with a low overhead way of determining the ready status of all units. Status is returned over the data channel in the form of a bit map. The bit map is made up of two words, the bits within the second word corresponding to logical unit numbers 7 through 0.

Issuing a unit ready status function every five to ten seconds gives the PPU the capability of monitoring operator action.

Previous unit connection, temporarily lost during execution of the unit ready status function, is restored after the last unit is tested.

The format for unit ready status is:



Bits 7 through 0 in word 2, corresponding to unit numbers 7 through 0, are specified with one of the following values.

0 Indicates that the unit is not available due to one or more of the following conditions.

- Unit is being rewound
- Unit is not ready
- Unit number is not dialed up
- Unit is reserved by the alternate MTS subsystem in a 2 x 8 configuration
- Unit was excluded from units ready status test via the last format TCU or units ready status function

1 Indicates that the unit is ready and available.

ERROR RECOVERY 4

This section describes recovery procedures for specific types of errors encountered using the 66X Magnetic Tape Subsystem. These generalized procedures must be coded by the user according to his unique system requirements. This section does not include recovery procedures for CYBER 170 channel parity errors (refer to appendix E).

READ RECOVERY

Recovery procedures for errors detected during read operations are outlined in the following subsections.

NOISE RECOVERY

Noise recovery is enabled for read operations by specifying a nonzero minimum block length in the format unit function parameter. When activated, noise recovery executes without PPU assistance for nondata tape motion operations.

Noise is detected if a block shorter than the minimum block length, specified in the format unit function, is read. When noise is encountered an inactive signal (CYBER 170/70 and 6000) or an end-of-record signal (CYBER 70 and 7600) is sent to the PPU and the noise field in general status (refer to General Status, section 3) is set. The MTS processor continues tape motion and waits for the PPU to execute noise recovery procedures.

An irrecoverable noise error occurs when the short block causing the error is too close to the next legitimate block. This results in inadequate time allowed for a repeat read function to be issued. A short access read error thus occurs and the tape stops at the next interblock gap without data being transmitted to the PPU.

Noise recovery requires execution of the following two operations.

1. Determination of whether or not noise was encountered by testing general status.
2. A read of the next block using the repeat read function.

These two operations must be repeated until a legitimate length block is read.

Execution of these operations requires procedural steps that differ slightly for the CYBER 170/70 and 6000 or CYBER 70 and 7600 systems. A summary of these steps for both the CYBER 170/70 and 6000 and CYBER 70 and 7600 systems appears in the following sections.

CYBER 170/70 AND 6000 NOISE RECOVERY PROCEDURES

To determine from general status whether or not noise was detected after completing the data transfer instruction:

1. Deactivate the channel if it is active. The channel will be active if the frame count requested was satisfied. The frame count could be satisfied by a short block if the number of frames requested is less than the minimum block size specified.
2. Issue a general status function (within 100 microseconds of step 1 if MTS deactivated channel) and wait for the channel to go inactive.
3. Activate the channel within 100 microseconds of step 2, read one byte, deactivate the channel, and test the noise bit.

If noise is detected, read the block as follows:

4. Reinitialize the conditions required to store data at the beginning of the buffer.
5. Issue the repeat read function within 200 microseconds of step 3 and wait for the channel to go inactive.
6. Activate the channel.
7. Execute the data transfer instruction.
8. Continue with step 1 to determine if a legitimate length block was read.

Failure to follow this procedure results in detailed status error code 60.

CYBER 70 AND 7600 NOISE RECOVERY PROCEDURES

To determine from general status whether or not noise was detected after completing the data transfer instruction:

1. Send an end-of-record signal if the frame count is satisfied.
2. Wait for the input word flag on the status channel.
3. Read one byte from the status channel and test for the noise bit.

If noise is indicated, read the block using the repeat read function as follows:

4. Reinitialize the conditions required to store data at the beginning of the buffer.
5. Issue the repeat read function within 200 microseconds of the input word flag in step 2.
6. Wait for the input word flag on the status channel (from the function echo).
7. Read one word from the status channel to clear the input word flag set by the echoed function code.
8. Wait for the output word flag on the function channel to be cleared by the function reply.
9. Execute the data transfer instruction.
10. Continue with step 1 to determine if a legitimate length block was read.

Failure to follow this procedure results in detailed status error code 60.

LOST DATA RECOVERY

A lost data error occurs when a read is not executed on the channel in time to receive data requiring transfer. Data transmission is terminated after a lost data error is detected.

Recovery from a lost data error involves the following steps.

1. Issue a backspace function for a forward read or a forespace function for a backward read.
2. Repeat the read operation. Include any noise recovery procedures if required.

READ PARITY ERROR RECOVERY

Parity error recovery procedures are required if the parity error bit in detailed status (refer to Detailed Status, section 3) is set. There are two recovery procedures depending on whether the read is forward or backward.

For a detailed description of automatic hardware data correction of correctable parity errors, refer to appendix D.

FORWARD READ RECOVERY PROCEDURES

1. Issue a backspace function.
2. † Issue a read function.

If the error is eliminated, exit with the recovered error.

If an error is still detected, perform steps 1 and 2 up to seven times. If the error persists, continue with step 3.

3. Issue a long backspace function.
4. Issue a reread forward function.

If the error is eliminated, exit with error corrected.

If the error continues, repeat steps 1 through 4 up to four more times. If the error is still detected, continue with step 5.

5. Repeat steps 1 through 4 up to five times issuing a low read clip function after steps 1 and 3.

If the error persists, continue with step 6.

6. Repeat steps 1 through 4 up to five times issuing a high read clip function after steps 1 and 3.

If the error persists, continue with step 7.

7. Repeat steps 1 through 4 up to five times issuing an increase read sprocket delay function after steps 1 and 3.

If the error persists, continue with step 8.

8. Repeat steps 1 through 4 up to five times, issuing a decreased read sprocket delay function and a high read clip function after steps 1 and 3.

If this fails to correct the error, an irrecoverable error condition must be indicated.

---

† For the second recovery attempt, issue an opposite parity mode function after step 1. The opposite parity mode function has no effect on a 9-track unit.

BACKWARD READ RECOVERY PROCEDURES

1. Issue a forespace function.
2. † Issue a read backward function, or if the odd count field in general status is set, issue a read backward with odd length parity function.

If the error is eliminated, exit with the recovered error.

If the error is still detected, perform steps 1 and 2 up to seven times. If the error persists, continue with step 3.

3. Issue a long forespace function.
4. Issue a reread backward function, or if the odd count field in general status is set, issue a reread backward with odd length parity function.

If the error is eliminated, exit with the error corrected.

If an error is still detected, perform steps 1 through 4 up to four more times. If the error is still detected, continue with step 5.

5. Repeat steps 1 through 4 up to five times issuing a low read clip function after steps 1 and 3.

If the error persists, continue with step 6.

6. Repeat steps 1 through 4 up to five times issuing a high read clip function after steps 1 and 3.

If the error persists, continue with step 7.

7. Repeat steps 1 through 4 up to five times issuing an increase read sprocket delay function after steps 1 and 3.

If the error persists, continue with step 8.

8. Repeat steps 1 through 4 up to five times issuing a decreased read sprocket delay function and a high read clip function after steps 1 and 3.

If this fails to correct the error, an irrecoverable error condition must be indicated.

† For the second recovery attempt, issue an opposite parity mode function after step 1. The opposite parity mode function has no effect on a 9-track tape unit.

WRITE RECOVERY

The following sections describe the types of errors encountered during write operations that require recovery, and the specific recovery procedures for these error conditions.

UNERASED NOISE RECOVERY

There is an unerased noise recovery procedure corresponding to one of the following conditions under which noise was detected.

1. Noise detected in the 6-inch interblock gap erased from an erase or write file mark function. This unerased noise results in the erase error field in detailed status (refer to Detailed Status, section 3) being set.
2. Noise detected in the second half of the interblock gap formed during execution of a write. This unerased noise results in the erase error field in detailed status being set.
3. Noise detected in the first half of the interblock gap is reported in the noise in gap field in detailed status.

Noise detected between loadpoint and the beginning of data during write operations is indicated with a fatal error code in detailed status. Noise recovery is not provided for this type of unerased noise.

Whenever tape is erased, the lowest read clip level is automatically selected for a read after write check. This is done to ensure that any remaining flux changes are safely below the clip level of subsequent read operations.

NOISE AFTER ERASE OR WRITE TAPEMARK FUNCTIONS

The recovery procedures for noise detected from faulty erase attempts by the erase or write tapemark functions are as follows:

1. Issue an erase reposition function or issue an erase reposition to erase function if noise was detected after execution of two consecutive erase functions.
2. Issue an erase or write tapemark function.

If the error persists, repeat steps 1 and 2 three times. An irrecoverable error condition should be indicated if the noise cannot be eliminated after the last recovery attempt.

NOISE AFTER WRITE FUNCTION

If noise was detected in the second half of the interblock gap during a write, use the write parity error procedures listed in this subsection for the block just written.

LOST DATA ERROR RECOVERY

Lost data error recovery is required when the next frame is to be written,but data for the write is not present on the data channel. The current write operation terminates when this error condition is detected. The error is indicated in the lost data field in detailed status.

Any data being transmitted by the CYBER 70 and 7600 after the error condition is detected is discarded by the MTS processor until the specified byte count is satisfied. In addition, the number of bytes discarded is not indicated by the MTS processor. However, general status is returned when the error occurs, rather than after the data is discarded.

If lost data is detected before any frames are written, a specific error code (appendix A) is returned to bits 5 through 0 in word 1 of detailed status (refer to Detailed Status, section 3).

There are two types of lost data error recovery procedures depending on whether or not the write operation was preceded by an erase function.

WRITE WITHOUT A PREVIOUS ERASE

The lost data error recovery procedures for a write operation without a previous erase are as follows:

1. Issue a write reposition function.

   The byte count to be specified with this function can be determined when using a CDC CYBER 170/70 and 6000 system by subtracting the starting byte count value located in the A register from the final value at the completion of the block transfer instruction. When using a CDC CYBER 70 and 7600 system, the byte count specified should equal the total byte count of all preceding block transfer operations used to write the block. Specify a byte count of zero if a single transfer was used.

2. Issue a write function.

If the error persists, repeat steps 1 and 2 as many times as desired. An irrecoverable error condition should be indicated if repeated recovery attempts fail to eliminate the error condition.

WRITE WITH A PREVIOUS ERASE

Use the write parity error recovery procedures if the write was preceded by an erase. The byte count specified for the write reposition function in these procedures must be reduced. This reduced byte count is determined when a CDC CYBER 170/70 and 6000 system is being used by subtracting the starting byte count value located in the A register from the final value at the completion of the block transfer instruction. When using a CDC CYBER 70 and 7600 system, the byte count specified should equal the total byte count of all preceding block transfer operations used to write the block. Specify a byte count of zero if a single transfer was used.

WRITE PARITY ERROR RECOVERY

Parity error recovery is required only if the parity error field in detailed status (refer to Detailed Status, section 3) is set, but is optional if the erase error field in detailed status is set.

During a write or write tapemark operation the high read clip level is automatically selected for the read after write check of the data written.

WRITE OR WRITE PRECEDED BY ERASE

The following procedures are required to recover from a parity error detected during a write operation.

1. Issue a write reposition function, or if the write was preceded by an erase, issue a write reposition to erase function.

   If a lost data error was indicated in addition to the parity error, the byte count specified with either the write reposition or write reposition to erase functions must reflect the size of the partial block. This byte count can be determined when using a CYBER 170/70 and 6000 system by subtracting the starting byte count value indicated in the A register from the final value at the completion of the block transfer instruction. When using CYBER 70 and 7600 systems, the byte count specified should equal the total byte count of all preceding block transfer operations used to write the block. Specify a byte count of zero if a single transfer was used.

2. Issue an erase function.
3. Issue a write function.

If a parity error is still detected, repeat steps 1 through 3 up to 20 times issuing the write reposition to erase function in step 1. The erased area must not exceed 25 feet.

To verify if the rewrite was successful, continue with step 4.

4. Issue a format unit function specifying a minimum block length of 31 frames. If blocks shorter than 31 frames are being written, the minimum block length specified can be increased only to the block length written or to the length of the previous block (specify minimum block length equal to the shorter of these two block lengths).
5. Issue a backspace function.

If a parity error is detected, continue write recovery with step 6.

Continue verification with step 7 if a parity error is not detected.

6. Issue a forespace function and repeat write recovery by continuing with step 1 using the write reposition to erase function in step 1.
7. Issue a backspace function.
8. Issue two consecutive forespace functions.

If after step 7 or step 8 the length of the largest noise block field in detailed status (refer to section 3, Detailed Status) is greater than half the original minimum block length specified, an irrecoverable write error should be indicated.

If no error condition was detected, verification is complete.

9. Issue a format unit function respecifying the original minimum block length.

WRITE TAPEMARK OR WRITE TAPEMARK PRECEDED BY ERASE

With the exception of substituting the following functions, recovery procedures for a parity error detected during a write tapemark operation are the same as those listed for a write.

1. Issue an erase reposition, or an erase reposition to erase function, if the write tapemark is preceded by an erase, in step 1.
2. Issue a write tapemark function in place of the write function in step 3.

AUTOLOAD/AUTODUMP PROCEDURES 5

The MTS software must be loaded and the processor activated before any tape operations can be performed. The procedures for autoloading MTS for the CDC CYBER 170/70 and 6000 and CDC CYBER 70 and 7600 system are listed in the following subsections.

MTS AUTOLOAD UNDER CDC CYBER 170/70 AND 6000

There are two autoload procedures for MTS depending on whether or not the MTS software is residing in the MTS processor at autoload time.

COLDSTART PROCEDURES

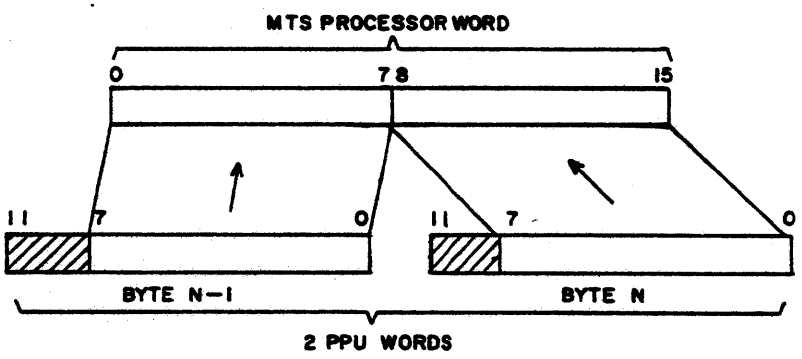
MTS software can be loaded into the MTS processor through a PPU or a maintenance console. The 7152 controller has a programmable read only memory (PROM), which allows loading controlware from a tape unit.

The CDC CYBER 170/70 and 6000 PPUs must be programmed by the user to perform the following steps if the MTS software is not presently residing in the MTS processor.

1. Issue the autoload function (function code 414) to enable the processor to receive the software.

   An inactive signal is returned by MTS hardware when the function is accepted.

2. Activate the channel and send all of the PPU words (maximum of 8192) that are in the software. These words are loaded using the following assembly for each word.



3. After the last PPU word is sent, the PPU must send an inactive signal to start the MTS processor which then initializes the subsystem.

If the MTS software is not available to the PPU, a maintenance console can be plugged into the MTS processor and the software loaded using paper tape.

LOADING MTS SOFTWARE INTO A 7152

To load MTS controlware via the PROM (7152 only), a routine must be entered on the deadstart panel. Refer to the routine under Warmstart Procedures and make two changes. Set locations 7 and 10 to 007U (load from logical unit U) and 0300 (loop), respectively. Pressing the deadstart switch initiates loading. The tape has 5 to 10 copies of the controlware block; therefore, pressing the deadstart switch more than once causes controlware to load if some of the blocks are damaged.

The following occurs in the MTS processor when the deadstart switch is pressed.

- A halt signal goes to the processor.
- A master clear signal goes to the processor.
- The MTS processor program address register is set to 1000_{16}.

The following sequence occurs when controlware loads from a tape unit.

1. When the deadstart switch is pressed, the loader program in the PROM enters the MTS processor and executes a function wait loop.

2. The loader program exits the wait loop when a function, which contains a 0 equipment code, is received. Bits 2^9 through 2^{11} of the 12-bit function word specify the equipment code. The following function codes are recognized.

   0X70 Load from logical unit 10†
   0X71 Load from logical unit 11
   .
   .
   .
   0X77 Load from logical unit 17

   The logical unit number is 10_8 through 17_8 to avoid accidental assignment to the wrong tape unit.

   If a function other than 0X7X is received, only steps 7 and 10 occur.

3. The tape unit specified by the function code is connected. This unit must be as follows:

   a. Present and powered on
   b. A 9-track unit
   c. Ready

4. If the tape is at loadpoint, density is automatically selected, depending upon the presence of a phase ID burst.

5. One controlware block is read into locations 0000 through $0FFD_{16}$ of the MTS processor memory. If a tape mark is read, the tape is rewound, and steps 4 and 5 are repeated.

6. Block length and data redundancy indicators are checked. Single track correction is allowed for 1600 CPI.

---

†X is a 3-bit unused field.

7. The MTS processor memory is checked to determine if it contains MB434 controlware. The following locations are checked.

    a. 0020 must contain 01FD.

    b. 0FFD must contain 434A.

    c. 0FFE must contain exclusive OR of words 000 through 0FFD if 0X7X function is being performed.

8. Tape is unloaded if load is successful.

9. A function reply is returned.

10. Execution enters standard controlware at location 0001.

For loading controlware into a 7152 from a maintenance console, use the following procedure.

1. Master clear the MTS processor.

2. Set P to 1018_{16}.

3. Set A to the desired logical unit number (10_8 through 17_8).

4. Press the GO switch. Controlware will load from the tape unit specified by A.

Execution begins at step 3 of the preceding PROM loading sequence.

WARMSTART PROCEDURES

If the MTS software is resident in MTS processor memory, the CDC CYBER 170/70 and 6000 operating systems can be deadstarted from a tape connected to the MTS processor through the warmstart read function. After it is up, the operating system must replace the MTS software to ensure compatibility.

The warmstart read function code performs the following.

Connects tape control unit (TCU) to the specified unit. Either the computer operator or the operating system must protect against problems that could arise if the warmstart read automatically clears opposite reserves. If the logical unit containing the deadstart tape is reserved by the alternate processor, the currently connected unit on the alternate side is also disconnected. This could result in a partially written block. If the unit is reserved, the function clears opposite reserves and returns to the main idle loop that restarts the warmstart read since no function

Selects specified density for a 7-track tape and automatically sets the density for a 9-track tape as it moves from the loadpoint

Selects odd parity and no code translation

Selects assembly/disassembly mode 0 for 7-track and mode 1 for 9-track

Selects TCU to peripheral processor data path

Deactivates the channel if connect is successful and the unit is ready

Rewinds the unit

Reads one block

Deactivates the channel regardless of errors

If the channel connecting MTS to the PPU is 0, 12_8, or 13_8, enter the following routine on the deadstart panel to initiate reading of the deadstart tape. Variations of this routine may be necessary, depending upon the operating system and hardware configuration.

Deadstart Panel Location	Deadstart Routine Instruction	Symbolic COMPASS Instruction	Description
1	75CC	DCN	Deactivate channel CC.
2	3615	AOD	Enter into location 15.
3	1014	SHN	Shift A to 10000_8.
4†	1701	SBN	Delay.
5†	0576	NJN	Delay.
6	77CC	FNC	Send warmstart function; specify logical unit U.††
7	026U		
10	74CC	ACN	Activate channel CC.
11	71CC	IAM	Input data from channel CC to PPU memory beginning at address XXXX.
12	XXXX		

If the channel connecting MTS to the PPU is 1 through 11_8, additional instructions determined by the user must be entered on the deadstart panel to stop the corresponding PPU from reading on the channel. The deadstart signal neither stops the MTS processor nor clears the TCU.

†If channel CC is connected to a 6681 or a 6684, locations 4 and 5 must be changed to 77CC and 2100 (deselect 6681 or 6684), respectively.
††U can be 10_8 through 17_8.

CHANNEL PARITY ERROR DETECTION DURING AUTOLOAD UNDER CYBER 170

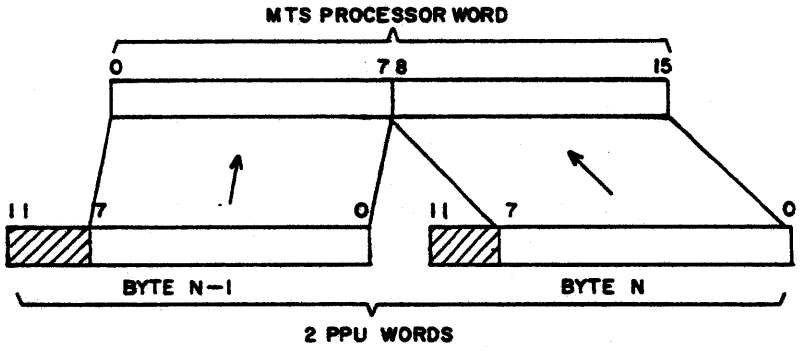
When autoloading under CYBER 170, the MTS processor will not initialize the subsystem if it detects channel parity error. Only 7021-21/22 hardware detects channel parity errors during autoload. Therefore, if the MTS processor does not immediately reply to functions after autoload, the operator must press STOP and MC on the controller to clear the channel parity error flip-flop in the coupler, and then reautoload.

MTS AUTOLOAD UNDER CDC CYBER 70 AND 7600

Autoload procedures for MTS under CDC CYBER 70 and 7600 do not require that MTS software is resident in MTS processor memory at autoload time. The PPU must be programmed by the user to perform the following steps to autoload MTS software.

1. Issue the autoload function (function code 14, subfunction code 4) to enable the processor to receive the MTS software. A resume signal is returned by MTS hardware when the function is accepted.

2. Activate the channel and send all of the PPU words (maximum of 8192) that are in the software. These 8192 words are loaded using the following assembly for each word.



3. Send a record signal on the data channel.

The MTS processor initiates the subsystem after autoloading is complete.

MTS AUTODUMP

Autodumping MTS processor memory to PPU memory requires that the following autodump routine residing in the PPU is autoloaded into the first 11 locations of MTS processor memory.

Autodump Routine Residing in PPU (Octal)	Symbolic BUCAL Instruction	MTS Processor Memory Location	Autodump Routine (Hex) Residing in MTS Processor	Description
0000 0000	CON FWA	0000	0000	Indicates the first word address of MTS processor memory to be dumped.
0011 0000	TCB 0,0	0001	0900	
0357 0001	FJR $-1	0002	EF01	Wait for function.
†0014 0060	IAN 3	0003	0C30	Get function code.
0017 0000	OAA 0	0004	0F00	Use function code as assembly/disassembly format.
0012 0000	SCB 0,0	0005	0A00	
0013 0000	CCB 0,0	0006	0B00	Send function reply.
0030 0001	LDN 1	0007	1801	Infinite word count.
0011 0006	TCB 0,6	0008	0906	
0357 0001	FJR $-1	0009	EF01	Wait for coupler ready.
0374 0000	0TI 0	000A	FC00	Dump memory from address specified in location 0000.

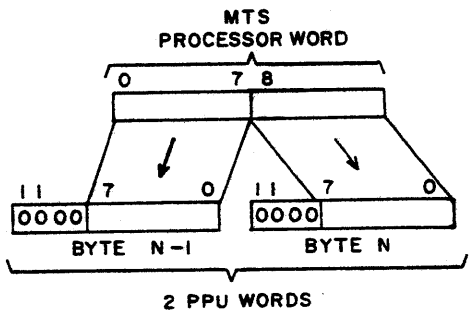
The PPU must be programmed by the user to perform the following operations for the autodump.

1. Autoload MTS processor memory with the autodump routine. Send exactly 22_{10} PPU words.
2. Send one of the following function codes to indicate the assembly/disassembly format to be used to dump MTS processor memory.

† The line listed above is for an autodump using CDC CYBER 170/70 and 6000 systems. For a CDC CYBER 70 and 7600 the same line is:

0014	0040	IAN 2	0003	0C20

Function Code	Assembly/Disassembly Format
01	Each MTS processor word is returned as two PPU bytes.



03	Three MTS processor words are returned as four PPU bytes described as mode 1 assembly/disassembly in appendix B.

3. Wait for a function reply.

4. Repeat execution of a block input instruction until the desired amount of memory is dumped.

5. Send an inactive (CDC CYBER 170/70 and 6000) or a record flag (CDC CYBER 70 and 7600) signal to terminate the MTS processor's output instruction.

The MTS processor then enters the autoload initialization routine which restores location 0001 and begins executing the function idle loop. The MTS processor does not require an autoload after executing an autodump.

ERROR CODES A

The error codes detected are returned to the error code field of detailed status to indicate specific error conditions. All error codes are shown in octal.

These error codes are categorized as follows:

Range	Description
01-07	Error codes in this range indicate that the unit requires operator attention.
10-17	Error codes in this range indicate that the tape is unusable for this operation.
30-37	Error codes in this range indicate that an illegal procedure was performed due to an error in the user's routine.
40-47	Error codes in this range indicate that the tape unit is inoperable and the customer engineer should be notified.
50-57	Error codes in this range indicate that an illegal procedure not involving tape motion was performed by PPU software.
60-67	Error codes in this range indicate that an illegal procedure involving tape motion was performed by PPU software.

Error Code	Description	Action
01	A connect was rejected because no transport was dialed to the requested unit number.	Issue an operator message.
02	A connect was rejected because the unit was already reserved by the alternate MTS subsystem in a 2 x 8 configuration.	Issue an operator message.
04	The function was rejected because the unit was not ready.	Issue an operator message.
05	Unit was declared not ready during last operation.	Backspace (forespace if last motion was backward) one block to make position match what is reported in the tape position field.
06	A write was not executed due to a missing write ring.	Request the operator to supply a write ring.
07	Unused	

Error Code	Description	Action
10†	More than 25 feet of blank tape (with or without noise blocks) was encountered; if EOT is encountered under forward motion, the limit beyond EOT is 25.6 inches.	Abandon further efforts to read the tape.
11	The stop motion function was executed. The alert bit is not set if operating under CDC CYBER 70/6000.	
12	Unable to write from loadpoint due to bad tape.	Request the operator to mount a new reel.
13†	(CDC CYBER 76 and 7600 only) False read end of operation occurred and read gap could not be located within 100 inches. (CDC CYBER 170/70 and 6000 only) False phase I.D. burst detected.	Abandon further efforts to read the tape.
14	Short read-access time preventing data from being sent to the PPU (see also other Error Code 14 description).	Backspace (forespace if read is backward) and reread.
14	If from erase reposition (0016) or write reposition using controlled backspace (0217) function, the function is terminated and positioning is uncertain.	Declare irrecoverable write error (see also description of these functions).
15	Permanent erase error; unable to form a reliable interblock gap within 30 inches.	Abandon further efforts to write the tape.
16-19	Unused	
20	MB434 controlware was not loaded. Either the wrong controlware tape was used, or a warmstart was attempted without controlware being loaded.	Check controlware tape.
21	A 7-track unit was specified. Controlware must be loaded from a 9-track tape.	Specify 9-track tape.
22	Controlware block was too short. MB434-D controlware consists of 4095 16-bit words. This requires 8190 8-bit frames on the tape.	Check controlware tape format.
23	Parity error was detected in controlware block. Attempting to read a 7-track tape may also cause this error (see Error Code 21).	Retry the deadstart.
24	Tape unit failed to rewind.	Check for problem in controller or tape unit hardware.
25-27	Unused	

† To prevent mispositioning if data is present immediately after the length of blank tape detected has executed the limit, a 1-inch controlled backspace (forespace if backward operation) is performed before this error code is reported.

Error Code	Description	Action
30	Backword motion or controlled forespace function was attempted at the loadpoint marker.	Return error code to the operating system.
31	Unused	
32	Motion was attempted on a rewinding unit.	Wait until unit is not busy.
33	Nonwrite forward motion was attempted when the previous operation was a write.	Return error code to the operating system.
34	A controlled backspace, erase reposition or write reposition function was attempted when the previous direction was backward.	A system error or a garbled function code should be suspected cause of error.
35	A controlled forespace or backspace, erase reposition, or write reposition function was attempted at a recording density of 200-cpi.	A system error or a garbled function code should be suspected cause of error.
36	A write was attempted at a recording density of 200 cpi.	Return error code to the operating system.
37	Unused	
40	Write permit status failed to occur within 34 milliseconds.	Logically remove unit from system.
41	Velocity failed to reach 93 percent of rated speed within 34 milliseconds.	Logically remove unit from system.
42	The unit failed to move the tape when tape movement was requested.	Logically remove the unit from the system.
43	The erase or write current failed to occur when a write was requested.	Logically remove the unit from the system.
44	Busy status was still indicated within 34 milliseconds after the stop signal was sent to the unit.	Logically remove the unit from the system.
45	Loadpoint status was still indicated 34 milliseconds after a forward signal was sent to the unit.	Logically remove the unit from the system.
46	A block was written, but no frames were detected in the read after write.	Logically remove the unit from the system.
47	Miscellaneous unit error.	Logically remove the unit from the system.
50	Unrecognizable function code.	Reissue the function.

Error Code	Description	Action
51	The function was rejected because the unit was logically but not physically connected.	The operator changing the unit number should be the suspected cause of error.
52	The function required one or more words of parameters that were not issued within 1 millisecond of the function.	Reissue function and required parameter(s). If error still persists a coupler hardware failure should be suspected cause of error.
53	The function required more parameter words than were issued.	Reissue function and the correct number of parameter words. If error still persists, a coupler hardware failure should be suspected cause of error.
54	Channel parity error (parameters or data)	Refer to appendix E.
55	Channel parity error during function transmission	Refer to appendix E.
56	An illegal value was specified in the format unit function parameters. The error is not detected until the following motion-type function. See section 2 for allowable parameter values.	A system error or garbled data should be suspected cause of error.
57	An illegal value was specified in a parameter required by a function other than the format unit function.	A system error or garbled data should be suspected cause of error.
60	The noise recovery procedure used did not satisfy MTS controlware requirements. Refer to noise recovery procedures in section 4.	A system error should be suspected cause of error.
61	Over 34 milliseconds elapsed before data was available to be written.	Perform lost data recovery procedure.
62	PPU was not ready to receive data 1.4 milliseconds after a read function. This error is not detected by the CDC CYBER 70 and 7600 version	Perform lost data recovery procedure.
66	The function was rejected because the unit was not logically connected.	A software error (attempting to use a unit that was either released or never connected) should be the suspected cause of error.

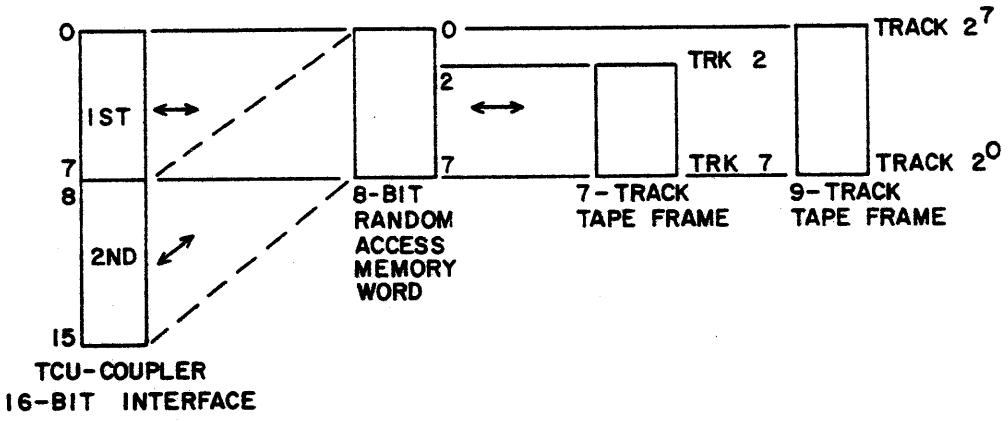
ASSEMBLY/DISASSEMBLY

B

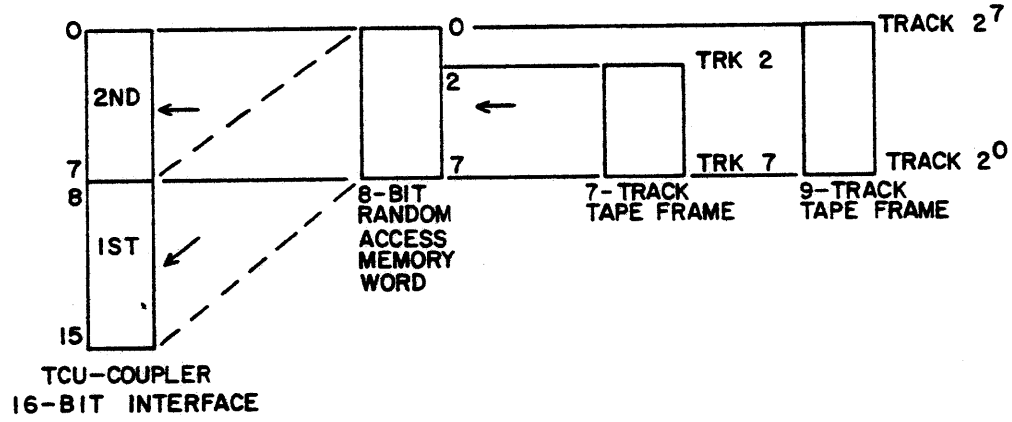
Assembly/disassembly is performed in both the coupler and tape control unit. A description of assembly/disassembly modes for both the tape control unit and the coupler is presented in the following sections.

TAPE CONTROL UNIT ASSEMBLY/DISASSEMBLY

The tape control unit uses two assembly/disassembly formats corresponding to 7-track or 9-track tape. Assembly/disassembly for a write or forward read is as follows:



Assembly/disassembly for a reverse read is represented as follows:



READ OPERATIONS (INPUT)

During read operation, frames are assembled as two frames per 16-bit word. The first frame of a forward read appears in the upper half (bits 0 through 7) of the word. A reverse read starts in the lower half (bits 8 through 15) of the word. If an odd number of frames is read, the lower half (upper half for a reverse read) of the last word is zero filled. When reading an odd length phase-encoded block, the unused half of the last word is filled with a value corresponding to an all 1's character from tape. The odd count field in general status (refer to General Status, section 3) is set.

WRITE OPERATIONS (OUTPUT)

Two frames are disassembled from every 16-bit word. The first frame appears in the upper half (bits 0 through 7) of the word. If odd count is specified, only one frame is written from the upper half of the last word.

CODE TRANSLATION

When a specific code translation table has been specified in the format unit statement, an 8 to 8-bit mapping is performed in both random access memories. If code translation is not specified a unity mapping is performed. When reading a 7-track tape, bits 0 and 1 sent to the read random access memory are zero. This results in referencing only the first 64 entries. Bit 1 of the read random access memory is also used to set the illegal character status field in detailed status. All 8 bits of the read random access memory are sent to the coupler. However, assembly/disassembly mode 0 discards the upper two bits.

COUPLER ASSEMBLY/DISASSEMBLY

The coupler provides for three types of assembly/disassembly modes. These three modes are described in the following subsections as they relate to read and write operations. To assist in designing PPU drivers for MTS, frame count tables are provided for the three modes.

The operating system must be informed of the number of valid bits in the last central memory word for read operations. The frame count tables for reads indicate how the word count and odd count status can be converted to the number of fill bits in the last word transmitted by MTS. When a read is terminated by the PPU before the entire block is transferred, there are no fill bits and the odd count field in general status should be ignored.

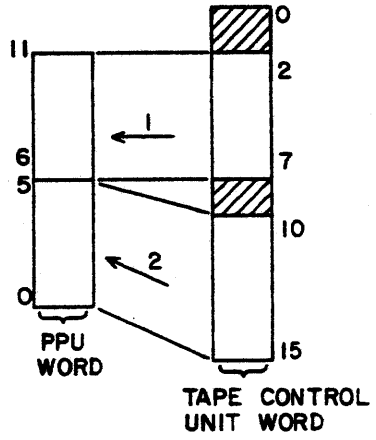
For write operations, the block size is requested by the operating system in terms of the number of central memory words plus the number of additional bits. The CDC CYBER 170/70 and 6000 and CDC CYBER 70 and 7600 systems limit the number of additional bits to a multiple of 12. The frame count tables indicate how the word and bit counts can be converted to a PPU word count and odd count option as required by MTS. However, any size block can be written by specifying when the block is to contain an odd number of frames.

MODE 0 (6-BIT MODE) ASSEMBLY/DISASSEMBLY

Mode 0 assembly/disassembly for read and write operations is described in the following section.

Mode 0 Read

The mode 0 assembly/disassembly format for read operations is:



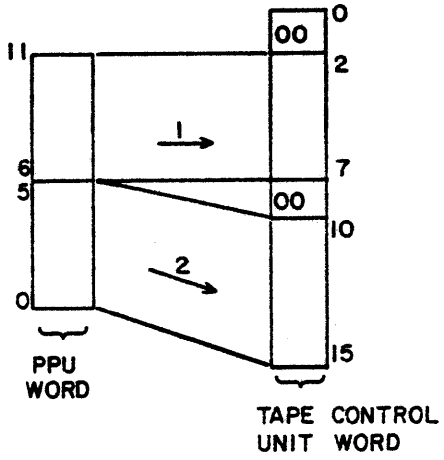
The following table is used to determine the number of frames read and whether fill bits were required for the last word read.

Given		Result	
PPU Words	Odd Count (General Status)	Frames Read	Bits 5-0 † 0-filled
1	1	1	x
	0	2	
2	1	3	x
	0	4	
3	1	5	x
	0	6	
4	1	7	x
	0	8	
5	1	9	x
	0	10	

†For reverse reads, bits 11 through 6 are zero filled.

Mode 0 Write

The mode 0 assembly/disassembly format for write operations is:



The following table defines how many frames are written for all combinations of PPU words transmitted for both even and odd write functions.

PPU Words	Write Function	Frames Written
1	Odd	1
	Even	2
2	Odd	3
	Even	4

1 PPU word = 2 frames

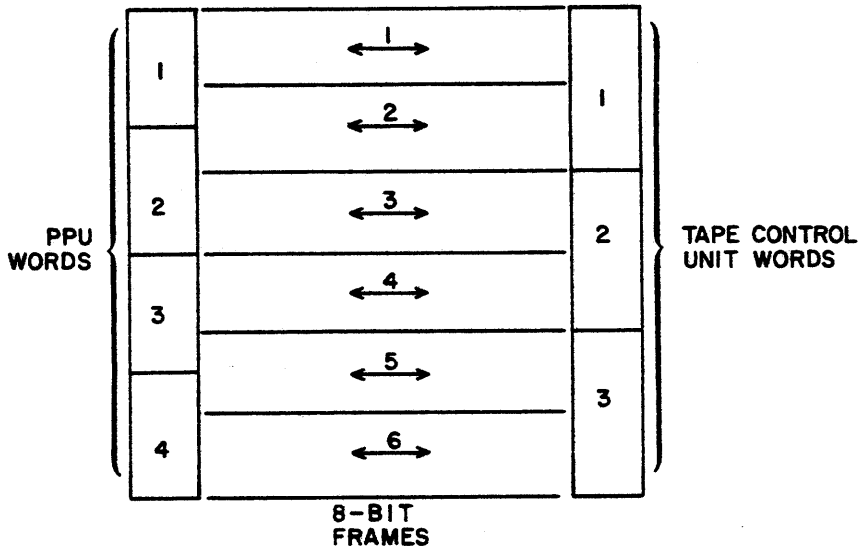
Modular Relationship

This table describes the number of PPU words that must be transmitted for a required number of PPU words to be written on tape. The write odd length function is not used.

Given		Result	
CM Words	Additional Bits	PPU Words	Required Frame Count
0	12	1	2
0	24	2	4
0	36	3	6
0	48	4	8
1	0	5	10
1	12	6	12
1	24	7	14
1	36	8	16
1	48	9	18
2	0	10	20

MODE 1 (PACKED MODE) ASSEMBLY/DISASSEMBLY

The mode 1 assembly/disassembly format for both read and write operations is:



Mode 1 Forward Read

The following table is used to determine the number of frames read and whether fill bits were required for the last word read.

Given					Result		
PPU Words	Odd Count Field (General Status)	Frames Read	2xPPU Words Modulo 4	Logical Ordinal	Adjusted PPU Words	Bits 3-0 0-filled	Bits 7-0 0-filled
1	1	Cannot occur	2	2			
	0	Cannot occur	2	3			
2	1	1	4	4	1	x	
	0	2	4	5	2		x
3	1	3	6	6	2		
	0	4	6	7	3	x	
4	1	5	0	0	4		x
	0	6	0	1	4		

Logical ordinal - Calculated by using one of the following formulas.

