EK-HSC70-SV-002

HSC70
Service Manual

digital

EK-HSC70-SV-002

HSC70

SERVICE MANUAL

Prepared by Educational Services

Digital Equipment Corporation

First Edition, March 1986

Second Edition, September 1986

Copyright (c) Digital Equipment Corporation 1985

All Rights Reserved
Printed in USA

The material in this manual is for informational purposes and is subject to change without notice.

Digital Equipment Corporation assumes no responsibility for any errors which may appear in this manual.

The HSC70 Mass Storage Server is designed to work with Digitql Equipment Corporation host computers, tape products, and disk products. Digital Equipment Corporation assumes no responsibility or liability if the computers, tape products, or disk products of another manufacturer are used with the HSC70 subsystem.

- Class A Computing Devices:

NOTICE: This equipment generates, uses, and may emit radio frequency energy. This equipment has been type tested and found to comply with the limits for a class A computing device pursuant to Subpart J of Part 25 of FCC rules designed to provide reasonable protection against such radio frequency interference when operated in a commercial environment. Operation of this equipment in a residential area may cause interference. In such a case, the owner, at his own expense, may be required to take measures to correct the interference.

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

DEC	DECnet	OMNIBUS
DECUS	DECsystem-10	OS/8
DIGITAL	DECSYSTEM-20	PDT
logo	DECwriter	RSTS
PDP	DIBOL	RSX
UNIBUS	EduSystem	VMS
RA81	RA60	UDA50
VAX	VT	KDA50-Q
RA91	RA60	IAS
HSC	RA80	TOPS-20
RA81		

CONTENTS

CHAPTER 1 GENERAL INFORMATION

1.1 INTRODUCTION 1-1
1.2 GENERAL INFORMATION 1-1
1.2.1 HSC70 Cabinet Layout 1-2
1.2.2 External Interfaces 1-7
1.2.3 Internal Software 1-8
1.2.4 Subsystem Block Diagram 1-10
1.3 MODULE DESCRIPTIONS 1-11
1.3.1 Port Link Module (LINK) Functions 1-12
1.3.2 Port Buffer Module (PILA) Functions 1-14
1.3.3 Port Processor Module (K.pli) Functions And Interfaces 1-14
1.3.4 Disk Data Channel Module (K.sdi) Functions . . 1-14
1.3.5 Tape Data Channel Module (K.sdi) Functions . . 1-15
1.3.6 Input/Output (I/O) Control Processor Module (P.ioj) Functions 1-15
1.3.7 Memory Module (M.std2) Functions 1-16
1.4 HSC70 MAINTENANCE STRATEGY 1-17
1.4.1 Maintenance Features 1-18
1.4.2 HSC70 Specifications 1-19
1.5 HSC70 RELATED DOCUMENTATION 1-20

CHAPTER 2 HSC70 CONTROLS/INDICATORS

2.1 INTRODUCTION 2-1
2.2 OPERATOR CONTROL PANEL (OCP) 2-1
2.3 INSIDE FRONT DOOR CONTROLS/INDICATORS 2-3
2.4 MODULE INDICATORS AND SWITCHES 2-6
2.4.1 Module Switches 2-9
2.5 POWER CONTROLLER 2-10
2.5.1 Operating Instructions 2-10

CHAPTER 3 REMOVAL AND REPLACEMENT PROCEDURES

3.1 INTRODUCTION 3-1
3.2 SAFETY PRECAUTIONS 3-1
3.3 POWER REMOVAL 3-1
3.4 FIELD REPLACEABLE UNIT (FRU) REMOVAL 3-4
3.4.1 Access From Cabinet Front Door 3-4
3.4.2 Access From Cabinet Back Door 3-5
3.5 RX33 COVER PLATE AND DISK DRIVE REMOVAL AND REPLACEMENT 3-5
3.5.1 RX33 Jumper Configuration 3-9
3.6 OPERATOR CONTROL PANEL (OCP) REMOVAL AND REPLACEMENT 3-9
3.7 LOGIC MODULES REMOVAL AND REPLACEMENT 3-11
3.8 BLOWER REMOVAL AND REPLACEMENT 3-13

3.9 AIRFLOW SENSOR ASSEMBLY REMOVAL AND REPLACEMENT 3-15
3.10 POWER CONTROLLER REMOVAL AND REPLACEMENT . . . 3-16
3.11 MAIN POWER SUPPLY REMOVAL AND REPLACEMENT . . . 3-18
3.12 AUXILIARY POWER SUPPLY 3-21

CHAPTER 4 INITIALIZATION PROCEDURES

4.1 INTRODUCTION 4-1
4.2 CONSOLE TERMINAL CONNECTION 4-1
4.3 HSC70 INITIALIZATION 4-2
4.3.1 Init P.io Test 4-3
4.3.1.1 Init P.io Test System Requirements 4-3
4.3.1.2 Init P.io Test Prerequisites 4-3
4.3.1.3 Init P.io Test Operation 4-3
4.3.2 Fault Code Interpretation 4-4
4.3.3 Init P.io Test Summaries 4-13

CHAPTER 5 INLINE DIAGNOSTICS

5.1 INTRODUCTION 5-1
5.1.1 Inline Diagnostics Commonalities 5-1
5.1.1.1 Inline Diagnostics Generic Error Message Format 5-2
5.2 INLINE RX33 DIAGNOSTIC TEST (ILRX33) 5-2
5.2.1 ILRX33 System Requirements 5-2
5.2.2 ILRX33 Operating Instructions 5-3
5.2.3 ILRX33 Test Parameter Entry 5-3
5.2.4 ILRX33 Setting/Clearing 5-4
5.2.5 ILRX33 Progress Reports 5-4
5.2.6 ILRX33 Test Termination 5-4
5.2.7 ILRX33 Error Message Example 5-4
5.2.8 ILRX33 Error Messages 5-4
5.2.9 ILRX33 Test Summary 5-6
5.3 INLINE MEMORY TEST (ILMEMY) 5-6
5.3.1 ILMEMY System Requirements 5-7
5.3.2 ILMEMY Operating Instructions 5-7
5.3.3 ILMEMY Progress Reports 5-7
5.3.4 ILMEMY Error Message Example 5-8
5.3.5 ILMEMY Error Messages 5-8
5.3.6 ILMEMY Test Summaries 5-8
5.4 INLINE DISK DRIVE DIAGNOSTIC TEST (ILDISK) . . . 5-9
5.4.1 ILDISK System Requirements 5-10
5.4.2 ILDISK Operating Instructions 5-10
5.4.3 ILDISK Availability 5-11
5.4.4 ILDISK Test Parameter Entry 5-11
5.4.5 Specifying Requestor And Port - ILDISK 5-12
5.4.6 ILDISK Progress Reports 5-13
5.4.7 ILDISK Test Termination 5-13
5.4.8 ILDISK Error Message Example 5-13
5.4.9 ILDISK Error Messages 5-13
5.4.9.1 MSCP Status Codes - ILDISK Error Reports . . 5-26
5.4.10 ILDISK Test Summaries 5-27

5.5 INLINE TAPE TEST (ILTAPE) 5-31
5.5.1 ILTAPE System Requirements 5-31
5.5.2 ILTAPE Operating Instructions 5-32
5.5.3 ILTAPE/User Dialogue 5-32
5.5.4 ILTAPE User Sequences 5-36
5.5.5 ILTAPE Progress Reports 5-38
5.5.6 ILTAPE Test Termination 5-38
5.5.7 ILTAPE Error Message Example 5-39
5.5.8 ILTAPE Error Messages 5-39
5.5.9 ILTAPE Test Summaries 5-43
5.5.9.1 K.sti Interface Test Summary 5-43
5.5.9.2 Formatter Diagnostics Test Summary 5-43
5.5.9.3 User Sequences Test Summary 5-44
5.5.9.4 Canned Sequence Test Summary 5-44
5.5.9.5 Streaming Sequence Test Summary 5-44
5.6 INLINE TAPE COMPATABILITY TEST (ILTCOM) 5-44
5.6.1 ILTCOM System Requirements 5-46
5.6.2 ILTCOM Operating Instructions 5-47
5.6.3 ILTCOM Test Parameter Entry 5-47
5.6.4 ILTCOM Test Termination 5-49
5.6.5 ILTCOM Error Message Example 5-49
5.6.5.1 ILTCOM Error Messages 5-50
5.6.6 ILTCOM Test Summaries 5-51
5.7 INLINE MULTIDRIVE EXERCISER (ILEXER) 5-51
5.7.1 ILEXER System Requirements 5-51
5.7.2 ILEXER Operating Instructions 5-52
5.7.3 ILEXER Test Parameter Entry 5-53
5.7.4 Disk Drive User Prompts 5-54
5.7.5 Tape Drive User Prompts 5-57
5.7.6 ILEXER Global User Prompts 5-58
5.7.7 ILEXER Data Patterns 5-60
5.7.8 Setting/Clearing Flags - ILEXER 5-62
5.7.9 ILEXER Progress Reports 5-62
5.7.10 ILEXER Data Transfer Error Report 5-62
5.7.11 ILEXER Performance Summary 5-63
5.7.12 ILEXER Communications Error Report 5-66
5.7.13 ILEXER Test Termination 5-66
5.7.14 ILEXER Error Message Format 5-66
5.7.14.1 ILEXER Prompt Error Format 5-66
5.7.14.2 ILEXER Data Transfer Compare Error Format . 5-67
5.7.14.3 ILEXER Communications Error Format 5-68
5.7.15 ILEXER Error Messages 5-68
5.7.15.1 ILEXER Informational Messages 5-68
5.7.15.2 ILEXER Generic Errors 5-69
5.7.15.3 ILEXER Disk Errors 5-71
5.7.15.4 ILEXER Tape Errors 5-73
5.7.16 ILEXER Test Summaries 5-75

CHAPTER 6 OFFLINE DIAGNOSTICS

6.1 INTRODUCTION 6-1
6.1.1 Offline Diagnostics Software Requirements . . 6-2
6.1.2 Offline Diagnostics Load Procedure 6-2

6.1.3 P.ioj ROM Bootstrap 6-2
6.1.3.1 Bootstrap Initialization Instructions . . . 6-3
6.1.3.2 Bootstrap Failures 6-3
6.1.3.3 Bootstrap Progress Reports 6-4
6.1.3.4 Bootstrap Error Information 6-5
6.1.3.5 Bootstrap Failure Troubleshooting 6-5
6.1.4 Bootstrap Test Summaries 6-6
6.1.5 Offline Diagnostics Error Reporting And Message Format 6-10
6.2 OFFLINE DIAGNOSTICS LOADER 6-11
6.2.1 Offline Diagnostic Loader System Requirements 6-11
6.2.2 Offline Diagnostic Loader Prerequisites . . . 6-12
6.2.3 Operating Instructions For The Offline Diagnostic Loader 6-12
6.2.4 Offline Diagnostic Loader Commands 6-12
6.2.4.1 Offline Diagnostic Loader HELP Command . . . 6-12
6.2.4.2 Offline Diagnostic Loader SIZE Command . . . 6-13
6.2.4.3 Offline Diagnostic Loader TEST Command . . 6-13
6.2.4.4 Offline Diagnostic Loader LOAD Command . . . 6-14
6.2.4.5 Offline Diagnostic Loader START Command . . 6-14
6.2.4.6 EXAMINE And DEPOSIT Commands 6-14
6.2.4.6.1 Offline Diagnostic Loader EXAMINE Command 6-14
6.2.4.6.2 Offline Diagnostic Loader DEPOSIT Command 6-15
6.2.4.6.3 Offline Diagnostic Symbolic Addresses . . 6-15
6.2.4.6.4 Repeating EXAMINE And DEPOSIT Commands . 6-16
6.2.4.6.5 Offline Diagnostics Relocation Register . 6-17
6.2.4.6.6 Offline Diagnostics EXAMINE And DEPOSIT Qualifiers (Switches) 6-18
6.2.4.6.7 Setting And Showing Defaults 6-19
6.2.4.6.8 Executing INDIRECT Command Files 6-20
6.2.5 Offline Diagnostics Unexpected Traps And Interrupts 6-20
6.2.5.1 Offline Diagnostics Trap And Interrupt Vectors 6-21
6.2.5.2 Offline Diagnostics Loader Help File 6-21
6.3 OFFLINE CACHE TEST 6-22
6.3.1 Offline Cache Test System Requirements 6-22
6.3.2 Offline Cache Test Operating Instructions . . 6-23
6.3.3 Offline Cache Test Parameter Entry 6-23
6.3.4 Offline Cache Test Progress Reports 6-24
6.3.5 Offline Cache Test Error Information 6-24
6.3.5.1 Specific Offline Cache Error Messages . . . 6-24
6.3.6 Offline Cache Test Troubleshooting 6-29
6.3.7 Offline Cache Test Descriptions 6-29
6.4 OFFLINE BUS INTERACTION TEST 6-33
6.4.1 Offline Bus Interaction Test System Requirements 6-34
6.4.2 Offline Bus Interaction Test Prerequisites . 6-34
6.4.3 Offline Bus Interaction Test Operating Instructions 6-35
6.4.4 Offline Bus Interaction Test Parameter Entry . 6-35
6.4.5 Offline Bus Interaction Test Progress Reports 6-37
6.4.6 Offline Bus Interaction Test Error Information 6-37
6.4.6.1 Requestor Error Summary 6-38

6.4.6.2 Offline Bus Interaction Memory Test Configuration 6-38
6.4.6.3 Offline Bus Interaction Test Error Messages 6-39
6.4.6.4 Offline Bus Interaction K Memory Test Algorithm 6-42
6.5 OFFLINE K TEST SELECTOR 6-43
6.5.1 Offline K Test Selector System Requirements . 6-43
6.5.2 Offline K Test Selector Operating Instructions 6-43
6.5.3 Offline K Test Selector Parameter Entry . . . 6-44
6.5.4 Offline K Test Selector Progress Reports . . . 6-45
6.5.5 Offline K Test Selector Error Information . . 6-45
6.5.5.1 K.ci Path Status Information 6-46
6.5.5.2 Offline K Test Selector Error Messages . . . 6-46
6.5.6 Offline K Test Selector Summaries 6-55
6.6 OFFLINE K/P MEMORY TEST 6-57
6.6.1 Offline K/P Memory Test System Requirements . 6-57
6.6.2 Offline K/P Memory Test Operating Instructions 6-58
6.6.3 Offline K/P Memory Test Parameter Entry . . . 6-58
6.6.4 Offline K/P Memory Test Progress Reports . . 6-60
6.6.5 Offline K/P Memory Test Parity Errors 6-60
6.6.6 Offline K/P Memory Test Error Information . . 6-61
6.6.6.1 Offline K/P Memory Test Error Summary Information 6-62
6.6.6.2 Offline K/P Memory Test Error Messages . . 6-62
6.6.7 Offline K/P Memory Test Summaries 6-72
6.7 OFFLINE MEMORY TEST 6-73
6.7.1 Offline Memory Test System Requirements . . . 6-73
6.7.2 Offline Memory Test Operating Instructions . 6-74
6.7.3 Offline Memory Test Parameter Entry 6-74
6.7.4 Offline Memory Test Progress Reports 6-75
6.7.5 Offline Memory Test Parity Errors 6-76
6.7.6 Offline Memory Test Error Information 6-76
6.7.6.1 Offline Memory Test Error Messages 6-76
6.7.7 Offline Memory Test Summaries 6-88
6.8 RX33 OFFLINE EXERCISER 6-89
6.8.1 RX33 Offline Exerciser System Requirements . . 6-90
6.8.2 RX33 Offline Exerciser Operating Instructions 6-90
6.8.3 RX33 Offline Exerciser Parameter Entry 6-90
6.8.4 RX33 Offline Exerciser Progress Reports . . . 6-91
6.8.5 RX33 Offline Exerciser Error Information . . 6-92
6.8.5.1 Specific RX33 Offline Exerciser Error Messages 6-92
6.8.6 RX33 Offline Exerciser Test Summaries 6-97
6.8.7 RX33 Offline Exerciser Data Patterns 6-99
6.9 OFFLINE REFRESH TEST 6-100
6.9.1 Offline Refresh Test System Requirements . . 6-100
6.9.2 Offline Refresh Test Operating Instructions 6-100
6.9.3 Offline Refresh Test Parameter Entry 6-101
6.9.4 Offline Refresh Test Progress Reports . . . 6-101
6.9.5 Offline Refresh Test Error Information . . . 6-101
6.9.5.1 Offline Refresh Test Error Messages . . . 6-102
6.9.6 Offline Memory Refresh Test Summaries . . . 6-103
6.10 OFFLINE OPERATOR CONTROL PANEL TEST 6-104

6.10.1 Offline Operator Control Panel Test System Requirements 6-104
6.10.2 Operator Control Panel Test Operating Instructions 6-104
6.10.3 Offline Operator Control Panel Test Parameter Entry 6-105
6.10.4 Offline Operator Control Panel Test Error Information 6-107
6.10.4.1 Offline Operator Control Panel Test Error Messages 6-107
6.10.5 Offline Operator Control Panel Test Summaries 6-108
6.10.6 Offline OCP Registers And Displays Via ODT 6-112
6.10.6.1 Offline OCP Test Switch Check Via ODT . . 6-112
6.10.6.2 Offline OCP Test Lamp Bit Check Via ODT . 6-113
6.10.6.3 Offline OCP Test Secure/Enable Switch Check Via ODT 6-114
6.10.6.4 Offline OCP Test State LED Check Via ODT 6-116

CHAPTER 7 UTILITIES

7.1 INTRODUCTION 7-1
7.2 OFFLINE DISK UTILITY (DKUTIL) 7-1
7.2.1 DKUTIL Initialization 7-1
7.2.2 DKUTIL Command Syntax 7-2
7.2.3 DKUTIL Command Modifiers 7-3
7.2.4 DKUTIL Sample Session 7-3
7.2.5 DKUTIL Command Descriptions 7-7
7.2.5.1 DKUTIL DEFAULT Command 7-8
7.2.5.2 DKUTIL DISPLAY Command 7-10
7.2.5.3 DKUTIL DUMP Command 7-12
7.2.5.4 DKUTIL EXIT Command 7-14
7.2.5.5 DKUTIL GET Command 7-14
7.2.5.6 DKUTIL POP Command 7-15
7.2.5.7 DKUTIL PUSH Command 7-15
7.2.5.8 DKUTIL REVECTOR Command 7-16
7.2.5.9 DKUTIL SET Command 7-16
7.2.6 DKUTIL Error Messages 7-17
7.2.6.1 DKUTIL Error Message Variables 7-17
7.2.6.2 DKUTIL Error Message Severity Levels 7-17
7.3 OFFLINE DISK VERIFIER UTILITY (VERIFY) 7-20
7.3.1 VERIFY Initiation 7-22
7.3.2 VERIFY Sample Session 7-23
7.3.3 VERIFY Errors And Information Messages 7-24
7.3.3.1 VERIFY Variable Output Fields 7-25
7.3.3.2 VERIFY Error Message Severity Levels 7-25
7.3.3.3 VERIFY Fatal Error Messages 7-25
7.3.3.4 VERIFY Information Messages 7-26
7.3.3.5 VERIFY Warning Messages 7-26
7.3.3.6 VERIFY Type Error Messages 7-28
7.3.3.7 VERIFY Informational Messages 7-29
7.4 OFFLINE DISK FORMATTER UTILITY (FORMAT) 7-30
7.4.1 FORMAT Initiation 7-31

7.4.2 FORMAT Sample Session 7-33
7.4.3 FORMAT Errors And Information Messages 7-34
7.4.3.1 FORMAT Error Message Variables 7-34
7.4.3.2 FORMAT Message Severity Levels 7-35
7.4.3.3 FORMAT Fatal Error Messages 7-35
7.4.3.4 FORMAT Warning Message 7-37
7.4.3.5 FORMAT Information Messages 7-37
7.4.3.6 FORMAT Error Messages 7-38
7.4.3.7 FORMAT Success Messages 7-38
7.5 RXFORMAT UTILITY 7-38
7.5.1 RXFORMAT Initiation 7-38
7.5.2 RXFORMAT Error Messages 7-40
7.6 VIDEO TERMINAL DISPLAY (VTDPY) 7-41
7.6.1 VTDPY Error Messages 7-42
7.6.2 VTDPY Display Example 7-42
7.6.2.1 VTDPY Display Explanation 7-43

CHAPTER 8 TROUBLESHOOTING TECHNIQUES

8.1 INTRODUCTION 8-1
8.2 HOW TO USE THIS CHAPTER 8-1
8.3 INITIALIZATION ERROR INDICATIONS 8-2
8.3.1 OCP Fault Code Displays 8-2
8.3.2 Module LEDs 8-4
8.3.2.1 P.ioj LEDs 8-4
8.3.2.2 Power-Up Sequence Of I/O Control Processor LEDs 8-4
8.3.2.3 Memory Module LEDs 8-5
8.3.2.4 Data Channel LEDs 8-6
8.3.2.5 Host Interface LEDs 8-6
8.3.3 Communication Errors 8-7
8.3.4 Requestor Status For Nonfailing Requestors . . 8-8
8.3.5 Boot Flowchart 8-8
8.3.6 Boot Diagnostic Indications 8-13
8.4 SOFTWARE ERROR MESSAGES 8-13
8.4.1 Mass Storage Control Protocol Errors 8-13
8.4.2 MSCP/TMSCP Error Format, Description, And Flags 8-13
8.4.2.1 MSCP/TMSCP Error Format 8-14
8.4.2.2 MSCP/TMSCP Error Message Fields 8-14
8.4.2.3 MSCP/TMSCP Error Flags 8-15
8.4.2.4 MSCP/TMSCP Controller Errors 8-16
8.4.2.4.1 Controller Error List 8-16
8.4.2.5 MSCP SDI Errors 8-20
8.4.2.6 Disk Transfer Errors 8-35
8.4.3 Bad Block Replacement Errors (BBR) 8-42
8.4.4 TMSCP-Specific Errors 8-46
8.4.4.1 STI Communication Or Command Errors 8-46
8.4.4.2 STI Formatter Error Log 8-48
8.4.4.3 STI Drive Error Log 8-49
8.4.4.4 Breakdown Of GEDS Text Field 8-53
8.4.4.5 Breakdown Of GSS Text Field 8-55
8.4.5 Out-of-Band Errors 8-70

8.4.5.1 CI Errors 8-71
8.4.5.2 Load Device Errors 8-81
8.4.5.3 Disk Functional Errors 8-84
8.4.5.4 Tape Functional Errors 8-92
8.4.5.5 Miscellaneous Errors 8-101
8.4.6 Traps 8-110
8.4.6.1 NXM (Trap Thru 4) 8-111
8.4.6.2 Reserved Instruction (Trap Thru 10) . . . 8-111
8.4.6.3 Parity Error (Trap Thru 114) 8-111
8.4.6.4 Level 7 K Interrupt (Trap Through 134) . . 8-112
8.4.6.5 Control Bus Error Conditions (Hardware Detected) 8-112
8.4.6.5.1 Level 7 K Interrupt Printout 8-112
8.4.6.6 MMU (Trap Thru 250) 8-115

APPENDIX A INTERNAL CABLING DIAGRAM

A.1 HSC70 INTERNAL CABLING A-1

APPENDIX B EXCEPTION CODES AND MESSAGES

APPENDIX C GENERIC ERROR LOG FIELDS

C.1 GENERIC ERROR LOG FIELDS C-1
C.2 MSCP/TMSCP EVENT CODES C-2

APPENDIX D INTERPRETATION OF STATUS BYTES

D.1 INTRODUCTION D-1
D.2 OVERVIEW D-3
D.3 HOW TO USE THE STATUS CODE TABLES D-4
D.4 EXAMPLE EXAMINATION D-5

APPENDIX E HSC70 REVISION MATRIX CHART

E.1 INTRODUCTION E-1

INDEX

EXAMPLES

8-1 MSCP/TMSCP Error Message Format 8-14
8-2 Controller Error Message Example 8-16
8-3 SDI Error Printout 8-21
8-4 Disk Transfer Error Printout 8-36
8-5 Bad Block Replacement Error Printout 8-43
8-6 STI Communication or Command Error Printout . . 8-47

8-7 STI Formatter Error Log Printout 8-48
8-8 STI Drive Error Log Printout 8-49
8-9 Tape Drive Related Error Message 8-53

FIGURES

1-1 Redundant Cluster Configuration 1-2
1-2 HSC70 Cabinet - Front 1-3
1-3 HSC70 - Inside Front View 1-4
1-4 HSC70 Module Utilization Label Example 1-5
1-5 HSC70 - Inside Rear View 1-6
1-6 HSC70 External Interfaces 1-7
1-7 HSC70 Internal Software 1-8
1-8 Subsystem Block Diagram 1-11
1-9 Memory Map (M.std2 - L0117) 1-17
1-10 HSC70 Specifications 1-19
2-1 Operator Control Panel 2-1
2-2 Controls/Indicators - Inside Front Door 2-4
2-3 RX33 and DC Power Switch 2-5
2-4 Module LED Indicators 2-6
2-5 HSC70 Module Utilization Label Example 2-7
2-6 Module (DIP) Switches 2-9
2-7 Power Controller - Front Panel Controls 2-11
2-8 881 Rear Panel 2-13
3-1 Location of Circuit Breaker on the Power Controller 3-2
3-2 DC Power Switch Location 3-3
3-3 FRU Removal Sequence 3-4
3-4 RX33 Cover Plate Removal 3-6
3-5 RX33 Disk Drive Removal 3-7
3-6 RX33 Jumper Configurations 3-8
3-7 Operator Control Panel Removal 3-10
3-8 Card Cage Cover Removal 3-12
3-9 Location of Node Address Switches 3-13
3-10 Main Cooling Blower Removal 3-14
3-11 Airflow Sensor Assembly Removal 3-15
3-12 Power Controller Removal 3-17
3-13 Main Power Supply Cables - Disconnection . . . 3-19
3-14 Main Power Supply Removal 3-20
3-15 Auxiliary Power Supply Cable Disconnection . . 3-22
3-16 Auxiliary Power Supply Removal 3-23
4-1 Console Terminal Connection 4-2
4-2 Operator Control Panel Fault Code Displays . . . 4-5
6-1 P.ioj Switch Display Register Layout 6-113
6-2 P.ioj Control and Status Register Layout . . . 6-115
8-1 Operator Control Panel Fault Codes 8-3
8-2 HSC70 Boot Flowchart (1 of 4) 8-9
8-3 HSC70 Boot Flowchart (2 of 4) 8-10
8-4 HSC70 Boot Flowchart (3 of 4) 8-11
8-5 HSC70 Boot Flowchart (4 of 4) 8-12
8-6 Request Byte Field 8-23
8-7 Mode Byte Field 8-24
8-8 Error Byte Field 8-25

8-9 Controller Byte Field 8-26
8-10 RX33 Floppy Controller CSR Breakdown 8-82
8-11 MMSR0 Bit Breakdown 8-117
A-1 HSC70 Internal Cabling (1 of 5) A-2
A-2 HSC70 Internal Cabling (2 of 5) A-3
A-3 HSC70 Internal Cabling (3 of 5) A-4
A-4 HSC70 Internal Cabling (4 of 5) A-5
A-5 HSC70 Internal Cabling (5 of 5) A-6
D-1 Subsystem Exception K-Detected Error (1 of 2) . D-2
E-1 HSC70 Revision Matrix Chart (1 of 4) E-2

TABLES

1-1 Module Nomenclature 1-12
2-1 Functions of Logic Module LEDs 2-7
4-1 UPAR Register Addresses 4-10
4-2 Control Program Bits 4-11
4-3 Status of Requestors For Level 7 Interrupt . . . 4-12
5-1 ILTCOM Header Record 5-45
5-2 ILTCOM Data Patterns 5-46
6-1 Error Table 6-8
6-2 RX33 Error Code Table 6-9
7-1 DKUTIL Command Summary 7-7
7-2 DKUTIL Error Messages 7-17
8-1 L0111-0 (P.ioj) LEDs 8-4
8-2 L0117-0 (M.std2) LEDs 8-5
8-3 L0108-YA/YB (K.sdi/K.sti) LEDs 8-6
8-4 K.ci (LINK, PILA, K.pli) LEDs 8-6
8-5 MSCP/TMSCP Error Message Field Description . . . 8-14
8-6 MSCP/TMSCP Error Flags 8-15
8-7 MSCP/TMSCP Controller Error Message Field Description 8-16
8-8 SDI Error Printout Field Description 8-21
8-9 Request Byte Field Description 8-24
8-10 Mode Byte Field Description 8-25
8-11 Error Byte Field Description 8-26
8-12 Controller Byte Field Description 8-27
8-13 Disk Transfer Error Printout Field Description 8-36
8-14 Original Error Flags Field Description 8-38
8-15 Recovery Flags Field Definition 8-39
8-16 Bad Block Replacement Error Printout Field Definition 8-43
8-17 Replace Flags Bit Description 8-44
8-18 STI Communication or Command Error Printout Field Description 8-47
8-19 STI Formatter Error Log Field Description . . . 8-48
8-20 Formatter E Log 8-49
8-21 STI Drive Error Log Field Description 8-50
8-22 GEDS Text 8-50
8-23 STI Drive Error Log 8-51
8-24 Status Register Summary 8-83
C-1 Generic Error Log Fields C-1
C-2 Error Flags C-2

C-3 MSCP/TMSCP Event Codes C-3
D-1 K.ci Status Bytes D-6
D-2 K.sdi Status Bytes D-12
D-3 K.sti Status Bytes D-14

This manual describes the HSC70 subsystem. It describes HSC70 controls and indicators, error reporting, field replaceable units, troubleshooting, and diagnostic procedures. All information in this manual is informational/instructional and is designed to assist field service personnel with HSC70 maintenance. Operational theory is included wherever such background is helpful to field service.

Installation procedures, most HSC utilities, and indepth technical descriptions are not included in this manual. For source material on these and other subjects not within the scope of this manual, refer to the list of related documentation at the end of Chapter 1.

CHAPTER 1
GENERAL INFORMATION

1.1 INTRODUCTION

This chapter includes general information about the HSC70 Mass Storage Server including:

- Subsystem block diagrams
- Packaging and logic module descriptions
- Maintenance features
- Physical specifications
- Related documentation

1.2 GENERAL INFORMATION

Defined as a disk and/or tape subsystem, the HSC70 can interface with multiple hosts using the Computer Interconnect (CI) bus. In case of bus failure, two CI buses are included with the subsystem. Refer to Figure 1-1 for a sample five-node cluster configuration utilizing two HSC70s and three host computers. In this figure, all three hosts access both HSC70s over the CI bus, and through dual porting, both HSC70s can access the tape formatter and the disks.

The HSC70 supports a combination of eight disk and tape data channels. Each disk data channel supports four drives over the Standard Disk Interface (SDI). Each tape data channel supports four tape formatters over the Standard Tape Interface (STI). Depending upon which formatter is used, from one to four tape transports can be supported by each formatter.

Consult the HSC70 software release notes for the maximum number of tape formatters conforming to the STI bus. These software release notes are shipped with each HSC70 and with updates of the software.

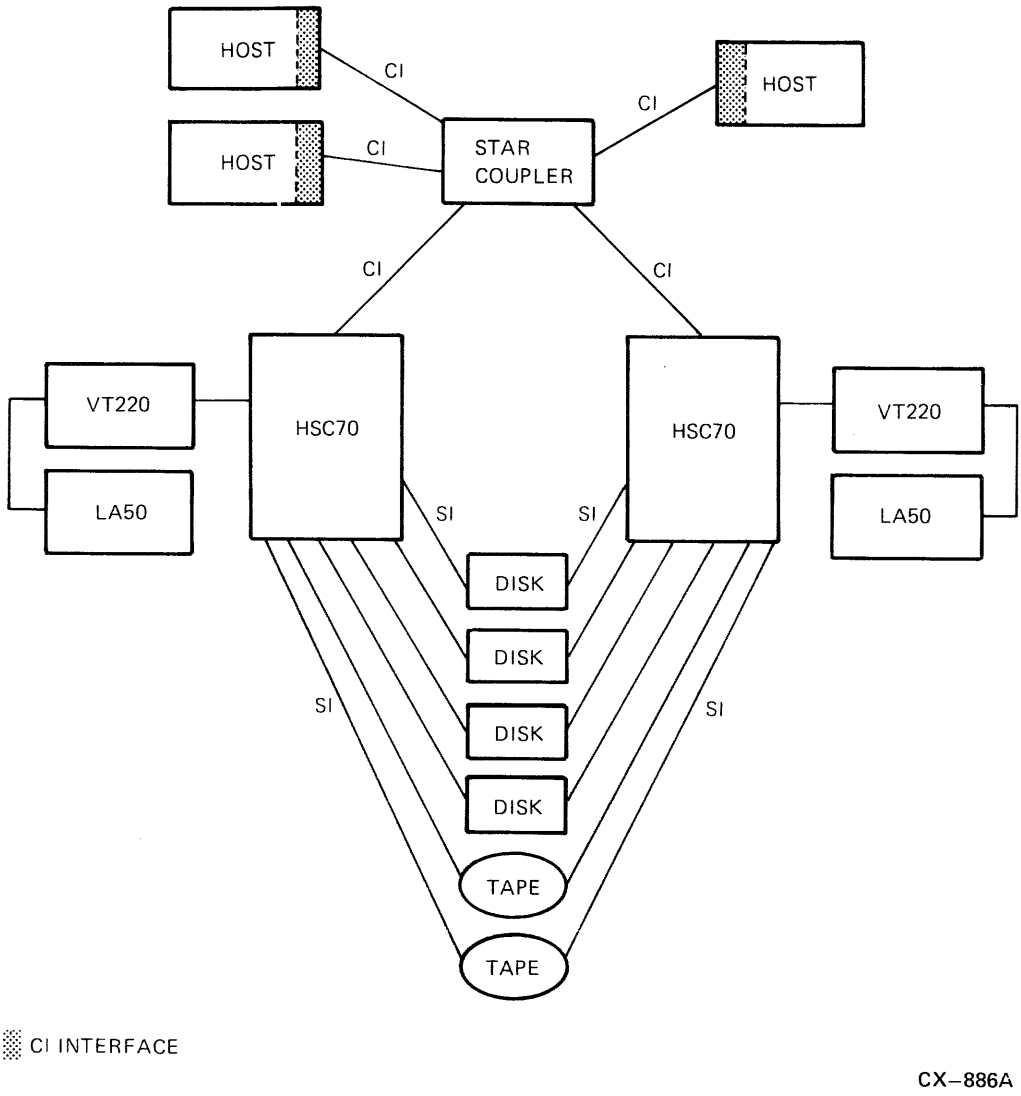


Figure 1-1 Redundant Cluster Configuration

1.2.1 HSC70 Cabinet Layout

HSC70 logic and power systems are housed in a modified H9642 cross-products cabinet with both front and rear access. See Figure 1-2 for front view of the cabinet.

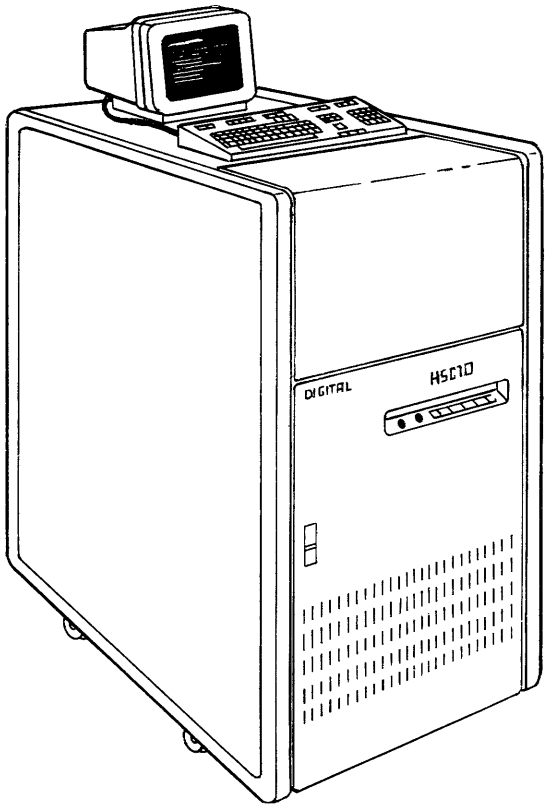


Figure 1-2 HSC70 Cabinet - Front

On the front of the cabinet are the operator control panel switches and indicators. Switch operation and indicator functions are described in Chapter 2.

To access the cabinet interior, open the front door with a key. The door key is part of the door-lock mechanism, part number 12-25411-01.

The upper right-hand portion of the cabinet houses the RX33 dual drives and connectors for the operator control panel.

The HSC70 contains two power supplies. Both are housed underneath the RX33. See Figure 1-3. Each power supply has a fan drawing air from the front of the cabinet across the power unit and exhausting it through a rear duct.

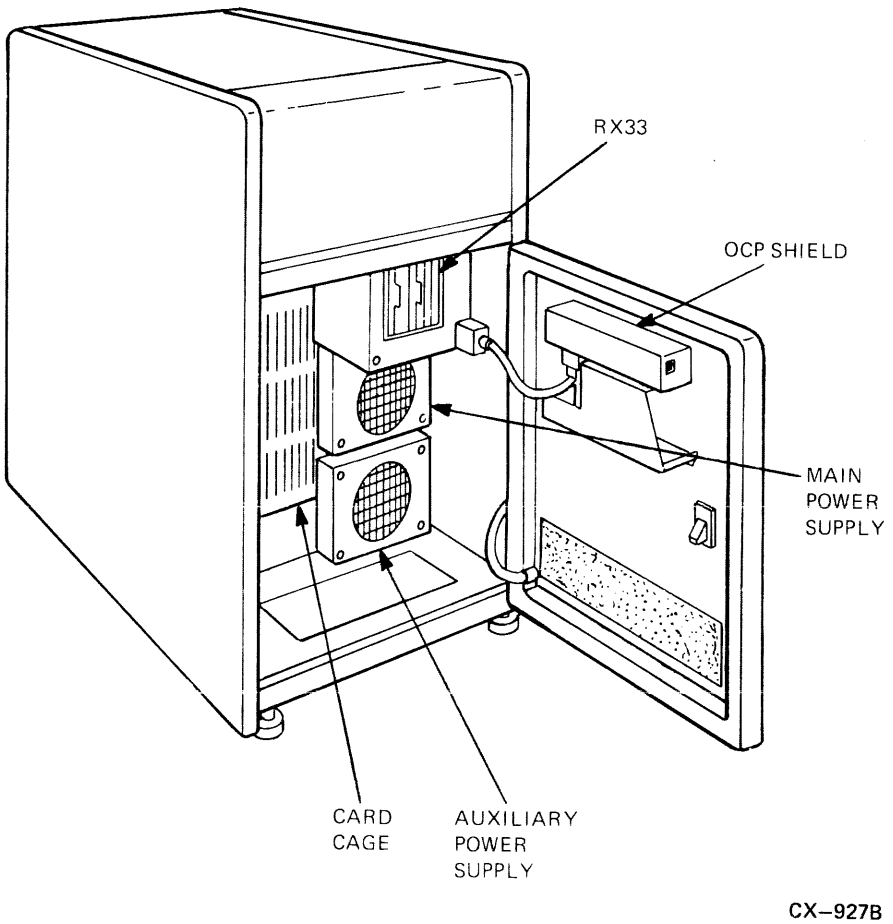


Figure 1-3 HSC70 - Inside Front View

A 14-slot card cage with a corresponding backplane provides housing for the HSC70 logic modules (L-series extended hex). When viewed from the front, the card cage occupies the upper left of the cabinet. Above the card cage is a module utilization label indicating the slot location of each module (Figure 1-4). All unassigned slots contain baffles.

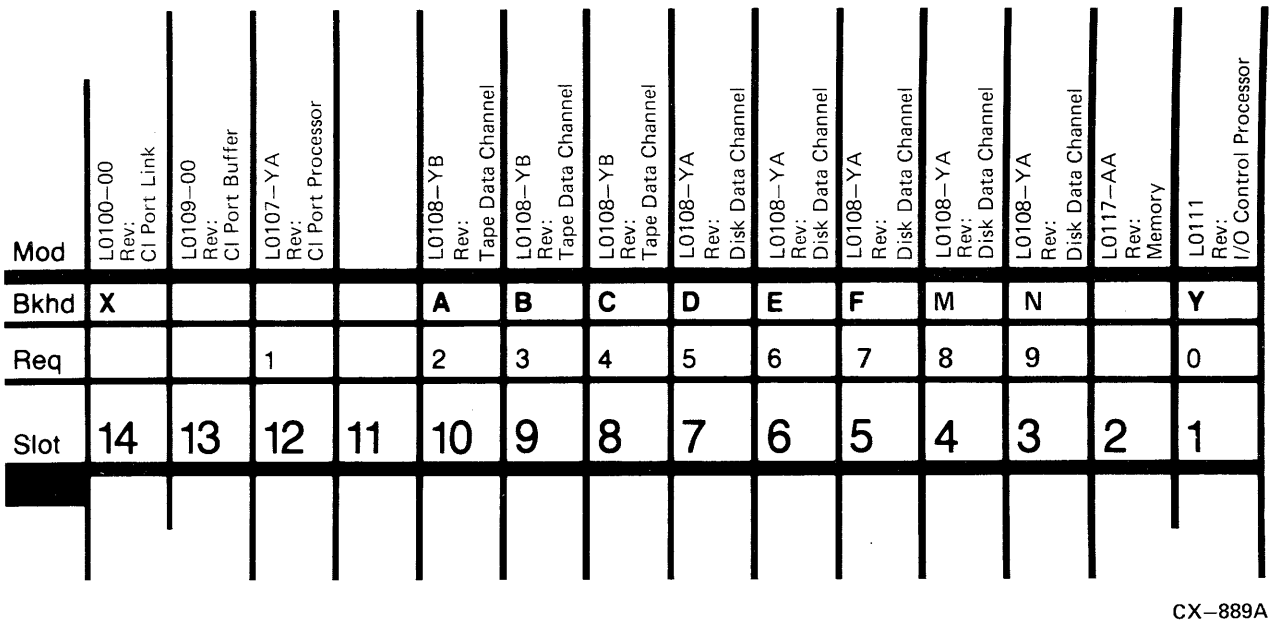


Figure 1-4 HSC70 Module Utilization Label Example

NOTE

Requestor slots A, B, C, D, E, F, M, and N, illustrated in Figure 1-4, are optional tape or disk data channels. Optional slot labels are blank when no module is present. Appropriate labels are provided with each data channel option ordered.

Logic modules are cooled by a blower mounted behind the card cage (Figure 1-5). Air is drawn in through the front door louver, up through the modules, and exhausted through the larger duct at the back.

NOTE

Figure 1-5 shows the blower motor outlet duct for current models. Early models have a smaller blower motor outlet duct.

Two levels of cable connections are found in the HSC70: backplane to bulkhead and bulkhead to outside the cabinet. All connections to the logic modules are made via the backplane. All cables attach to the backplane with press-on connectors.

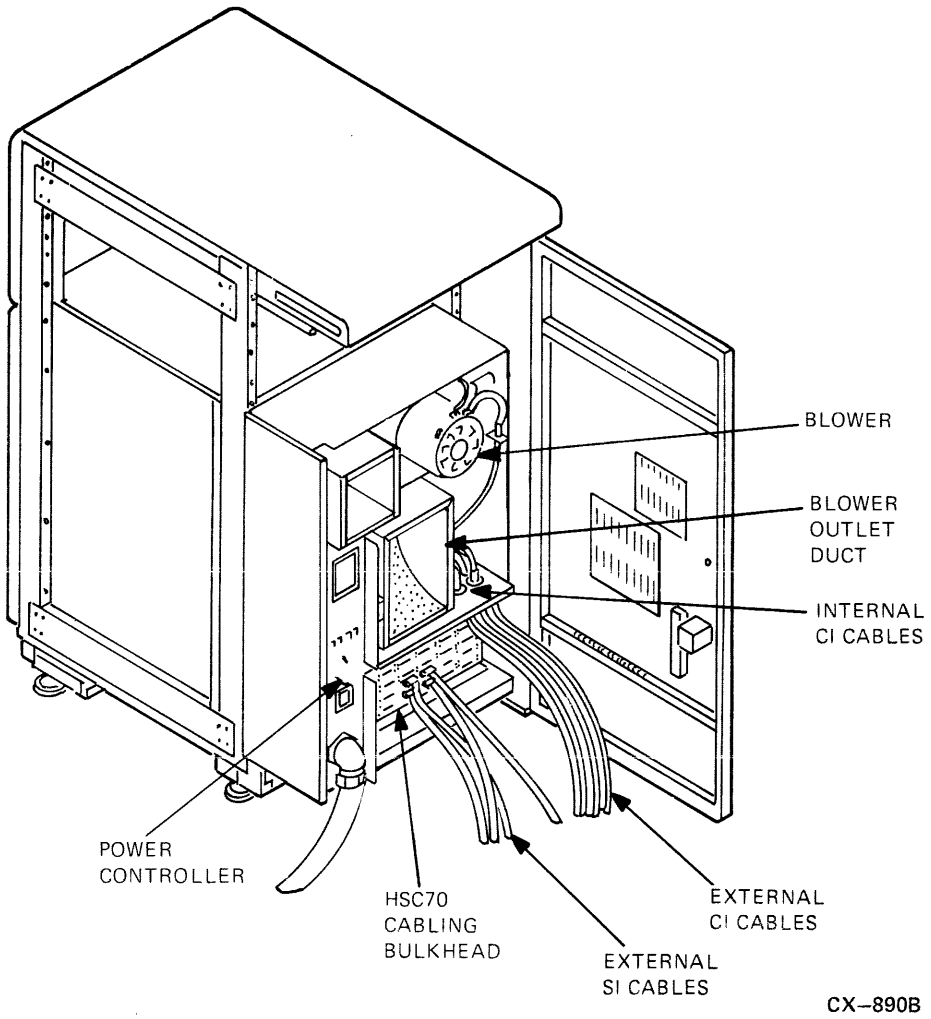


Figure 1-5 HSC70 - Inside Rear View

The power controller is in the lower left-hand rear corner of the HSC70. The power control bus, delayed output line, and noise isolation filters are housed in the power controller.

Exterior CI, SDI, and STI buses are shielded up to the HSC70 cabling bulkhead. These cables are attached to bulkhead connectors located at the bottom rear of the cabinet. From the interior of the I/O bulkhead connectors, unshielded cables are routed to the backplane.

1.2.2 External Interfaces

Figure 1-6 shows the external hardware interfaces used by the HSC70.

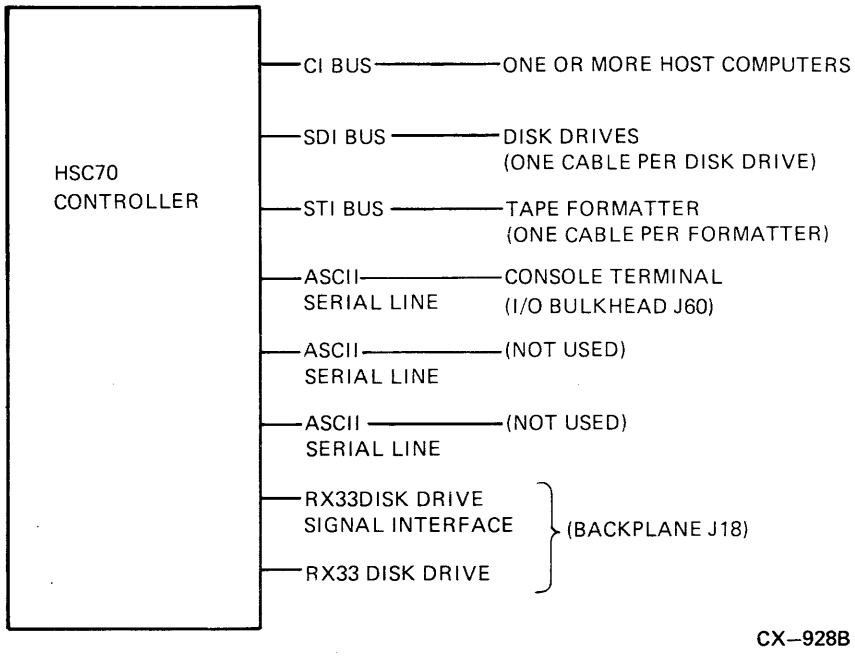


Figure 1-6 HSC70 External Interfaces

External interface lines include:

- **CI Bus** - Four coaxial cables (BNCIA-XX): two-path serial bus with a transmit and receive cable in each path. The communication path between system host(s) and the HSC70.
- **SDI Bus** - Four shielded wires for serial communication between the HSC70 and disk drives (one SDI cable per drive per controller) (BC26V-XX).
- **STI Bus** - Four shielded wires for serial communication between the HSC70 and the tape formatter (one STI cable per formatter) (BC26V-XX).

- **Serial Line Interface** - RS-232-C cable for console terminal communication with the I/O control processor module.

- **RX33 Signal Interface** - Cable linking RX33 drives with the RX33 controller located on the M.std2 module.

1.2.3 Internal Software

Major HSC70 software modules operating internally are shown at a block level in Figure 1-7. Each software module is described in the following lists.

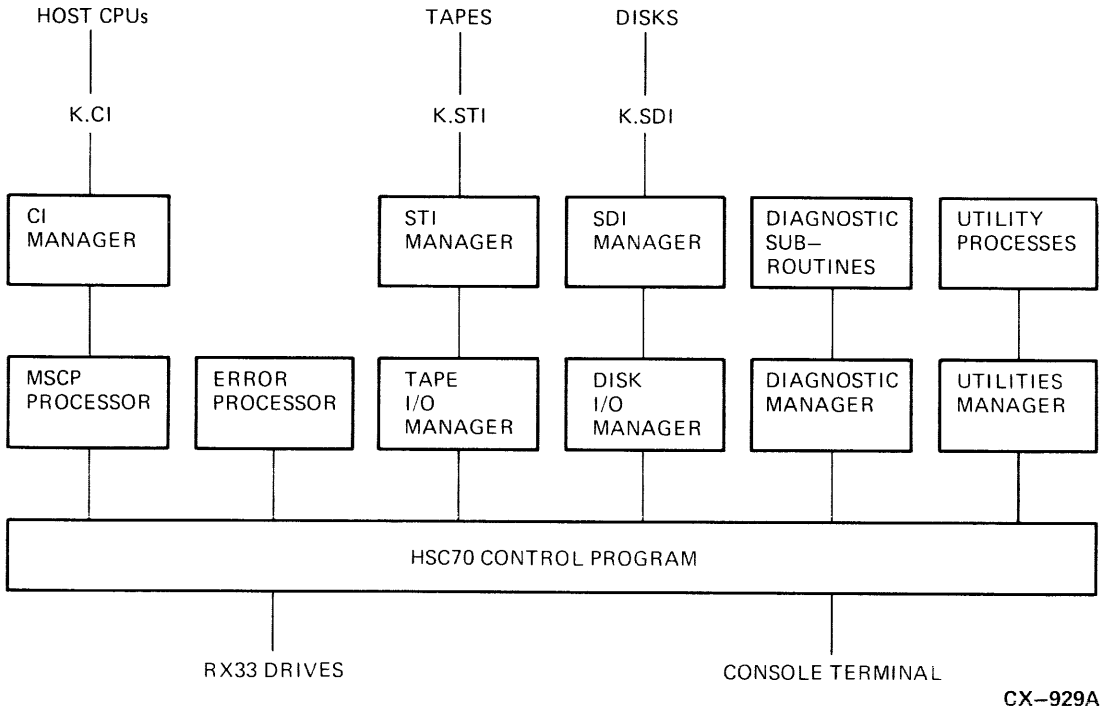


Figure 1-7 HSC70 Internal Software

- **HSC70 Control Program** - (found on the System diskette) is the lowest level manager of the subsystem. It provides a set of subroutines and services shared by all HSC70 processes. This program performs the following functions:

- Initializing and reinitializing the subsystem
- Managing the RX33 local storage media
- Executing all auxiliary terminal I/O
- Scheduling processes (both functional and diagnostic) for execution by the P.ioc
- Providing a set of system services and system subroutines to HSC70 processes.

Functional processes within the HSC70 communicate with each other and the HSC70 control program. They communicate through shared data structures and send/receive messages.

- **MSCP Processor** - is responsible for validating, interpreting, and routing incoming MSCP commands and dispatching MSCP completion acknowledgments.
- **CI Manager** - is responsible for handling virtual circuit and server connection activities.
- **Error Processor** - responds to all detected error conditions. It reports errors to the diagnostic manager and attempts to recover from errors (ECC, bad-block replacement, retries, etc.). When recovery is not possible, a diagnostic is run to determine if the subsystem can function without the failing resource. Then appropriate action is taken to remove the failing resource or to terminate subsystem operation.
- **Tape I/O Manager** - sets up the data transfer structures for tape operations and manages the physical positioning of the tape.
- **STI Manager** - handles the STI protocol, responds to attention conditions, and manages the online/offline status of the tape drives.
- **Disk I/O Manager** - performs the following functions:
  - Translates logical disk addresses into drive-specific physical addresses
  - Organizes the data-transfer structures for disk operations
  - Manages the physical positioning of the disk heads

- **SDI Manager** - performs the following:
  - Handles the SDI protocol
  - Responds to attention conditions
  - Manages the online/offline status of the disk drives

- **Diagnostic Manager** - is responsible for all diagnostic requests, for error reporting, and for error logging. It also provides decision-making and diagnostic-sequencing functions and can access a large set of resource-specific diagnostic subroutines.

- **Diagnostic Subroutines** - run under the control of the Diagnostic Manager and are classified as inline diagnostics.

- **Utilities Manager** - performs the following functions:
  - Interpreting incoming utility requests
  - Setting up the appropriate subsystem environment for operation of the requested utility
  - Invoking the utility process
  - Returning the subsystem to its normal environment upon completion of the utility execution

- **Utility Processes** - perform volume-management functions (formatting, disk-to-disk copy, disk-to-tape copy, tape-to-disk restore). They also handle miscellaneous operations required for modifying subsystem parameters or for analyzing subsystem problems (such as COPY, PATCH, and error dump).

1.2.4 Subsystem Block Diagram

The HSC70 is a multimicroprocessor subsystem with two shared memory structures, one for control and one for data. In addition, the HSC70 I/O control processor fetches its own instructions from a private (program) memory. Figure 1-8 shows an HSC70 block diagram and the position of each component in the subsystem.

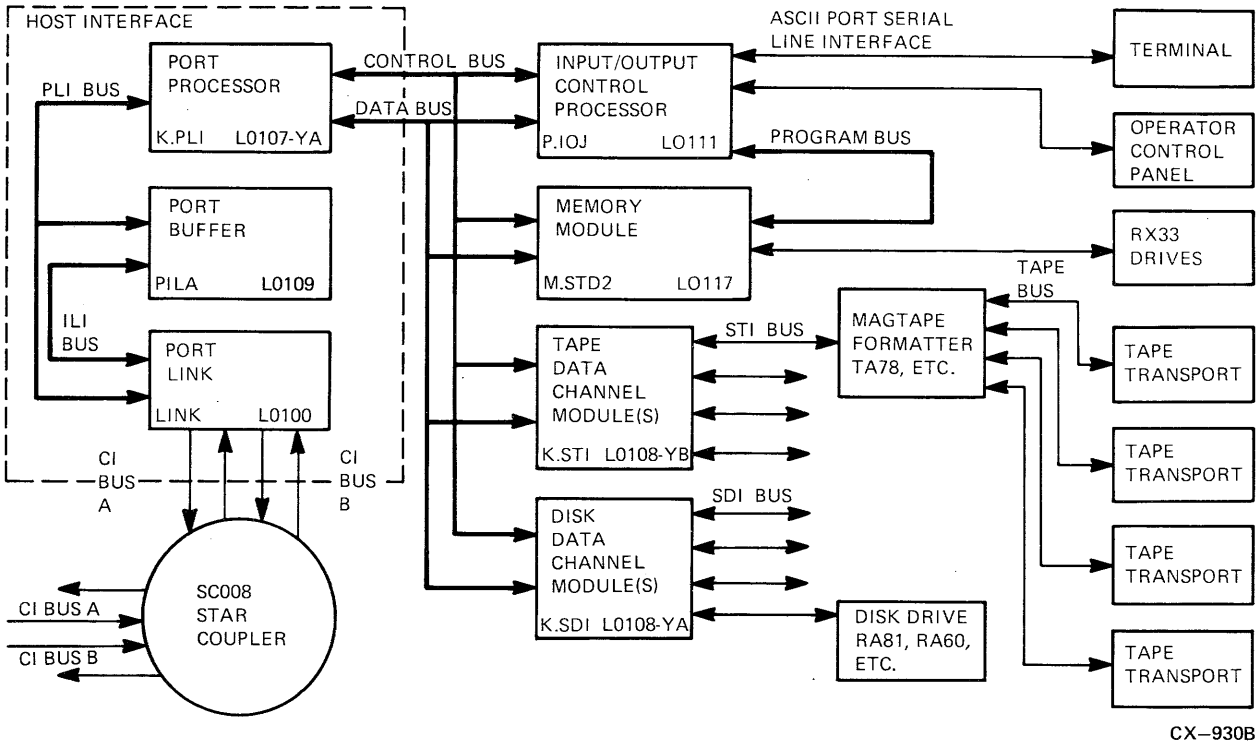


Figure 1-8 Subsystem Block Diagram

1.3 MODULE DESCRIPTIONS

This section describes each of the HSC70 logic modules. References to modules by their engineering terms appear throughout HSC70 documentation as well as on diagnostic printouts. For this reason, the engineering term is shown in parentheses after the formal name for each module. These relationships are also indicated in Figure 1-8 and Table 1-1.

Table 1-1 Module Nomenclature

Module Name	Engineering Name	Module Designation
Port Link	LINK or Interprocessor Link Interface	L0100
Port Buffer	PILA	L0109
Port Processor	K.pli	L0107
Disk Data Channel	K.sdi	L0108-YA (HSC5X-BA)
Tape Data Channel	K.sti	L0108-YB (HSC5X-CA)
Input/Output Control Processor	P.ioj	L0111
Memory	M.std2	L0117
Host Interface	K.ci	Consists of Port Link, Port Buffer, and Port Processor Modules

1.3.1 Port Link Module (LINK) Functions

The port link module (L0100), a part of the host interface module set (K.ci) performs the following functions:

- **Serialization/deserialization, encoding/decoding, dc isolation** - permits transmission of a self-clocking stream over the CI. (Information transmitted over the CI bus is serialized and Manchester encoded.) The driver circuit includes a transformer for ac coupling the encoded signal to the coaxial cable. Information received from a CI transmission is decoded and converted to bit-parallel form. The circuitry also provides carrier detection for determining when the CI is in use by another node.

- **Cyclic redundancy check (CRC) generation/checking** - checks the 32-bit CRC character generated and appended to a message packet when it is received. An incorrect CRC means either errors were induced by noise or a packet collision occurred.

- **ACK/NAK generation** - generates an ACK upon receipt of a packet addressed to the LINK if the following conditions exist:

  - Error-free CRC
  - Buffer space available for the message

  Upon receipt of a packet addressed to this node, a NAK is generated if the following conditions exist:

  - Error-free CRC
  - No buffer space available for the message

  No response is made if a packet addressed to this node is received with CRC error.

- **Packet transmission** - performs the following functions:

  - Executes the CI arbitration algorithm
  - Transmits the packet header
  - Moves the stored information from the transmit packet buffer to the Manchester encoder
  - Calculates and appends the CRC to the end of the packet
  - Receives the expected ACK packet

- **Packet reception** - performs the following functions:

  - Detects the start of the CI transmission
  - Detects the sync characters
  - Decodes the packet header information
  - Checks the CRC
  - Moves the data from the Manchester decoder
  - Returns the appropriate ACK packet

The port link module interfaces via line drivers/receivers directly to the CI coaxial cables. On the HSC70 interior side, the port link module interfaces to the port buffer module through a set of interconnect link (ILI) signals. The port link module also interfaces to the port processor module (indirectly through the port buffer module) using a set of port link interface (PLI) signals.

1.3.2 Port Buffer Module (PILA) Functions

The port buffer module (L0109) provides a limited number of high-speed memory buffers to accommodate the difference between the burst data rate of the CI bus and HSC70 internal memory buses. It also interfaces to the port link (CI link) module via the ILI signals and the port processor module via port/link interface (PLI) signals.

1.3.3 Port Processor Module (K.pli) Functions And Interfaces

The port processor module (L0107-YA) performs the following functions:

- Executes and validates low-level CI protocol
- Moves command/message packets to/from HSC70 control memory and notifies the correct server process of incoming messages
- Moves data packets to/from HSC70 data memory

The port processor module interfaces to three buses:

- PLI bus interfaces the port buffer and port link modules
- Control memory bus interfaces HSC70 control memory
- Data memory bus interfaces HSC70 data memory

1.3.4 Disk Data Channel Module (K.sdi) Functions

Disk data channel module (L0108-YA) operation is controlled by an onboard microprocessor with a local programmed read-only memory (PROM). This data channel module performs the following functions:

- Transmits control and status information to the disk drives
- Monitors real-time status information from the disk drives

- Monitors in real time the rotational position of all the disk drives attached to it
- Transmits data between HSC70 data memory and the disk drives
- Generates and compares error correction code (ECC) and error detection code (EDC) during data transfers

Commands and responses pass between the disk data channel microprocessor and other internal HSC70 processes through control memory. The disk data channel module interfaces to the control memory bus and to the data memory bus. It can also interface to four disk drives with four individual SDI buses. Currently, combinations of up to eight disk data channel modules are possible in the HSC70. Configuration guidelines are found in the *HSC70 Installation Manual*.

1.3.5 Tape Data Channel Module (K.sdi) Functions

Tape data channel module (L0108-YB) operation is controlled by an onboard microprocessor with a local programmed read-only memory (PROM). The tape data channel performs the following functions:

- Transmits control and status information to the tape formatters
- Monitors real-time status information from the tape formatters
- Transmits data between the data memory and the tape formatters
- Generates and compares the EDC during data transfers

Commands and responses pass between the tape data channel microprocessor and other internal HSC70 processes through control memory. The tape data channel module interfaces to the control memory bus and to the data memory bus. Maximum configurations are outlined in the software release notes.

1.3.6 Input/Output (I/O) Control Processor Module (P.ioj) Functions

The I/O control processor module (L0111) uses a PDP-11 ISP (J-11) processor with memory management and memory interfacing logic. This processor executes the HSC70 internal software. Also, the I/O control processor module contains the following:

- Bootstrap read-only memory (ROM)
- Arbitration and control logic for the control and data buses
- Program-addressable registers for subsystem initialization and operator control panel communications
- Handles all parity checking and generation for its accesses to memory
- Contains program memory instruction cache, 8 Kbytes of direct map high-speed memory

The I/O control processor module interfaces to:

- Program memory on the program memory bus
- Control memory through the signals of the backplane control bus
- Data memory through signals of the backplane data bus
- RX33 disk drives
- Console terminal RS-423 compatible signal levels

1.3.7 Memory Module (M.std2) Functions

The memory module (L0117) contains three separate and independent memories each residing on a different bus within the HSC70. In addition, the memory module contains the diskette controller. The three memories and diskette controller are known as:

- **Control Memory (M.ctl)** - Two banks of 256 Kbytes of dynamic RAM for subsystem control blocks and interprocessor communication structures storage.
- **Data Memory (M.dat)** - 512 Kbytes of status RAM to hold the data from/to a data channel module.
- **Program Memory (M.prog)** - 1 megabyte of RAM for the control program loaded from the RX33 diskette.
- **RX33 Diskette Controller (K.rx)** - resides on the program bus and performs direct memory access word transfers when reading or writing data to the RX33 diskette.

Using physical addresses, the memory space allocations for the three memories are illustrated in Figure 1-9.

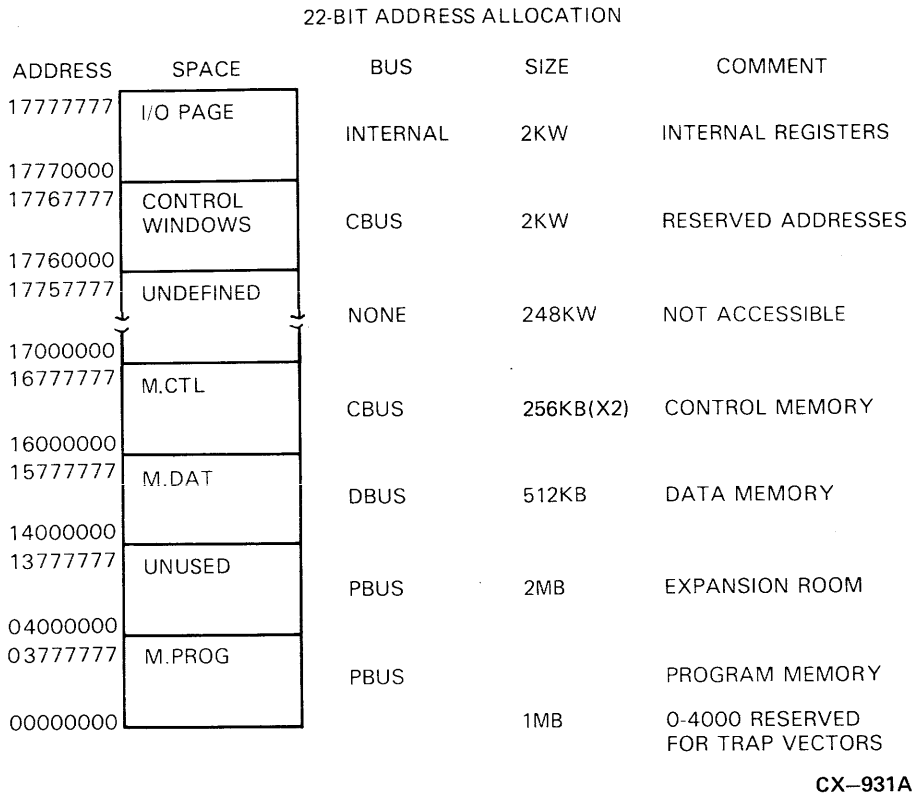


Figure 1-9 Memory Map (M.std2 - L0117)

NOTE

Two completely redundant memory banks make up control memory. Only one bank at a time is usable during functional operation. Bank failure detection and bank swapping are done at boot time.

Interface to control memory is by the backplane control bus and to data memory by the backplane data bus. The interface to the I/O control processor local program memory is via a set of backplane signals to the program memory module. In addition, the memory module houses the control circuitry for the RX33 disk drives.

1.4 HSC70 MAINTENANCE STRATEGY

Maintenance of the HSC70 is accomplished with field replaceable units (FRUs). Procedures for removal and replacement are described in Chapter 4. Field service personnel should not attempt to replace or repair component parts within FRUs.

Isolation of solid failures can be accomplished efficiently due to the logical partitioning of the modules and extensive internal diagnostics. In addition to the device-resident diagnostics, the HSC70-resident offline diagnostics are available to support and verify corrective maintenance decisions.

1.4.1 Maintenance Features

The following features assist in troubleshooting the HSC70:

- Self-contained and self-initiated diagnostics
- Operator control panel fault code display
- Console terminal
- Module LED indicators

Various levels of diagnostics execute in the HSC70. Read-only memory (ROM) diagnostics test each microprocessor in the disk and tape data channels, port processor, and I/O processor modules. Pressing the HSC70 Init button starts all internal ROM diagnostics that test 95 percent of the HSC70.

The OCP or the console terminal displays any failures. If further diagnostics are needed, use the terminal to initiate diagnostics stored on the RX33 diskettes.

The RX33 loads all HSC70 troubleshooting diagnostics upon operator demand to check SDI/STI communication and interaction between the HSC70 and disk or tape. Powerup, subsystem initialization, or operator command can initiate these diagnostics. Also, certain resource failure detections can initiate them automatically.

The HSC70 subsystem allows logical assignment of a disk drive or tape formatter to the diagnostics. Inline diagnostics allow drive diagnosis even though other active drives are connected to the HSC70.

Background (periodic) diagnostics test HSC70 logic not currently in use by the subsystem. Failures cause the HSC70 to reboot and execute the initialization diagnostics.

Requestor detected data memory errors cause an initiation of the inline memory diagnostics to test the buffer causing the error. Failures found in any data buffer cause removal of that buffer from service. If no failure is found, the tested buffer is returned to service, with one exception. If the same buffer is sent to test twice, it is retired from service even though no failure is found.

1.4.2 HSC70 Specifications

Figure 1-10 lists the HSC70 physical and environmental specifications.

DESCRIPTION	OPTION DESIGNATION
HSC70 MASS STORAGE SERVER	HSC70–AA = 60 HZ, 120/208V
	HSC70–AB = 50 HZ, 380–415V

MECHANICAL

MOUNTING CODE	WEIGHT		HEIGHT		WIDTH		DEPTH		CAB TYPE (IF USED)
	LBS	KG	IN	CM	IN	CM	IN	CM	
FS	400	181.2	42	106.7	21.3	54.1	36	91.4	MODIFIED H9642

POWER (AC)

AC VOLTAGE NOMINAL	AC VOLTAGE TOLERANCE	FREQUENCY & TOLERANCE	PHASE	STEADY–STATE CURRENT (RMS)	POWER CONSUMPTION (MAX)
120/208	104–128/180–222	60 HZ ± 1	3	SEE BELOW	2250 WATTS
380–415	331–443	50 HZ ± 1	3	SEE BELOW	2250 WATTS

POWER (AC)

STEADY-STATE CURRENT (MAX AMPS) BY PHASE			
120/208V	PHASE A = 0.7	380–415V	PHASE A = 0.44
	PHASE B = 12.4		PHASE B = 6.8
	PHASE C = 11.8		PHASE C = 6.4
	NEUTRAL = 17.1		NEUTRAL = 9.4

POWER (AC)

PLUG TYPE (NEMA NO.)	POWER CORD LENGTH	INTERRUPT TOLERANCE	APPARENT POWER (KVA)
NEMA – L21 – 30P	15 FT (4.5 M)	4MS (MIN)	3.0 (KVA)

POWER (AC)

INRUSH CURRENT 60HZ	INRUSH CURRENT 50HZ	SURGE DURATION
175 A PEAK	175 A PEAK	1 CYCLE

DEVICE ENVIRONMENT

TEMPERATURE		RELATIVE HUMIDITY		RATE OF CHANGE		HEAT DISSIPATION	
OPERATING*	STORAGE	OPERATING	STORAGE	TEMP	HUMIDITY	BTU/HR	KJ/HR
59 – 90° F	–40 – +151°F	20 – 80%	<96%	20° F/HR	20%/HR	7675	8100
15 – 32° C	–40 – +66° C			11° C/HR			

DEVICE ENVIRONMENT

ALTITUDE (MAX)		AIR VOLUME (AT INLET)		AIR QUALITY
OPERATING	STORAGE	FT^3 /MIN	M3/MIN	PARTICLE COUNT (MAX)
8000 FT	16,000 FT	210	5.92	N/A
2.4 KM	4.9 KM			

* ALTITUDE CHANGES: DE-RATE THE MAXIMUM TEMPERATURE 1.8° C PER THOUSAND METERS (1.0° F PER THOUSAND FEET).

CX–912C

Figure 1-10 HSC70 Specifications

1.5 HSC70 RELATED DOCUMENTATION

Documents related to the HSC70 are available under the following part numbers:

- *HSC User Guide* AA-GMEAA-TK
- *HSC70 Installation Manual* EK-HSC70-IN
- *HSC70 Illustrated Parts Breakdown* EK-HSC70-IP
- *Star Coupler User Guide* EK-SC008-UG
- *VT220 Owners Manual* EK-VT220-UG
- *VT220 Programmer Pocket Guide* EK-VT220-HR
- *VT220 Installation Guide* EK-VT220-IN
- *Installing and Using the LA50 Printer* EK-0LA50-UG
- *LA50 Printer Programmer Reference Manual* EK-0LA50-RM

These documents (except for the User Guide) can be ordered from Publication and Circulation Services, 10 Forbes Road, Northboro, Massachusetts 01532 (RCS Code: NR12, Mail Code: NR03/W3).

The User Guide can be ordered from the Software Distribution Center, Digital Equipment Corporation, Northboro, Massachusetts 01532

NOTE

Please consult the HSC Software Release Notes for the latest hardware revision levels.

CHAPTER 2
HSC70 CONTROLS/INDICATORS

2.1 INTRODUCTION

This chapter describes the controls and indicators located in five areas of the HSC70:

- Operator Control panel (OCP)
- Inside front door
- RX33 disk drives
- Logic modules
- Power controller

2.2 OPERATOR CONTROL PANEL (OCP)

Figure 2-1 illustrates the controls and indicators on the Operator Control panel (OCP).

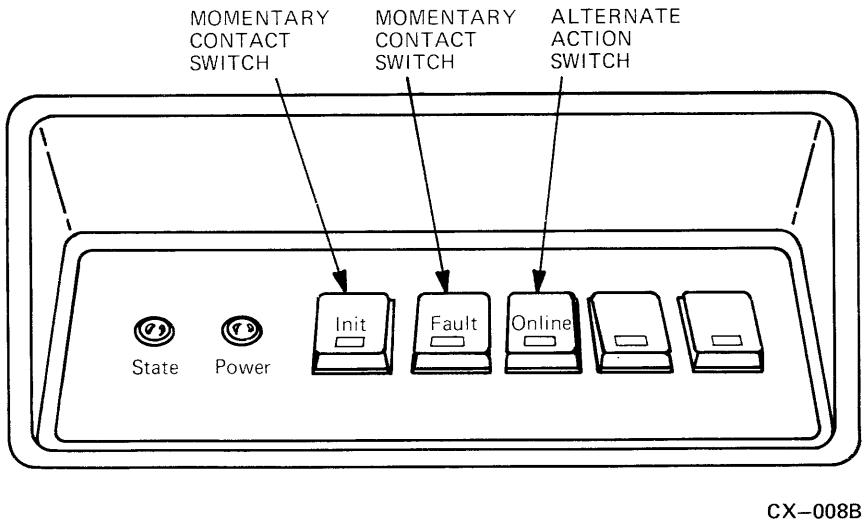


Figure 2-1 Operator Control Panel

The OCP controls and indicators are described in the following list.

- **State and Init Indicators** - describe the state of the HSC70. Under runtime conditions, the Init indicator is off while the State indicator is pulsing. During initialization, these indicators change to reflect the current initialization phase of the subsystem.

- **Init Switch** - causes the HSC70 to start its initialization routine. The Secure/Enable switch must be in the ENABLE position for this switch to be operational.

- **Power Indicator** - goes OFF if the dc voltage levels drop below one-third of minimal. The power indicator is driven from a dc comparator circuit on the I/O Control Processor module (L0111) that constantly monitors the +5, +12, and -5.2 voltages. The power indicator is also driven by a logic gate that monitors the Power Fail signal from the power supplies. If this signal is asserted, the power indicator goes OFF.

NOTE

The power indicator ON does not mean these voltages are within specification (±5 percent).

- **Fault Indicator and Switch** - comes on when the HSC70 logic detects a fault. The Fault switch is used for the OCP lamp test.

  - **Fault Codes** - When the Fault switch is pressed and released, the lamps in Init, Online, Fault, and the two blanks function as an error display. If the fault code is a hard fatal error, the fault code blinks on and off until the HSC70 is powered down or the Init switch is pressed again.

    If the displayed fault code is a soft nonfatal failure, the fault code clears on subsequent toggling of the Fault switch. Multiple soft fault codes can be queued in the fault code buffer. Subsequent toggling of the Fault switch displays each soft fault code until the buffer is emptied.

Soft fault codes are identified by the Fault indicator ON (or displayed fault code) while the State indicator is pulsing. With soft faults, the HSC continues to operate without the use of the failing resource. Hard fault codes are identified by the fault indicator ON (or displayed fault code) while the HSC State indicator is **not** pulsing. With hard faults the HSC will **not** continue operation until the failure is remedied.

Error codes associated with the OCP display are defined in Chapter 8 and in Chapter 4.

- **Lamp Test** - Pushing and holding the Fault switch causes all the OCP indicators to light and function as a lamp test. Even if the Fault indicator is already on before the switch is pushed, the lamp test can be executed.

- **Online Switch** - puts the HSC70 logic in the available state when pushed to the IN position and allows a host to establish a virtual circuit with the HSC70. When this switch is released to the OUT position, no new virtual circuits can be made.

- **Online Indicator** - shows a virtual circuit exists between the HSC70 and a host CPU when the Online indicator is on. When this indicator is off, no virtual circuits are established with any host.

- **Blank Indicators** - form the lowest two bits of a 5-bit fault code.

2.3 INSIDE FRONT DOOR CONTROLS/INDICATORS

Figure 2-2 shows the controls and indicators available when the front door is opened.

- **Secure/Enable Switch** - disables the Init switch from the OCP when in the SECURE position. Also, the SET utility program cannot run, and the BREAK character from the terminal is disabled. With the Secure/Enable switch in the ENABLE position, the Init switch and all the utility programs can be used.

  The SHOW utility is operable with the Secure/Enable switch in either position.

- **Enable Indicator** - indicates the Secure/Enable switch is in the ENABLE position when the Enable LED is illuminated (all switches can be used). When the Enable indicator is off, the OCP is secure.

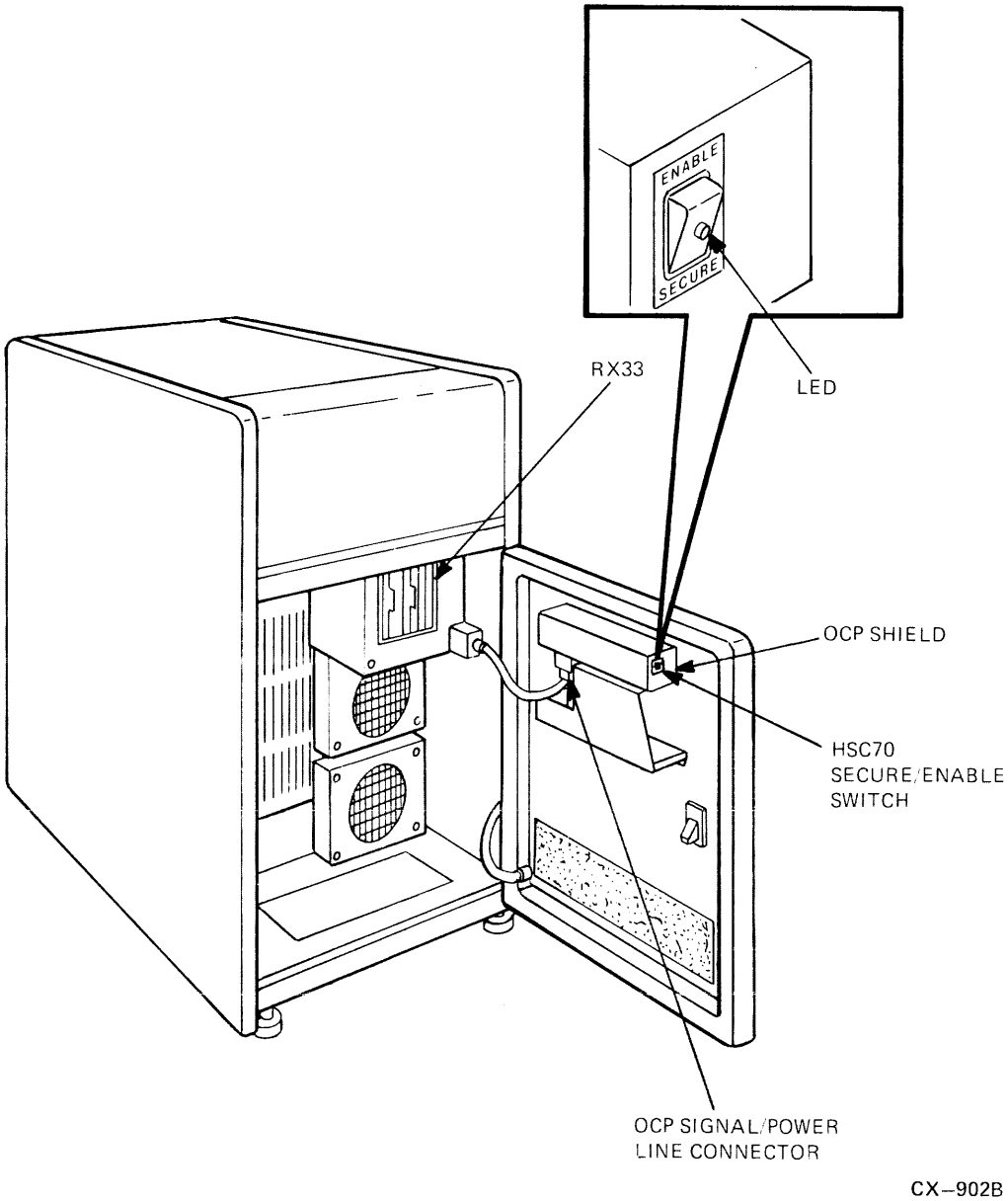


Figure 2-2 Controls/Indicators - Inside Front Door

- **RX33 LEDS** - are lit to indicate which particular drive is in use. There is an LED on the front panel of each drive. When not in use, the RX33 diskettes are stored inside the front door. See Figure 2-3.

- **DC Power Switch** - is located on the left side of the RX33 housing. See Figure 2-3. When the DC Power switch is in the 0 position, the HSC70 is without dc power. Moving the switch to the 1 position restores dc power.

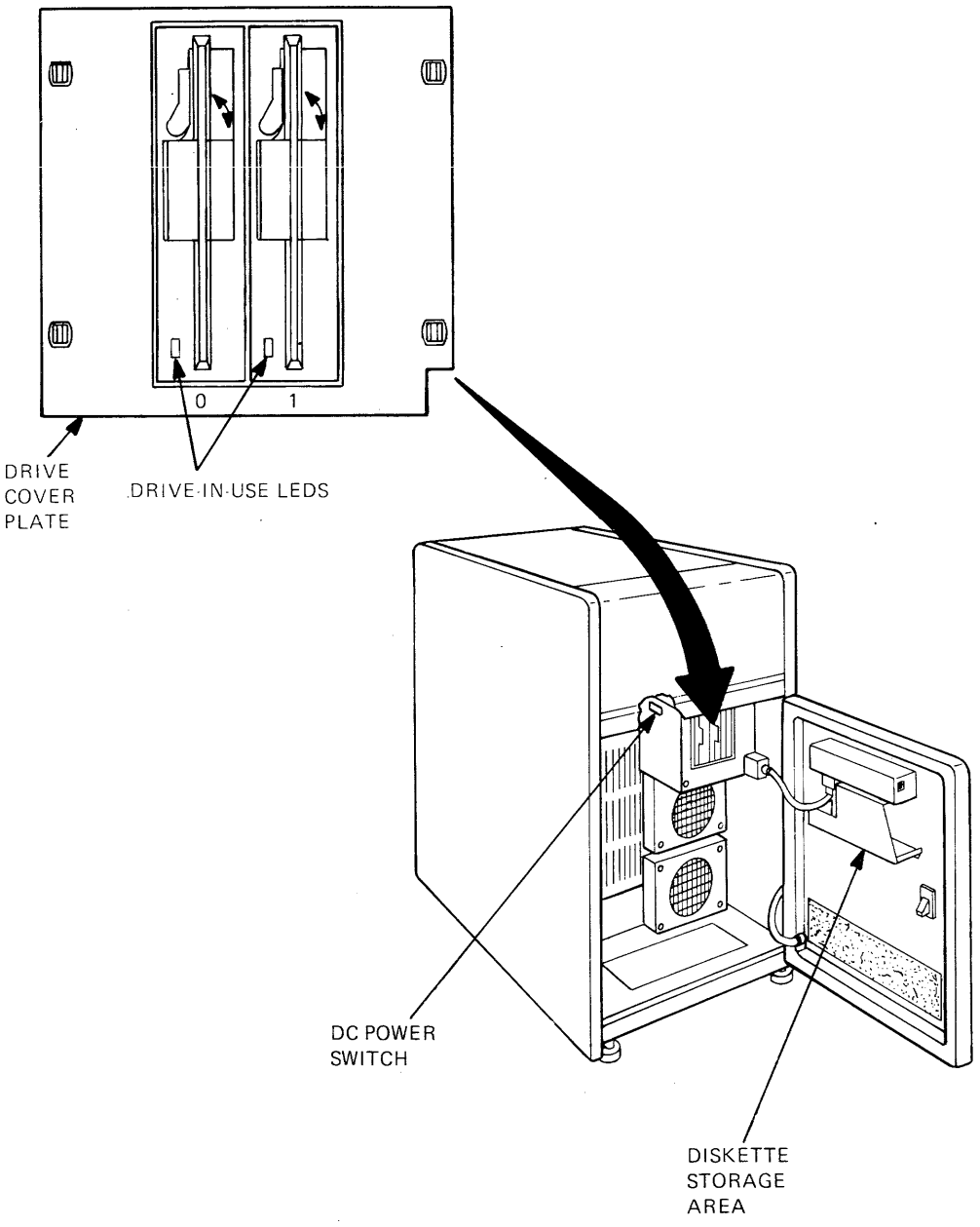


Figure 2-3 RX33 and DC Power Switch

2.4 MODULE INDICATORS AND SWITCHES

All logic modules have at least one LED to indicate board status. Refer to Figure 2-4 for the locations of these LEDs and the Module Utilization Label. Additionally, two of these logic modules contain specific switches. Figure 2-5 shows the slot location for each of the modules. Table 2-1 shows the functions of the various module LEDS.

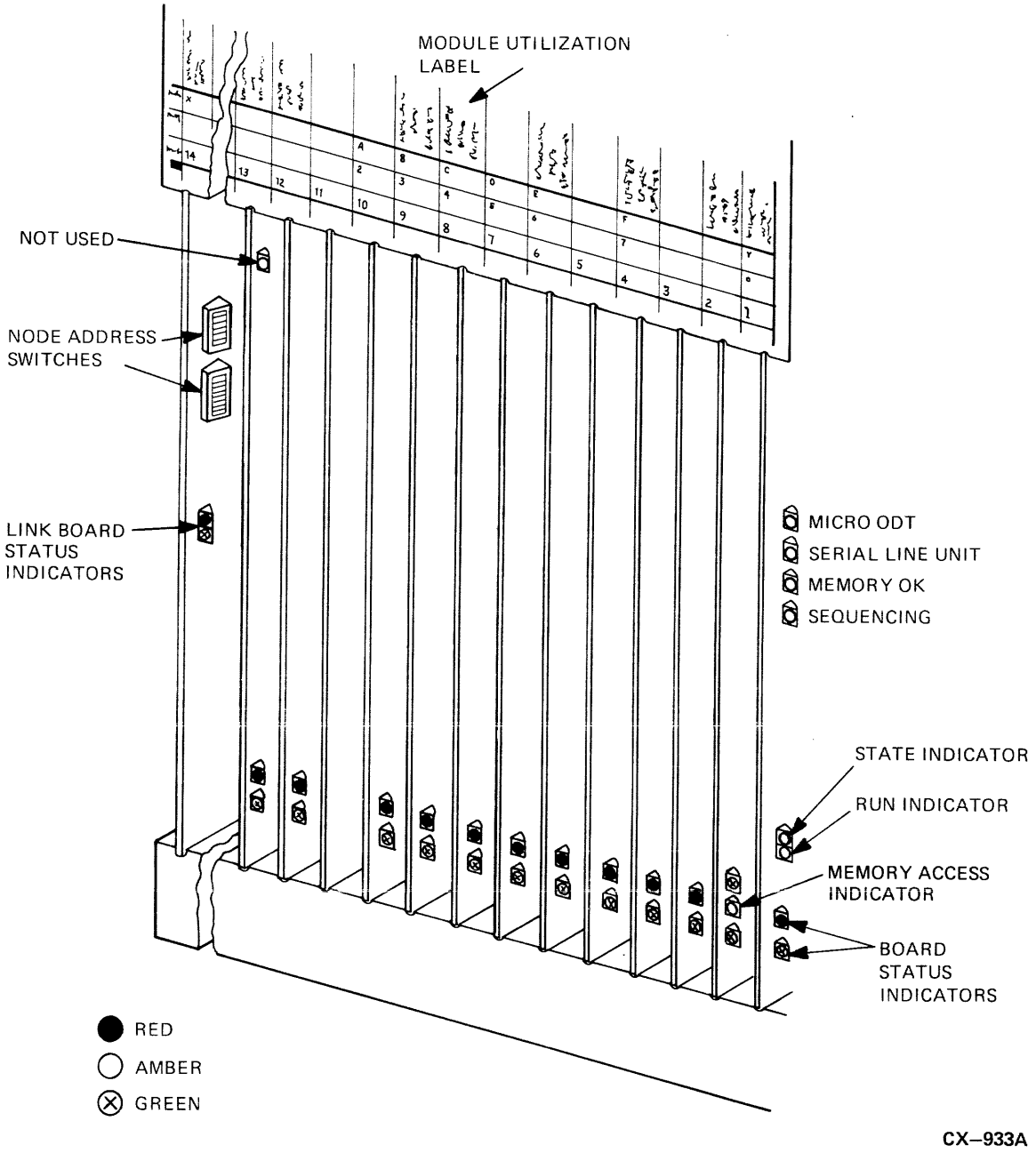
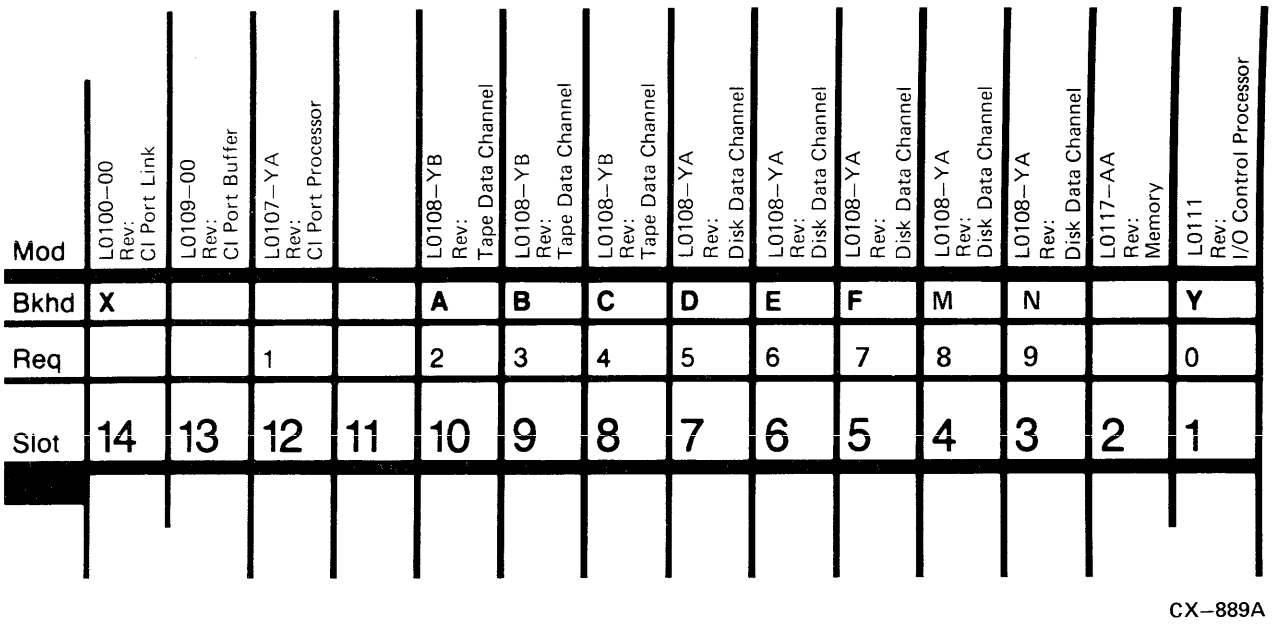


Figure 2-4 Module LED Indicators



CX-889A

Figure 2-5 HSC70 Module Utilization Label Example

Table 2-1 Functions of Logic Module LEDs

Module	Color	Function
L0111	D1 Amber	Micro-ODT -- Used during J-11 power-up microdiagnostics.
	D2 Amber	Terminal Port OK -- Used during J-11 power-up microdiagnostics.
	D3 Amber	Memory OK -- Used during J-11 power-up microdiagnostics.
	D4 Amber	Sequencing Indicator -- Used during J-11 power-up microdiagnostics.
	D5 Amber	State Indicator -- mirrors the OCP State indicator.
	D6 Amber	Run Indicator -- pulses at the on-board microprocessor run rate.
	D7 Red	Board Status -- indicates an inoperable module except during initialization when it comes on during module testing.

Module	Color	Function
	D8 Green	Board Status -- indicates the module has passed all applicable diagnostics.
L0117	Green	Board Status -- indicates the operating software is running and has successfully tested this module.
	Amber	Indicates "Memory Active" - lit during every memory cycle.
	Red	Board Status -- indicates an inoperable module except during initialization when it comes on during module testing.
L0108-YA L0108-YB	Green	Board Status -- indicates the operating software is running and that self-test module microdiagnostics have completed successfully.
	Red	Board Status -- indicates an inoperable module except during initialization when it comes on during module testing.
L0107-YA	Green	Board Status -- indicates the operating software is running and that self-test module microdiagnostics have completed successfully.
	Red	Board Status -- indicates an inoperable module except during initialization when it comes on during module testing.
L0109	Green	Board Status -- indicates the operating software is running and that all applicable diagnostics have completed successfully.
	Red	Board Status -- indicates an inoperable module except during initialization when it comes on during module testing.
	Amber	Always on. (Used only for engineering test purposes.)
L0100	Green	Indicates the node is either transmitting or receiving. Dims or brightens relative to the amount of local CI activity.
	Red	Indicates the module is in the Internal Maintenance mode.

2.4.1 Module Switches

Specific switches are found on L0100, L0107, and L0109, as follows:

- **Port Link Module (L0100)** - Figure 2-4 shows the location of the CI node address switches mounted on the L0100 module. Both sets of switches must be identically set to avoid CI addressing errors. The chosen address must not exceed the current maximum of 15 (decimal). Addresses higher than 15 cause port link module faults on the OCP (error code of 25 octal).

- **Port Link and Port Link Buffer Modules (L0107 and L0109)** - Both the L0107 and L0109 modules have dual inline pack (DIP) switches to indicate the hardware revision level. DIP switch positions should not be changed except as directed by a Field Change Order. Figure 2-6 shows the location of the these switches.

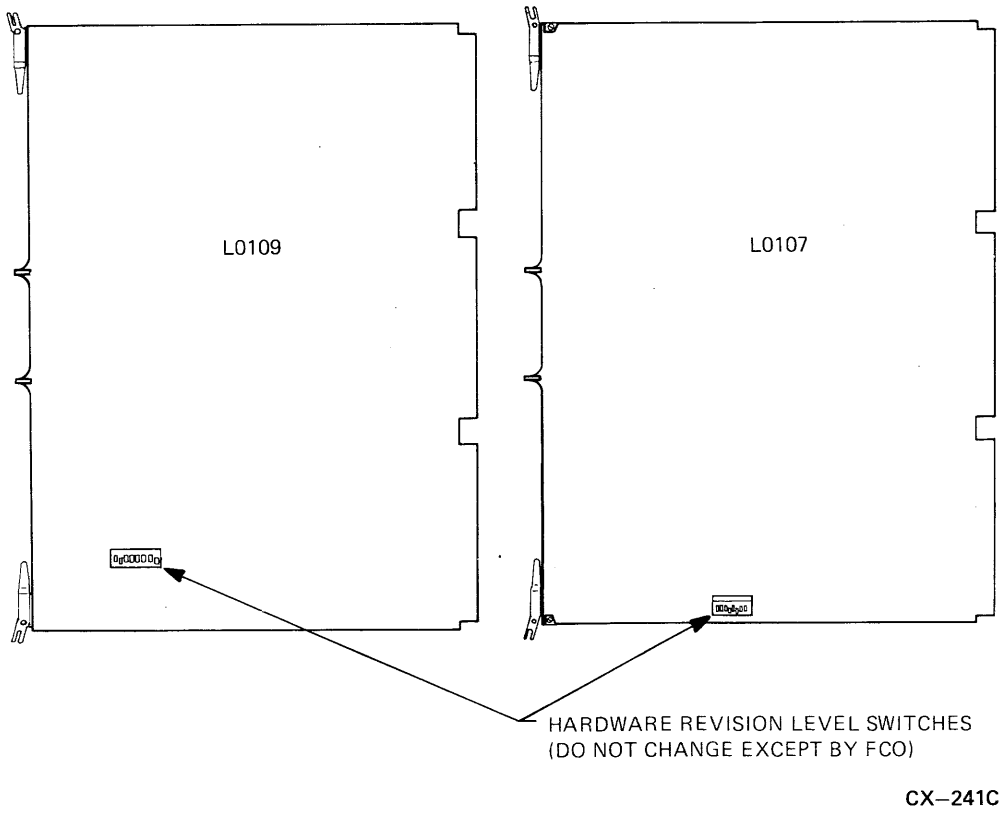


Figure 2-6 Module (DIP) Switches

- **P.ioj (L0111)** - The L0111 module contains two punch-out connector packs used to assign an unique value to the P.ioj serial number register. The switch settings should **never** be modified in the field.

The P.ioj module serial number is only used when a default HSC SCS-ID is generated. The SDS-ID is a hexadecimal number uniquely identifying the HSC as a node in the cluster. This ID is usually generated by initializing the HSC70 (toggling the Init switch on the OCP) while holding in the OCP Fault switch until the INIPIO banner is printed on the console. For all other reboot cases, the HSC70 P.ioj serial number is not used.

2.5 POWER CONTROLLER

The 881 (Figure 2-7) is a general-purpose, three-phase power controller that controls and distributes ac power to various ac devices (power supplies, fans, blower motor, etc.) packaged within an HSC70. The 881:

- Controls large amounts of ac power with low level signals.
- Provides ac power distribution to single-phase loads on a three-phase system.
- Protects data equipment from electrical noise.
- Disconnects ac power for servicing and in case of overload.

In addition, the 881 features:

- Local and remote switching
- SWITCHED receptacles only
- Convection cooling
- Rack-mounting
- AC line filtering
- DIGITAL power control bus inputs
- DIGITAL power control bus delayed output (to allow sequencing of other controllers)

2.5.1 Operating Instructions

The two basic controls on the power controller are the circuit breaker and the BUS/OFF/ON switch. These and all but one of the other controls are located on the front panel of the controller (Figure 2-7).

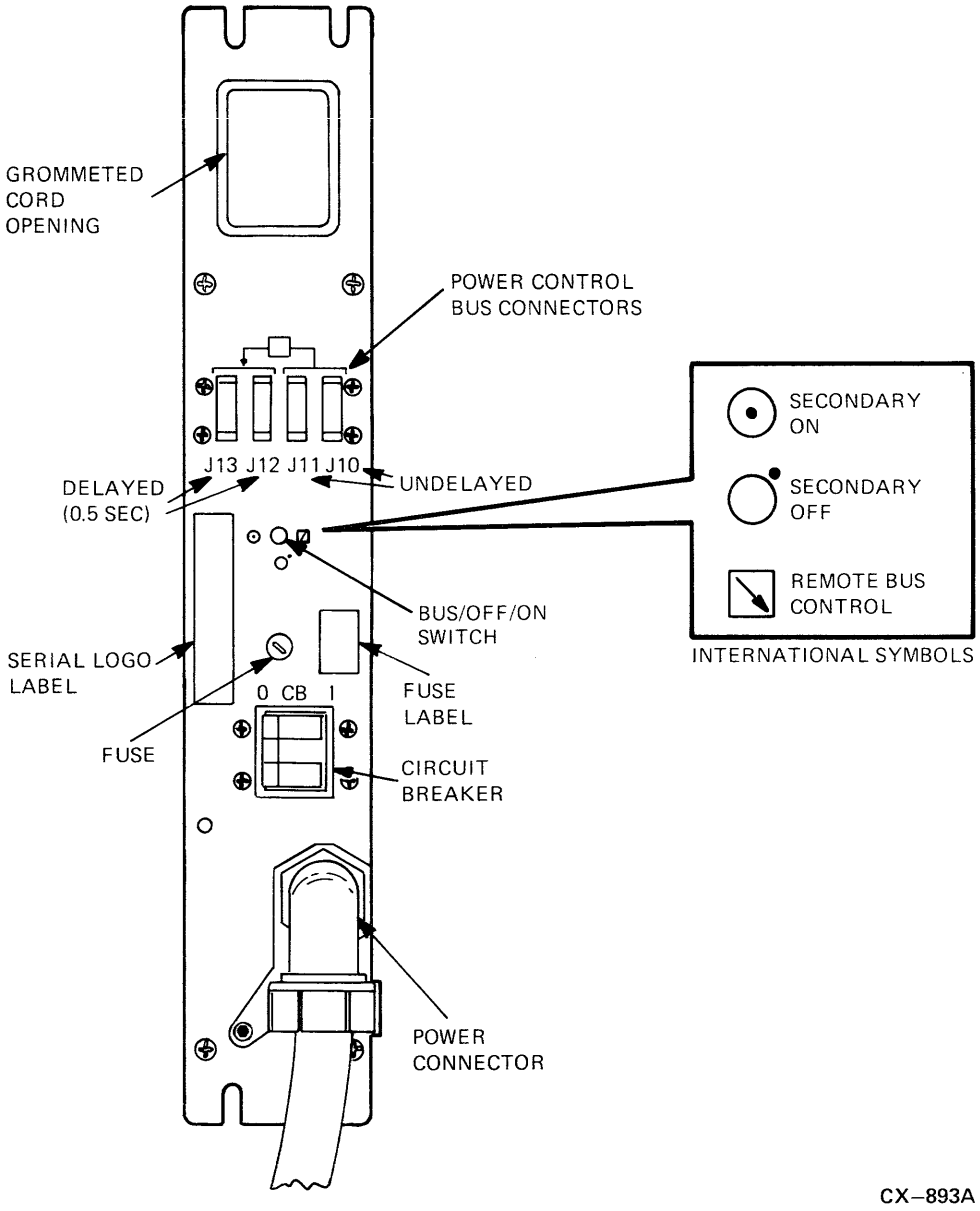


Figure 2-7 Power Controller - Front Panel Controls

The operator controls are described in the following list:

- **Power Controller Circuit Breaker** - controls the ac power to all outlets on the controller. It also provides overload protection for the ac line loads and is unaffected by switching the BUS/OFF/ON control.

- **Fuse** - protects the ac distribution system from an overload of the power control bus circuitry. The fuse is located on the front panel of the power controller.

- **DIGITAL Power Control Bus Connections** - used if control bus connections to another cabinet are required. DIGITAL power control bus MATE-N-LOK connectors are J10, J11, J12 and J13. Connectors J10 and J11 are not delayed. Connectors J12 and J13 are delayed.

- **BUS/OFF/ON Switch** - are the three positions of this switch. Assuming the circuit breaker for the power controller is ON, the ac outlets are:

  - Energized when the BUS/OFF/ON switch is in the ON position.

  - De-energized when the BUS/OFF/ON switch is in the OFF position.

NOTE

The BUS position is intended for remote sensing of DIGITAL power control bus instructions. The switch is left in the ON position when the DIGITAL power control bus is not used.

- **Total Off Connector** - is a two-pin male receptacle on the back of the power controller (Figure 2-8). It removes power from the HSC70 whenever the air flow sensor detects system air-flow loss. To reset the TOTAL OFF, cycle the circuit breaker off and then back on again.

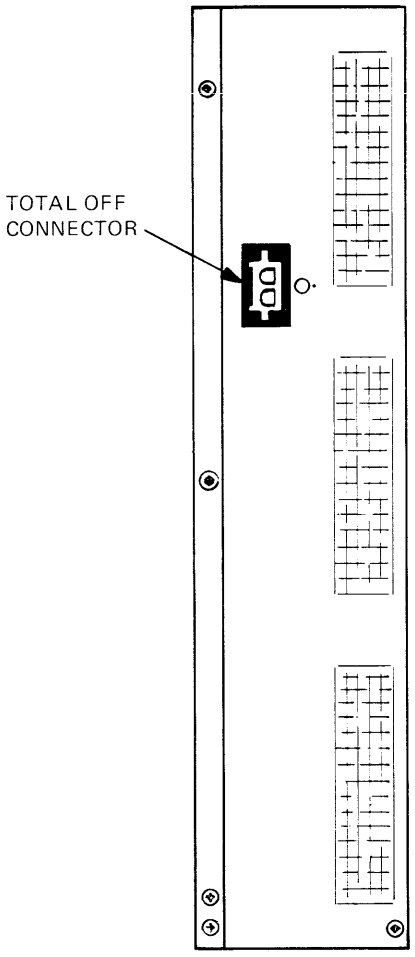


Figure 2-8 881 Rear Panel

CHAPTER 3
REMOVAL AND REPLACEMENT PROCEDURES

3.1 INTRODUCTION

This chapter describes procedures for removing and replacing the field replaceable units (FRUs) in an HSC70. Observe the following safety precautions before starting removal and replacement procedures.

3.2 SAFETY PRECAUTIONS

Because hazardous voltages exist inside the HSC70, only a qualified service representative should service the subsystem. Bodily injury or equipment damage can result from improper servicing. Always use the anti-static wrist strap provided when removing and replacing logic modules.

WARNING

Always remove power from the HSC70 before replacing internal parts or cables.

3.3 POWER REMOVAL

Before removing/replacing an FRU, turn off the ac power from the power controller CB1. Open the back door with 5/32-inch hex wrench. The power controller is located on the lower left side of the cabinet. Figure 3-1 shows the location of ac circuit breaker CB1.

To remove ac power, turn off CB1 (Figure 3-1). To ensure absolute safety, disconnect the ac plug from its receptacle.

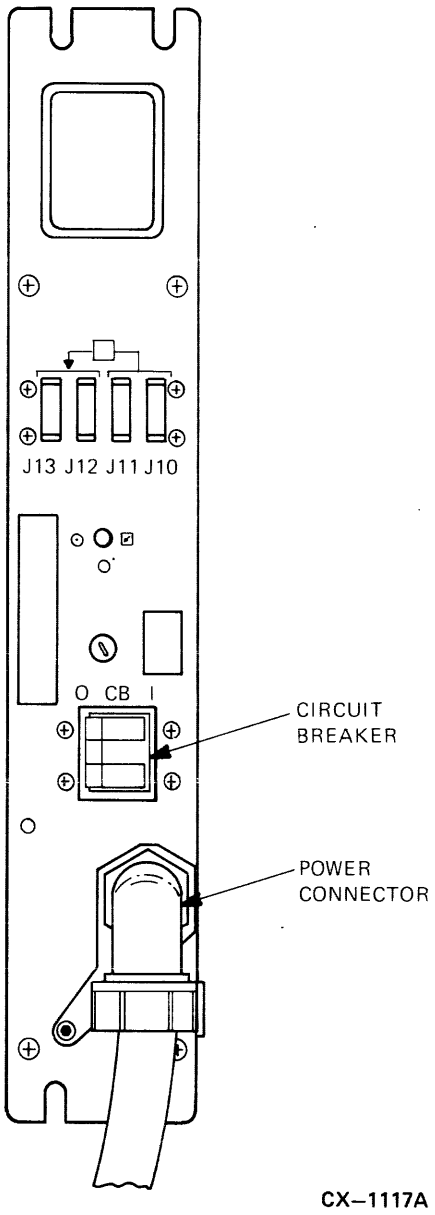


Figure 3-1 Location of Circuit Breaker on the Power Controller

Following are the two methods for removing dc power:

- Turning off the dc power switch, located on the side of the RX33 housing. See Figure 3-2.
- Turning off CB1 (ac power).

WARNING

Ensure the OCP Signal/Power line indicator is connected; otherwise the power indicator on the OCP can show power off when the power is on.

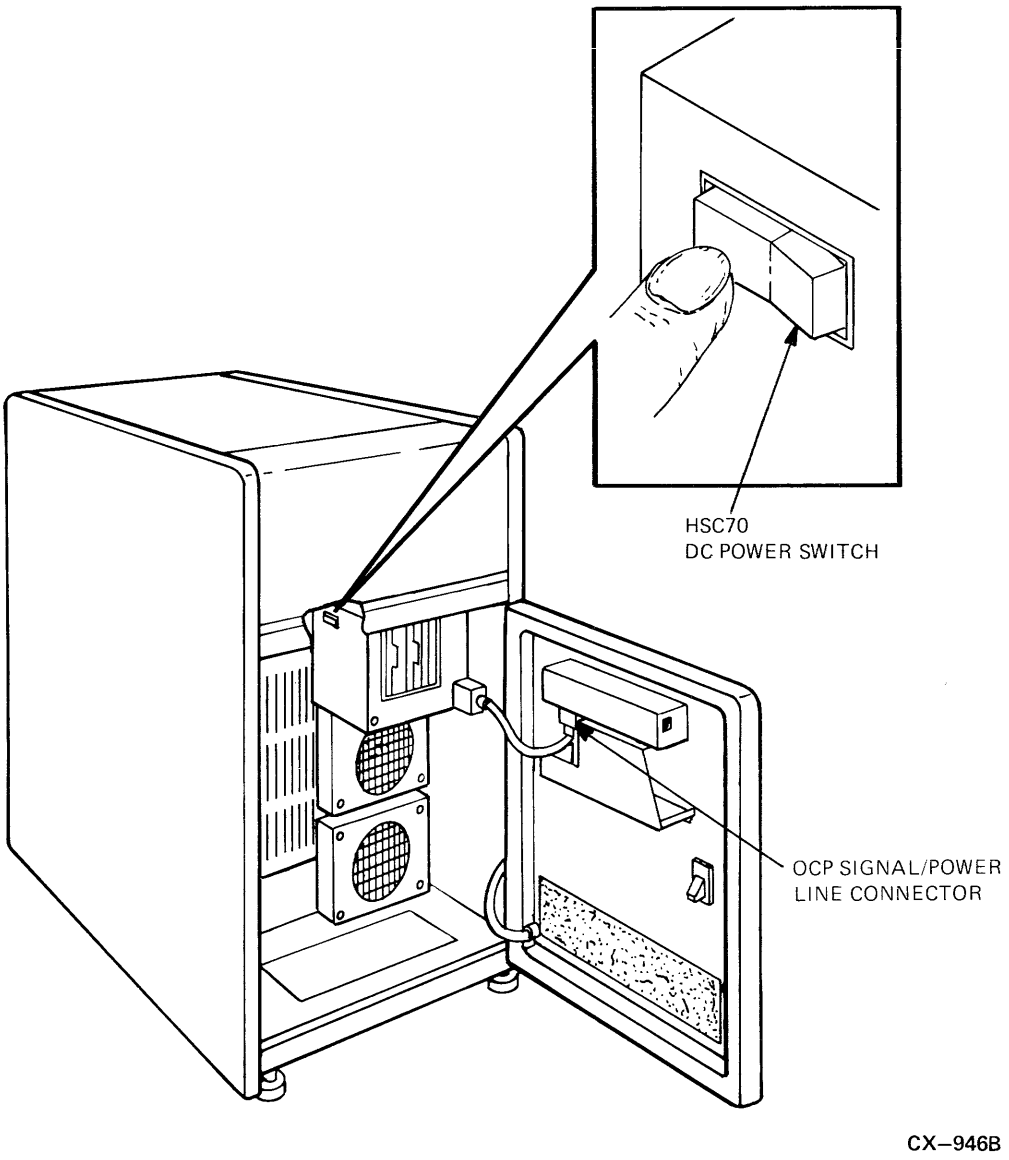


Figure 3-2 DC Power Switch Location

3.4 FIELD REPLACEABLE UNIT (FRU) REMOVAL

Figure 3-3 shows the FRU removal sequence for an HSC70.