- If odd count field in general status is 1:
  2·(PPU words modulo 4)
- If odd count field in general status is 0:
  2·(PPU words modulo 4)+ 1

Fill calculation - If the logical ordinal is:

- 4 or 6, reduce the PPU word count by 1.
- 4 or 7, bits 3 through 0 are zero filled.
- 0 or 5, bits 7 through 0 are zero filled.

Mode 1 Write

The following table describes how many frames are written for all combinations of PPU words transmitted for both even and odd write functions.

PPU Words	Write Function	Frames Written
1	Odd	1
	Even	0†
2	Odd	3
	Even	2
3	Odd	5†
	Even	4
4	Odd	5
	Even	6

4 PPU words = 6 frames

Modular Relationship

†Not required for writing an arbitrary number of frames.

This table describes the number of PPU words that must be transmitted for a required number of PPU words to be written on tape and whether the write or write odd length function is used.

Given				Result			
CM Words	Additional Bits	PPU Words	PPU Words Modulo 4	PPU Words to Write	Write Function	Frames Written	Bits 3-0 0-filled
0	12	1	1	2	Even	2	x
0	24	2	2	2	Odd	3	
0	36	3	3	3	Odd	5	x
0	48	4	0	4	Even	6	
1	0	5	1	6	Even	8	x
1	12	6	2	6	Odd	9	
1	24	7	3	7	Odd	11	x
1	36	8	0	8	Even	12	
1	48	9	1	10	Even	14	x
2	0	10	2	10	Odd	15	
2	12	11	3	11	Odd	17	x
2	24	12	0	12	Even	18	
2	36	13	1	14	Even	20	x
2	48	14	2	14	Odd	21	
3	0	15	3	15	Odd	23	x
3	12	16	0	16	Even	24	
3	24	17	1	18	Even	26	x
3	36	18	2	18	Odd	27	
3	48	19	3	19	Odd	29	x
4	0	20	0	20	Even	30	

The number of frames written was selected as the minimum number of frames required to record all bits in the required word count field. Bits 2^3 through 2^0 of the last frame written are zero filled if the mod 4 of PPU words column is 1 or 3.

An extra PPU word must be transmitted if the mod 4 of PPU words column is 1. The word is discarded by the hardware and can contain any value. Use the write odd count function if the mod 4 of PPU words column is 2 or 3.

Reverse Read

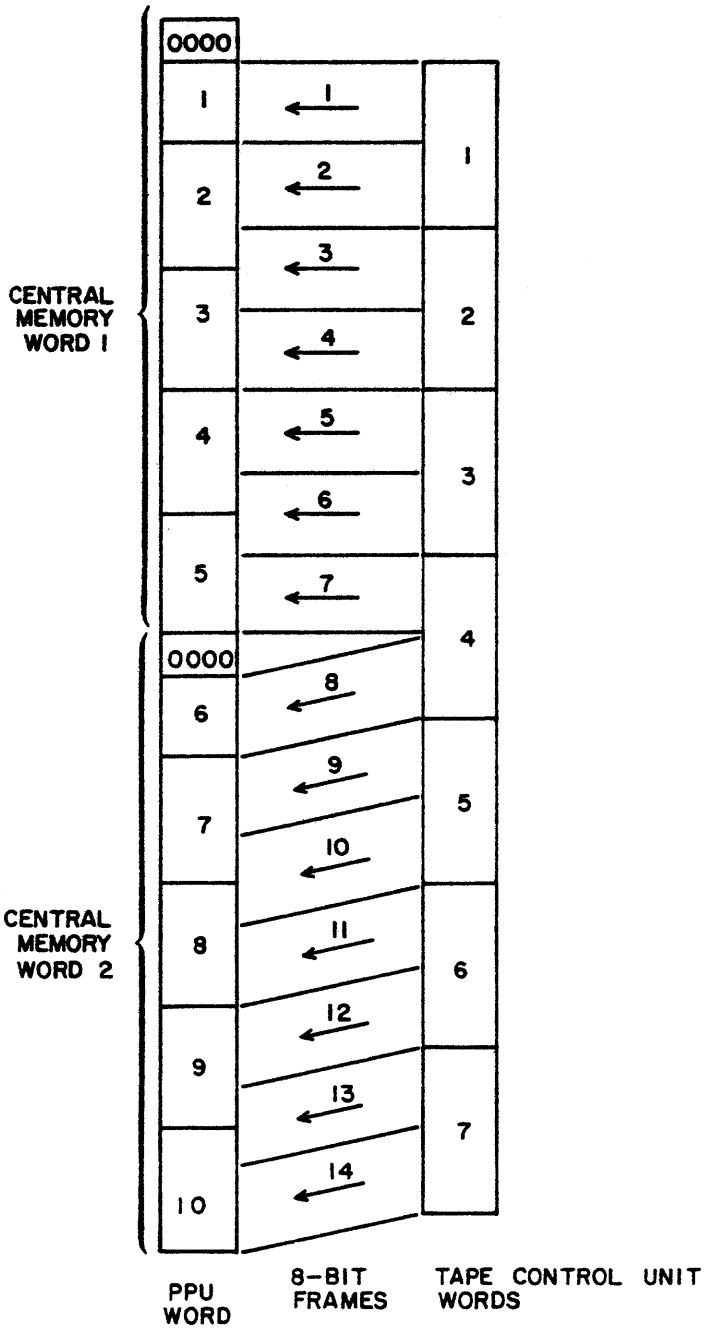
Reverse reads cannot be performed using packed mode because the assembly/disassembly performed by the coupler is not modified by motion direction.

MODE 2 (CPU MODE) ASSEMBLY/DISASSEMBLY

Mode 2 assembly/disassembly for read and write operations is described in the following sections.

Mode 2 Forward Read

The mode 2 assembly/disassembly format for read operations is:



The following table is used to determine the number of frames read and whether fill bits were required for the last word read.

Given						Result		
CM Words	PPU Words Remainder	Odd Count (General Status)	Frames Read	2xPPU Words Remainder	Logical Ordinal	Adjusted PPU Words	Bits 3-0 0-filled	Bits 3-0 0-filled
0	1	1 0	† †	2 2	2 3	† †		
0	2	1 0	1 2	4 4	4 5	1 2	x	
0	3	1 0	3 4	6 6	6 7	3 3		x
0	4	1 0	† †	8 8	8 9	† †		
0	5	1 0	5 6	10 10	10 11	4 5	x	x
1	1	1 0	7 8	2 2	2 3	0 1		
1	2	1 0	† †	4 4	4 5	† †		
1	3	1 0	9 10	6 6	6 7	2 3	x	x
1	4	1 0	11 12	8 8	8 9	3 4	x	
1	5	1 0	13 14	10 10	10 11	5 5		x

†Combination cannot occur.

Logical ordinal - Calculated by using one of the following formulas:

- If the odd count field in general status is 1:
  2•(PPU words modulo 4)
- If the odd count field in general status is 0:
  2•(PPU words modulo 4)+ 1

Fill calculation - If there is an even number of CM (central memory) words and the logical ordinal is:

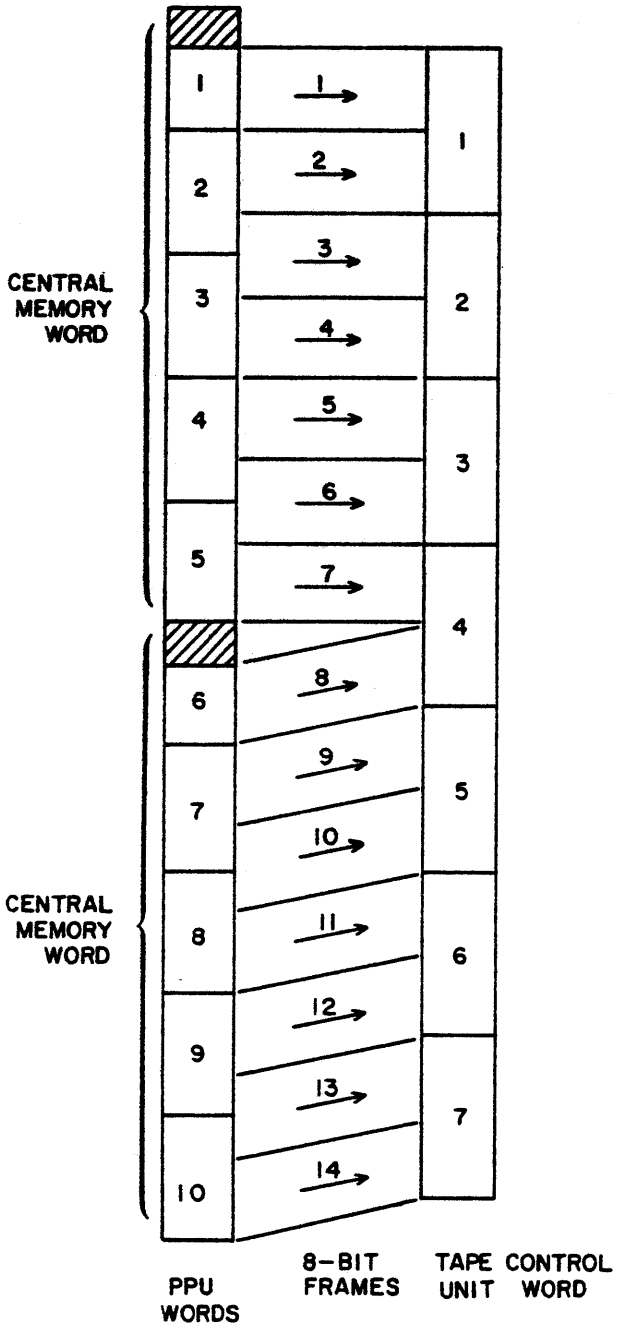
- 4 or 10, reduce the PPU word remaining by 1.
- 5 or 10, bits 3 through 0 are zero filled.
- 6 or 11, bits 7 through 0 are zero filled.

If there is an odd number of CM (central memory) words and the logical ordinal is:

- 2, 6, or 8, reduce the PPU words remaining by 1.
- 6 or 9, bits 3 through 0 are zero filled.
- 7 or 10, bits 7 through 0 are zero filled.

Mode 2 Write

The mode 2 assembly/disassembly format for write operations is:



The following table describes how many frames are written for all combinations of PPU words transmitted for both even and odd write functions.

PPU Words	Write Function	Frames Written
1	Odd	1
	Even	0†
2	Odd	3†
	Even	2
3	Odd	3
	Even	4
4	Odd	5
	Even	4†
5	Odd	7
	Even	6
6	Odd	7†
	Even	8
7	Odd	9
	Even	8†
8	Odd	11
	Even	10
9	Odd	13†
	Even	12
10	Odd	13
	Even	14

10 PPU words = 14 frames

Modular Relationship

†Not required for writing an arbitrary number of frames.

This table describes the number of PPU words that must be transmitted for a required number of words to be written on tape and whether the write or write odd length function is used.

Required			Result			
CM Words	Additional Bits	Additional PPU Words	PPU Words to Write	Write Function	Frames Written	Bits 3-0 0-filled
2n	0	0	10n	Even	14n	
2n	12	1	10n+1	Odd	14n+1	
2n	24	2	10n+2	Odd	14n+3	x
2n	36	3	10n+3	Even	14n+4	
2n	48	4	10n+5	Even	14n+6	x
2n+1	0	0	10n+5	Odd	14n+7	
2n+1	12	1	10n+6	Even	14n+8	
2n+1	24	2	10n+8	Even	14n+10	x
2n+1	36	3	10n+8	Odd	14n+11	
2n+1	48	4	10n+9	Odd	14n+13	x

The number of frames written was selected as the minimum number of frames required to record all bits in the required word count. Bits 2^3 through 2^0 of the last frame written are zero filled if the additional PPU words column is 2 or 4.

An extra byte must be transmitted if:

1. The CM words column is 2n and the additional PPU words column is 4.
2. The CM words column is 2n+1 and the additional PPU words column is 2.

Use the write odd length function if:

1. The CM words column is 2n and the additional PPU words column is 1 or 2.
2. The CM words column is 2n+1 and the additional PPU words column is 0, 3, or 4.

Reverse Read

Reverse reads cannot be performed using CPU mode because the assembly/disassembly performed by the coupler is not modified by motion direction.

DATA CONVERSION C

Data can be converted to and from specific tape formats using code translation tables and/or the assembly/disassembly modes described in appendix B.

There is space reserved in the MTS processor memory for three code translation tables. The MTS controlware contains default character sets preloaded (using the 63-character set) with the three tables. A summary of these tables is as follows:

TABLE C-1. CODE TRANSLATION TABLES

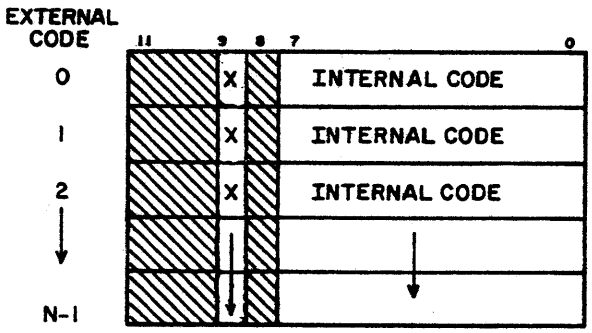
Table Number	Length (characters)	Default Character Set
1	256	ASCII to Display
2	256	EBCDIC to Display
3	64	Internal BCD or External BCD to Display

In addition to the code translation offered by the three code translation tables, a fourth kind of code translation can be used by specifying a 7 in word 1, bits 10 through 8 of the format unit function. When 7 is specified, the MTS processor synthesizes internal and external code values as it loads the read and write random access memories. No code translation table is used. The resulting code translation depends on the type of tape unit (7-track or 9-track) and the parity mode being used. A summary of the possible kinds of code translation available given specific tape unit types and parity mode is as follows:

TABLE C-2. 7-OPTION CODE TRANSLATION

Unit Type	Parity Mode	Code Translation
7-track	Odd	None
7-track	Even	External BCD to Internal BCD
9-track	Odd	Code translation using the current contents of the read or write random access memories. The random access memory used is not loaded.
9-track	Even	Illegal

The user can replace any of the three default code translation tables with character sets (standard or of the users design) initially constructed by the user and loaded into PPU memory. These character sets must be loaded into PPU memory in the following format.



external code — Indicates an ordinal value corresponding to the position of a specific internal code. These external code values are what physically appear on the tape for the corresponding internal codes.

The external codes for tables 1 and 2 range from 0 through 255. For table 3 the range of values is 0 through 63.

x — Bit 9 is used to indicate the order in which an entry is to be loaded into the read and write random access memories.

- 0 — The entry is to be loaded in the same order as it exists in PPU memory.
- 1 — To ensure that the specified internal code generates the corresponding external code when writing, this bit must be set. The entry is loaded in reverse order after all entries with bit 9=0 are loaded.

NOTE

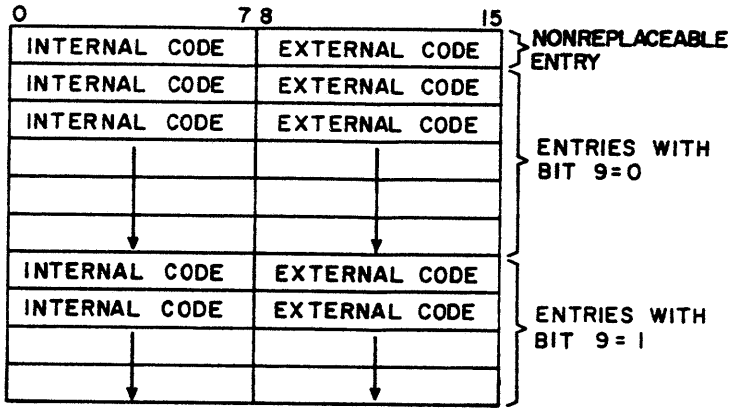
Bit 9 may be 0 for 1 code translation tables 1 and 2. Bit 9 must be 0 for code translation table 3, although the MTS processor does not test for, nor report the absence of, this condition.

internal code — Indicates the internal code values as they appear in memory.

Bits 7 and 6 are zero for 7-track tapes.

For 9-track tapes written with the 64-character subset, bits 7 and 6 are normally zero. Bit 6 can be used as a flag to indicate an illegal character for status information.

Replacing the default character sets residing in MTS processor memory with new character sets located in processor memory is accomplished by issuing the code translation table to processor memory function (refer to section 2). The code translation table when transferred and located in processor memory is formatted as follows:



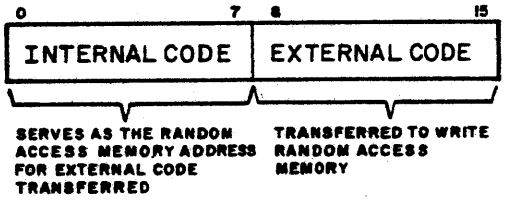
A preloaded nonreplaceable entry is appended to the character set as the first entry to be loaded into both the read and write random access memories. This nonreplaceable entry provides for up to 257 entries. It permits the same external code to be written for two internal codes. The following nonreplaceable entries are associated with each table.

Table	Nonreplaceable Entry (Hexadecimal Values)
1	0 7 8 15 — 00 | 20
2	0 7 8 15 — 00 | 40
3	0 7 8 15 — 00 | 00

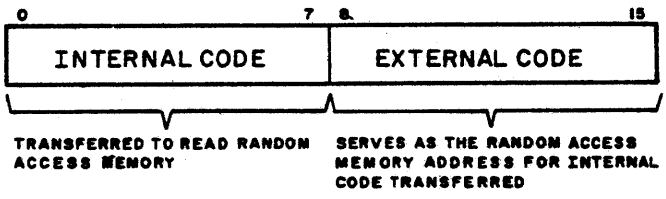
During read and write operations the code translation table specified in the format unit statement for a specific read or write operation is transferred by the MTS processor from its memory into the read and write random access memories of the tape control unit.

As the code translation table is moved into both random access memories each 16-bit word of the table in MTS memory is used to simultaneously load an 8-bit entry in both the write and read random access memories. Up to 256 entries can be loaded. As the RAMs are loaded, table 3 is converted by the MTS processor to the external BCD character set if even vertical parity is being used.

Loading the write random access memory involves transferring the external code stored in bits 8 through 15 from each word of the code translation table in MTS memory. Bits 0 through 7 of each word in the table serves as the random access memory address for the external codes transferred. When writing in assembly/disassembly mode 0 (appendix B) only the first 64 entries of the write random access memory are used.



The read random access memory is loaded by transferring the internal code stored in bits 0 through 7 from each word of the code translation table in MTS memory. Bits 8 through 15 of each word in the table serve as the random access memory address for the internal codes transferred. When reading 7-track tape, only the first 64 entries of the read random access memory are used.



The following tables summarize the types of 7- and 9-track data conversion and the corresponding assembly/disassembly modes, parity, and code translation option number (word 1, bits 10- to 8-format unit function) if required, for each type of data conversion.

7-TRACK TAPE

DATA REPRESENTATION		PARITY	ASSEMBLY / DISASSEMBLY MODE (APPENDIX B)	CODE TRANSLATION OPTION NUMBER
PPU	TAPE			
DISPLAY CODE	INTERNAL BCD CODE	ODD	0	1,2,3
DISPLAY CODE	EXTERNAL BCD	EVEN	0	3
INTERNAL BCD	EXTERNAL BCD	EVEN	0	7
CORE IMAGE	2 FRAMES PER PPU WORD	ODD	0	0 OR 7

9-TRACK TAPE

DATA REPRESENTATION		PARITY	ASSEMBLY / DISASSEMBLY MODE (APPENDIX B)	CODE TRANSLATION OPTION NUMBER
PPU	TAPE			
DISPLAY CODE	ASCII (64-CHAR. SUBSET)	ODD	0	1
DISPLAY CODE	EBCDIC(64-CHAR. SUBSET)	ODD	0	2
PPU CORE IMAGE	3 FRAMES PER 2 PPU WORDS	ODD	1	0
CPU CORE IMAGE (UPPER 4 BITS OF EACH WORD ARE UNUSED)	7 FRAMES PER CPU WORD	ODD	2	0

CORRECTABLE PARITY ERRORS D

Data correction is available only if the detected error is limited to one track of a 9-track tape. Hardware then reconstructs data for this single track from the eight valid tracks.

There are two ways data correction is initiated depending on whether 800 cpi (NRZI) or 1600 cpi (phase) density is being used.

800 CPI (NRZI) DENSITY

At the completion of a read operation the cyclic redundancy check register and the error pattern register are checked by the MTS processor to determine if an error was limited to one track and which track was affected. The track in error number is saved in MTS processor memory by any of the following functions for automatic correction the next time one of these functions is executed on this tape unit. If a 2 x 8 configuration is being used, corrective reread must be performed on the side of the configuration last using the unit.

- Read forward
- Read backward
- Read backward with odd length parity
- Reread forward
- Reread backward
- Reread backward with odd length parity

If a vertical parity error is detected during the corrective reread, the track in error field and one of the bits in the tracks corrected field in detailed status (refer to section 3) are set.

Data for the track in error is generated by the values required for correct vertical parity. If no vertical parity error is detected, corrective recovery is not necessary and the track in error field is zero.

Corrective recovery is available for a backward read of odd length blocks in NRZI, only if the read is performed by issuing the read backward with odd length parity function. This function instructs the hardware to require even parity for the cyclic redundancy check character allowing for correct identification of the track in error.

1600 CPI (PHASE) DENSITY

Corrective recovery for 1600 cpi 9-track tape is performed as data is being transferred. Any of the nine tracks which loses synchronization, where a full cycle of flux reversals does not occur within 1.5 cell times, is corrected. No special recovery procedures are required for a backward read since the phase preamble and postamble are symmetrical.

The parity error status field in detailed status is zero when the correction is successful, indicating that no further recovery is required.

If more than one track loses synchronization, the multitrack error field in detailed status is set requiring that the parity error recovery procedures listed in section 4 be executed.

CDC CYBER 170 CHANNEL PARITY ERROR RECOVERY E

This appendix describes the four types of channel parity error recovery procedures for the CYBER 170 series.

It is the operating system programmer's resonsibility to code these procedures as required by the system requirements.

FUNCTION ERROR FROM PPU TO MTS PROCESSOR

The MTS controller performs the following procedure when the coupler detects a channel parity error on a function code from the PPU.

The MTS processor:

1. Sets the general status alert bit.
2. Sets the detailed status error code 55_8.
3. Clears the function bit in the coupler.
4. Clears the channel parity error bit in the coupler.

The coupler:

1. Does not transmit an inactive (acceptance) reply to the PPU.

The PPU should:

1. Time out the functions to avoid hanging the channel.
2. Disconnect the channel.
3. Request general and/or detailed status.
4. Resend the function several times before abandoning the operation.

PPU TRANSFER OF PARAMETERS TO MTS PROCESSOR

The following procedure is performed when a channel parity error occurs on PPU transfer of parameters to the MTS processor.

The PPU should:

1. Complete the transfer.
2. Send a general status function (0012_8). The MTS processor does not reply to any other function in this case.

The MTS processor:

1. Clears the channel parity error bit in the coupler.
2. Sets the general status alert bit.
3. Sets the detailed status error code 54_8 indicating a channel parity error occurred during transfer of parameters.

The PPU should:

Resent the parameters several times before abandoning the operation.

PPU OUTPUT DATA TRANSFER TO TCU

The following procedure is performed when a channel parity error occurs on a PPU output data transfer to TCU (tape).

The PPU should:

1. Complete the output data transfer.
2. Request a general status function (0012_8). The MTS processor does not reply to any other function in this case.

The MTS processor:

1. Clears the channel parity error bit in the coupler.
2. Sets the general status alert bit.
3. Sets the detailed status error code 54_8 indicating a channel parity error occurred during transfer of data.

To recover the data, use the write parity error recovery procedures described in section 4 of this manual.

INPUT TRANSFER OF PARAMETERS OR DATA TO PPU

The following procedure is performed when a channel parity error occurs on input transfer of parameters or data to the PPU.

The PPU should:

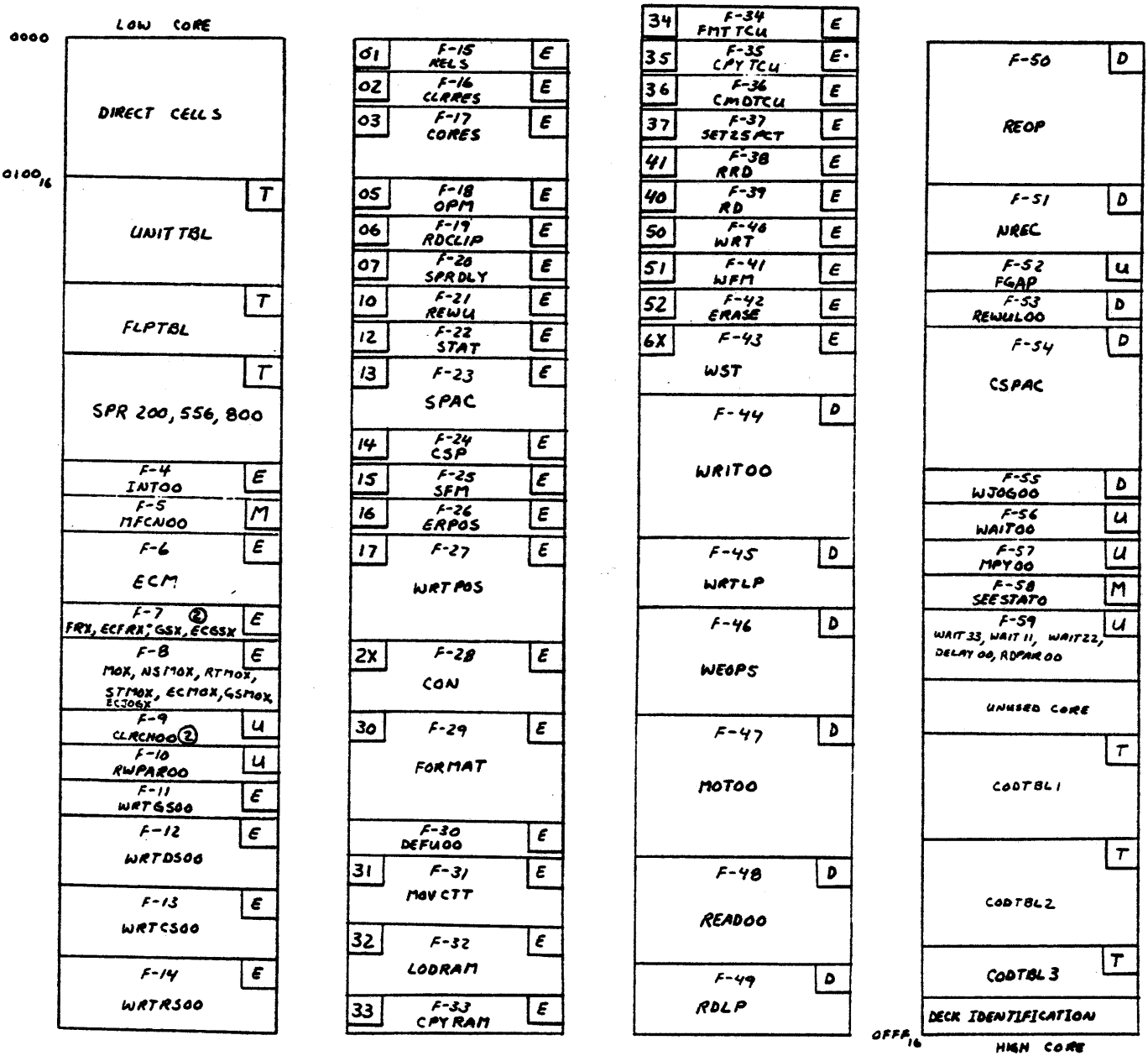
1. Test the appropriate bit in the status and control registers prior to sending the next function.
2. Perform one of the following to attempt recovery:
   a. Parameters; resend the function and input the parameters several times before downing the channel.
   b. Data (MTS read forward); backspace and reread the data block via the read forward function several times before abandoning the operation.
   c. Data (MTS read backward); forespace and reread the data block via the read backward function several times before abandoning the operation.

MTS CONTROLWARE FLOWCHARTS F

This appendix contains flowcharts for the following MTS controlware.

- MB434 - Used with CDC CYBER 70/170 and 6000 computers
- MB435 - Used with CDC CYBER 70 and 7600 computers

Figure F-1 shows controlware routine placement in MTS processor memory, and figure F-2 shows the general flow of control between routines. Figure F-3 is a key to the symbols used in figures F-4 through F-59.