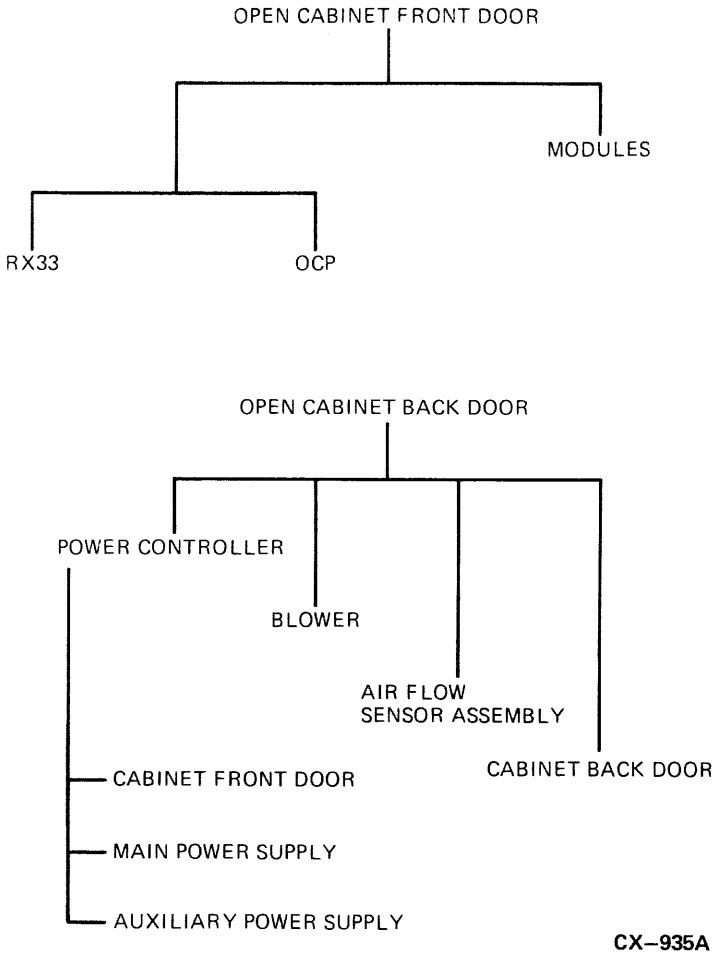


Figure 3-3 FRU Removal Sequence

3.4.1 Access From Cabinet Front Door

The FRUs accessed via the front door include the RX33, the Operator Control Panel, and the logic modules. Should you decide to remove the front door use the following procedure:

1. Unlock the cabinet front door and lift the latch to open the door.

CAUTION

When performing the following steps, take care not to damage the front spring fingers.

2. Remove HSC70 power by pushing the dc power switch to the "0" position.

3. Disconnect the ground wire from the door.

4. Disconnect the OCP cable at the bottom of the OCP shield (Figure 3-2).

5. Pull down on the spring-loaded rod on the top hinge inside the cabinet and then lift the door off its bottom pin.

Reverse the removal procedure to replace the front door.

3.4.2 Access From Cabinet Back Door

The FRUs accessed via the back door include the Power Controller, Blower, Air Flow Sensor Assembly, Main Power Supply and Auxiliary Power Supply. To remove the back door, use the following procedure.

1. Open the back door with a 5/32-inch hex wrench.

2. Pull down on the spring-loaded rod on the top hinge inside the cabinet and then lift the door off its bottom pin.

Reverse the removal procedure to replace the back door.

3.5 RX33 COVER PLATE AND DISK DRIVE REMOVAL AND REPLACEMENT

The RX33 disk drives are slide mounted in the HSC70 cabinet. A cover plate ensures proper air flow and cooling. Use the following procedure to remove the RX33 cover plate (Figure 3-4).

1. Unlock the cabinet front door and lift the latch to open the door.

2. Turn off DC Power (Figure 3-2).

3. Rotate the four fasteners on the RX33 cover plate 1/4 turn and remove the cover plate.

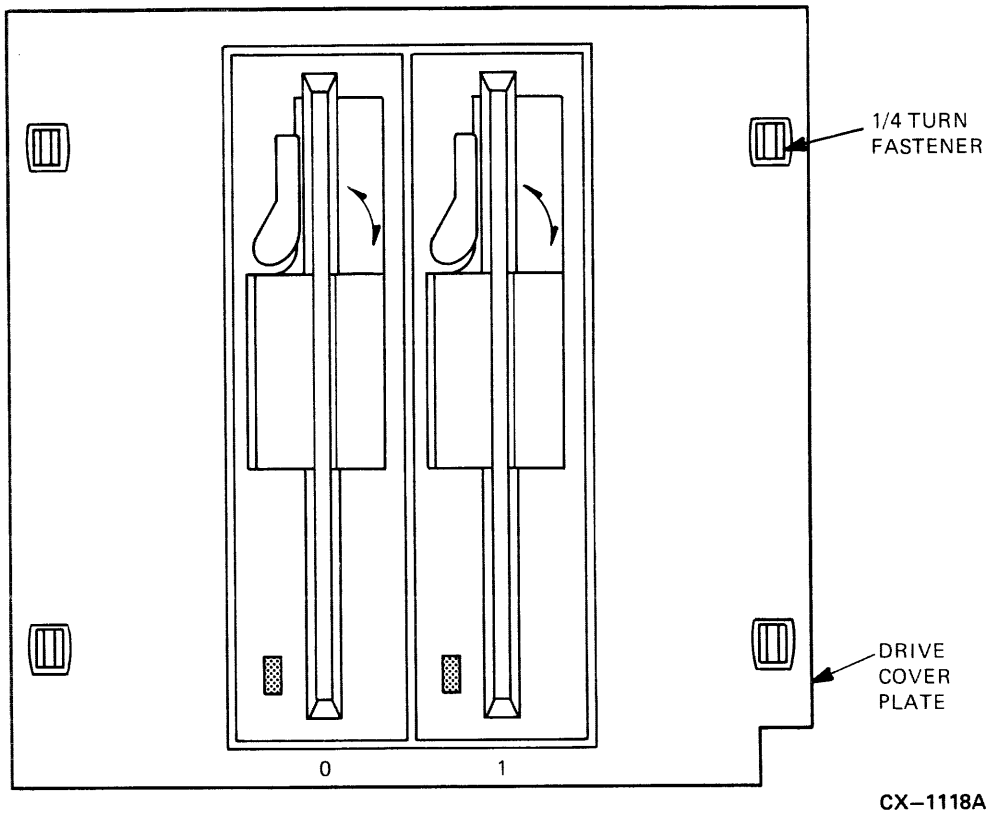


Figure 3-4 RX33 Cover Plate Removal

Use the following procedure to remove the RX33 disk drives.

1. Completely loosen the two captive screws holding the drive assembly and mounting plate to the cabinet frame.

CAUTION

Avoid snagging the cables attached to the rear of the drives during the next step.

2. Carefully pull out the slide mounted RX33s until they clear their housing.

3. Support the drives with one hand, and remove the flat ribbon cables and power cables from the rear of the drives.

4. Determine whether drive 0 or drive 1 should be replaced.

5. Loosen the captive mounting screws with a flat bladed screw driver on the drive to be replaced as shown in Figure 3-5.

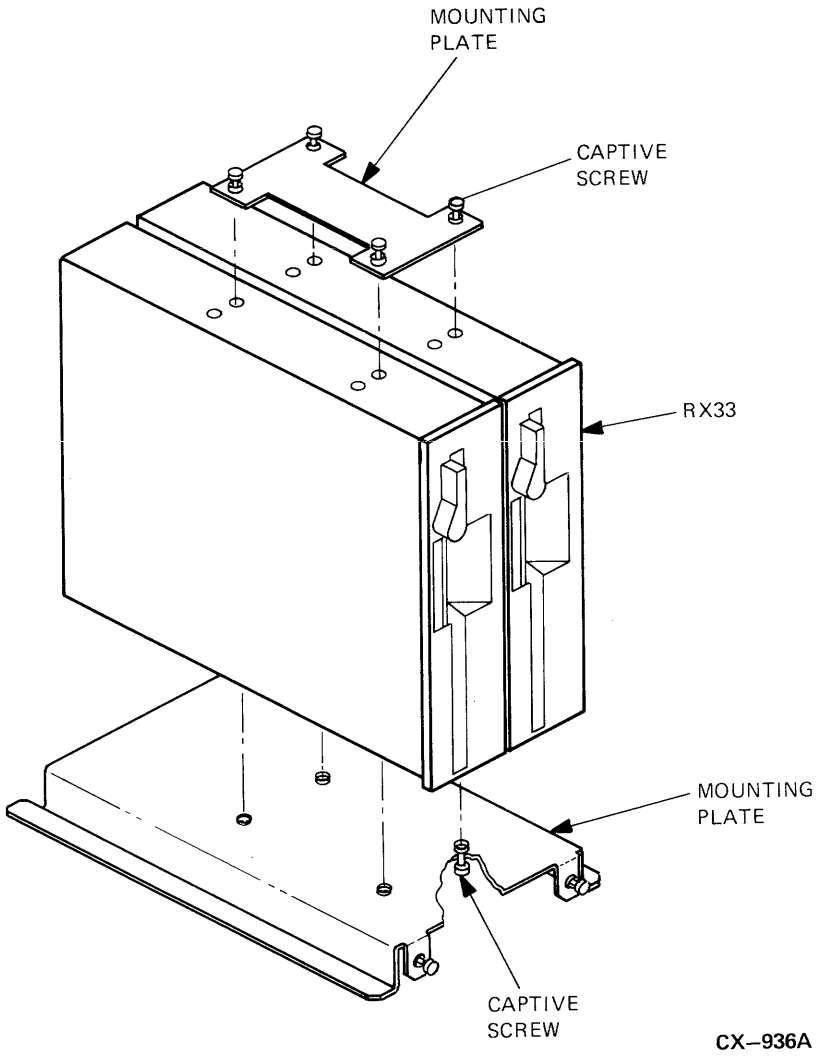


Figure 3-5 RX33 Disk Drive Removal

6. Configure RX33 jumpers on the replacement drive as shown in Figure 3-6. If replacing drive 0, be sure to insert jumper DS0. If replacing drive 1, be sure to insert jumper DS1. Section 3.5.1 briefly describes the function of each jumper.

NOTE

Replacement RX33 drives shipped from the vendor are not configured for HSC70 application. Two identical jumpers, DEC part number 12-18783-00, must be added. If no extra jumpers are available, remove two jumpers from the defective drive. Correct jumper configuration is necessary for proper operation of the replacement RX33 drive (see next section).

7. Replace the defective drive with a new one.

Reverse the removal procedure to replace the RX33 drives.

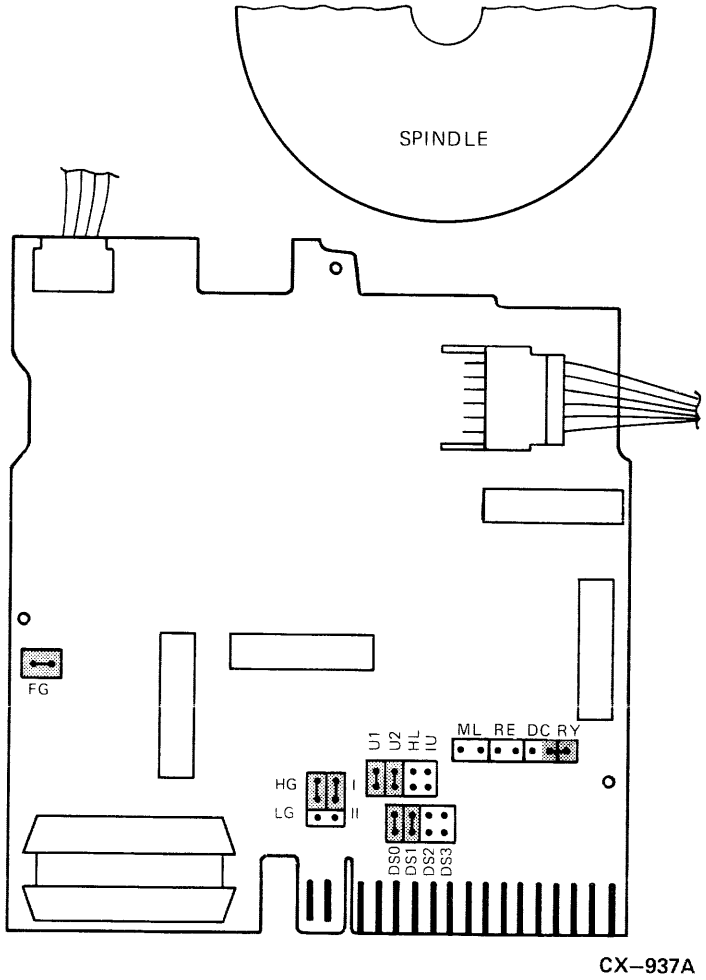


Figure 3-6 RX33 Jumper Configurations

3.5.1 RX33 Jumper Configuration

This section defines the RX33 jumpers. Jumpers identified with an asterisk are connected for HSC operation.

- * FG = Frame ground connection
- LG = Logic low on NORMAL/HI DENSITY signal enables high-density mode
- * HG = Logic high on NORMAL/HI DENSITY signal enables high-density mode
- * DS0, 1, 2, 3 = Drive select number 0, 1, 2, 3
- * I = Speed Mode I (dual speed mode)
- II = Speed Mode II (single speed mode, 360 RPM only)
- * U1, U2 = Selects mode of operation for loading heads and lighting bezel LED (see note).
- HL, IU = Selects mode of operation for loading heads and lighting bezel LED (see note).
- DC = Drive will assert DISK CHANGED signal on pin 34 of interface cable
- * RY = Drive will assert DRIVE READY signal on pin 34 of interface cable
- ML = Motor enable. No jumper installed for HSC70 application
- RE = Recalibration. No jumper installed for HSC70 application

NOTE

The HSC70 loads heads and lights the drive-in-use LED when DRIVE SELECT n and READY are both true.

3.6 OPERATOR CONTROL PANEL (OCP) REMOVAL AND REPLACEMENT

If any OCP lamp fails, replace the entire OCP as follows:

1. Open the front door by turning the key clockwise and lifting the latch.
2. Remove dc power (Figure 3-2).
3. Remove the four Kepnuts securing the OCP shield to the studs on the front door.

4. Remove the OCP shield.

5. Remove the two screws securing the OCP to the shield (Figure 3-7).

6. Remove the two connectors from the printed circuit board on the OCP.

7. Pull out the OCP carefully allowing for indicator and switch clearance.

Reverse the removal procedure to replace the OCP.

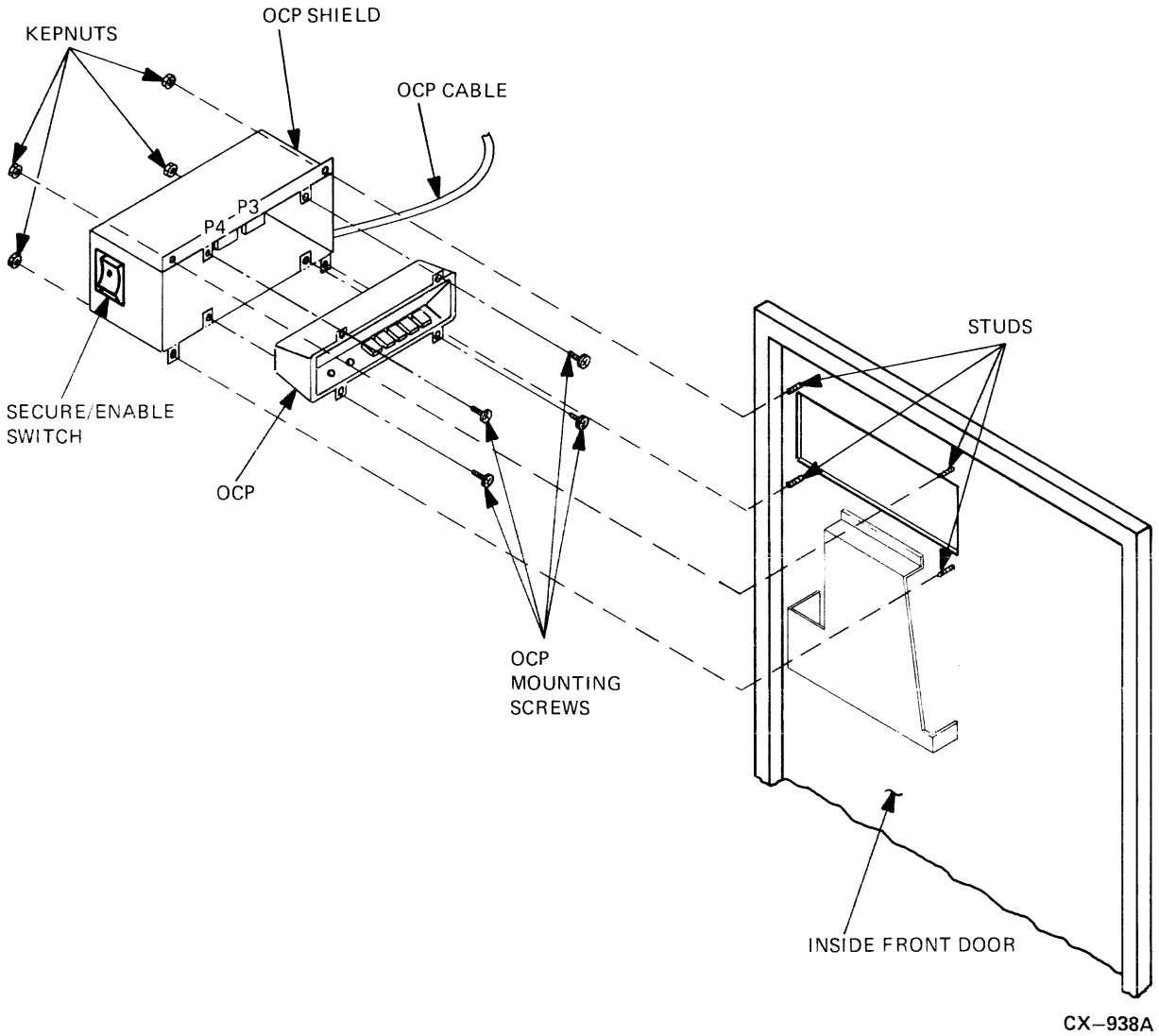


Figure 3-7 Operator Control Panel Removal

3.7 LOGIC MODULES REMOVAL AND REPLACEMENT

A Velostat (antistatic) kit must be used during module removal/replacement. The Velostat kit part number is 29-11762. For convenience, an antistatic wrist strap is included in the front door diskette storage area.

1. Open the front door by turning the key clockwise.
2. Push the DC Power switch to the 0 position (off). See Figure 3-2.
3. Turn the two nylon latches on the module cover plate one-quarter turn (Figure 3-8).
4. Pull the card cage cover up and out.
5. Check the module utilization label above the card cage for the location of the desired module. The module slots are numbered from right to left when viewed from the front.
6. Remove the module and replace with a new one.

   To remove the L0100 port link module, the door latch plate attached to the left side of the cabinet frame must be moved away from the module removal path. In production model HSC70s, the latch plate is swivel mounted. Lift the plate slightly and press it flat against the cabinet frame. Before closing the cabinet door, return the door latch plate to its locked position.

Reverse the removal procedure to replace the card cage cover.

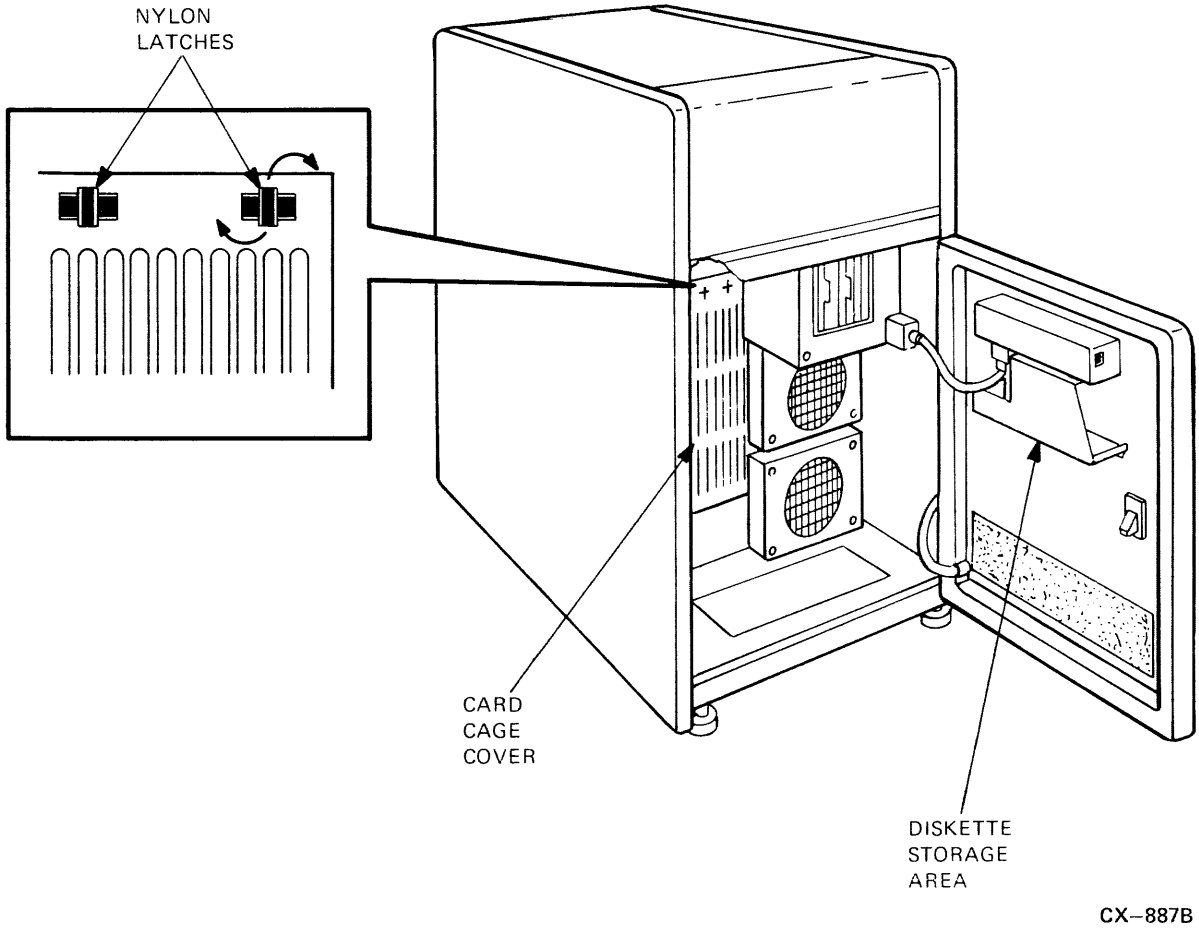


Figure 3-8 Card Cage Cover Removal

NOTE

The I/O control processor module is identified by factory-set jumpers. Each module has a unique serial number that matches the pattern of the jumpers. Do not reconfigure these jumpers.

If the port link module is being replaced, ensure the node address switches are properly set on the new module. Figure 3-9 shows the location of the switches. See the system manager for the correct node address.

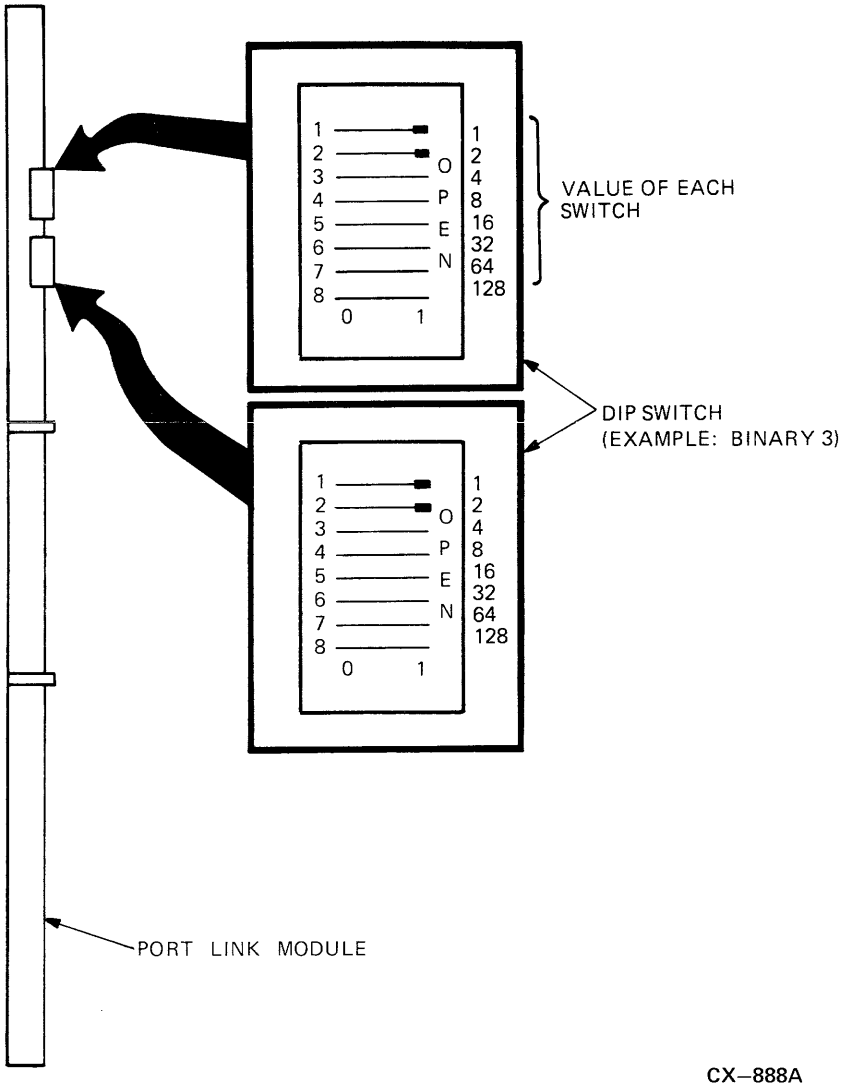


Figure 3-9 Location of Node Address Switches

3.8 BLOWER REMOVAL AND REPLACEMENT

The blower, which provides forced air cooling for the cabinet, is removed by using the following procedure:

1. Open the back door using a 5/32-inch hex wrench.
2. Turn off ac power (CBl on the power controller).
3. Disconnect the blower power connector.
4. Remove the exhaust duct from the bottom of the blower by lifting up the quick release latches on each side of the duct (Figure 3-10).

5. Disconnect the airflow sensor power connector (J70) to allow removal of the exhaust duct.

NOTE

Figure 3-10, Figure 3-11, and Figure 3-12 show the blower outlet duct for current HSC70s. Early models have a smaller blower motor outlet duct.

6. Loosen, but do not remove, the three Phillips screws holding the blower mounting bracket to the cabinet.

7. Lift the blower and bracket up and out of the cabinet.

Reverse the removal procedure to replace the cooling blower.

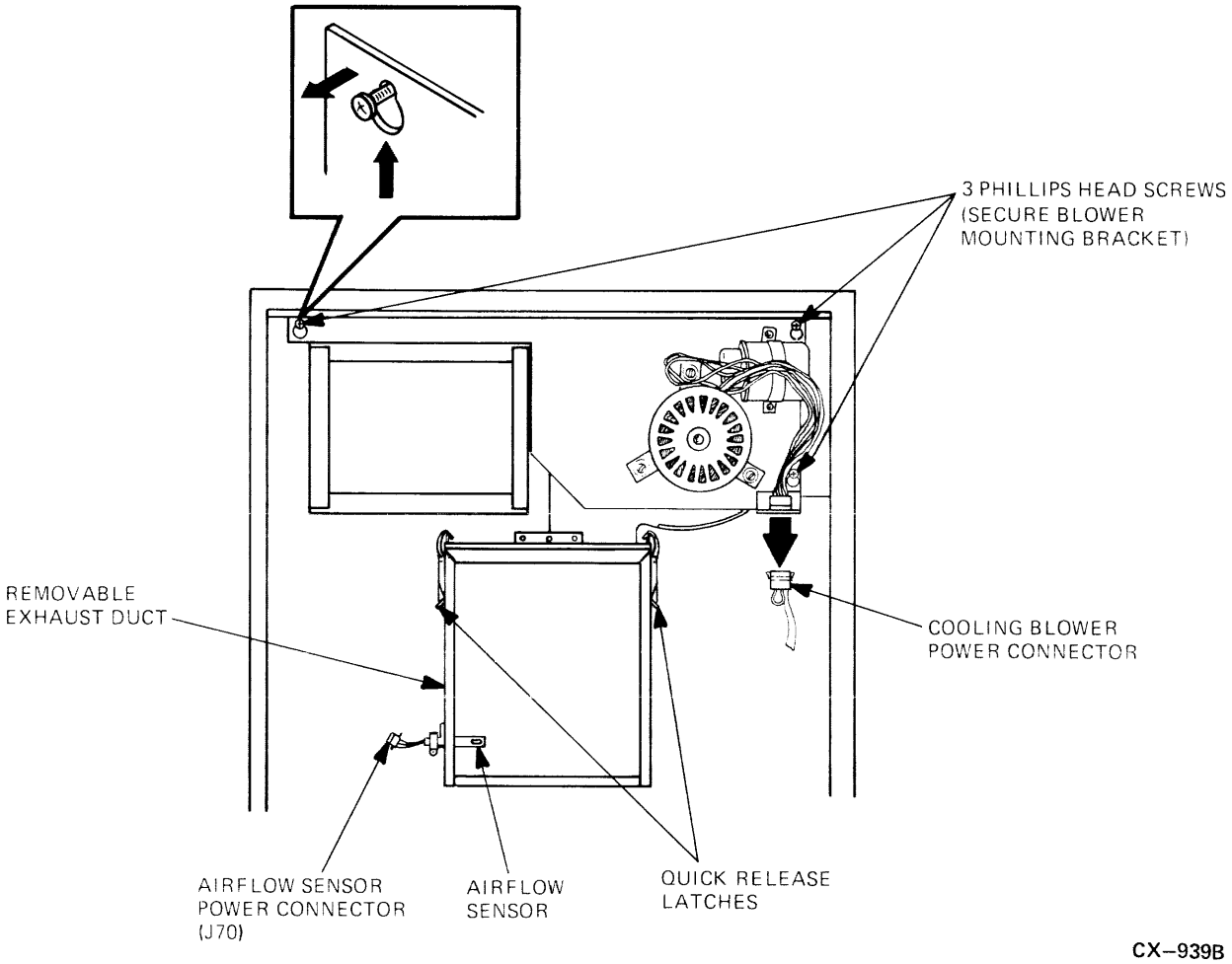


Figure 3-10 Main Cooling Blower Removal

3.9 AIRFLOW SENSOR ASSEMBLY REMOVAL AND REPLACEMENT

The airflow sensor assembly, housed in the cooling duct, is removed by the following procedure:

1. Open the back door using a hex wrench.
2. Turn off the ac circuit breaker (CB1) on the HSC70 power controller.
3. Disconnect J70 (Figure 3-11).
4. Remove Phillips head screw that holds mounting clamp to the duct.
5. Slide sensor assembly out of duct.

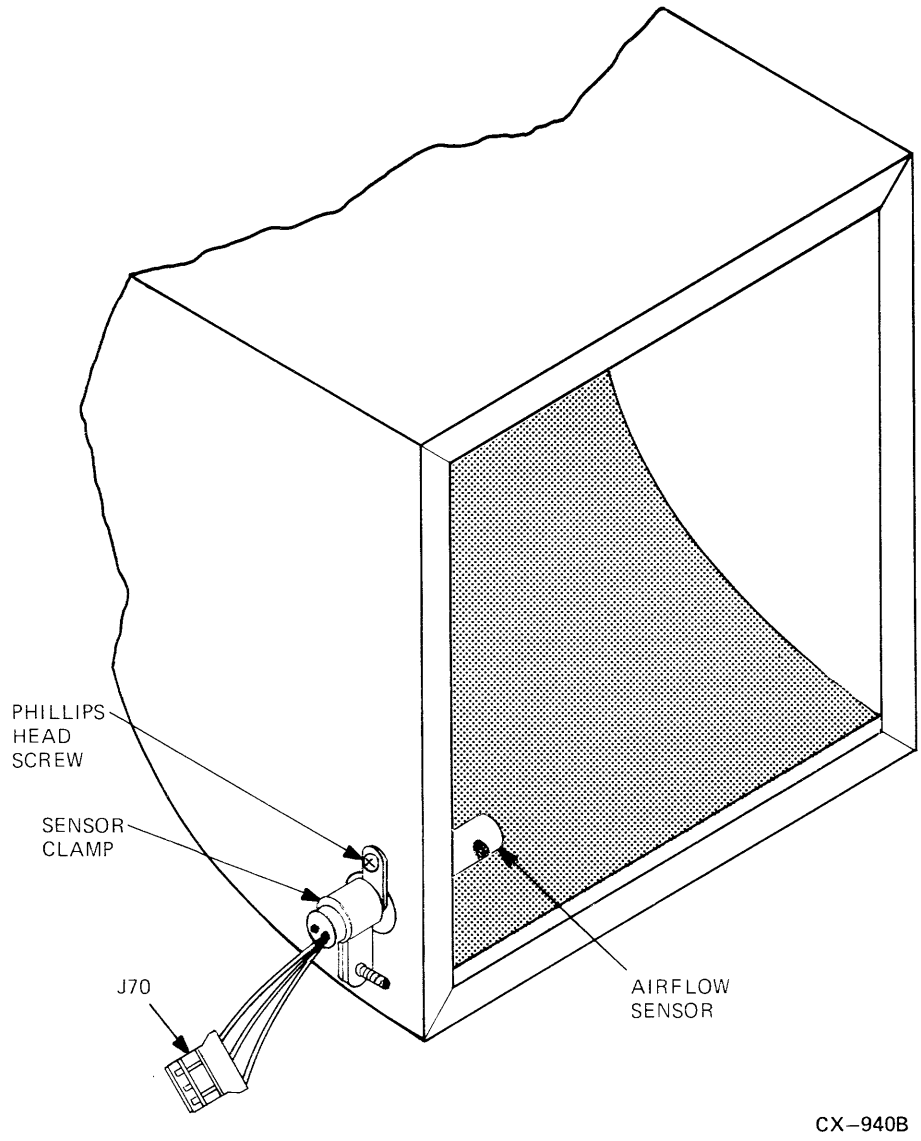


Figure 3-11 Airflow Sensor Assembly Removal

Reverse the removal procedure to replace the airflow sensor assembly and follow these three steps:

1. Align the slots in the airflow sensor tip horizontally with the floor.

2. After turning on ac power to the HSC70, test the new airflow sensor for proper operation.

3. Ensure the sensor is operable by blocking the flow of air. Pinching the sensor should trip CB1.

3.10 POWER CONTROLLER REMOVAL AND REPLACEMENT

The power controller must be removed to replace either of the power supplies.

1. Open the back door.

2. Remove rear door latch to allow clearance for power controller removal.

3. Remove ac power by placing CB1 in the off position (Figure 3-1).

4. Unplug the power controller from the power source.

5. Remove the two top screws and then the two bottom screws securing the power controller to the cabinet (Figure 3-12). While removing the two bottom screws, push up on the power controller to take the weight off the screws.

CAUTION

Do not pull the power controller out too far because cables are connected to the back and top.

6. Pull the power controller towards you and then out.

7. Remove the power control bus cables from J10, J11, J12, and J13 connectors at the front of the power controller. Refer to Figure 3-12.

8. Disconnect the total off connector at the rear of the power controller.

9. Disconnect all line cords from the top of the power controller.

NOTE

Be sure to rotate the line cord elbow to the vertical position if replacing a defective power controller with a new one. To rotate the elbow remove the set screw, rotate the elbow to the position shown in Figure 3-12 and replace the set screw in the other hole.

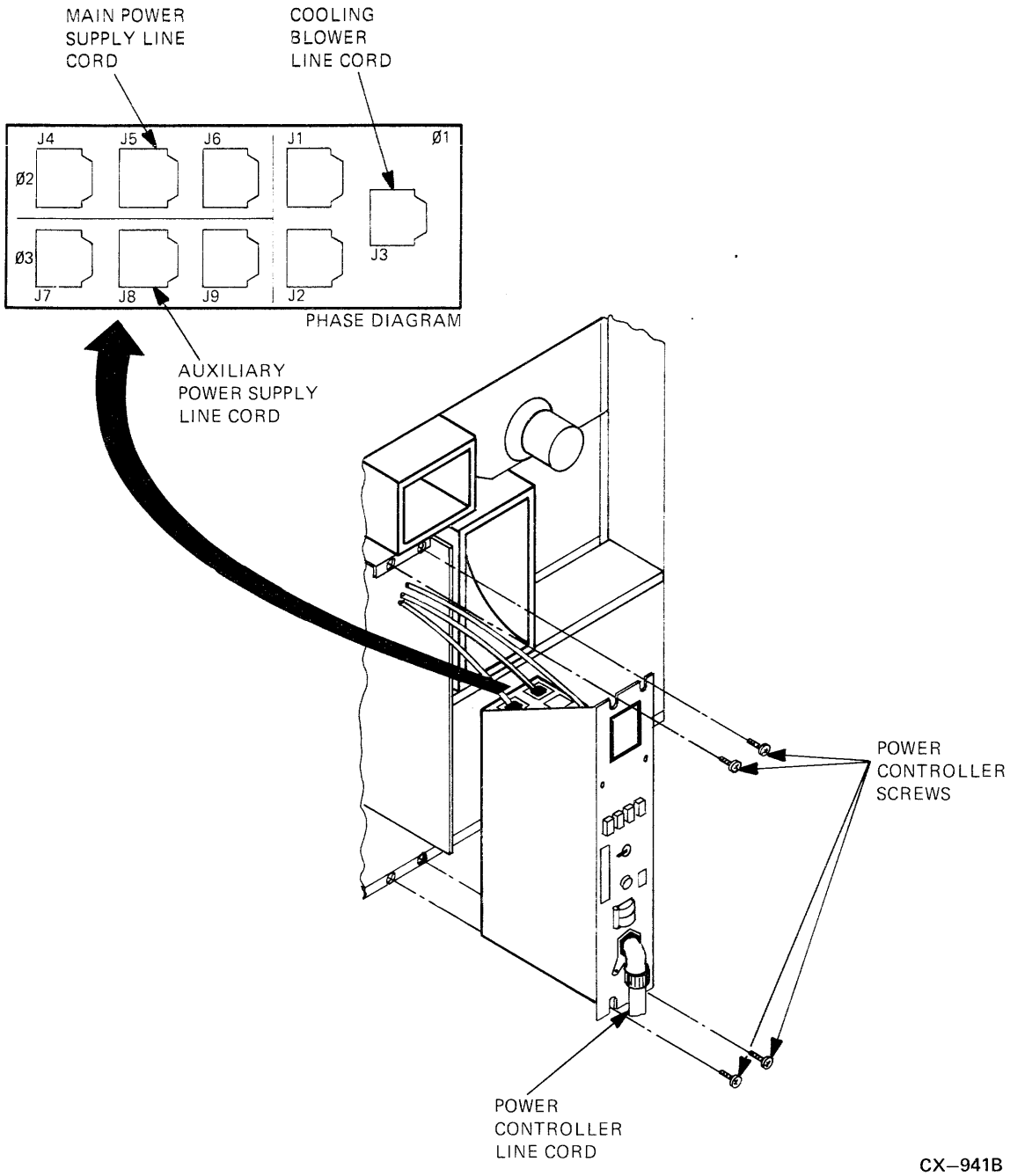


Figure 3-12 Power Controller Removal

Reverse the removal procedure to replace the power controller.

NOTE

To ensure proper phase distribution, reconnect the main power supply, auxiliary power supply and cooling blower line cords as shown in Figure 3-12.

3.11 MAIN POWER SUPPLY REMOVAL AND REPLACEMENT

The following procedure covers the removal of the main power supply:

WARNING

The power supply is heavy. Support it with both hands to prevent dropping it.

1. Open the back door using a 5/32-inch hex wrench.
2. Turn off CB1 (ac power) on the power controller.
3. Unplug the power controller from the power source.
4. Remove the front door.
5. Remove the power controller (Section 3.10) to access the back of the power supply.
6. Unplug the main power supply line cord at the power controller.
7. Remove the nut from the -V1 stud (ground) on the back of the power supply (Figure 3-13).
8. Remove the nut from the +V1 stud (+5 volts) on the back of the power supply.
9. Remove the nut from the -V2 (-5.2 volt) stud on the back of the power supply.
10. Remove the nut from the +V2 (ground) stud.
11. Unplug J31 (+12 Vdc output from the supply to backplane).
12. Unplug P32 (+12 VDC and +5 Vdc sense lines).

WIRE LIST					
COLOR	POSITION	SIGNAL	COLOR	POSITION	SIGNAL
PUR	TBI-3-5	12 V	PUR	TBI-3-1	12 V SENSE
PUR	TBI-3-6	12 V	BLU	TBI-2-7	ACC
BLK	TBI-3-3	GND (12 V)	BRN	TBI-2-6	AC
BLK			GRN/YEL	TBI-2-5	GND
ORN	TBI-2-2	−5 V SENSE	YEL	TBI-2-3	ON/OFF (−5, 3 V)
BLK	TBI-2-1	GND (−5 V SENSE)	ORN	TBI-2-2	−5 V SENSE (S2−)
BRN	TBI-1-4	POWER FAIL	BLU	TBI-1-3	ON/OFF 5 V
BLK	TBI-1-2	GND (5 V SENSE)	BLK	TBI-1-2	GND (5 V SENSE)
RED	TBI-1-1	5 V SENSE	PUR	TBI-3-2	12 V
BLK	TBI-3-4	GND (12 V SENSE)			

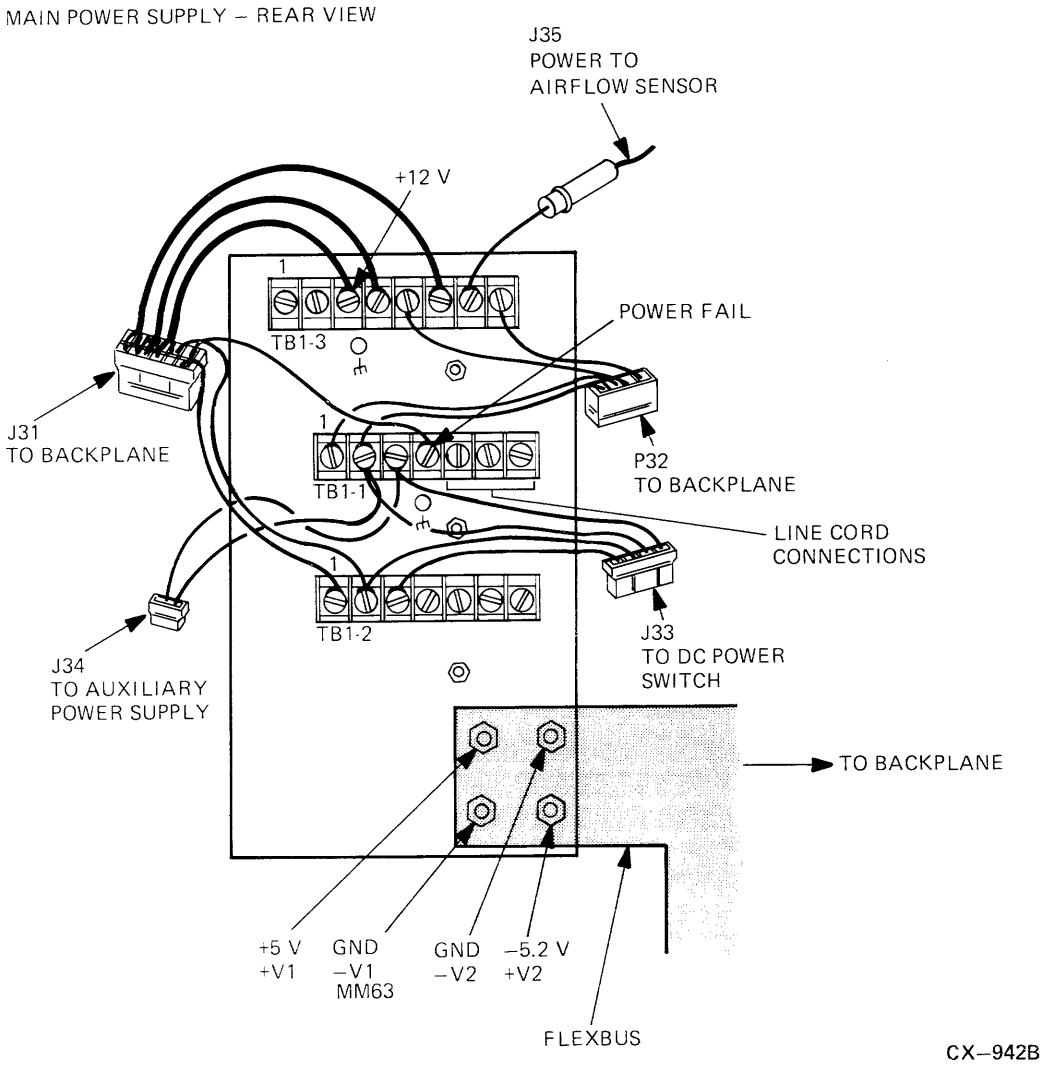


Figure 3-13 Main Power Supply Cables - Disconnection

13. Unplug J33 (to DC power switch).

14. Unplug J34 (remote on/off jumper to auxiliary power supply).

15. Unplug J35 (+12 Vdc power to the airflow sensor).

16. Turn the four captive screws on the front of the power supply counterclockwise (Figure 3-14).

17. Pull the power supply out about an inch. Check the back of the cabinet to ensure the cables and flexbus connectors are clear and will not snag when the supply is completely removed.

18. Carefully pull the power supply all the way out of the cabinet.

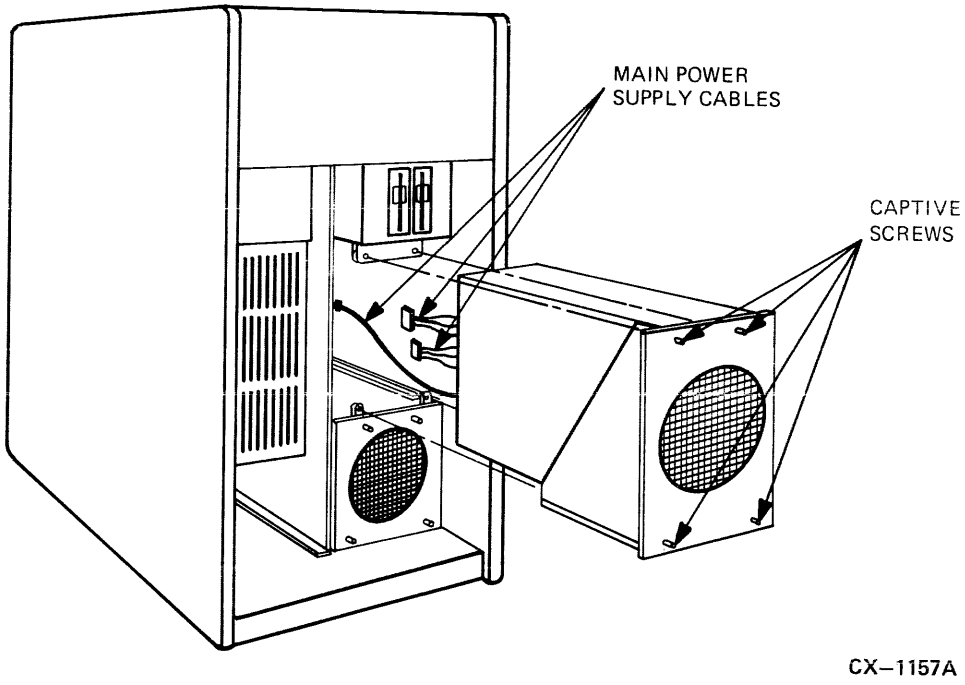


Figure 3-14 Main Power Supply Removal

19. Remove the power cord from the failing unit and install it on the new power supply.

NOTE

Spare power supplies are not shipped with a power cord.

Reverse the removal procedure to replace the main power supply.

3.12 AUXILIARY POWER SUPPLY

An HSC70 requires an auxiliary power supply. The auxiliary power supply is mounted directly beneath the main power supply. The procedure for mounting the auxiliary power supply follows:

WARNING

This power supply is heavy. When removing, support it with both hands to prevent dropping it.

1. Open the back door using a 5/32-inch hex wrench.
2. Turn off CB1 (ac power) on the power controller.
3. Unplug the power controller from the power source.
4. Remove the front door.
5. Remove the power controller to access the back of the power supply (Section 3.10).
6. Unplug the auxiliary power supply line cord at the power controller.
7. Remove the nut from the +V1 stud (+5 volt) on the back of the power supply (Figure 3-15).
8. Remove the nut from the -V1 stud (ground) on the back of the power supply.
9. Disconnect J50 (sense line to voltage comparator).
10. Disconnect J51 (dc on/off jumper).
11. Turn the four captive screws on the power supply counterclockwise (Figure 3-16).

WIRE LIST		
COLOR	POSITION	SIGNAL
BLACK	TBI-2	GROUND (5 V SENSE)
RED	TBI-1	5 V SENSE
BROWN	TBI-4	POWER FAIL
BLUE	TBI-7	ACC
BROWN	TBI-6	AC
GRN/YEL	TBI-5	CHASSIS GROUND
BLUE	TBI-3	ON/OFF
BLACK	TBI-2	GROUND (5 V SENSE)

AUXILIARY POWER SUPPLY – REAR VIEW

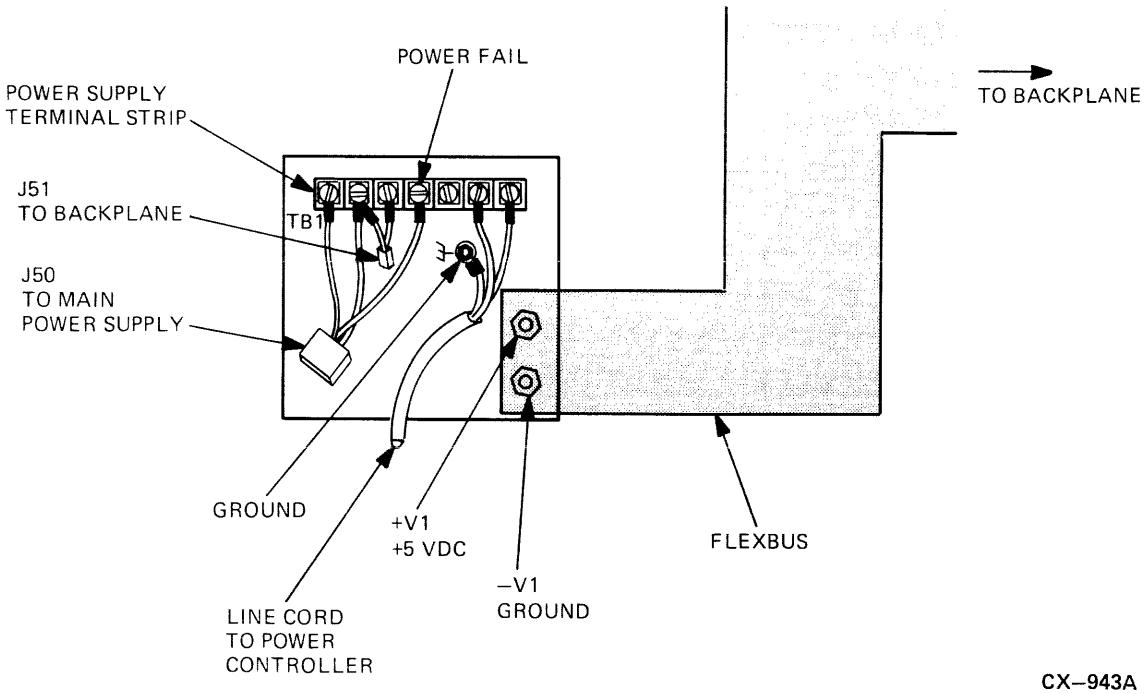


Figure 3-15 Auxiliary Power Supply Cable Disconnection

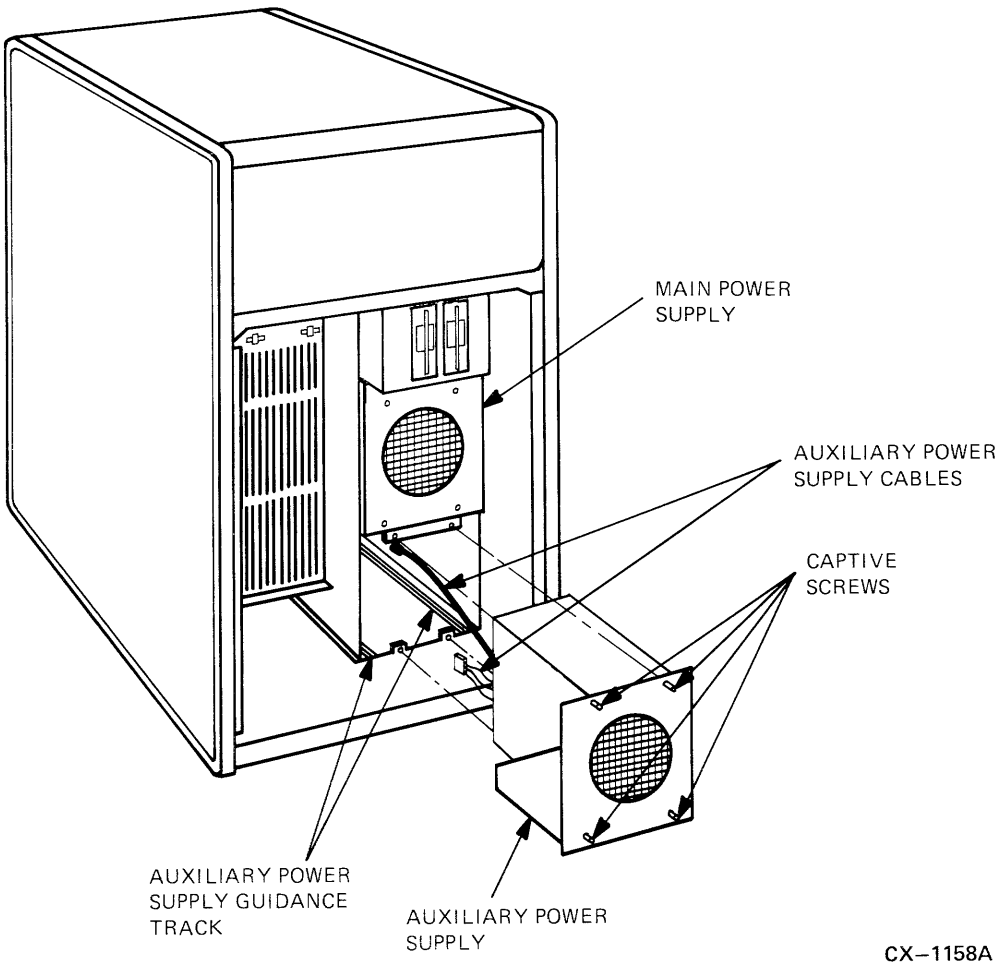


Figure 3-16 Auxiliary Power Supply Removal

12. Pull the power supply out about an inch. Check the back of the cabinet to ensure the cables and flexbus connectors are clear.