NOTES:
① M8434-A ONLY
② M8435-A ONLY

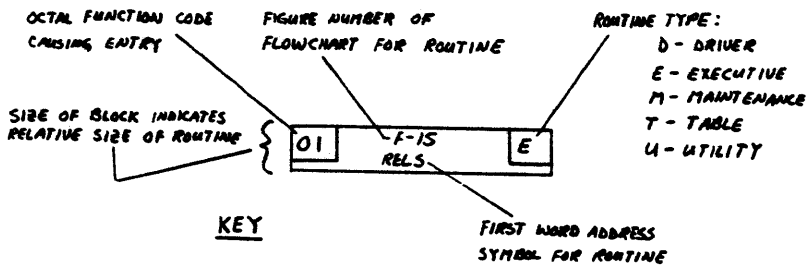


Figure F-1. Controlware Memory Map

ECO 38303
ECO 38797
ECO 34206
ECO 34436
ECO 35305
ECO 35982
ECO 34757

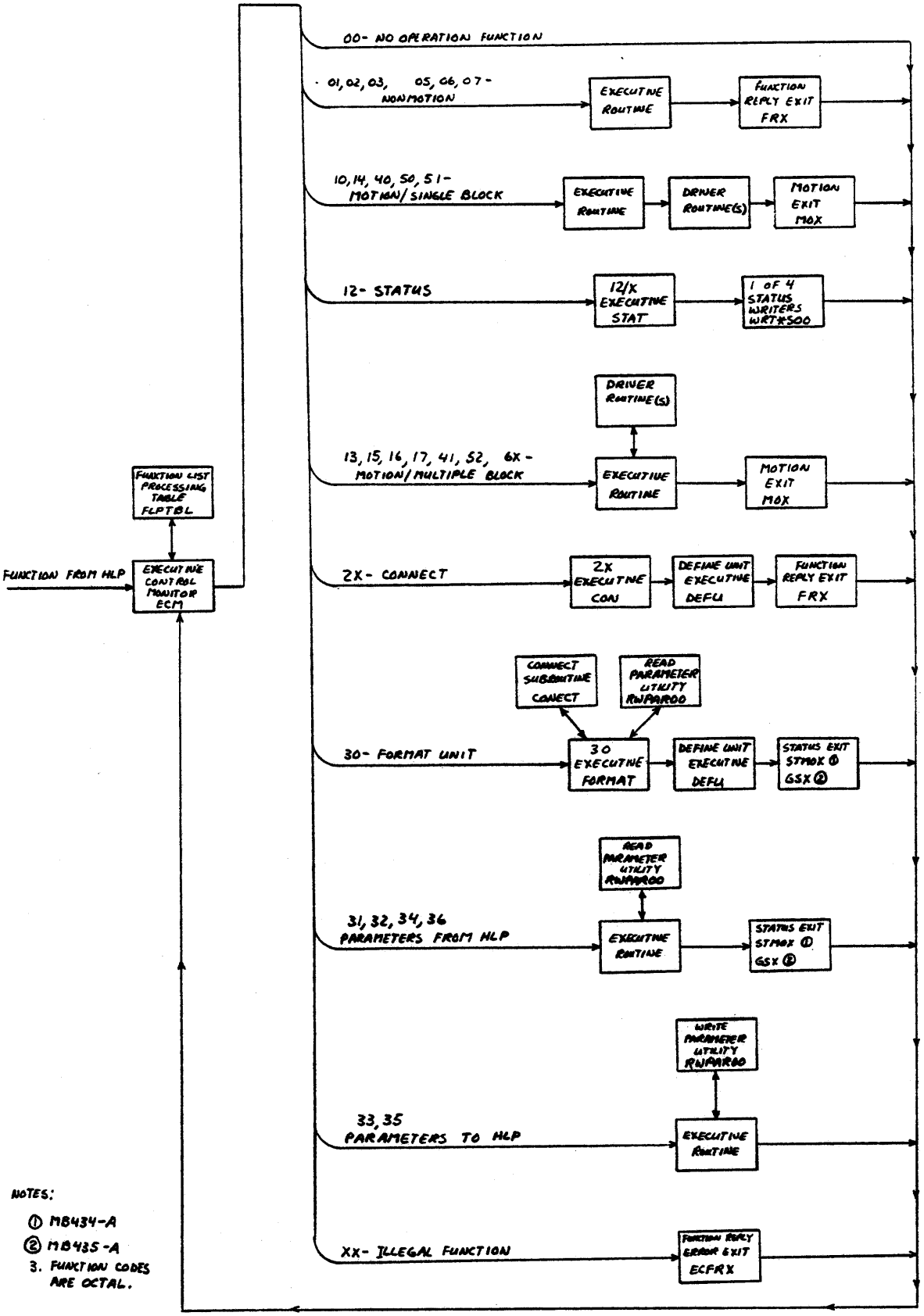


Figure F-2. Control Flow Between Routines

ECO 34206
ECO 36757
ECO 38797

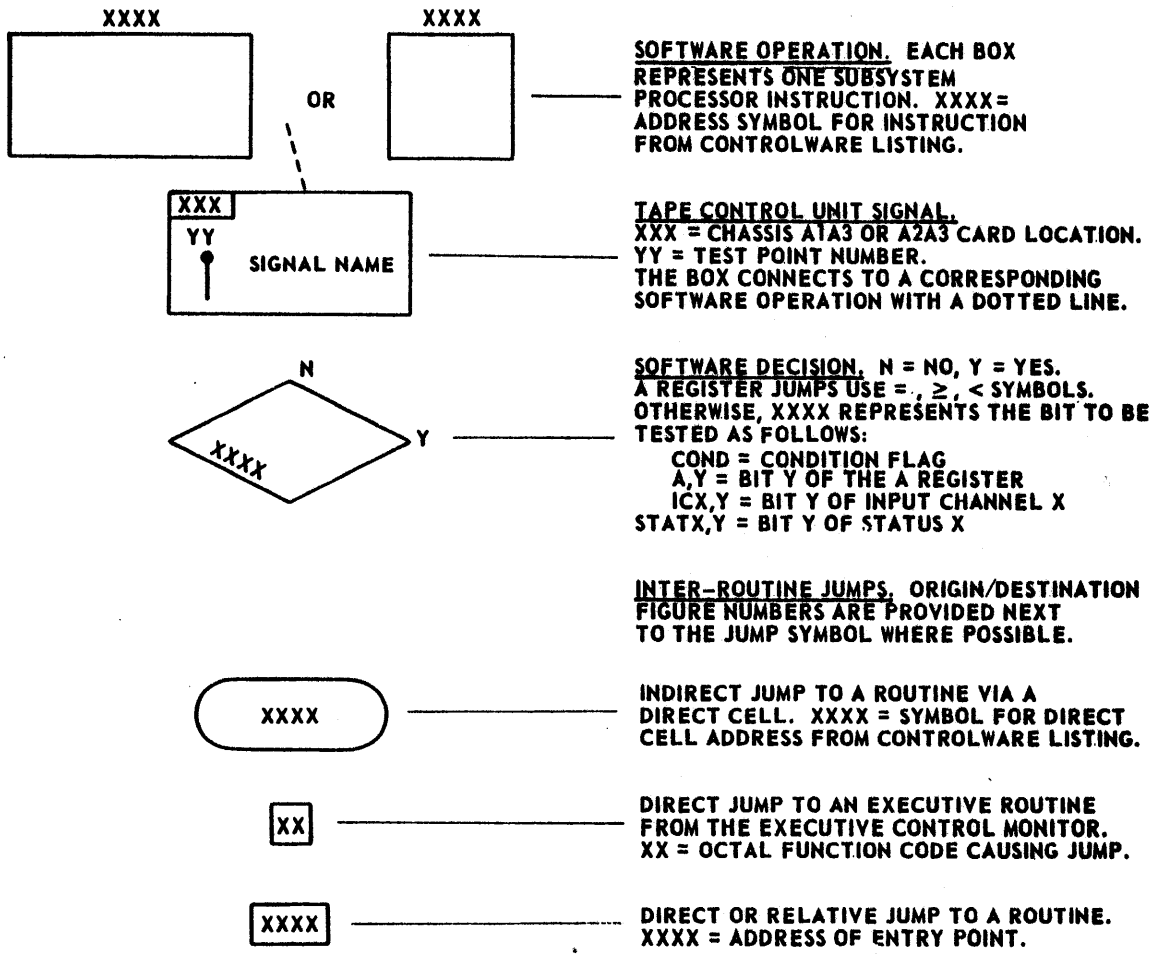


Figure F-3. Key to Flowchart Symbols

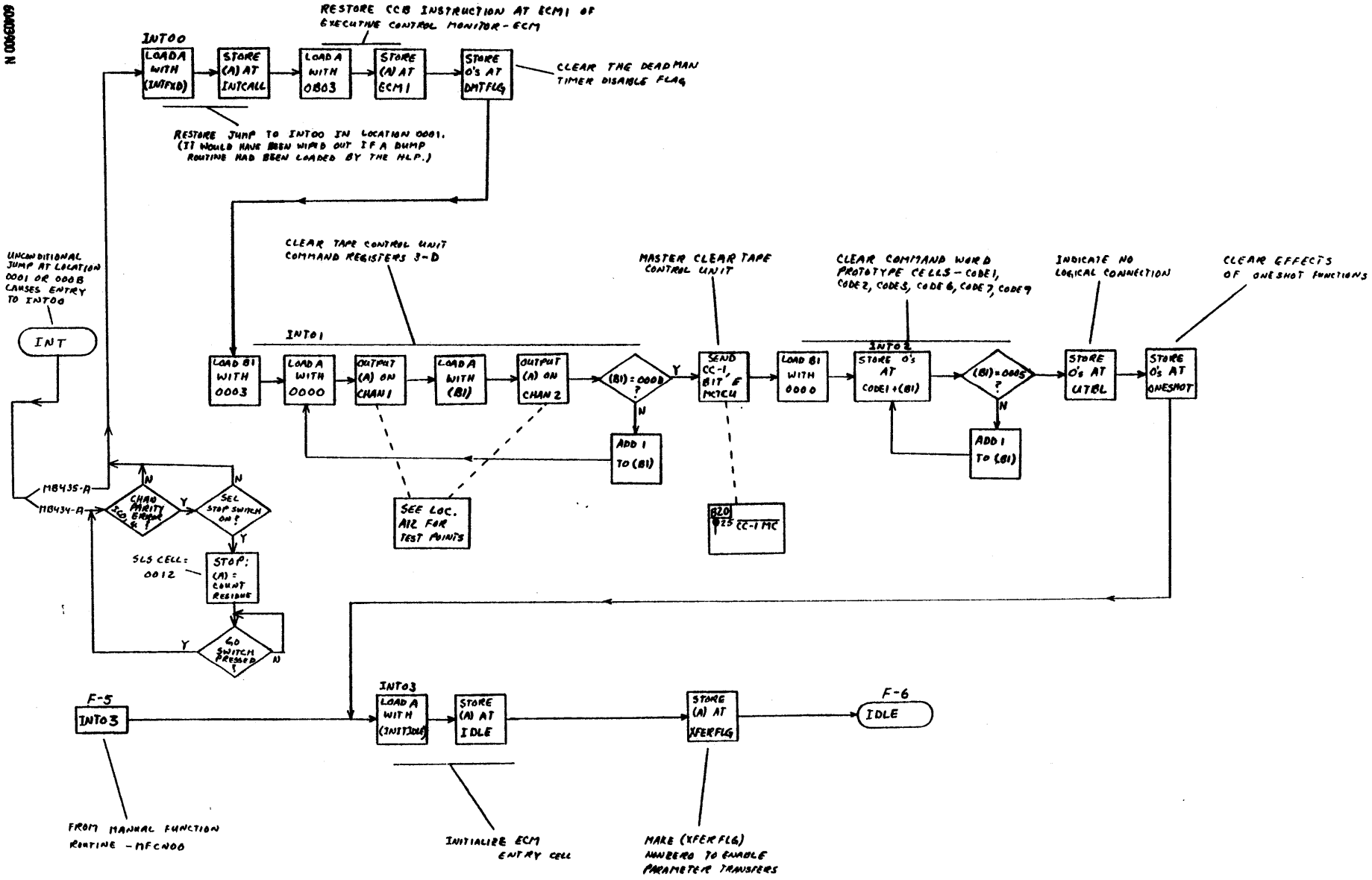


Figure F-4. Initialization Procedure - INT00

ECO 34081
ECO 35982
ECO 38303

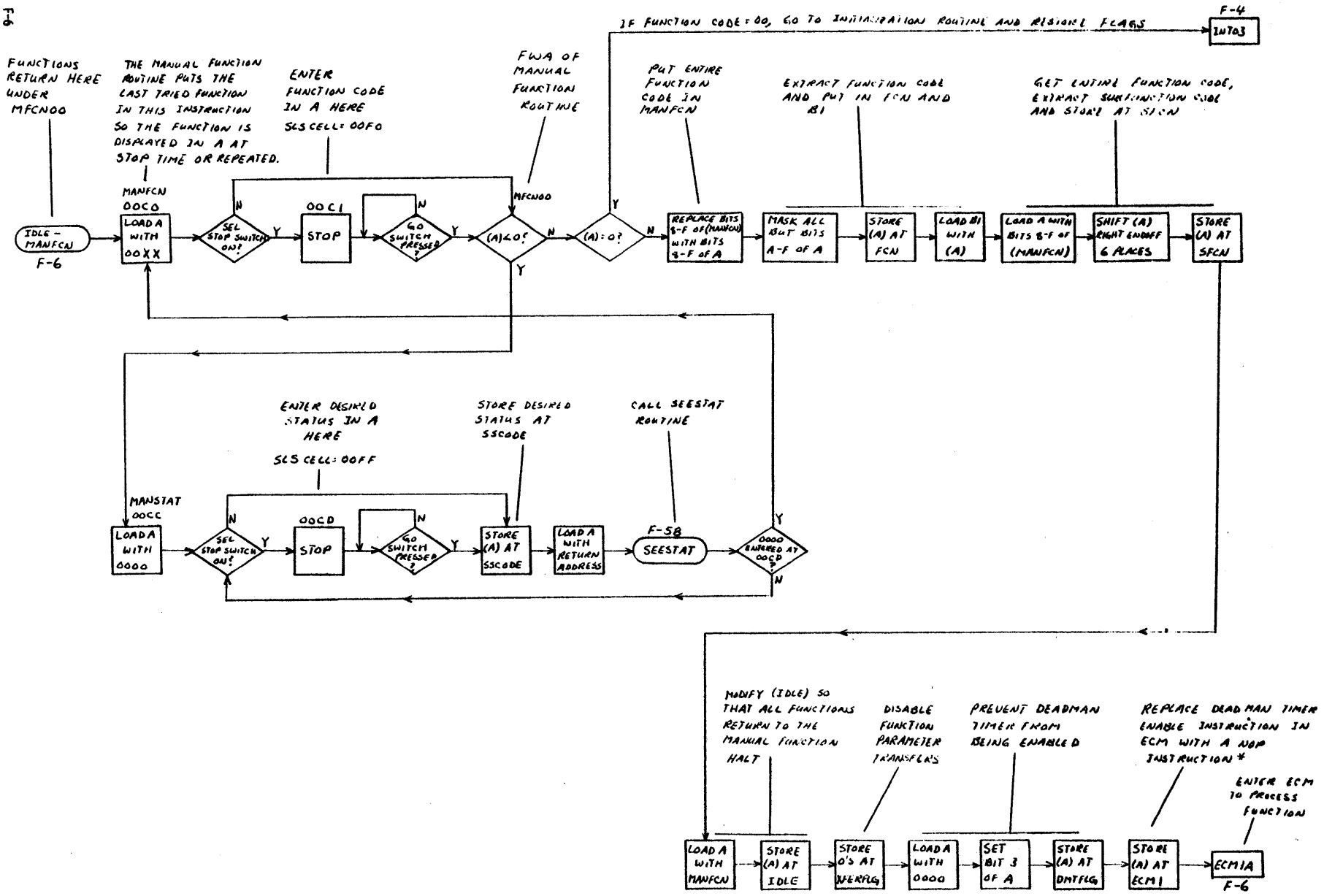


Figure F-5. Manual Function Entry Routine - MFCN00

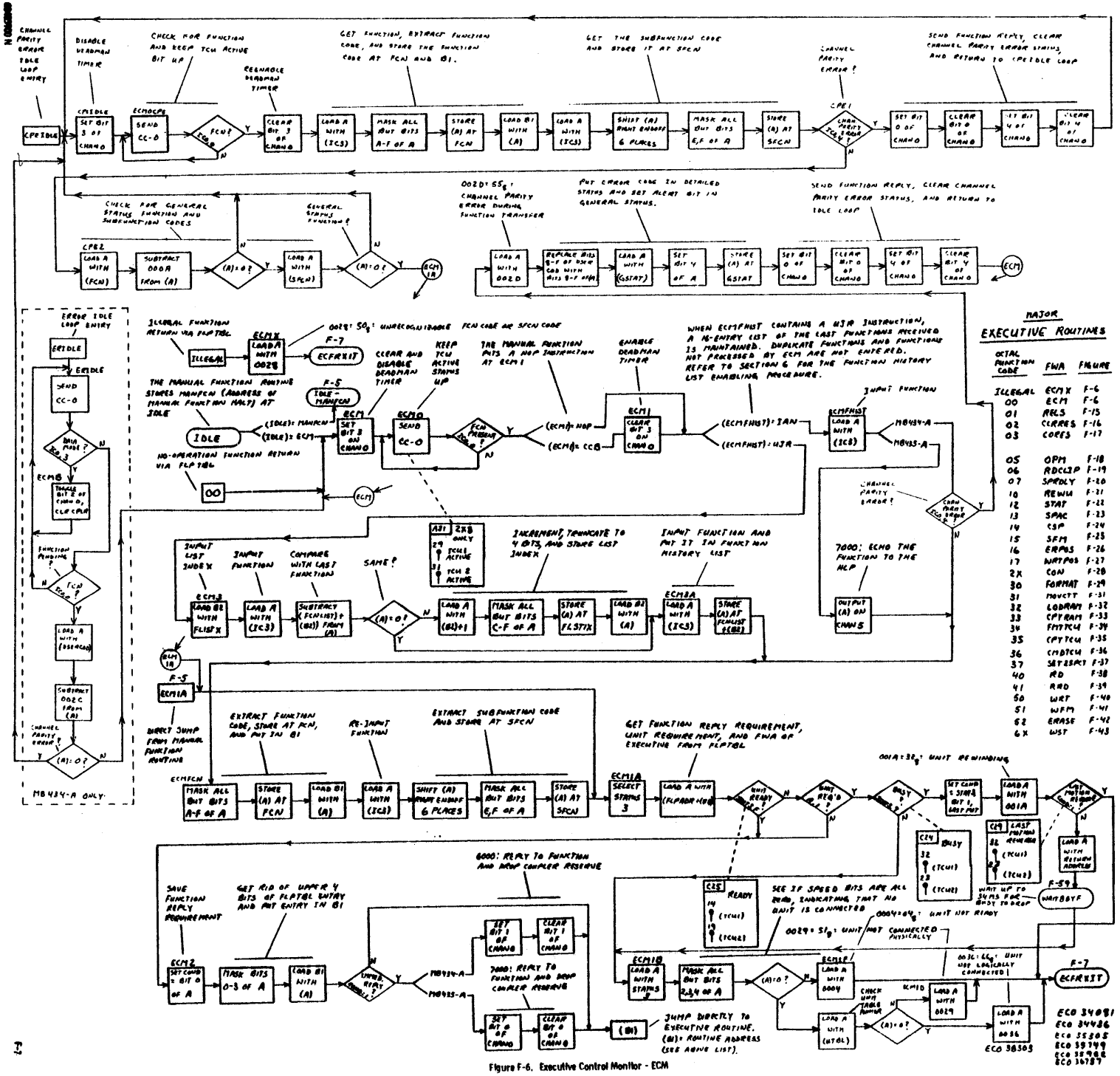


Figure F-6. Executive Control Monitor - ECM

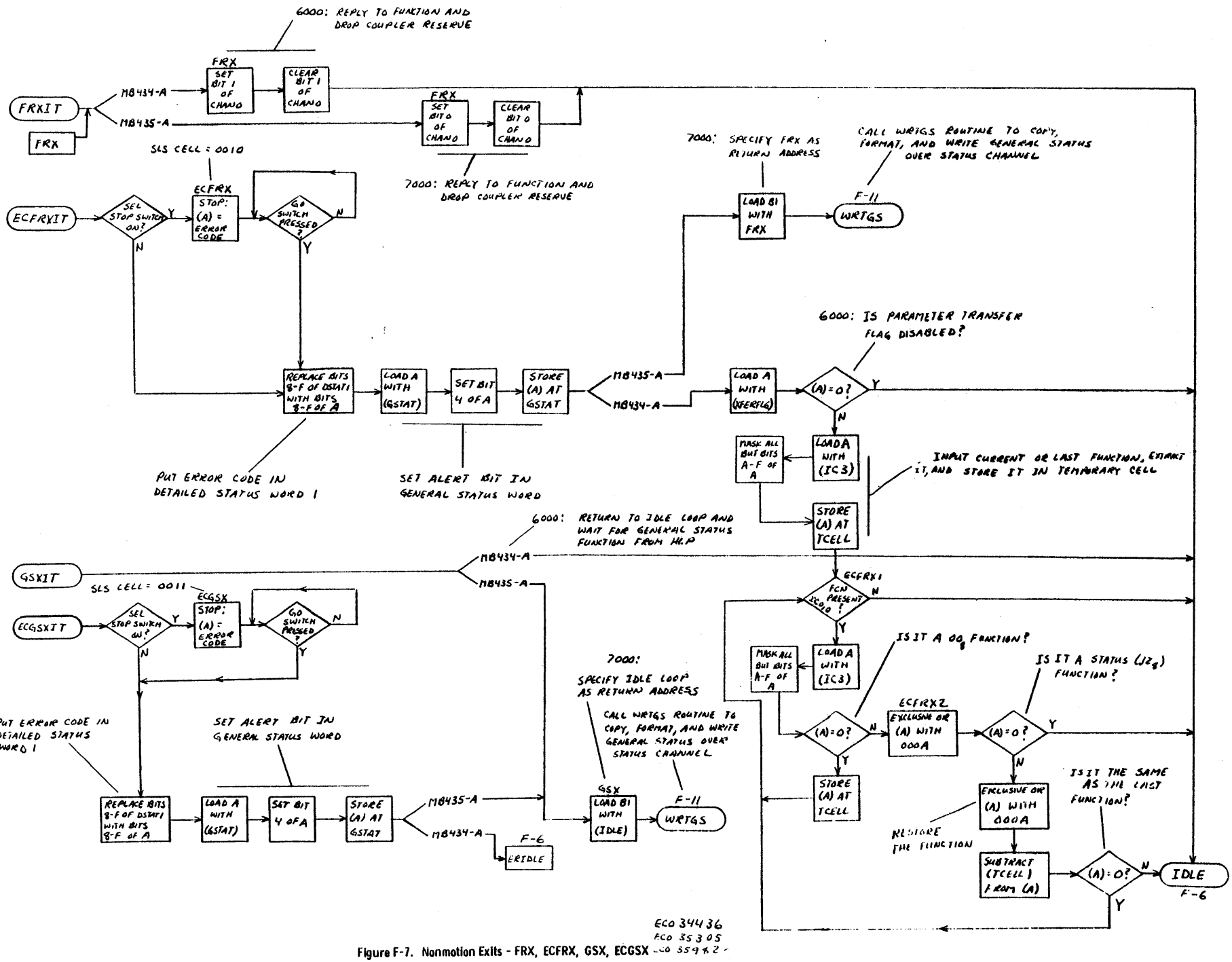


Figure F-7. Nonmotion Exits - FRX, ECFRX, GSX, ECGSX

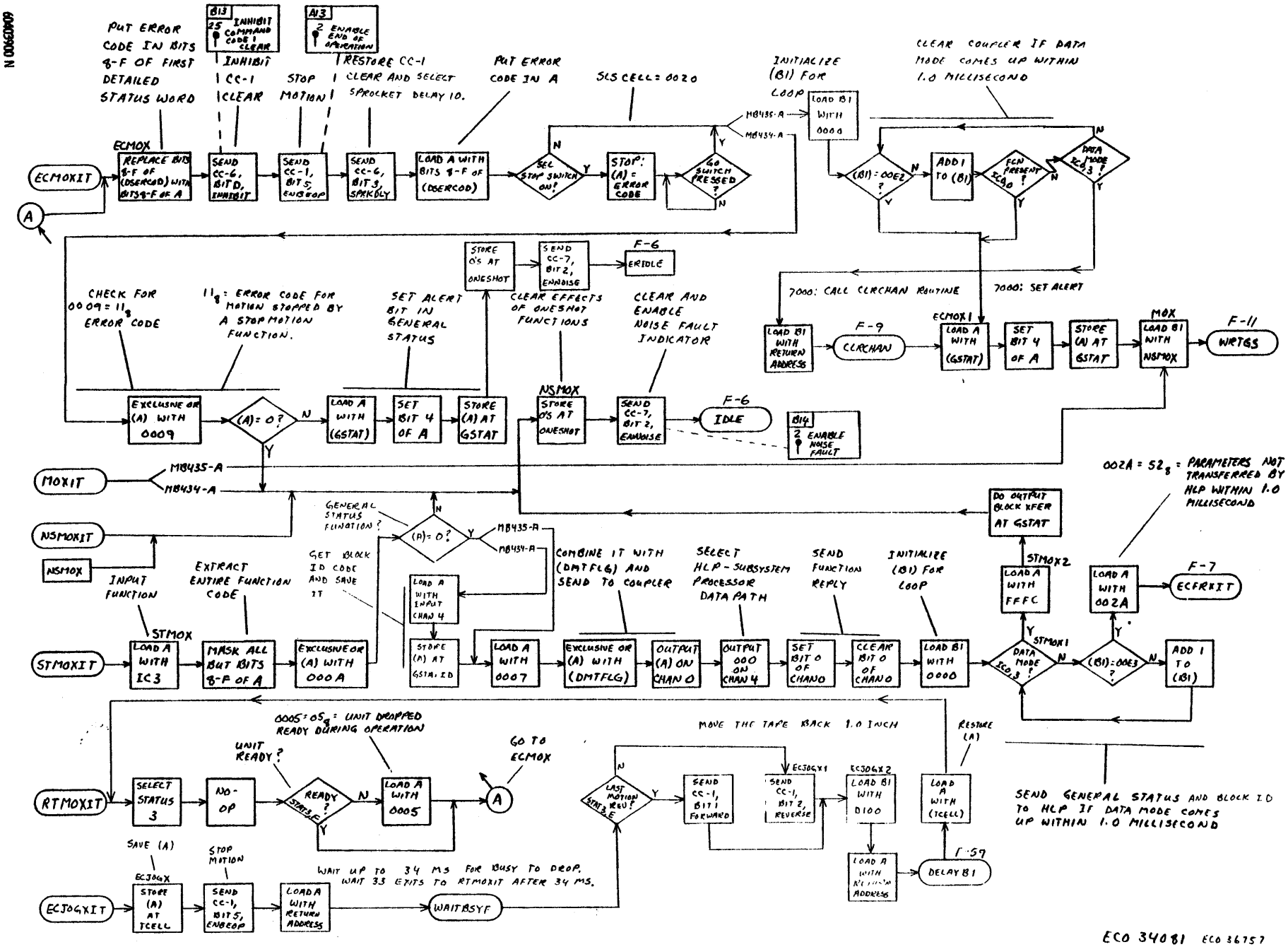


Figure F-8. Motion Exits - ECMOX, MOX, NSMOX, STMOX, RTMOX, ECJOGX

ECO 34081 ECO 36757
ECO 34436 ECO 37442
ECO 35305
ECO 35749
ECO 35982

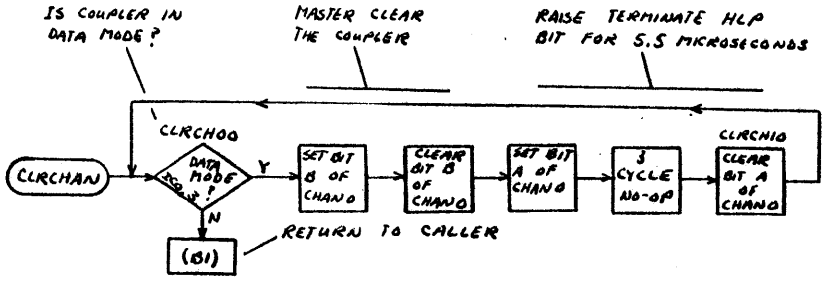
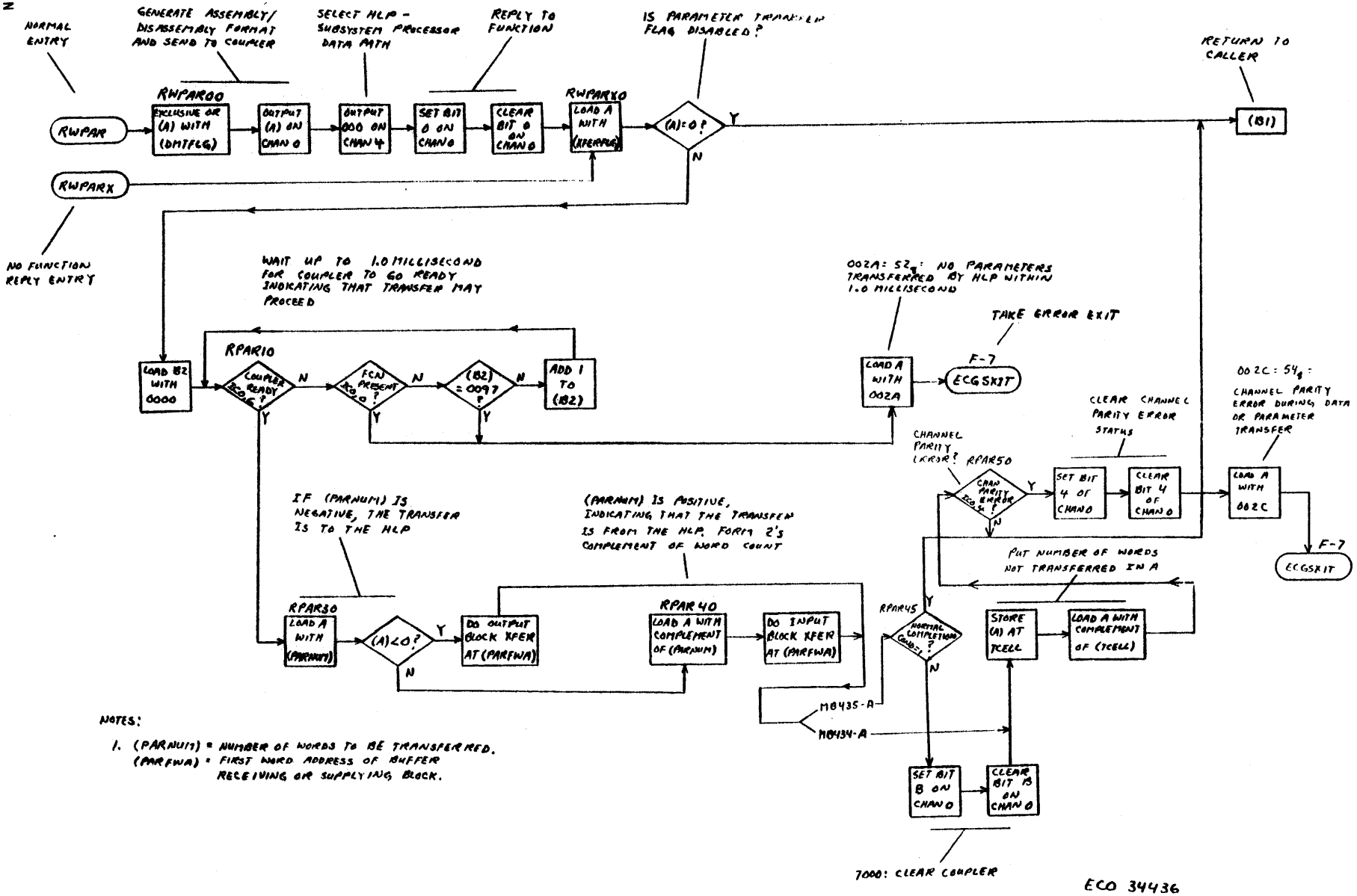


Figure F-9. MB435-A only; Clear Channel - CLRCH00

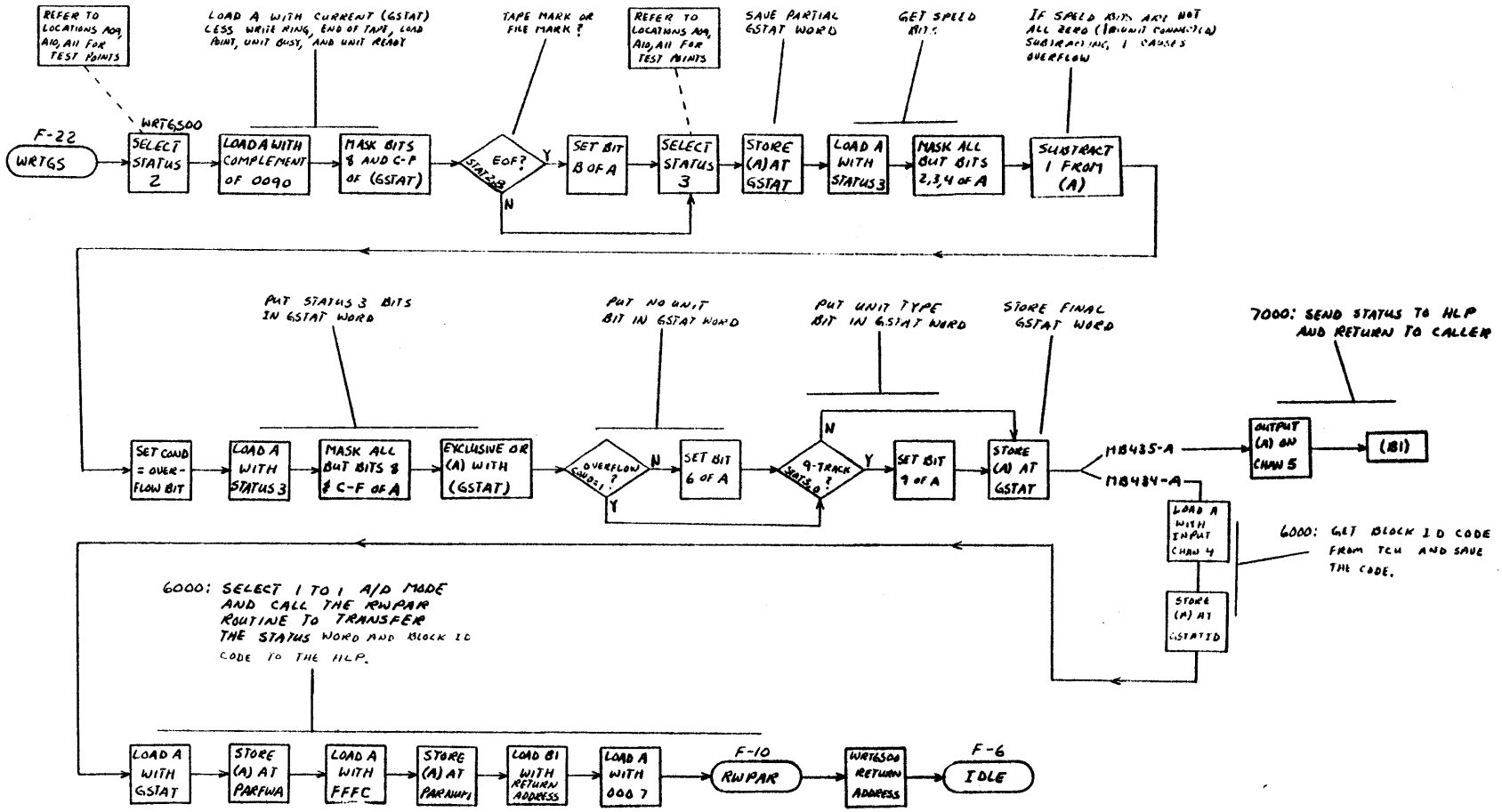


NOTES:

1. (PARNUM) = NUMBER OF WORDS TO BE TRANSFERRED.
   (PARFWA) = FIRST WORD ADDRESS OF BUFFER RECEIVING OR SUPPLYING BLOCK.

ECO 34436
ECO 35305
ECO 35982
ECO 37303

Figure F-10. Read/Write Parameters - RWPAR00



ECO 34206
ECO 34436
ECO 37441

Figure F-11. Write General Status - WRTGS00

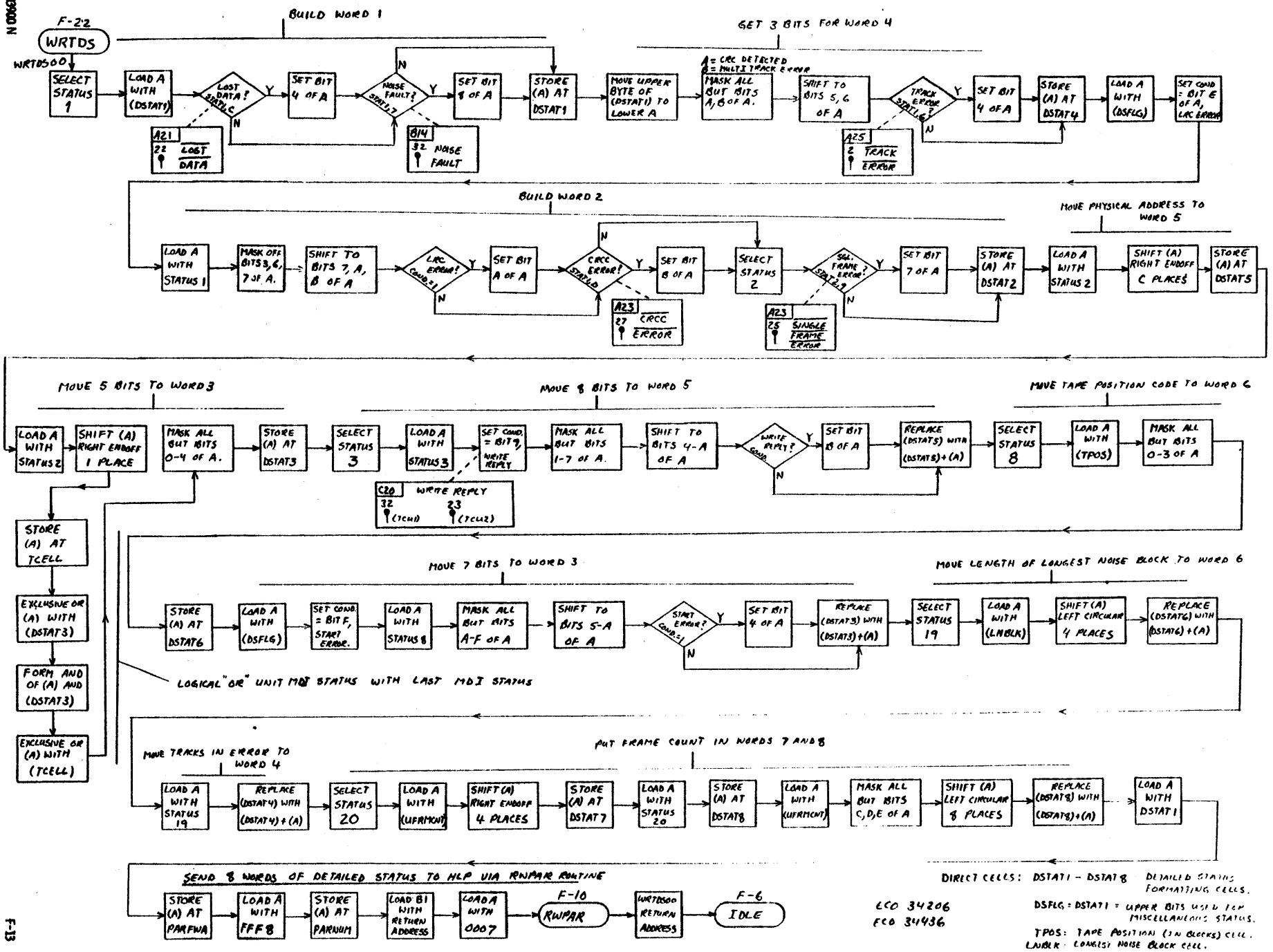
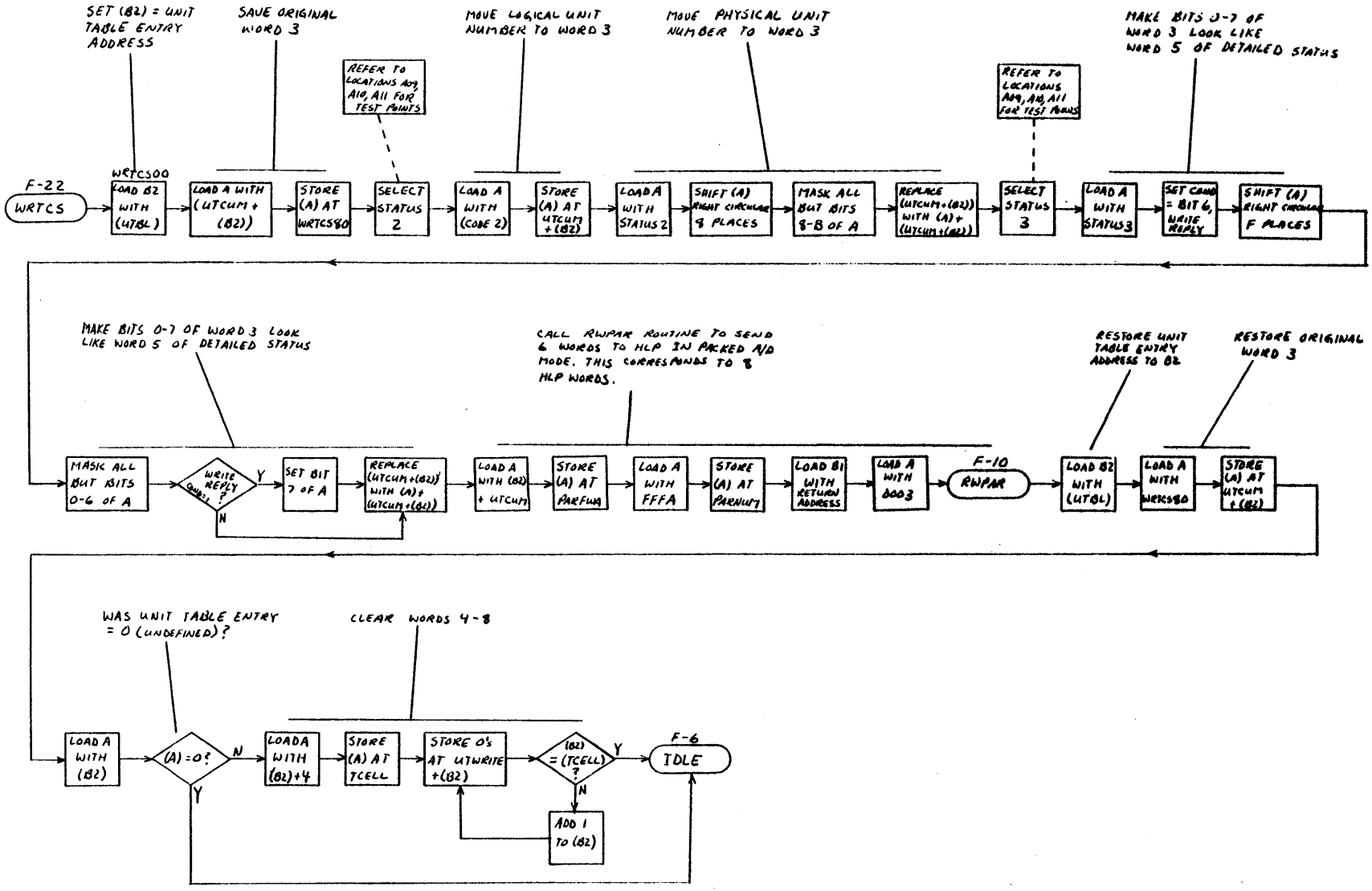
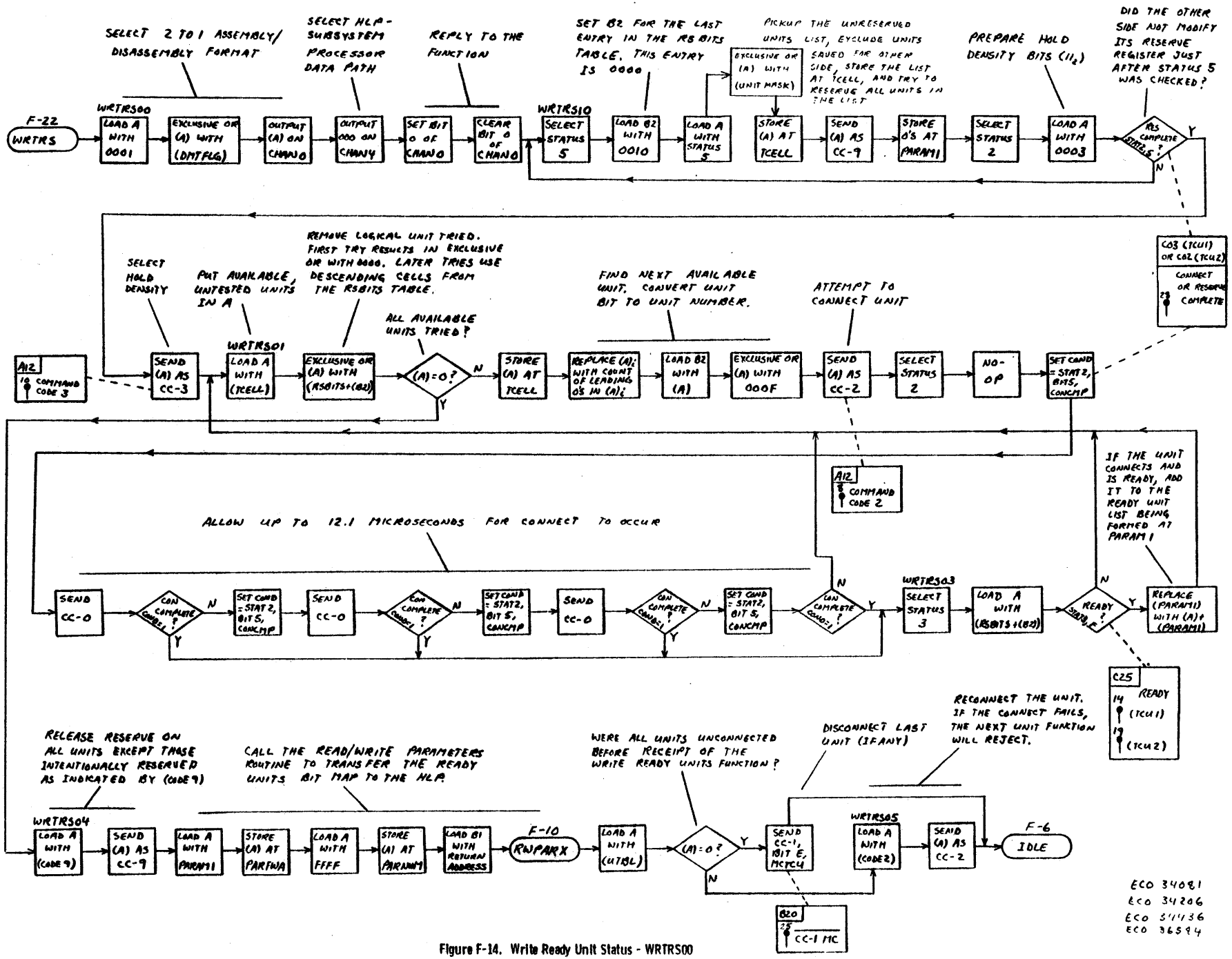


Figure F-12. Write Detailed Status - WRTDS00



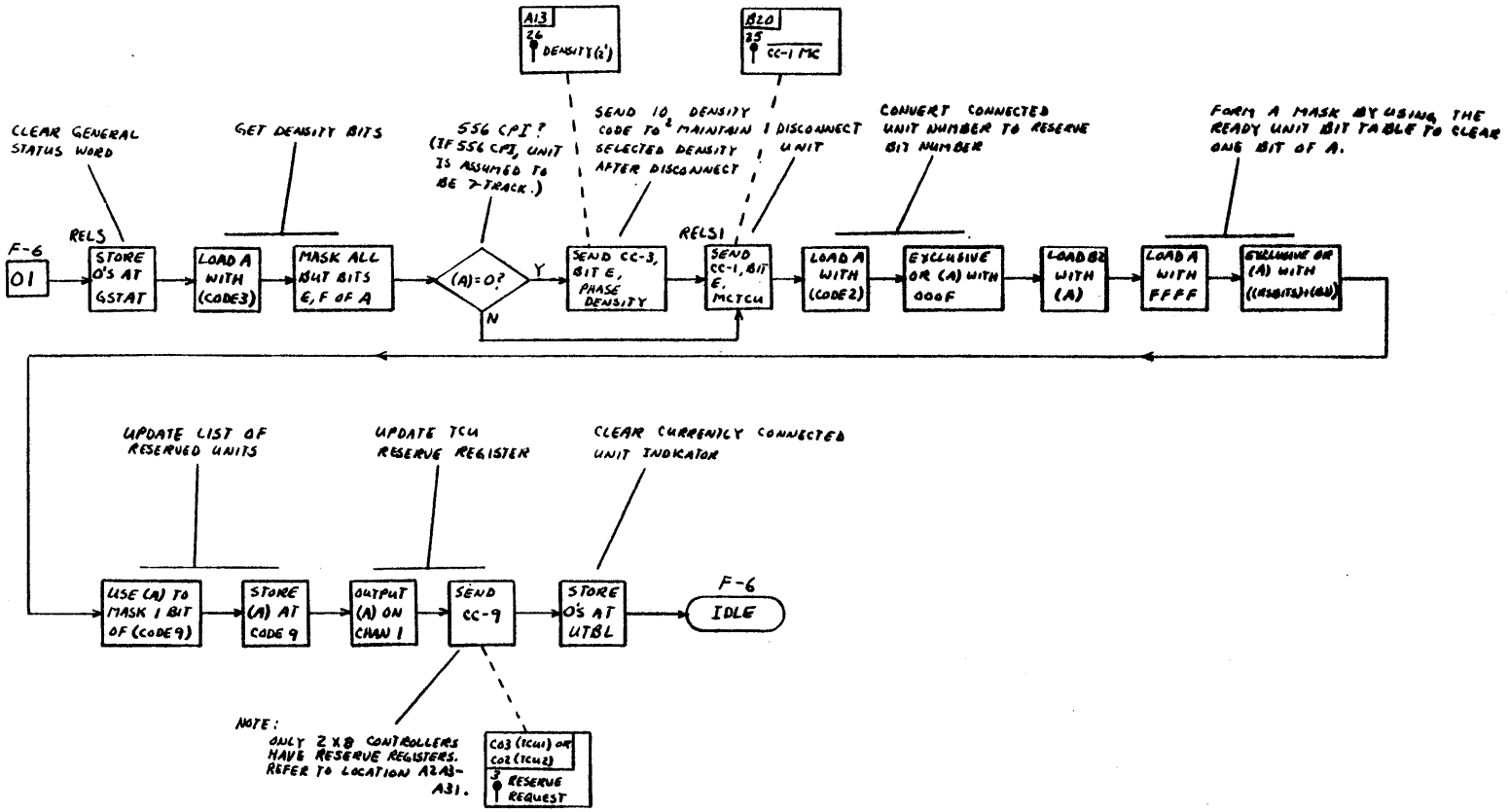
ECO 34081
ECO 34206

Figure F-13. Write Cumulative Status - WRTCS00



ECO 34081
ECO 34206
ECO 341136
ECO 36594

Figure F-14. Write Ready Unit Status - WRTRS00



ECO 34206

Figure F-15. Release Unit - RELS

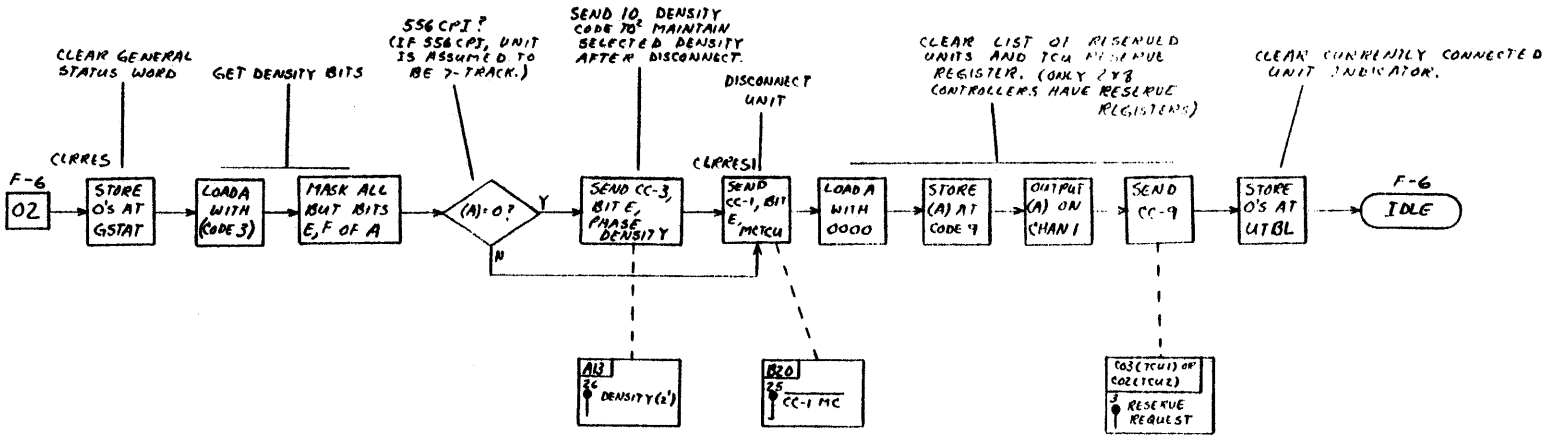
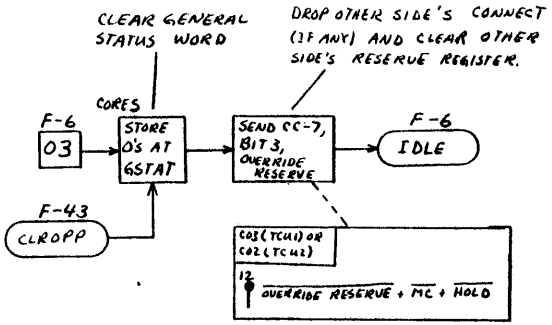


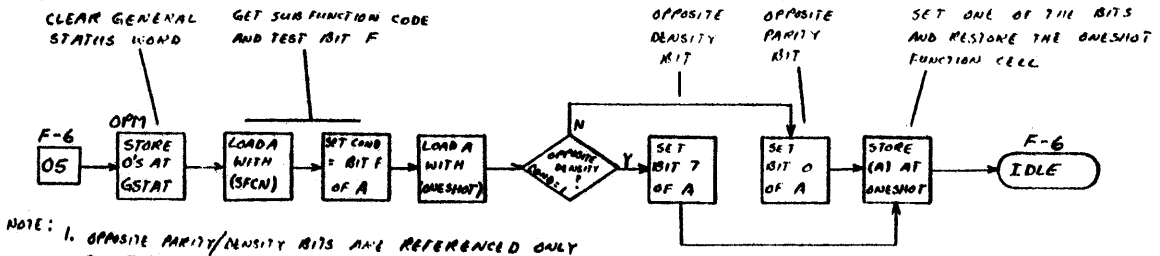
Figure F-16. Clear All Reserves - CLRRES



NOTE: THIS ROUTINE HAS NO EFFECT ON 1X8 CONTROLLERS.

ECO 35305

Figure F-17. Clear Opposite Reserve - CORES



NOTE:
1. OPPOSITE PARITY/DENSITY BITS ARE REFERENCED ONLY FOR 7-TRACK UNITS.
2. DENSITY REMAINS UNCHANGED WHEN 200 CPI WAS LAST REQUESTED DENSITY.

ECO 34681

Figure F-18. Select Opposite Parity/Density Mode - OPM

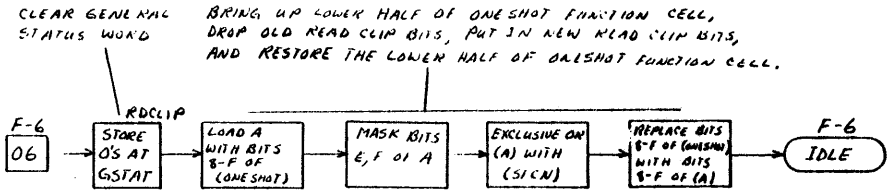
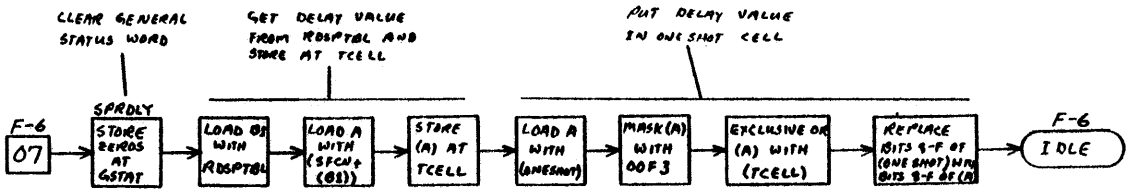


Figure F-19. Set Read Clip - RDCLIP



ADDRESS	CONTENTS		
RDSPTBL + 0	0000	=	50% DELAY
RDSPTBL + 1	0001	=	52% DELAY
RDSPTBL + 2	0004	=	48% DELAY

ECO 38303

Figure F-20. Select Read Sprocket Delay - SPRDLY

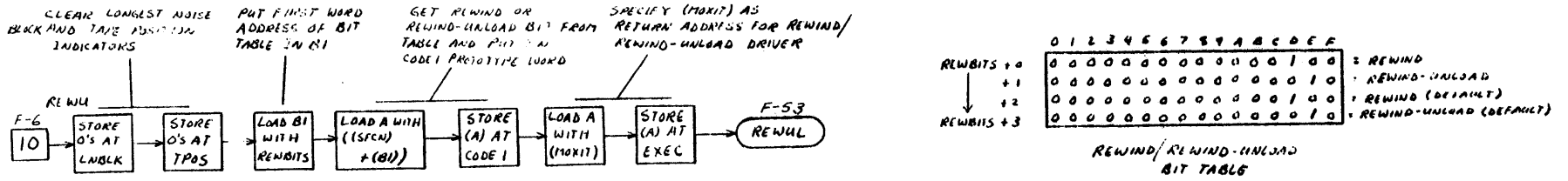


Figure F-21. Rewind/Rewind - Unload Executive - REWLL

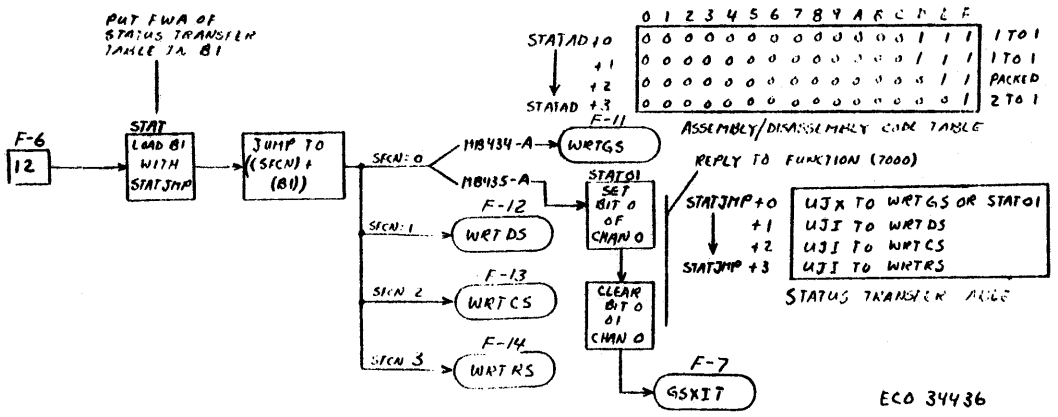


Figure F-22. Process Status Function - STAT

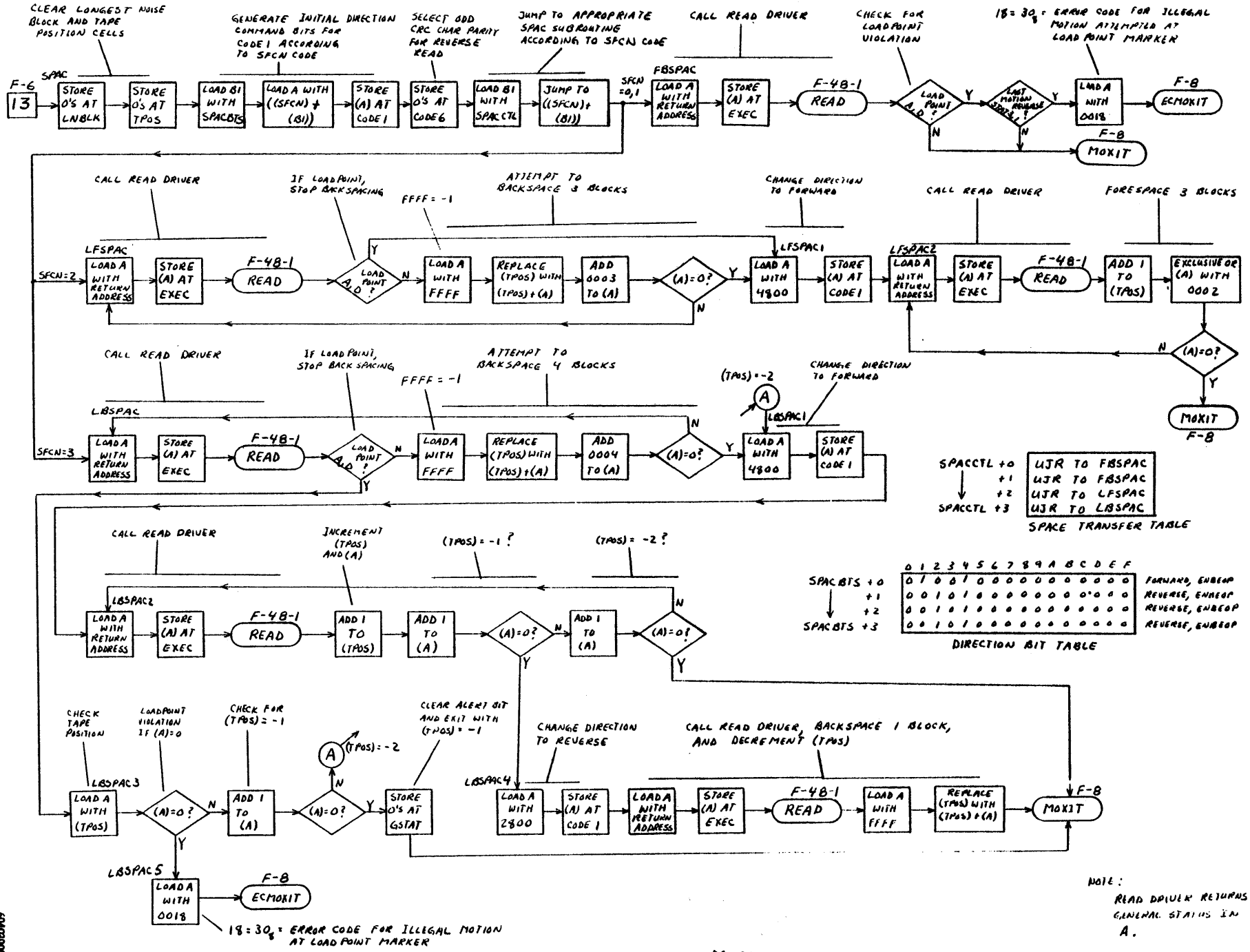


Figure F-23. Fore/Back, Long Fore/Back Space - SPAC

ECO 34081
ECO 34206

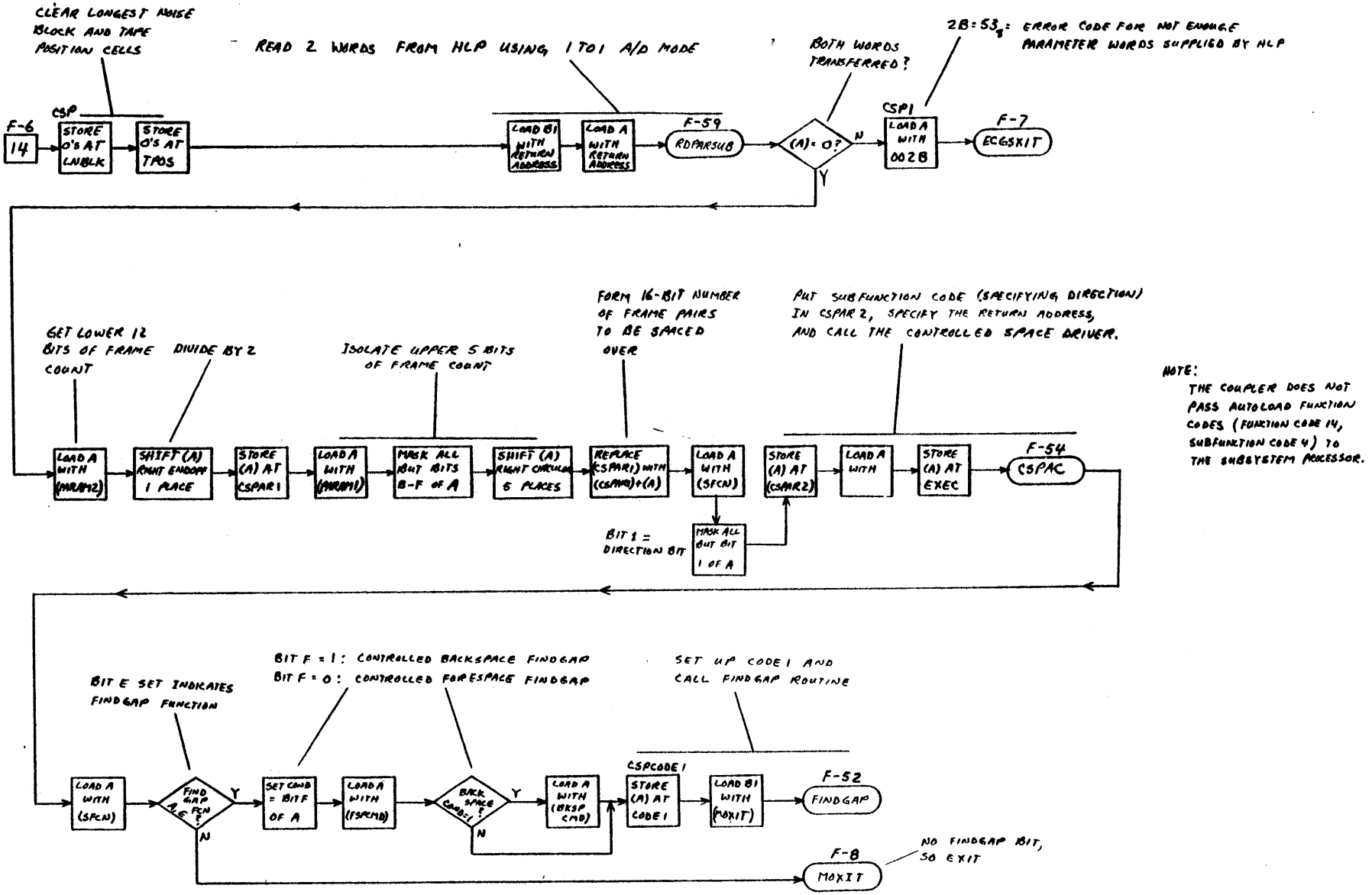


Figure F-24. Controlled Space Executive - CSP

ECO 35982
ECO 36757

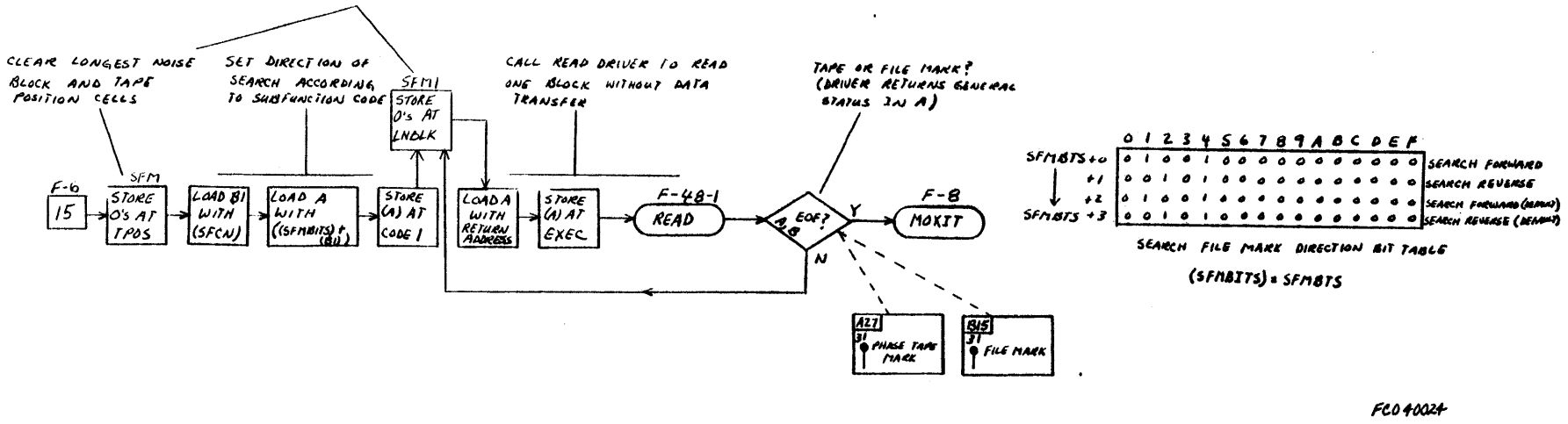


Figure F-25. Search File Mark - SFM

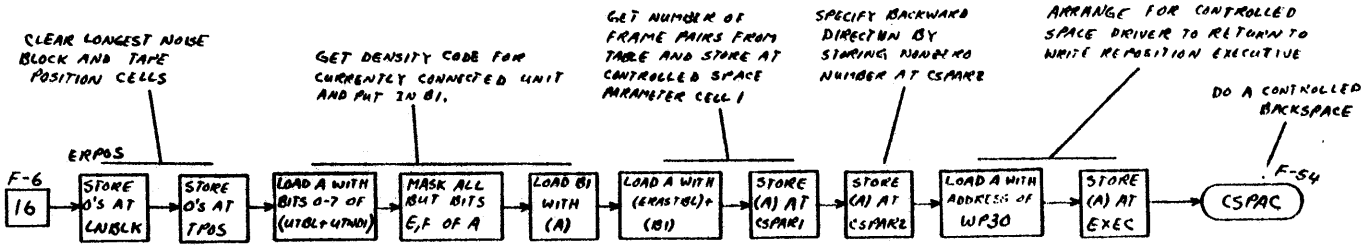
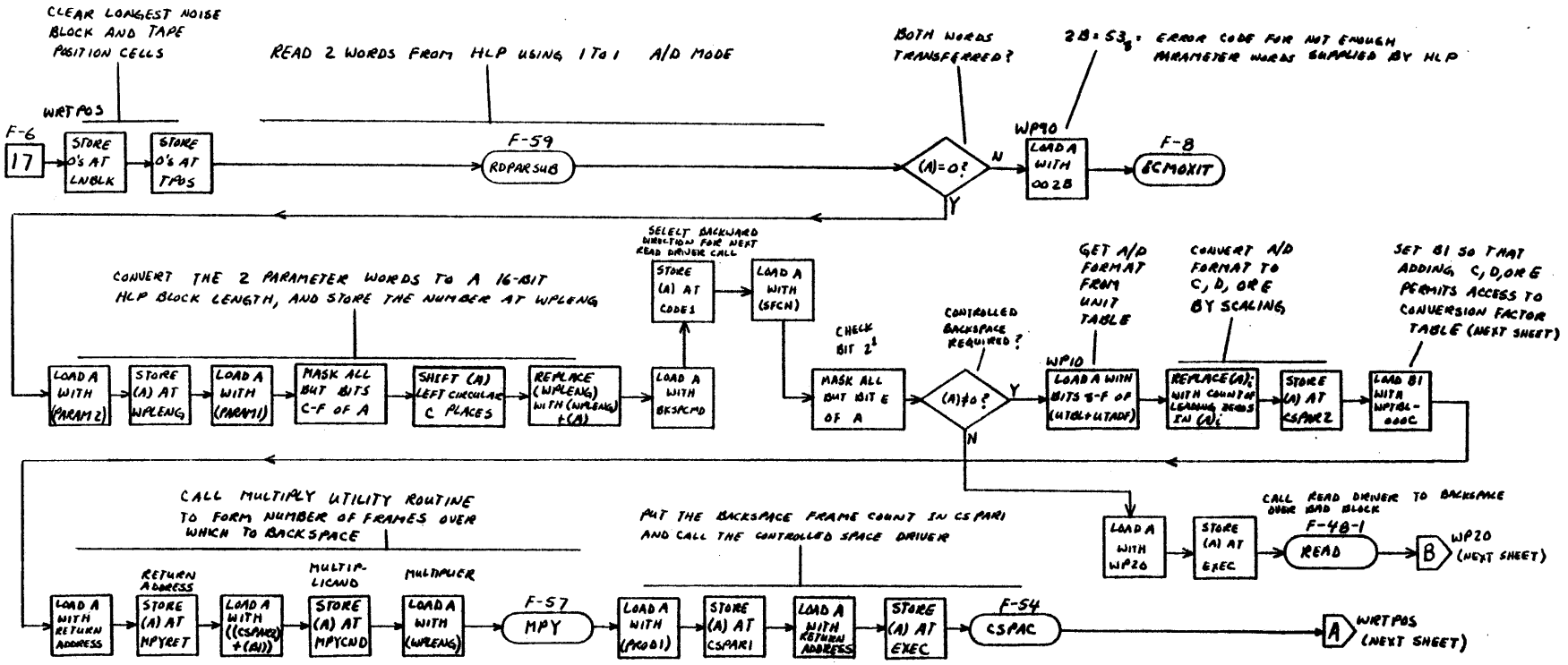


Figure F-26. Erase Reposition/Erase Reposition to Erase - ERPOS



ECO 36757 ECO 34081
ECO 34206
ECO 35982
ECO 36757

Figure F-27. Write Reposition/Write Reposition to Erase - WRTPOS (Sheet 1 of 2)

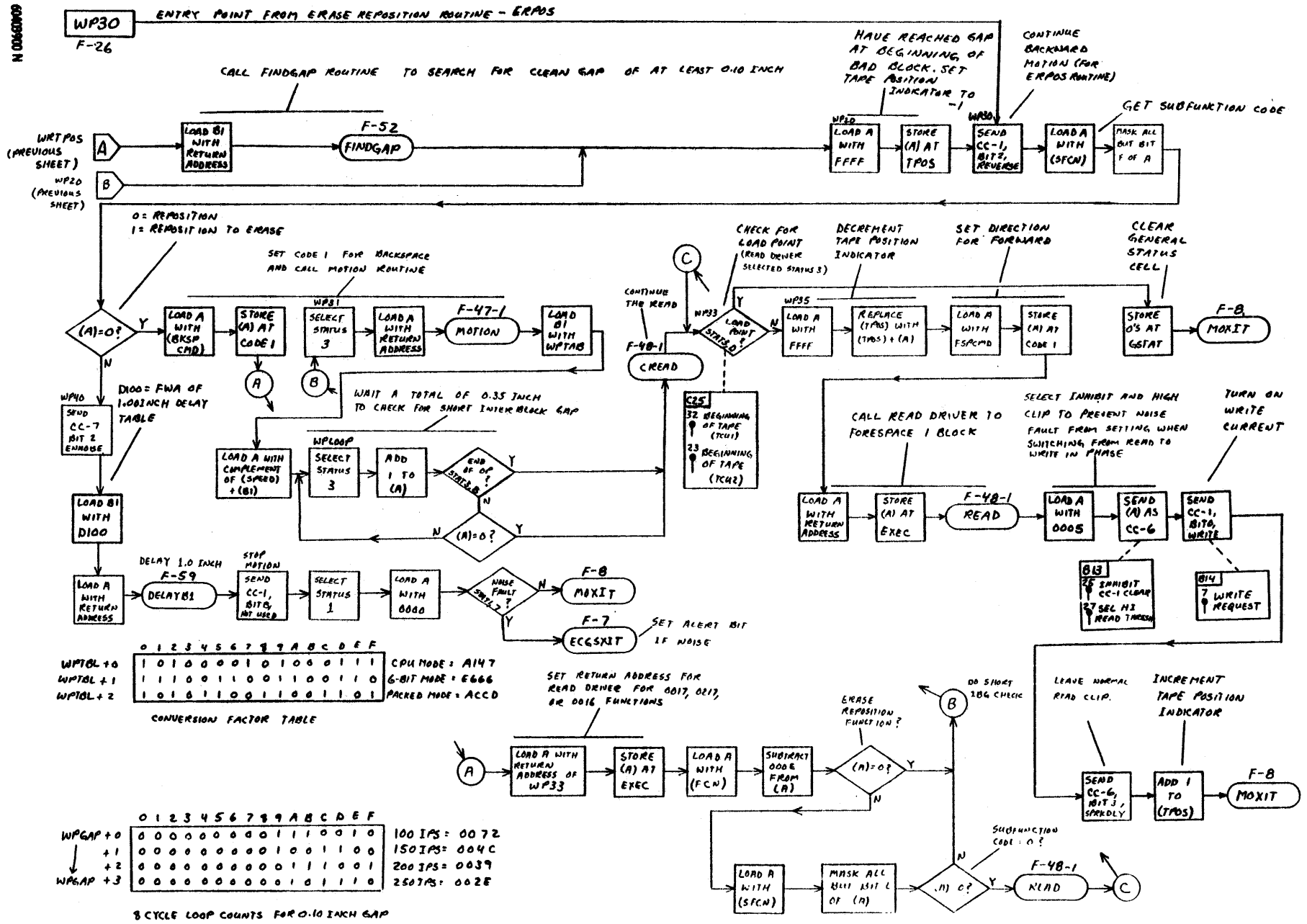


Figure F-27. Write Reposition/Write Reposition to Erase - WRTPOS (Sheet 2 of 2) (Cont'd)

ECO 36757 ECO 34083
ECO 37093 ECO 34206
ECO 35305
ECO 35982

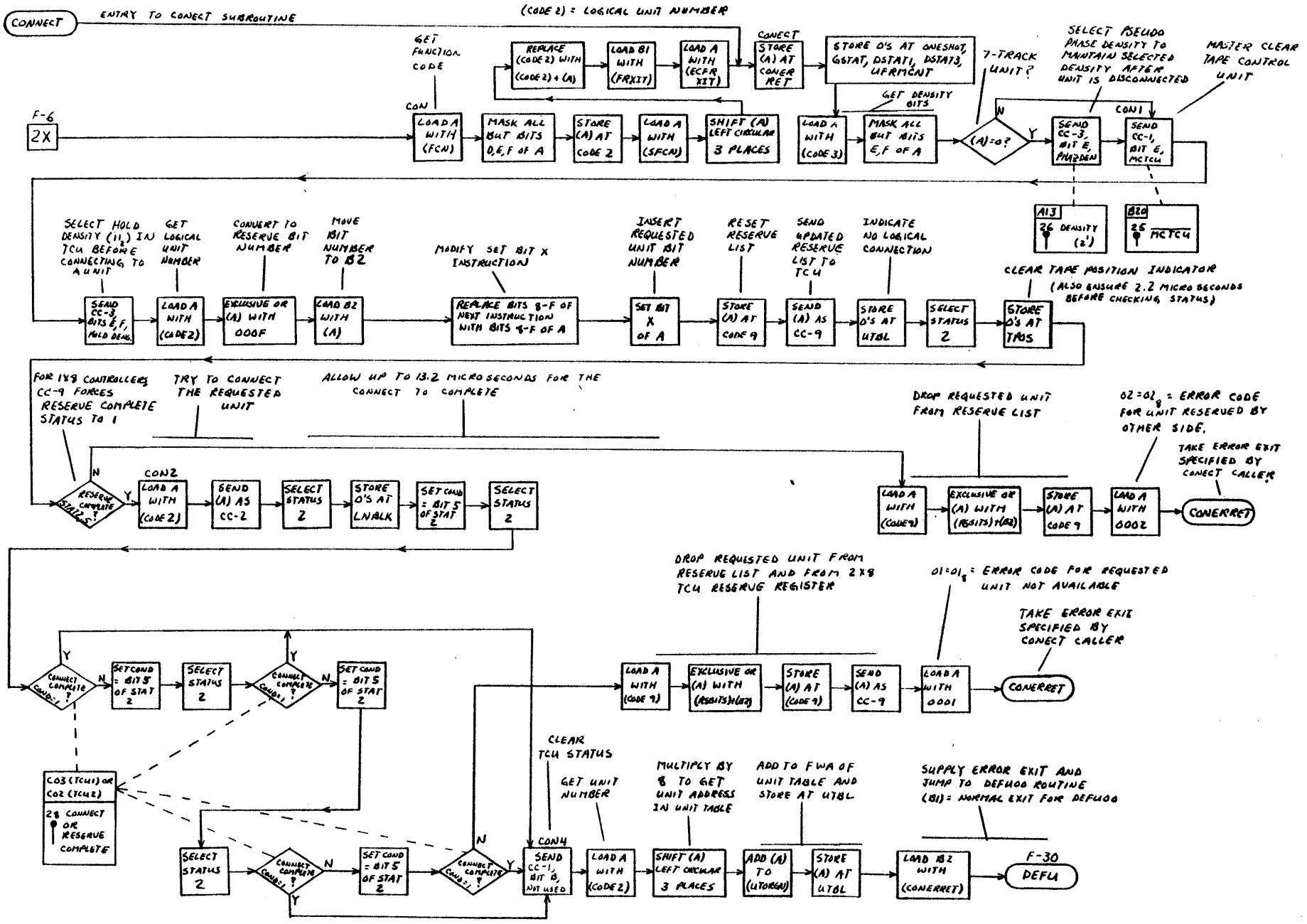


Figure F-28. Connect Unit - CON

ECO 34081
ECO 34206
ECO 34436
ECO 37442

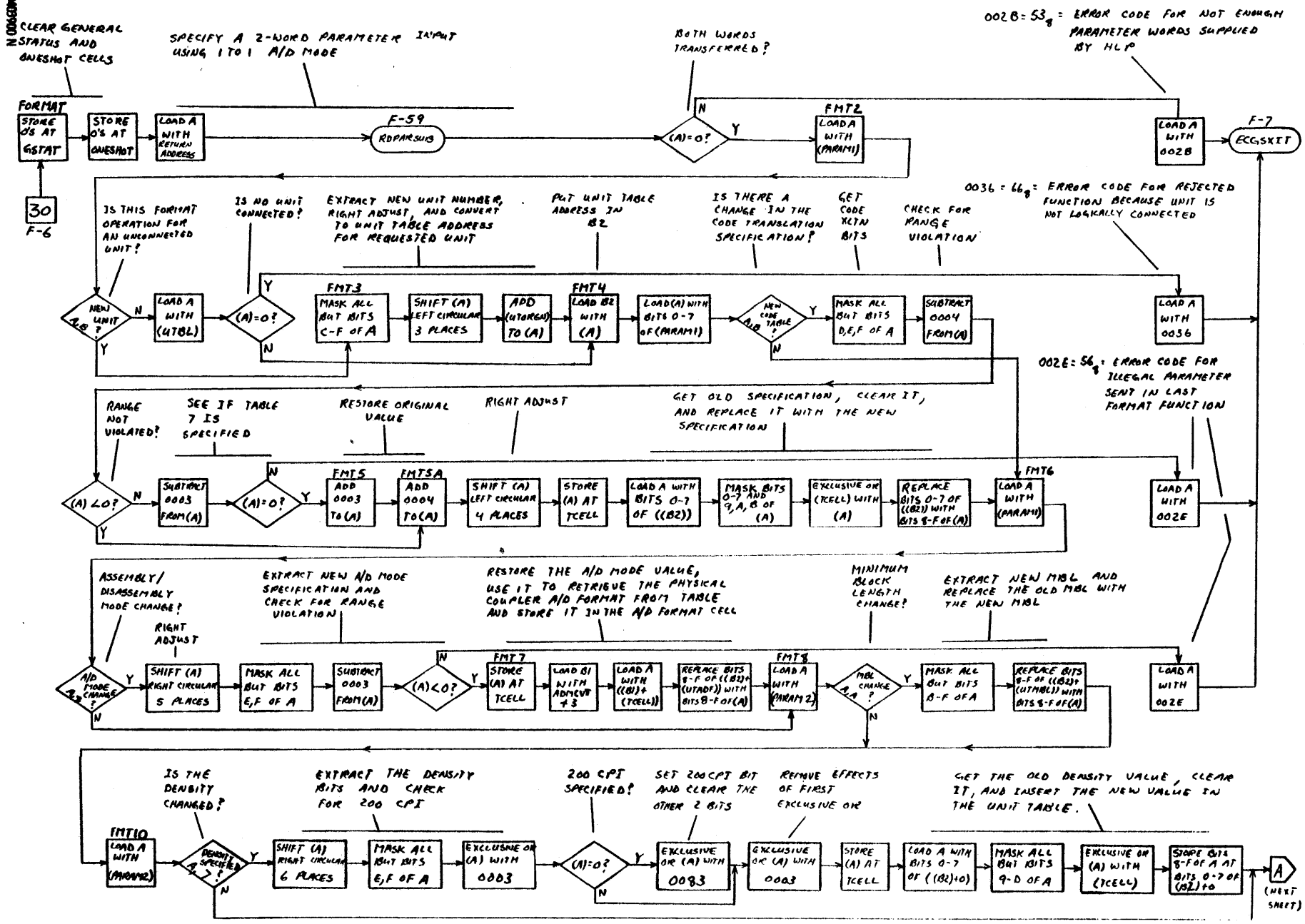
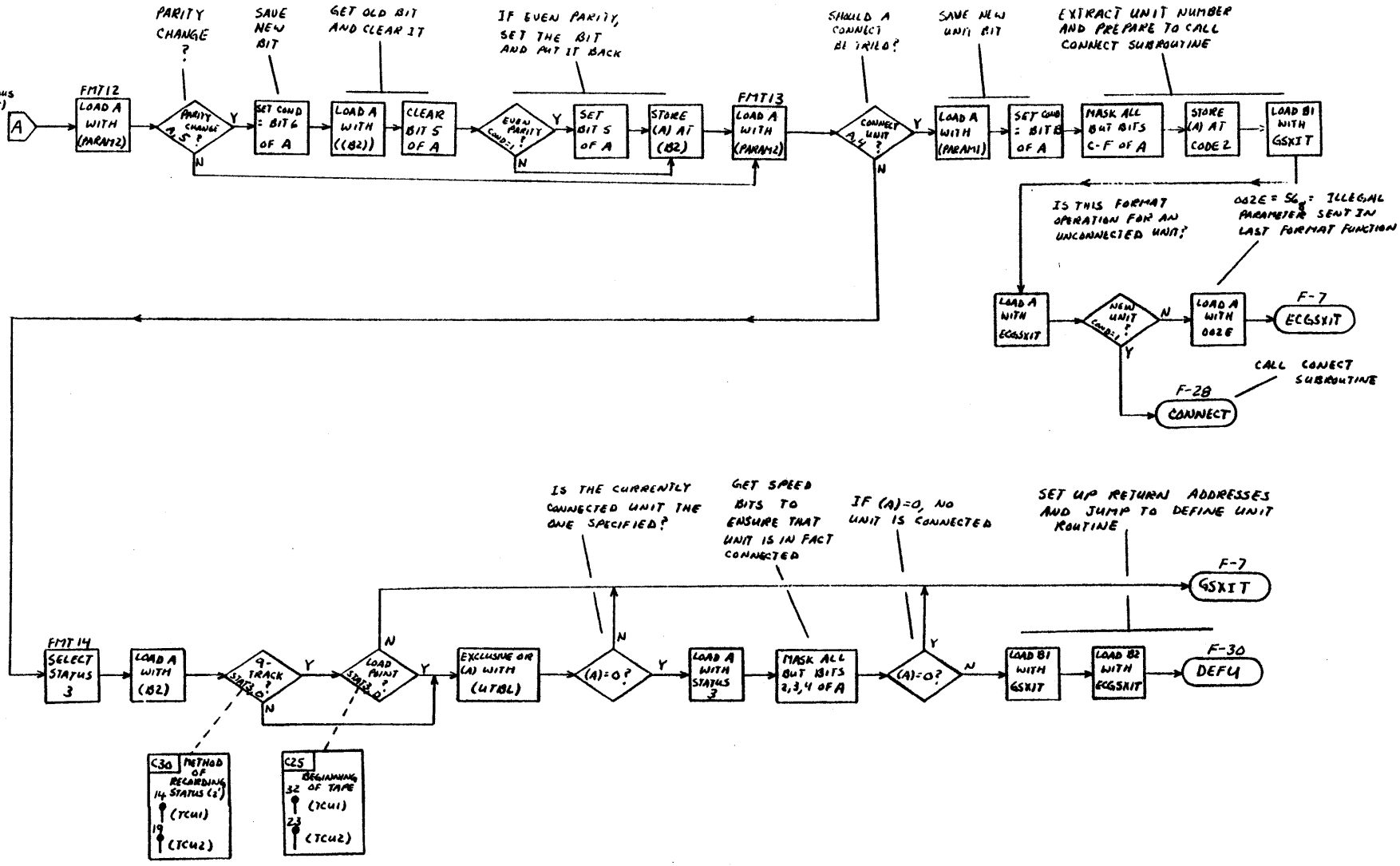


Figure F-29. Format Unit - FORMAT (Sheet 1 of 2)

ECO 34081
ECO 34436
ECO 35982
ECO 36757

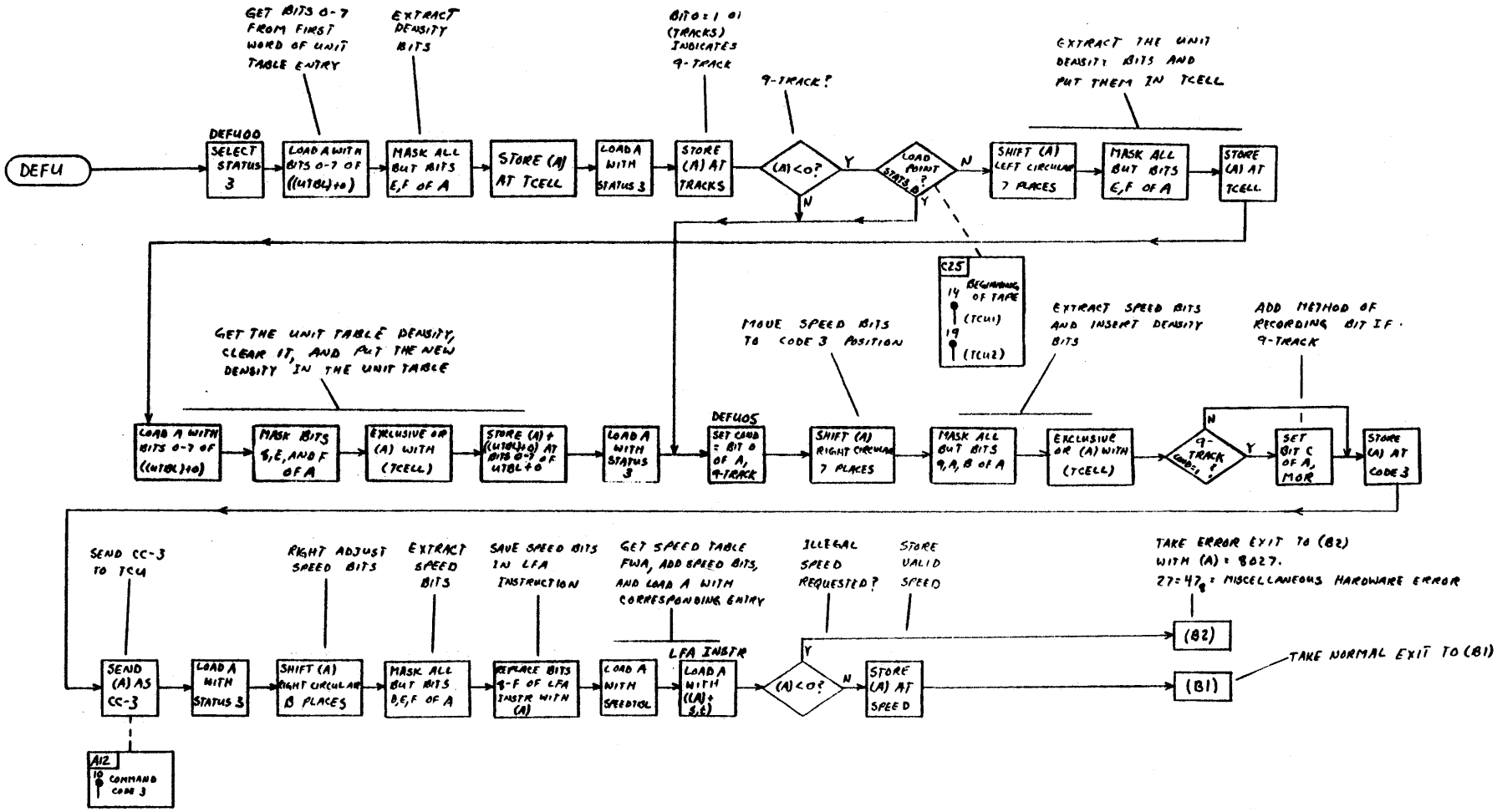


	0 1 2 3 4 5 6 7 8 9 A B C D E F	MODE NAME	MODE NO.	COUPLER A/D FORMATS WRITE	READ
ADMCUT+0	0 0 0 0 0 0 0 0 0 0 0 0 0 1 0 0	6-BIT	0	4	5
ADMCUT+1	0 0 0 0 0 0 0 0 0 0 0 0 0 0 1 0	PACKED	1	2	3
ADMCUT+2	0 0 0 0 0 0 0 0 0 0 0 0 0 1 0 0	CPU	2	8	9

A/D MODE TO PHYSICAL COUPLER A/D FORMAT CONVERSION TABLE

ECO 341011
ECO 341156

Figure F-29. Format Unit - FORMAT (Sheet 2 of 2) (Cont'd)



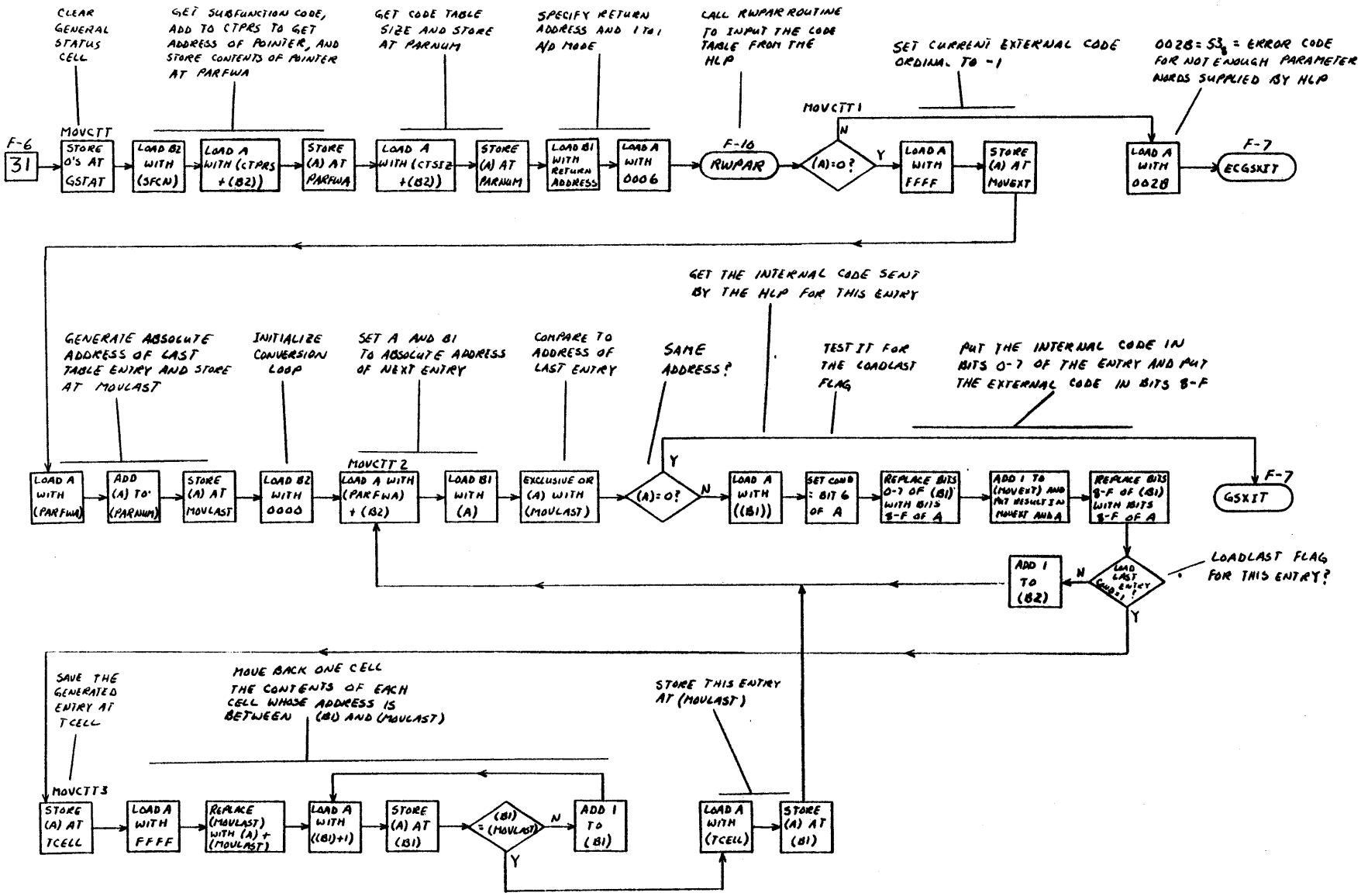
	0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F	
SPEEDTBL+0	1	0	0	0	0	0	0	0	0	0	1	0	0	1	1	1	(A) NEGATIVE FLAG & 27 ERROR CODE
+1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	100 IPS: PHYS 1 : LGCL 0
+2	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	150 IPS: PHYS 2 : LGCL 1
+3	1	0	0	0	0	0	0	0	0	0	1	0	0	1	1	1	(A) NEGATIVE FLAG & 27 ERROR CODE
+4	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	200 IPS: PHYS 4: LGCL 2
+5	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	1	250 IPS: PHYS 5 LGCL 3
+6	1	0	0	0	0	0	0	0	0	0	1	0	0	1	1	1	(A) NEGATIVE FLAG & 27 ERROR CODE
SPEEDTBL+7	1	0	0	0	0	0	0	0	0	0	1	0	0	1	1	1	(A) NEGATIVE FLAG & 27 ERROR CODE

PHYSICAL SPEED TO LOGICAL SPEED CONVERSION TABLE

NOTE: $27_{16} : 47_{8}$ = ERROR CODE FOR MISCELLANEOUS HARDWARE ERROR.