13. Carefully slide the power supply out through the front of the HSC70.

14. Remove the power cord from the failing unit and install on the new power supply.

NOTE

Spare supplies are not shipped with a power cord.

Reverse the removal procedure to replace the auxiliary power supply.

CHAPTER 4
INITIALIZATION PROCEDURES

4.1 INTRODUCTION

This chapter tells how to connect the console terminal and how to initialize the HSC70. Error reporting by fault codes displayed on the OCP is also described.

4.2 CONSOLE TERMINAL CONNECTION

The console terminal designated for the HSC70 is the VT220. An LA50 printer is connected to the terminal for hardcopy output. Detailed operating information is provided in the owner manuals accompanying the VT220 and LA50.

Figure 4-1 shows the placement of the EIA terminal connectors on the HSC70 rear bulkhead. The console terminal connects to the J60 connector as shown. Although three EIA connectors are shown, two terminals cannot simultaneously connect to an HSC70.

Preferably, power is turned off before the console terminal is installed. If power must be left on while connecting the terminal, use the following procedure:

- Put the Secure/Enable switch in the SECURE position.
- Change terminal state (plug in, remove power, connect EIA line)
- Type three space characters on the terminal keyboard.
- Put the Secure/Enable switch in the ENABLE position if it is necessary to do so at this point.

NOTE

If this procedure is not followed, the HSC70 may enter micro-Online Debugging tool (ODT) mode. An @ symbol on the screen indicates this mode. Typing a P (proceed) exits this mode.

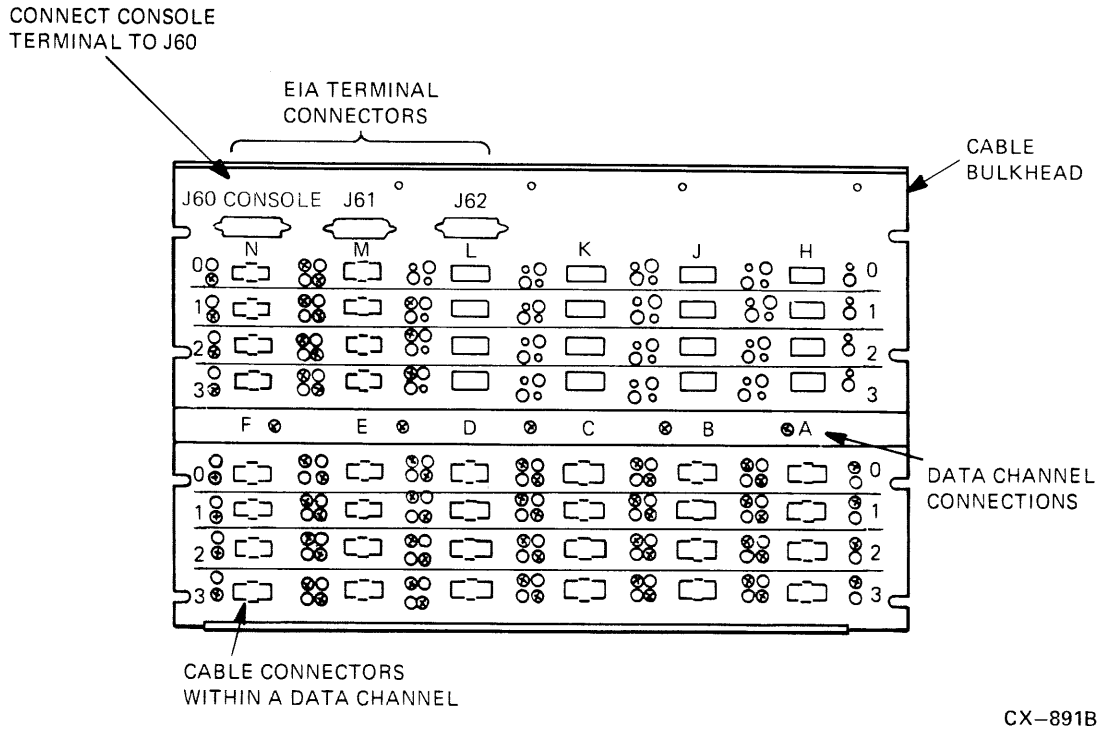


Figure 4-1 Console Terminal Connection

4.3 HSC70 INITIALIZATION

This section describes the booting procedures for the HSC70 System diskette. This diskette also contains the software necessary to execute the inline diagnostics and the utilities. To boot and run the offline diagnostics from a separate Offline diskette, refer to Chapter 6.

NOTE

Blank RX33 diskettes are unformatted. The format procedure is described in Chapter 7.

In order to run the HSC70 inline, the System diskette must reside in the RX33 drive. Customarily, this diskette resides in RX33 drive 0. However, drive 1 and drive 0 are identical, and disk placement is arbitrary.

System boot is initiated by either powering on the unit or (if the unit is already on) by depressing and releasing the Init switch with the Secure/Enable switch in the ENABLE position. This initiates the P.io ROM bootstrap tests and then loads the Init P.io Test.

4.3.1 Init P.io Test

The Init P.io Test completes the P.ioj module and the HSC70 memory testing previously started by the ROM bootstrap tests. All P.ioj logic not tested by the bootstrap is completed. In addition, the HSC70 Program, Control, and Data memories are tested.

This test runs in a stand-alone environment (no other HSC70 processes are running). If a failure is detected, the failing module is flagged. If the test runs without finding any errors, the HSC70 operational software is loaded and started. The Init P.io Test is not a repair level diagnostic. If a repair level test is needed, run the Offline P.io test that provides standard HSC70 error messages.

4.3.1.1 Init P.io Test System Requirements - In order to run this test, the following hardware is required:

- o P.ioj (processor) module with HSC70 Boot ROM
- o At least one M.std2 (memory) module
- o RX33 controller with at least one working drive

In addition, an HSC70 System diskette (RX33 media) is required.

4.3.1.2 Init P.io Test Prerequisites - The Init P.io Test is loaded by the HSC70 ROM Bootstrap program. The bootstrap tests the basic J-11 instruction set, the lower 2048 bytes of Program Memory, an 8 Kword partition in Program memory, and the RX33 subsystem used by the bootstrap. When the Init P.io Test begins to execute, most J-11 logic has been tested and is considered working. Likewise, the Program memory occupied by the test and the RX33 subsystem used to load the test are also considered tested and working. The RX33 diskette is checked to ensure it contains a bootable image.

4.3.1.3 Init P.io Test Operation - Follow these steps to start the Init P.io Test:

1. Insert the HSC70 System diskette in the RX33 unit 0 drive (left-hand drive).

2. Power on the HSC70, or depress and release the Init button on the HSC70 OCP with the Secure/Enable switch enabled. The Init lamp should light and the following should occur:

   - The RX33 drive-in-use LED should light within 10 seconds indicating the bootstrap is loading the Init P.io Test to the Program memory.
   - The I/O State light is on after diskette motion stops and the Init P.io Test begins testing.
   - The Init P.io Test displays the following message on the HSC console when it begins: INIPIO-I BOOTING.
   - HSC70 operational software indicates it has loaded properly when the State light blinks.
   - HSC70 displays its name and version indicating it is ready to perform host I/O.

Once initiated, the Init P.io Test is only terminated by halting and rebooting the HSC. If the test fails to load using the preceding start-up procedure, perform the next three steps.

1. Boot the diskette from the RX33 unit 1 drive (right-hand drive).
2. Boot another diskette. If that diskette boots, the original diskette is probably damaged or worn.
3. Boot the HSC70 Offline Diagnostic diskette. This diskette contains the Offline P.io Test, which provides extensive error reporting features. A console terminal must be connected to run the offline tests.

The progress of the Init P.io Test is displayed in the State LED. Before the test starts, the State LED is off. When the test starts, the State LED is turned on, and the INIPIO-I BOOTING message is printed on the HSC console. When the test completes with no fatal errors, the State LED begins to blink on and off. If the test detects an error, the Fault lamp on the HSC70 OCP is lit.

4.3.2 Fault Code Interpretation

All failures occurring during the Init P.io test are reported on the operator control panel LEDs. When the Fault lamp is lit, pressing the Fault switch results in the display of a failure code in the OCP LEDs. This code indicates which HSC70 module is the most probable cause of the detected failure. The failure

code blinks on and off at 1-second intervals until the HSC is rebooted if the fault code represents a fatal fault. A soft fault code is cleared in the OCP by depressing the fault switch a second time. To restart the boot procedure, press the Init switch. This procedure is detailed in Chapter 8. To identify the probable failing module, refer to Figure 4-2.

DESCRIPTION	HEX	OCT	BINARY	OCP INDICATORS				
				INIT	FAULT	ONLINE		
PORT PROCESSOR MODULE FAILURE†	01	01	0 0001					
DISK DATA CHANNEL MODULE FAILURE†	02	02	0 0010					
TAPE DATA CHANNEL MODULE FAILURE†	03	03	0 0011					
INSTRUCTION CACHE PROBLEM IN I/O CONTROL PROCESSOR*	08	10	0 1000					
HOST INTERFACE ERROR*	09	11	0 1001					
DATA CHANNEL ERROR*	0A	12	0 1010					
I/O CONTROL PROCESSOR MODULE FAILURE	11	21	1 0001					
MEMORY MODULE FAILURE	12	22	1 0010					
BOOT DEVICE FAILURE**	13	23	1 0011					
PORT LINK MODULE FAILURE	15	25	1 0101					
MISSING FILES REQUIRED	16	26	1 0110					
NO WORKING K.SDI, K.STI, OR K.CI	18	30	1 1000					
REBOOT DURING BOOT	19	31	1 1001					
SOFTWARE DETECTED INCONSISTENCY	1A	32	1 1010					

† INCORRECT VERSION OF MICROCODE.

* THESE ARE THE SO-CALLED SOFT OR NON-FATAL ERRORS.

**POSSIBLE MEMORY MODULE/CONTROLLER ON HSC70

CX-905B

Figure 4-2 Operator Control Panel Fault Code Displays

The following paragraphs describe specific fault codes displayed in the OCP lamps. (All fault codes are indicated with octal values.)

1. **Fault Code 1 - K.pli error** - indicates the CIMGR initialization routine discovered bad requestor status from a previously-tested good requestor module in requestor slot 1. The expected requestor status should be 001. The FRU is the L0107.

   During CIMGR initialization, the K.ci is directed to set the HSC node address into its own control structure. If the K.ci failed to modify this node address field after one-half second from K.ci requestor initialization, this fault code is displayed. In addition, the K.pli microcode version is checked to ensure it is compatible with this functional version. If compatibility checks fail, this is the fault code displayed.

   Run offline diagnostics to test the K.ci requestor. Replace the K.pli module on failure. If the fault code persists, refer to the HSC revision control document to verify all HSC components are at the current revision.

2. **Fault Code 2 - K.sdi incorrect version of microcode** - All K.sdi modules are initialized during the Disk Server functional code initialization. If a K.sdi passes initialization, the Disk Server initialization code checks the K.sdi microcode version number to ensure it is compatible with this version of functional code. If code versions are not compatible, this fault code is displayed. The FRU is the L0108-YA.

3. **Fault Code 3 - K.sti incorrect version of microcode** - indicates tape data channel microcode is incompatible.

4. **Fault Codes 10, 11, and 12 - soft errors** - are the so-called soft or nonfatal errors related to the data channels, the K.ci host interface, and the P.ioj cache. None of these errors causes the HSC70 functional operation to suspend when the fault is reported. Once displayed, soft error indicators cannot be recalled. The HSC may buffer up to eight soft fault codes. Subsequent toggling of the Fault switch displays all remaining soft fault codes until the buffer is empty.

   - **Fault Code 10 - P.ioj cache failure** - results in disabling the cache and displaying this soft fault code for any failure detected in the J-11 instruction cache during HSC70 subsystem initialization while the HSC70 continues operation. Replace the P.ioj module (L0111) and reboot.

- **Fault Code 11 - K.ci failure** - is not present or has failed its initialization tests. This soft fault is displayed while the HSC continues to operate. The most probable FRU is the Port Link module (L0100).

- **Fault Code 12 - Data channel module failure** - is used to report an unknown requestor type was found in a requestor slot other than 0 or 1. Expected valid requestor types for requestor slots 2 through 8 are either 002 (L0108-YA) or 203 (L0108-YB). The data channel with the red LED on is the failing module.

5. **Fault Code 21 - P.ioj module failure** - indicates the P.ioj module is the most probable cause of the failure detected by the Init P.io Test. If possible, run the Offline P.io Test for a more definitive report on the error. Otherwise, replace the P.ioj module, and run the Init P.io test again. If the test still fails, run the Offline P.io test to help further isolate the failure.

6. **Fault Code 22 - M.std2 module failure** - indicates the M.std2 (memory) module is the most probable cause of this bootstrap failure. Possible causes include:

   - The failure of the memory test of the first 1 Kword (vector area) of Program memory as well as the use of the Swap Banks bit in the P.ioj in trying to correct the problem (Test 2).

   - A contiguous 8 Kword partition not found in Program memory below address 00160000 (Test 3).

   - A hard fault detected in the RX33 controller logic (Test 4).

   Determine the error that occurred by examining physical location 172340 which contains the number of the failing boot ROM test. In each of these cases, replace the M.std2 module, and run the initialization tests again. If the module still fails, run the Offline P.io Test.

   Enter the SETSHO utility and execute the SHO MEM command. If any memory locations appear in the suspect or disabled memory locations list, set the Secure/Enable switch to ENABLE and execute the SET MEM ENABLE/ALL command.

7. **Fault Code 23 - RX33 failure** - indicates a problem with an RX33 drive, the diskette, the RX33 controller, or the Read/Write logic on the memory module. This fault can be any of the following, in order of probability:

   - o A failure in the Read/Write logic of the M.std2 module. Replace M.std2.
   - o A faulty RX33 controller/drive interface cable. Replace the cable.
   - o No diskettes installed in the drives.
   - o Doors were left open on the RX33 drives.
   - o Neither diskette contains a bootable image.

   Ensure a known good HSC70 bootable media is properly loaded in one of the RX33 drives. If checking the obvious, doors and diskettes, does not remedy the situation, refer to Chapter 6 for more information before beginning repair. Running the Offline P.io and Offline RX33 tests (if possible) is strongly recommended before modules are replaced. These tests may help further isolate or define the problem.

8. **Fault Code 25 - Port Link node address switches out of range** - indicates the L0100 module node address switches are set to a value outside the currently-suggested range of 15 decimal.

9. **Fault Code 26 - missing files required** - indicates the System diskette does not contain one of the files necessary for operation of the HSC70 Control Program. This failure should occur only if one of the required files is inadvertently deleted from the HSC70 System diskette.

   Note the condition of the State light must be observed next prior to the fault occurrence. The State light is always steady (either ON or OFF) when the Fault light is lit during boot faults.

   While the State light is steady (ON) it can mean:

   - o SYSCOM.INI is not present on the load device.
   - o EXEC70.INI is not present on the load device.
   - o A version mismatch was found between either EXEC, SUBLIB, or SYSCOM and OLBVSN (Object Library Version Number).

While the State light is blinking it can mean:

- Any of the the normally-loaded programs (SINI, CERF, DEMON, etc.) is not present on the load device.
- A version mismatch was found on any one of the normally-loaded programs.

Replace the diskette with a backup copy.

10. **Fault Code 30 - No working K.ci, K.sdi, or K.sti in subsystem** - indicates the HSC70 does not contain any working K.ci, K.sti, or K.sdi modules. Either none are installed in the HSC70, or all the ones installed failed their initialization diagnostics. Also, if the Disk Server code is loaded, and no working K.sdi is found, this fault code is displayed.

    Insert the HSC70 Offline Diagnostic diskette into the RX33 and reboot the HSC70. When the Offline Loader prompts with ODL>, type **SIZE** followed by a carriage return. The SIZE command displays the status of all the Ks. This status indicates whether the modules are missing or are failing initialization diagnostics.

    If all else fails, replace the P.ioj (L0111) and check subsystem power for proper operation.

11. **OCP error code of 31** - indicates a crash occurred while the HSC70 was attempting to load and initialize its control program. Use Micro-ODT to diagnose these initialization crashes as follows:

    a. Press the break key on the local console terminal.

    b. Type 17 777 656/

       This is the address of the UPAR7 register. The reason for reboot codes are stored in UPAR7 bits 8 to 11 when an OCP code of 31 has been detected. The other UPAR registers store useful information for some of the errors related to an OCP fault code of 31. Refer to the fault code 31 reasons in the following paragraphs for UPAR content usage. Table 4-1 shows the addresses of the UPAR registers.

    c. Analyze bits 8 to 11 of the 16-bit message displayed by examining UPAR7. Table 4-2 shows the bit/error relationship.

4. If this error occurs repeatedly, it indicates an intermittent hardware error or degraded diskette media. The boot-in-progress flag is indicated by KPDR7 bit 3 set. The KPDR7 register address is 17 772 316. Use micro-ODT to examine bit 3 (it can be reset).

12. **OCP error code 32** - indicates an inconsistency in the software. Reboot the HSC. If this failure persists, use a backup copy of the System diskette. If the failure still persists, use the Offline diagnostics to help isolate any hardware failures in the subsystem. Also, try using an earlier version of the HSC operating software.

Table 4-1 UPAR Register Addresses

Register	Address
UPAR0	17 777 640
UPAR1	17 777 642
UPAR2	17 777 644
UPAR3	17 777 646
UPAR4	17 777 650
UPAR5	17 777 652
UPAR6	17 777 654
UPAR7	17 777 656

Table 4-2 Control Program Bits

16 BIT MESSAGE	MEANING	FRUS
X XXX XXX 1XX XXX XXX	NXM	LO111 LO117 Software
X XXX XX1 0XX XXX XXX	Illegal Inst.	LO111 LO117 Software
X XXX XX1 1XX XXX XXX	Parity Trap	LO117 LO111
X XXX X10 0XX XXX XXX	Level 7 Interrupt	LO108 LO107
X XXX X10 1XX XXX XXX	MMU Trap	LO111 Software
X XXX X11 0XX XXX XXX	Software Crash	Software
X XXX 111 XXX XXX XXX	K.ci Host Reset	LO117
X XXX 100 0XX XXX XXX	User Requested Reboot	N/A

The following list describes actions to be taken for each type of error related to an OCP fault code of 31 as pointed out by examining UPAR7.

- **NXM Trap:** Examine UPAR1 to find the lower 16 bits of the failing memory address by typing 17 777 642/. Examine UPAR2's lower byte for the high 6 bits of the failing memory address by typing 17 777 644/.
- **Illegal Inst:** Obtain a crash dump and analyze the crash to find the failing instuction.
- **Parity Trap:** Use the same method for parity traps as you did for NXM traps to determine the failing address.
- **Level 7 Interrupt:** Determine which K has interrupted the system by examining UPAR0 through UPAR4. Refer to Table 4-1 for the address of each UPAR register. Each byte of each register contains module status for each requester (K) in the HSC70. Refer to Appendix C to determine a failing status code is. Refer to Table 4-3 for the designation of requesters to UPAR registers for a level 7 interrupt.

- **Memory Management Unit (MMU) Trap:** Examine UPAR1, UPAR2, and UPAR3 to determine the status of the MMU at the time of the OCP fault code of 31. When a MMU trap occurs, status of the MMU is found in these registers.

- **Software crash:** Check the first word on the kernal stack to determine the reason for failing software. Refer to Appendix B.

- **K.ci Host Reset:** Hit the break key again and at the @ symbol type **17 770 000/** when a host reset is known as the reason for an OCP fault code of 31. This is the address of control memory window 0. When the / is hit, the contents of control window 0 are displayed. Enter a **0** into this location followed by a carriage return. Then type **16 000 002/**. This is the second location in control memory. The number displayed as the contents of 16 000 002 is the number of the host that issued the HOST RESET command.

Table 4-3 Status of Requestors For Level 7 Interrupt

REGISTER	HIGH BYTE	LOW BYTE
UPAR0	REQ 2	REQ 1
UPAR1	REQ 4	REQ 3
UPAR2	REQ 6	REQ 5
UPAR3	REQ 8	REQ 7
UPAR4	N/A	REQ 9

4.3.3 Init P.io Test Summaries

The Init P.io Test does not use a test numbering scheme for the following reasons:

1. Test numbering adds overhead to the program both in execution time and the memory size required for the program. Because boot time is critical, the extra overhead is not justified.

2. The only goal of the Init P.io Test is to provide module callout on the fault code display.

3. The Offline P.io Test is provided for those situations where a repair level diagnostic is needed. This offline test produces standard HSC70 error reports. Chapter 6 describes each of the tests. They are functionally identical to the tests provided in the Init P.io Test.

CHAPTER 5
INLINE DIAGNOSTICS

5.1 INTRODUCTION

Inline diagnostics executing in the HSC do not interfere with normal operation. The following sections describe these tests:

- Inline RX33 Diagnostic Test
- Inline Memory Test
- Inline Disk Drive Diagnostic Test
- Inline Tape Test
- Inline Tape Compatibility Test
- Inline Multidrive Exerciser

5.1.1 Inline Diagnostics Commonalities

All inline diagnostics have two common areas: all test prompts and error messages conform to standard formats. All prompts issued by these diagnostics use a generic syntax.

- Prompts requiring user action or input are always followed by a question mark.
- Prompts offering a choice of responses show those choices in parentheses.
- A capital D in parentheses indicates the response should be in decimal.
- The square brackets enclose the prompt default or if empty, indicate no default exists for that prompt.

5.1.1.1 Inline Diagnostics Generic Error Message Format - All inline diagnostics follow a generic error message format, as follows:

```
XXXXXX>D>tt:tt T#aaa E#bbb   U-ccc
<Text string describing error>
FRU1-dddddd FRU2-dddddd
MA -eeeeee
EXP-yyyyyy
ACT-zzzzzz
```

where:

XXXXXX> = Appropriate inline diagnostic prompt

D> = Letter indicating the diagnostic was initiated on demand. This field can contain a D, an A (diagnostic initiated automatically), or a P (diagnostic initiated as part of the periodic diagnostics).

tt:tt	=	Current time
aaa	=	Decimal number denoting test that failed
bbb	=	Decimal number denoting error detected
ccc	=	Unit number of drive being tested
FRU1	=	Most likely Field Replaceable Unit (FRU)
FRU2	=	Next most likely FRU
dddddd	=	Name of Field Replaceable Unit
MA	=	Media Address
eeeeee	=	Octal number denoting Offset within block
yyyyyy	=	Octal number denoting data expected
zzzzzz	=	Octal number denoting data actually found

The first line of the error message contains general information concerning the error. The second line describes the nature of the error. Lines 1 and 2 are mandatory and appear in all error messages. Line 3 and any succeeding lines display additional information and are optional.

5.2 INLINE RX33 DIAGNOSTIC TEST (ILRX33)

The Inline RX33 diagnostic tests either of the RX33 drives attached to the HSC70. This test runs concurrently with other HSC70 processes and uses the services of the HSC70 Control Program and the Diagnostic Execution Monitor (DEMON). The Inline RX33 test performs several writes and reads to verify the RX33 internal data paths and read/write electronics.

5.2.1 ILRX33 System Requirements

Hardware requirements include:

- P.io (processor) module with HSC70 boot ROMs
- At least one M.std2 (memory) module

- RX33 controller with at least one working drive
- Console terminal

NOTE

A scratch diskette is not required. This test does not destroy any data on the system software diskette.

This program tests only the RX33 and the data path (serial line) between the P.ioj and the RX33. All other system hardware is assumed working.

Software requirements include:

- HSC70 Control Program
- Diagnostic Execution Monitor (DEMON)

5.2.2 ILRX33 Operating Instructions

Typing a **CTRL Y** starts ILRX33. The keyboard monitor responds with a KMON prompt (HSC>). Next, typing either **RUN ILRX33** or **RUN DX0:ILRX33** followed by a carriage return initiates the Inline RX33 Test.

If the Inline RX33 Test cannot load from the specified diskette, try loading the test from the other diskette. For example, if RUN ILRX33 fails, try RUN DX1:ILRX33.

5.2.3 ILRX33 Test Parameter Entry

The device name of the RX33 drive to be tested is the only parameter sought by this test. When the test is invoked, the following prompt is displayed:

```
Device Name of RX33 to test (DX0:, DX1:, LB:) [] ?
```

NOTE

The string, LB:, indicates the RX33 drive last used to boot the HSC70 Control Program.

One of the indicated strings must be entered. If one of these strings in not entered, the test prints **Illegal Device Name**, and the prompt is repeated.

5.2.4 ILRX33 Setting/Clearing

ILRX33 only verifies a particular RX33 drive and controller combination is working or failing and should not be used as a troubleshooting aid. This test does not support any flags. Because the test always reads and writes the same block of the diskette, looping the test would eventually result in media damage. If the test indicates a particular controller or drive is not operating correctly, the proper repair strategy is to replace the drive and/or controller.

5.2.5 ILRX33 Progress Reports

At the end of the test, the following message is displayed:

ILRX33>D>tt:tt Execution Complete

where: tt:tt = current time

5.2.6 ILRX33 Test Termination

This test is terminated by typing a ^Y (CTRL Y). The test automatically terminates after reporting an error with one exception. If the error displayed is RETRIES REQUIRED, the test continues.

5.2.7 ILRX33 Error Message Example

All error messages produced by the Inline RX33 Test conform to the HSC diagnostic error message format (Section 5.1.1.1). Following is a typical ILRX33 error message:

```
ILRX33>D>00:00 T 001 E 003 U- 50182
ILRX33>D> No Diskette Mounted
ILRX33>D> FRU1-Drive
```

Other optional lines are found on different error messages.

5.2.8 ILRX33 Error Messages

The following paragraphs list specific information about each of the errors produced by the Inline RX33 Test. Hints about the possible cause of the error are provided where feasible.

- **Error 000 - RETRIES REQUIRED** - indicates a Read or Write operation failed when first attempted, but succeeded on one of the retries performed automatically by the RX33 driver software. This error normally indicates the diskette media is degrading and the diskette should be replaced.

- **Error 001 - OPERATION ABORTED** - is reported if the ILRX33 test is aborted by a CTRL Y.

- **Error 002 - WRITE PROTECTED** - indicates the RX33 drive being tested contains a write-protected diskette. Write enable the diskette and try again. If the diskette is not write protected, the RX33 drive or controller is faulty.

- **Error 003 - NO DISKETTE MOUNTED** - indicates the RX33 drive being tested does not contain a diskette. Insert a diskette before repeating the test. If this error is displayed when the drive does contain a diskette, the drive or controller is at fault.

- **Error 004 - HARD I/O ERROR** - indicates the program encountered a hard error while attempting to read or write the diskette.

- **Error 005 - BLOCK NUMBER OUT OF RANGE** - indicates the RX33 driver detected a request to read a block number outside the range of legal block numbers (0 thru 2399 decimal). Because the Inline RX33 Test reads and writes disk block 001, it may indicate a software problem.

- **Error 006 - UNKNOWN STATUS**
  STATUS=xxx - indicates the Inline RX33 Test received a status code it did not recognize. The octal value xxx represents the status byte received. RX33 reads and writes are performed for the Inline RX33 Test by the HSC Control Program's RX33 driver software. At the completion of each Read or Write operation, the driver software returns a status code to the RX33 test, describing the result of the operation. The test decodes the status byte to produce a description of the error.

  An UNKNOWN STATUS error indicates the status value received from the driver did not match any of the status values known to the test. The status value returned (xxx) is displayed to help determine the cause of the problem. Any occurrence of this error should be reported via a Software Performance Report (SPR). See Appendix B for detailed information on SPR submission.

- **Error 007 - DATA COMPARE ERROR**
  MA -aaaaaa
  EXP-bbbbbb
  ACT-cccccc - indicates data written to the diskette does not agree with the data subsequently read back. The field aaaaaa represents the address of the failing word within the block (512 bytes) that was read. The field bbbbbb represents the data written to the word and the field cccccc represents the data read back from the word. Because this test only reads and writes block 1 of the diskette, all failures occur while trying to access physical block 1.

- **Error 008 - ILLEGAL DEVICE NAME** - indicates the user specified an illegal device name when the program prompted for the name of the drive to be tested. Legal device names include: DX0:, DX1: and LB:. LB: indicates the drive from which the system was last booted. After displaying this error, the program again prompts for a device name. Enter one of the legal device names to continue the test.

5.2.9 ILRX33 Test Summary

The test summary for this diagnostic is contained in the following paragraphs.

- **Test 001 - Read/Write Test** - verifies data can be written to the diskette and read back correctly. All reads and writes access physical block 1 of the RX33 (the RT-11 Volume ID Block). This block is not used by the HSC operating software.

  Initially, the contents of block 1 are read and saved. Then three different data patterns are written to block 1, read back, and verified. This checks the read/write electronics in the drive and the internal data path between the RX33 controller and the drive. Following the Read/Write Test, the original contents of block 1 are written back to the diskette.

  If the data read back from the diskette does not match the data written, a Data Compare Error is generated. The error report lists the word (MA) in error within the block together with the expected (EXP) and actual (ACT) contents of the word.

5.3 INLINE MEMORY TEST (ILMEMY)

The Inline Memory test is designed to test HSC70 data buffers. This test can be initiated automatically or on demand. It is initiated automatically to test data buffers that produced a parity error when in use by the HSC70 Control Program. Buffers that fail the memory test are removed from service by sending them to the Disabled Buffer Queue. Buffers sent twice to this test, but not failing the memory test are also sent to the Disabled Buffer Queue. Buffers that pass the memory test and have not been tested previously are sent to the Free Buffer Queue for further use by the HSC70 Control Program.

When the test is initiated on demand, any buffers on the Disabled Buffer Queue are tested and, the results of the test are displayed on the terminal from which the test was initiated.

This test runs concurrently with other HSC70 processes and uses the services of the HSC70 Control Program and the Diagnostic Execution Monitor (DEMON).

5.3.1 ILMEMY System Requirements

Hardware requirements include:

- P.ioj (processor) module with HSC70 boot ROMs
- At least one M.std2 (memory) module
- RX33 controller with at least one working drive
- A console terminal (demand initiation only)

This program only tests data buffers located in the HSC70 Data memory. All other system hardware is assumed to be working.

Software requirements include:

- HSC70 Control Program (System diskette)
- Diagnostic Execution Monitor (DEMON)

5.3.2 ILMEMY Operating Instructions

To start this test, type a **CTRL Y** to get the attention of the HSC70 keyboard monitor. The keyboard monitor responds to the CTRL Y with a prompt

```
HSC70>
```

Type **RUN DX0:ILMEMY** and a carriage return to initiate the Inline Memory Test. This program has no user-supplied parameters or flags.

If the Inline Memory test is not contained on the specified diskette (DX1:), an error message is displayed.

5.3.3 ILMEMY Progress Reports

Error messages are displayed as needed. At the end of the test, the following message is displayed (by DEMON):

```
ILMEMY>D>tt:tt Execution Complete
```

where: tt:tt = current time

5.3.4 ILMEMY Error Message Example

All error messages produced by the Inline Memory test conform to the HSC70 diagnostic error message format (Section 5.1.1.1). Following is a typical ILMEMY error message:

```
ILMEMY>A>09:33  T 001 E 000
ILMEMY>A>Tested Twice with no Error (Buffer Retired)
ILMEMY>A>FRU1-M.std2  FRU2-
ILMEMY>A>Buffer Starting Address (physical) = 15743600
ILMEMY>A>Buffer Ending Address   (physical) = 15744776
```

5.3.5 ILMEMY Error Messages

The following list shows specific information about each of the errors displayed by the Inline Memory Test.

- **Error 000 TESTED TWICE WITH NO ERROR** - indicates the buffer under test passed the memory test. However, this is the second time the buffer was sent to the memory test and passed it. Because the buffer has a history of two failures while in use by the Control Program yet does not fail the memory test, intermittent failures on the buffer are assumed. The buffer is retired from service and sent to the Disabled Buffer Queue.

- **Error 001 RETURNED BUFFER TO FREE BUFFER QUEUE** - indicates a buffer failed during use by the Control Program but the Inline Memory test detected no error. Because this is the first time the buffer was sent to the Inline Memory test, it is returned to the Free Buffer Queue for further use by the HSC70 Control Program. The address of the buffer is stored by the Inline Memory test in case the buffer again fails when in use by the Control Program.

- **Error 002 MEMORY PARITY ERROR** - indicates a parity error occurred while testing a buffer. The buffer is retired from service and sent to the Disabled Buffer Queue.

- **Error 003 MEMORY DATA ERROR** - indicates the wrong data was read while testing a buffer. The buffer is retired from service and sent to the Disabled Buffer Queue.

5.3.6 ILMEMY Test Summaries

Test 001 receives a queue of buffers for testing. If the Inline Memory test is initiated automatically, the queue consists of buffers from the Suspect Buffer Queue.

When the HSC70 Control Program detects a parity error in a data buffer, the buffer is sent to the Suspect Buffer Queue. While on this queue, the buffer is not used for data transfers. The HSC70 Continuous Scheduler periodically checks the Suspect Buffer Queue

to see if it contains any buffers. If buffers are found on the queue, they are removed, and the Inline Memory test is automatically initiated to test those buffers.

If the ILMEMY test is initiated on demand, it retests only buffers already known as disabled (a rather useless exercise).

If the test is initiated automatically, and the buffer passes the test, the program checks to see if this is the second time the buffer was sent to the Inline Memory test. If this is the case, the buffer is probably producing intermittent errors. The buffer is retired from service and sent to the Disabled Buffer Queue. If this is the first time the buffer is sent to the Inline Memory test, it is returned to the Free Buffer Queue for further use by the HSC70 Control Program. In this last case, the address of the buffer is saved in case the buffer again fails and is sent to the Inline Memory test a second time.

When all buffers on the test queue are tested, the Inline Memory Test terminates.

5.4 INLINE DISK DRIVE DIAGNOSTIC TEST (ILDISK)

The Inline Disk Drive Diagnostic (ILDISK) isolates disk drive-related problems to one of the following three Field Replaceable Units (FRUs):

1. Disk drive
2. SDI cable
3. HSC Disk Data Channel module

The Inline Disk Drive Diagnostic runs in parallel with disk I/O from a Host CPU. However, the drive being diagnosed cannot be Online to any host. This diagnostic can be initiated upon demand via the console terminal or automatically by the HSC70 Control Program when an unrecoverable disk drive failure occurs.

Currently, ILDISK is automatically invoked by default whenever (with one exception) a drive is declared inoperative. The exception is if a drive is declared inoperative while in use by a diagnostic or utility. Automatic initiation of ILDISK can be inhibited by issuing the SETSHO command, SET AUTO DISABLE. If the SET AUTO DISABLE command is issued, ILMEMY (a test for suspect buffers) is also disabled. For this reason, leaving ILDISK automatically enabled is preferable.

The tests performed vary, depending on whether the drive is known to the HSC70 Control Program.

1. DRIVE UNKNOWN - to the HSC70 Control Program. It is either unable to communicate with the HSC70 or was

communicating and declared inoperative when it failed during use by the HSC70. In this case, because the drive cannot be identified by unit number, the user must supply the requestor number and port number of the drive. Then the SDI verification tests can execute. The SDI verification tests check the path between the K.sdi and the disk drive and command the drive to run its self-test diagnostics. If the SDI verification tests fail, the most probable FRU is identified in the error report. If the SDI verification tests pass, presume the drive is the FRU.

2. DRIVE KNOWN - to the HSC70 Control Program, (i.e. identifiable by unit number). Read/write/format tests are performed in addition to the SDI verification tests. If an error is detected, the most probable FRU is identified in the error report. If no errors are detected, presume the FRU is the drive.

5.4.1 ILDISK System Requirements

Software requirements of this test include the HSC70 Control Program, the Control Program disk functional code, and DEMON. Hardware requirements include the disk drive and a disk data channel, connected by an SDI cable. The test assumes the I/O Control Processor module, and the memory module are working.

A service manual for the disk drive is required to interpret errors that occur in the drive's self-test diagnostics.

5.4.2 ILDISK Operating Instructions

Use the following steps to initiate ILDISK:

1. Type a **CTRL Y**.

2. In response to the prompt

   HSC70>

   type **RUN DX0:ILDISK**, followed by a carriage return.

3. Wait until ILDISK is read from the system software load media into the HSC70 Program memory.

4. Enter parameters after ILDISK is started. Refer to Section 5.4.4.

5.4.3 ILDISK Availability

If a diskette containing the Inline Disk Drive Diagnostic is not loaded when you enter the **R ILDISK** command, an error message is displayed. Insert the Operating System diskette containing ILDISK and repeat Section 5.4.2.

5.4.4 ILDISK Test Parameter Entry

Upon demand initiation, ILDISK first prompts:

DRIVE UNIT NUMBER (U) [] ?

Enter the unit number of the disk drive for test. Unit numbers are in the form Dnnnn, where nnnn is a decimal number between 0 and 4095 corresponding to the number printed on the drive unit plug. Terminate the unit number response with a carriage return. ILDISK attempts to acquire the specified unit via the HSC70 Diagnostic Interface. If the unit is acquired successfully, ILDISK next prompts for the drive diagnostic to be executed. If the acquire fails, one of the following conditions was encountered:

1. The specified drive is UNAVAILABLE. This indicates the drive is connected to the HSC70 but is currently online to a host CPU or an HSC70 utility. Online drives cannot be diagnosed. ILDISK repeats the prompt for the unit number.

2. The specified drive is UNKNOWN to the HSC70 Disk Functional software. Drives are UNKNOWN for one of the following reasons:

   - The drive and/or disk data channel port is broken and cannot communicate with the disk functional software.

   - The drive was previously communicating with the HSC70 but a serious error occurred, and the HSC70 has ceased communicating with the drive (marked the drive as inoperative).

   In either case, ILDISK asks if you desire to enter a requestor number and port number. Refer to Section 5.4.5.

After receiving the unit number (or requestor and port), ILDISK prompts:

RUN A SINGLE DRIVE DIAGNOSTIC (Y/N) [N] ?

Typing a carriage return causes the drive to execute its entire diagnostic set. Typing a **Y** followed by a carriage return

executes a single drive diagnostic. If a single drive diagnostic is selected, the test prompts:

1 DRIVE TEST NUMBER (H) [] ?

Enter a hexadecimal number specifying the drive diagnostic to be executed. Consult the appropriate disk maintenance or service manual to determine the number of the test to perform. Entering a test number not supported by the drive results in an error #13 generated in Test 5.

The test prompts for the number of passes to perform:

OF PASSES TO PERFORM (1 to 32767) (D) [1] ?

Enter a decimal number between 1 and 32767 specifying the number of test repetitions. Terminate the response with a carriage return. Typing a carriage return, without entering a number, runs the test once.

5.4.5 Specifying Requestor And Port - ILDISK

Drives unknown to the HSC70 disk functional software are tested by specifying the requestor number and port number of the drive. Requestor number is any number 2 through 9 specifying the disk data channel connected to the drive under test.

Port number is 0 through 3 specifying which of four disk data channel ports is connected to the drive under test. The requestor number and port number can be determined in one of two ways:

1. By tracing the SDI cable from the desired disk drive to the HSC70 bulkhead connector, then tracing the bulkhead connector to a specific port on one of the disk data channels.

2. By using the SHOW DISKS command to display the requestor and port numbers of all known drives. To use this method, exit ILDISK by typing a CTRL Y. Type **SHOW DISKS** in response to the HSC70 prompt. This command displays a list of all known drives including the requestor number and port number for each drive. Each disk data channel has four possible ports to which a drive can be connected. By inference, the port number of the unknown unit must be one not listed in the SHOW DISKS display (assuming the unknown drive is not connected to a defective disk data channel). A defective disk data channel illuminates red LED on the lower front edge of the module. Refer to Chapter 2.

After a requestor number and a port number are supplied to ILDISK, the program checks to ensure the specified requestor and

port do not match any drive known to the HSC70 software. If the requestor and port do not match a known drive, ILDISK prompts for the number of passes to perform, as described in Section 5.4.4. If the requestor and port do match a known drive, ILDISK reports Error 08.

5.4.6 ILDISK Progress Reports

ILDISK produces an end-of-pass report at the completion of each pass of the diagnostic. One pass of the program can take several minutes depending upon the type of drive being diagnosed.

5.4.7 ILDISK Test Termination

ILDISK is terminated by typing a CTRL Y or CTRL C. A CTRL Y/CTRL C may not take effect immediately because certain parts of the program cannot be interrupted. An example would be during SDI commands. Two minutes may be necessary to respond to a CTRL Y or CTRL C if either is entered while an SDI DRIVE DIAGNOSE command is in progress.

5.4.8 ILDISK Error Message Example

All error messages produced by the Inline Disk Drive diagnostic conform to the HSC70 diagnostic error message format (Section 5.1.1.1). Following is a typical ILDISK error message.

```
ILDISK>D>09:35 T 005 E 035 U-D00082
ILDISK>D>Drive Diagnostic Detected Fatal Error
ILDISK>D>FRU1-Drive   FRU2-
ILDISK>D>Requestor Number 04
ILDISK>D>Port Number 03
ILDISK>D>Test  0025 Error  007F
ILDISK>D>End Of Pass  00001
```

5.4.9 ILDISK Error Messages

Messages produced by ILDISK are described in the following list:

- **Error 01 DDUSUB INITIALIZATION FAILURE** - The HSC70 diagnostic interface did not initialize. Error 01 is not recoverables and is caused by:

  1. Insufficient memory to allocate buffers and control structures required by the diagnostic interface.

  2. HSC Disk Functional software is not loaded.

- **Error 02 UNIT SELECTED IS NOT A DISK** - The response to the unit number prompt was not of the form Dnnnn. Refer to Section 5.4.4.

- **Error 03 DRIVE UNAVAILABLE** - The selected disk drive is not available for diagnostic use.

- **Error 04 UNKNOWN STATUS FROM DDUSUB** - A call to the diagnostic interface resulted in the return of an unknown status code. This indicates a software error and should be reported via a Software Performance Report (SPR). See Appendix B for detailed information on SPR submission.

- **Error 05 DRIVE UNKNOWN TO DISK FUNCTIONAL CODE** - The disk drive selected is not known to the HSC Disk Functional software. The drive may not be communicating with the HSC, or the disk functional software may have disabled the drive due to an error condition. ILDISK prompts the user for the drive's requestor and port. Refer to Section 5.4.5 for information on specifying requestor and port.

- **Error 06 INVALID REQUESTOR OR PORT NUMBER SPECIFIED** - The Requestor number given was not in the range 2 through 9, or the port number given was not in the range 0 through 3. Specify a requestor and port within the allowable ranges.

- **Error 07 REQUESTOR SELECTED IS NOT A K.SDI** - The requestor specified was not a Disk Data Channel (K.sdi). Specify a requestor that contains a Disk Data Channel.

- **Error 08 SPECIFIED PORT CONTAINS A KNOWN DRIVE** - The requestor and port specified contain a drive known to the HSC Disk Functional software. The unit number of the drive is supplied in the report. ILDISK does not allow testing a known drive via requestor number and port number.

- **Error 09 DRIVE CAN'T BE BROUGHT ONLINE** - A failure occurred when ILDISK attempted to bring the specified drive Online. One of the following conditions occurred:

  1. UNIT IS OFFLINE - The specified unit went to the OFFLINE state and now cannot communicate with the HSC70.

  2. UNIT IS IN USE - The specified unit is now marked as in use by another process.

  3. UNIT IS A DUPLICATE - Two disk drives are connected to the HSC70, both with the same unit number.

  4. UNKNOWN STATUS FROM DDUSUB - The HSC70 diagnostic interface returned an unknown status code when ILDISK attempted to bring the drive Online. Refer to Error 04 for related information on this error.

- **Error 10 K.SDI DOES NOT SUPPORT MICRODIAGNOSTICS** - The K.sdi connected to the drive under test does not support microdiagnostics. This indicates the K.sdi microcode is not at the latest revision level. This is not a fatal error, but the K.sdi should probably be updated with the latest microcode to improve error detection capabilities.

- **Error 11 CHANGE MODE FAILED** - ILDISK issued an SDI CHANGE MODE command to the drive and the command failed. The drive is presumed the failing unit, because the SDI interface was previously verified.

- **Error 12 DRIVE DISABLED BIT SET** - The SDI verification test issued an SDI GET STATUS command to the drive under test. The Drive Disabled bit was set in the status returned by the drive, indicating the drive detected a serious error and is now disabled.

- **Error 13 COMMAND FAILURE** - The SDI verification test detected a failure while attempting to send an SDI command to the drive. One of the following occurred:

  1. DID NOT COMPLETE - The drive did not respond to the command within the allowable time. Further SDI operations to the drive are disabled.

  2. K.SDI DETECTED ERROR - The K.sdi detected an error condition while sending the command or while receiving the response.

  3. UNEXPECTED RESPONSE - The SDI command resulted in an unexpected response from the drive. This error can be caused by a DIAGNOSE command if a single drive diagnostic was selected, and the drive does not support the specified test number.

- **Error 14 CAN'T WRITE ANY SECTOR ON TRACK** - As part of test 04, ILDISK attempts to write a pattern to at least one sector of each track in the Read/Write area of the drive DBN space. (DBN space is an area on every disk drive reserved for diagnostic use.) During the write process, ILDISK detected a track with no sector that passed the Read/Write test. (ILDISK could not write a pattern and read it back successfully on any sector on the track.) The error information for the last sector accessed is identified in the error report. The most probable cause of this error is a disk media error.

  If test 03 also failed, the problem could be in the disk Read/Write electronics, or the DBN area of the disk may not be formatted correctly. To interpret the MSCP status code, refer to Section 5.4.9.1.

- **Error 15 READ/WRITE READY NOT SET IN ONLINE DRIVE** - The SDI verification test executed a command to interrogate the Real Time Drive State line of the drive. The line status reported the drive was in the Online state, but the Read/Write Ready bit was not set in the status.

- **Error 16 ERROR RELEASING DRIVE** - ILDISK attempted to release the drive under test. The release operation failed. One of the following occurred:

  1. COULD NOT DISCONNECT - An SDI DISCONNECT command to the drive failed.

  2. UNKNOWN STATUS FROM DDUSUB - Refer to Error 04.

- **Error 17 INSUFFICIENT MEMORY, TEST NOT EXECUTED** - The SDI verification test could not acquire sufficient memory for control structures. The SDI verification test could not be executed. Use the SETSHO command, SHOW MEMORY, to display available HSC memory. If any disabled memory appears in the display, consider further testing of the memory module. If no disabled memory is displayed, and no other diagnostic or utility is active on this HSC, submit an SPR.

- **Error 18 K MICRODIAGNOSTIC DID NOT COMPLETE** - The SDI verification test directed the disk data channel to execute one of its microdiagnostics. The microdiagnostic did not complete within the allowable time. All drives connected to the disk data channel may now be unusable (if the microdiagnostic never completes), and the HSC70 probably must be rebooted. The disk data channel module is the probable failing FRU.

- **Error 19 K MICRODIAGNOSTIC REPORTED ERROR** - The SDI verification test directed the disk data channel to execute one of its microdiagnostics. The microdiagnostic completed and reported an error. The disk data channel is the probable FRU.

- **Error 20 DCB NOT RETURNED, K FAILED FOR UNKNOWN REASON** - The SDI verification test directed the disk data channel to execute one of its microdiagnostics. The microdiagnostic completed without reporting any error, but the disk data channel did not return the Dialogue Control Block (DCB). All drives connected to the disk data channel may now be unusable. The disk data channel is the probable FRU and the HSC70 will probably have to be rebooted.

- **Error 21 ERROR IN DCB ON COMPLETION** - The SDI verification test directed the disk data channel to

execute one of its microdiagnostics. The microdiagnostic completed without reporting any error, but the disk data channel returned the Dialogue Control Block (DCB) with an error indicated. The disk data channel is the probable FRU.

- **Error 22 UNEXPECTED ITEM ON DRIVE SERVICE QUEUE** - The SDI verification test directed the disk data channel to execute one of its microdiagnostics. The microdiagnostic completed without error, and the disk data channel returned the Dialogue Control Block with no errors indicated. However, the disk data channel sent the Drive State Area to its service queue, indicating an unexpected condition in the disk data channel or drive.

- **Error 23 FAILED TO REACQUIRE UNIT** - In order for ILDISK to allow looping, the drive under test must be released and then reacquired. (This method is required to release the drive from the Online state.) The release operation succeeded, but the attempt to reacquire the drive failed. One of the following conditions occurred:

  1. DRIVE UNKNOWN TO DISK FUNCTIONAL CODE - A fatal error caused the HSC70 Disk Functional software to declare the drive inoperative, hence the drive unit number is not recognized. The drive must now be tested by specifying requestor and port number.

  2. DRIVE UNAVAILABLE - The specified drive is now not available for diagnostic use.

  3. UNKNOWN STATUS FROM DDUSUB - Refer to Error 04.

- **Error 24 STATE LINE CLOCK NOT RUNNING** - The SDI verification test executed a command to interrogate the Real Time Drive State of the drive. The returned status indicates the drive is not sending State Clock to the disk data channel. Either the port, SDI cable, or drive is defective or the port is not connected to a drive.

- **Error 25 ERROR STARTING I/O OPERATION** - ILDISK detected an error when initiating a disk read or write operation. One of the following conditions occurred:

  1. INVALID HEADER CODE - ILDISK did not supply a valid header code to the HSC70 diagnostic interface. This indicates a software error and should be reported via a Software Performance Report (SPR). See Appendix B for detailed information on SPR submission.

2. COULD NOT ACQUIRE CONTROL STRUCTURES - The HSC70 diagnostic interface could not acquire sufficient control structures to perform the operation.

3. COULD NOT ACQUIRE BUFFER - The HSC70 diagnostic interface could not acquire a buffer needed for the operation.

4. UNKNOWN STATUS FROM DDUSUB - The HSC70 diagnostic interface returned an unknown status code. Refer to Error 04.

NOTE

Retry ILDISK during lower HSC activity for problems 2 and 3, if these errors persist.

- **Error 26 INIT DID NOT STOP STATE LINE CLOCK** - The SDI verification test sent an SDI INITIALIZE command to the drive. When the drive receives this command, it should momentarily stop sending State Line Clock to the disk data channel. The disk data channel did not see the State Line Clock stop after sending the Initialize. The drive is the most probable FRU.

- **Error 27 STATE LINE CLOCK DID NOT START UP AFTER INIT** - The SDI verification test sent an SDI INITIALIZE to the drive. When the drive receives this command, it should momentarily stop sending State Clock to the disk data channel. The disk data channel saw the State Clock stop, but the clock never restarted. The drive is the most probable FRU.

- **Error 28 I/O OPERATION LOST** - While ILDISK was waiting for a disk read or write operation to complete, the HSC70 diagnostic interface notified ILDISK that no I/O operation was in progress. This error may be induced by a hardware failure but indicates a software problem that should be reported by a Software Performance Report (SPR). See Appendix B for detailed information on SPR submission.

- **Error 29 ECHO DATA ERROR** - The SDI verification test issued an SDI ECHO command to the drive. The command completed but the wrong response was returned by the drive. The SDI set and the disk drive are the probable FRUs.

- **Error 30 DRIVE WENT OFFLINE** - The drive, previously acquired by the diagnostic, is now unknown to the disk functional code. This indicates the drive spontaneously went Offline or stopped sending clocks and is now

unknown. The test should be restarted using the requestor and port numbers instead of drive unit number.

- **Error 31 DRIVE ACQUIRED BUT CAN'T FIND CONTROL AREA** - The disk drive was acquired, and ILDISK obtained the requestor number and port number of the drive from the HSC70 diagnostic interface. However, the specified requestor does not have a control area. This indicates a software problem and should be reported via a Software Performance Report (SPR). See Appendix B for detailed information on SPR submission.

- **Error 32 REQUESTOR DOES NOT HAVE CONTROL AREA** - ILDISK cannot find a control area for the requestor supplied by the user. One of the following conditions exists:

  1. The HSC70 does not contain a disk data channel (or other type of requestor) in the specified requestor position.

  2. The disk data channel (or other type of requestor) in the specified requestor position failed its initialization diagnostics and is not in use by the HSC70.

  Open the HSC70 front door and remove the cover from the card cage. Locate the module slot in the card cage that corresponds to the requestor. Refer to the module utilization label above the card cage to help locate the proper requestor. If a blank module (air baffle) is in the module slot, the HSC70 does not contain a requestor in the specified position. If a requestor is in the module slot, ensure the red LED on the lower front edge of the module is lit. If so, the requestor failed and was disabled by the HSC70. If the red LED is not lit, a software problem exists and should be reported via a Software Performance Report (SPR). See Appendix B for detailed information on SPR submission.

- **Error 33 CAN'T READ ANY SECTOR ON TRACK** - As part of Test 03, ILDISK attempts to read a pattern from at least one sector of each track in the Read Only area of the drive DBN space. (DBN space is an area on every disk drive reserved for diagnostic use.) All drives have the same pattern written to each sector in the Read Only DBN space.

  During the read process, ILDISK detected a track that does not contain any sector with the expected pattern. Either ILDISK detected errors while reading or the read succeeded, but the sectors did not contain the correct pattern. The error information for the last sector accessed is supplied in the error report. The most

likely cause of this error is a disk media error. If Test 04 also fails, the problem may be in the disk Read/Write electronics, or the DBN area of the disk may not be formatted correctly. To interpret the MSCP status code, refer to Section 5.4.9.1.

- **Error 34 DRIVE DIAGNOSTIC DETECTED ERROR** - The SDI verification test directed the disk drive to run an internal diagnostic. The drive indicated the diagnostic failed, but the error is not serious enough to warrant removing the drive from service. The test number and error number for the drive are displayed (in hex) in the error report. For the exact meaning of each error, refer to the service manual for that drive.

- **Error 35 DRIVE DIAGNOSTIC DETECTED FATAL ERROR** - The SDI verification test directed the disk drive to run an internal diagnostic. The drive indicated the diagnostic failed and the error is serious enough to warrant removing the drive from service. The test and error number are displayed (in hex) in the error report. For the exact meaning of each error, refer to the service manual for that drive.

- **Error 36 ERROR BIT SET IN DRIVE STATUS ERROR BYTE** - The SDI verification test executed an SDI GET STATUS command to the drive under test. The error byte in the returned status was nonzero indicating one of the following conditions:

  1. Drive error
  2. Transmission error
  3. Protocol error
  4. Initialization diagnostic failure
  5. Write lock error

  For the exact meaning of each error, refer to the service manual for that drive.

- **Error 37 ATTENTION SET AFTER SEEK** - The SDI verification routine the SEEK command issued to the drive completed but resulted in an unexpected ATTENTION condition. The drive status is displayed with the error report.

- **Error 38 AVAILABLE NOT SET IN AVAILABLE DRIVE** - The SDI verification routine executed a command to interrogate the Real Time Drive State line of the drive. ILDISK found Available is not set in a drive that should be available.

- **Error 39 ATTENTION NOT SET IN AVAILABLE DRIVE** - The SDI verification routine executed a command to interrogate the Real Time Drive State line of the drive and found Attention is not asserted even though the drive is Available.

- **Error 40 RECEIVER READY NOT SET** - The SDI verification routine executed a command to interrogate the Real Time Drive State line of the drive. The routine expected to find Receiver Ready asserted but it was not.

- **Error 41 READ/WRITE READY SET IN AVAILABLE DRIVE** - The SDI verification routine executed a command to interrogate the Real Time Drive State line of the drive and found Available asserted. However, Read/Write Ready was also asserted. Read/Write Ready should never be asserted when a drive is in the Available state.

- **Error 42 AVAILABLE SET IN ONLINE DRIVE** - The SDI verification routine issued an ONLINE command to the disk drive. Then a command was issued to interrogate the Real Time Drive State line of the drive. The line status indicates the drive is still asserting Available.

- **Error 43 ATTENTION SET IN ONLINE DRIVE** - The SDI verification routine issued an ONLINE command to the drive. The drive entered the Online state, but an unexpected Attention condition was encountered.

- **Error 44 DRIVE CLEAR DID NOT CLEAR ERRORS** - When ILDISK issued a GET STATUS command, error bits were set in the drive response. Issuing a DRIVE CLEAR failed to clear the error bits. The drive is the probable FRU.

- **Error 45 ERROR READING LBN** - As part of Test 14, ILDISK alternates between reading DBNs and LBNs. This tests the drive's ability to seek properly. The error indicates an LBN read failed. The drive is the probable FRU.

- **Error 46 ECHO FRAMING ERROR** - The framing code (upper byte) of an SDI ECHO command response is incorrect. The expected and actual ECHO frames are displayed with the error message. The SDI set and the drive are the probable FRUs.

- **Error 47 K.SDI DOES NOT SUPPORT ECHO** - The disk data channel connected to the drive under test does not support the SDI ECHO command because the disk data channel microcode is not the latest revision level. This is not a fatal error, but the disk data channel microcode should be updated to allow for improved isolation of drive-related errors.

- **Error 48 REQ/PORT NUMBER INFORMATION UNAVAILABLE** - ILDISK was unable to obtain the requestor number and port number from HSC70 disk software tables. The drive may have changed state and disappeared while ILDISK was running. This error can also be caused by inconsistencies in HSC70 software structures.

- **Error 49 DRIVE SPINDLE NOT UP TO SPEED** - ILDISK cannot continue testing the drive because the disk spindle is not up to speed. If the drive is spun down, it must be spun up before ILDISK can completely test the unit. If the drive appears to be spinning, it may be spinning too slowly or the drive may be returning incorrect status information to the HSC70.

- **Error 50 CAN'T ACQUIRE DRIVE STATE AREA** - ILDISK cannot perform the low-level SDI tests, because it cannot acquire the drive state area for the drive. The drive state area is a section of the K Control Area used to communicate with the drive via the SDI interface. To perform the SDI tests ILDISK must take exclusive control of the drive state area; otherwise, the HSC70 operational software may interfere with the tests. The drive state area must be in an inactive state (No interrupts in progress) before it can be acquired by ILDISK. If the drive is rapidly changing its SDI state and generating interrupts, ILDISK may be unable to find the drive in an inactive state.

- **Error 51 FAILURE WHILE UPDATING DRIVE STATUS** - When in the process of returning the drive to the same mode as ILDISK found it originally, an error occurred while performing an SDI GET STATUS command. When a drive is acquired by ILDISK, the program remembers whether the drive was in 576-byte mode or 512-byte mode (reflected by the S7 bit of the mode byte in the drive status). When ILDISK releases the drive (once per pass of the program), the drive mode is returned to the state the drive was in when ILDISK first acquired it. In order to ensure the HSC70 disk functional software is aware of this mode change, ILDISK calls the diagnostic interface routines to perform a GET STATUS to the drive. These routines also update the disk functional software information on the drive to reflect the new mode.

  Error 51 indicates the drive status update failed. The diagnostic interface returns one of three different status codes with this error:

  1. DRIVE ERROR - The GET STATUS command could not be completed due to an error during the command. If informational error messages are enabled (via a SET ERROR INFO command), an error message describing the failure should be printed on the console terminal.

2. BAD UNIT NUMBER - The diagnostic interface could not find the unit number specified. The drive may have spontaneously transitioned to the OFFLINE state (no clocks) since the last ILDISK operation. For this reason, the unit number is unknown when the diagnostic interface tries to do a GET STATUS command.

3. UNKNOWN STATUS FROM DDUSUB - Refer to Error 04.

- **Error 52 576-BYTE FORMAT FAILED** - The program attempted to perform a 576-byte format to the first two sectors of the first track in the R/W DBN area. No errors were detected during the actual formatting operation, but subsequent attempts to read either of the reformatted blocks failed. The specific error detected is identified in the error report.

- **Error 53 512-BYTE FORMAT FAILED** - The program attempted to perform a 512-byte format to the first two sectors of the first track in the R/W DBN area. No errors were detected during the actual formatting operation, but subsequent attempts to read either of the reformatted blocks failed. The specific error detected is identified in the error report.

- **Error 54 INSUFFICIENT RESOURCES TO PERFORM TEST** - This error indicates further testing must be aborted due to lack of required memory structures. To perform certain drive tests ILDISK needs to acquire Timers, a Dialogue Control Block (DCB), Free Control Blocks (FCBs), Data Buffers, and enough control memory to construct two Disk Rotational Access Tables (DRATs). If any of these resources are unavailable, testing cannot be completed. Under normal conditions these resources should always be available.

- **Error 55 DRIVE TRANSFER QUEUE NOT EMPTY BEFORE FORMAT** - ILDISK found a transfer already queued to the K.sdi when the format test began. ILDISK should have exclusive access to the drive at this time, and all previous transfers should have been completed before the drive was acquired. To avoid potentially damaging interaction with some other disk process, ILDISK aborts testing when this condition is detected.

- **Error 56 K.SDI DETECTED ERROR DURING FORMAT** - The K.sdi detected an error during a format operation. Each error bit set in the Fragment Request Block (FRB) is translated into a text message which accompanies the error report.

- **Error 57 WRONG STRUCTURE ON COMPLETION QUEUE** - While formatting, ILDISK checks each structure returned by the K.sdi to ensure the structure was sent to the proper completion queue. An error 57 indicates one of these structures was sent to the wrong completion queue. This type of error indicates a problem with the K.sdi microsequencer or a control memory failure.

- **Error 58 READ OPERATION TIMED-OUT** - To guarantee the disk is on the correct cylinder and track while formatting, ILDISK queues a read operation immediately preceding the format command. The read operation did not complete within 16 seconds indicating the K.sdi is unable to sense sector/index pulses from the disk, or the disk is not in the proper state to perform a transfer. ILDISK aborts the format test following this error report.

- **Error 59 K.SDI DETECTED ERROR IN READ PRECEDING FORMAT** - To guarantee the disk is on the correct cylinder and track while formatting, ILDISK queues a read operation immediately preceding the format command. The read operation failed so ILDISK aborts the format test. Each error bit set in the Fragment Request Block (FRB) is translated into a text message which accompanies the error report.

- **Error 60 READ DRAT NOT RETURNED TO COMPLETION QUEUE** - To guarantee the disk is on the correct cylinder and track while formatting, ILDISK queues a read operation immediately preceding the format command. The read apparently completed successfully, because the Fragment Request Block (FRB) for the read was returned with no error bits set. However the Disk Rotational Access Table (DRAT) for the read operation was not returned indicating a problem with the K.sdi.

- **Error 61 FORMAT OPERATION TIMED-OUT** - The K.sdi failed to complete a format operation. A format operation consists of a read followed by a format. The read completed successfully, but after waiting a 16-second interval the format was not complete. A change in drive state may prevent formatting, the drive may no longer be sending sector/index information to the K.sdi, or the K.sdi may be unable to sample drive state. The format test aborts on this error to prevent damage to the existing disk format.

- **Error 62 FORMAT DRAT WAS NOT RETURNED TO COMPLETION QUEUE** - The K.sdi failed to complete a format operation. A format operation consists of a read followed by a format. The read completed successfully, and the Fragment Request Block (FRB) for the format was returned by the K.sdi with no error indicated. However the Disk

Rotational Access Table (DRAT) for the format operation was never returned indicating a probable K.sdi failure. After reporting this error, the format test aborts.

- **Error 63 CAN'T ACQUIRE SPECIFIED UNIT** - ILDISK was initiated automatically to test a disk drive declared inoperative. When initiated by the disk functional software, ILDISK was given the requestor number, port number, and unit number of the drive to test. ILDISK successfully acquired the drive by unit number, but the requestor and port number of the acquired drive did not match the requestor and port given when ILDISK was initiated. This indicates the HSC is connected to two separate drives with the same unit number plugs. To prevent inadvertent interaction with the other disk drive, ILDISK performs only the low-level SDI tests on the unit specified by the disk functional software. Read/Write tests are skipped because the drive must be acquired by unit number to perform read/write transfers.

- **Error 64 DUPLICATE UNIT DETECTED** - At times during the testing sequence, ILDISK must release, then reacquire, the drive under test. After releasing the drive and reacquiring it, ILDISK noted the requestor and port number of the drive it was originally testing do not match the requestor and port number of the drive just acquired. This indicates the HSC is connected to two separate drives with the same unit number. To prevent inadvertent interaction with the other disk drive, ILDISK discontinues testing if this error is detected.

- **Error 65 FORMAT TESTS SKIPPED DUE TO PREVIOUS ERROR** - To prevent possible damage to the existing disk format, ILDISK does not attempt to format if any errors were detected in the tests preceding the format tests. This error message informs the user that formatting tests will not be performed.

- **Error 66 TESTING ABORTED** - ILDISK was automatically initiated to test a disk drive which was declared inoperative by the disk functional code of the HSC. The disk drive had previously been automatically tested at least twice and somehow was returned to service. Because the tests performed by ILDISK may be causing the inoperative drive to be returned to service, ILDISK does not attempt to test an inoperative drive more than twice. On all succeeding invocations of ILDISK, an Error 66 message prints and ILDISK exits without performing any tests on the drive. This prevents ILDISK from automatically initiating and dropping the drive from the test over and over again.

- **Error 67 NOT ENOUGH GOOD DBNS FOR FORMAT** - In order to guarantee the disk is on the proper cylinder and track,

all formatting operations are immediately preceded by a read operation on the same track where the format is planned. This requires the first track in the drive's R/W DBN area to contain at least one good block which can be read without error. An Error 67 indicates no good block was found on the first track of the R/W DBN area, so the formatting tests are skipped.

5.4.9.1 MSCP Status Codes - ILDISK Error Reports - This section lists some of the MSCP status codes that may appear in ILDISK error reports. All status codes are listed in the octal radix. Further information on MSCP status codes is provided in Appendix C.

007 - Compare Error

010 - Forced Error

052 - SERDES Overrun

053 - SDI Command Timeout

103 - Drive Inoperative

110 - Header Compare or Header Sync Timeout

112 - EDC Error

113 - Controller Detected Transmission Error

150 - Data Sync Not Found

152 - Internal Consistency Error

153 - Position or Unintelligible Header Error

213 - Lost Read/Write Ready

253 - Drive Clock Dropout

313 - Lost Receiver Ready

350 - Uncorrectable ECC Error

353 - Drive Detected Error

410 - One Symbol ECC Error

412 - Data Bus Overrun

413 - State or Response Line Pulse or Parity Error

450 - Two Symbol ECC Error

452 - Data Memory NXM or Parity Error

453 - Drive Requested Error Log

510 - Three Symbol ECC Error

513 - Response Length or Opcode Error

550 - Four Symbol ECC Error

553 - Clock Did Not Restart After Init

610 - Five Symbol ECC Error

613 - Clock Did Not Stop After Init

650 - Six Symbol ECC Error

653 - Receiver Ready Collision

710 - Seven Symbol ECC Error

713 - Response Overflow

750 - Eight Symbol ECC Error

5.4.10 ILDISK Test Summaries

Test summaries for ILDISK follow:

- **TEST 0 - Parameter Fetching** - The part of ILDISK that fetches parameters is identified as Test 0. The user is prompted to supply a unit number and/or a requestor and port number. This part of ILDISK also prompts for the number of passes to perform.

- **TEST 01 - RUN K.SDI Microdiagnostics** - Test 1 commands the disk data channel to execute two of its resident microdiagnostics. If the revision level of the disk data channel microcode is not up to date, the microdiagnostics are not executed. The microdiagnostics executed are the Partial SDI test (K.sdi Test 7)and the SERDES/RSGEN test (K.sdi Test 10).

- **TEST 02 - Check for Clocks and Drive Available** - Test 02 issues a command to interrogate the Real Time Drive State of the drive. This command does not require an SDI exchange, but the real time status of the drive is returned to ILDISK. The real time status should indicate the drive is supplying clocks and the drive should be in the Available state.

- **TEST 03 - Drive Initialize Test** - Test 03 issues an DRIVE INITIALIZE command to the drive under test. This checks both the drive and the Controller Real Time State line of the SDI cable. The drive should respond by momentarily stopping its clock and then restarting it.

- **TEST 04 - SDI Echo Test** - Test 04 first ensures the disk data channel microcode supports the ECHO command. If not, a warning message is issued, and the rest of Test 04 is skipped. Otherwise, the test directs the disk data channel to conduct an ECHO exchange with the drive. An ECHO exchange consists of the disk data channel sending a frame to the drive and the drive returning it. An ECHO exchange verifies the integrity of the Write/Cmd Data and the Read/Res Data lines of the SDI cable.

- **TEST 05 - Run Drive Diagnostics** - Test 05 directs the drive to run its internal diagnostics. The drive is commanded to run a single diagnostic or its entire set of diagnostics depending upon user response to the following prompt:

  Run a Single Drive Diagnostic ?

  Before commanding the drive to run its diagnostics, the drive is brought Online to prevent the drive from giving spurious Available indications to its other SDI port. The drive diagnostics are started when the disk data channel sends a DIAGNOSE command to the drive. The drive does not return a response frame for the DIAGNOSE until it is finished performing diagnostics. This can require two or more minutes. While the disk data channel is waiting for the response frame, ILDISK cannot be interrupted by a CTRL Y.

- **TEST 06 - Disconnect From Drive** - Test 06 sends a DISCONNECT command to the drive and then issues a GET LINE STATUS internal command to the K.sdi to ensure the drive is in the Available state. The test also expects Receiver Ready and Attention are set in drive status and Read/Write Ready is not set.

- **TEST 07 - Check Drive Status** - Test 07 issues a GET STATUS command to the drive to check that none of the drive's error bits are set. If any error bits are set, they are reported and the test issues a DRIVE CLEAR command to clear the error bits. If the error bits fail to clear, an error is reported.

- **TEST 08 - Drive Initialize** - Test 08 issues a command to interrogate the Real Time Drive State of the drive. The test then issues a DRIVE INITIALIZE command to ensure the previous DIAGNOSE command did not leave the drive in an undefined state.

- **TEST 09 - Bring Drive Online** - Test 09 issues an ONLINE command to the drive under test. Then a GET LINE STATUS command is issued to ensure the drive's real time state is proper for the Online state. Read/Write Ready is expected to be true; Available and Attention are expected to be false.

- **TEST 10 - Recalibrate and Seek** - Test 10 issues a RECALIBRATE command to the drive. This ensures the disk heads start from a known point on the media. The a SEEK command is issued to the drive, and the drive's real time status is checked to ensure the SEEK did not result in an Attention condition. Then another RECALIBRATE command is issued returning the heads to a known position.

- **TEST 11 - Disconnect From Drive** - Test 10 issues a DISCONNECT command to return the drive to the Available state. Then the drive's real time status is checked to ensure Available, Attention and Receiver Ready are true and Read/Write Ready is false.

- **TEST 12 - Bring Drive Online** - Test 12 attempts to bring the disk drive to the Online state. Test 12 is only executed for drives known to the HSC70 disk functional software. Test 12 consists of the following steps:

  1. GET STATUS - ILDISK issues an SDI GET STATUS command to the disk drive.

  2. ONLINE - ILDISK directs the HSC70 Diagnostic Interface to bring the drive Online.

  If the GET STATUS and the ONLINE commands succeed, ILDISK proceeds to Test 13. If the GET STATUS and the ONLINE commands fail, ILDISK goes directly to Test 17 (Termination). Note the Online is performed via the HSC70 diagnostic interface, invoking the same software operations a host invokes to bring a drive Online. An Online at this level constitutes more than just sending a SDI ONLINE command. The FCT and RCT of the drive are also read and certain software structures are modified to indicate the new state of the drive. If the drive is unable to read data from the disk media, the Online operation fails. If Test 12 fails, ILDISK skips the remaining tests and goes to Test 17.

- **TEST 13 - Read Only I/O Operations Test** - Test 13 tests that all R/W heads in the drive can seek and properly locate a sector on each track in the drive Read Only DBN space. (DBN space is an area on all disk media devoted to diagnostic use.) Test 13 attempts to read at least one sector on every track in the Read Only area of the

drives DBN space. The sector is checked to ensure it contains the proper data pattern. Bad sectors are allowed, but there must be at least one good sector on each track in the Read Only area. After each successful DBN read, ILDISK reads one LBN to further enhance seek testing. This ensures the drive can successfully seek to and from the DBN area from the LBN area of the disk media. ILDISK proceeds to Test 16 and Test 13 completes.

- **TEST 14 - Format 576-Byte Mode** - This test is not yet implemented.

- **TEST 15 - Format 512-Byte Mode** - This test is not yet implemented.

- **TEST 16 - I/O Operations Test (Read/Write)** - Test 16 checks to see if the drive can successfully write a pattern and read it back from at least one sector on every track in the drive Read/Write DBN area. (Read/Write DBN space is an area on every disk drive devoted to diagnostic Read/Write testing.) Bad sectors are allowed, but at least one sector on every track in the Read/Write area must pass the test. After Test 16 completes, ILDISK proceeds to Test 17.

- **TEST 17 - Terminate ILDISK** - Test 17 is the ILDISK termination routine. The following steps are performed:

  1. If the drive is unknown to the HSC70 disk functional software, or if the SDI verification test failed, proceed to step 5 of this test.

  2. An SDI CHANGE MODE command is issued to the drive. The CHANGE MODE command directs the drive to disallow access to the DBN area and changes the sector size (512 or 576 bytes) back to its original state.

  3. The drive is released from exclusive diagnostic use. This returns the drive to the Available state.

  4. The drive is reacquired for exclusive diagnostic use. This is to allow looping if more than one pass is selected.

  5. If more passes are left to perform, the test is reinitiated.

  6. If no more passes are left to perform, ILDISK releases the drive, returns all structures acquired, and terminates.

5.5 INLINE TAPE TEST (ILTAPE)

ILTAPE initiates tape formatter resident diagnostics or a functional test of the tape transport. In addition, the test permits selection of a full test of the K.sti interface. When a full interface test is selected, the K.sti microdiagnostics are executed, line state is verified, an ECHO test is performed, and a default set of formatter diagnostics is executed. See the DRIVE UNIT NUMBER prompt in Section 5.5.3 for information on initiating a full test. Detected failures result in fault isolation to the FRU level.

See Section 5.5.9 for a summary of three types of tape transport tests listed below:

- Fixed canned sequence
- User sequence supplied at the terminal
- Fixed streamer sequence.

5.5.1 ILTAPE System Requirements

The following hardware and software are necessary to run ILTAPE.

Hardware requirements include:

- HSC70 subsystem with K.sti
- STI compatible tape formatter
- TA78 tape drive (for transfer commands only)
- Console terminal
- RX33 disk drive or equivalent

The I/O control processor, Program memory, and Control memory must be working.

Software requirements include:

- CRONIC
- DEMON
- K.sti microcode
- Tape Functional Code (TFUNCT)
- Diagnostic/Utility Interface (TDUSUB)

5.5.2 ILTAPE Operating Instructions

The following steps outline the procedure for running ILTAPE. The test assumes an HSC70 is configured with a terminal and STI interface. If the HSC70 is not booted, start with step 1. If the HSC70 is already booted, proceed to step 2.

1. Boot the HSC70

   Press the INIT button on the HSC70 OCP of the. The following message should appear at the terminal:

   INIPIO-I Booting...

   The boot process takes about one minute, and then the following message should appear at the terminal:

   HSC Version xxxx Date Time System n

2. Type **CTRL Y**

   This causes the KMON prompt

   HSC70>

3. Type **R DXn:ILTAPE**

This invokes the inline tape diagnostic program, ILTAPE. The DX in step 3 is the RX33 device name. The n refers to the unit number of the specific RX33 drive. For example, DX1: refers to RX33 drive number one. The following message should appear at the terminal:

ILTAPE>D>hh:mm Execution Starting

5.5.3 ILTAPE/User Dialogue

The following paragraphs describe ILTAPE/user dialogue during execution of ILTAPE. Note default values for input parameters appear within the brackets of the prompt. The absence of a value within the brackets indicates the input parameter is not defaultable.

DRIVE UNIT NUMBER (U) []?

If you want to run formatter diagnostics or transport tests, enter Tnnn, where nnn is the MSCP unit number (such as T316).

If you want a full interface test, enter Xm (where m is any number). Typing X, instead of T, requires a requestor number and slot number. The following two prompts solicit requestor/slot numbers.

ENTER REQUESTOR NUMBER (2-9) []?

Enter the requestor number. The range includes numbers two through nine, with no default value.

ENTER PORT NUMBER (0-3) []?

Enter the port number. The port number must be zero, one, two, or three with no default value. After this prompt is answered, ILTAPE executes the K.sti interface test.

EXECUTE FORMATTER DIAGNOSTICS (YN) [Y]?

Enter Y (for yes) if you want to execute formatter diagnostics. This is the default. Enter N if you do not want to run formatter diagnostics.

MEMORY REGION NUMBER (H) [0]?

This prompt appears only if the response to the previous prompt was Y. A formatter diagnostic is named according to the formatter memory region where it executes. Enter the memory region (hexadecimal) in which the formatter diagnostic is to execute. ILTAPE continues at the prompt for iterations. Refer to the appropriate tape drive service manual for more information on formatter diagnostics.

EXECUTE TEST OF TAPE TRANSPORT (YN) [N]?

To test the tape transport, enter Y (the default is N). If no transport testing is desired, the dialogue continues with the ITERATIONS prompt. Otherwise, the following prompts appear.

IS MEDIA MOUNTED (YN) [N]?

This test writes to the tape transport, requiring a mounted scratch tape. Enter Y if a scratch tape is already mounted.

FUNCTIONAL TEST SEQUENCE NUMBER (D) [1]?

You may select one of five transport tests. The default is 1 (the canned sequence). Enter 0 if a new user sequence will be input from the terminal. Enter 2, 3 or 4 to select a user sequence previously input and stored on the RX33 diskette. User sequences are described in Section 5.5.4. Enter 5 to select the streaming sequence.

INPUT STEP 00:

This prompt appears only if the response to the previous prompt was 0. See Section 5.5.4 for a description of user sequences.

ENTER CANNED SEQUENCE RUN TIME IN MINUTES (D) [1]?

Answering this prompt determines the time limit for the canned sequence. It appears only if the canned sequence is selected. Enter the total run time limit in minutes. The default is one minute.

SELECT DENSITY (0=ALL, 1=1600, 2=6250) [0]?

This prompt permits selection of the densities used during the canned sequence. It appears only if the canned sequence is selected. One or all densities may be selected; the default is ALL.

SELECT DENSITY (1=800, 2=1600, 3=6250) [3]?

This prompt appears only if a user-defined test sequence was selected. The prompt permits selection of any one of the possible tape densities. The default density is 6250 bpi.

Enter 1, 2, or 3 to select the desired tape density.

1 = 800 bpi

2 = 1600 bpi

3 = 6250 bpi

The next series of prompts concern speed selection. The particular prompts depend upon the type of speeds supported, fixed or variable. ILTAPE determines the speed types supported and prompts accordingly.

If fixed speeds are supported, ILTAPE displays a menu of supported speeds, as follows:

Fixed Speeds Available:

(1) sss1 IPS

(2) sss2 IPS

. . .

(n) sssn IPS,

where sssn is a supported speed in inches per second. The maximum number of supported speeds is four. Thus, n cannot be greater than four. The prompt for a fixed speed is:

SELECT FIXED SPEED (D) [1]?

To select a fixed speed, enter a digit (n) corresponding to one of the above displayed speeds. The default is the lowest supported speed. ILTAPE continues at the data pattern prompt.

If variable speeds are supported, ILTAPE displays the lower and upper bounds of the supported speeds as follows:

VARIABLE SPEEDS AVAILABLE:

LOWER BOUND = lll IPS

UPPER BOUND = uuu IPS

where lll is the lower bound and uuu is the upper bound of supported speeds. The prompt for a variable speed is:

SELECT VARIABLE SPEED (D) [0 = LOWEST]?

To select a variable speed, enter a number within the bounds, inclusively, of the displayed supported variable speeds. The default is the lower bound.

NOTE

If only a single speed is supported, ILTAPE does not prompt for speed. It runs at the single speed supported.

DATA PATTERN NUMBER (D) [3]?

Choose one of five data patterns.

0 - User supplied

1 - All zeros

2 - All ones

3 - Ripple zeros

4 - Ripple ones

The default is three. If the response is zero, the following prompts appear.

HOW MANY DATA ENTRIES (D) []?

Enter the number of unique words in the data pattern. Up to 16 (10) words are permitted.

DATA ENTRY (H) []?

Enter the data pattern word in hexadecimal, for example, ABCD. This prompt repeats until the all data words specified in the previous prompt are exhausted.

SELECT RECORD SIZE (GREATER THAN OR EQUAL TO 1) (D) [8192]?

Enter the desired record size in decimal bytes. The default is 8192 bytes.

NOTE

This prompt does not appear if streaming is selected.

ITERATIONS (D) [1]?

Enter the number of times the selected tests are to run. After the number of iterations is entered, the selected tests begin execution. Errors encountered during execution cause display of appropriate messages at the terminal.

5.5.4 ILTAPE User Sequences

In order to test/exercise a tape transport, write a sequence of commands at the terminal. This sequence may be saved on the RX33 diskette and be recalled for execution at a later time. Up to three user sequences can be saved on the RX33.

Following is a list of supported user sequence commands:

WRT	Write one record
RDF	Read one record forward
RDFC	Read one record forward with compare
RDB	Read one record backward
RDBC	Read one record backward with compare
FSR	Forward space one record
FSF	Forward space one file
BSR	Backspace one record
BSF	Backspace one file
REW	Rewind
RWE	Rewind with erase
UNL	Unload (after rewind)
WTM	Write tape mark
ERG	Erase gap

Cnnn	Counter set to nnn (0 = 1000.)
Dnnn	Delay nnn ticks (0 = 1000.)
BRnn	Branch unconditionally to step nn
DBnn	Decrement counter and branch if nonzero to step nn
TMnn	Branch on Tape Mark to step nn
NTnn	Branch on no Tape Mark to step nn
ETnn	Branch on EOT to step nn
NEnn	Branch on not EOT to step nn
QUIT	Terminate input of sequence steps

Typing 0 in response to the prompt

FUNCTIONAL TEST SEQUENCE NUMBER (D) [1]?

initiates the user sequence dialogue. The following paragraphs describe the ILTAPE/user dialogue during a new user sequence.

INPUT STEP nn

Enter one of the user sequence commands listed previously. ILTAPE keeps track of the step numbers and automatically increments them. Up to 50 steps may be entered. Typing QUIT in response to the prompt

INPUT STEP nn

terminates the user sequence. At that time, the following prompt appears:

STORE SEQUENCE AS SEQUENCE NUMBER (0,2,3,4) [0]?

The sequence entered at the terminal may be stored on the RX33 in one of three files. To select one of these files, type **2**, 3 or **4**. Once stored, the sequence may be recalled for execution at a later time by referring to the appropriate file (typing 2, 3 or 4) in response to the sequence number prompt.

Typing 0 (the default) indicates the user sequence just entered should not be stored. In this case, the sequence cannot be run at a later time.

An example of entering a user sequence follows:

```
INPUT STEP  00  REW     ;Rewind the tape

INPUT STEP  01  C950    ;Set counter to 950
```

```
INPUT STEP  02  WRT     ;Write one record

INPUT STEP  03  ET07    ;If EOT branch to step 7

INPUT STEP  04  RDB     ;Read backward one record

INPUT STEP  05  FSR     ;Forward space one record

INPUT STEP  06  DB02    ;Decrement counter, branch

                        ;To step 2 if nonzero

INPUT STEP  07  REW     ;Rewind the tape

INPUT STEP  08  QUIT    ;Terminate sequence input

STORE SEQUENCE AS SEQUENCE NUMBER (0,2,3,4) [0]? 3
```

This sequence writes a record, reads it backwards and skips forward over it. If an EOT is encountered prior to writing 950 records, the tape is rewound and the sequence terminates. Note, the sequence is saved on the RX33 as sequence number 3 and can be recalled at a later execution of ILTAPE.

5.5.5 ILTAPE Progress Reports

When transport testing is finished, a summary of soft errors appears on the terminal upon completion of the test. The format of this summary is:

```
SOFT ERROR SUMMARY:     READ   WRITE  COMPARE

                        xxxxxx xxxxxx xxxxxx
```

Successful completion of a formatter diagnostic is indicated by the following message on the terminal:

TEST nnnn DONE

where nnnn is the formatter diagnostic test number.

When an error is encountered, an appropriate error message is printed on the terminal.

5.5.6 ILTAPE Test Termination

ILTAPE terminates normally after the selected tests successfully complete. The program also terminates after typing a CTRL Y or CTRL C at any time. Further, certain errors cause ILTAPE to terminate automatically.

5.5.7 ILTAPE Error Message Example

ILTAPE conforms to the diagnostic generic error message format (Section 5.1.1.1). An example of an ILTAPE error message follows:

```
ILTAPE>D>09:31 T 011 U-T00101
ILTAPE>D>COMMAND FAILURE
ILTAPE>D>MSCP WRITE MULTIPLE COMMAND
ILTAPE>D>MSCP STATUS:   000000
ILTAPE>D>POSITION    001792
```

The test number reflects the state level where ILTAPE is executing when an error occurs. This number does not indicate a separate test that can be called. Test levels are defined as follows:

Test Number	ILTAPE State
0	Initialization of tape software interface
1	Device (port, formatter, unit) acquisition
2	STI interface test in execution
3	Formatter diagnostics executing in response to Diagnostic Request (DR) bit
4	Tape transport functional test
5	User-selected formatter diagnostics executing
6	Termination and clean-up

The optional text is dependent upon the type of error.

5.5.8 ILTAPE Error Messages

The following list describes ILTAPE error messages.

- **Error 1 - INITIALIZATION FAILURE** - Tape path software interface cannot be established due to insufficient resources (buffers, queues, timers, etc..)
- **Error 2 - SELECTED UNIT NOT A TAPE** - Selected drive is not known to the HSC as a tape.
- **Error 3 - INVALID REQUESTOR/PORT NUMBER** - Selected requestor number or port number is out of range or port

selected is not known to the system.

- **Error 4 - REQUESTOR NOT A K.STI** - Selected requestor is not known to the system as a tape data channel.

- **Error 5 - TIMEOUT ACQUIRING DRIVE SERVICE AREA** - While attempting to acquire the Drive Service Area (port) in order to run the STI interface test, a timeout occurred. If this happens, the tape functional code is corrupted. ILTAPE invokes a system crash.

- **Error 6 - REQUESTED DEVICE UNKNOWN** - Device requested is not known to the tape subsystem.

- **Error 7 - REQUESTED DEVICE IS BUSY** - Selected device is Online to another controller or host.

- **Error 8 - UNKNOWN STATUS FROM TAPE DIAGNOSTIC INTERFACE** - interface An unknown status was returned from the diagnostic software interface, TDUSUB.

- **Error 9 - UNABLE TO RELEASE DEVICE** - Upon termination of ILTAPE or upon an error condition, the device(s) could not be returned to the system.

- **Error 10 - LOAD DEVICE WRITE ERROR - CHECK IF WRITE LOCKED** - An error occurred while attempting to write a user sequence to the RX33. Check to see if the RX33 diskette is write protected. You are reprompted for a user sequence number. To break the loop of reprompts, type **CTRL Y**.

- **Error 11 - COMMAND FAILURE** - A command failed during execution of ILTAPE. The command in error may be one of several types such as an MSCP or Level 2 STI command. The failing command is identified in the optional text of the error message. For example,

  ILTAPE>D>MSCP READ COMMAND

  ILTAPE>D>MSCP STATUS: nnnnnn

- **Error 12 - READ MEMORY BYTE COUNT ERROR** - The requested byte count used in the read (formatter) memory command is different from the actual byte count received.

  EXPECTED COUNT: xxxx ACTUAL COUNT: yyyy -

- **Error 13 - FORMATTER DIAGNOSTIC DETECTED ERROR** - A diagnostic running in the formatter detects an error. Any error text from the formatter is displayed.

- **Error 14 - FORMATTER DIAGNOSTIC DETECTED FATAL ERROR** - A diagnostic running in the formatter detects a fatal

error. Any error text from the formatter is displayed.

- **Error 15 - RX33 READ ERROR** - While attempting to read a user sequence from the RX33, a read error was encountered. Ensure a sequence has been stored on the RX33 as identified by the user sequence number. The program reprompts for a user sequence number. To break the loop of reprompts, type a **CTRL Y**.

- **Error 16 - INSUFFICIENT RESOURCES TO ACQUIRE SPECIFIED DEVICE** - During execution, ILTAPE was unable to acquire the specified device due to a lack of necessary resources. This condition is identified to ILTAPE by the tape functional code via the diagnostic interface, TDUSUB. ILTAPE has no knowledge of the specific unavailable resource.

- **Error 17 - K MICRODIAGNOSTIC DID NOT COMPLETE** - During the STI interface test, the requestor microdiagnostic timed out.

- **Error 18 - K MICRODIAGNOSTIC REPORTED ERROR** - During the STI interface test, an error condition was reported by the K microdiagnostics.

- **Error 19 - DCB NOT RETURNED, K FAILED FOR UNKNOWN REASON** - During the STI interface test, the requestor failed for an undetermined reason and the Diagnostic Control Block (DCB) was not returned to the completion queue.

- **Error 20 - IN DCB UPON COMPLETION** - During the STI interface test, an error condition was returned in the DCB.

- **Error 21 - UNEXPECTED ITEM ON DRIVE SERVICE QUEUE** - During the STI interface test, an unexpected entry was found on the drive service queue.

- **Error 22 - STATE LINE CLOCK NOT RUNNING** - During the STI interface test, execution of an internal command to interrogate the Real Time Formatter State line of the drive indicated the state line clock is not running.

- **Error 23 - INIT DID NOT STOP STATE LINE CLOCK** - During the STI interface test, after execution of a formatter INITIALIZE command, the state line clock did not drop for the time specified in the STI specification.

- **Error 24 - STATE LINE CLOCK DID NOT START UP AFTER INIT** - During the STI interface test, after execution of a formatter INITIALIZE command, the state line clock did not start up within the time specified in the STI specification.

- **Error 25 - FORMATTER STATE NOT PRESERVED ACROSS INIT** - The state of the formatter prior to a formatter initialize was not preserved across the initialization sequence.

- **Error 26 - ECHO DATA ERROR** - Data echoed across the STI interface was incorrectly returned.

- **Error 27 - RECEIVER READY NOT SET** - After issuing an ONLINE command to the formatter, the Receiver Ready signal was not asserted.

- **Error 28 - AVAILABLE SET IN ONLINE FORMATTER** - After successful completion of a formatter ONLINE command to the formatter, the Available signal is set.

- **Error 29 - RX33 ERROR - FILE NOT FOUND** - During the user sequence dialogue, ILTAPE was unable to locate the sequence file associated with the specified user sequence number. Check that a RX33 system diskette is properly installed. The program reprompts for a user sequence number. To break the loop of reprompts, type a **CTRL Y**.

- **Error 30 - DATA COMPARE ERROR** - During execution of the user or canned sequence, ILTAPE encountered a software compare mismatch on the data written and read back from the tape. The software compare is actually carried out by a subroutine in the diagnostic interface, TDUSUB. The results of the compare are passed to ILTAPE. Information in the text of the error message identifies the data in error.

- **Error 31 - EDC ERROR** - During execution of the user or canned sequence, ILTAPE encountered an EDC error on the data written and read back from the tape. This error is actually detected by the diagnostic interface, TDUSUB and reported to ILTAPE. Information in the text of the error message identifies the data in error.

- **Error 32 - INVALID MULTIUNIT CODE FROM GUS COMMAND** - After a unit number is input to ILTAPE and prior to acquiring the unit, ILTAPE attempts to obtain the unit's multiunit code via the GET UNIT STATUS command. This error indicates a multiunit code of zero was returned to ILTAPE from the tape functional code. Because a multiunit code of zero is invalid, this error is equivalent to a device unknown to the tape subsystem.

- **Error 33 - INSUFFICIENT RESOURCES TO ACQUIRE TIMER** - ILTAPE was unable to acquire a timer from the system; insufficient buffers are available in the system to allocate timer queues.

- **Error 34 - UNIT UNKNOWN OR ONLINE TO ANOTHER CONTROLLER** - The device identified by the selected unit number is either unknown to the system, or it is online to another controller. Verify the selected unit number is correct and run ILTAPE again.

5.5.9 ILTAPE Test Summaries

Summaries of the tests contained in this diagnostic follow.

5.5.9.1 K.sti Interface Test Summary - This portion of ILTAPE tests the STI interface of a specific tape data channel and port. It also performs low-level testing of the formatter by interfacing to the K.sti Drive Service Area (port) and executing various level 2 STI commands. The testing is limited to dialogue operations; no data transfer is done. The operations performed are DIAGNOSE, READ MEMORY, GET DRIVE STATUS, and READ LINE STATUS.

K.sti microdiagnostics are executed to verify the tape data channel. A default set of formatter diagnostics (out of memory region 0) is executed to test the formatter, and an echo test is performed to test the connection between the port and the formatter.

Failures detected are isolated to the extent possible and limited to tape data channel, the STI set, or the formatter. The STI set includes a small portion of the K.sti module, and the entire STI (all connectors and cables and a small portion of the drive). The failure probabilities of the STI set are:

1. STI cables or connectors (most probable)
2. Formatter
3. K.sti (least probable)

When the STI set is identified as the FRU, replacement should be in the order indicated in the preceding list.

5.5.9.2 Formatter Diagnostics Test Summary - Formatter diagnostics are executed out of a formatter memory region selected by the user. Refer to the particular tape drive service manual (for example, TA78 Magnetic Tape Drive Service Manual) for a description of the formatter tests. Failures detected identify the formatter as the FRU.

5.5.9.3 **User Sequences Test Summary** - User sequences are used to exercise the tape transport. The particular sequence is totally user-defined. Refer to Section 5.5.4.

5.5.9.4 **Canned Sequence Test Summary** - The canned sequence is a fixed routine for exercising the tape transport. The canned sequence first performs a quick verify of the ability to read and write tape at all supported densities. Using a user-selected record size, it then writes, reads, and compares the data written over a 200-foot length of tape. Positioning over this length of tape is also performed. Finally, random record sizes are used to write, read, compare, and position over a 50-foot length of tape. Errors encountered during the canned sequence are reported at the terminal.

5.5.9.5 **Streaming Sequence Test Summary** - The streaming sequence is a fixed sequence which attempts to write and read the tape at speed (without hesitation). The entire tape is written, the tape is rewound, and the entire tape is read back. Execution may be terminated at any time by typing CTRL Y.

NOTE

In reading the tape, ILTAPE uses the ACCESS command. This allows the tape to move at speed. This is necessary because of the buffer size restrictions existing for diagnostic programs.

5.6 INLINE TAPE COMPATABILITY TEST (ILTCOM)

ILTCOM tests the compatibility of tapes which may have been written on different systems and different drives with STI compatible (TA78) drives connected to an HSC via the STI bus. ILTCOM may generate, modify, read, or list a compatibility tape. Data read from the compatibility tape is compared to the expected pattern. A compatibility tape consists of groups of files, called bunches, of records of specific data patterns. The format of the compatibility tape is described in following paragraphs.

Each bunch contains a header record and several data records of different sizes. Each bunch is terminated by a Tape Mark. The last bunch on a tape is followed by an additional Tape Mark (thus forming logical EOT).

Each bunch contains a total of 199 records: one header record followed by 198 data records. The header record contains 48 (decimal) bytes of 6 bit-encoded descriptive information as follows:

Table 5-1 ILTCOM Header Record

Field	Description	Length	Example
1	Drive type	6 Bytes	TA78
2	Drive Serial Number	6 Bytes	123456
3	Processor Type	6 Bytes	HSC-70
4	Processor Ser. No.	6 Bytes	123456
5	Date	6 Bytes	093083
6	Comment *	18 Bytes	Comment

* ILTCOM can read but cannot generate a comment field.

The data records are arranged as follows:

- Sixty-six records 24 (decimal) bytes in length. These records sequence through 33 different data patterns. The 1st and 34th records contain pattern 1, the 2nd and 35th records contain pattern 2, ..., the 33rd and 66th records contain pattern 33.

- Sixty-six records 528 (decimal) bytes in length. These records sequence through the 33 data patterns as described above.

- Sixty-six records 12,024 (decimal) bytes in length. These records sequence through the 33 data patterns in the same manner as the preceding ones.

The data patterns used are shown below:

Table 5-2 ILTCOM Data Patterns

Pattern Number	Pattern	Description
1	377	Ones
2	000	Zeros
3	274,377,103,000	Peak shift
4	000,377,377,000	Peak shift
5	210,104,042,021	Floating one
6	273,167,356,333	Floating zero
7	126,251	Alternate bits
8	065,312	Square pattern
9	000,377	Alternate frames
10	001	Track 0 on
11	002	Track 1 on
12	004	Track 2 on
13	010	Track 3 on
14	020	Track 4 on
15	040	Track 5 on
16	100	Track 6 on
17	200	Track 7 on
18	376	Track 0 off
19	375	Track 1 off
20	373	Track 2 off
21	367	Track 3 off
22	357	Track 4 off
23	337	Track 5 off
24	277	Track 6 off
25	177	Track 7 off
26	207,377,370,377	Bit peak shift
27	170,377,217,377	
28	113,377,264,377	
29	035,377,342,377	
30	370,377,207,377	
31	217,377,170,377	
32	264,377,113,377	
33	342,377,035,377	

5.6.1 ILTCOM System Requirements

The following hardware and software are necessary to run ILTCOM. Hardware requirements include HSC subsystem with K.sti, STI-compatible tape formatter and drive. Because ILTCOM is not diagnostic in nature, all of the necessary hardware is assumed to be working. Errors are detected and reported but fault isolation is not a goal of ILTCOM. Software requirements include CRONIC, DEMON, K.sti microcode, TFUNCT (Tape Functional Code), TDUSUB (Diagnostic/Utility Interface).

5.6.2 ILTCOM Operating Instructions

The following steps outline the procedure for running ILTCOM. ILTCOM assumes the HSC is configured with a terminal, STI interface, and a TA78 tape drive (or STI compatible equivalent). If the HSC is already booted, proceed to step 2 below. If the HSC needs to be booted, start with step 1.

1. Boot the HSC

   Press the INIT button on the OCP of the HSC. The following message should appear at the terminal:

   INIPIO-I Booting...

   The boot process takes about one minute, and then the following message should appear at the terminal:

   HSC70 Version xxxx Date Time System n

2. Type **CTRL Y**

   This causes the KMON prompt,

   HSC70>

3. Type **R DXn:ILTCOM**, where n equals the number of the RX33 drive containing the system diskette.

   This invokes the compatibility test program, ILTCOM. The following message should appear at the terminal:

   ILTCOM>D>hh:mm Execution Starting

   The subsequent program dialogue is described in the next section.

5.6.3 ILTCOM Test Parameter Entry

ILTCOM allows the writing, reading, listing, or modifying of compatibility tapes. The following describes the user dialogue during the execution of ILTCOM.

DRIVE UNIT NUMBER (U) []?

Enter the tape drive MSCP unit number (such as T21)

SELECT DENSITY FOR WRITES (1600, 6250) []?

Enter the write density by typing (up to) 4 characters of the density desired (1600 or 1 for 1600 bpi).

SELECT FUNCTION (WR=WRITE,REA=READ,ER=ERASE,
LI=LIST,REW=REWIND,EX=EXIT) []?

Enter the function by typing up to four characters which uniquely identify the desired function (for instance, READ or REA for read).

The subsequent dialogue is dependent upon the function selected.

WRITE -

The write function writes new bunches on the compatibility tape. Bunches are either written one at a time or over the entire tape. Bunches are written from the current tape position. If the write function is selected, the following prompts occur.

PROCEED WITH INITIAL WRITE (YN) [N]?

Type **Y** or **N** (for yes or no, respectively.) The default is no, in which case program control is continued at the function selection prompt. If the response is yes, the following prompt ocurrs:

WRITE ENTIRE TAPE (YN) [N]?

Type **Y** (for yes), if the entire tape is to be written. Writing of bunches begins at the current tape position and continues to physical EOT (end-of-tape). Type **N** (for no), which is the default, if the entire tape is not to be written. In this case, only one bunch is written from the current tape position. This prompt only appears on the initial write selection. After the bunch(es) have been written, control continues at the function selection prompt.

READ -

The read function reads and compares the data in the bunches with an expected (predefined) data pattern. As the reads occur, the bunch header information is displayed at the terminal. The format of the display is shown in the following example:

```
BUNCH   01 WRITTEN BY TA78  SERIAL NUMBER 002965
ON A HSC70 SERIAL NUMBER 005993 ON 09-18-84
```

The number of bunches to be read is user selectable. All reads are from BOT. If the read function is selected, the following prompt appears:

READ HOW MANY BUNCHES (D) [0=ALL]?

Type the number of bunches to be read. The default (0) causes all bunches to be read. After the requested number of bunches have been read and compared, control continues at the function selection prompt.

LIST -

The list function reads and displays the header of each bunch on the compatibility tape from BOT. The display is the same as the one described under the READ function. The data contents of the bunches are NOT read and compared. After listing the tape bunch headers, control continues at the function selection prompt.

ERASE -

The erase function erases a user-specified number of bunches from the current tape position toward BOT. ILTCOM backs up the specified number of tape marks and writes a second tape mark (logical EOT). This effectively erases the specified number of bunches from the tape. Thus, for example, if the current tape position is at bunch 5 and the user wishes to erase two bunches, three bunches are left on the tape after the ERASE command completes.

If the erase function is selected, the following prompt appears at the terminal:

```
ERASE HOW MANY BUNCHES FROM CURRENT POSITION (D) [0]?
```

Type the number of bunches to be erased. The default of zero results in no change in tape contents or position. Control continues at the function selection prompt.

REWIND -

The rewind function rewinds the tape to BOT.

EXIT -

The exit function rewinds the tape and exits the tape compatibility program, ILTCOM.

5.6.4 ILTCOM Test Termination

ILTCOM is terminated normally by selecting the exit function (EXIT) or by typing a CTRL Y or CTRL C. Further, certain errors which occur during execution cause ILTCOM to terminate automatically.

5.6.5 ILTCOM Error Message Example

ILTCOM conforms to the diagnostic generic error message format (Section 5.1.1.1). An example of an ILTCOM error message follows:

```
ILTCOM>D>09:29 T 000 E 003 U-T00100
ILTCOM>D>COMMAND FAILURE
```

where:

E nnn is an error number

U-Txxxxx indicates the Tape MSCP Unit Number

The optional text is dependent upon the type of error. Some error messages contain the term, object count, in the optional text. Object count refers to tape position (in objects) from BOT.

5.6.5.1 **ILTCOM Error Messages** - The following are the ILTCOM error messages.

- **Error 1 - INITIALIZATION FAILURE** - Tape path cannot be established due to insufficient resources.

- **Error 2 - SELECTED UNIT NOT A TAPE** - User selected a drive not known to system as a tape.

- **Error 3 - COMMAND FAILURE** - A command failed during execution of ILTCOM. The command in error may be one of several types (MSCP level, STI level 2, etc.). The failing command is identified in the optional text of the error message. For example,

  ```
  ILTCOM>D>tt:tt T 000 E 003 U-T00030
  ILTCOM>D>COMMAND FAILURE
  ILTCOM>D>MSCP READ COMMAND
  ILTCOM>D>MSCP STATUS:  nnnnnn
  ```

- **Error 5 - SPECIFIED UNIT NOT AVAILABLE** - The selected unit is Online to another controller.

- **Error 6 - SPECIFIED UNIT CANNOT BE BROUGHT ONLINE** - The selected unit is offline or not available.

- **Error 7 - SPECIFIED UNIT UNKNOWN** - The selected unit is unknown to the HSC configuration.

- **Error 8 - UNKNOWN STATUS FROM TDUSUB** - An unknown error condition returned from the software interface, TDUSUB.

- **Error 9 - ERROR RELEASING DRIVE** - After completion of execution or after an error condition, the tape drive could not be successfully returned to the system.

- **Error 10 - CAN'T FIND END OF BUNCH** - The compatibility tape being read or listed has a bad format.

- **Error 11 - DATA COMPARE ERROR** - A data compare error has been detected. The actual and expected data are displayed in the optional text of the error message. For example,

```
ILTCOM>D>tt:tt T 000 E 011 U-T00030
ILTCOM>D>DATA COMPARE ERROR
ILTCOM>D>EXPECTED DATA: XXXXXX  ACTUAL DATA: YYYYYY
ILTCOM>D>NUMBER OF FIRST WORD IN ERROR:  nnnnn
ILTCOM>D>NUMBER OF WORDS IN ERROR:  mmmmm
ILTCOM>D>OBJECT COUNT = cccccc
```

- **Error 12 - DATA EDC ERROR** - An EDC error was detected. Expected and actual values are displayed in the optional text of the error message.

5.6.6 ILTCOM Test Summaries

ILTCOM writes, reads, and compares compatibility tapes upon user selection. The testing that takes place looks for compatibility of tapes written on different drives (and systems). As incompatibilities are found, due to data compare errors or unexpected formats, they are reported. ILTCOM makes no attempt to isolate faults during execution; it merely reports incompatibilities and other errors as they occur.

5.7 INLINE MULTIDRIVE EXERCISER (ILEXER)

The Inline Multidrive Exerciser exercises the various disk drives and tape drives attached to the HSC subsystem. The exerciser is initiated upon demand. Drives to be tested are selected by the operator. The exerciser will issue random READ, WRITE, and COMPARE commands to exercise the drives. The results of the exerciser are displayed on the terminal from which it was initiated. The reports given by ILEXER do not provide any analysis of the errors reported nor explicitly call out a specific FRU. This is strictly an exerciser.

This exerciser runs with other processes on the HSC subsystem. It is loaded from the RX33 and uses the services of DEMON (Diagnostic Execution Monitor) and the HSC control software. Bad block replacement is disabled for any disk unit being exercised.

CAUTION

Do not run ILEXER through DUP.

5.7.1 ILEXER System Requirements

In order for this program to run, the following hardware and software items must be available:

1. HSC subsystem including:

   a. Console terminal

b. P.io

c. K.sdi and/or K.sti

d. Program, Control, and Data memories

e. RX33 System diskette or equivalent local HSC load device

2. SDI compatible disk drive

and/or

3. STI compatible tape drive

4. HSC system software including:

a. HSC internal operating system

b. DEMON

c. K.sdi microcode and/or

d. K.sti microcode

e. SDI Manager and/or

f. STI Manager or equivalent

g. Disk functional code and/or

h. Tape functional code

i. Error Handler

j. Diagnostic Interface to Disk functional code and/or

k. Diagnostic Interface to Tape Functional code

Tests cannot be performed on drives if their respective interface is not available (K.sdi or K.sti.)

5.7.2 ILEXER Operating Instructions

Perform the following steps to initiate ILEXER (Multidrive Exerciser):

1. Type CTRL Y

2. The HSC responds with an

HSC70> prompt

3. Type: **RUN DX0:ILEXER.DIA**

The system loads the program from the specified local HSC load media (any appropriate media with the image ILEXER.DIA in an RT11 format). When the program is successfully loaded, the following message is displayed:

ILEXER>D>hh:mm Execution Starting

where 'hh:mm' is the current time.

ILEXER then prompts for parameters. After all prompts are answered, the execution of the diagnostic proceeds. Error reports and performance summaries are returned from ILEXER.

When ILEXER has run for the specified time interval, reported any errors found, and generated a final performance summary, the exerciser concludes with the following message:

ILEXER>D>hh:mm Execution Complete

5.7.3 ILEXER Test Parameter Entry

The parameters in ILEXER follow the format:

PROMPT DESCRIPTION (DATATYPE) [DEFAULT]?

- The PROMPT DESCRIPTION explains the type of information ILEXER needs from the operator.
- The DATATYPE is the form ILEXER expects and can be one of the following:

  Y/N - Yes/No response

  D - Decimal number

  U - Unit number (see form below)

  H - Hexadecimal number

- The DEFAULT is the value used if a carriage return is entered for that particular value. If a default value is not allowed, it appears as [].

The next prompt is:

DRIVE UNIT NUMBER (U) [] ? T0 (tape)

Enter the unit number of the drive to be tested. This prompt has no default. Unit numbers are either in the form Dnnnn or Tnnnn, where nnnn is a decimal number between 0 and 4095 which corresponds to the number printed on the drive's unit plug and the D or T indicates either a disk drive or tape drive, respectively. Terminate the unit number with a carriage return. ILEXER attempts to acquire the specified unit via the HSC Diagnostic Interface. If the unit is acquired successfully, ILEXER continues with the next prompt. If the acquire fails with an error, one of the following conditions was encountered:

1. The specified drive is unavailable. This indicates the drive is connected to the HSC but is currently online to a Host CPU or HSC utility. Online drives cannot be diagnosed. ILEXER repeats the prompt for the unit number.