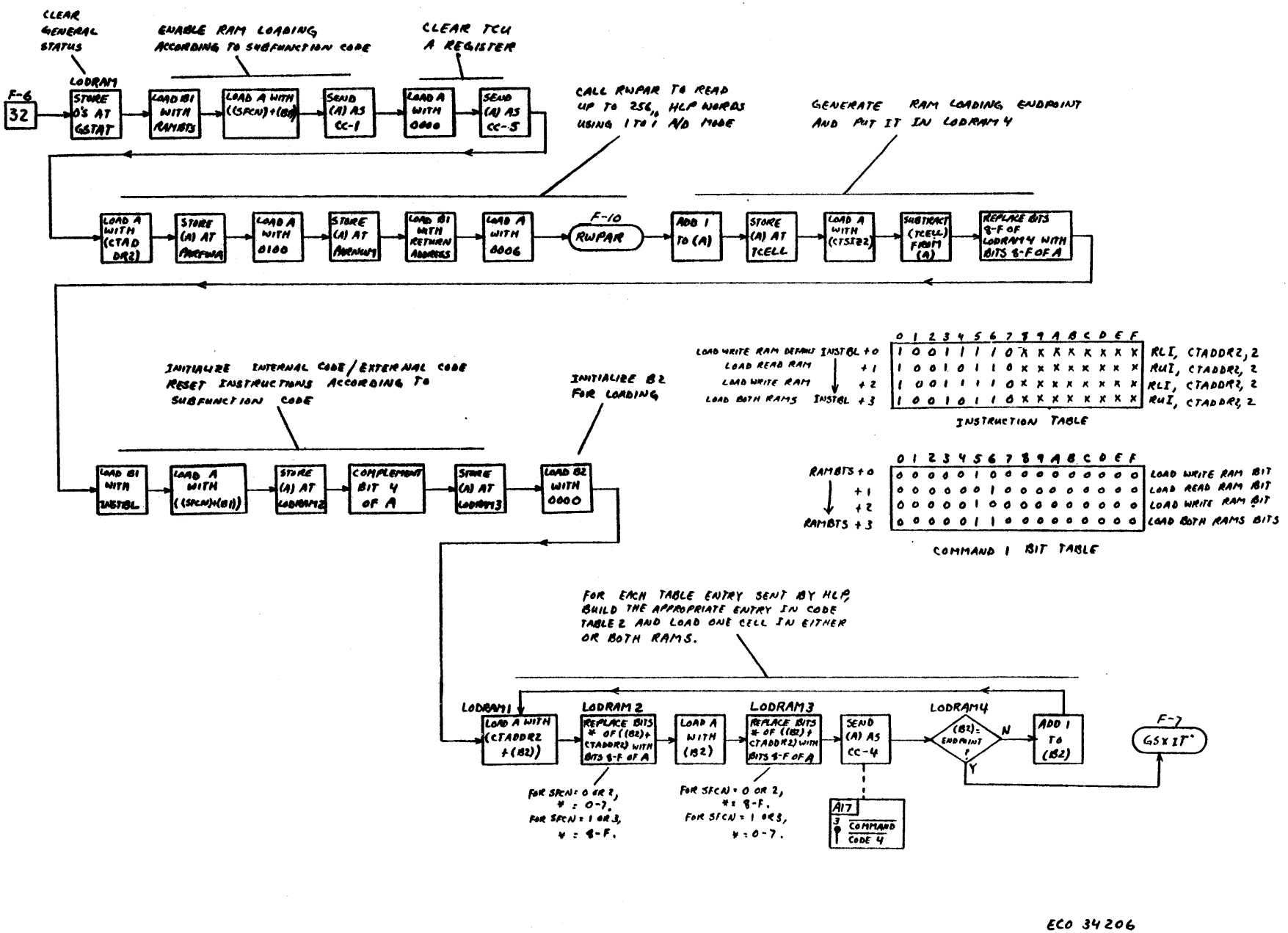
ECO 34081
ECO 34436

Figure F-30. Define Unit - DEFU00



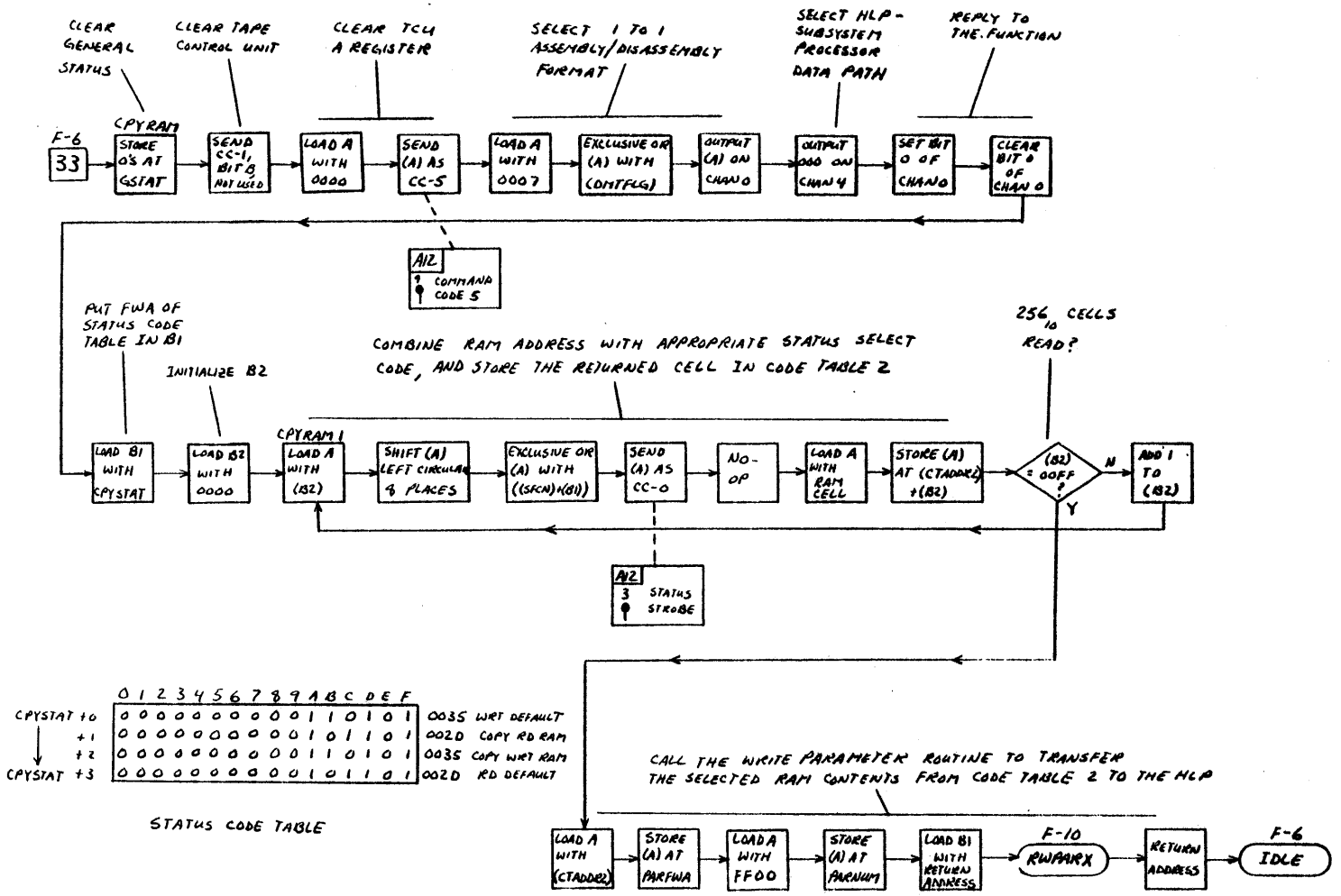
ECO 34081
ECO 34436

Figure F-31. Move Code Translation Table - MOVCTT



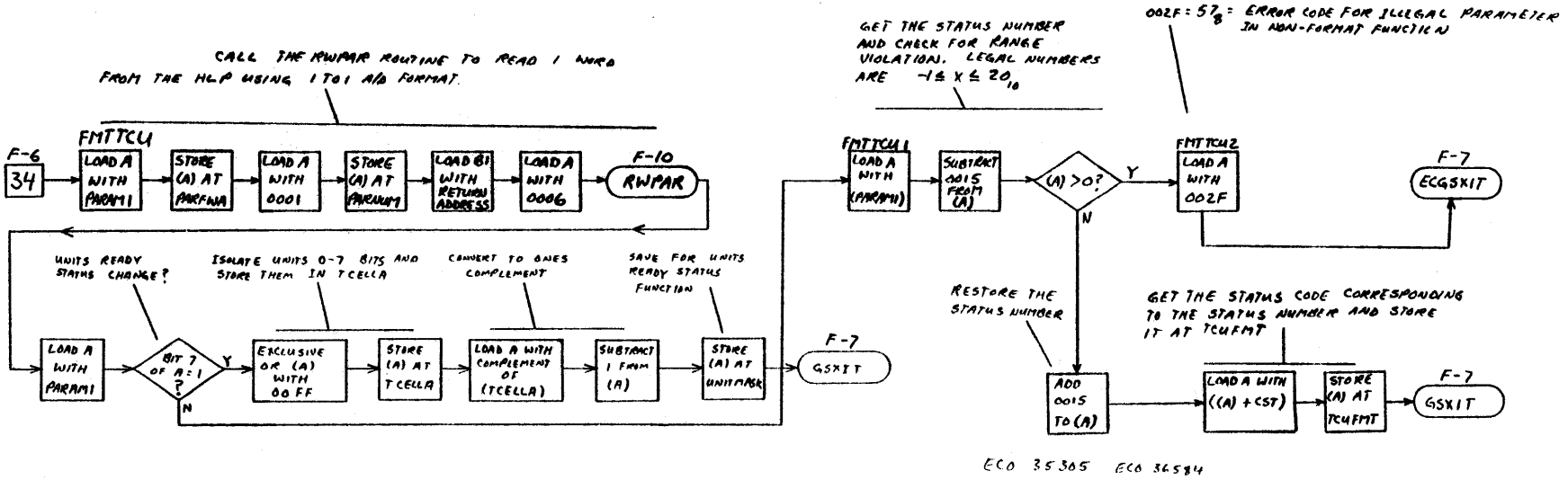
ECO 34206

Figure F-32. Load Read/Write RAM - LODRAM



ECO 34206
ECO 31442

Figure F-33. Copy Read/Write RAM - CPYRAM



ECO 35305 ECO 36584

Figure F-34. Format TCU/Coupler/Units Ready Status - FMTTCU

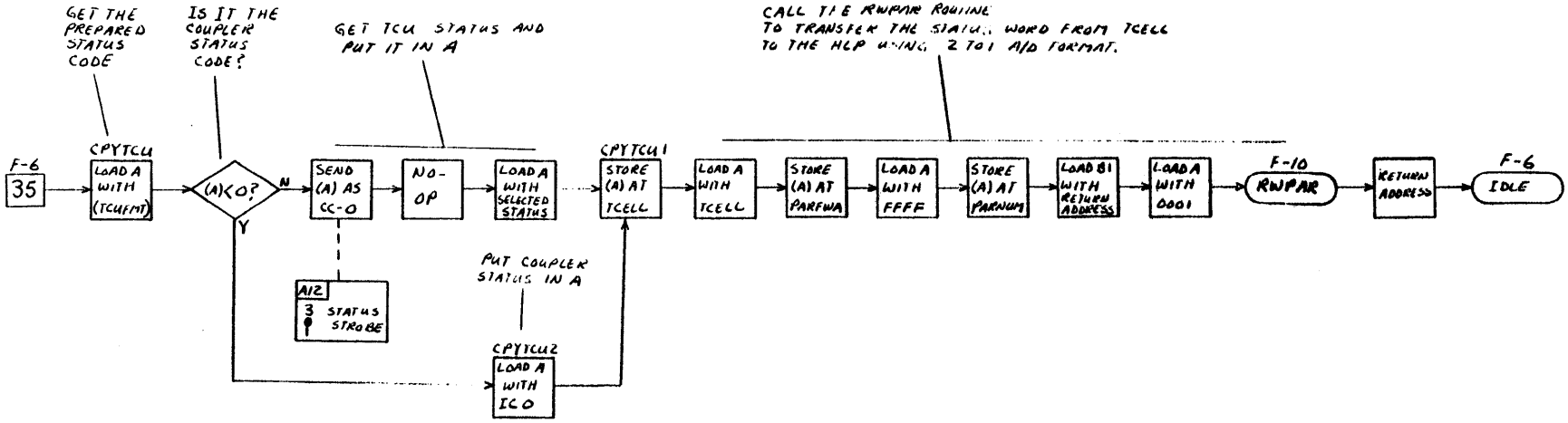
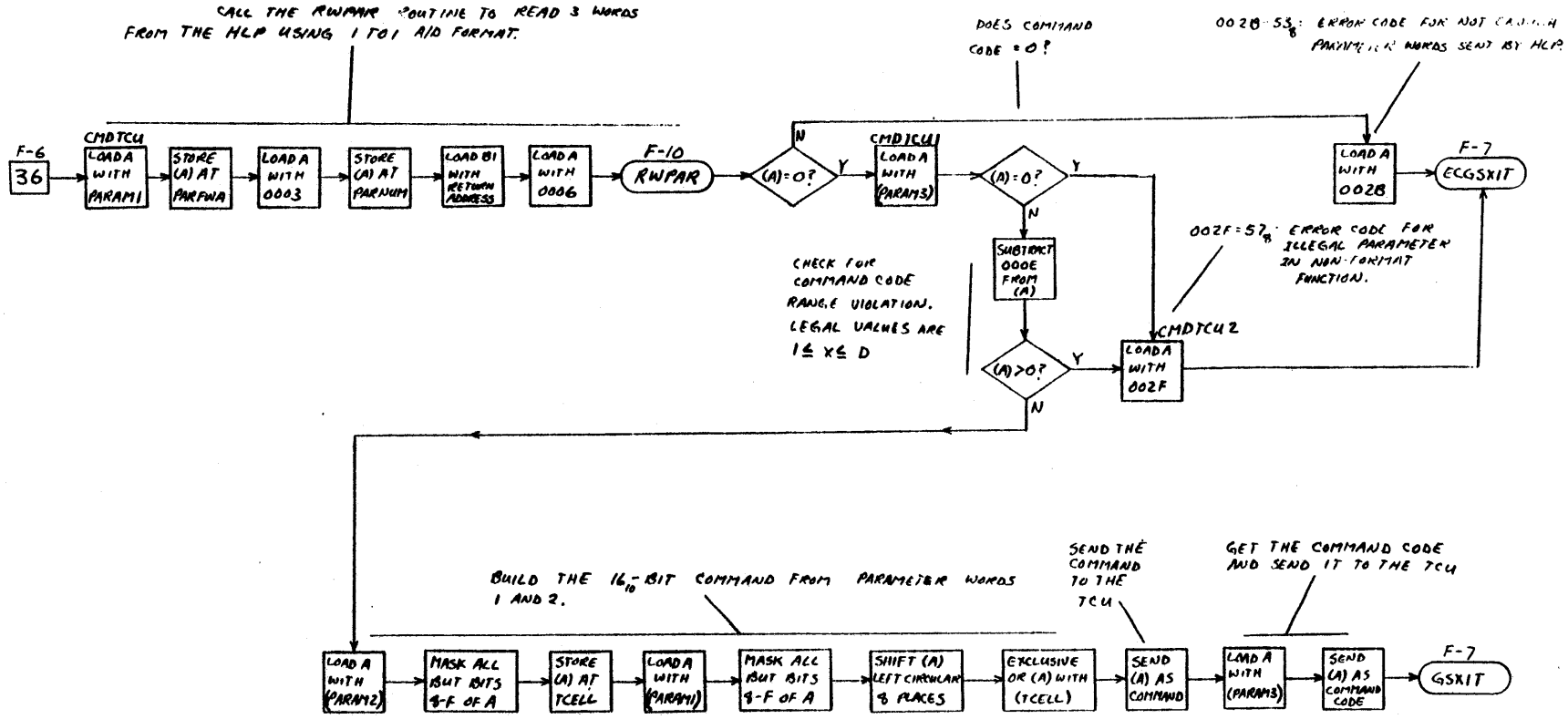
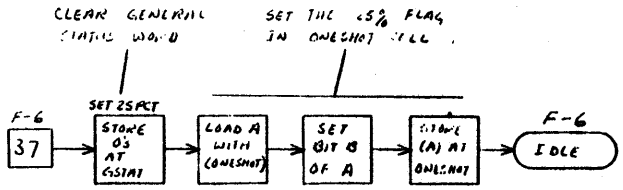


Figure F-35. Copy TCU/Coupler Status - CPYTCU



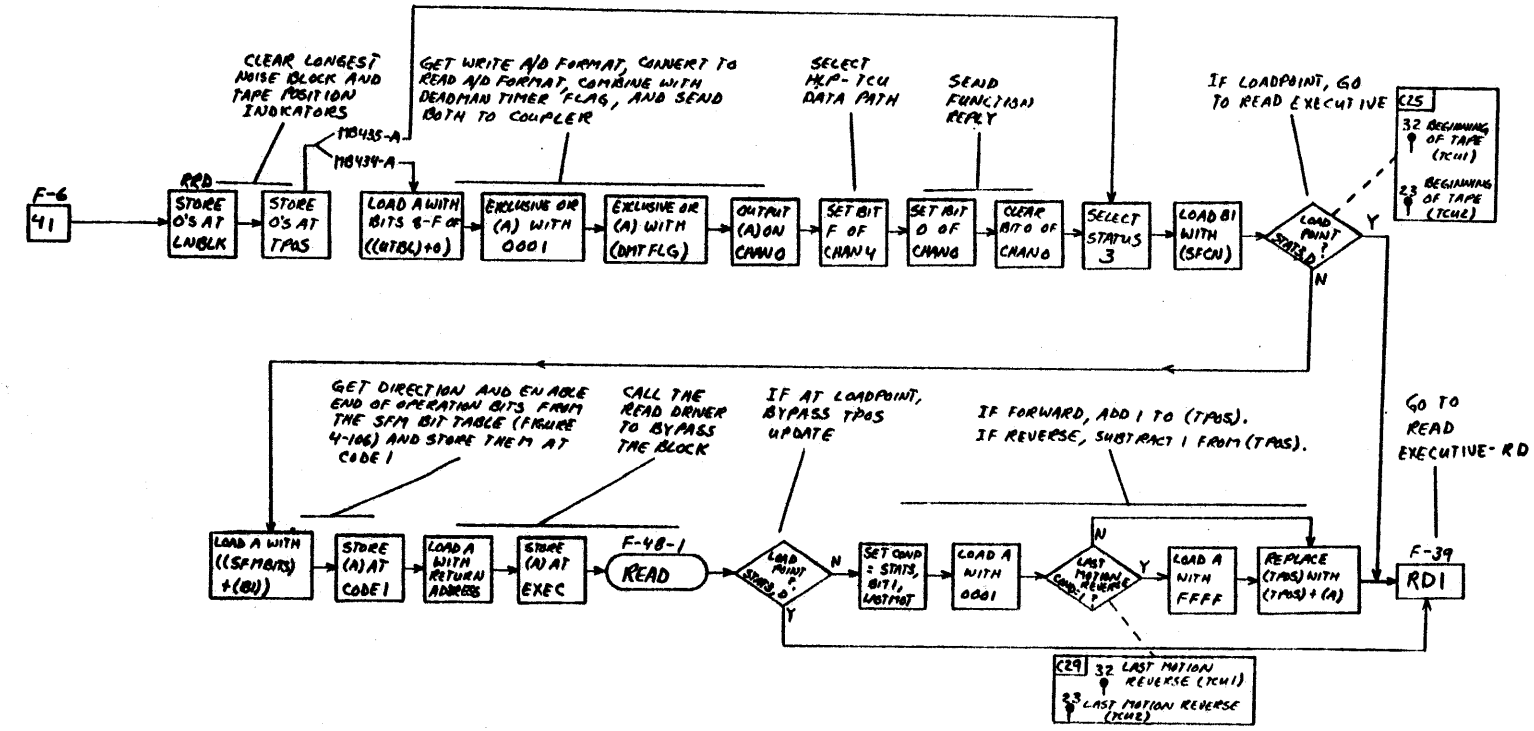
ECO 34031

Figure F-36. Issue TCU Command - CMDTCU



ECO 38303

Figure F-37. Set 25 Percent Sprocket Delay - SET25PCT



ECO 34436
ECO 36757

Figure F-38. Reread - RRD

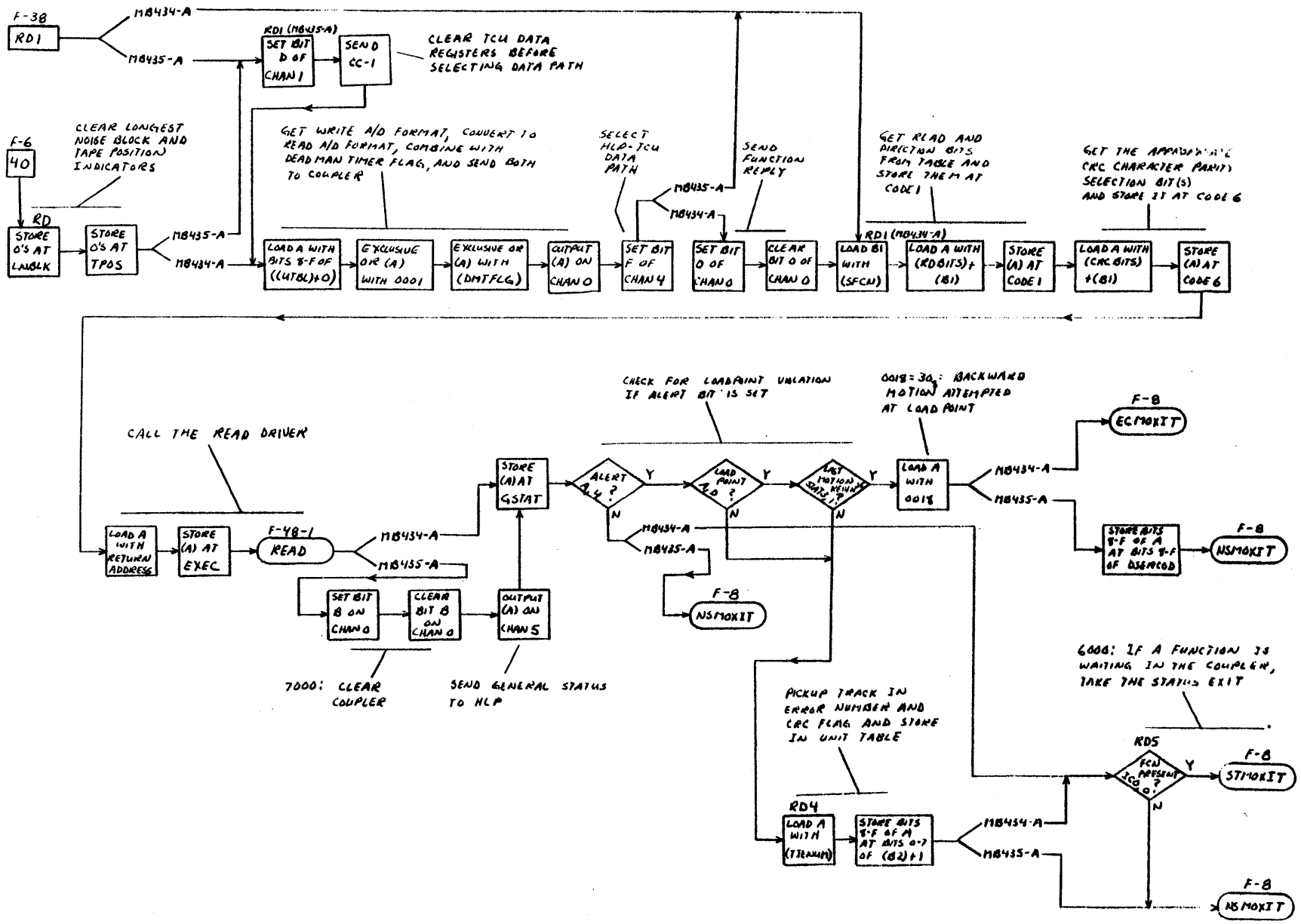


Figure F-39. Read Executive - RD

ECO 341206
ECO 341436
ECO 341517

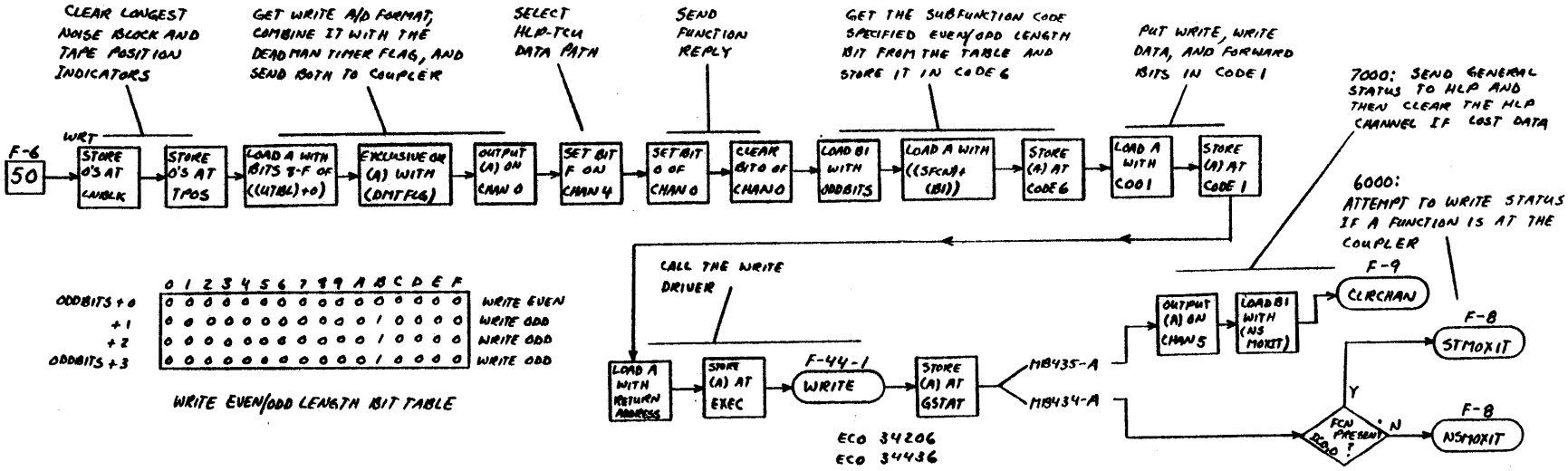
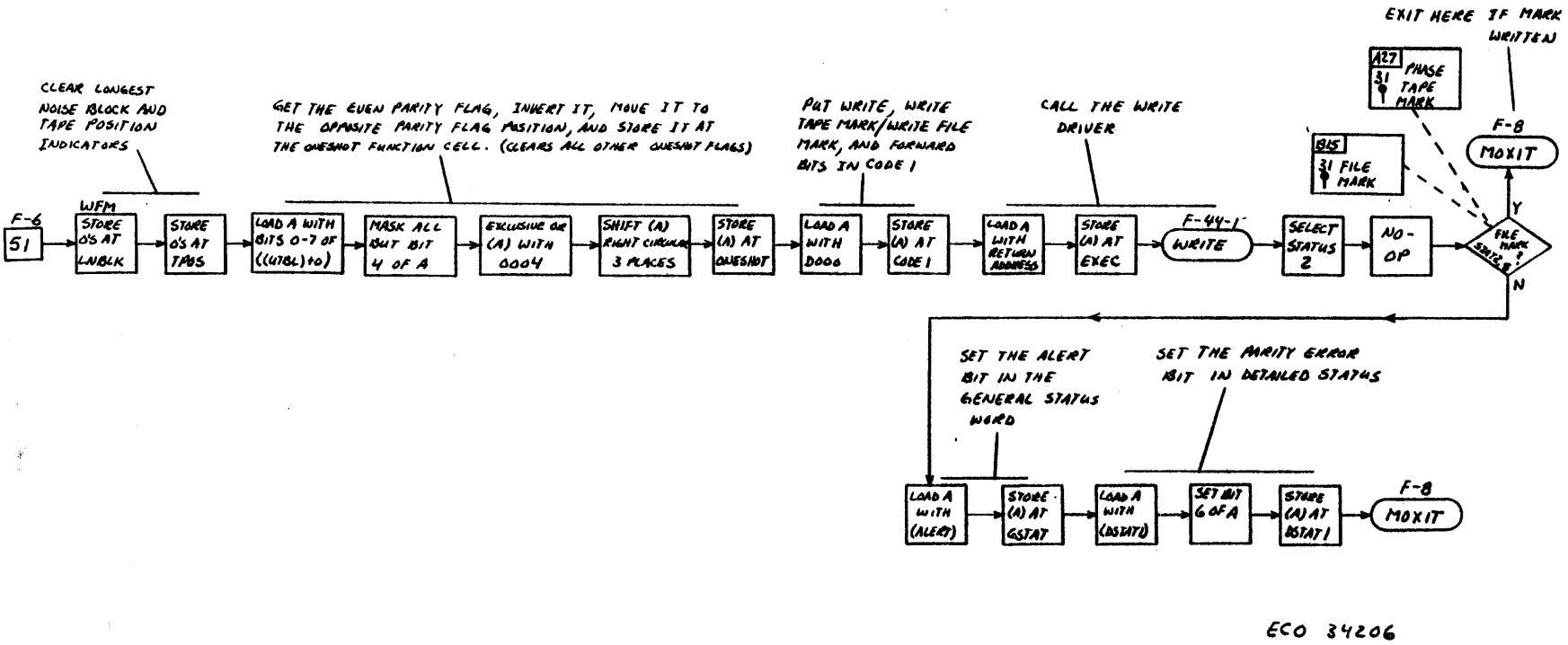
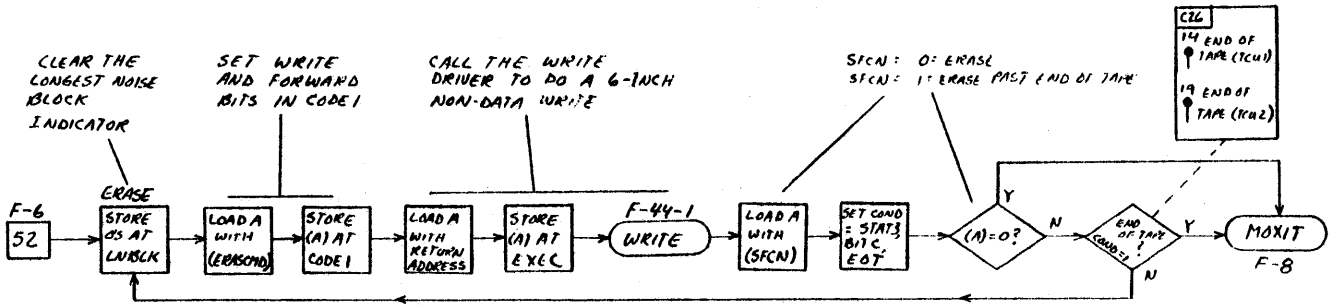


Figure F-40. Write Executive - WRT



ECO 34206

Figure F-41. Write File Mark - WFM



ECO 341206

Figure F-42. Erase/Erase Past End of Tape - ERASE

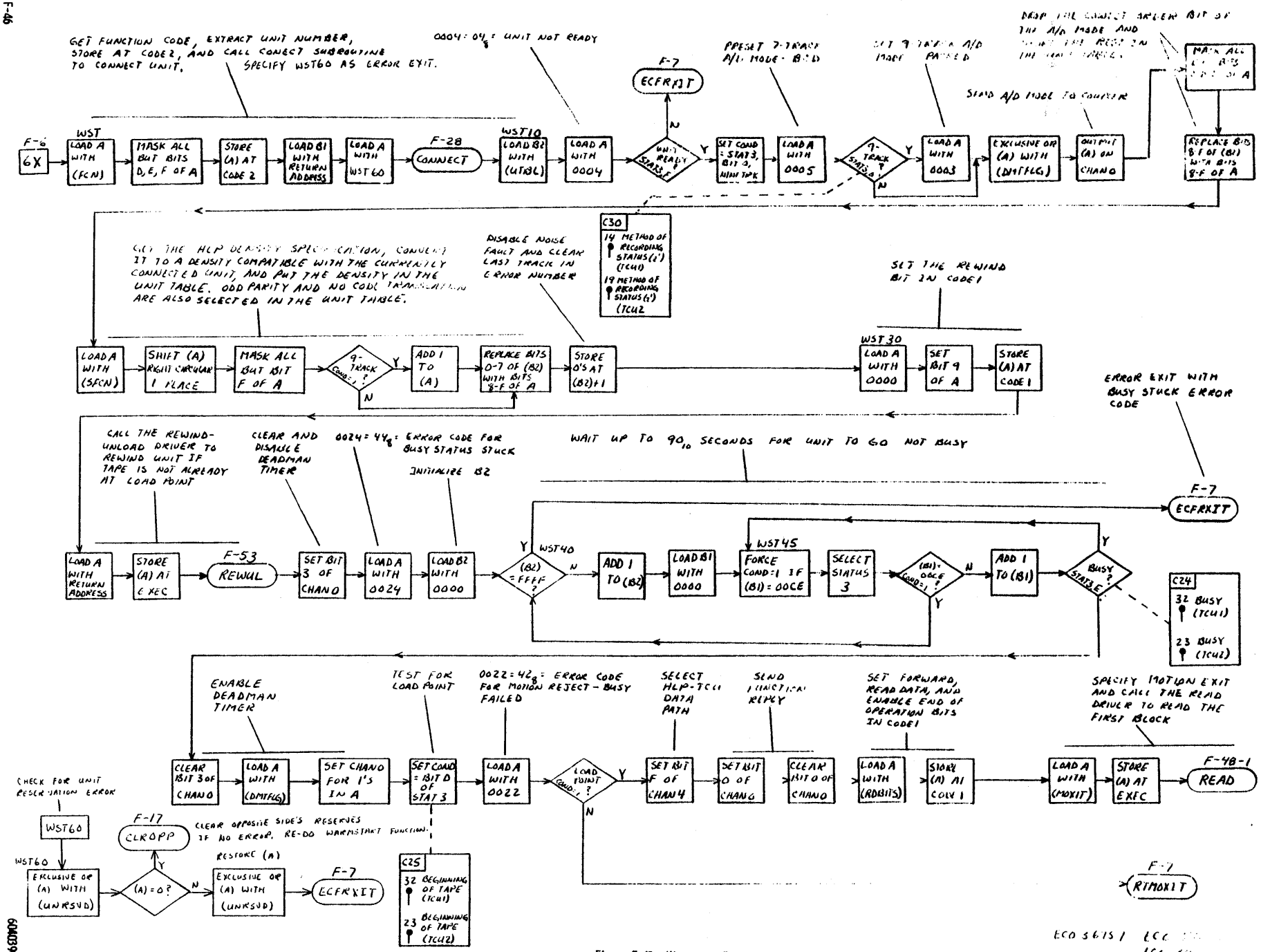


Figure F-43. Warmstart Function Processor - WST

ECO 56151 ECC
ECO 54
ECO 55306

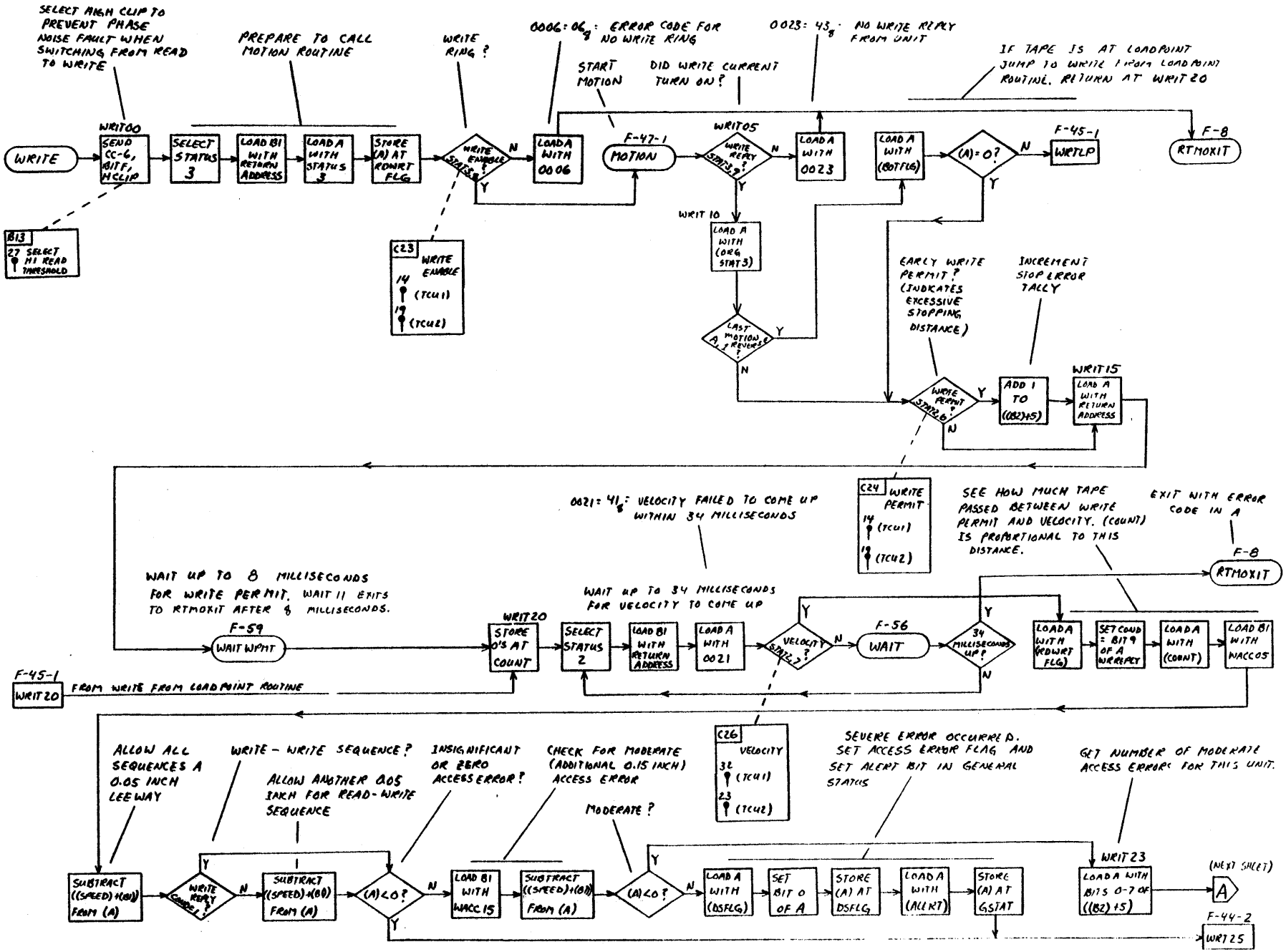


Figure F-44. Write Driver - WRIT00 (Sheet 1 of 3)

ECO 36757 ECO 35982 ECO 34081
ECO 38303 ECO 35305 ECO 34436

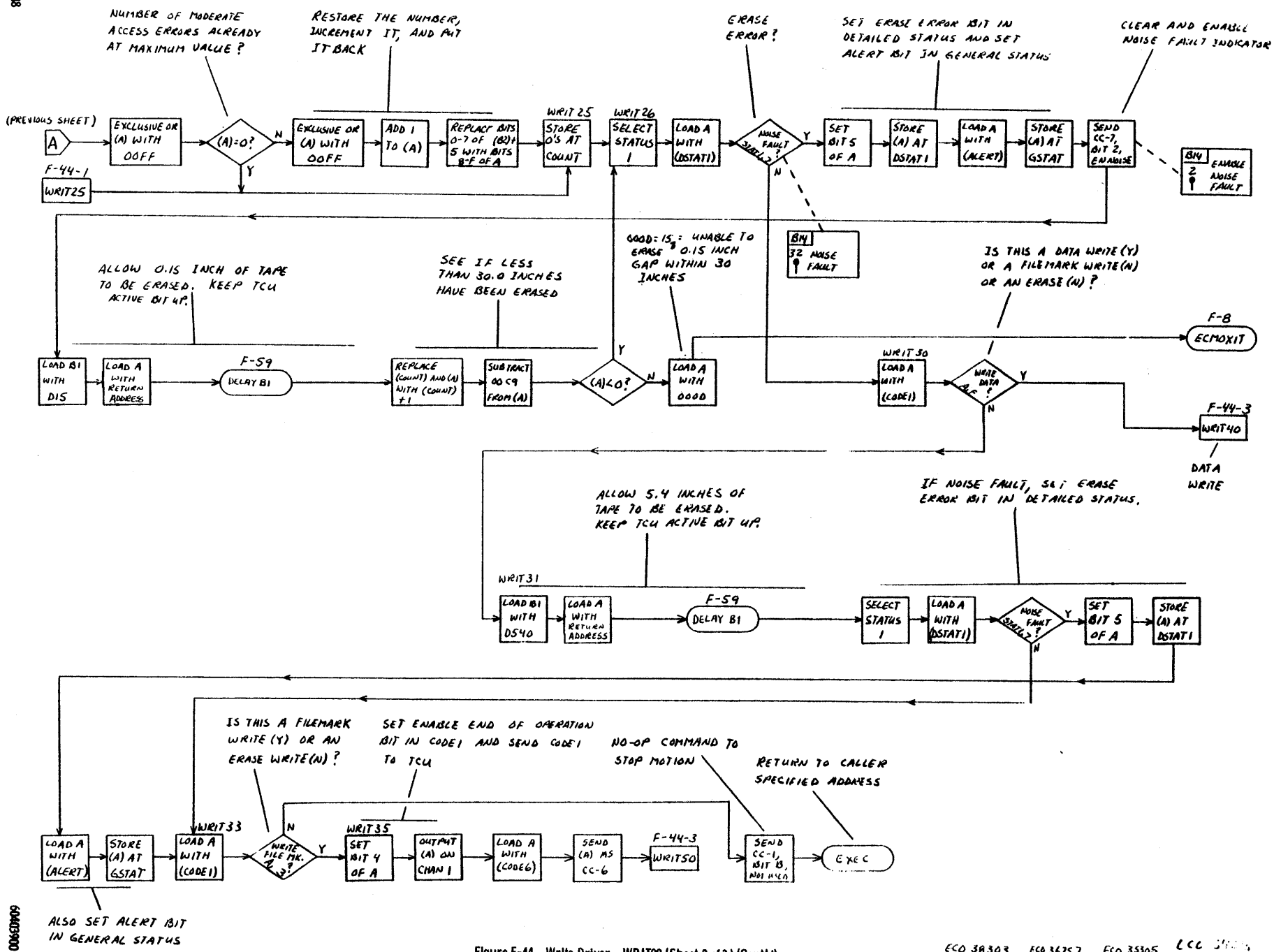
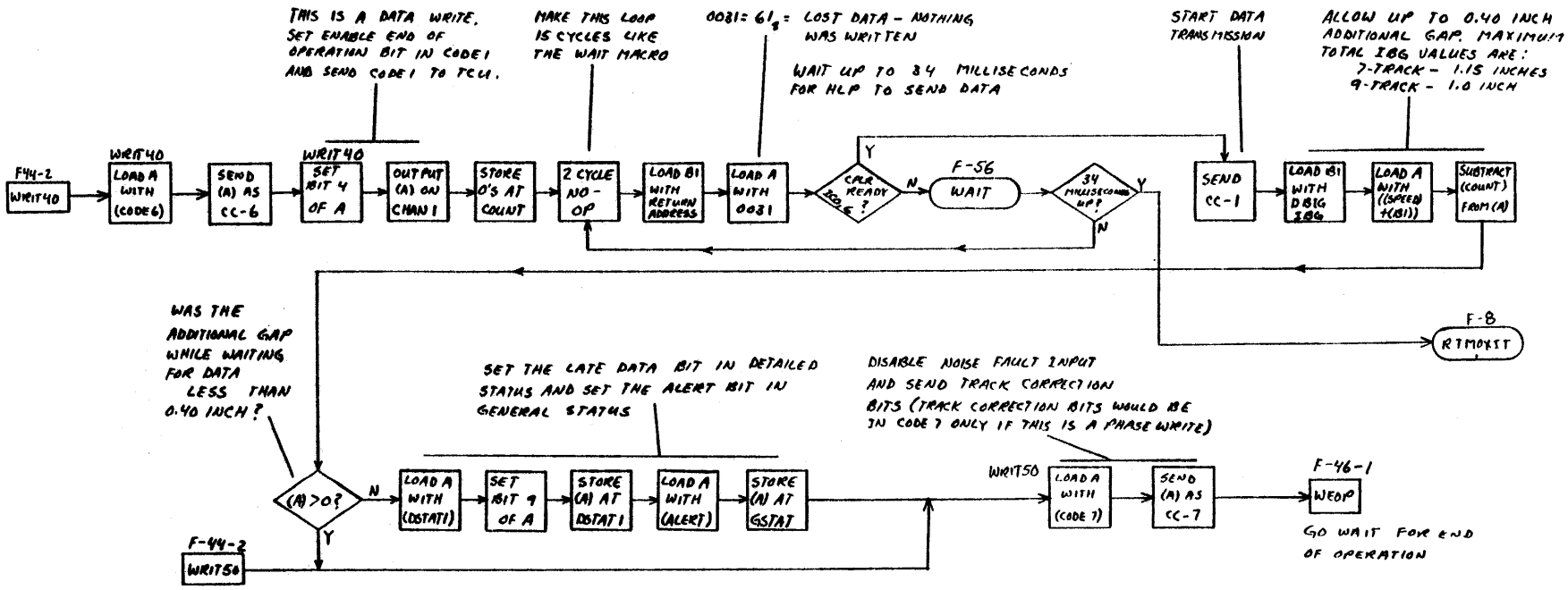


Figure F-44. Write Driver - WRIT00 (Sheet 2 of 3) (Cont'd)

ECO 38303 ECO 38979 ECO 36757 ECO 37442 ECO 35305 ECO 35982 ECO [illegible] ECO 34436



15-CYCLE DELAY TABLES

0.05 INCH DELAY TABLE

	0 1 2 3 4 5 6 7 8 9 A B C D E F	
WACC05+0	0 0 0 0 0 0 0 0 0 0 0 1 1 1 1 0	001E = 100 IPS
+1	0 0 0 0 0 0 0 0 0 0 0 1 0 1 0 0	0014 = 150 IPS
+2	0 0 0 0 0 0 0 0 0 0 0 0 1 1 1 1	000F = 200 IPS
WACC05+3	0 0 0 0 0 0 0 0 0 0 0 0 1 1 0 0	000C = 250 IPS

0.15 INCH DELAY TABLE

	0 1 2 3 4 5 6 7 8 9 A B C D E F	
WACC15+0	0 0 0 0 0 0 0 0 0 1 0 1 1 0 1 1	005B = 100 IPS
+1	0 0 0 0 0 0 0 0 0 0 1 1 1 1 0 1	003D = 150 IPS
+2	0 0 0 0 0 0 0 0 0 0 1 0 1 1 0 1	002D = 200 IPS
WACC15+3	0 0 0 0 0 0 0 0 0 0 1 0 0 1 0 0	0024 = 250 IPS

0.40 INCH DELAY TABLE

	0 1 2 3 4 5 6 7 8 9 A B C D E F	
DBIGIBG+0	0 0 0 0 0 0 0 1 1 1 1 1 0 0 1 0	00F2 = 100 IPS
+1	0 0 0 0 0 0 0 0 1 0 1 0 0 0 0 1	00A1 = 150 IPS
+2	0 0 0 0 0 0 0 0 0 1 1 1 1 0 0 1	0079 = 200 IPS
DBIGIBG+3	0 0 0 0 0 0 0 0 0 1 1 0 0 0 0 1	0061 = 250 IPS

Figure F-44. Write Driver - WRITOO (Sheet 3 of 3) (Cont'd)

ECO 34081 ECO 36303
ECO 34446 ECO 38979

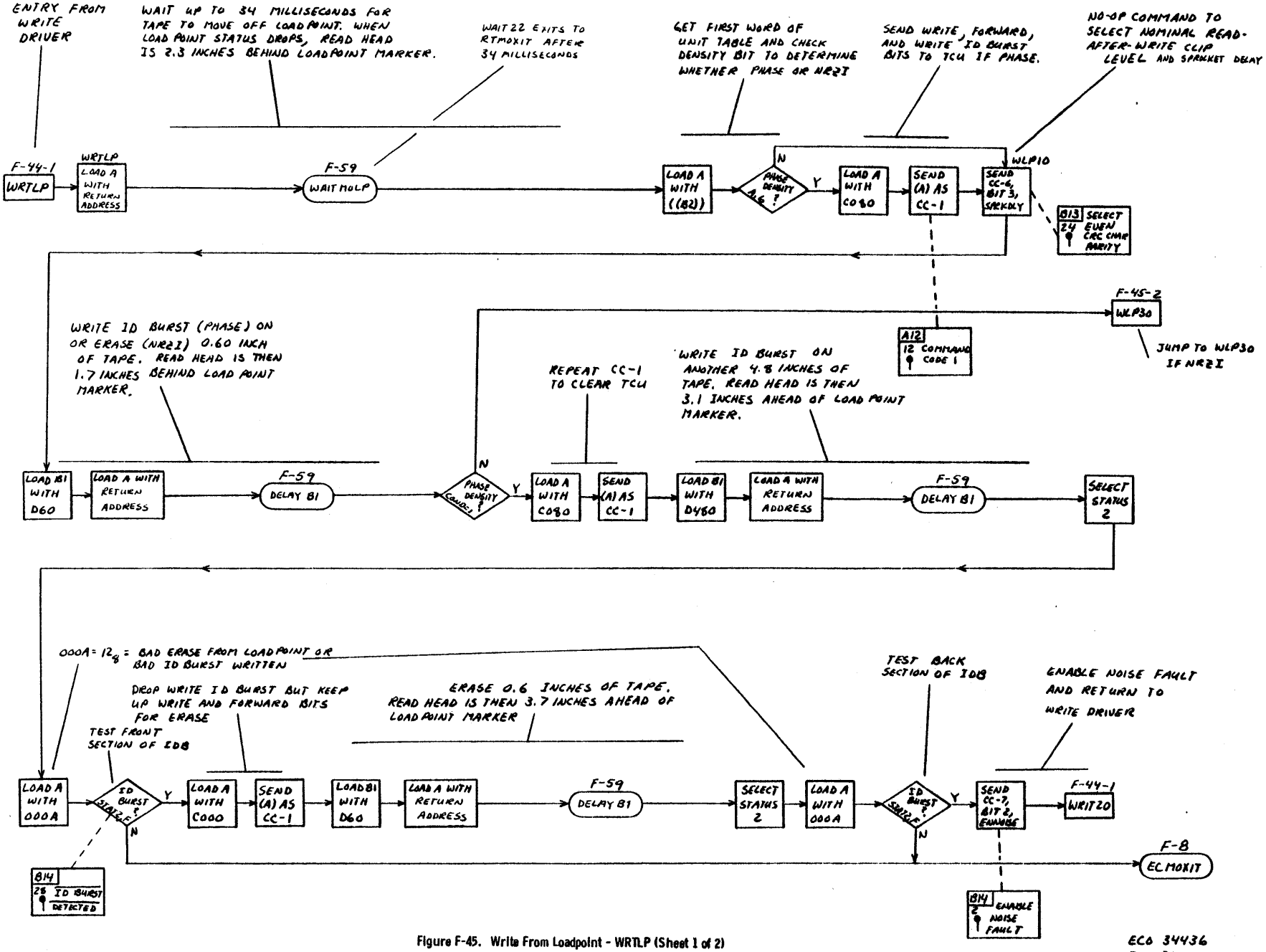
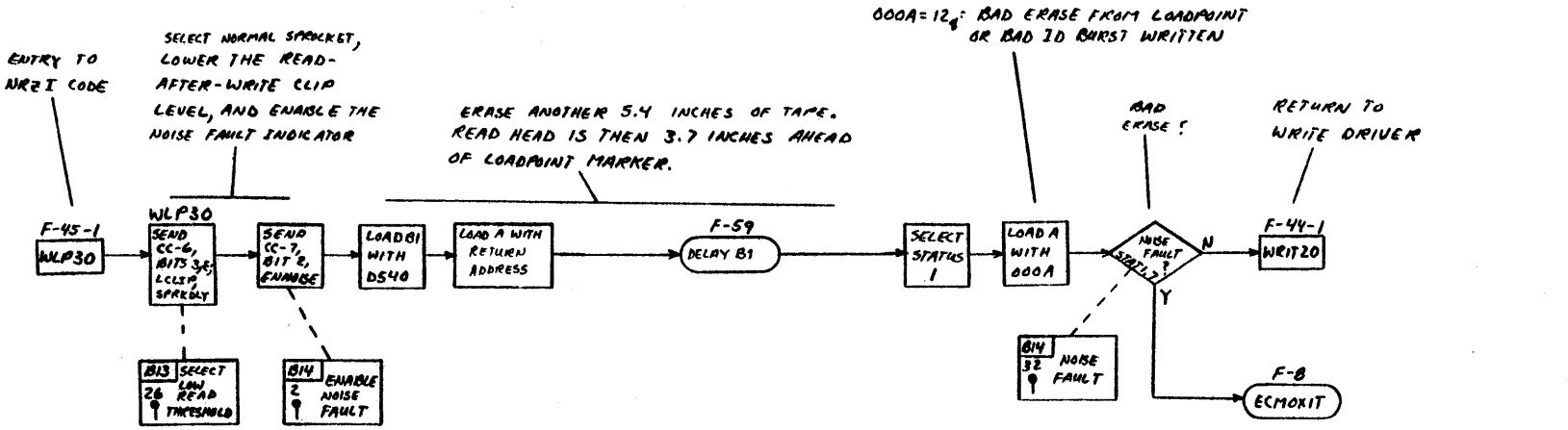
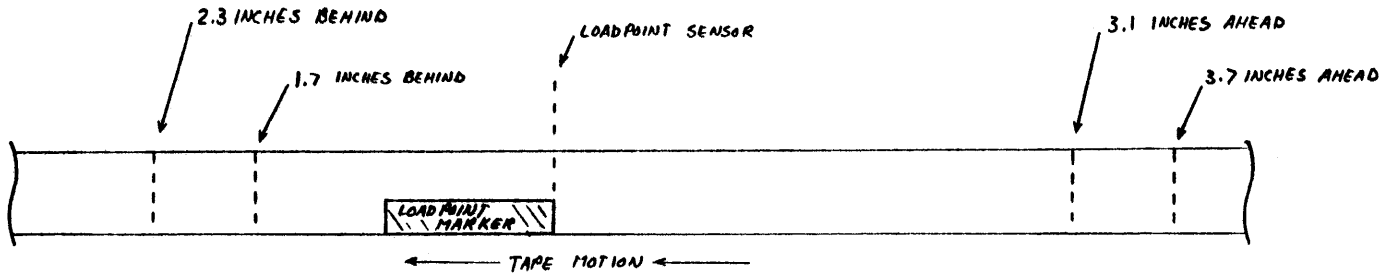


Figure F-45. Write From Loadpoint - WRTLP (Sheet 1 of 2)

ECO 34436
ECO 35982
ECO 36757





READ HEAD POSITIONS DURING WRITE FROM LOADPOINT

ECO 34436
ECO 35982
ECO 36757

Figure F-45. Write From Loadpoint - WRTLP (Sheet 2 of 2) (Cont'd)

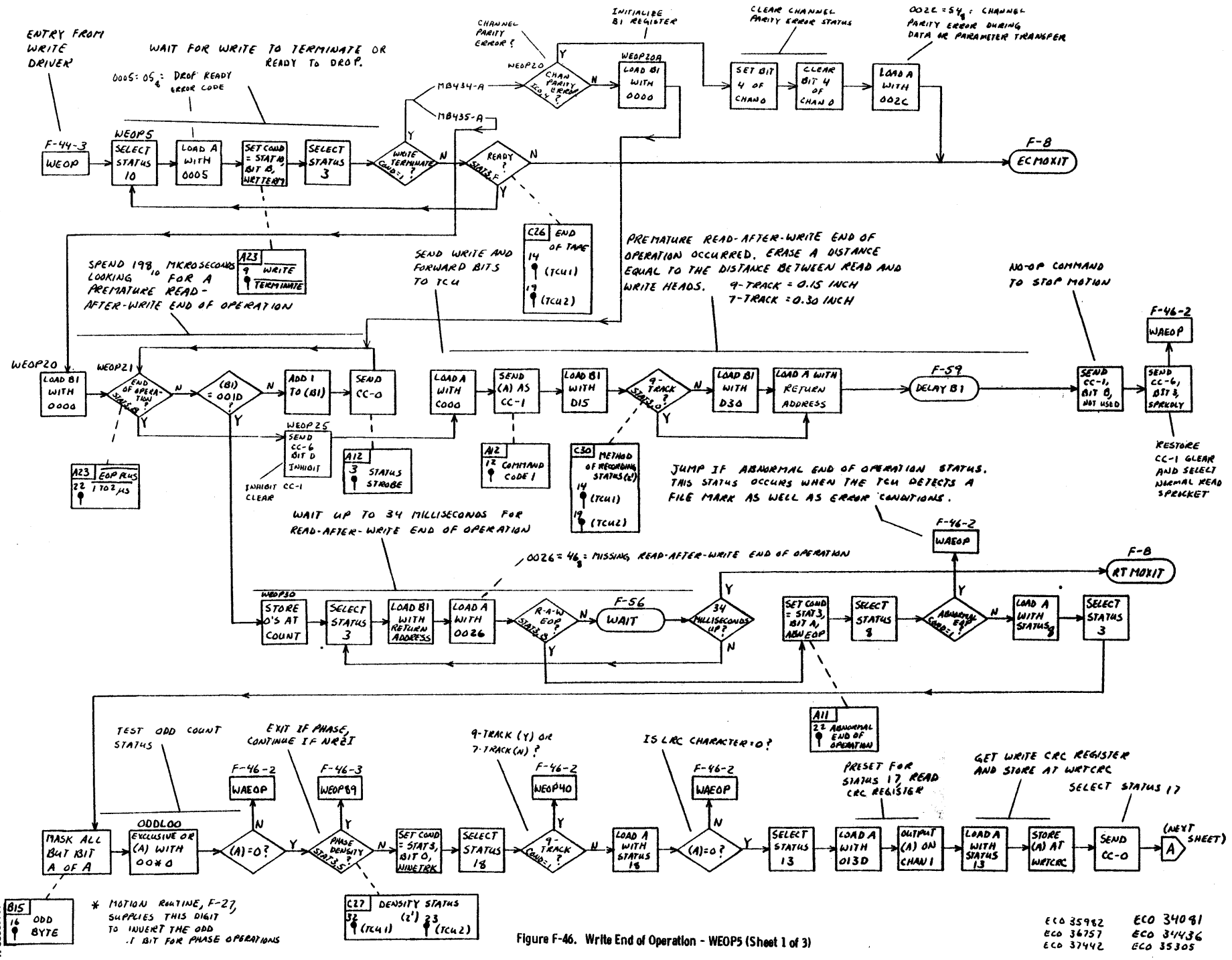


Figure F-46. Write End of Operation - WEOP5 (Sheet 1 of 3)

ECO 35982 ECO 34091
ECO 36757 ECO 34436
ECO 37442 ECO 35305

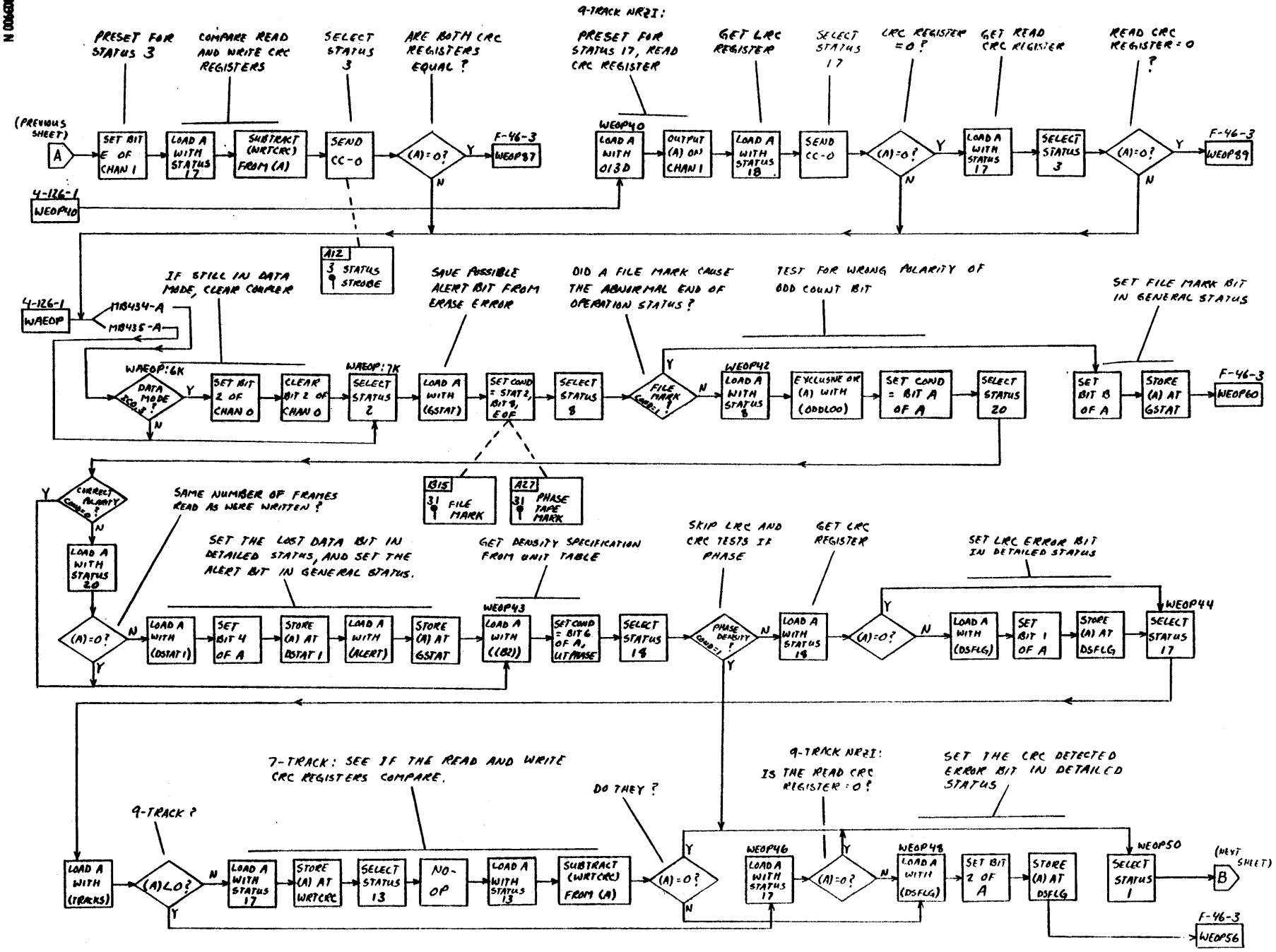


Figure F-46. Write End of Operation - WEOP5 (Sheet 2 of 3) (Cont'd)

ECO 34081
ECO 34436

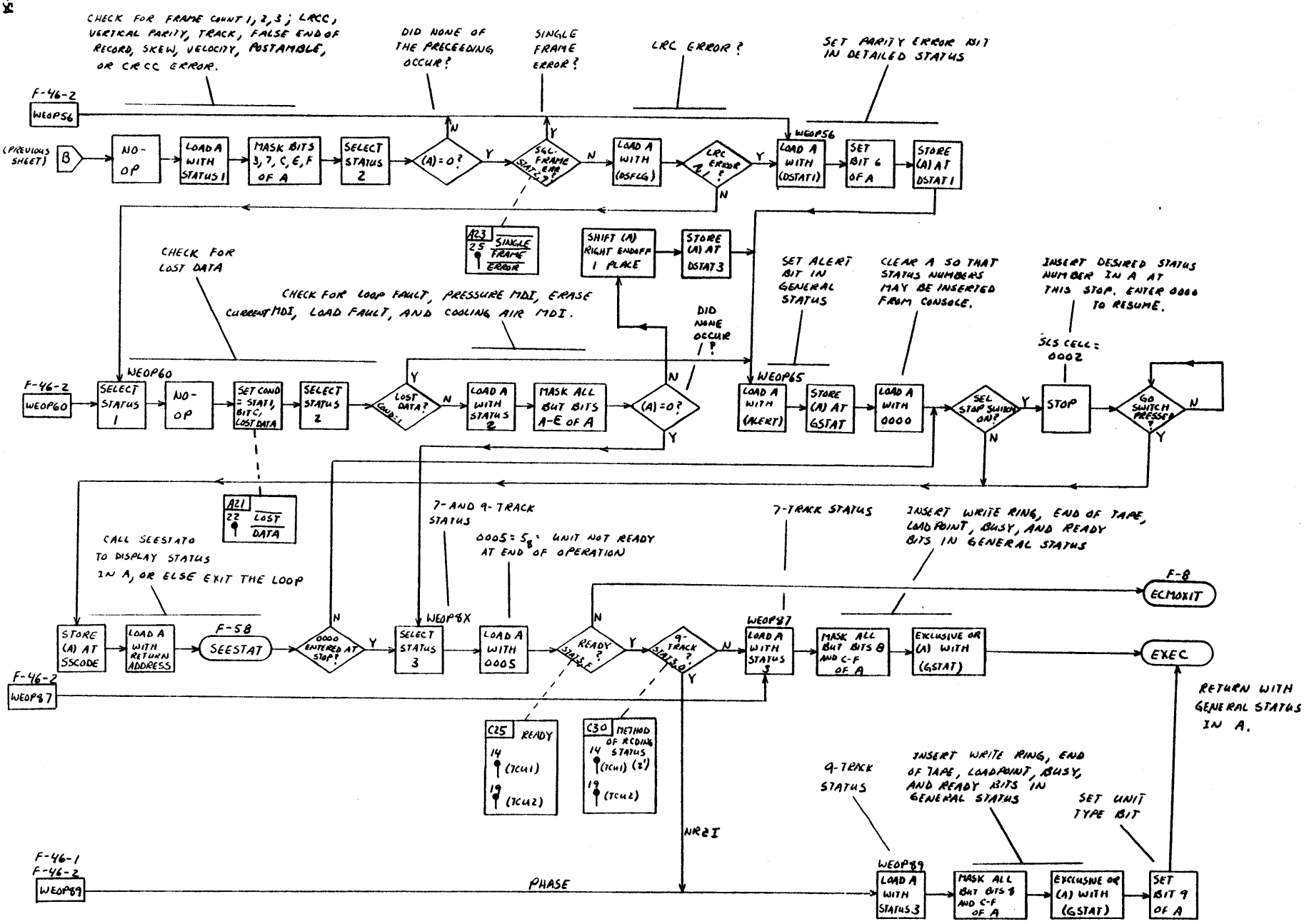


Figure F-46. Write End of Operation - WEOP5 (Sheet 3 of 3) (Cont'd)

ECO 34081
ECO 34136

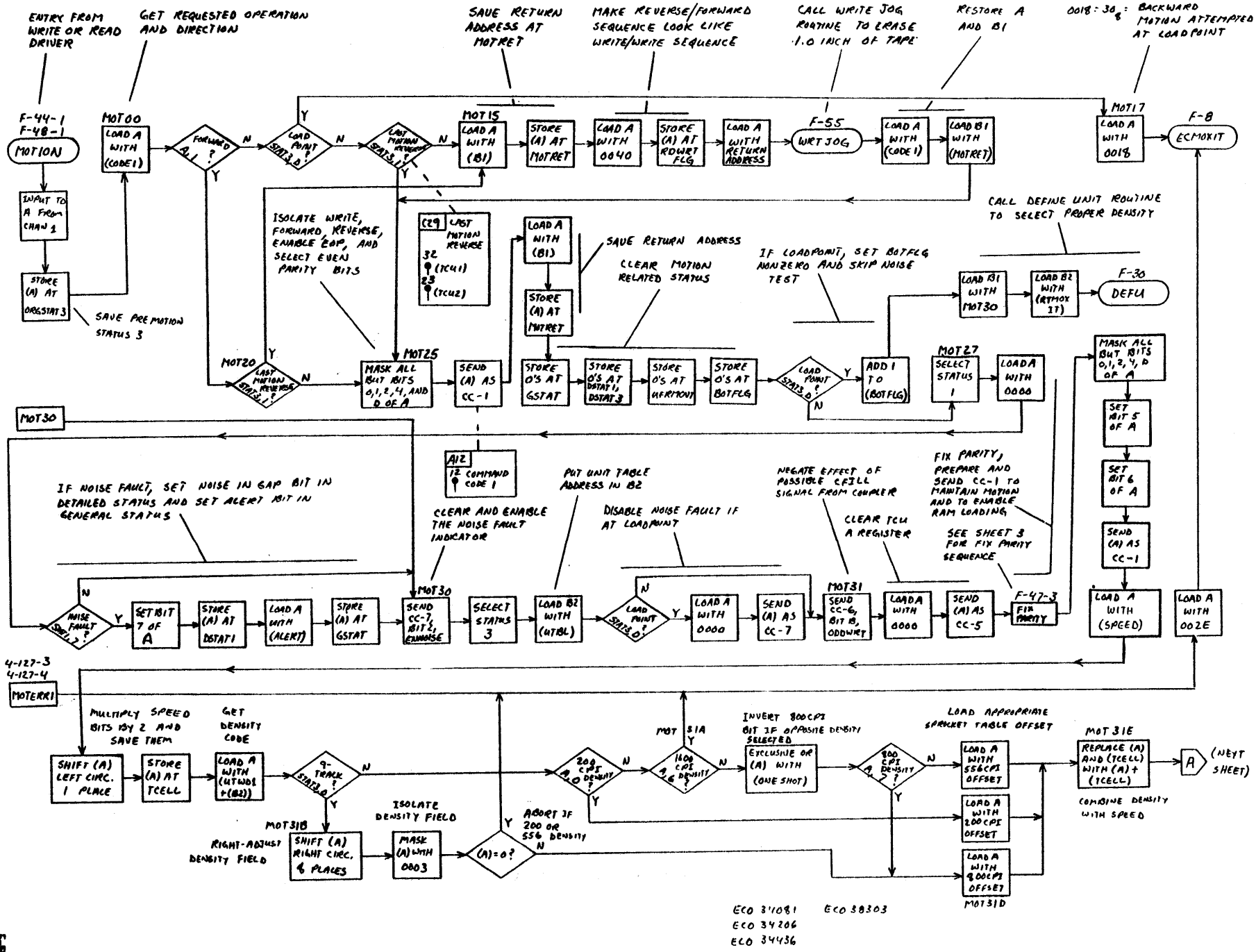
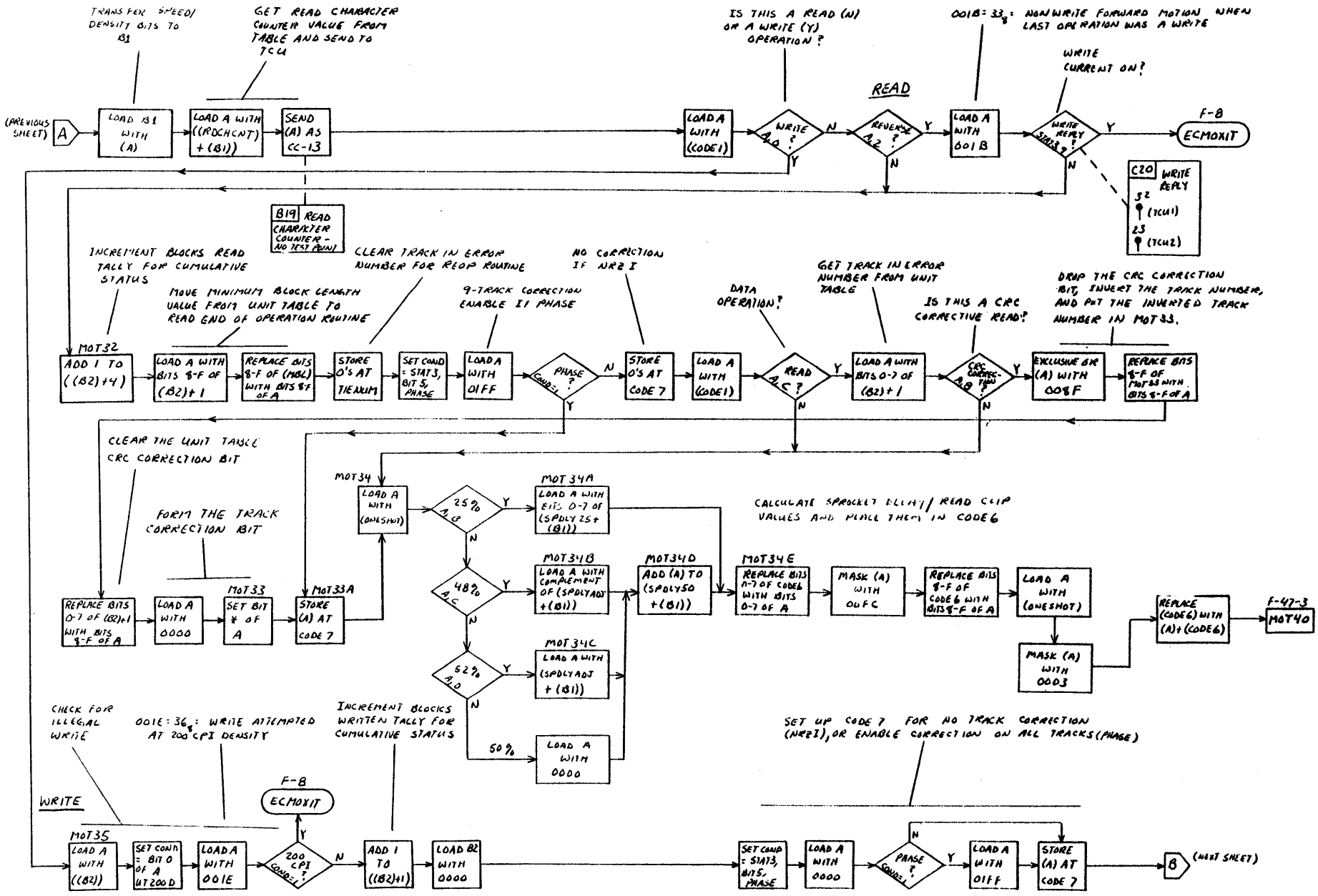


Figure F-47. Motion - MOT00 (Sheet 1 of 5)



ECO 3410B1 ECO 38303
ECO 341206
ECO 341136

Figure F-47. Motion - MOT00 (Sheet 2 of 5) (Cont'd)