2. The specified drive is unknown to the HSC Disk functional software. Drives are Unknown for one of the following reasons:

   - The drive and/or K.sdi port is broken and cannot communicate with the disk functional software.

   - The drive was previously communicating with the HSC when a serious error occurred and the HSC ceased communicating with the drive.

   In either case, ILEXER asks the operator if another drive will be selected. If so, it asks for the unit number. If not, ILEXER begins to exercise the drives selected. If no drives are selected, ILEXER terminates.

After a drive is selected and ILEXER has both acquired the drive and brought it online, the following prompts appear. If a disk drive was specified, one set of prompts is presented. If a tape drive was selected, an entirely different set of prompts is presented. A CTRL Z at any time during parameter input selects the default values for the remaining parameters. If a nondefaultable parameter is encountered, the following message appears and the test prompts for new parameters:

ILEXER>D>hh:mm Nondefaultable Parameter

Select up to 12 drives to be exercised; either all disk drives, all tape drives, or a combination of the two.

5.7.4 Disk Drive User Prompts

The following prompts are presented if the drive selected is a disk drive.

ACCESS USER DATA AREA (Y/N) [N]?

A Y answer to this and the following prompt directs ILEXER to perform testing in the user data area. It is the operator's responsibility to see to it the data contained there is either backed up or of no value. If this prompt is answered with an N, or carriage return, testing is confined to the disk area reserved for diagnostics (DBN area). When testing is confined to the DBN area, the following five prompts are not displayed.

ARE YOU SURE (Y/N) [N]?

An N response causes the previous prompt to be repeated. A Y response allows the exercise to take place in the user data area of the disk.

START BLOCK NUMBER (D) [0]?

This value specifies the starting block of the area ILEXER exercises when the user data area is selected. If block 0 is specified, ILEXER will exercise beginning with the first LBN on the disk.

END BLOCK NUMBER (D) [0=MAX]?

This parameter specifies the ending block of the area ILEXER exercises when the user data area is selected. If block 0 is specified as the ending block, ILEXER exercises up to the last LBN on the disk.

INITIAL WRITE TEST AREA (Y/N) [N]?

Answering Y to this prompt causes ILEXER to write the entire test area before beginning random testing. If the prompt is answered with an N or a carriage return, the prompt immediately following this is omitted.

TERMINATE TEST ON THIS DRIVE FOLLOWING INITIAL WRITE (Y/N) [N]?

This question allows an initial write on the drive and terminates the test at that point. The default answer, N, permits this initial write. After completing the initial write, the test continues to exercise the drive.

NOTE

The following prompts specify the test sequence for that part of the test following the initial write portion. That is, even if the operator requests Read Only mode, the drive will not be write protected until after any initial write has been completed.

SEQUENTIAL ACCESS (Y/N) [N]?

The operator has the option of requesting all disk data access be performed in a sequential manner.

READ ONLY (Y/N) [N]?

If answered N, the operator is asked for both a pattern number and the possibility of Write Only mode. If the answer is Y, ILEXER does not prompt for Write Only mode, but only asks for a data pattern number if an initial write was requested.

DATA PATTERN NUMBER (0-15) (D) [15]?

The operator has the option of selecting one of 16 disk data patterns. Selecting data pattern 0 allows selection of a pattern with a maximum of 16 words. The default data pattern (15) is the factory format data pattern.

WRITE ONLY (Y/N) [N]?

This option permits only write operations on a disk. This prompt is not displayed if Read Only mode was selected.

DATA COMPARE (Y/N) [N]?

If this prompt is answered with an N or a carriage return, data read from the disk is not checked; for example, disk data is not compared to the expected pattern. If the prompt is answered with a Y, the following prompt is issued. The media must have been previously written with a data pattern in order to do a data compare.

DATA COMPARE ALWAYS (Y/N) [N]?

Answering a Y causes ILEXER to check the data returned by every disk read operation. Answering with an N or carriage return, causes data compares on 15 percent of the disk reads.

NOTE

Selection of data compares significantly reduces the number of disk sectors transferred in a given time interval.

ANOTHER DRIVE (Y/N) []?

Answering with a Y permits selection of another drive for exercising. This prompt has no default. Answering with an N causes ILEXER to prompt:

AVERAGE DISK TRANSFER LENGTH IN SECTORS (1 TO 400) []?

AVERAGE DISK TRANSFER LENGTH IN SECTORS (1 TO 35) []?

This prompt requests the selection of the average size (in sectors) of each data transfer issued to the disk drives.

Once the preceding parameters are entered, ILEXER continues with the prompts listed as global user prompts (Section 5.7.6).

5.7.5 Tape Drive User Prompts

ILEXER displays the following prompts if the drive selected is a tape drive:

IS A SCRATCH TAPE MOUNTED (Y/N) [N]?

An N response results in a reprompt for the drive unit number. A Y response displays the next prompt.

ARE YOU SURE (Y/N) [N]?

If the answer is N, the operator is reprompted for the drive unit number. If answered with a Y, the following prompts are displayed.

DATA PATTERN NUMBER (16-22) (D) [21]?

Seven data patterns are available for tape. The default pattern (pattern 21) is defined in Section 5.7.7.

DENSITY (1=800, 2=1600, 3=6250) (D) [2] ?

The response to this prompt is a 1, a 2, or a 3. Any other response is illegal, and the prompt is displayed again. The default is 2 or a density of 1600 bpi.

SELECT AUTOMATIC SPEED MANAGEMENT (Y/N) [N]?

Either Automatic Speed Management (if the feature is supported) or a tape drive speed is selected at this point. If the choice is Automatic Speed Management, the available speeds are not displayed.

ILEXER>D>FIXED [VARIABLE] SPEEDS AVAILABLE:

This is an informational message identifying the speeds available for the tape drive. If the speeds are fixed, the value is presented. If the speed is variable within a range, the range is listed, and the next prompt asks the operator to select a speed. See the tape drive user manual for available speeds.

SELECT FIXED [VARIABLE] SPEED (D) [1]?

This prompt allows selection of the variable speed for the tape drive selected. See the tape drive user manual for available speeds.

RECORD LENGTH IN BYTES (1 or 12288) (D) [8192]?

Response to this prompt specifies the size in bytes of a tape record. Maximum size is 12K bytes. The default value is 8192, the standard record-length size for 32-bit systems. Constraints on the HSC diagnostic interface prohibit selection of the maximum allowable record length of 64K bytes.

DATA COMPARE (Y/N) [N]?

Answering N results in no data compares performed during a read from tape. A Y response causes the following prompt.

DATA COMPARE ALWAYS (Y/N) [N]?

A Y response selects data compares to be performed on every tape read operation. An N response causes data compares to be performed on 15 percent of the tape reads.

ANOTHER DRIVE (Y/N) []?

Answering Y, the prompts beginning with the prompt for DRIVE UNIT NUMBER, are repeated. If answered NO, the following global prompts are presented. This prompt has no default, allowing the operator to default all other prompts and be able to parameterize another drive for this pass of ILEXER.

5.7.6 ILEXER Global User Prompts

The following prompts are presented to the operator when no more drives or drive-specific parameters are to be entered into the testing sequence. These prompts are global in the sense they pertain to all the drives.

RUN TIME IN MINUTES (1 TO 32767) [10]?

The minimum time is 1 minute, and the default is 10. After the exerciser has executed for that period of time, all testing terminates and a final performance summary is displayed.

HARD ERROR LIMIT (D) [20]?

You are allowed to specify the number of hard errors allowable for the drives being exercised. When a drive reaches this limit, it is removed from any further exercising on this pass of ILEXER. Hard errors include the following types of errors:

- Tape drive BOT encountered unexpectedly
- Invalid MSCP response received from functional code
- UNKNOWN MSCP status code returned from functional code
- Write on write-protected drive
- Tape formatter returned error
- Read compare error
- Read data EDC error
- Unrecoverable read or write error
- Drive reported error
- Tape mark error (ILEXER does not write tape marks)
- Tape drive truncated data read error
- Tape drive position lost
- Tape drive short transfer occurred on read operation
- Retry limit exceeded for a tape read, write, or read reverse operation
- Drive went OFFLINE or AVAILABLE unexpectedly

NARROW REPORT (Y/N) [N]?

Answering Y presents a narrow report which displays the performance summaries in 32 columns. The default display, selected by answering N, or carriage return, is 80 columns. The format of this display is described in further detail in Section 5.7.11. This report format is intended for use by small hand-held terminals.

ENABLE SOFT ERROR REPORTS (Y/N) [N]?

This prompt enables soft error reports by answering Y. By default, the operator does not see any soft error reports specific to the number of retires required on a tape I/O operation. A N response results in no soft error report. Soft errors are classified as those errors that eventually complete successfully after explicit controller-managed retry operations. They include read, write, and read-reverse requested retries.

DEFINE PATTERN 0 - HOW MANY WORDS (16 MAX) (D) [16]?

If data pattern 0 was selected for any preceding drive, the size of the data pattern must be defined at this time. The pattern can contain as many as 16 words, also the default. If a number larger than 16 is supplied, an error message is displayed and this prompt is presented again. When a valid response is presented, the following prompt is displayed the specified number of times.

DATA IN HEX (H) [0]?

This prompt is displayed as many times as the number of words specified in the previous response. ILEXER is expecting a 4-character hex value as the answer to this prompt.

5.7.7 ILEXER Data Patterns

The data patterns available for use with ILEXER are listed in the following sections. Note that Pattern 0 is a user-defined data pattern. Space is available for a repeating pattern of up to 16 words.

Pattern 0	Pattern 1	Pattern 2	Pattern 3
User Defined	105613	031463	030221

Pattern 4	Pattern 5	Pattern 6	Pattern 7
Shifting 1s	Shifting 0s	Alter 1s,0s	B1011011011011001
000001	177776	000000	133331
000003	177774	000000	
000007	177770	000000	
000017	177760	177777	
000037	177740	177777	
000077	177700	177777	
000177	177600	000000	
000377	177400	000000	
000777	177000	177777	
001777	176000	177777	
003777	174000	000000	
007777	170000	177777	
017777	160000	000000	
037777	140000	177777	
077777	100000	000000	
177777	000000	177777	

Pattern 8	Pattern 9	Pattern 10	Pattern 11
B0101../B1010..	B110...	26455/151322	
052525	155554	026455	066666
052525		026455	
052525		026455	
125252		151322	
125252		151322	
125252		151322	
052525		026455	
052525		026455	
125252		151322	
125252		151322	
052525		026455	
125252		151322	
052525		026455	
125252		151322	
052525		026455	
125252		151322	

Pattern 12	Pattern 13	Pattern 14	Pattern 15
Ripple 1	Ripple 0	Manufacture -	Patterns
000001	177776	155555	155555
000002	177775	133333	133333
000004	177773	155555	066666
000010	177767	155555	155555
000020	177757	133333	133333
000040	177737	155555	066666
000100	177677	155555	155555
000200	177577	133333	133333
000400	177377	155555	066666
001000	176777	155555	155555
002000	175777	133333	133333
004000	173777	155555	066666
010000	167777	155555	155555
020000	157777	133333	133333
040000	137777	155555	066666
100000	077777	155555	155555

Data patterns for tapes follow:

Pattern 16	Pattern 17	Pattern 18	Pattern 19
Alternating one and zero bits	All ones	Alternating bytes of all	
125252		all ones	
125252			
.			
.			
.			

Pattern 20	Pattern 21	Pattern 22
Alternating two bytes ones and two bytes zeros	Alternating three bytes ones and one byte zeros	

5.7.8 Setting/Clearing Flags - ILEXER

One parameter is specified in Section 5.7.6 which allows the operator to inhibit the display of soft error reports. No other error reports can be inhibited.

5.7.9 ILEXER Progress Reports

ILEXER has three basic forms of progress reports; the data transfer error report, the performance summary, and the communication error report.

- The data transfer error report is printed each time an error is encountered in one of the drives being tested.
- The performance summary report is printed when ILEXER completes this pass on each drive being exercised or when the operator terminates the pass via a CTRL Y. This performance summary is also printed on a periodic basis during the execution of ILEXER.
- The communication error report is sent to the console terminal any time ILEXER is unable to establish and maintain communications with the drive selected for exercising.

5.7.10 ILEXER Data Transfer Error Report

The report described here is printed on the terminal each time a data transfer error is found during the execution of this pass of ILEXER. The report describes the nature of the error and all data pertinent to the error found.

The data transfer error report is a standard HSC error log message. It contains all data necessary to identify the error.

The only exception to this is when the error encountered by performed a data check and found an error during the compare, resulting in an ILEXER error report.

5.7.11 ILEXER Performance Summary

The Performance Summary is printed on the terminal at the end of the testing session, when manually terminated, or every specified number of minutes for the periodic performance summary. This report provides statistical data which was being tabulated by ILEXER during the execution of this test.

The performance summary presents the statistics which are maintained on each drive. This summary contains the drive unit number, the drive serial number, the number of position commands performed, the number of 0.5 Kbytes read and written, the number of hard errors, the number of soft errors, and the number of software correctable transfers. For tape drives being exercised by ILEXER, an additional report breaks down the software correctable errors into eight different categories.

The frequency of report display is altered in the following fashion:

1. Type CTRL G during the execution of ILEXER

2. The following prompt is displayed:

```
MFGEXR>D>    Options are:
MFGEXR>D>         0 = No action
MFGEXR>D>         1 = (not implemented)
MFGEXR>D>         2 = (not implemented)
MFGEXR>D>         3 = frequency of performance summary
Enter Option (0,1,2,3) (D) []:
```

3. Enter in the preferred option. Currently, the only options available are 0 and 3.

4. If option 3 is selected, the following prompt is displayed. The valid values range from 1 to 3599 for the number of seconds between printings of performance summaries. From that point on, the summary is displayed as often as specified. The operator can enter a 1, which prints a performance summary immediately but does not alter the frequency of the report. Also note, a value equal to or greater than one hour is not allowed to avoid a reboot of the HSC.

   ```
   Interval time for performance summary in seconds (D)
   [30]?
   ```

The format of the Performance Summary follows:

PERFORMANCE SUMMARY (DEFAULT)

UNIT NO	R -	SERIAL NUMBER	POSITION	KBYTE READ	KBYTE WRITTEN	HARD ERROR	SOFT ERROR	SOFTWARE CORRECTED
Dddd		HHHHHHHHHHH	ddddd	ddddddddd	ddddddddd	ddddd	ddddd	ddddd
Tddd		HHHHHHHHHHH	ddddd	ddddddddd	ddddddddd	ddddd	ddddd	ddddd

A performance summary is displayed for each disk drive and tape drive active
on the HSC.

where:

1. UNIT NUMBER - the unit number of the drive. D for disk, T for tape. The number is reported in decimal.

2. R - the status of the drive. If an asterisk (*) appears in this field, the drive was removed from the test and the operator was previously informed. If the field is blank, the drive is being exercised.

3. SERIAL NUMBER - the serial number (hexadecimal) for each drive.

4. POSITION - the number of seeks.

5. KBYTE READ - the number of 1 Kbytes read by ILEXER on each drive.

6. KBYTE WRITTEN - the number of Kbytes written by ILEXER.

7. HARD ERROR - the number of hard errors reported by ILEXER for a particular drive.

8. SOFT ERROR - the number of soft tape errors reported by the exerciser if enabled by the operator.

9. SOFTWARE CORRECTED - the number of ECC correctable reads encountered by ILEXER. Only ECC correctable errors above the specific drive ECC error threshold are reported via normal functional code error reporting mechanisms. ECC correctable errors below this threshold are not reported via an error log report, but only included in this count maintained by ILEXER.

If any tape drives were being exercised, the following summary is displayed within each performance summary.

```
UNIT  MEDIA  DOUBLE  DOUBLE  SINGLE  SINGLE  OTHER  OTHER  OTHER
NO    ERROR  TRKERR  TRKREV  TRKERR  TRKREV  ERR A  ERR B  ERR C

Tddd  ddddd  ddddd   ddddd   ddddd   ddddd   ddddd  ddddd  ddddd
etc.
```

where:

1. MEDIA ERROR - the number of bad spots detected on the recording media.

2. DOUBLE TRKERR - the number of double track errors encountered during a read or write forward.

3. DOUBLE TRKREV - the number of double track errors encountered during a reverse read or write.

4. SINGLE TRKERR - the number of single track errors detected during a read or write in the forward direction.

5. SINGLE TRKREV - the number of single track errors encountered during a reverse read or write.

6. Other Err A-C - reserved for future use.

PERFORMANCE SUMMARY (NARROW)

```
ILEXER>D>PER SUM
D[T]ddd
SN HHHHHHHHHHH
P ddddd
R ddddddddddd
W ddddddddddd
HE ddddd
SE ddddd
SC ddddd
```

This report is repeated for each drive tested.

If any tape drives are being tested, the following report is issued for each tape drive following the disk drive performance summaries.

```
ILEXER>D>ERR SUM
ILEXER>D>Tddd
ILEXER>D>ME ddddd
ILEXER>D>DF ddddd
ILEXER>D>DR ddddd
ILEXER>D>SF ddddd
ILEXER>D>SR ddddd
ILEXER>D>OA ddddd
ILEXER>D>OB ddddd
ILEXER>D>OC ddddd
```

5.7.12 ILEXER Communications Error Report

Whenever ILEXER encounters an error that prevents it from communicating with one of the drives to be exercised, ILEXER issues a standard error report. This report gives details enabling the operator to identify the problem. For further isolation of the problem, the operator should run another diagnostic specifically designed to isolate the failure (ILDISK or ILTAPE).

5.7.13 ILEXER Test Termination

Upon completion of the exercise on each selected drive, reporting of any errors found, and display of final performance summary, ILEXER terminates normally. All resources, including the drive being tested, are released. The operator may terminate ILEXER before normal completion by typing a CTRL Y. The following output is displayed, plus a final performance summary:

```
ILEXER>D>hh:mm DIAGNOSTIC ABORTED
ILEXER>D>PLEASE WAIT - CLEARING OUTSTANDING I/O
```

Certain parts of ILEXER cannot be interrupted, so the CTRL Y may have no effect for a brief moment and may need repetition. Whenever ILEXER is terminated, whether normally or by operator abort, ILEXER always completes any outstanding I/O requests and prints a final performance summary.

5.7.14 ILEXER Error Message Format

ILEXER outputs four types of error formats: prompt errors, data compare errors, pattern word errors, and communication errors. These formats agree with the generic diagnostic error message format (Section 5.1.1.1).

5.7.14.1 ILEXER Prompt Error Format - Prompt errors occur when the operator enters the wrong type of data or the data is not within the specified range for a parameter. The general format of the error message is:

```
ILEXER>D>error message
```

Where the error message is an ASCII string describing the type of error discovered.

5.7.14.2 ILEXER Data Transfer Compare Error Format - A data transfer compare error occurs when an error is detected during the exercise of a particular drive.

The two formats for the data transfer compare error are, depending upon the type of error, data compare error and pattern word error.

A data compare error occurs when the data read does not match the expected pattern. The format of the data compare error is:

```
ILEXER>D>hh:mm T ddd E ddd U-uddd
ILEXER>D>Error Description
ILEXER>D>MA - HHHHHHHHHH
ILEXER>D>EXP - HHHH
ILEXER>D>ACT - HHHH
ILEXER>D>MSCP STATUS CODE = HHHH
ILEXER>D>FIRST WORD IN ERROR = ddddd
ILEXER>D>NUMBER OF WORDS IN ERROR = ddddd
```

where:

hh:mm - a time stamp since the start of ILEXER

T - the test number in the exerciser

E - corresponds to the error number

U - the unit number for which the error is being reported

MA - the media address (block number) where the error occurred

EXP - the expected data

ACT - the data (or code) actually received

MSCP STATUS CODE - the code received from the operation

FIRST WORD IN ERROR - describes the number of the first word found in error.

NUMBER OF WORDS IN ERROR - once an error is found, the routine continues to check the remainder of the data returned and counts the number of words found in error.

The format for the pattern word error is slightly different from the data compare error. A pattern word error occurs when the first data word in a block is not a valid pattern number. The format is:

```
ILEXER>D>hh:mm T ddd E ddd U-uddd
ILEXER>D>Error Description ILEXER>D>MA - HHHHHHHHHH
ILEXER>D>EXP - HHHH
ILEXER>D>ACT - HHHH
```

The MSCP status code, first word in error, and number of words in error are not relevant for this type of error. The other fields are as described for the data compare error.

5.7.14.3 ILEXER Communications Error Format - Communications errors occur when ILEXER cannot establish/maintain communications with a selected drive The error message appears in the following format:

```
ILEXER>D>hh:mm T ddd E ddd U-uddd
ILEXER>D>Error Description
ILEXER>D>Optional Data lines follow here
```

where:

hh:mm - time stamp for the start of ILEXER

T - the test number in the exerciser

E - corresponds to the error number

U - the unit number for which the error is being reported.

Error Description - an ASCII string describing the error encountered.

Optional Data lines - a maximum of eight optional lines per report.

5.7.15 ILEXER Error Messages

The following section is a list of the informational messages and error messages and an explanation of the cause of the error. A typical error message looks like:

```
ILEXER>D>09.32 T#006 E#204 U-T00100

ILEXER>D>Comm Error: TBUSUB call failed
```

5.7.15.1 **ILEXER Informational Messages** - These messages are not fatal to the exerciser. They alert the user to incorrect input to parameters, indicate missing interfaces, or are informational.

#1 **Number must be between 0 and 15** - reported when the user entered an erroneous value for the data pattern on a disk.

#2 **Pattern Number must be within specified bounds** - reported when the operator tries to specify a disk pattern number for a tape.

#3 **You May Enter at Most 16 Words in a Data Patter n** - reported if the operator specifies more than 16 words for a user defined pattern, and the operator is reprompted for the value.

#4 **Starting BN is either Larger than Ending BN or Larger than Total BN on Disk** - reprompts for the correct values. The operator selected a starting block number for the test which was greater than the ending block number selected, or it is greater than the largest block number for the disk.

#5 **Please Mount a Scratch Tape** - appears after an N response to the prompt asking if the scratch tape is mounted on the tape drive to be tested.

#6 **Disk Interface Not Available** - indicates the disk functionality is not available to exercise disk drives. This means the K.sdi is not available or not operable.

#7 **Tape Interface Not Available** - indicates the tape functionality is not available to exercise tape drives. This means the K.sti is not available or not operable.

#8 **Please Wait - Clearing Outstanding I/O** is printed when the operator enters a CTRL Y to stop ILEXER. All outstanding I/O commands are aborted at this time.

5.7.15.2 **ILEXER Generic Errors** - The following list indicates the number, text, and cause of errors displayed by ILEXER.

epur

#1 **No Disk or Tape Functionality...Exerciser Terminated** - Neither the K.sdi or K.sti interfaces are available to run the exercise. This terminates ILEXER.

#2 **Could not Get Control Block For Timer - Stopping Multi-Drive Exerciser** - ILEXER could not obtain a transmission queue for a timer. This should occur only on a heavily loaded system and is fatal to ILEXER.

#3 **Could not Get Timer For MDE - Stopping Multi-Drive Exerciser** - The exerciser cannot obtain a timer. Two timers are required for ILEXER. This should only occur on a loaded system and is fatal to ILEXER.

#4 **Disk functionality Unavailable-Choose Another Drive** - The disk interface is not available. A previous message is printed at the start of ILEXER if any of the interfaces are missing. This error prints when the operator still chooses a disk drive for the exercise.

#5 **Tape Functionality Unavailable-Choose Another Drive** - The tape interface is not available. A previous message is printed at the start of ILEXER if any of the interfaces are missing. This error prints when the operator still chooses a tape drive for the exercise.

#6 **Couldn't Get Drive Status-Choose Another Drive** - ILEXER was unable to obtain the status of a drive for one of the following conditions:

1. The drive is not communicating with the HSC. Either the formatter or the disk is not Online.

2. The cables to the K.sdi or K.sti are loose.

#7 **Drive is Unknown-Choose Another Drive** - The drive chosen for the exerciser is not known to the HSC functional software for that particular drive type. Either the drive is not communicating with the HSC or the functional software has been disabled due to an error condition on the drive.

#8 **Drive is Unavailable-Choose Another Drive** - This may be the result of:

1. The drive port button is disabled for this port.

2. The drive is Online to another controller.

3. The drive is not able to talk to the controller on the port selected.

#9 **Drive Cannot Be Brought Online** - ILEXER was unable to bring the selected drive online. One of the following conditions occurred:

1. The unit went into an off-line state and cannot communicate with the HSC.

2. The unit specified is now being used by another process.

3. There are two drives of same type with duplicate unit numbers on the HSC.

4. An unknown status was returned from the HSC diagnostic interface when ILEXER attempted to bring the drive online.

#10 **Could not return Drive to Available State** - The release of the drive from ILEXER was unsuccessful. This is the result of a drive being taken from the test due to reaching an error threshold or going off line during the exercise.

#11 **User Requested Write on Write Protected Unit** - The operator should check the entry of parameters and also check the write protection on the drive to make sure they are consistent.

#12 **No Tape Mounted on Unit...Mount and Continue** - The operator specified a scratch tape was mounted on the tape drive selected when it was not mounted. Mount a tape and continue.

#13 **Record Length larger than 12K or less than 0** - The record length requested for the transfer to tape was either greater than 12K or less than 0.

#14 **This unit already acquired** - A duplicate unit number was specified for a drive and the drive had already been acquired.

#15 **Invalid time entered...must be from 1 to 3599** - This is reported when the user enters an erroneous value to the performance summary time interval prompt.

#16 **Could not get buffers for transfers** - This message is reported when the buffers required for a tape transfer cannot be acquired.

#17 **Tape rewind commands were lost - cannot continue** - This error message results from the drive being unloaded during ILEXER execution.

5.7.15.3 **ILEXER Disk Errors** - The following list includes the number, text, and cause of ILEXER disk errors.

#102 **Drive Spindle not Up to Speed. Spin Up Drive And Restart** - The disk drive is not spun up.

#103 **This Drive Removed From Test** - This is reported when a disk drive reaches the hard error limit or the drive goes off line to the HSC during the exercise.

#104 **Couldn't Put Drive in DBN Space - Removed From Test** - An error or communication problem occurred during the delivery of an SDI command to put the drive in DBN space.

#105 **No DACB Available - Notify Field Support, submit SPR** - This is reported if no DACBs can be acquired. If this happens, contact Field Support as soon as possible and submit an SPR.

#106 **Some Disk I/O Failed to Complete** - An I/O transfer did not complete during an allotted time period.

#107 **Command Failed - Invalid Header Code** - ILEXER did not pass a valid header code to the diagnostic interface for the HSC.

#108 **Command Failed - No Control Structures Available** - The diagnostic interface could not obtain disk access control blocks to run the exercise. The HSC could be overloaded. Try ILEXER on a quiet system. If the error still occurs, test the HSC memory.

#109 **Command Failed - No Buffer Available** - The diagnostic interface could not obtain buffers to run the exercise. The HSC could be overloaded. Try ILEXER on a quiet system. If the error still occurs, test the HSC memory.

#110 **Write Requested on Write Protected Drive** - The operator requested an initial Write operation on a drive which was already write protected. The operator should pop out the write protect button on the drive reporting the error or have ILEXER do a READ ONLY operation on the drive.

#111 **Data Compare Error** - Bad data was detected during a read operation.

#112 **Pattern Number Error** - The first two bytes of each sector, which contain the pattern number, did not match.

#113 **EDC Error** - Error Detection Code error: invalid data was detected during a read operation.

#114 **Unknown Unit number not allowed in ILEXER...** - The operator attempted to enter in a unit number of the form, 'Xnnnn', which is not accepted by ILEXER.

#115 **Disk Unit numbers must be between 0 and 4095 decimal** - The operator specified a disk unit number out of the allowed range of values.

#116 **Hard Failure on Disk** - A hard error occurred on the disk drive being exercised.

The following errors identify the function attempted by ILEXER which caused an error to occur. Error logs do not indicate the operation attempted.

#117 **Hard Failure on COMPARE Operation** - A hard failure occurred during a compare of data on the disk drive.

#118 **Hard Failure on WRITE Operation** - A hard fault occurred during a write operation on the specified disk drive.

#119 **Hard Failure on READ Operation** - A hard failure occurred during a read operation on the disk drive being exercised.

#123 **Hard Failure on INITIAL WRITE Operation** - A hard failure occurred during the first write to the disk drive.

#124 **Drive went spontaneously available** - A drive which was being exercised went into an Available state. This could be caused by the operator releasing the port button on the drive. A fatal drive error could also cause the drive to go into this state.

5.7.15.4 **ILEXER Tape Errors** - The following list includes the number, text, and cause of ILEXER tape errors.

#201 **Couldn't Get Formatter Characteristics** - A communications problem with the drive is indicated. It could be caused by the unit not being online.

#202 **Couldn't Get Unit Characteristics** - The drive is not communicating with ILEXER. The unit could be off line.

#203 **Some Tape I/O Failed to Complete** - The drive or formatter stopped functioning properly during a data transfer.

#204 **Communication Error: TDUSUB call failed** - ILEXER cannot talk to the drive via interface structures. They have been removed. Either the drive went available from online, or is off-line, or a fault occurred.

#205 **Read Data Error** - A read operation failed during a data transfer, and none was transferred.

#206 **Tape Mark Error...rewinding to restart** - ILEXER does not write tape marks. If this error occurs, it indicates a drive failure.

#207 **Tape Position Lost...rewinding to restart** - An error occurred during a data transfer or a retry of one.

#209 **Data Pattern word Error...Possible Media Defect** - The first two bytes of a record containing the data pattern did not match.

#210 **Data Read EDC Error..continuing...** - Error Detection Code error - incorrect data was detected.

#211 **Could Not Set Unit Char...removing from test** - The drive is off line and not communicating.

#213 **Truncated Record Data Error...rewinding to restart** - More data was received than expected indicating a drive problem.

#214 **Drive Error...Hard Error...continuing** - A hard failure occurred with the drive being exercised.

#215 **Unexpected Error Condition...removing drive from test** - This is caused by MSCP error conditions which are not allowed (i.e., invalid commands, unused codes, write-protected drive write, etc.).

#216 **Unexpected BOT encountered...will try to restart** - The drive is experiencing a positioning problem.

#217 **Unrecoverable Write Error...rewinding to restart** - A hard error occurred during a write operation. The write did not take place due to this error.

#218 **Unrecoverable Read Error...rewinding to restart** - A hard error occurred during a read operation and a data transfer did not take place.

#219 **Controller Error...Hard Error..rewinding to restart** - This indicates a communications problem between the controller and the formatter.

#220 **Formatter Error...Hard Error...continuing** - A communications problem exists between the formatter and the controller and/or drive.

#221 **Retry Required on Tape Drive** - A read/write operation which failed required a retry before succeeding.

#222 **Hard Error Limit Exceeded...removing drive from test** - The drive exceeded the threshold of hard errors determined by a global user parameter (Section 5.7.6). The drive is then removed from the exercise.

#224 **Drive Went Offline...removing from test** - The drive went off line during the exercise. This is caused by the operator taking the drive off line or a hard failure forcing the drive off line.

#225 **Drive Went Available...removing from test** - The drive became available to ILEXER and was not at the beginning of the exercise.

#226 **Short Transfer Error...rewinding to restart** - Less data was received than transferred.

#227 **Tape Position Discrepancy** - The tape position was lost indicating a hard failure.

5.7.16 ILEXER Test Summaries

The test numbers in ILEXER correspond to the module being executed within ILEXER itself. The main module is called MDE, and it calls all other modules.

- **Test Number 1 - Main Program: MDE**

  Multi-drive Exerciser is the main program withi ILEXER. It is responsible for calling all other portions of ILEXER. It obtains the buffers and control structures for the exerciser. It verifies either disk or tape functionalities are available before allowing ILEXER to continue.

- **Test Number 2 - INITT**

  INITT is called to initialize drive statistic tables. It obtains the parameters and verifies the values of each one entered. This routine calls INICOD to obtain drive specific parameters.

- **Test Number 3 - INICOD**

  INICOD is the initialization code for ILEXER. It gets the various parameters for the drives from the operator and fills in the drive statistic tables with initial data for each drive. It also verifies the validity of the input for the parameters. INICOD, in turn, calls ACQUIRE to acquire the disk and/or tape drive.

- **Test Number 4 - ACQUIRE**

  ACQUIRE is responsible for acquiring the drives as specified by the parameters. It brings all selected drives online to the controller and spins up the disk drives. Errors reported in this routine cause the removal of the drive from the exercise.

- **Test Number 5 - INITD**

  INITD initializes the disk drives for the exercise. This routine clears all disk access control blocks and invokes the initial write.

- **Test Number 6 - TPINIT**

  TPINIT initializes the tape drives for the exercise. It rewinds all acquired tape drives and verifies the drives are at the BOT. If an error occurs, the drive is removed from the exerciser. TPINIT is also responsible for obtaining buffers for each acquired tape drive.

- **Test Number 7 - Exerciser**

  EXER is the main code of the exerciser. It dispatches to the disk exerciser (QDISK and CDISK) and the tape exerciser (TEXER). It continuously queues up I/O commands to disk and tape, and checks for I/O completion. The subroutines EXER calls are responsible for sending commands and checking for I/O completion.

- **Test Number 8 - QDISK**

  QDISK is part of the disk exerciser which selects commands to send to the disk drives. If the initial write is still in progress, it returns to EXER. QDISK calls a routine to select the command to exercise the disk drive. The following scenario is the algorithm used to select the command:

  - If the drive is read only and data compare is not requested, a Read operation is queued to the drive.

- If read only and data compare (occasional) are requested, a Read operation is queued along with a random choice of compare/not-compare.

- If read only and data compare (always) are requested by the operator, a READ-COMPARE command is queued to the drive.

- If write only is requested, and data compare is not, then a WRITE request is queued up to the disk drive.

- If write only and data compare (occasional) are requested, a Write operation is queued along with a random choice of compare/not-compare.

- If write only and data compare (always) are requested, a WRITE-COMPARE command is queued to the drive.

- If only data compare (occasional) is requested, then a random selection of READ/WRITE and compare/not-compare will be done.

- If only data compare (always) is requested, a COMPARE command is paired with a random selection of READ/WRITE.

QDISK randomly selects the number of blocks for the selected operation.

o **Test Number 9 - RANSEL**

RANSEL is the part of the tape exerciser which is responsible for sending commands to the tape drives. This routine is called by TEXER, the tape exerciser routine. RANSEL selects a command for a tape drive using a random number generator. Following are some constraints for the selection process:

- No reads when there are no records before or after the current position.

- No writes when there are records after current position.

- No position of record when no records are before or after the current position.

- Reverse commands are permitted on the drive when 16 reverse commands have previously been selected. That is, 1 out of every 16 reverse commands are sent to the drive. Immediately following a reverse command, a position to the end-of-written-tape is performed. The reason for forward biasing the tape is to prevent thrashing.

The following commands are executed in exercising the tape drives:

1. READ FORWARD
2. WRITE FORWARD
3. POSITION FORWARD
4. READ REVERSE
5. REWIND
6. POSITION REVERSE

RANSEL randomly selects the number of records to read, write, or skip.

o **Test Number 10 - CDISK**

CDISK checks for the completion of disk I/O specified by QDISK. CDISK checks the return status of a completed I/O operation and if any errors occur, they are reported.

o **Test Number 11 - TEXER**

TEXER is the main tape exerciser which selects random writes, reads, and position commands. TEXER processes the I/O once it is completed and reports any errors encountered.

o **Test Number 12 - EXCEPT**

EXCEPT is the ILEXER exception routine. This is the last routine called by MDE. EXCEPT is called when a fatal error occurs, when ILEXER is stopped with a CTRL/Y, or when the program expires its allotted time. It cleans up any outstanding I/O, as necessary, returns resources, and returns control to DEMON.

CHAPTER 6
OFFLINE DIAGNOSTICS

6.1 INTRODUCTION

This chapter describes the offline diagnostics, how to run them, errors that can occur, and summaries of the tests in each diagnostic. Included in the offlines are:

- Offline Diagnostic Loader
- Offline Cache Test
- Offline Bus Interaction Test
- Offline K Test Selector
- Offline K/P Memory Test
- Offline Memory Test
- RX33 Offline Exerciser
- Offline Refresh Test
- Offline Operator Control Panel Test

The offline diagnostics contain specific common characteristics, discussed in the following three sections. They are listed below.

- Identical software requirements
- Common load procedure
- Identical bootstrap initialization procedures
- Generic error message format

6.1.1 Offline Diagnostics Software Requirements

All offline diagnostics require an RX33 Offline Diagnostic diskette containing a bootable image of the offlines software programs.

6.1.2 Offline Diagnostics Load Procedure

The Offline diagnostics diskette boots from either RX33 drive and should not be write-enabled. This diskette contains the necessary software to run all the HSC70 Offline diagnostics. Booting is done either by powering on or by depressing and releasing the Init switch with the Secure/Enable switch in the ENABLE position. This causes the P.ioj ROM bootstrap tests to run followed by the Offline P.ioj test.

NOTE

For offline diagnostics, the HSC70 must be booted with the Secure/Enable switch in the ENABLE position. If a hardware error occurs during boot, the software executes a halt instruction on certain errors. A halt instruction, even in Kernel mode, is valid only if the Secure/Enable switch is in the ENABLE position. Otherwise, the result can be an illegal instruction trap in addition to the error causing the halt.

In order for the bootstrap to complete successfully, the following must be operational:

- Basic instruction set of the PDP-11
- First 2048 bytes of Program memory plus 8 Kwords of contiguous Program memory below address 160000
- RX33 controller and at least one drive containing a diskette with a bootable image

Before control is turned over to the HSC70 bootstrap ROMs, internal microcode tests execute in the J11 chip set. Refer to Table 2-1 for definitions of the J11 module (P.ioj) LEDs. Also, refer to Figure 8-4 for details of the P.ioj internal self-test procedures.

6.1.3 P.ioj ROM Bootstrap

The HSC70/J11 P.ioj ROM Bootstrap verifies the basic integrity of the P.ioj module, Program memory, and the RX33 controller/drive subsystem. The goal of the bootstrap tests is to test enough of the HSC70 to allow further test loading from the RX33.

The bootstrap test is the first step in the HSC70 initialization process. It is run for every bootstrap or reload of the HSC70 operating system (CRONIC). The bootstrap is initiated automatically each time the HSC70 is powered on and is also initiated by CRONIC when a software reboot is required.

The bootstrap is a PDP-11 program written to execute in a DCJ11 CPU in a stand-alone environment. This means no other software processes co-exist with the bootstrap.

Bootstrap failures are reported via the Fault lamp mechanism which specifies the module most likely causing the problem. The fault codes are defined in Figure 4-2. An error table is maintained in Program memory addresses 00000400 through 00000412. These addresses contain the reasons for each RX33 drive boot failure.

6.1.3.1 Bootstrap Initialization Instructions - The following procedure lists the operating instructions for the P.ioj ROM Bootstrap. Operating instructions for the P.ioj ROM bootstrap are in the following list. Refer to Section 6.1.3.2 if this procedure fails.

1. Insert a Offline Diagnostics diskette with a bootable image into the RX33 unit 0 drive (left-hand drive).

2. Turn power ON.

3. Set the Secure/Enable switch to the ENABLE position, then depress the Init switch. The bootstrap will initiate automatically.

At this point, the J11 P.ioj module executes internal microdiagnostics and then begins to execute from the boot ROM. The Init lamp lights on the HSC70 operator control panel when the bootstrap PDP-11 tests are done. The RX33 drive-in-use LED should light within 8 to 10 seconds, indicating the bootstrap is attempting to load software into Program memory. If the load is successful, the bootstrap transfers control to the first instruction of the image just loaded from the diskette.

6.1.3.2 Bootstrap Failures - Most bootstrap failures result in lighting the Fault lamp on the HSC70 operator control panel. When this happens, depress the Fault switch momentarily, and read the failure code displayed in the operator control panel lamps. Section 6.1.3.5 indicates the HSC70 modules most likely causing the bootstrap failure. Momentarily depressing the Init switch on the operator control panel reinitiates the bootstrap.

The microdiagnostic LEDs on the J11 module indicate if a hard fault exists causing the J11 to hang before control is passed the boot ROM. Section 6.1.3.5 contains an explanation of these LEDs.

If a failure occurs in the tests of the PDP-11 basic instruction set, the Fault lamp mechanism does not report the failure. Instead, the PDP-11 executes a Branch dot (BR .) and does not continue the bootstrap program. A failure of this type is easily detected because the Init lamp does not light. (The Init lamp does light immediately after the basic PDP-11 tests successfully complete.)

When a console terminal is connected to the P.ioj, the exact instruction that failed is determined by depressing the terminal BREAK key and noting the address displayed on the terminal. With a bootstrap listing, this address indicates the instruction that failed. Notify Field Service Support to investigate such failures.

NOTE

The bootstrap does not accept user-modifiable flags.

6.1.3.3 Bootstrap Progress Reports - The bootstrap does not issue progress reports in the usual sense; however, certain indications of bootstrap progress are shown in the following list:

- **Lamps Clear** - clears all of the HSC operator control panel lamps. If the lamps fail to clear immediately after the bootstrap is initiated, a failure of the P.ioj is probable. (Circuitry on the P.ioj module is responsible for initiating the bootstrap program.)

- **Init Lamp** - lights as soon as the basic tests of the PDP-11 instruction set are finished. These tests normally complete within milliseconds after the bootstrap is initiated. Failure of the Init lamp to light indicates a failure in the P.ioj PDP-11 processor.

- **RX33 Drive-in-Use** - lights as the bootstrap tries to load the Init P.ioj Test (or Offline P.ioj Test) from the RX33 following the test of the PDP-11 and Program memory.

- **State Lamp** - lights when the bootstrap completes and initiates the Init P.ioj Test (or Offline P.ioj Test). When the State lamp is ON, the Init lamp is OFF.

- **Fault Lamp** - lights during the boot process if the ROM bootstrap tests have detected a fatal error (Section 6.1.3.4).

6.1.3.4 Bootstrap Error Information - Specific error codes for the P.ioj bootstrap (Codes 21, 22, and 23) are described in detail in Chapter 4.

Because the bootstrap operates in a stand-alone environment, it does not use the terminal as an error reporting mechanism. Instead, the HSC70 operator control panel lamps are used to report errors and to indicate the module most likely causing the error.

When the bootstrap detects an error, it lights the Fault lamp on the operator control panel. When the Fault switch is depressed, the bootstrap displays a failure code in the operator control panel lamps. The failure code blinks on and off at one-half second intervals.

6.1.3.5 Bootstrap Failure Troubleshooting - The ODT program (built into the PDP-11 microcode) contains further information about bootstrap failures. This information is shown in the following list.

- **Init is Off, Fault is Lit** - a failure was detected after control was passed to the bootable image loaded from the diskette.

- **Init and Fault Both Lit** - the fault code displays when the Fault lamp is momentarily depressed. The program is halted by depressing the BREAK key on the console terminal. Now type: 172340/. ODT responds by displaying the contents of address 172340, the test number. Use the test number to refer to the appropriate test in Section 6.1.4.

- **Init and Fault Lamps are Both Off** - either the bootstrap program was not automatically initiated, or the bootstrap PDP-11 instruction test failed.

  Before proceeding, ensure the Secure/Enable switch is set to the ENABLE position. If the switch was not in the ENABLE position when the Init switch was depressed, the HSC70 did not initiate its boot sequence. If the Secure/Enable switch is in the correct position, the J11 microdiagnostics may have failed.

To check the microdiagnostics, remove the card cage cover and examine the four LEDs on the central edge of the J11 module. At powerup, all the LEDs should be set and then turned off as the J11 proceeds through its microdiagnostic sequence. When viewed from the edge of the P.ioj module, the LEDs ON or OFF are as follows:

ODT LED - Lit while in console ODT.

SLU LED - Lit when SLU failed to respond at 1777560 (console UART present).

MEM LED - Lit when Program memory did not respond during microdiagnostics.

SEQ LED - Lit when very basic J11 internal sequence test failed.

LED SEQ	LED MEM	LED SLU	LED ODT	PROBABLE FAILURE CAUSE
ON	ON	ON	ON	P.ioj
OFF	ON	OFF	OFF	M.std2 first, then P.ioj
OFF	ON	ON	OFF	P.ioj
OFF	OFF	ON	OFF	P.ioj
OFF	OFF	OFF	ON	P.ioj

6.1.4 Bootstrap Test Summaries

This section summarizes the bootstrap tests:

- **Test 0 - Basic PDP-11 Instruction Set** - This test verifies the correct operation of a PDP-11 instruction subset. This instruction subset includes only those instructions required for completion of the bootstrap. The following instructions are tested:

  - Single Operand Instructions Tested (both word and byte mode):

    ADC,CLR,COM,INC,DEC,NEG,TST,ROR,ROL,ASR,ASL,SWAB,NOP

  - Double Operand Instructions (both word and byte modes):

    MOV,CMP,BIT,BIC,BIS,ADD,SUB

  - Branch Instructions Tested:

BR,BNE,BEQ,BPL,BMI,BCC(BHIS),BCS(BLO),

BGE,BLT,BGT,BLE,BHI,BLOS,BVC,BVS

- Jump and Miscellaneous Instructions Tested:

  JMP,JSR,RTS,SOB,MTPS,MFPS,

  CCC,CLN,CLV,CLZ,SEN,SEV,SEZ

- Addressing Modes Tested:

  All eight addressing modes

The PDP-11 instruction set test uses two methods of reporting errors. During the initial part of the test, errors result in an infinite program loop at the location of the detected error. During the latter part of the test (when enough instructions have been tested), the Fault lamp mechanism is used to report failures. Refer to Section 6.1.3.2.

o **Test 1 - Program Memory (Swap Bank)** - The HSC70 memory module includes special logic that permits changing the address range of Program memory. This address range is controlled by the Swap Banks bit in the P.ioj Control and Status Register (CSR). This test verifies the Swap Banks bit can be set and cleared. (The actual memory switching is not tested, only the setting and clearing of the bit is tested.) A failure in this test indicates the P.ioj module must be replaced.

o **Test 2 - Program Memory (Vector Area)** - In order for the HSC70 Control Program to function, the first 2048 bytes (addresses 00000000 through 00003777) of Program memory must be working. This test verifies the first part of Program memory is operating properly. If the test fails, the SWAP BANKS feature is used, attempting to swap a portion of memory into the 00000000 through 00003777 address range. If the test still fails after SWAP BANKS has been invoked, a Program memory error is reported via the Fault lamp mechanism (Section 6.1.3.2). A failure in this test indicates the M.std2 module must be replaced.

o **Test 3 - Program Memory (8 Kword Partition)** - After verifying the first part of Program memory is working, the bootstrap tries to find a 8 Kword piece of Program memory between address 00004000 and address 00160000. This partition is used to load the Init P.ioj Test from the RX33. If insufficient memory is available, a Program memory error is reported via the Fault lamp mechanism.

A failure in this test indicates the M.std2 module must be replaced.

- **Test 4 - RX33 Controller Test** - This test verifies basic functionality of the control logic on the M.std2 module. The four controller registers are tested for stuck bits. The DMA hardware is checked for correct cycling and addressing. The interrupt logic is checked to ensure interrupts are properly acknowledged. With the control hardware verified, proceed to the next step, and try to read data from one of the drives.

- **Test 5 - RX33 Drive/Interface Test** - The goal of this test is to find a working RX33 drive containing a diskette with a bootable image. Such an image is identified by a PDP-11 NOP instruction in the first word of the image. The intended drive is checked for DRIVE READY from the interface. Then RECAL/VERIFY commands the drive to seek to track zero. This command then reads the diskette header to verify the recal did move the head to track 0.

  After a suitable drive is found, the first eight blocks of the diskette are loaded into the 8 Kword partition found in Test 3. The eight blocks loaded consist of the first five blocks of the Init P.ioj Test (or Offline P.ioj Test), the RT-11 Volume ID block, and the first RT-11 directory segment on the diskette. (The directory blocks are loaded at this time to save directory look-up time in the Init P.ioj Test or the Offline P.ioj Test.)

  RX33 drive 0 is tested first. A failure with drive 0 causes the bootstrap to proceed to drive 1 and begin the tests again. If neither RX33 drive is working correctly, an RX33 error is displayed by the Fault lamps. An error table is maintained in Program memory addresses 00000400 through 00000412 which remembers why each rejected RX33 drive failed the boot.

  The error table follows:

Table 6-1 Error Table

Address	Meaning
00000400	: Contains controller error code (code 1 or code 2)
00000402	: RX33 address being accessed, if applicable
00000404	: Expected result
00000406	: Actual result
00000410	: Drive error code, byte-encoded: Drive 1/Drive 0 (high-byte/low-byte)

NOTE

It is not possible to simultaneously have information in addresses 00000400 and 00000410.

If the boot fails with a RX33 error, the ODT feature of the PDP-11 is used to examine the RX33 error table to determine why each RX33 drive failed the test. (Remember the bootstrap tries both drives before declaring an error.) Use following to examine the RX33 error table:

- Depress the BREAK key on the console terminal.

  The terminal should type out the address of the current instruction of the bootstrap, and then prompt for input with an @ character.

- Type nnn (appropriate address).

  The terminal should print the (octal) contents of that address.

- Type linefeed to examine Table 6-2.

Table 6-2 RX33 Error Code Table

Controller Error	Failure Information
1	NXM occurred while accessing RX33 registers.
2	A bit was stuck in the registers. See expected/actual for more information.
3	Force mode interrupt did not occur.
4	DMA test mode hardware error occurred.
5	DMA address counters were wrong after transfer.
6	Incorrect data found after DMA test operation.
7	Data parity was bad after DMA test operation.

10	Drive was not ready (no diskette inserted or door was open).
11	Hard error (CRC or Record Not Found) occurred on recal/verify.
12	Track 0 bit was not set after recal.
13	SEEK command timeout occurred.
14	Seek error (CRC or Record Not Found) occurred.
15	READ SECTOR command timeout
16	Hard error (CRC or Record Not Found) occurred on read.
17	Nonbootable image (non-NOP instruction) is the first word.

Failure information for both drives in address 00000410 is possible. In this case, nonzero data is in both bytes. Only when failures are detected on both drives does the boot ROM generate a LOADFAL failure code and branch to the fault light routine.

- **Test 6 - Transfer Control to Loaded Image** - This part of the bootstrap is not actually a test. However, it is given a test number in case an error occurs in this section of code. The PDP-11 general registers are loaded with certain parameters (CSR and unit of load device, base address, and size of partition, etc.). The image loaded from the RX33 is initiated by jumping to the first instruction. Any errors occurring in this part of the bootstrap are probably unexpected traps or interrupts caused by intermittent P.ioj or M.std2 failures. When the loaded image is started, the State lamp is lit, and the Init lamp is turned off.

6.1.5 Offline Diagnostics Error Reporting And Message Format

The method of reporting errors and the message format are common to the offline diagnostics. All errors are reported on the console terminal as they occur. In all offline diagnostics, error messages conform to the HSC diagnostic error message format.

The first line of an error message contains general information concerning the error and is mandatory. The second line of an error message consists of text describing the error and is also mandatory. The third and succeeding lines of the message are used for additional information where required, and are optional.

The generic error message format follows:

```
XXXXXX>hh:mm Tn En U000
 SEEK error detected during positioning operation
 optional line 1
 optional line 2
 optional line 3
```

where:

XXXXXX> is the prompt for the particular diagnostic in question (such as OFLCXT> or OBIT>, hh:mm is the number of hours and minutes since system boot, tn is a test number in the range of the number of tests in the specific test, en is an error number with a range of 1 through 77 (octal), and U000 is the unit number. The final field in the first line appears only in diagnostics where such information is appropriate. Each error number has a unique text string associated with it. For errors that consist of results that did not compare with the expected value, the diagnostic uses the optional lines to show expected/actual (EXP/ACT) data. Errors on data transfers and SEEK commands use the optional lines to print out the LBN, track, sector, and side to help isolate problems to the media or the drive.

6.2 OFFLINE DIAGNOSTICS LOADER

The Offline Diagnostic Loader provides a software environment for the HSC70 Offline diagnostics. The Loader supports a command language that loads and executes an offline diagnostic from the RX33 into Program memory. The Loader command language also permits the display and modification of any address contents in the HSC70 Program, Data, or Control memories.

The software environment provided for Offline diagnostics includes a RX33 driver and a terminal driver. A standard software interface between the diagnostics and the RX33 and terminal devices takes the place of individual interface routines within the diagnostics. The Loader also maintains a timer that keeps track of the relative time since the Loader was last booted. This allows diagnostic error messages to be time-stamped.

6.2.1 Offline Diagnostic Loader System Requirements

Hardware required to run the Offline Diagnostic Loader includes:

- I/O control processor module with HSC70 Boot ROM.
- At least one M.std2 (memory) module.
- RX33 controller with at least one working drive.
- Terminal connected to I/O control processor console interface.

6.2.2 Offline Diagnostic Loader Prerequisites

In the process of loading the Offline Diagnostic Loader, several diagnostics are run. The ROM Bootstrap tests the basic PDP-11 instruction set, tests a partition in Program memory, and tests the RX33 used for the boot. Then the bootstrap loads the Offline P.ioj Test which completes the PDP-11 tests and the remainder of the I/O control processor module tests.

After these tests, the Offline Diagnostic Loader is loaded from the RX33 to memory and control is passed to the Loader. Due to the sequence of tests that precede the Loader, the Loader assumes the I/O control processor module and the RX33 are tested and working.

6.2.3 Operating Instructions For The Offline Diagnostic Loader

Follow these steps to start the Offline Loader:

1. Insert the HSC70 Offline diagnostics diskette into the RX33 Unit 0 drive (left-hand drive).

2. Power on the HSC70, or depress and release the Init button on the HSC70 OCP.

3. The RX33 drive-in-use LED should light within a few seconds, indicating the Bootstrap is loading the Offline Diagnostic Loader to Program memory.

4. In less than 30 seconds, the Offline Diagnostic Loader indicates it has loaded properly by displaying the following:

```
HSC70 OFL Diagnostic Loader, Version Vnnn
Radix=Octal,Data Length=Word,Reloc=00000000
ODL>
```

5. The Offline Loader is now ready to accept commands. Section 6.2.4 contains information on the Loader command language.

6.2.4 Offline Diagnostic Loader Commands

The following list describes the commands recognized by the Offline Loader. Section 6.2.5.2 of this document is a copy of the Offline Loader Help file.

6.2.4.1 Offline Diagnostic Loader HELP Command - The HELP command supplies an abbreviated list of all commands the Loader recognizes. In response to the HELP command, the Loader reads the file OFLLDR.HLP from the RX33 and displays the contents of this file on the HSC70 console terminal. Section 6.2.5.2 contains a listing of the Loader Help file.