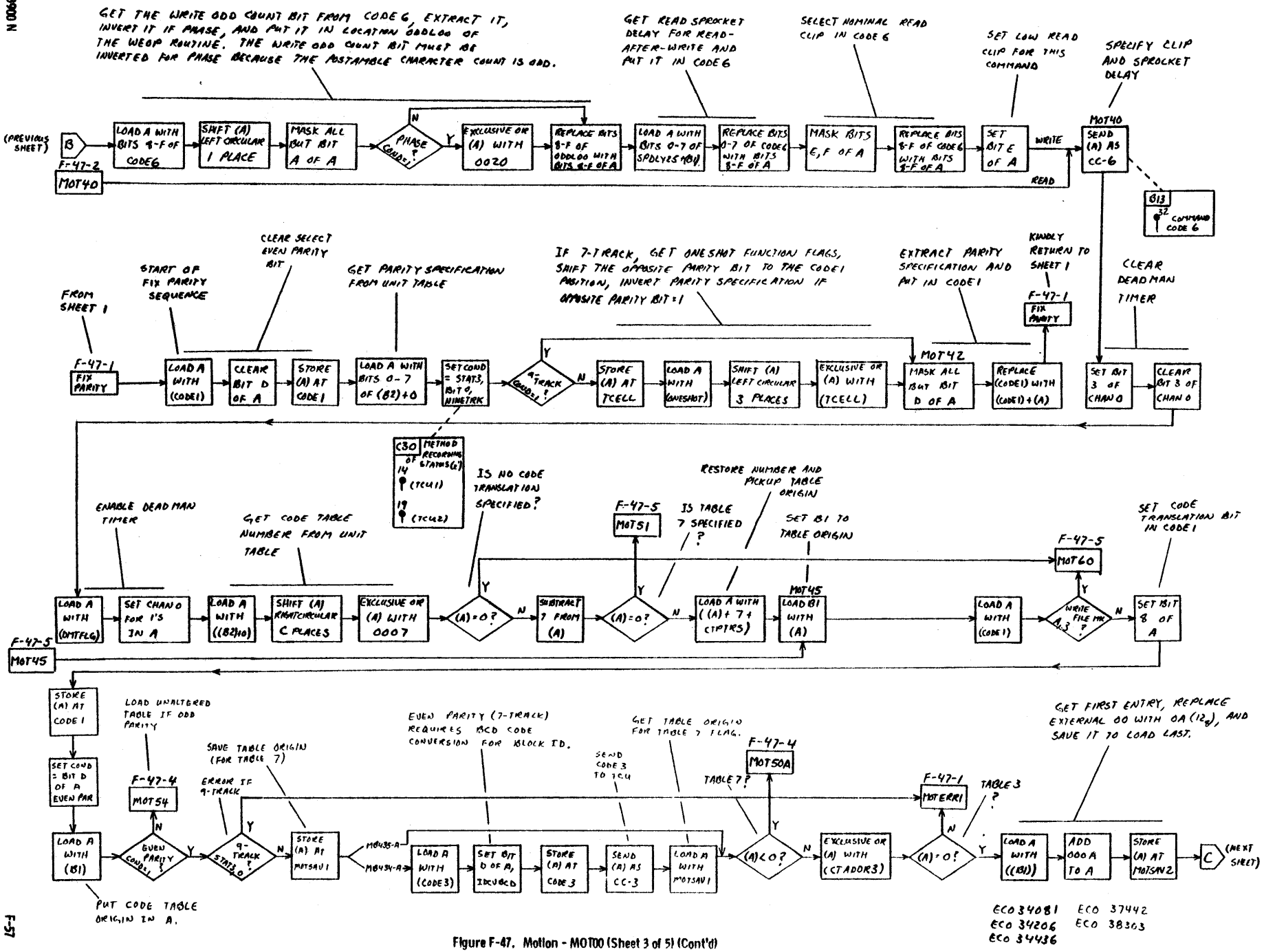
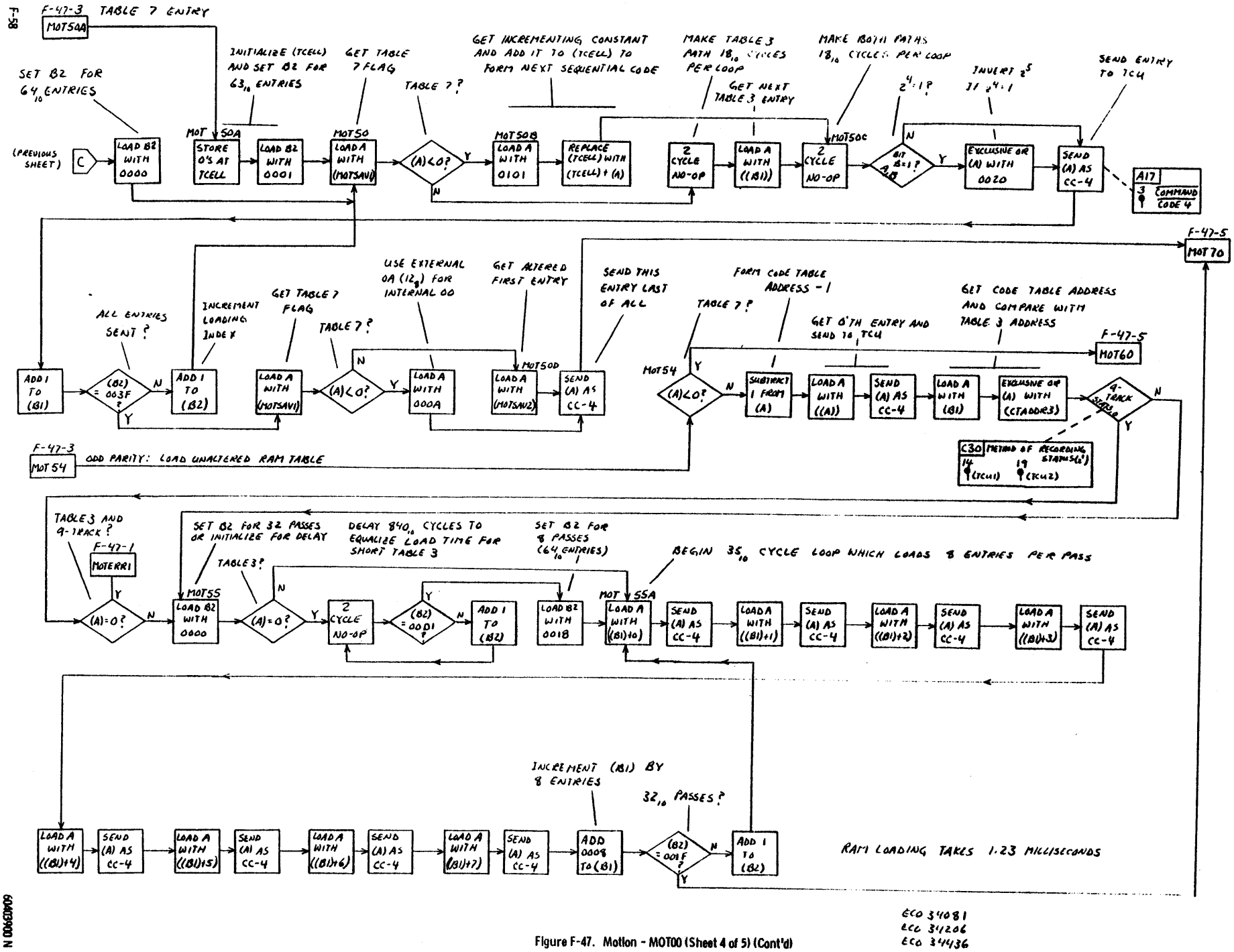


Figure F-47. Motion - MOT00 (Sheet 3 of 5) (Cont'd)



ECO 34081
ECO 34206
ECO 34436

Figure F-47. Motion - MOT00 (Sheet 4 of 5) (Cont'd)

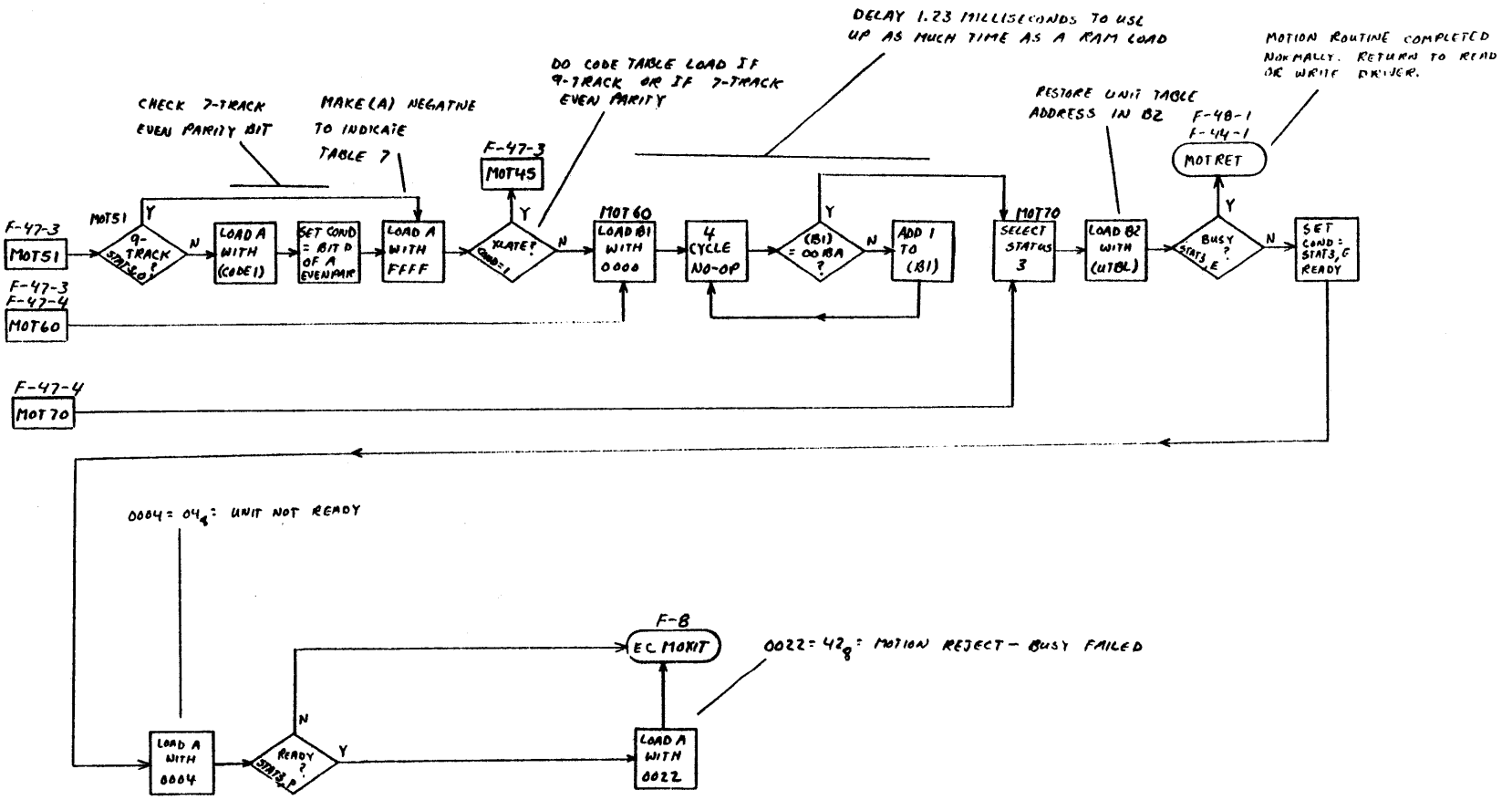


Figure F-47. Motion - MOT00 (Sheet 5 of 5) (Cont'd)

ECO 34081
ECO 31206
ECO 34136
ECO 55305

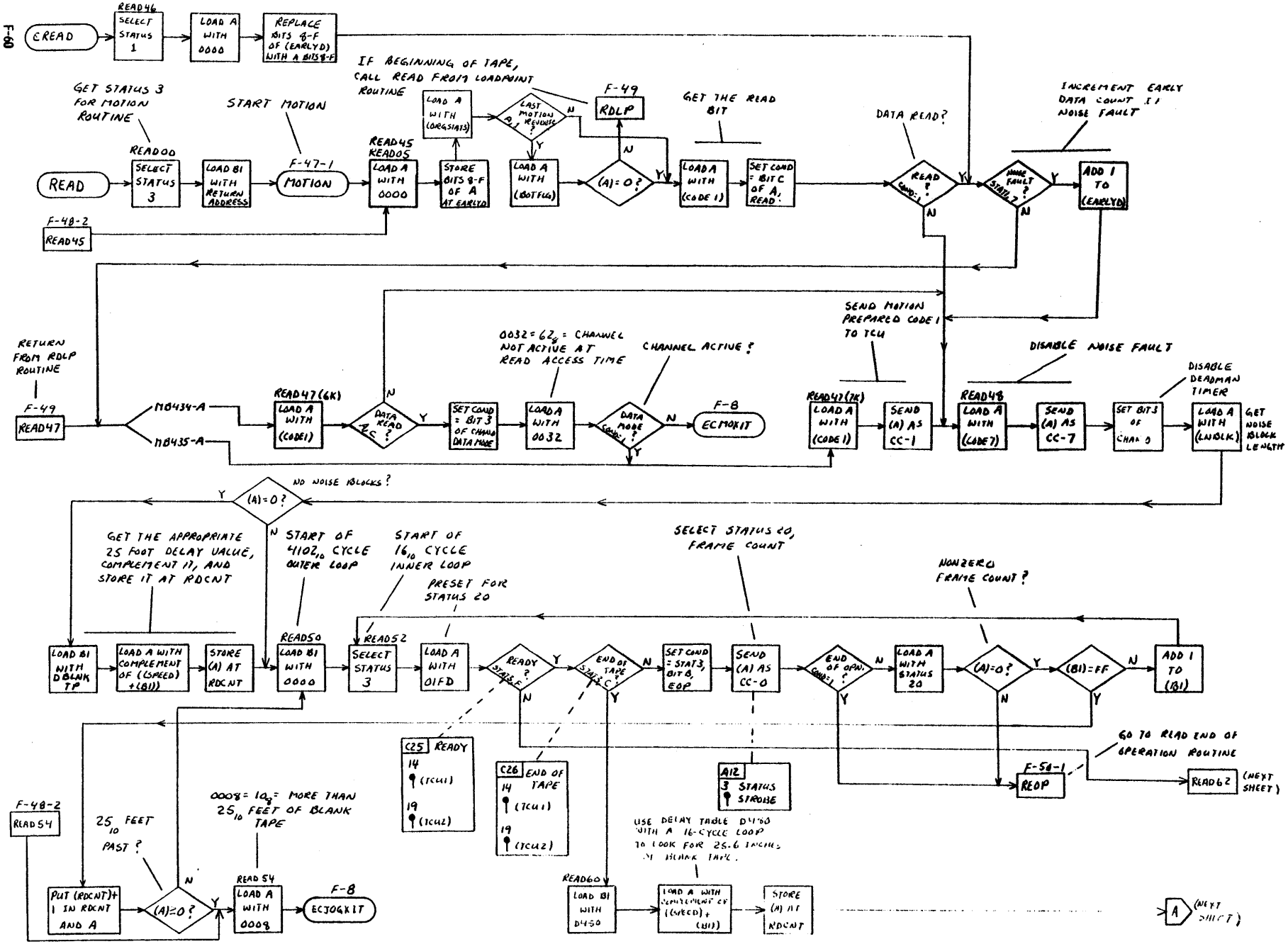
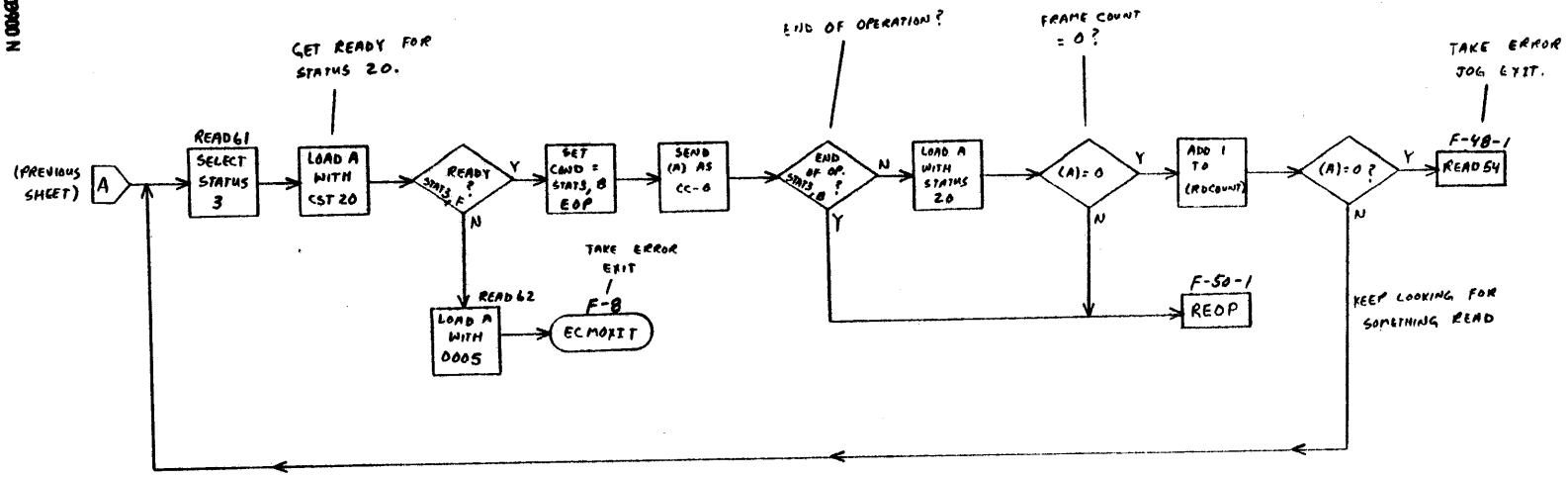
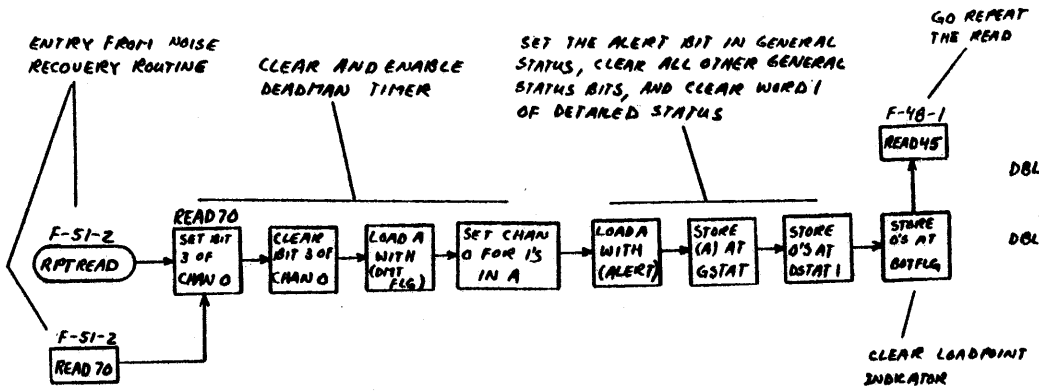


Figure F-48. Read Driver - READ00 (Sheet 1 of 2)

ECO 35982 ECO 34081
ECO 36757 ECO 34436
ECO 38303 ECO 35305





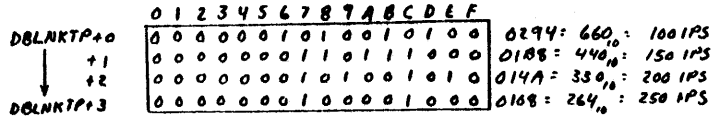


TABLE OF NUMBER OF 4101_{10} CYCLE LOOPS REQUIRED FOR 25$_{10}$ FEET OF TAPE

Figure F-48. Read Driver - READ00 (Sheet 2 of 2) (Cont'd)

LC034081
ECO 34436
FCO 35305

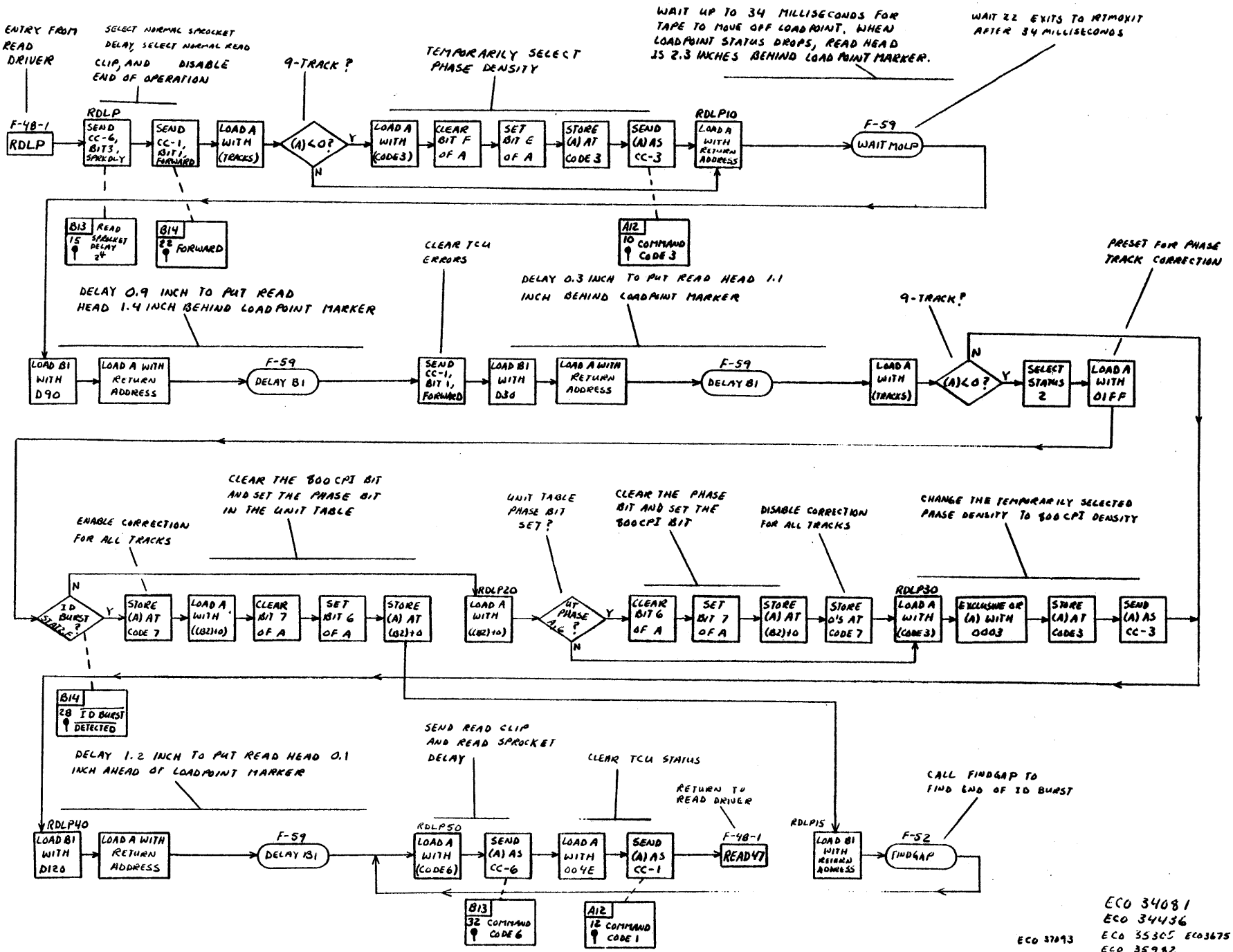


Figure F-49. Read From Loadpoint - RDLP

ECO 37093

ECO 34081
ECO 34436
ECO 35305 ECO36757
ECO 35982
ECO 36584

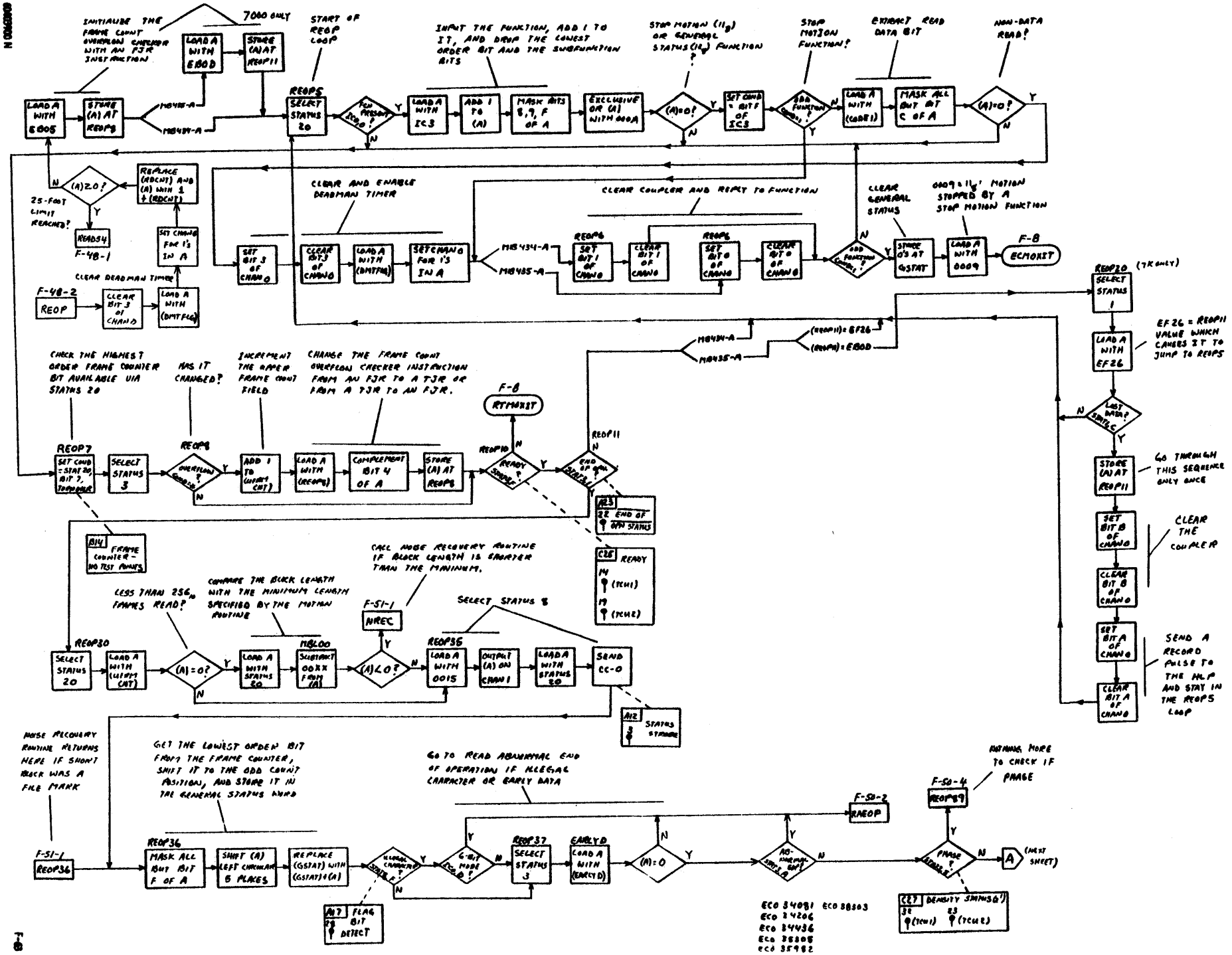


Figure F-50. Read End of Operation - REOP (Sheet 1 of 4)

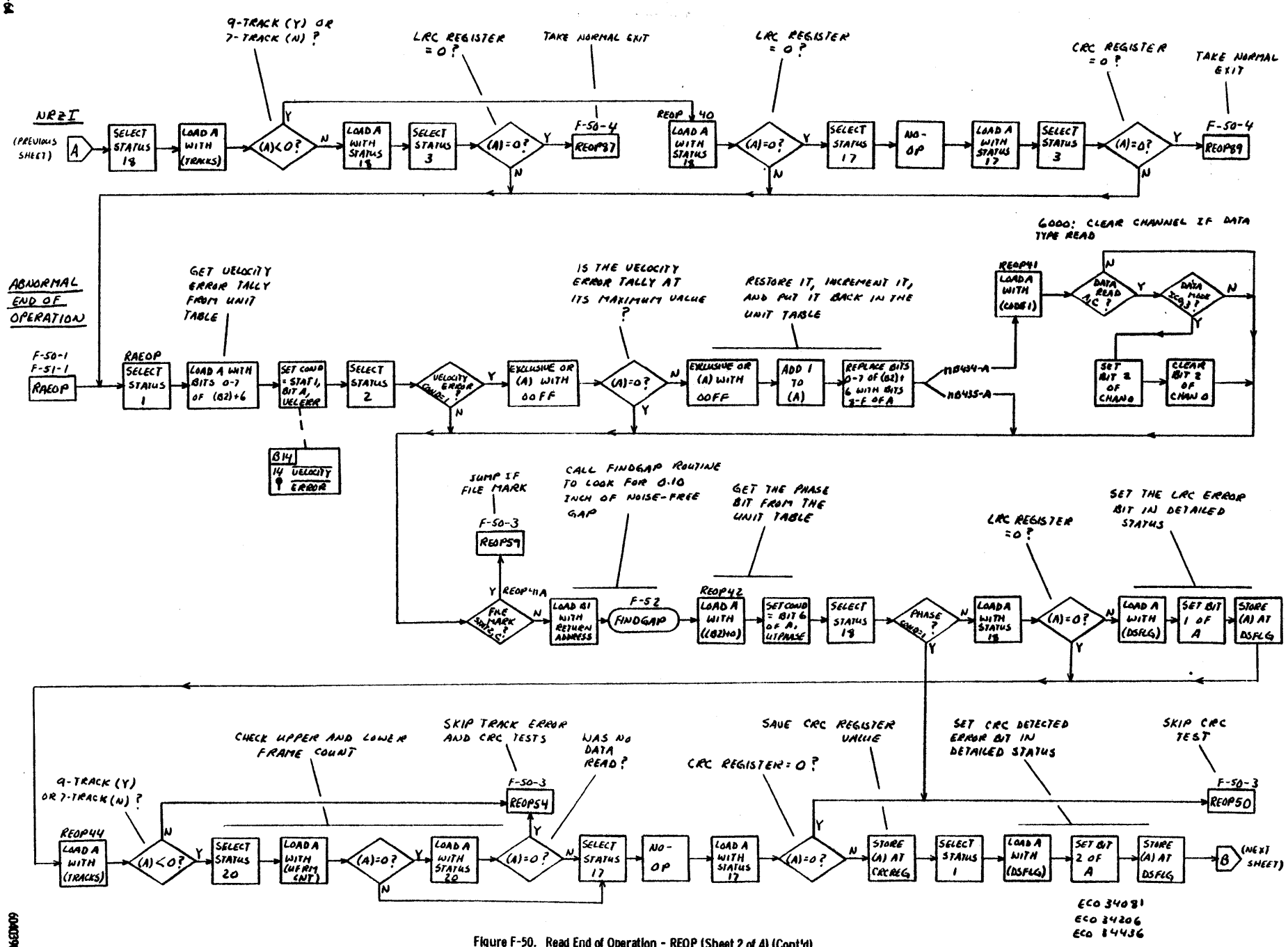


Figure F-50. Read End of Operation - REOP (Sheet 2 of 4) (Cont'd)

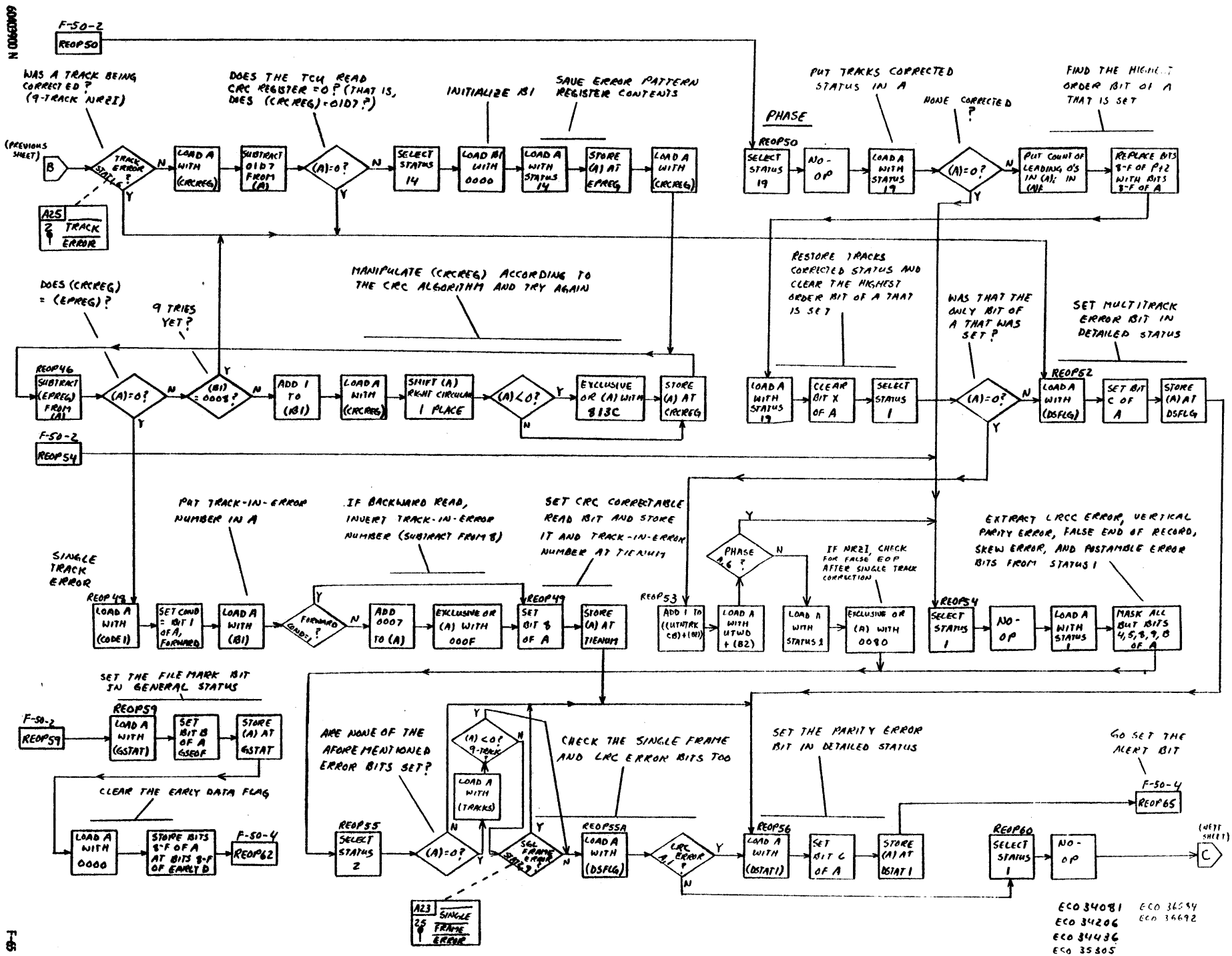


Figure F-50. Read End of Operation - REOP (Sheet 3 of 4) (Cont'd)

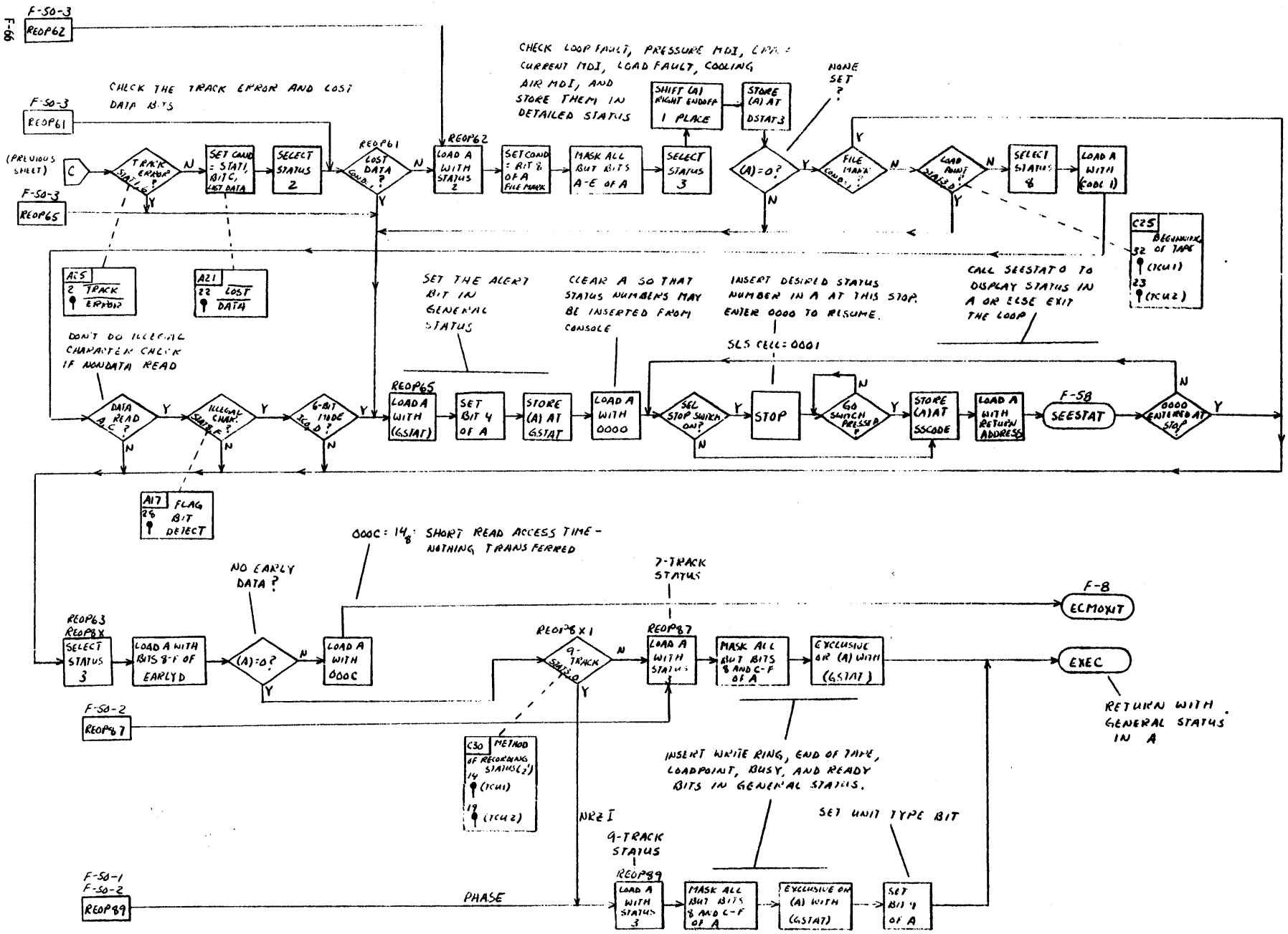
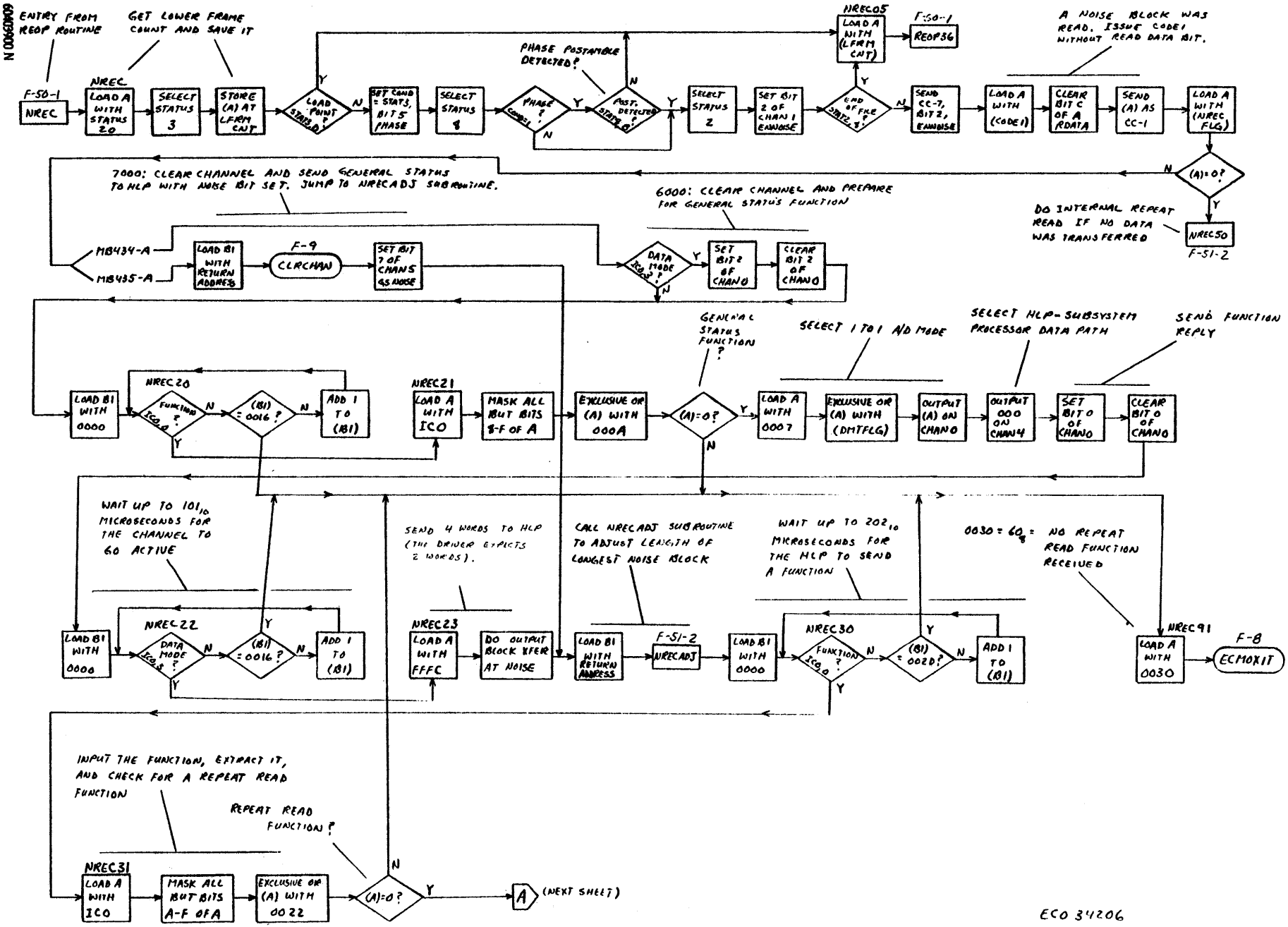


Figure F-50. Read End of Operation - REOP (Sheet 4 of 4) (Cont'd)

ECO 341081
ECO 34206
ECO 341136
ECO 35505



ECO 34206
ECO 34136
ECO 31112

Figure F-51. Noise Recovery - NREC (Sheet 1 of 2)

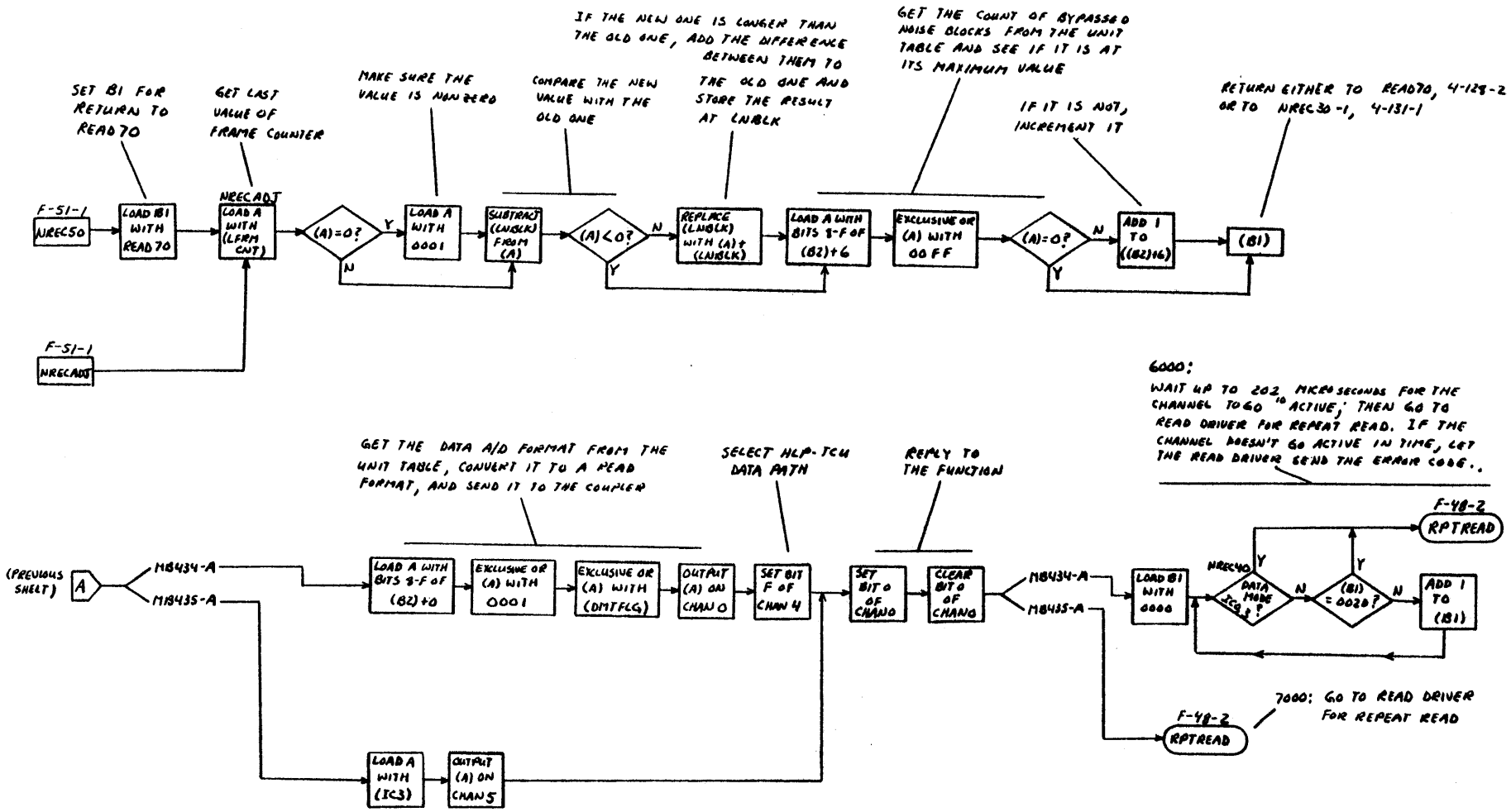


Figure F-51. Noise Recovery - NREC (Sheet 2 of 2) (Cont'd)

ECO 341206
ECO 344436

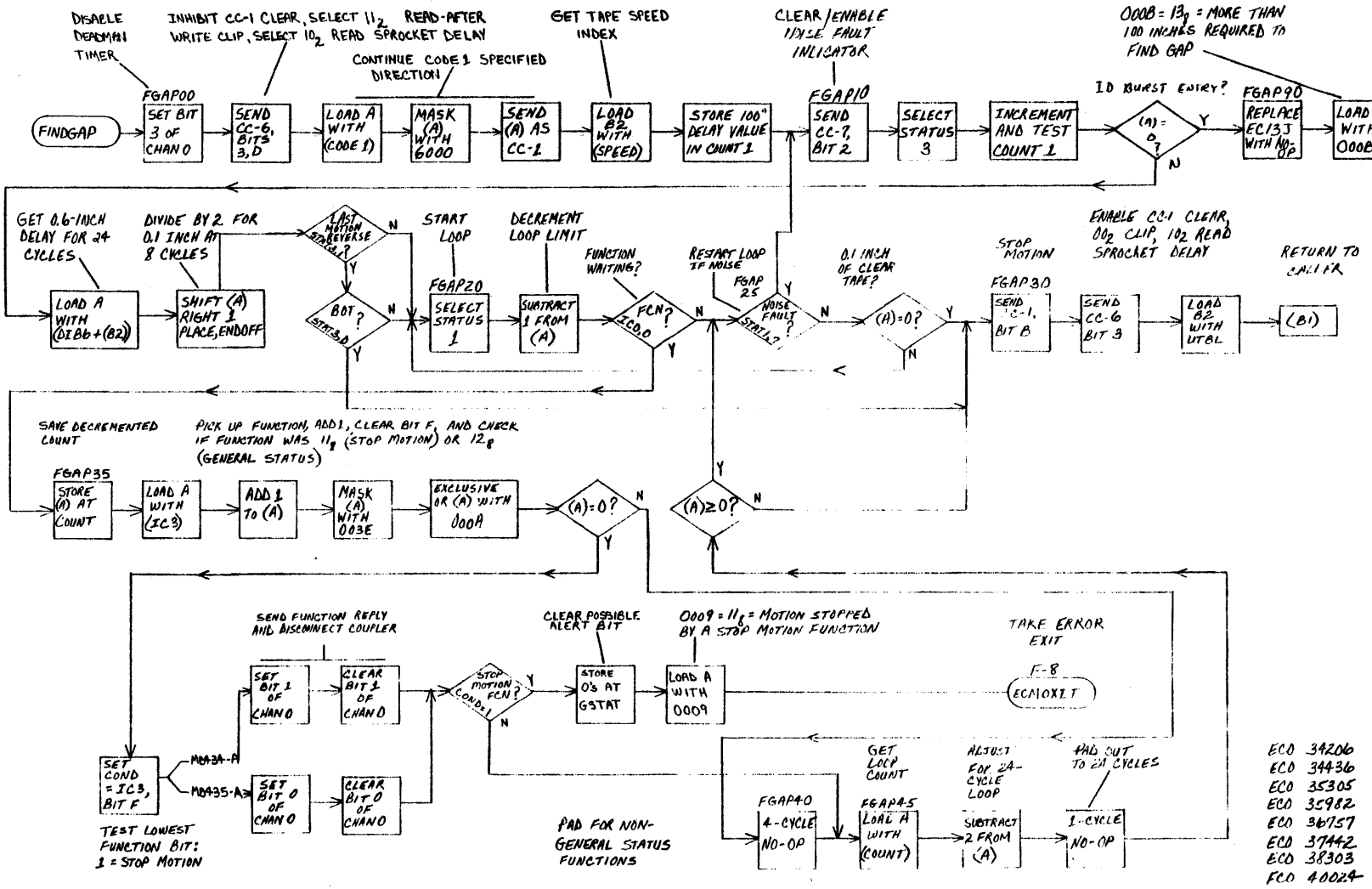


Figure F-52. Find Interblock Gap - FGAP00

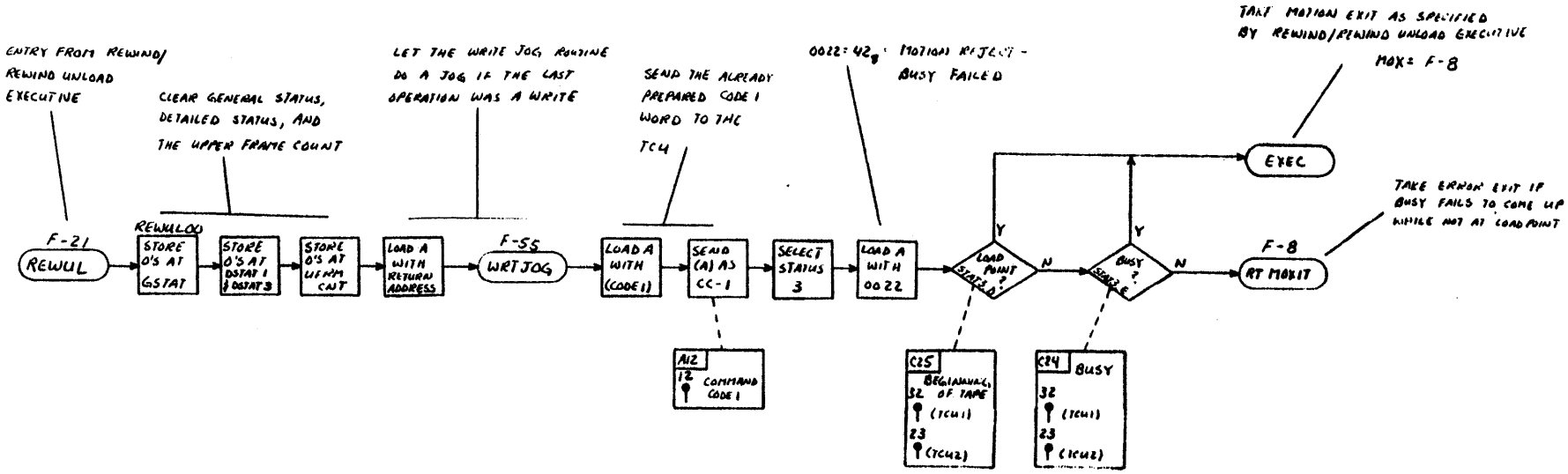


Figure F-53. Rewind/Rewind Unload Driver - REWUL00

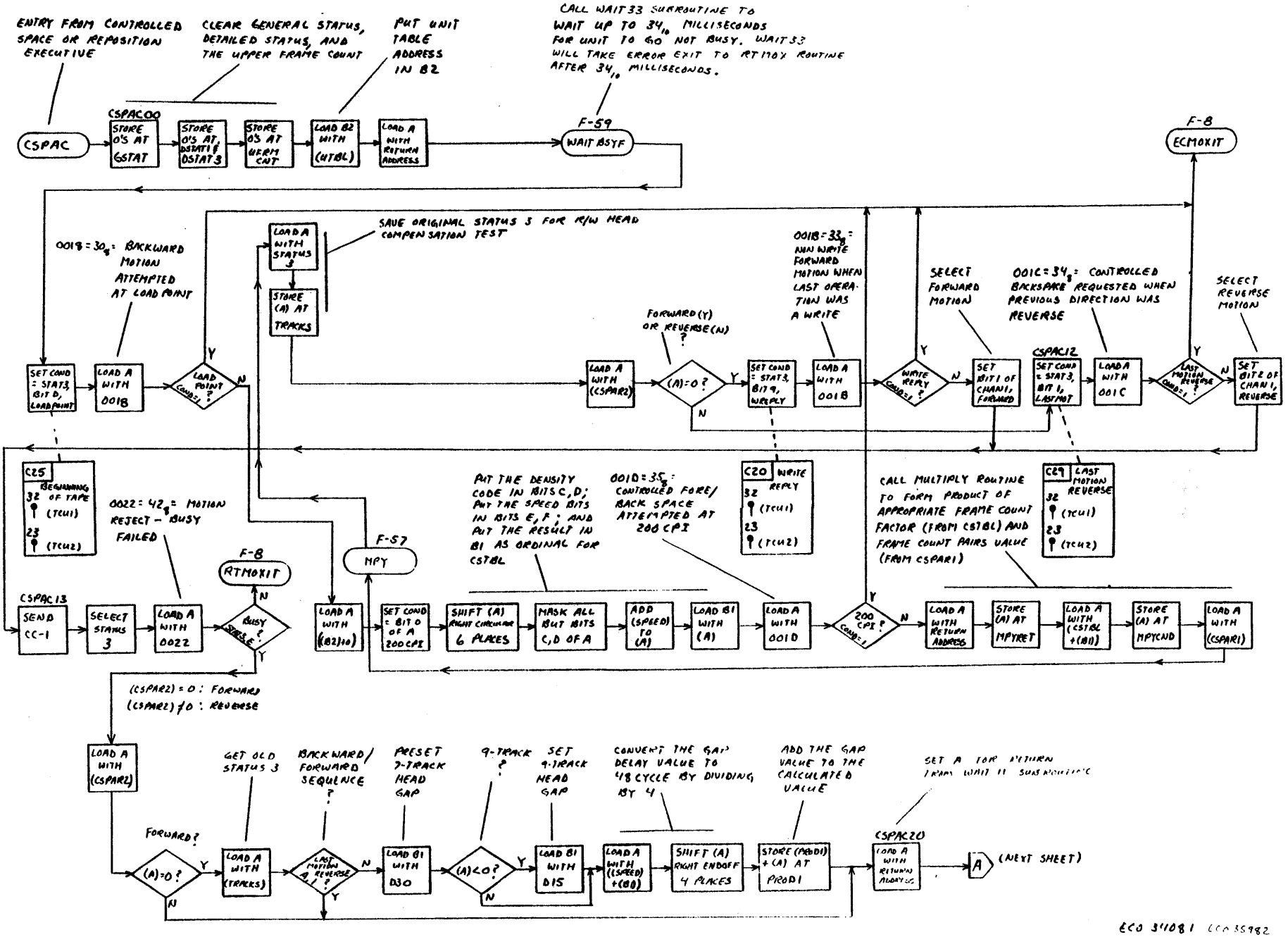
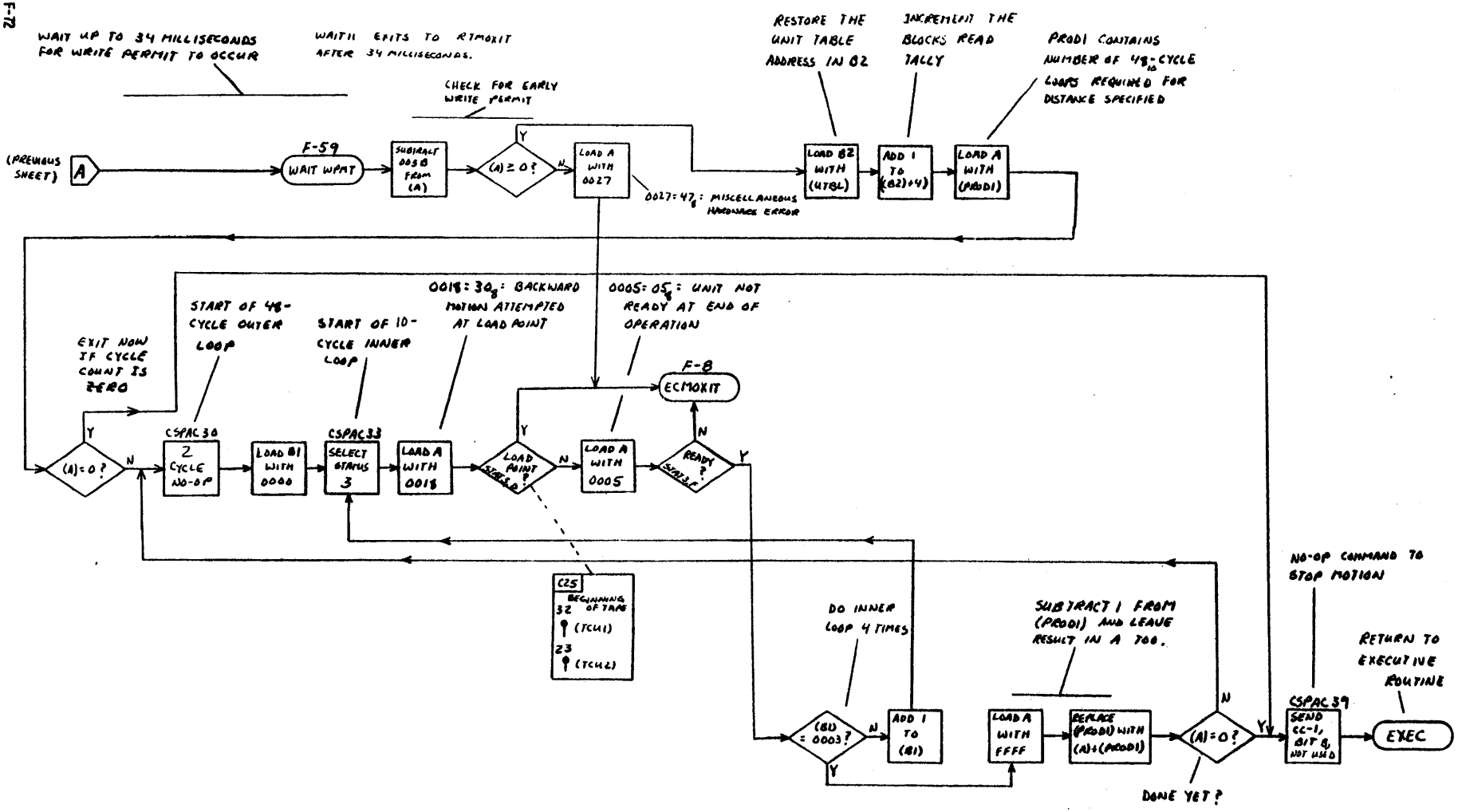


Figure F-54. Controlled Space Driver - CSPAC (Sheet 1 of 2)

ECO 341081 ECO 35982
ECO 34206 ECO 36757
ECO 34436 ECO 38305
ECO 35305



	0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F			
CSTBL+0	1	0	1	0	1	1	1	0	1	0	0	1	1	0	0	0	AE68 =	556 CPI,	100 IPS
+1	0	1	1	1	0	1	0	0	0	1	0	0	0	1	0	1	7445 =	556	150
+2	0	1	0	1	0	1	1	1	0	0	1	1	0	1	0	0	5734 =	556	200
+3	0	1	0	0	0	1	0	1	1	1	0	0	0	0	1	1	45C3 =	556	250
+4	0	1	1	1	1	0	0	1	0	0	1	1	0	1	1	0	7936 =	800	100
+5	0	1	0	1	0	0	0	0	1	1	0	0	1	1	1	1	50CF =	800	150
+6	0	0	1	1	1	1	0	0	1	0	0	1	1	0	1	1	3C9B =	800	200
+7	0	0	1	1	0	0	0	0	0	1	1	1	1	1	0	0	307C =	800	250
+8	0	0	1	1	1	1	0	0	1	0	0	1	1	0	1	1	3C9B =	1600	100
+9	0	0	1	0	1	0	0	0	0	1	1	0	0	1	1	1	2367 =	1600	150
+A	0	0	0	1	1	1	1	0	0	1	0	0	1	1	1	0	1E4E =	1600	200
CSTBL+B	0	0	0	1	1	0	0	0	0	0	1	1	1	1	1	0	183E =	1600 CPI,	250 IPS

TABLE OF FRAME COUNT FACTORS

WHEN THE 16_{10}-BIT FRAME COUNT PAIRS VALUE IS MULTIPLIED BY ONE OF THESE FACTORS, THE UPPER 16_{10} BITS OF THE 32_{10}-BIT PRODUCT GIVE THE NUMBER OF TIMES THE 48_{10}-CYCLE DELAY LOOP MUST BE REPEATED

ECO 34081 ECO 35982
ECO 34206 ECO 36757
ECO 34436 ECO 37[illegible]
ECO 35305

Figure F-54. Controlled Space Driver - CSPAC (Sheet 2 of 2) (Cont'd)

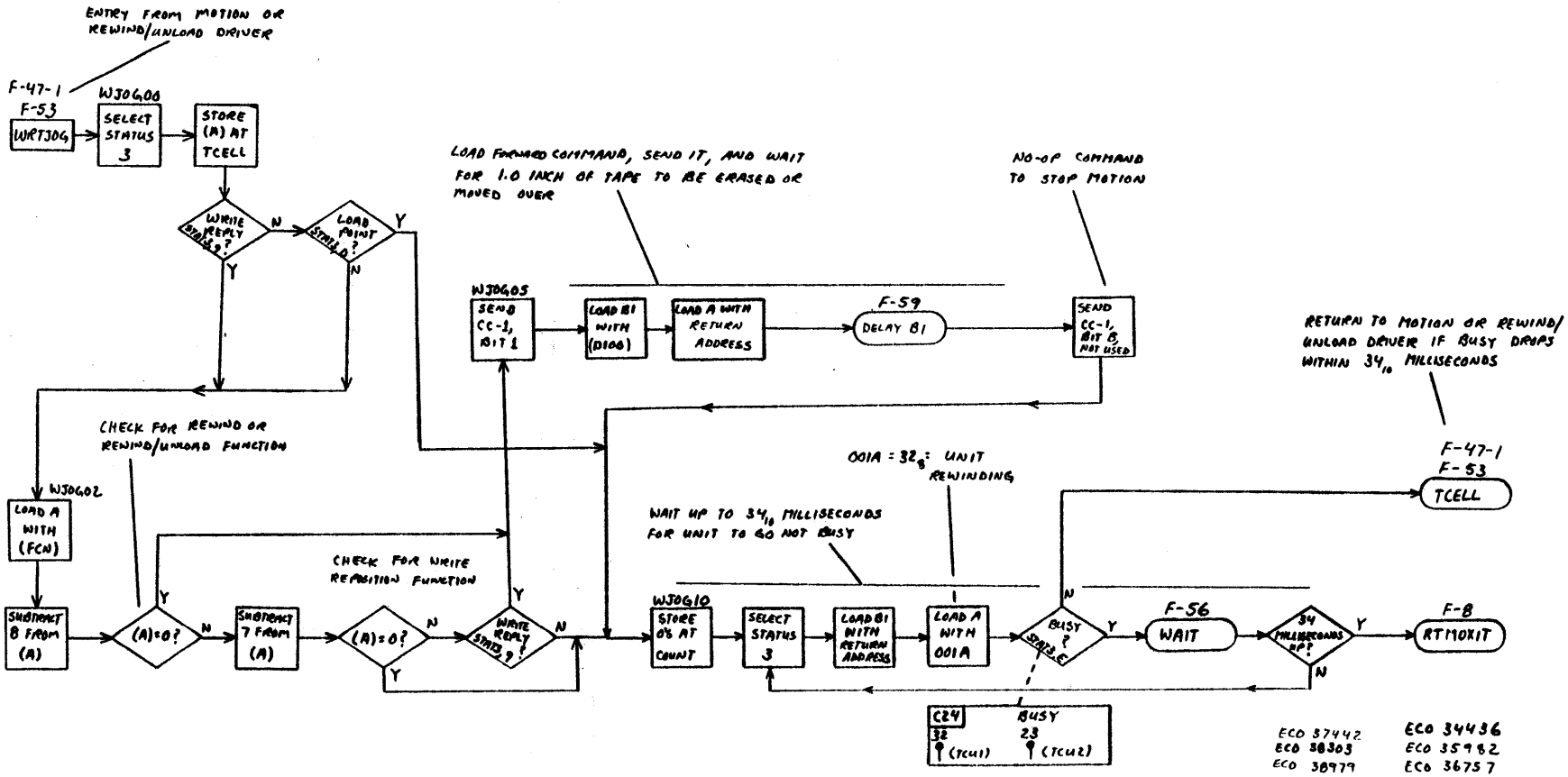
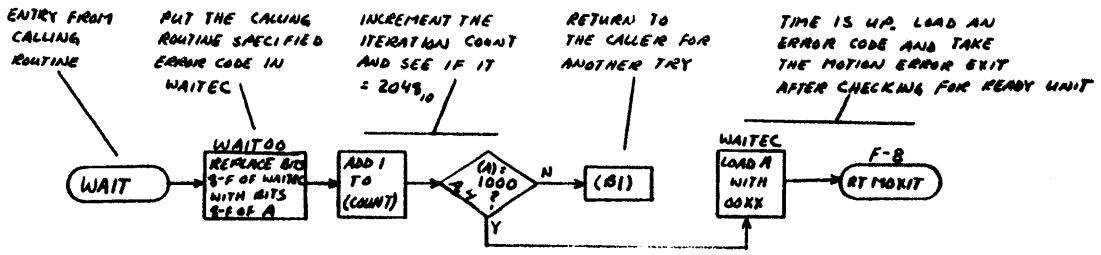


Figure F-55. Write Jog or Forward Jog - WJOG00



ECO 34436

Figure F-56. Wait - WAIT00

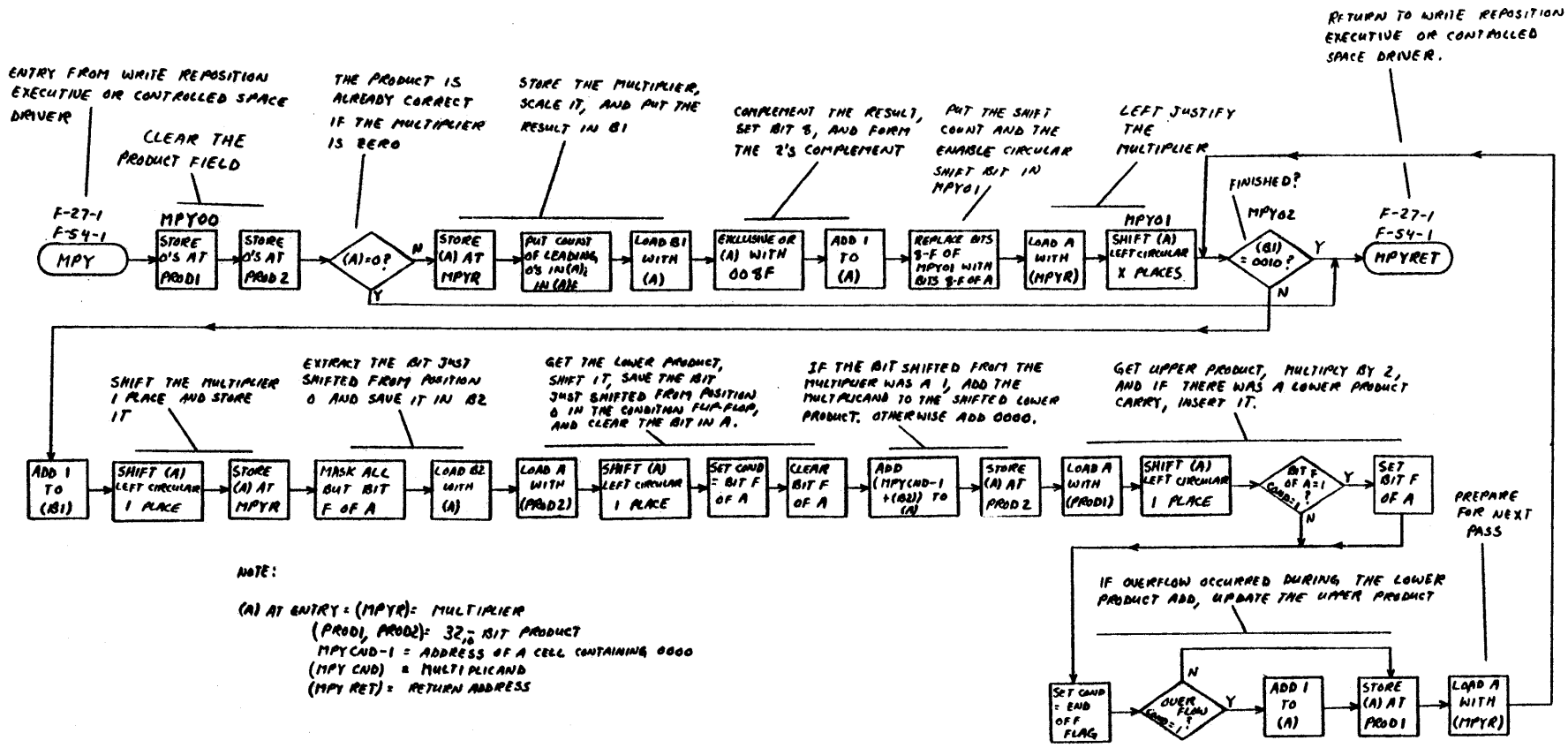
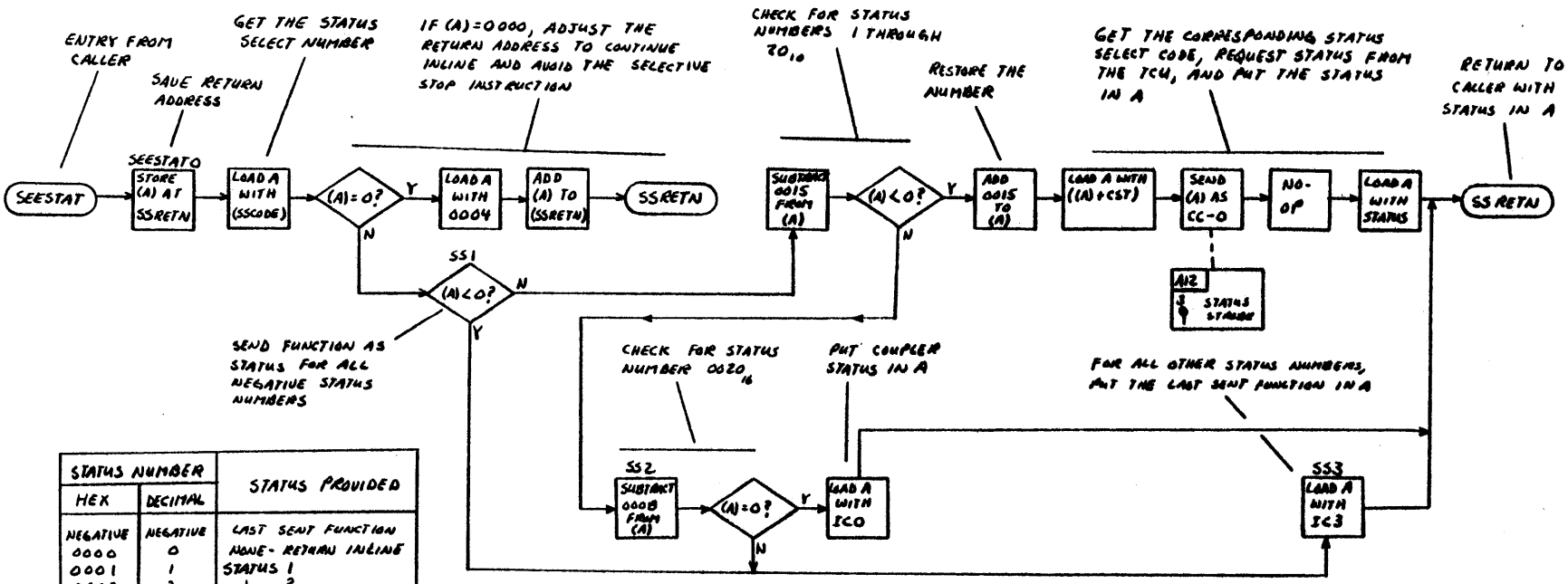


Figure F-57. Multiply - MPY00



STATUS NUMBER		STATUS PROVIDED
HEX	DECIMAL	
NEGATIVE	NEGATIVE	LAST SENT FUNCTION
0000	0	NONE - RETURN INLINE
0001	1	STATUS 1
0002	2	2
0003	3	3
0004	4	4
0005	5	5
0006	6	6
0007	7	7
0008	8	8
0009	9	9
000A	10	10
000B	11	11
000C	12	12
000D	13	13
000E	14	14
000F	15	15
0010	16	16
0011	17	17
0012	18	18
0013	19	19
0014	20	STATUS 20
0020	32	COMPLEX STATUS
ALL OTHERS	ALL OTHERS	LAST SENT FUNCTION

ECO 34081
ECO 34436

Figure F-58. Display Status in A - SEESTAT0

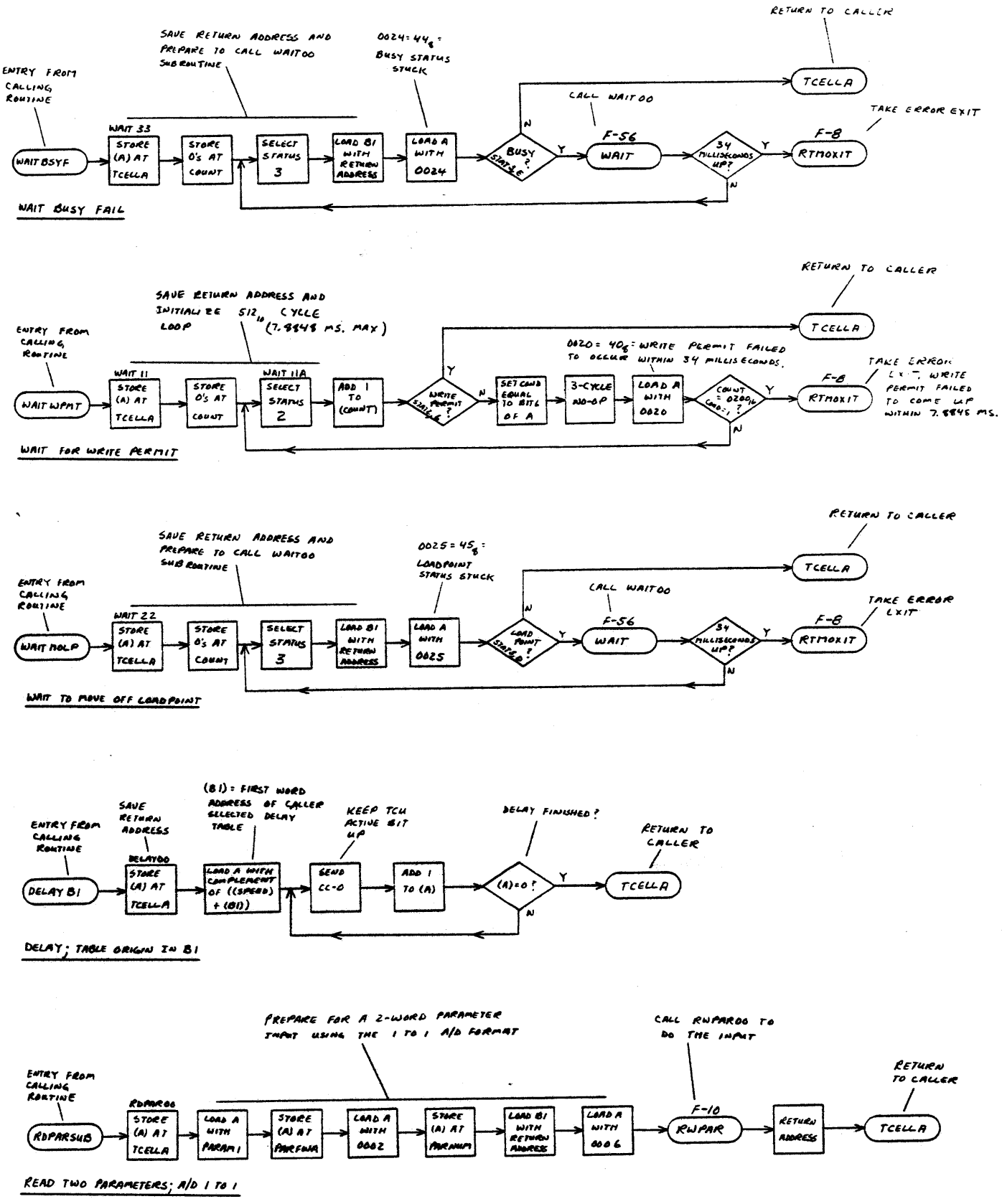


Figure F-59. Utility Subroutines - WAIT 33, WAIT 11, WAIT 22, DELAY00, RDPAR00

ECO 35982
ECO 36757

MEDIA REQUIREMENTS FOR MAGNETIC TAPE OPERATION G

Magnetic tapes used on Control Data subsystems must meet the following ANSI standards.

- X3.40-1976 — Unrecorded Tape for Information Interchange (9-track 200 and 800 CPI, NRZI, and 1600 CPI, PE)
- X3.54-1976 — Recorded Magnetic Tape for Information Interchange (6250 CPI, Group Coded Recording)
- X3.39-1973 — Recorded Magnetic Tape for Information Interchange (1600 CPI, NRZI)
- X3.22-1973 — Recorded Magnetic Tape for Information Interchange (800 CPI, NRZI)
- X3.14-1973 — Recorded Magnetic Tape for Information Interchange (200 CPI, NRZI)

Failure to meet these standards may result in unsatisfactory performance and interchange.

NOTE

1.0 mil tape is not allowed by ANSI X3.40-1976.

Most tapes properly certified for 1600-cpi operation will operate satisfactorily at 6250 cpi.

Control Data ALPHA PHI and OMEGA PHI will meet ANSI standards and are recommended for use on Control Data equipment.

Inhibitor Tapes

- Definition — Inhibitor tapes are tapes that degrade performance of the tape transport such that unsatisfactory operation is experienced with the tape in question, or with an otherwise satisfactory tape mounted on a transport in subsequent passes.
- Requirement — The tape shall not be an inhibitor tape.
- Examples — Inhibitor characteristics include poor edge conditions, excessive tape wear products, interlayer slippage, tendency for oxide coating to transfer to the back of the reel's next layer, and tendency for tape constituents to separate and cause deposits leading to tape stick or to inhibiting proper performance of other tapes. Tapes which exhibit these characteristics may not give satisfactory performance and may result in excessive errors.

Control Data employees may order ANSI standards from Control Data's System Standards Department located at HQW11H. All others may order them directly from:

American National Standards Institute, Inc.
1430 Broadway
New York, New York 10018

INDEX

COMMENT SHEET

MANUAL TITLE: CDC Magnetic Tape Subsystem
Operation and Programming Manual
PUBLICATION NO.: 60403900 **REVISION:** V

NAME: _____

COMPANY: _____

STREET ADDRESS: _____

CITY: _____ **STATE:** _____ **ZIP CODE:** _____

This form is not intended to be used as an order blank. Control Data Corporation welcomes your evaluation of this manual. Please indicate any errors, suggested additions or deletions, or general comments below (please include page number references).

☐ Please Reply ☐ No Reply Necessary

CUT ALONG LINE

AA3419 REV. 4/79 PRINTED IN U.S.A.

NO POSTAGE STAMP NECESSARY IF MAILED IN U.S.A.

FOLD FOLD

NO POSTAGE
NECESSARY
IF MAILED
IN THE
UNITED STATES

BUSINESS REPLY MAIL

FIRST CLASS PERMIT NO. 8241 MINNEAPOLIS, MINN.

POSTAGE WILL BE PAID BY

CONTROL DATA CORPORATION

Publications and Graphics Division
ARH219
4201 North Lexington Avenue
Saint Paul, Minnesota 55112

CUT ALONG LINE

FOLD FOLD

CORPORATE HEADQUARTERS, P.O. BOX 0, MINNEAPOLIS, MINN. 55440
SALES OFFICES AND SERVICE CENTERS IN MAJOR CITIES THROUGHOUT THE WORLD

LITHO IN U.S.A.

CONTROL DATA CORPORATION