6.2.4.2 Offline Diagnostic Loader SIZE Command - The Offline System Sizer is invoked by the SIZE command. The Sizer determines the sizes of the HSC70 Program, Control and Data memories, and the type of requestor in each HSC70 requestor position. (The requestor position refers to the priority of a particular requestor on the Data and Control memory buses. It does not match the numbering of module slots.)

6.2.4.3 Offline Diagnostic Loader TEST Command - The Offline Diagnostic Loader TEST Command is used to invoke the various offline diagnostics available on the HSC70. The following list shows the particular form of the TEST command used to invoke each diagnostic. In general, the TEST command format allows specification of the system component to be tested. For instance, the TEST MEMORY command invokes the Offline Memory Test.

- **Offline Cache Test** - verifies the full functionality of the onboard cache. The Offline Cache Test is invoked by the TEST CACHE command.

- **Bus Interaction Test** - is invoked by the TEST BUS command. The Bus Interaction test generates contention on the HSC70 Data and Control memory buses by two or more Ks simultaneously testing different sections of the Control and Data memories. Two or more working requestors are required to run this test (including the K.ci).

- **K Test Selector** - is invoked by the TEST K command. The K Test Selector allows you to run specific requestor microdiagnostics.

- **K/P Memory Test** - is invoked by the TEST MEMORY BY K command. The K/P Memory test uses one of the HSC70 requestors to test either Data or Control memory. This test runs faster than the Offline Memory Test because a requestor is roughly seven times faster than the I/O control processor. Program memory cannot be tested using the K/P memory test as the Ks do not have an interface to the Program memory bus.

- **Offline Memory Test** - is invoked by the TEST MEMORY command. This test uses the I/O control processor to test Program, Control, or Data memories.

- **Offline RX33 Exerciser** - is a combined hardware diagnostic and exerciser for the M.std2/RX33 subsystem of the HSC70. Invoke the Offline RX33 Exerciser by the TEST RX command.

- **Memory Refresh Test** - is invoked by the TEST REFRESH command. The Memory Refresh test allows the refresh feature of the memories to be tested.

- **OCP Test** - is invoked by the TEST OCP command. The OCP (Operator Control Panel) test checks the HSC70 lights and switches. The test requires manual intervention by an operator.

6.2.4.4 Offline Diagnostic Loader LOAD Command - The LOAD command loads a program into HSC70 Program memory without starting it. The command format is LOAD <filename>, where <filename> is the name of any file on the HSC70 OFFLINE diskette. The Loader finds the specified file and loads it into Program memory. This command is useful when you want to patch a program image before starting execution. After the patch is made, the program can be initiated via the START command described next.

6.2.4.5 Offline Diagnostic Loader START Command - The START command initiates the Loader program currently loaded in Program memory. The START command can be used in conjunction with the LOAD command (see preceding section), or it may be used to reinitiate the last loaded offline diagnostic. This saves the time required to reload the program from the RX33. For example, you have previously typed SIZE to initiate the Offline System Sizer program and after the Sizer completes, you wish to run it again. Typing START and then carriage return restarts the Sizer without reloading the program from the RX33 saving many seconds of load time.

6.2.4.6 EXAMINE And DEPOSIT Commands - The EXAMINE and DEPOSIT commands are used to display or modify the contents of any location in the HSC70 Program, Control, and Data memories. Qualifiers (switches) can be used with these commands to display bytes, words, long words or quad words. The radix (octal, decimal, hex) of the displayed data can also be controlled by qualifiers. Alternately, the SET DEFAULT command can be used to set the default data length and radix for all EXAMINE and DEPOSIT commands (Section 6.2.4.6.7).

6.2.4.6.1 Offline Diagnostic Loader EXAMINE Command - The EXAMINE command is used to display the contents of any location in the HSC70 Program, Data, or Control memories. The format of the command is: EXAMINE <address>. The <address> can be a string of digits in the current (default) radix. Certain symbolic addresses are also permitted (see Section 6.2.4.6.3).

```
EXAMPLE:  ODL> E 14017776
            (D) 14017776 125252
```

In the example, the user entered a command to examine the contents of location 14017776. (Notice the EXAMINE command can be abbreviated to a single E.) When the Loader displays the contents of location 14017776, the address is preceded by a (D) indicating the location is within Data memory. The display shows the location contains the value 125252.

6.2.4.6.2 Offline Diagnostic Loader DEPOSIT Command - The DEPOSIT command is used to modify the contents of any location in the HSC70 Program, Control, or Data memories. The format of the command is: DEPOSIT <address> <data>. The <address> can be a string of digits in the current (default) radix. Certain symbolic addresses are also permitted (Section 6.2.4.6.3).

EXAMPLE: ODL> D 14017776 123456

In this example, the user entered a command to store the value 123456 in the contents of address 14017776. The previous contents of this Data memory location are replaced with the value specified in the DEPOSIT command (123456).

6.2.4.6.3 Offline Diagnostic Symbolic Addresses - The four symbols used as symbolic addresses in a DEPOSIT or EXAMINE command are described in the following list.

- **Asterisk (*)** - indicates the Loader is to use the same address as used in the last EXAMINE or DEPOSIT command. For example, if you just examined the contents of address 16012344, and you now wish to deposit the value 1234 into the same address, you can type DEPOSIT * 1234 instead of typing DEPOSIT 16012344 1234.

- **Plus sign (+)** - is also used as a symbolic address. This symbol means the Loader is to use the address following the last address used by an EXAMINE or DEPOSIT command. When the Loader sees a + as an address, it takes the last address used by EXAMINE or DEPOSIT and adds an offset which depends on the current default data length (Section 6.2.4.6.7).

  If the current default data length is a byte, the Loader adds one to the last address. If the default was a word, the Loader adds two to the last address. The offset is four for longword data length and eight for quadword. This feature is useful when examining a number of items stored in successive locations.

  For example, if you are examining a table of words beginning at address 14125234, you would examine the first location by typing EXAMINE 14125234. The next location could now be examined by typing EXAMINE + instead of typing EXAMINE 14125236.

- **Minus sign (-)** - is also used as a symbolic address. It indicates the Loader is to use the address preceding the last address used by either command. When the Loader sees a - symbol as an address, the Loader takes the last address used by an EXAMINE or DEPOSIT and subtracts an offset which depends on the current default data length (Section 6.2.4.6.7.)

  If the current default data length is a byte, the Loader subtracts one from the last address. If the default was a word, the Loader subtracts two from the last address. The Loader subtracts four for longword data length and eight for quadword. This feature is useful in the same way as the + symbol, but examines a table starting at the highest address and proceeding down to lower addresses.

  For example, if you want to examine a table of words that ends at address 14012346, you would examine the last location of the table by typing EXAMINE 14012346. The preceding location in the table could now be accessed by typing EXAMINE - instead of having to type **EXAMINE 14012344.**

- **At symbol (@)** - is used as a symbolic address. This symbol means the Loader should use the data from the last EXAMINE or DEPOSIT command as an address. This feature is useful when following linked lists. For example, you first examine location 123434 which contains a pointer to a linked list. Now, you can type **EXAMINE @** to examine the location pointed to by the first location.

6.2.4.6.4 Repeating EXAMINE And DEPOSIT Commands - When troubleshooting memory problems, continuously executing an EXAMINE or DEPOSIT command is sometimes useful. The REPEAT command is used for this continuous execution. Type **REPEAT,** followed by the EXAMINE or DEPOSIT command to be repeated.

EXAMPLE 1 - Repeating a DEPOSIT command

```
REPEAT DEPOSIT 14017776 125252
or RE D 14017776 125252
```

In this example, the value 125252 is continuously deposited into address 14017776. The format of the DEPOSIT command does not change. The DEPOSIT command is just preceded by the word REPEAT. Also the REPEAT command can be abbreviated to RE.

EXAMPLE 2 - Repeating an EXAMINE command

```
REPEAT EXAMINE 14017776
or RE E 14017776
```

In using this example, you can continuously examine the contents of address 14017776. The format of the EXAMINE command does not change. The EXAMINE command is just preceded by the word REPEAT.

In the example shown, the contents of location 14017776 are displayed continuously on the terminal. This slows down the repetition of the command and wastes paper on hard copy devices. Stop output to the terminal by typing a CTRL/O. However, the Loader also provides a special EXAMINE command qualifier (/INHIBIT) for suppressing output to the terminal. This qualifier is discussed in Section 6.2.4.6.6. To stop a repeated command, type CTRL/C.

6.2.4.6.5 Offline Diagnostics Relocation Register - The Loader provides a relocation register. It can be used to reduce the number of address digits typed for an EXAMINE or DEPOSIT command when all addresses are in either the Control or Data memories. The contents of the relocation register are added to the address given with an EXAMINE or DEPOSIT command. The relocation register contains a zero when the Loader is initiated, so it normally has no effect on the addresses typed in an EXAMINE or DEPOSIT command.

If you wish to examine a large number of locations in Data memory, use the following example:

EXAMPLE 1 - Relocation to Data memory

```
ODL> SET RELOCATION:14000000
ODL> EXAMINE 0
 (D) 14000000 123432

ODL> EXAMINE 1234
 (D) 14001234 154323
```

Load the relocation register with the address of the first location in Data memory (14000000). When you issue an EXAMINE command with an address of 0, the Loader adds the relocation register to the address given resulting in the examination of address 14000000. Likewise, when an EXAMINE command with an address of 1234 is issued, the Loader displays the contents of location 14001234.

The following example shows how to examine a large number of locations in Control memory.

EXAMPLE 2 - Relocation to Control memory

```
ODL> SET RELOCATION:16000000
ODL> EXAMINE 0
 (C) 16000000 125252

ODL> EXAMINE 4320
 (C) 16004320 125432
```

The relocation register is loaded with the address of the first location in Control memory (16000000). When an EXAMINE command is issued with an address of 0, the Loader adds the relocation register to the address given, displaying the contents of address 16000000. Likewise, when the user issues an EXAMINE command with an address of 4320, the Loader displays the contents of location 16004320.

6.2.4.6.6 Offline Diagnostics EXAMINE And DEPOSIT Qualifiers (Switches) -

- **/NEXT** - allows an EXAMINE or DEPOSIT command to work on successive addresses. When used with a valid EXAMINE command, it specifies that after the command location has been displayed, the Loader should also display the next number of locations following the first. For example, the command E 1000/NEXT:5 results in the display of locations 1000, 1002, 1004, 1006, 1010, and 1012 (assuming the default data length is a word). The number of the argument can be any value in the current default radix that can be contained in 15 binary bits or less. For instance, if the default radix is octal, the number of the argument can be any value between 1 and 77777.

  The /NEXT qualifier works the same way for the DEPOSIT command, except that the data given with the DEPOSIT command are stored in the location specified and the next number of locations following.

- **/BYTE/WORD/LONG/QUAD** - are used to control the data-length of examined or deposited data. Normally, the Loader uses the default data-length (Section 6.2.4.6.7) when data is examined or deposited. However, the data-length qualifiers can be used to override the default for a single examine or deposit. For instance, assume the default data-length is currently a word, and you wish to examine a byte quantity at address 16001234. The command EXAMINE 16001234/Byte followed by a carriage return would display the proper byte without affecting the default data length.

- **/OCTAL/DECIMAL/HEX** - can be used with an EXAMINE command to control the radix of the address and data displayed. They are NOT used to control the radix of the address supplied in the EXAMINE command. The radix of the address and data displayed by an EXAMINE command is usually controlled by the current Default Radix (Section 6.2.4.6.7), but the /BYTE/WORD/LONG/QUAD qualifiers are used to override the default radix for a single EXAMINE command. For example, assume the default radix is octal. The command EXAMINE 14001234/Hex followed by a carriage return displays the contents of address 14001234(8) in the hexadecimal radix. The EXAMINE display would be as follows: (D) 30029C HHHH. HHHH represents the contents (hex) of the location displayed. The address is also displayed in hex.

- **/INHIBIT** (abbreviated to /INH) - inhibits the display of examined date when repeating an EXAMINE command. This is useful both for saving paper on hardcopy devices and for speeding up the EXAMINE operation for scope-loop purposes. For example, the command REPEAT EXAMINE 16012346/INH results in the Loader continuously reading the contents of location 16012346 without displaying anything at the console.

6.2.4.6.7 Setting And Showing Defaults - The SET DEFAULT command is used to change the default radix and/or data length. The default radix controls the radix of parameters supplied with EXAMINE or DEPOSIT commands and the radix of data displayed by the EXAMINE command. The default data length controls the length (byte, word, long, quad) of data displayed by the EXAMINE command or data stored by a DEPOSIT command.

The default radix may be set to octal, decimal, or hexadecimal. When the Offline Loader first starts, it sets the default radix to octal. Type in **Set Default Hex** followed by a carriage return to set the default radix to hexadecimal. After the default radix is set, it remains so until another SET DEFAULT command is issued or the Loader is rebooted.

The default data length may be set to byte, word, longword, or quadword. When the Loader is first started, it sets the default data length to word (16 bits.) Type in **Set Default Long** followed by a carriage return to set the default data length to longword (32 bits). Setting the default data length to longword causes an EXAMINE command to display longword quantities and causes the DEPOSIT command to store longword quantities. (Because the Loader is executing in a PDP-11, longwords are stored and retrieved as two successive 16-bit words.) After the default data length is set, it remains so until changed by another SET DEFAULT command or until the Loader is rebooted.

6.2.4.6.8 Executing INDIRECT Command Files - The Loader is capable of executing indirect command files stored on the RX33. These command files consist of valid Offline Loader commands terminated by a carriage return (<CR>) and a line feed (<LF>). Comments may also be placed in indirect command files by preceding a comment line with an exclamation mark (!). Comment lines must also be terminated with a <CR> and <LF>. As an example, the Offline Loader Help file is an indirect command file that contains only comments (Section 6.2.5.2).

Indirect command files cannot be created by the Loader or by CRONIC. The command files must be created in RT-11 format and stored on the Offline Diagnostics diskette. Any editor that does not insert line numbers in the output files can be used to create command files.

6.2.5 Offline Diagnostics Unexpected Traps And Interrupts

When the Loader detects an unexpected trap or interrupt, the following message is displayed:

```
Unexpected trap through www, VPC=xxx, PSW=yyy Error Address = zzz
```

where:

```
www = Address of the trap or interrupt vector
xxx = Virtual PC of Loader at time of trap
yyy = Contents of PSW at time of trap
zzz = Address of location causing NXM or parity trap
```

The first line of the unexpected trap report is issued for all unexpected traps or interrupts. The second line is only issued if the trap was through vector addresses 000004 (NXM trap) or 000114 (parity trap). The address of the vector is a direct clue to the cause of the trap. Refer to Section 6.2.5.1 for a list of the devices and error conditions associated with each vector.

The Virtual PC (VPC) of the instruction executing when the trap occurs is sometimes useful in determining the cause of the trap. The VPC can be referenced in the listing to find the instruction causing the trap. Remember, the VPC is the address of the instruction following the instruction executing when the trap occurred. Notify Field Service Support to analyze such failures.

NXM traps can be caused by EXAMINE or DEPOSIT commands if you specify an address not contained in a particular HSC70. For example, if an HSC70 only contains data memory from addresses 14000000 through 14177776, and you try to examine or deposit address 14200000, the Loader reports an NXM trap. In this example, the NXM trap would *not* represent an error condition.

Parity traps can be caused by an EXAMINE command if a user examines an address not initialized with good parity. For

example, when the HSC70 memories are powered on, the parity bits are in random states. Thus, if a user examines a location not written since poweron, the location may generate a parity error. This does not constitute an error condition.

However, if a location produces a parity error and that location has been written since poweron, a memory error is indicated. (Also note the I/O control processor and Ks have bits allowing them to write bad parity for testing the parity circuit. These bits should never be used except by diagnostics.)

6.2.5.1 Offline Diagnostics Trap And Interrupt Vectors - Following is a list of trap and interrupt vectors for various devices and error conditions recognized by the I/O control processor PDP-11 processor:

Vector	Device or Error Condition
000004	Non-Existent Memory, Stack Overflow, Halt in User Mode, and Odd Address Trap
000010	Illegal Instruction
000014	BPT Instruction
000020	IOT Instruction
000024	Power Fail Interrupt
000030	EMT Instruction
000034	TRAP Instruction
000060	Console Terminal - Receiver Interrupt
000064	Console Terminal - Transmitter Interrupt
000100	Line Clock Interrupt
000114	Parity Trap
000120	Control Bus Interrupt - Level 4
000124	Control Bus Interrupt - Level 5
000130	Control Bus Interrupt - Level 6
000134	Control Bus Interrupt - Level 7
000230	RX33 Interrupt
000250	MMU Abort (Trap)
000300	SLU (Serial Line Unit) 1, Receiver Interrupt
000304	SLU (Serial Line Unit) 1, Transmitter Interrupt
000314	SLU (Serial Line Unit) 2, Receiver Interrupt
000310	SLU (Serial Line Unit) 2, Transmitter Interrupt

6.2.5.2 Offline Diagnostics Loader Help File - An example of the Offline Diagnostics Loader Help File follows:

```
!HSC70 OFL Diagnostic Loader Help File - Vnn-nn
!Capital letters = required input, lower case = optional

!COMMANDS (terminated by CR):
! 'Examine <address>'          ;display data at <address> specified
! 'Deposit <address> <data>'   ;deposit <data> to <address>
!      <address> = digit string in current default radix or:
!           '*'  = use same address as last Ex or De
```

```
!           '+'  = use address following last address
!           '-'  = use address preceding last address
!           '@'  = use <data> from last Ex or De as <address>
! ;HElp'                          ;print this file
! '@filename'                     ;execute indirect command file
! 'Load filename'                 ;load file to diagnostic partition
! 'REpeat <command>'              ;repeat specified command until ^C
! 'SEt Default <option>'          ;set default radix or data length
!       <option> = Byte,Word,Long,Quad,Hex,Octal,Decimal
! 'SEt Relocation:#'              ;set relocation register to #

!  NOTE:  Relocation register is 22-bit positive # added to address
!         of all EXAMINE and DEPOSIT commands.

! 'SHow'                          ;display defaults and Loader version #
! 'SIze'                          ;Size HSC70 memories and display K status
! 'Start'                         ;start program in diagnostic partition
! 'Test Bus'                      ;load and start the OFL Bus Test
! 'Test MEmory'                   ;load and start the OFL Memory Test
! 'Test MEmory By K'              ;load and start the OFL K/P Memory Test
! 'Test K'                        ;load and start the OFL K Test Selector
! 'Test OCP'                      ;load and start the OFL OCP Test
! 'Test Refresh'                  ;load and start the OFL Memory Refresh
                                   Test

! QUALIFIERS (switches) for 'Ex' and 'De'
! '/Next:#'                       ;repeat Ex or De on next '#' addresses
! '/Byte,/Word,/Long,/Quad'       ;use specified length vs.  default
! '/Octal,/Decimal,/Hex'          ;use specified radix for Examine display
! '/INHibit'                      ;inhibit display of examined data
! <end of help file>
```

6.3 OFFLINE CACHE TEST

The Offline Cache Test is a diagnostic that runs under the Offline Loader in a stand-alone environment. It provides indepth testing of the cache logic on the J11 P.ioj. It verifies the full functionality of the onboard cache. Execution time for a single pass is between 16 seconds and 4 minutes depending on the options selected.

6.3.1 Offline Cache Test System Requirements

The Offline Cache Test is loaded into memory via the Offline Loader. This test requires 8 Kwords of memory to run. One-half of this memory space contains the program; the other half is used as a cached buffer. All terminal I/O and handling of the line clock is done by the Offline Loader.

6.3.2 Offline Cache Test Operating Instructions

This section contains operating instructions specific to the Offline Cache Test. If the HSC70 is not booted and running the Offline Loader, necessary instructions are found in Section 6.1.2, Section 6.1.3, and Section 6.2. If the HSC70 is already booted and running the Offline Loader, enter the **TEST CACHE** command at the ODL> prompt and press RETURN.

This command loads the Offline Cache Test from the media and transfers control to the diagnostic. When it starts, the Offline Cache Test should display the following:

HSC OFFLINE Cache Test Vxxx

Where Vxxx is a 3-digit version/edit number.

User-modifiable parameters are described in the following section.

6.3.3 Offline Cache Test Parameter Entry

Following are the three user-modifiable parameters for the cache test. In each case the default (invoked by a carriage return) is shown in brackets. If no default is possible, the brackets are empty.

- **Select Data Reliability Test** - is the first user-modifiable parameter, an optional selection of the data reliability tests. It is a moving-inversions style test for exercising the RAM array. The Offline Cache Test prints:

  Run extended cache ram test (Y/N) [N] ?

  Selection of this optional test increases test time per pass to about four minutes. It is useful for the manufacturing burn-in and test areas. It is not necessary to run this optional test in order to fully verify the health of the cache.

- **Leave Cache Enabled** - determines the cache state at the termination of the diagnostic. The Offline Cache Test prints out:

  Leave cache enabled after successful completion (Y/N) [N] ?

  This feature allows enabling the cache for further use after running the diagnostic to verify the cache is working. If the diagnostic detects any hard failures in the cache, it is not enabled at the end of the diagnostic. This prevents complications if the cache contains hard failures and is inadvertently turned on.

- **Number of Passes** - accepts a total number of passes from 1 to 32767 (decimal). The test prompts for this number as follows :

  # of passes to perform (D) [1] ?

  Any decimal number up to 32767 can be used. Fatal errors can cause the diagnostic to terminate before the specified number of passes executes.

At the completion of the total passes requested by the user, the diagnostic prompts:

reuse parameters (Y/N) [Y] ?

Answering this prompt with a Y allows you to rerun the diagnostic with the same parameters as before. Answering with an N causes repetition of the parameter entry questions.

6.3.4 Offline Cache Test Progress Reports

The Offline Cache Test provides summary information at the end of each pass. The end of pass message is similar to this:

End of Pass 00001, 00000 Errors, 00000 Total Errors

The Errors field contains the number of errors for the pass. The Total Errors field contains a running total of errors accumulated since the start of the diagnostic.

6.3.5 Offline Cache Test Error Information

The Offline Cache Test displays the errors detected during execution on the console terminal. All error messages follow the offlines generic error message format (Section 6.1.5) preceded by an OFLCXT> prompt.

Each error number has a unique text string associated with it. For errors with results that did not compare with the expected value, the diagnostic uses the optional lines to show expected/actual data.

Soft errors (such as cache parity errors) can accumulate to a point where the diagnostic classes them as fatal. The test then terminates on a fatal error.

6.3.5.1 Specific Offline Cache Error Messages - The following list describes in detail each possible error message. The errors are listed in numerical order.

- **Error 00 - Memory parity error, VPC = xxxxxx** (Applicable to all tests.) - can occur at any time during execution of the diagnostic. The virtual PC on the stack is printed to help identify the program area where the error occurred. The content of the error address register is also displayed.

  Both the virtual PC and the error address register content are optional lines. Detection of this error causes the testing to cease. Then the diagnostic returns to the **Reuse parameters** prompt.

- **Error 01 - NXM Trap, VPC = xxxxxx** (Applicable to all tests.) - causes the diagnostic to return to the **Reuse parameters** prompt. Additional data (such as the virtual PC of the instruction which caused the trap and the physical address contained in the error address register) are printed as optional lines.

- **Error 02 - Cache parity error, VPC = xxxxxx** (Applicable to Tests 2 through 16.) - results when a trap through the parity error vector is detected and the cache is enabled. The virtual PC where the error was detected is printed, as well as the content of the error address register. If the 22-bit value in the error address register is 177770024, no main memory error was present. You can assume the parity error is from the cache.

- **Error 03 - Bit stuck in cache control register** (Applicable to Test 2.) - indicates a bit is stuck-at-fault in the cache control register. The expected and actual data values are printed as optional lines.

- **Error 04 - Forced miss operation failed.** (Applicable to Test 3.) - bit 2 of the cache control register does not prevent the cache from allocating a test location. This could be a problem in the cache control gate array or in the hit/miss compare logic.

- **Error 05 - Forced miss with abort failed** (Applicable to Test 3.) - bit 3 did not prevent the cache from allocating when set. Failures of this nature mean the cache cannot be disabled, and all memory references may be allocating cache regardless of the intent of the code being executed. The cache control gate array or the tag compare logic may be at fault.

- **Error 06 - Expected cache hit did not occur** (Applicable to Tests 4, 6, 9, 12, and 14.) - did not allocate a given test location to the cache as expected, causing a miss condition in the hit/miss register.

- **Error 07 - Expected cache miss did not occur** (Applicable to Tests 7, 9, and 10.) - shows a test location not expected to be allocated, or valid, as a hit on access.

- **Error 10 - Value in hit/miss register incorrect** (Applicable to Test 5.) - indicates the 6-bit value in the hit/miss register was incorrect after a certain sequence of instructions. The expected values, as well as the actual content of the hit/miss register, are printed as optional lines.

- **Error 11 - Write byte operation caused cache update** (Applicable to Test 6.) - A byte operation (on a miss) did not cause cache to deallocate the test location. Thus, when the test location was read back, a cache hit resulted.

- **Error 12 - Write byte did not cause cache update** (Applicable to Test 6.) - A byte-value did not get written into cache or main memory.

- **Error 13 - Cache failed to flush successfully** (Applicable to Test 8.) - When checking cache after a flush command was executed, one or more locations still contained valid data (were detected as cache hits).

- **Error 14 - Access with force bypass did not cause invalidate** (Applicable to Test 9.) - The second access to an allocated location, with the force bypass bit (bit 9) set in the control register, did not result in a miss as expected.

- **Error 15 - Tag Parity error did not set** (Applicable to Test 10.) - The diagnostic could not set the tag parity error bit in the memory system error register when faced with an actual tag parity error.

- **Error 16 - Abort on cache parity error did not occur** (Applicable to Test 11.) - The cache logic did not abort the instruction under execution when a cache parity error was forced and the abort bit (bit 7) was set in the control register.

- **Error 17 - Unexpected parity trap during abort test** (Applicable to Test 10.) - Although expected to, cache control Bit 0 did not prevent the cache logic from taking a trap on bad parity. The address where the trap occurred is printed as optional information.

- **Error 20 - Content of memory system error register incorrect** (Applicable to Test 11.) - The error bits in the memory system error register (1777744) do not reflect the correct status for the operation under test. The expected and actual content are printed as optional lines.

- **Error 21 - Return PC wrong during abort/interrupt test** (Applicable to Test 11.) - The return PC on the stack is not equal to the value expected during an abort or interrupt operation caused by a cache parity error. The state sequencer gate array is most likely defective.

- **Error 22 - Cache data parity bit(s) did not set** (Applicable to Test 10.) - The diagnostic was unable to set the data parity error bit(s) in the memory system error register on a forced parity error. The parity logic may not be detecting parity errors or one of the bits in the memory system error register may be stuck low.

- **Error 23 - Interrupt on parity error did not occur** (Applicable to Test 11.) - The cache did not interrupt through vector 114 on a forced parity error. The state sequencer or the parity detection logic may be faulty.

- **Error 24 - Expected NXM trap did not occur** (Applicable to Test 13.) - A NXM trap was not detected during an access to location 1777757776. The timeout logic that detects a NXM may be defective, or some problem may exist in the cache data path gate array that prevents it from acting on timeout.

- **Error 25 - Parity error was not blocked by NXM** (Applicable to Test 13.) - When accessing a location expected to result in a NXM, the parity error flag set instead, and a trap occurred through vector 114. The NXM signal may not have been detected by the cache data path gate array.

- **Error 26 - Cache data miscompare on word operation** (Applicable to Test 14.) - A word address in the cache array did not have the correct data when read. This may indicate address line faults or data path faults allowing the location to be rewritten after the test value was placed there. The expected/actual data values are printed as optional lines.

- **Error 27 - Cache data miscompare on byte operation** (Applicable to Tests 14 and 15.) - A location in the cache, when addressed in a byte fashion, did not have the expected data pattern. This may indicate address line faults or data path control faults which allowed overwriting the expected value.

- **Error 30 - DMA write to memory did not cause cache to invalidate** (Applicable to Test 12.) - A DMA write by the RX33 controller to a test location, allocated to cache, still resulted in a hit status after the transfer. The cache has stale data.

- **Error 31 - Instruction still completed during abort condition** (Applicable to Test 11.) - With the abort bit set in the cache control register, an instruction set up to detect a parity error on an operand fetch still finished execution modifying the destination of the instruction.

- **Error 32 - Load device error during DMA test** (Applicable to Test 12.) - The RX33 subsystem did not respond correctly to the DMA test operation. There may be faults in the RX33 controller or the interrupt service logic. This message is informational in nature, and this error is outside the scope of this diagnostic.

- **Error 33 - PDR cache bypass failed** (Applicable to Test 7.) - Setting the PDR bypass bit in the PAR/PDR pair under test did not bypass the cache. This points to a MMU or cache data path gate array problem. The PDR number and the CPU execution mode (Kernel or User) are printed as optional lines in the error message.

- **Error 34 - Tag store address hit failure** (Applicable to Test 16.) - Changing the value of the tag bits (bits 16:22 of the physical address) still resulted in a hit condition (even though the address should not have compared) forcing a fetch to main memory. There may be a problem in the tag RAMs or the tag compare logic in the cache data path is not working.

- **Error 35 - Tag store address miss failure** (Applicable to Test 16.) - When going through the possible values for the tag bits (16:22 of the physical address), the cache failed to allocate for some combination of the bits. Possible problems are stuck bits in the address lines going to the cache array, bad RAMs in the cache array, or a fault in the tag compare logic.

- **Error 41 - Processor type is not J11** (Applicable to Test 1.) - The processor type register does not show the correct value for a J11 chip set. Attempting to run this diagnostic on anything other than a J11 produces this error.

6.3.6 Offline Cache Test Troubleshooting

All of the logic under test is contained on the J11 P.ioj module with the exception of the memory used by the diagnostic. Main memory parity errors usually point to the memory module. Because much of the logic tested is buried within the two gate arrays on the module, troubleshooting is often limited to a best-guess replacement of one or both of these gate arrays.

Cache parity errors and data miscompare errors can usually be traced to specific RAMs if proper attention is paid to the data content and address.

For scope loops, the cache test should be run with a large number of passes, and a CTRL O typed on the console to inhibit error message printout.

Constant hit/miss errors, or tag address hit problems, may also be caused by the tag compare logic, which is separate from the gate arrays and the data path.

6.3.7 Offline Cache Test Descriptions

Following are descriptions of the Offline Cache Tests 1 through 16.

- **Test 1 - Cache Register Access Test** - checks for the presence of the necessary cache control/status registers, the cache control register (1777746), the hit/miss register (1777752), and the memory system error register (1777744). To perform further diagnosis, these registers must respond.

- **Test 2 - Cache Control Register Bits** - tests the read/write bits of the cache control register (1777746) for stuck-at faults. In addition, bits (8,11:15), which are write-only, are checked for read data of zero. Bits 6 and 10 which cause data and tag parity to be written incorrectly on new data allocated to cache are treated as special cases. After writing/reading each of these bits, the cache is flushed to remove any bad parity locations.

- **Test 3 - Force Miss Action** - verifies all references made with either bit 3 or bit 2 of the cache control register set that cause a cache miss and leave the cache entry unchanged. To perform this test, first write a test address with bits 3:2 cleared to allocate cache and place a known data pattern into the cache. Then bit 2 is set, and the same test location is written again. With bit 2 set, the cache will not update, and the data in cache is still considered valid. When bit 2 is cleared, and the test location is accessed again, the old data from cache should be the result. If not, the force miss action of bit 2 did not work. The same sequence is repeated for bit 3, and the same results are expected.

- **Test 4 - Hit/Miss Register Part I** - checks the basic operation of the hit/miss register in logging hit/miss information on instruction fetches and data reads/writes. The hit/miss register is critical to further cache diagnosis, because it is the window into what is actually going on inside the cache.

  First, a test location is allocated with cache enabled. Then cache is bypassed, and the test location is accessed again by a write. This write should go directly to main memory and bypass the cache. The cache is enabled, and a read access to the test location should result in a hit condition in the hit/miss register. Then the test location offset by 8 Kwords is accessed. This should result in a miss, since the upper bits of the address (tag) will not match.

- **Test 5 - Hit/Miss Register Part II** - checks all the combinations of the six bits in the hit/miss register for a single miss at different bit positions. This is done by caching a certain sequence of instructions and executing them, with miss conditions forced at each bit position. At the completion of this test the hit/miss register has been checked for both ones and zeros at each bit position.

- **Test 6 - Byte Accesses** - ensures byte references to the cache are handled correctly by the control logic. The first operation is a write byte to the test location not allocated followed by a byte-read of the test location. The read should result in a miss. Then the entire word at the test location is allocated. The upper byte of the test location is modified, and a cache hit is expected. The entire word is also read and compared against the expected result to see if the byte-write occurred. A similar chain of events follows, this time modifying the low byte.

- **Test 7 - PDR Cache Bypass Test** - tests all of the Kernel PDRs <0:7> as well as the user PDRs. It is very important for the bypass cache bit (Bit 15 of any PDR) to work correctly in the multiprocessing environment of the HSC70.

  To test PDR bypass, select from a table the PAR/PDR pair to test. This PDR is remapped to point to Control memory. Control memory is then written via the MMU writing a data pattern and allocating cache. Control memory windows are used to write Control memory to a second pattern without involving the cache control logic. When Control memory is read through the MMU with the bypass bit set, the actual Control memory content (second pattern) should be the result if the bypass bit is actually set. If the old content (first pattern) is read back, the bypass bit is not working. PARs 1, 2, 3, 5, and 6 are tested in this way.

  PARs 0, 4, and 7 are treated as special cases due to programming environment restrictions. They are tested by allocating cache with some location mapped by the PAR/PDR under test and then setting the bypass bit. When the test location is read, the hit/miss register should record a hit and then invalidate the location. If the location is written or read again, it should result in a miss as long as the bypass bit is set.

  After all the Kernel PAR/PDR registers are tested, the program maps user space identical to Kernel space and switches into User mode to re-execute all the tests. After all User PAR/PDR pairs have been tested, the program swaps back into Kernel mode and proceeds to the next test.

- **Test 8 - Cache Flush Action** - allocates all 4 Kwords of cache, and then executes a flush command by setting bit 8 in the cache control register. The cache control logic then writes every location in cache with the data value 17777746 and resets the valid bit for each location. All 4 Kwords of cache allocated before the flush are read again, and if any location responds with a hit when read, an error is declared.

- **Test 9 - Unconditional Bypass to Main Memory** - checks the correct operation of Bit 9 of the cache control register. Bit 9 is used to bypass cache in a fashion similar to the bypass bits in the PAR/PDRs. Any location allocated in cache before the bypass bit is set results in a hit on the first access, and further accesses all show as misses.

This function is used when it is desirable to temporarily disable the cache in a fashion that does not leave the cache with stale data when re-enabled. A test location is allocated, and then the bypass bit is set. The first access of the test location should be a hit, and the second should be a miss.

- **Test 10 - Force Tag/Data Parity Errors** - forces parity errors in the tag and data fields of the cache array to test the parity detection logic. A special diagnostic mode is used, with bit 0 of the cache control register and one of the force parity error bits set. When bit 0 is set, any trap through 114 is disabled on a parity error detected in cache. If a parity trap does occur, an error is declared.

  First, tag errors are forced using bit 10 in the cache control register. When this bit is set, locations allocated to cache do so with bad tag parity. When accessed again (resulting in a cache hit), they should set the tag parity error bit (bit 5 in the memory system error register). The force data parity error bit (bit 6 of the cache control register) is checked next. After a location is allocated to cache with bad data parity, further reads of that location result in setting the data parity error bits (bits 6:7 of the memory system error register). After using the force bad parity bits, the program flushes the cache to remove these parity errors.

- **Test 11 - Abort/Interrupt on Parity Errors** - uses the force parity error bits in the cache control register to force parity errors in the cache array. Because testing of the detection of such errors has been done, testing of the other logic related to cache data or tag parity errors can be done.

  Different combinations of tag and parity errors are forced, with the cache control register set to interrupt through 114, or abort through 114 on parity errors. An interrupt through 114 should set the correct error bit(s) in the memory system error register. Also, the instruction detecting the parity error should complete.

  On an abort through 114, the correct error bit(s) should be set, but the instruction should not complete. If the parity error is detected on the fetch of the source data, the data in the destination of the instruction is not modified. The PC on the stack after each interrupt or abort instruction is checked against the PC that is expected.

- **Test 12 - DMA Invalidate** - modifies a location resulting in the cache acquiring stale data unless cache logic detects the DMA change. The RX33/M.std2 subsystem is used to generate DMA operations to program memory. A DMA write to a program memory location allocated to cache should result in a cache miss when it accessed after the DMA write.

- **Test 13 - Check Blockage of Parity Error on NXM Abort** - generates simultaneous NXM and parity errors. The NXM trap should occur overriding the parity error.

- **Test 14 - Cache Data RAM Test** - tests the cache data RAMs by mapping one PAR and using the cache solely for data storage. A data pattern to detect dual-addressing is written to the cache. Failures of the cache data to match the expected data on read-back are considered miscompare errors. The test is first done using word addresses and test values, and then repeated with byte addresses and byte data patterns. Each location allocated is expected to be a hit from cache, and the content is checked as well.

- **Test 15 - Tag Store RAM Test** - checks the tag bits of the cache array for dual address errors and stuck-at faults. With the cache flushed and completely deallocated, the first 256 locations of the cache are written with a unique data value in each address. Then the entire cache is read. Only the 256 locations written should be cache hits, and only these locations should have the expected data pattern. Then the upper address bits are changed so a new combination of tag bits results. This test is repeated 15 times until all of the tag bits have been tested.

- **Test 16 - Data RAM Reliability Test** - performs a modified moving inversions test on the cache data RAM array. Due to the geometry of the data RAMs, every fourth bit is done concurrently to save time. This results in using the same pattern in both nibbles of the data word. This test must be selected by the user as it does not normally run by default. About four minutes are required to complete one pass of this test.

6.4 OFFLINE BUS INTERACTION TEST

The Offline Bus Interaction Test creates Control and Data bus contention among the requestors in the HSC70 subsystem. The contention is generated by simultaneously testing different portions of the same memory (Control and/or Data) from different requestors. In the process of testing the memories, the various requestors in the subsystem contend with each other for the use of the Control and Data buses.

In addition to the bus contention generated by the requestors, you can select I/O control processor interaction with the Program, Control, and Data memories, with the Operator Control Panel (OCP), and/or the load device. If I/O control processor interaction is selected, it occurs simultaneously with the bus contention generated by the requestors.

This test requires a minimum of two working requestors in order to operate and uses a maximum of seven requestors if they are available. The more requestors available for use by this test, the greater the amount of bus contention. A larger number of requestors makes it easier to isolate failures to a particular source. Also, the run time of this test increases linearly as the number of requestors is increased.

If the Bus Interaction Test fails, you must first determine if the failure was caused by an interaction problem. Determine this by running the Offline K/P Memory Test (Test Memory By K). When the test prompts for parameters, specify the requestor number of the requestor that detected the failure in the Bus Interaction Test. Also specify the same starting and ending addresses displayed with the error report from the Bus Interaction Test. If the requestor also fails the Offline K/P Memory Test, the original problem was not an interaction problem. The problem should be localized in the same manner as any ordinary memory failure.

6.4.1 Offline Bus Interaction Test System Requirements

Hardware required to run this test is shown in the following list.

- I/O control processor module with HSC70 Boot ROMs
- At least one M.std2 (memory) module
- Working Control and Data memories
- RX33 controller with at least one working drive
- Terminal connected to I/O control processor console interface
- At least two working requestors (K.sdi, K.sti, or K.ci.)

6.4.2 Offline Bus Interaction Test Prerequisites

Booting procedures and testing through successful loading of the Offline Diagnostic Loader program is described in Section 6.1.2, Section 6.1.3, and Section 6.2.

Due to the sequence of tests that precede the memory test, the memory test assumes the I/O control processor module and the load device are tested and working. This test also assumes the Control and Data memories were previously tested with the Offline Memory Test or the Offline K/P Memory Test and are working.

6.4.3 Offline Bus Interaction Test Operating Instructions

At the Loader prompt (ODL>), the operator types the TEST BUS command and the Offline Bus Interaction test is loaded and started. The test indicates it has been loaded properly by displaying the following:

```
HSC OFL Bus Interaction Test
```

The test then sizes the Program, Control, and Data memories and determines the number of requestors available for testing.

6.4.4 Offline Bus Interaction Test Parameter Entry

After displaying the program name and version, the Program, Control and Data memories are sized. The bounds of each memory are displayed on the terminal.

NOTE

> For any of the Bus Interaction Test prompts, use the DELete key to delete mistyped parameters before the terminating carriage return is typed. If you note an error in a parameter already terminated with a carriage return, type a CTRL/C to return to the Offline Loader. Then type START, followed by carriage return, to restart the test from the beginning.

The test prompts you to select the requestors used for the test, as follows:

```
Use requestor  001, K.ci (Y/N) [Y] ?
```

Answer with a carriage return (or a Y followed by a carriage return) if the K.ci should be used. Answer with an N followed by a carriage return if the K.ci should not be used.

At least two working requestors must be used to run the bus contention test because one requestor cannot generate bus contention by itself. The program displays the following error message if less than two requestors remain after you have indicated which requestors should be used:

```
Not Enough Ks Available for Test
```

Next, the program prompts for the type of I/O control processor interaction desired:

P.ioj Memory Interaction desired (Y/N) [Y] ?

Answer the prompt with a carriage return if you want I/O control processor interaction with memory. Answer with an N followed by a carriage return if you do not want I/O control processor interaction with memory. If you answer the prompt with an N, the following three prompts are skipped. If you answer the prompt with a carriage return, the following prompts are displayed:

Interact with Program Memory (Y/N) [Y] ?
Interact with Control memory (Y/N) [Y] ?
Interact with Data Memory (Y/N) [Y] ?

For each prompt, answer with a carriage return if you want the I/O control processor to interact with the specified memory while the requestors are generating contention on the Control and Data buses. Answer with an N followed by a carriage return if you do not want the I/O control processor to interact with the specified memory. (If I/O control processor interaction is selected, the I/O control processor interacts with the memory at the same time the requestors are generating Control and Data bus contention.) The program next prompts for OCP interaction:

OCP Interaction Desired (Y/N) [Y] ?

If you want I/O control processor interaction with the OCP, answer with a carriage return. If you do not want OCP interaction, answer with an N, followed by a carriage return. The test then prompts for load device interaction:

Interact with load device (Y/N) [Y] ?

If you want I/O control processor interaction with the load device, answer with a carriage return. If you do not want such interaction, answer with an N, followed by a carriage return. The program then prompts:

Number of passes to perform (D) [1] ?

Enter a decimal number between 1 and 2,147,483,647 (omitting commas), to specify the number of times the bus interaction test should be repeated. (Entering a 0, or just a carriage return, causes one pass of the test.) After the number of passes is entered, the bus contention test begins. The test can be aborted at any time by typing a CTRL/C. (The test may continue running for a few seconds after CTRL/C is typed.)

After the specified number of passes is completed, the following prompt is issued:

```
Reuse parameters (Y/N) [Y] ?
```

To repeat the last test specified using the parameters, answer this prompt with a carriage return or a Y followed by a carriage return. To cause the test to prompt for new parameters, answer the prompt with a N followed by a carriage return. Answering the prompt with CTRL/C returns control to the Offline Loader.

6.4.5 Offline Bus Interaction Test Progress Reports

Each time the program completes one full set of bus contention tests, an end of pass report is displayed. A pass consists of completing a full set of contention tests, including: Control Bus Tests, Data Bus Tests and Combined Control and Data Bus Tests. The end of pass message is displayed as follows:

End of Pass nnnnnn, xxxxxx errors, yyyyyy total errors.

where:

nnnnnn = decimal count of the number of passes completed

xxxxxx = decimal count of the number of errors detected on current pass

yyyyyy = decimal count of the total number of errors detected since the test was initiated

6.4.6 Offline Bus Interaction Test Error Information

All error messages produced by this test conform to the generic diagnostic error message format (Section 6.1.5). Following is a typical Bus Interaction Test error message:

```
OBIT>hh:mm T aaa E bbb U-000
Memory Test Error
Detected By K.sdi, requestor 006
MA -xxxxxxxx
EXP-yyyyyy
ACT-zzzzzz
<K-Error-Summary-Info>
Memory Test Configuration:
K.ci , requestor 001, M.ctl 16000700 - 16100274
K.sdi ,requestor 006, M.ctl 16100300 - 16177674
```

where:

```
hh       = Hours since Offline Loader was last booted
mm       = Minutes since Offline Loader was last booted
aaa      = decimal number denoting test
bbb      = decimal number denoting the error detected
xxxxxxxx = Address of location causing the error
yyyyyy   = Data that was expected
zzzzzz   = Data that was actually found
```

<K-Error-Summary-Info> = Refer to Section 6.4.6.1
Memory Test Configuration = Refer to Section 6.4.6.2

6.4.6.1 Requestor Error Summary - When the requestor reports a memory test failure to the I/O control processor, the following information is supplied:

1. Address of the failing memory location

2. Data expected and data actually found

3. Error summary information

The error summary information is supplied as a 3-bit field including:

1. A bit indicating a parity error occurred while reading the location

2. A bit indicating an NXM error occurred while accessing the location

3. A bit indicating a Control Bus (CBUS) error occurred while accessing the location

When a memory error report is issued for an error detected by the requestor, the last line of the error report includes a list of the error summary bits that were set (if any).

A Control Bus (CBUS) Error indicates the requestor asserted an illegal combination of the three CCYCLE lines when accessing Control memory. Because these lines were previously tested from the I/O control processor (in the OFL P.ioj Test), a Control Bus Error is probably caused by a problem with the requestor's drivers that assert the CCYCLE lines.

6.4.6.2 Offline Bus Interaction Memory Test Configuration - The memory test configuration lists each requestor being used for bus interaction tests along with the section of memory each requestor was testing when the failure occurred. The configuration information consists of:

1. Type of requestor (K.ci, K.sdi, K.sti) and the requestor number

2. Memory being tested by the requestor (M.ctl = Control memory, M.data = Data memory)

3. First address of the chunk of memory being tested

4. Last address of the chunk of memory being tested

6.4.6.3 **Offline Bus Interaction Test Error Messages** - The following list describes the nature of the failure indicated by each error number:

- **Error 000 - Memory Test Error** - indicates one of the requestors detected a memory error in the Control or Data memories. The following is a sample error report:

```
Memory Test Error
Detected by K.ci, requestor 001
MA -16010234
EXP-000177
ACT-000377
Parity error
Memory Test Configuration:
 K.ci ,requestor 001, M.ctl 16000700 - 16100274
 K.sdi ,requestor 007, M.ctl 16100300 - 16177674
```

  MA = The 22-bit address of the failing location.
  EXP = The data pattern expected by the requestor
  ACT = The data pattern found by the requestor
  Memory Test Configuration = other requestors enabled when failure occurred.

  This sample error report indicates the K.sdi detected a memory parity error while reading address 16010234 of Control memory (M.ctl). The requestor expected to find the value 000177 in the location but instead found the value 000377. At the time the error occurred, the K.ci in requestor 1 was testing addresses 16000700 through 16100274 of Control memory, and the K.sdi in requestor 7 was testing addresses 16100300 through 16177674 of the Control memory.

- **Error 001 - K Timed-out During Init** - is displayed when a requestor fails to complete its Init sequence in time. This error usually indicates the specified requestor failed one of its internal microdiagnostics. A sample error report follows:

```
K Timed-out During Init
K.ci , requestor 001, Status = 104
Other Ks Enabled:
 K.sdi, requestor 6
 K.sdi, requestor 7
```

  This sample error report indicates the K.ci in requestor 1 did not finish its initialization diagnostics in the required time. The requestor status displayed with the error report indicates the requestor failed test 4 of its microdiagnostics (1XX in status = failed test XX). Two other requestors were enabled at the time the requestor K.ci timed-out. (One of these requestors may be responsible for K.ci time-out.)

When the I/O control processor enables the requestor to perform the memory test, the requestor begins its initialization sequence (which includes executing certain microdiagnostics). At the end of the requestor's Init sequence, the list indicates it found the K Control Area by complementing a pointer word in Control memory. If the requestor fails to complement this pointer word within 50 milliseconds (4.2 seconds for the K.ci) of being enabled, error 001 is reported. The contents of the K Status Register are displayed with the error report.

- **Error 002 - K Timed-out During Test** - indicates the specified requestor failed to complete its memory test within the expected time. A sample error report follows:

```
K Timed-out During Test
K.sdi, requestor 007, Status = 002
Memory Test Configuration:
K.ci , requestor 1, M.ctl 16000700 - 16100274
K.sdi, requestor 7, M.ctl 16100300 - 16177674
```

The sample error report indicates the K.sdi in requestor 7 never completed the memory test it was assigned. (Ks are allowed up to one minute to complete a memory test.) The memory configuration displayed with the error report shows all Ks testing at the same time the K.ci timed-out. In this example, the K.ci in requestor 1 was also testing at the time the K.sdi timed-out.

Test time-out failures may be caused by a failure in the requestor that timed-out. They may also be caused by a failure in one of the other requestors that was testing at the same time.

- **Error 003 - Parity Trap** - indicates the I/O control processor detected a parity error. The 22-bit address of the location causing the error is displayed as the MA data in the error report, where:

MA = The address causing the parity trap.
VPC = The Virtual PC of the memory test at the time the trap occurred. Reference this address in the listing to locate the area of the test where the error occurred.

The data is lost when a parity trap occurs so no expected or actual data can be displayed.

- **Error 004 - NXM Trap** - indicates the I/O control processor detected a Non-Existent Memory (NXM) error. An NXM error is caused when no memory responds to a particular address. The MA data in the error report indicates the address which produced the NXM trap.

After the trap is reported, the program attempts to restart the test from the beginning. The MA and VPC fields have the same meanings as Error 003.

If this error occurs at a memory address that should be in your memory configuration, the memory in question is not supplying an ACK to the I/O control processor when the specified address is presented on the memory bus. The most probable point of failure is the logic on the memory module that compares addresses on the memory bus with the range of addresses to which the module should respond. Also, the comparator itself could be faulty or the [C IN, C OUT], [D IN, D OUT] or [P IN, P OUT] lines on the backplane could be in error.

- **Error 005 - Memory Test Error (P.ioj Detected)** - indicates the I/O control processor detected an error while testing Program memory. This error can only occur if I/O control processor interaction with Program memory is selected. This interaction consists of:

  1. A series of PDP-11 instructions that perform Read/Modify/Write (RMW) cycles to selected Program memory locations.

  2. Quick-verify tests of the entire Program memory (done 6 Kwords at a time).

  Error 005 can be caused by cross-talk between the Program memory bus and either the Control or Data bus. It can also be caused by a failure in the Program memory logic which inhibits refresh cycles in the middle of a RMW cycle.

NOTE

Errors 006 through 009 are HSC50 specific and do not apply to the HSC70.

- **Error 010 (12 octal) - Cache Parity Trap, VPC = xxxxxx** - can happen during any test. The J11 trapped through the parity vector. The error was caused by the cache.

NOTE

Errors 011 through 017 can occur on an HSC70 when load device interaction is enabled.

- **Error 011 - RX33 Drive Not Ready** - indicates the drive selected for the operation was not ready. The door may be open or the diskette absent during a READ or POSITION command.

- **Error 012 - RX33 CRC Error During Seek** - indicates the RX33 detection a CRC error during a seek. The RX33 could not verify position when reading header information from the diskette.

- **Error 013 - RX33 Track 0 Not Set on Recalibrate** - indicates a recalibrate (seek to track 0) operation is performed before each block of read operations. If the RX33 does not show correct status after the recal command, error 013 is printed.

- **Error 014 - RX33 Seek Timeout** - prints if during a seek the RX33 does not respond by interrupting.

- **Error 015 - RX33 Seek Error** - sets the seek error bit (Bit 4 of the CRS). At the end of a Seek operation, the RX33 found out it is not where it thought it should be.

- **Error 016 - RX33 Read Timeout** - indicates the RX33 did not interrupt at the end of a READ command.

- **Error 17 - RX33 CRC/RNF Error on Read Command** - can be caused by a soft error or bad spot(s) on the disk. For informational purposes, the following additional message prints out:

  First LBN In Transfer = xxxx

  where:

  xxx is the LBN of the first block in the transfer. The Offline Interaction Bus Test performs reads in blocks of four.

6.4.6.4 Offline Bus Interaction K Memory Test Algorithm - The Moving Inversions Memory Test (MOVI) is used to generate bus contention among the requestors. Each requestor in an HSC contains the Moving Inversions test as part of its microdiagnostic software set. The Moving Inversions RAM test is used to detect data and addressing problems in dynamic semiconductor memories.

The following are the steps in the Moving Inversions Algorithm:

1. Write 000000 in each location being tested.

2. Read all locations in order from lowest to highest. After reading a location and checking for a zero, rewrite the same location with a single one in the least-significant bit. Then reread the location and verify the write worked correctly.

3. Again, read all locations in order from lowest to highest, checking to see each location contains the data previously written. Then rewrite the data found with a single additional one bit and reread to check that the write worked properly.

4. Repeat step 3 until the test pattern consists of a word containing all ones (pattern 17777777).

5. Repeat steps 1 through 4, but this time start at the highest memory address each time and work down to the lowest. However, instead of adding an additional 1, add an additional 0. This changes each memory location from all ones back to all zeros.

6. End of test. All memory is cleared to 000000.

6.5 OFFLINE K TEST SELECTOR

The Offline K Test Selector allows you to command a K to perform an internal microdiagnostic self-test. This Offline K Test executes from the P.ioj and uses the HSC K Control Area for instruction. You select the K for testing and the test number of the microdiagnostic test for execution.

6.5.1 Offline K Test Selector System Requirements

The following hardware is required to run this test:

- P.ioj (processor) module with HSC Boot ROMs
- M.std2 (memory) module
- A working section of Control memory for use as a K Control Area
- One working RX33 drive
- Terminal connected to the P.ioj console interface
- At least one working K (K.sdi, K.sti, or K.ci)

Due to the sequence of tests that precede this test, you can assume the P.ioj, Program memory, and RX33 are working.

6.5.2 Offline K Test Selector Operating Instructions

If the HSC70 is not booted and loaded, refer to Section 6.1.2, Section 6.1.3, and Section 6.2. If the Loader prompt (ODL>) is displayed, follow these steps to start the K Test Selector:

1. Type the TEST K command. The RX33 drive-in-use LED lights as the test is loaded.

2. The test indicates it has been loaded properly by displaying the following:

   HSC OFL K Test Selector

3. The test next prompts for parameters.

6.5.3 Offline K Test Selector Parameter Entry

This section gives detailed information on how to enter the test parameters for the Offline K Test Selector. Items in square brackets are the default value for each particular prompt. If no default is possible, the brackets are empty.

NOTE

For any of the Offline K Test prompts, use the DELete key to delete mistyped parameters before the terminating carriage return is typed. If you note an error in a parameter already terminated with a carriage return, type CTRL/C to return to the initial prompt and re-enter all parameters.

The Offline K Test Selector first prompts:

K requestor # (1 thru 7) [] ?

Answer this question with single digit (1 through 7) that specifies the requestor number of the K to be used. Terminate the response by typing a carriage return. After the requestor number is supplied, a K Control Area is located in Control memory and tested. This area is required for communicating with the K that will run its microdiagnostics. The test then prompts:

Test # (1 thru 11) (O) [] ?

Legal test numbers are octal numbers between 1 and 11(8), except for Test 5. (Test 5 is the K's Control and Data memory test, which is supported by the OFL K/P Memory Test.) Terminate the test number with a carriage return. The test then prompts:

of passes to perform (D) [1] ?

Enter a decimal number between 1 and 2,147,483,647 (omitting commas) to specify the number of times the memory test should be repeated. (Entering a zero, or just a carriage return, results in performance of one pass.)

The P.ioj next instructs the K to perform the selected test, and allows up to 4.2 seconds for the K to complete its test. If the K completes the test within this time, the P.ioj displays an end-of-pass message. If the K fails to complete within 4.2 seconds, the P.ioj displays a K Time-Out Error (Error 009).

The K microdiagnostics are designed to hang when an error is detected, so all failures in the microdiagnostics are reported as time-out errors. The current test may be aborted at any time by typing CTRL/C.

After the first test has been specified and completed, the following prompt is issued:

Reuse parameters (Y/N) [Y] ?

If you answer this prompt with a carriage return or a Y followed by a carriage return, the last test specified is repeated, using the same parameters. If you answer the prompt with an N, followed by a carriage return, the test prompts for new parameters.

6.5.4 Offline K Test Selector Progress Reports

Each time the K completes one full pass through the test specified, an end-of-pass report is displayed. A full pass is defined as:

1. The K completes the test with no errors detected.

2. The K fails its test, and the P.ioj times-out.

The end-of-pass message is displayed as follows:

End of Pass nnnnnn, xxxxxx Errors, yyyyyy Total Errors

The pass count nnnnnn is a decimal count of the number of complete passes made. The Errors count (xxxxxx) indicates the number of errors detected during the current pass. The Total Errors count (yyyyyy) indicates the number of errors detected during all passes completed so far.

6.5.5 Offline K Test Selector Error Information

All error messages produced by this test conform to the HSC generic diagnostic error message format (Section 6.1.5). Offline K Selector Test error messages are preceded by an OKTS> prompt. In this test optional lines three, four, and five show the address of the failing location (MA), expected data (EXP), and actual data (ACT).

6.5.5.1 K.ci Path Status Information - Whenever a K.ci is enabled, it runs the CI Link test as part of its microdiagnostics. The Link test performs loop-back tests on CI paths A and B of the K.ci. To pass the Link test, one of the paths must work (one failing path is not a fatal error). The microdiagnostics then return information in the K Control Area which specifies which paths worked, and how many retries were required. (The test retries 64 times before declaring a failure.)

The Offline K Test selector reports the CI path status each time the K.ci is initialized. If the Link test is selected (K.ci Test 11), the path status is reported only after the Link test completes. (When the K.ci is enabled, it runs all of its microdiagnostics, including the Link test. If the Link test was selected, the K.ci runs that test once more.)

The CI path status display indicates which path failed the Link test, if any. If both paths fail, the microdiagnostics fail in Test 11, and no path status information is displayed. The status display also includes the number of retries required for paths that passed the Link test.

6.5.5.2 Offline K Test Selector Error Messages - Errors detected by this test fall into one of three classes:

1. Control memory errors occur when the P.ioj is testing the portion of Control memory used to communicate with the K. (The P.ioj does not test Data memory.) Error numbers 000 through 007 are all Control memory errors detected by the P.ioj. The difference between these errors is the exact step in the memory test where they are detected. The step where an error was detected can be a helpful clue to the cause of the error.

2. Failures in a K microdiagnostic detected by a time-out. Error 008 indicates the K failed to initialize properly. Error 009 indicates the K failed the selected microdiagnostic.

3. Unexpected traps detected by the P.ioj (NXM and Parity). Errors 010 and 011 are unexpected trap errors detected by the P.ioj. Error 010 signifies a parity trap occurred, and error 011 indicates a Non-Existent Memory trap. The reports for unexpected trap errors differ slightly from a data error report, since they do not display expected and actual data. Error 012 indicates no working control memory could be found for a K Control Area. Error 13 is a cache parity trap.

The following list describes the nature of the failure indicated by each error number:

- **Error 000** - occurs in the Moving Inversions test when the P.ioj is testing the K Control Area at a memory location that did not contain the expected pattern, where:

  MA = The address of the failing location.
  EXP = The data pattern expected.
  ACT = The data pattern actually found.

  This error can be caused by a data error in the address specified, or it may indicate a dual-addressing problem (the location was incorrectly addressed and written when some other location was written). At this step in the test, a dual-addressing problem is characterized by:

  1. The ACTual data contains a single additional one.

  2. The additional one bit occurs immediately to the left of the leftmost one in the EXPected data. For example:

     EXP=000377, ACT=000777
     EXP=077777, ACT=177777
     EXP=000000, ACT=000001

     For the first example, the location in error was probably written with the pattern 000777 when a lower numbered address was being written with the same pattern. When the location in error was subsequently checked to ensure it still contained the previous pattern (000377), it contained the next pattern (000777). Data errors at this step of the test fall into one of the following classes:

     a. The ACTual and EXPected data differ by more than one bit:

        EXP=017777, ACT=017477

     b. The ACTual data contains fewer ones than the expected data:

        EXP=003777, ACT=001777

     c. The bit in error is not in the bit position immediately to the left of the leftmost one in the expected data:

        EXP=000777, ACT=002777

- **Error 001** - occurs in the Moving Inversions Test when the P.ioj is testing the K Control Area at a location

written with a pattern. Immediately after the write, the location was read and found to contain an incorrect pattern, where:

MA = The address of the failing location
EXP = The data pattern expected
ACT = The data pattern actually found

This error indicates a memory data problem. One of the following hardware failures is indicated:

1. A bit was picked up or dropped when the location was written.

2. A bit was picked up or dropped when the location was read.

If the error occurs repeatedly, but only in a single location, the memory chip containing the failing bit for that address is probably defective.

If the error occurs in many locations, but only occurs in a particular nibble (4-bit field), one of the bus data transceivers for that nibble is probably defective.

If the error occurs in many locations, and the bits in error are randomly spaced throughout the word, the memory or bus timing is probably faulty.
If the error occurs in more than one location, but the addresses of the failing locations are similar, there could be crosstalk between the memory data and addressing lines. For instance, all failing addresses end with either 2 or 6.

- **Error 002** - occurs in the Moving Inversions test when the P.ioj is testing the K Control Area. A memory location did not contain the expected pattern, where:

MA = The address of the location in error.
EXP = The data pattern expected.
ACT = The data pattern actually found

This error can be caused by a data error in the address specified, or it may indicate a dual-addressing problem. (The location was incorrectly addressed and written when some other location was being written.) At this step in the test, a dual-addressing problem is characterized by:

1. The ACTual data contains one more zero than the EXPected data.

2. The additional zero occurs in the same bit position as the leftmost one in the expected data:

EXP=003777, ACT=001777
EXP=000017, ACT=000007
EXP=177777, ACT=077777

In the first example, the location in error was probably written with the pattern 001777 when a lower numbered address was being written with the same pattern. When the location in error was subsequently checked to ensure it still contained the previous pattern (003777), it contained the next pattern (001777).

Data errors in this step of the Moving Inversions test fall into one of the following categories:

1. The ACTual and EXPected data differ by more than one bit:

   EXP=177777, ACT=174777

2. The ACTual data contains more ones than the expected data:

   EXP=037777, ACT=077777

3. The bit in error is not in the same bit position as the leftmost one in the EXPected data:

   EXP=001777, ACT=001377

- **Error 003** - occurs in the Moving Inversions Test when the P.ioj is testing the K Control Area. A location was written with a pattern. Immediately after the write, the location was read and found to contain an incorrect pattern, where:

  MA = The address of the failing location
  EXP = The data pattern expected
  ACT = The data pattern actually found

  This error indicates a memory data problem. One of the following hardware failures is indicated:

  1. A bit was picked up or dropped when the location was written.

  2. A bit was picked up or dropped when the location was read.

  If the error occurs repeatedly but only in a single location, the memory chip containing the failing bit for that address is probably defective.

If the error occurs in many locations but only occurs in a particular nibble (4-bit field), one of the bus data transceivers for that nibble is probably defective.

If the error occurs in many locations, and the bits in error are randomly spaced throughout the word, the memory or bus timing is probably faulty.

If the error occurs in more than one location, but the addresses of the failing locations are similar, there could be crosstalk between the memory data and addressing lines. For instance, all failing addresses end with either 2 or 6.

- **Error 004** - occurs in the Moving Inversions test when the P.ioj is testing the K Control Area. A memory location did not contain the expected pattern, where:

  MA = The address of the failing location.
  EXP = The data pattern expected.
  ACT = The data pattern actually found.

  This error can be caused by a data error in the address specified, or it may indicate a dual-addressing problem (the location was incorrectly addressed and written when some other location was written). At this step in the test, a dual-addressing problem is characterized by:

  1. The ACTual data contains a single additional one.

  2. The additional one bit occurs immediately to the left of the leftmost one in the expected data:

     EXP=000377, ACT=000777
     EXP=077777, ACT=177777
     EXP=000000, ACT=000001

  In the first example, the location in error was probably written with the pattern 000777 when a higher numbered address was being written with the same pattern. When the location in error was subsequently checked to ensure it still contained the previous pattern (000377), it contained the next pattern (000777). Data errors at this step of the test fall into one of the following classes:

  1. The ACTual and EXPected data differ by more than one bit:

     EXP=017777, ACT=017477

  2. The ACTual data contains fewer ones than the expected data:

EXP=003777, ACT=001777

3. The bit in error is not in the bit position immediately to the left of the leftmost one in the EXPected data:

   EXP=000777, ACT=002777

- **Error 005** - occurs in the Moving Inversions Test when the P.ioj is testing the K Control Area. A location was written with a pattern. Immediately after the write, the location was read and found to contain an incorrect pattern, where:

  MA = The address of the failing location
  EXP = The data pattern expected
  ACT = The data pattern actually found

  This error indicates a memory data problem. One of the following hardware failures is indicated:

  1. A bit was picked up or dropped when the location was written.

  2. A bit was picked up or dropped when the location was read.

  If the error occurs repeatedly but only in a single location, the memory chip containing the failing bit for that address is probably defective. If the error occurs in many locations but only occurs in a particular nibble (4-bit field), one of the bus data transceivers for that nibble is probably defective.

  If the error occurs in many locations, and the bits in error are randomly spaced throughout the word, the memory or bus timing is probably faulty.

  If the error occurs in more than one location, but the addresses of the failing locations are similar, there could be crosstalk between the memory data and addressing lines. For instance, all failing addresses end with either 2 or 6.

- **Error 006** - occurs in the Moving Inversions test when the P.ioj is testing the K Control Area. A memory location did not contain the expected pattern, where:

  MA = The address of the location in error.
  EX P= The data pattern expected.
  ACT = The data pattern actually found.

This error can be caused by a data error in the address specified, or it may indicate a dual-addressing problem. (The location was incorrectly addressed and written when some other location was being written). At this step in the test, a dual-addressing problem is characterized by:

1. The ACTual data containing one more zero than the expected data.

2. The additional zero occuring in the same bit position as the leftmost one in the EXPected data. For example:

   EXP=003777, ACT=001777
   EXP=000017, ACT=000007
   EXP=177777, ACT=077777

In the first example, the location in error was probably written with the pattern 001777 when a higher numbered address was being written with the same pattern. When the location in error was subsequently checked to ensure it still contained the previous pattern (003777), it contained the next pattern (001777). Data errors in this step of the Moving Inversions Test fall into one of the following categories:

1. The ACTual and EXPected data differ by more than one bit:

   EXP=177777, ACT=174777

2. The ACTual data contains more ones than the expected data:

   EXP=037777, ACT=077777

3. The bit in error is not in the same bit position as the leftmost one in the EXPected data:

   EXP=001777, ACT=001377

- **Error 007** - occurs in the Moving Inversions Test when the P.ioj is testing the K Control Area. A location was written with a pattern. Immediately after the write, the location was read and found to contain an incorrect pattern, where:

  MA = The address of the failing location
  EXP = The data pattern expected
  ACT = The data pattern actually found

This error indicates a memory data problem. One of the following hardware failures is indicated:

1. A bit was picked up or dropped when the location was written.

2. A bit was picked up or dropped when the location was read.

If the error occurs repeatedly but only in a single location, the memory chip containing the failing bit for that address is probably defective.

If the error occurs in many locations but only occurs in a particular nibble (4 bit field), one of the bus data transceivers for that nibble is probably defective.

If the error occurs in many locations, and the bits in error are randomly spaced throughout the word, the memory or bus timing is probably faulty.

If the error occurs in more than one location, but the addresses of the failing locations are similar, there could be crosstalk between the memory data and addressing lines. For example, all failing addresses end with either 2 or 6.

- **Error 008** - indicates the selected K did not complete its Init sequence properly. When the P.ioj enables the K to perform a test, the K begins its Init sequence (which includes executing certain microdiagnostics). At the end of the K's Init sequence, the K indicates it found the K Control Area by complementing a pointer word in the Control memory. If the K fails to complement this pointer word within 4.2 seconds of being enabled, Error 008 is reported.

  The contents of the K Status Register are displayed with the error report.

  If this error occurs, make sure the Requestor Number parameter given matches the actual requestor number of the K.

- **Error 009** - indicates the K failed the selected microdiagnostic test. This usually indicates a serious hardware problem in the K. The contents of the K Status Register are displayed with the error report.

- **Error 010** - indicates the P.ioj detected a Parity trap. The 22-bit address of the location that caused the trap is displayed as the MA data in the error report, where:

MA = The address causing the parity trap.
VPC = The Virtual PC of the memory test at the time the trap occurred. Reference this address in the listing to locate the area of the test where the error occurred.

Because the data is lost when a parity trap occurs, no EXPected or ACTual data is displayed. After the trap is reported, the program attempts to restart the test from the beginning.

- **Error 011** - indicates the P.ioj detected a Non-Existent Memory trap. A NXM error is caused when no memory responds to a particular address. The MA data in the error report indicates the address which produced the NXM trap. After reporting the trap, the program attempts to restart the test from the beginning, where:

  MA = The address causing the NXM trap.
  MA = The address causing the parity trap.
  VPC = The Virtual PC of the memory test at the time the trap occurred. Reference this address in the listing to locate the area of the test where the error occurred.

  If this error occurs at a memory address that should be in your memory configuration, the memory in question is not supplying an ACK to the P.ioj when the specified address is presented on the Memory bus. The most probable point of failure is the logic on the memory module that compares addresses on the memory bus to the range of addresses the module should respond to. Also, the comparator itself could be faulty, or the [C IN, C OUT], [D IN, D OUT], or [P IN, P OUT] lines on the backplane could be in error.

- **Error 012** - indicates no working Control memory could be found for a K Control Area. A K Control Area is required to communicate with a K. The Control memory must be repaired before the K Test Selector can be used to test a K. Use the Offline Loader command TEST MEMORY to test Control memory.

- **Error 013 - Cache Parity Trap, VPC = xxxxxx** - can happen during any test. The J11 trapped through the parity vector. The error was caused by the cache.

  During the run of the diagnostic, the J11 took a trap through the parity error vector. This is a cache error. The virtual PC at the time of the trap is printed.

6.5.6 Offline K Test Selector Summaries

The following is a list of Offline K Selector test summaries.

- **Test 000 - Moving Inversions Test** - is the Moving Inversions (MOVI) memory test used by the P.ioj to test a K Control Area. The K Control Area is used to pass memory test parameters to the K and to return the results of memory tests to the P.ioj. The Moving Inversions RAM test is used to detect data and addressing problems in dynamic semiconductor memories.

  The following are the steps in the Moving Inversions Algorithm:

  1. Write 000000 in each location being tested.

  2. Read all locations in order from lowest to highest. After reading a location and checking for a zero, rewrite the same location with a single one in the least significant bit. Then reread the location and verify the write worked correctly.

  3. Again read all locations in order from lowest to highest. Check that each location contains the data previously written. Rewrite the data found with a single additional one bit. Reread it to verify the write operation worked properly

  4. Repeat step 3 until the test pattern consists of a word containing all ones (pattern 177777).

  5. Repeat step 3, but this time substitute a single extra zero each time, instead of a one.

  6. Continue step 5 until the test pattern consists of a word of all zeros (pattern 000000).

  7. Repeat steps 1 through 6, but this time start at the highest memory address each time and work down to the lowest. This will work each memory location from all zeros to all ones, and back to all zeros.

  8. End of test. All memory is cleared to 000000.

- **Test 001 through Test 011 (K Microdiagnostics)** - Refer to the following three lists for the names of each microdiagnostic. Included in each list is the type of K being used and the failing test number.

  1. K.ci Microdiagnostics - The following list shows the test number and name of each of the K.ci microdiagnostics:

Test 0 - Sequencer Test

Test 1 - ALOE Test

Test 2 - Data Bus Test

Test 3 - Control Bus Test

Test 4 - PROM Parity Test

Test 5 - Memory Test (Unavailable via K Test Selector)

Test 6 - RAM Test

Test 7 - PLY Interface Test

Test 10- Packet Buffer Test

Test 11- Link Test

2. K.sdi Microdiagnostics - The following list shows the test number and name of each of the K.sdi microdiagnostics:

   Test 0 - Sequencer Test

   Test 1 - ALOE Test

   Test 2 - Data Bus Test

   Test 3 - Control Bus Test

   Test 4 - PROM Parity Test

   Test 5 - Memory Test (Not available via K Test Selector)

   Test 6 - RAM Test

   Test 7 - SERDES/RSGEN Test

   Test 10 - Partial SDI Interface Test

3. K.sti Microdiagnostics - The following list shows the test number and name of each of the K.sti microdiagnostics:

   Test 0 - Sequencer Test

   Test 1 - ALOE Test

Test 2 - Data Bus Test

Test 3 - Control Bus Test

Test 4 - PROM Parity Test

Test 5 - Memory Test (Not available via K Test Selector)

Test 6 - RAM Test

Test 7 - SERDES Test

Test 10 - Partial STI Interface Test

6.6 OFFLINE K/P MEMORY TEST

The Offline K/P Memory Test tests the HSC Control and Data memories from a K.sdi, K.c, or K.sti. This test executes from the I/O control processor and uses the HSC K Control Area to instruct one of the subsystem requestors to test either the Control or Data memories. You select the K to be used as well as the starting and ending addresses of the section of memory to be tested. The test algorithm used by the K stresses the memories trying to detect transient errors caused by bus and memory timing problems. Errors are reported at the console terminal as they occur.

6.6.1 Offline K/P Memory Test System Requirements

Hardware required by this test includes:

- I/O control processor module with HSC70 Boot ROMs
- At least one memory module
- RX33 controller with at least one working drive
- Terminal connected to I/O control processor console interface
- At least one working K.sdi, K.sti, or K.ci
- Working Control memory for a K Control Area

6.6.2 Offline K/P Memory Test Operating Instructions

If the HSC70 is not booted and loaded, refer to Section 6.1.2, Section 6.1.3, and Section 6.2. If these preceding steps are complete, you are at the ODL> prompt. Follow these next steps to start the memory test.

1. Type **TEST MEMORY BY K** in response to the Loader prompt (ODL>). The RX33 LED lights as the memory test is loaded.

2. The memory test indicates it has been loaded properly by displaying the following:

   HSC70 OFL K/P Memory Test

3. The memory test then prompts for parameters.

6.6.3 Offline K/P Memory Test Parameter Entry

This section describes the various parameters for the Offline K/P Memory Test.

NOTE

For any of the Offline K/P Memory Test prompts, use the DELete key to delete mistyped parameters before the terminating carriage return is typed. If you note an error in a parameter already terminated with a carriage return, type a CTRL/C to return to the initial prompt and re-enter all parameters.

The Offline K/P Memory Test first prompts:

requestor # of K (1 through 9) [] ?

Answer this question with the single digit (1 through 9), that specifies the requestor number to be used. Terminate the response by typing a carriage return. After the requestor number is supplied, a K Control Area is located in Control memory and tested. This area is required for communicating with the requestor that performs tests of Data and Control memory. The test then prompts:

Control (0) or Data (1) memory [0] ?

Type a zero to test Control memory or type a one to test Data memory. Type a carriage return to terminate your response. (Typing just a carriage return selects the Control memory test.) The memory test next prompts for the first address to test:

First (in=XXXXXXXX) [in] ?

Enter the first address to be tested. Addresses are eight octal digits in length. The [in] address displayed is the lowest address that may be entered for the memory chosen. After typing the address, terminate your response with a carriage return. (Typing just a carriage return causes the first address to default to the in address.)

NOTE

Because requestors test Control memory in 4-byte units, the lowest two bits of the starting address are ignored (treated as binary zeros). For example, if address 16000223 is entered as the first address, the requestor starts testing at address 16000200.

Because requestors test Data memory in 64-byte units, the lower six bits of the starting address are ignored (treated as binary zeros). For example, if address 14012376 is entered as the first address, the K starts testing at address 14012300.

The test next prompts for the last address to test:

```
Last (max=XXXXXXXX) [] ?
```

Enter the last address to be tested. The max address displayed is the highest address still within the memory chosen. If your system does not have a fully populated memory, the last address that may be tested is less than the max address displayed. If you choose a last address that exceeds the amount of memory in your system, the memory test displays a Non-Existent Memory (NXM) error when the test reaches the first address beyond the end of your memory. (Use the Offline Loader command SIZE to determine the actual last address in a given HSC.)

NOTE

Because requestors test control memory in 4-byte units, the lower 2 bits of the ending address are ignored (treated as binary ones). For instance, if address 16023400 is specified as the last address, the K will test up to and including address 16023403.

Because requestors test data memory in 64-byte units, the lower 6 bits of the ending address are ignored (treated as binary ones). If address 14005400 is specified as the last address, the requestor will test up to and including, address 14005477.

Finally, the memory test prompts:

of passes to perform (D) [1] ?

Enter a decimal number between 1 and 2,147,483,647 (omitting commas) to specify the number of times the memory test should be repeated. (If you enter a zero or a carriage return, the test performs one pass.) The test can be aborted at any time by typing CTRL/C.

After the first memory test completes, the following prompt is issued:

Reuse parameters (Y/N) [Y] ?

Answering this prompt with a carriage return or a Y followed by a carriage return repeats the last test specified using the same parameters. Answering the prompt with a N followed by a carriage return causes the prompt for new parameters.

6.6.4 Offline K/P Memory Test Progress Reports

Each time the requestor completes one full pass through the memory specified, an end-of-pass report is displayed. A full pass is defined as:

1. A complete test of the memory specified with no errors detected

2. Testing the memory specified until an error occurs

The end-of-pass message is displayed as follows:

End of Pass nnnnnn, xxxxxx Errors, yyyyyy Total Errors

The Pass count nnnnnn is a decimal total of the complete passes made. The Errors count (xxxxxx) indicates the number of errors detected on the current pass. The Total Errors count (yyyyyy) indicates the number of errors detected during the passes completed so far.

6.6.5 Offline K/P Memory Test Parity Errors

When a parity error occurs, it is desirable to know whether the error was produced by the loss or gain of a data bit or by the loss or gain of a parity bit. When a parity trap occurs in the I/O control processor, the data that was read is discarded by the PDP-11. However, a feature of the I/O control processor allows parity traps to be disabled. Using this feature, a user can determine if a parity error is being caused by a data or parity bit as follows:

1. After a parity trap (P.ioj detected) is reported, type a **CTRL/C** to terminate the memory test.

2. Type another **CTRL/C** to return to the OFL Diagnostic Loader. The Loader prompts: ODL>.

3. Type **Ex 17770042** followed by a carriage return.

   The contents of the I/O control processor Switch Control and Status Register (SWCSR) are displayed as follows: "(I) 17770042 nnnnnn".

4. Type **De * nnnn4n** followed by a carriage return. The nnnn4n represents the previous contents of the register, including a one in bit 5. I/O control processor parity traps are now disabled.

5. Return to the memory test by typing **Start** followed by a carriage return.

6. Rerun the memory test with the original parameters.

   If the location that previously produced a parity trap then produces a data error, the original parity trap was caused by a data bit problem. The error report indicates the failing bit via the EXPected and ACTual data displayed.

   If the location that previously produced a parity trap does not fail again when the memory test is rerun, the original parity trap was caused by an error in one of the parity bits (high or low byte) for that word.

7. Type a **CTRL/C** to return to the Loader, and re-enable parity errors by typing **De 17770042 nnnn0n** followed by a carriage return. The nnnn0n represents original contents of the I/O control processor SWCSR, before parity traps were disabled. (Refer to step 5.)

6.6.6 Offline K/P Memory Test Error Information

For generic diagnostic error message format, refer to Section 6.1.5. Listed below is a typical error message from Test Memory by K:

```
OKPM>hh:mm T aaa E bbb U-000
 6<Text describing error>
 MA -xxxxxxxx
 EXP-yyyyyy
 ACT-zzzzzz
 <K-Error-Summary-Info>
```

where:

```
hh      = Elapsed hours since last bootstrap
mm      = Elapsed minutes
aaa     = Decimal number denoting test
bbb     = Decimal number denoting the error detected
xxxxxxxx = Address of location causing the error
yyyyyy  = Data that was expected
zzzzzz  = Data that was actually found

<K-Error-Summary-Info>  =  See next section.
```

6.6.6.1 Offline K/P Memory Test Error Summary Information - When the requestor reports a memory test failure to the I/O control processor, the following information is supplied:

1. Address of the failing memory location
2. Data expected and data actually found
3. Error summary information

The error summary information is supplied as a 3-bit field, including the following:

1. A bit indicating a parity error occurred while reading the location
2. A bit indicating an NXM error occurred while accessing the location
3. A bit indicating a Control Bus (CBUS) error occurred while accessing the location

When a memory error report is issued for an error detected by the K, the last line of the error report includes a list of the error summary bits that were set (if any).

A Control Bus (CBUS) Error indicates the requestor asserted an illegal combination of the three CCYCLE lines when accessing Control memory. As these lines were previously tested from the I/O control processor (in the OFL P.ioj Test), a Control Bus error is most likely caused by a problem with the requestor's drivers that assert the CCYCLE lines.

6.6.6.2 Offline K/P Memory Test Error Messages - Error messages produced by this test can be caused by a memory error detected either by the I/O control processor or by the requestor being used to test memory. Errors detected by the I/O control processor occur when the I/O control processor is testing the portion of Control memory used to communicate with the K. (The I/O control processor does not test Data memory.)

To determine whether the I/O control processor or the requestor detected an error, examine the second line of the error message. The text either begins with a (P) or a (K). If the text begins with a (P), the I/O control processor detected the error. If the text begins with a (K), the requestor detected the error.

Error numbers 000 through 007 are all Control memory errors detected by the I/O control processor. The difference between these errors is the exact step in the memory test where they are detected. The step where an error was detected can be a helpful clue to the cause of the error.

Error 008 indicates the requestor failed to initialize properly.

Error 009 indicates a Control or Data memory error detected by the K. In addition to the normal error information, the last line of the error report contains a K Error Summary (see previous section).
Errors 010 and 011 are unexpected trap errors detected by the I/O control processor. Error 010 signifies a parity trap occurred; error 011 indicates a Non-Existent memory trap. The reports for unexpected trap errors differ slightly from a data error report because they do not display expected and actual data.

Error 012 indicates no working Control memory could be found for a K Control Area. Error 013 indicates a parity trap caused by cache.

The following list describes the nature of the failure indicated by each error number:

- **Error 000** - occurs in the Moving Inversions test (see Section 6.6.7) when the I/O control processor is testing the K Control Area. A memory location did not contain the expected pattern, where:

  MA = Address of the failing location
  EXP = Data pattern expected
  ACT = Data pattern actually found

  This error can be caused by a data error in the address specified, or it may indicate a dual-addressing problem (the location was incorrectly addressed and written when some other location was written). At this step in the test, a dual-addressing problem is characterized by:

  1. The ACTual data contains a single additional one.

  2. The additional one bit occurs immediately to the left of the leftmost one in the EXPected data, such as:

```
EXP=000377, ACT=000777
EXP=077777, ACT=177777
EXP=000000, ACT=000001
```

In the first example, the location in error was probably written with the pattern 000777 when a lower numbered address was being written with the same pattern. When the location in error was subsequently checked to ensure it still contained the previous pattern (000377), it contained the next pattern (000777).

Data errors at this step of the test fall into one of the following classes:

1. The ACTual and EXPected data differ by more than one bit:

   EXP=017777, ACT=017477

2. The ACTual data contains fewer ones than the EXPected data:

   EXP=003777, ACT=001777

3. The bit in error is not in the bit position immediately to the left of the leftmost one in the EXPected data:

   EXP=000777, ACT=002777

- **Error 001** - occurs in the Moving Inversions Test (Section 6.6.7) when the I/O control processor is testing the K Control Area. A location was written with a pattern. Immediately after the write, the location was read. It contained an incorrect pattern, where:

  MA = Address of the failing location
  EXP = Data pattern expected
  ACT = Data pattern actually found

  This error indicates a memory data problem. One of the following hardware failures is indicated:

  1. A bit was picked up or dropped when the location was written.

  2. A bit was picked up or dropped when the location was read.

If the error occurs repeatedly but only in a single location, the memory chip containing the failing bit for that address is probably defective.

If the error occurs in many locations but only occurs in a particular nibble (4-bit field), one of the bus data transceivers for that nibble is probably defective.

If the error occurs in many locations, and the bits in error are randomly spaced throughout the word, the memory or bus timing is probably the problem.

If the error occurs in more than one location, but the addresses of the failing locations are similar, crosstalk between the memory data and addressing lines may be present. (For example, all failing addresses end with either 2 or 6.)

- **Error 002** - occurs in the Moving Inversions test (Section 6.6.7) when the I/O control processor is testing the K Control Area. A memory location did not contain the expected pattern, where:

  MA = Address of the location in error
  EXP = Data pattern expected
  ACT = Data pattern actually found

  This error can be caused by a data error in the address specified, or it may indicate a dual-addressing problem. (The location was incorrectly addressed and written when some other location was being written.)

  At this step in the test, a dual-addressing problem is characterized by:

  1. The ACTual data contains one more zero than the EXPected data.

  2. The additional zero occurs in the same bit position as the leftmost one in the EXPected data, such as:

     EXP=003777, ACT=001777
     EXP=000017, ACT=000007
     EXP=177777, ACT=077777

  In the first example, the location in error was probably written with the pattern 001777 when a lower numbered address was being written with the same pattern. When the location in error was subsequently checked to ensure it still contained the previous pattern (003777), it contained the next pattern (001777).

Data errors in this step of the Moving Inversions Test fall into one of the following categories:

1. The ACTual and EXPected data differ by more than one bit:

   EXP=177777, ACT=174777

2. The ACTual data contains more ones than the EXPected data:

   EXP=037777, ACT=077777

3. The bit in error is not in the same bit position as the leftmost one in the EXPected data:

   EXP=0017777, ACT=00377

- **Error 003** - occurs in the Moving Inversions Test (Section 6.6.7) when the I/O control processor is testing the K Control Area. A location was written with a pattern. Immediately after the write, the location was read. It contained an incorrect pattern, where:

  MA = Address of the failing location
  EXP = Data pattern expected
  ACT = Data pattern actually found

  This error indicates a memory data problem. One of the following hardware failures is indicated:

  1. A bit was picked up or dropped when the location was written.

  2. A bit was picked up or dropped when the location was read.

  If the error occurs repeatedly but only in a single location, the memory chip containing the failing bit for that address is probably defective.

  If the error occurs in many locations but only occurs in a particular nibble (4-bit field), one of the bus data transceivers for that nibble is probably defective.

  If the error occurs in many locations and the bits in error are randomly spaced throughout the word, the memory or bus timing is probably the problem.

If the error occurs in more than one location, but the addresses of the failing locations are similar, crosstalk between the memory data and addressing lines could be present. (For example, all failing addresses end with either 2 or 6.)

- **Error 004** - occurs in the Moving Inversions test (see Section 6.6.7) when the I/O control processor is testing the K Control Area. A memory location did not contain the expected pattern, where:

  MA = Address of the failing location
  EXP = Data pattern expected
  ACT = Data pattern actually found

  This error can be caused by a data error in the address specified, or it may indicate a dual-addressing problem (the location was incorrectly addressed and written when some other location was written). At this step in the test, a dual-addressing problem is characterized by:

  1. The ACTual data containing a single additional one.

  2. The additional one bit occuring immediately to the left of the leftmost one in the EXPected data, such as:

     EXP=000377, ACT=000777
     EXP=077777, ACT=177777
     EXP=000000, ACT=000001

  In the first example, the location in error was probably written with the pattern 000777 when a higher numbered address was being written with the same pattern. When the location in error was subsequently checked to ensure it still contained the previous pattern (000377), it contained the next pattern (000777).

  Data errors at this step of the test fall into one of the following classes:

  1. The ACTual and EXPected data differ by more than one bit:

     EXP=017777, ACT=017477

  2. The ACTual data contains fewer ones than the EXPected data:

     EXP=003777, ACT=001777

  3. The bit in error is not in the bit position immediately to the left of the leftmost one in the

EXPected data:
EXP=000777, ACT=002777

- **Error 005** - occurs in the Moving Inversions Test (Section 6.6.7) when the I/O control processor is testing the K Control Area. A location was written with a pattern. Immediately after the write, the location was read. It contained an incorrect pattern, where:

  MA = Address of the failing location
  EXP = Data pattern expected
  ACT = Data pattern actually found

  This error indicates a memory data problem. One of the following hardware failures is indicated:

  1. A bit was picked up or dropped when the location was written.

  2. A bit was picked up or dropped when the location was read.

  If the error occurs repeatedly but only in a single location, the memory chip containing the failing bit for that address is probably defective.

  If the error occurs in many locations but only occurs in a particular nibble (4-bit field), one of the bus data transceivers for that nibble is probably defective.

  If the error occurs in many locations and the bits in error are randomly spaced throughout the word, the memory or bus timing is probably the problem.

  If the error occurs in more than one location, but the addresses of the failing locations are similar, crosstalk between the memory data and addressing lines could be present. (For example, all failing addresses end with either 2 or 6.)

- **Error 006** - occurs in the Moving Inversions test (Section 6.6.7) when the I/O control processor is testing the K Control Area. A memory location did not contain the expected pattern, where:

  MA = Address of the location in error
  EXP = Data pattern expected
  ACT = Data pattern actually found

This error can be caused by a data error in the address specified or it may indicate a dual-addressing problem. (The location was incorrectly addressed and written when some other location was being written.)

At this step in the test, a dual-addressing problem is characterized by:

1. The ACTual data contains one more zero than the EXPected data.

2. The additional zero occurs in the same bit position as the leftmost one in the EXPected data, such as:

   EXP=003777, ACT=001777
   EXP=000017, ACT=000007
   EXP=177777, ACT=077777

In the first example, the location in error was probably written with the pattern 001777 when a higher numbered address was being written with the same pattern. When the location in error was subsequently checked to ensure it still contained the previous pattern (003777), it contained the next pattern (001777).

Data errors in this step of the Moving Inversions test fall into one of the following categories:

1. The ACTual and EXPected data differ by more than one bit:

   EXP=177777, ACT=174777

2. The ACTual data contains more ones than the EXPected data:

   EXP=037777, ACT=077777

3. The bit in error is not in the same bit position as the leftmost one in the EXPected data:

   EXP=001777, ACT=001377

- **Error 007** - occurs in the Moving Inversions Test (Section 6.6.7) when the I/O control processor is testing the K Control Area. A location was written with a pattern. Immediately after the write, the location was read. It contained an incorrect pattern, where:

  MA = Address of the failing location
  EXP = Data pattern expected
  ACT = Data pattern actually found

This error indicates a memory data problem. One of the following hardware failures is indicated:

1. A bit was picked up or dropped when the location was written.

2. A bit was picked up or dropped when the location was read.

If the error occurs repeatedly but only in a single location, the memory chip containing the failing bit for that address is probably defective.

If the error occurs in many locations but only occurs in a particular nibble (4-bit field), one of the bus data transceivers for that nibble is probably defective.

If the error occurs in many locations and the bits in error are randomly spaced throughout the word, the memory or bus timing is probably the problem.

If the error occurs in more than one location, but the addresses of the failing locations are similar, crosstalk between the memory data and addressing lines may be present. For example, all failing addresses end with either 2 or 6.

- **Error 008** - indicates the selected requestor did not complete its Init sequence properly. When the I/O control processor enables the requestor to perform the memory test, the requestor begins its Init sequence (which includes executing certain microdiagnostics). At the end of the requestor's Init sequence, the requestor indicates it found the K Control Area by complementing a pointer word in Control memory. If the requestor fails to complement this pointer word within 50 milliseconds (4.2 seconds for K.ci) of being enabled, error 008 is reported.

  The contents of the K Status Register are displayed with the error report. If this error occurs, make sure the Requestor Number parameter given matches the actual requestor number.

- **Error 009** - indicates a Control or Data memory error detected by the K, where:

  MA = 22-bit address of the failing location
  EXP = Data pattern expected by the K
  ACT = Data pattern found by the K

In addition to the address and the expected/actual data, the K returns an error summary, displayed as the last line of the error report. The error summary information indicates whether the error was caused by a parity error, a Non-Existent Memory (NXM) error, or a Control Bus (CBUS) error. If the error was not caused by any of the these, the error summary line does not appear in the error report. Refer to Section 6.4.6.1 for further information on the error summary.

- **Error 010** - indicates the I/O control processor detected a parity trap. The 22-bit address of the location that caused the trap is displayed as the MA data in the error report, where:

  MA = The address causing the parity trap.
  VPC = The Virtual PC of the memory test at the time the trap occurred. Reference this address in the listing to locate the area of the test where the error occurred.

  Because the data is lost when a parity trap occurs, no EXPected or ACTual data can be displayed. To further localize the problem, disable parity errors and rerun the test (Section 6.6.5). If the original failure was in a data-bit position, the memory test detects and reports the error, displaying the EXPected and ACTual data. This helps to trace the error to a particular address and/or bit position. If no further errors are detected after disabling parity errors, the original failure was in one of the parity-bits for the address displayed in the parity trap report.

- **Error 011** - indicates the I/O control processor detected a Non-Existent Memory trap. An NXM error is caused when no memory responds to a particular address. The MA data in the error report indicates the address which produced the NXM trap. After the trap is reported, the program attempts to restart the test from the beginning, where:

  MA = The address causing the NXM trap.
  VPC = The virtual PC of the memory test at the time the trap occurred. Reference this address in the listing to locate the area of the test where the error occurred.

  If this error occurs at a memory address that should be in your memory configuration, the memory in question is not supplying an ACK message to the I/O control processor when the specified address is presented on the Memory bus. The most probable point of failure is the compare logic on the memory module. This logic compares addresses on the Memory bus with the range of addresses to which the module should respond. The comparator

itself could be faulty or the [C IN, C OUT], [D IN, D OUT], or [P IN, P OUT] lines on the backplane could be in error.

- **Error 012** - indicates no working Control memory could be found for a K Control Area. A K Control Area is required to communicate with a requestor. Control memory must be repaired before the K/P Memory Test can be used. Use the Offline Loader command TEST MEMORY to test the Control memory.

- **Error 013** - Cache Parity Trap, VPC = xxxxxx - indicates the J11 took a trap through the parity error vector during the run of the diagnostic. This is a cache error. The virtual PC at the time of the trap is printed.

6.6.7 Offline K/P Memory Test Summaries

The following is a summary of individual K/P memory tests:

- **Test 000 - Moving Inversions Test from P.ioj** - is the Moving Inversions (MOVI) memory test used by the I/O control processor to test a requestor Control Area. The K Control Area is used to pass memory test parameters to the requestor and to return the results of memory tests to the I/O control processor. The Moving Inversions RAM test is used to detect data and addressing problems in dynamic semiconductor memories.

  The following are the steps in the Moving Inversions Algorithm:

  1. Write 000000 in each location being tested.

  2. Read all locations in order from lowest to highest. After reading a location and checking for a zero, rewrite the same location with a single one in the least-significant bit. Then reread the location and verify the write worked correctly.

  3. Again read all locations in order from lowest to highest. Check each location for the data previously written. Rewrite the data found with a single additional one bit. Reread it to verify the write operation worked properly.

  4. Repeat step 3 until the test pattern consists of a word containing all ones (pattern 17777777).

  5. Repeat step 3 but this time substitute a single extra zero each time, instead of a one.

6. Continue step 5 until the test pattern consists of a word of all zeros (pattern 000000).

7. Repeat steps 1 through 6 but this time start at the highest memory address each time and work down to the lowest. This changes each memory location from all zeros to all ones and back to all zeros.

8. End of test. All memory is cleared to 000000.

- **Test 001 - Moving Inversions Test from K** - is the Moving Inversions test implemented in the K microcode. The algorithm is identical to that described in the previous test, except steps 5 and 6 are omitted to save time. When the requestor detects an error, the remainder of the test is aborted, and the information concerning the error is returned to the I/O control processor via the K Control Area. The I/O control processor is responsible for displaying the error report.

6.7 OFFLINE MEMORY TEST

The Offline Memory test exercises the HSC memories. You may select Control, Data, or Program memory for testing. Three memory testing algorithms are used: the Quick Verify algorithm, the Moving Inversions algorithm, and the Walking Ones algorithm.

The Quick Verify algorithm quickly uncovers stuck data and address bits. The other two algorithms stress the memories, attempting to detect transient errors caused by bus and memory timing problems.

Errors are reported at the console terminal as they occur. After reporting a data error, or a parity error from a location being tested, testing continues where it left off. If an NXM error occurs during the memory test, testing is restarted from the beginning.

6.7.1 Offline Memory Test System Requirements

Hardware required for the Offline Memory Test follows:

- I/O control processor module with HSC Boot ROMs
- At least one memory module
- RX33 controller with at least one working drive
- Terminal connected to I/O control processor console interface

6.7.2 Offline Memory Test Operating Instructions

If the HSC70 is not booted and loaded, refer to Section 6.1.2, Section 6.1.3, and Section 6.2. If the HSC70 is booted and loaded, the terminal displays an ODL> prompt. At this point, follow these steps to start the memory test:

1. Type SIZE in response to the Loader prompt (ODL>). The RX33 drive-in-use LED lights as the Offline System Sizer is loaded.

   The Sizer displays the bounds of the various memories in the HSC. The memory size information includes the last address of each memory.

2. Type TEST MEMORY in response to the Loader prompt (ODL>). The RX33 drive-in-use LED lights as the memory test is loaded.

   The memory test indicates it has been loaded properly by displaying the following:

   HSC OFL Memory Test

   The memory test next prompts for parameters. Refer to the following section.

6.7.3 Offline Memory Test Parameter Entry

This section describes the Offline Memory Test parameter entry.

NOTE

For any of the Offline Memory Test prompts, use the DELete key to delete mistyped parameters before the terminating carriage return is typed. If you note an error in a parameter already been terminated with a carriage return, type a CTRL/C to return to the initial prompt and re-enter all parameters.

The Offline Memory Test first prompts:

Control(0), Data(1), or Program(2) Memory [0] ?

Type a 0 to test Control memory, type a 1 to test Data memory, or type a 2 to test Program memory. Type a carriage return to terminate your response. (Typing just a carriage return causes the Control memory test to be selected.) The memory test next prompts for the first address to test:

First (in=XXXXXXXX) [in] ?

Enter the first address to be tested. Addresses are eight digits in length. The [in] address displayed is the lowest address that may be entered for the memory chosen. Terminate your response with a carriage return. (Typing just a carriage return causes the first address to default to the in value.) The test next prompts for the last address to test:

Last (max=XXXXXXXX) [] ?

Type the last address to be tested. The max address displayed is the highest address still within the memory chosen. If your system does not have a fully-populated memory, the last address to be tested is less than the max address displayed. Use the memory size information displayed by the ODL SIZE command to answer this prompt with the correct address for the HSC under test. If you choose a last address that exceeds the amount of memory in your system, the memory test displays a Non-Existent Memory (NXM) error when the test reaches the first address beyond the end of your memory. The test then prompts:

of passes to perform (D) [1] ?

Enter a decimal number between 1 and 2,147,483,647 (omitting commas) to specify the number of times the memory test should be repeated. (Entering a 0, or just a carriage return, results in one pass.) It can be aborted at any time by typing a CTRL/C or CTRL Y.

After the first memory test is complete, the following prompt is issued:

Reuse parameters (Y/N) [Y] ?

Answering this prompt with a Y followed by a carriage return or with a carriage return alone, repeats the last test specified, using the same parameters. If you answer the prompt with an N, followed by a carriage return, the test prompts for new parameters.

6.7.4 Offline Memory Test Progress Reports

A complete pass through the memory test consists of one pass through the Quick Verify test, one pass through the Moving Inversions test, and one pass through the Walking Ones test. After each complete pass, an end-of-pass message is displayed as follows:

End of Pass nnnnnn, xxxxxx Errors, yyyyyy Total Errors

The Pass count nnnnnn is a decimal count of the number of complete passes made. The Errors count xxxxxx indicates the number of errors detected on the current pass. The Total Errors count (yyyyyy) indicates the number of errors detected on all passes of the test completed so far.

NOTE

A complete pass through the memory test for Program memory may take about eight hours. Unless exhaustive memory testing is required, allow this test to run only until the Quick Verify Pass Complete message is displayed. This should take no more than 10 minutes.

6.7.5 Offline Memory Test Parity Errors

The process to disable P.ioj parity traps is identical for the Offline Memory Test and the Offline K/P Memory Test. This process is described in Section 6.6.5.

6.7.6 Offline Memory Test Error Information

Refer to Section 6.1.5 for the generic diagnostic error message format. The following is a typical Offline Memory Test error message:

```
OMEM>hh:mm T aaa E bbb U-000
 <Text describing error>
 MA -xxxxxxxx
 EXP-yyyyyy
 ACT-zzzzzz
```

where:

hh	=	Elapsed hours since last bootstrap
mm	=	Elapsed minutes
aaa	=	Decimal number denoting test
bbb	=	Decimal number denoting the error detected
xxxxxxxx	=	Address of location causing the error
yyyyyy	=	Data that was expected
zzzzzz	=	Data that was actually found

Parity trap and NXM trap errors do not include EXPected and ACTual data.

6.7.6.1 Offline Memory Test Error Messages - Error messages produced by the memory test can be classed as either data errors or unexpected traps. Error numbers 000 through 010 are all memory data errors. The only difference between these errors is the exact step in the testing algorithm where they are detected. The step at which a data error occurs can be an important clue to the cause of the error. Errors 000 through 007 are declared in the Moving Inversions algorithm; while errors 008 through 010 are declared in the Walking Ones algorithm.

Errors 011 and 012 are unexpected trap errors. Error 011 signifies a parity trap occurred and error 012 indicates a Non-Existent Memory trap. The reports for unexpected trap errors differ slightly from a data error report because they do not display expected and actual data.

The following list describes the nature of the failure indicated by each error number:

- **Error 000** - occurs in the Moving Inversions test (see Section 6.6.7). A memory location did not contain the expected pattern.

  MA = Address of the failing location
  EXP = Data pattern expected
  ACT = Data pattern actually found

  This error can be caused by a data error in the address specified, or it may indicate a dual-addressing problem. In the second case, the location was incorrectly addressed and written when some other location was written. At this step in the test, a dual-addressing problem is characterized by:

  1. The ACTual data contains a single additional one.

  2. The additional one bit occurs immediately to the left of the leftmost one in the EXPected data, such as:

     EXP=000377, ACT=000777
     EXP=077777, ACT=177777
     EXP=000000, ACT=000001

  In the first example, the location in error was probably written with the pattern 000777 when a lower numbered address was being written with the same pattern. When the location in error was subsequently checked to ensure it still contained the previous pattern (000377), it contained the next pattern (000777).

  Data errors at this step of the test fall into one of the following classes:

  1. The ACTual and EXPected data differ by more than one bit:

     EXP=017777, ACT=017477

2. The ACTual data contains fewer ones than the EXPected data:

   EXP=003777, ACT=001777

3. The bit in error is not in the bit position immediately to the left of the leftmost one in the EXPected data:

   EXP=000777, ACT=002777

- **Error 001** - occurs in the Moving Inversions Test (Section 6.6.7) when the I/O control processor was testing the K Control Area. A location was written with a pattern. Immediately after the write, the location was read. It contained an incorrect pattern.

  MA = Address of the failing location
  EXP = Data pattern expected
  ACT = Data pattern actually found

  This error indicates a memory data problem. One of the following hardware failures is indicated:

  1. A bit was picked up or dropped when the location was written.

  2. A bit was picked up or dropped when the location was read.

  If the error occurs repeatedly but only in a single location, the memory chip containing the failing bit for that address is probably defective.

  If the error occurs in many locations but only occurs in a particular nibble (4-bit field), one of the bus data transceivers for that nibble is probably defective.

  If the error occurs in many locations, and the bits in error are randomly spaced throughout the word, the memory or bus timing is probably the problem.

  If the error occurs in more than one location, but the addresses of the failing locations are similar, crosstalk could exist between the memory data and addressing lines. (For example, all failing addresses end with either 2 or 6.)

- **Error 002** - occurs in the Moving Inversions Test (Section 6.6.7). A memory location did not contain the expected pattern, where:

MA = Address of the location in error
EXP = Data pattern expected
ACT = Data pattern actually found

This error can be caused by a data error in the address specified, or it may indicate a dual-addressing problem. (The location was incorrectly addressed and written when some other location was being written).

At this step in the test, a dual-addressing problem is characterized by:

1. The ACTual data containing one more zero than the EXPected data.

2. The additional zero occuring in the same bit position as the leftmost one in the EXPected data, for example:

   EXP=003777, ACT=001777
   EXP=000017, ACT=000007
   EXP=177777, ACT=077777

In the first example, the location in error was probably written with the pattern 001777 when a lower numbered address was being written with the same pattern. When the location in error was subsequently checked to ensure it still contained the previous pattern (003777), it contained the next pattern (001777.)

Data errors in this step of the Moving Inversions test fall into one of the following categories:

1. The ACTual and EXPected data differ by more than one bit:

   EXP=177777, ACT=174777

2. The ACTual data contains more ones than the EXPected data:

   EXP=037777, ACT=077777

3. The bit in error is not in the same bit position as the leftmost one in the EXPected data:

   EXP=001777, ACT=001377

- **Error 003** - occurs in the Moving Inversions Test (Section 6.6.7). A location was written with a pattern. Immediately after the write, the location was read and found to contain an incorrect pattern.

MA = Address of the failing location
EXP = Data pattern expected
ACT = Data pattern actually found

This error indicates a memory data problem and one of the following hardware failures is indicated:

1. A bit was picked up or dropped when the location was written.

2. A bit was picked up or dropped when the location was read.

If the error occurs repeatedly but only in a single location, the memory chip containing the failing bit for that address is probably defective.

If the error occurs in many locations but only occurs in a particular nibble (4-bit field), one of the bus data transceivers for that nibble is probably defective.

If the error occurs in many locations, and the bits in error are randomly spaced throughout the word, the memory or bus timing is probably the problem.

If the error occurs in more than one location, but the addresses of the failing locations are similar, crosstalk could be present between the memory data and addressing lines. For example, all failing addresses end with either 2 or 6.

- **Error 004** - occurs in the Moving Inversions test (see Section 6.6.7.) A memory location did not contain the expected pattern, where:

MA = Address of the failing location
EXP = Data pattern expected
ACT = Data pattern actually found

This error can be caused by a data error in the address specified, or it may indicate a dual-addressing problem. In the latter case, the location was incorrectly addressed and written when some other location was written.

At this step in the test, a dual-addressing problem is characterized by:

1. The ACTual data containing a single additional one.

2. The additional one bit occuring immediately to the left of the leftmost one in the EXPected data, for instance:

   EXP=000377, ACT=000777
   EXP=077777, ACT=177777
   EXP=000000, ACT=000001

In the first example, the location in error was probably written with the pattern 000777 when a higher numbered address was being written with the same pattern. When the location in error was subsequently checked to ensure it still contained the previous pattern (000377), it contained the next pattern (000777.)

Data errors at this step of the test fall into one of the following classes:

1. The ACTual and EXPected data differ by more than one bit:

   EXP=017777, ACT=017477

2. The ACTual data contains fewer ones than the EXPected data:

   EXP=003777, ACT=001777

3. The bit in error is not in the bit position immediately to the left of the leftmost one in the EXPected data:

   EXP=000777, ACT=002777

- **Error 005** - occurs in the Moving Inversions Test (Section 6.6.7). A location was written with a pattern. Immediately after the write, the location was read and it contained an incorrect pattern.

  MA = Address of the failing location
  EXP = Data pattern expected
  ACT = Data pattern actually found

  This error indicates a memory data problem. One of the following hardware failures is indicated:

  1. A bit was picked up or dropped when the location was written.

  2. A bit was picked up or dropped when the location was read.

If the error occurs repeatedly but only in a single location, the memory chip containing the failing bit for that address is probably defective.

If the error occurs in many locations but only occurs in a particular nibble (4-bit field), one of the bus data transceivers for that nibble is probably defective.

If the error occurs in many locations, and the bits in error are randomly spaced throughout the word, the memory or bus timing is probably the problem.

If the error occurs in more than one location, but the addresses of the failing locations are similar, crosstalk between the memory data and addressing lines could be present. (For example, all failing addresses end with either 2 or 6.)

- **Error 006** - occurs in the Moving Inversions test (Section 6.6.7). A memory location did not contain the expected pattern, where:

  MA = Address of the location in error
  EXP = Data pattern expected
  ACT = Data pattern actually found

  This error can be caused by a data error in the address specified, or it may indicate a dual-addressing problem. (The location was incorrectly addressed and written when some other location was being written.)

  At this step in the test, a dual-addressing problem is characterized by:

  1. The ACTual data containing one more zero than the EXPected data.

  2. The additional zero occuring in the same bit position as the leftmost one in the EXPected data, for example:

     EXP=003777, ACT=001777
     EXP=000017, ACT=000007
     EXP=177777, ACT=077777

  In the first example, the location in error was probably written with the pattern 001777 when a higher numbered address was being written with the same pattern. When the location in error was subsequently checked to ensure it still contained the previous pattern (003777), it contained the next pattern (001777.)

Data errors in this step of the Moving Inversions test fall into one of the following categories:

1. The ACTual and EXPected data differ by more than one bit:

   EXP=177777, ACT=174777

2. The ACTual data contains more ones than the EXPected data:

   EXP=037777, ACT=077777

3. The bit in error is not in the same bit position as the leftmost one in the EXPected data:

   EXP=001777, ACT=001377

- **Error 007** - occurs in the Moving Inversions Test (Section 6.6.7). A location was written with a pattern. Immediately after the write, the location was read and found to contain an incorrect pattern, where:

  MA = Address of the failing location
  EXP = Data pattern expected
  ACT = Data pattern actually found

  This error indicates a memory data problem. One of the following hardware failures is indicated:

  1. A bit was picked up or dropped when the location was written.

  2. A bit was picked up or dropped when the location was read.

  If the error occurs repeatedly but only in a single location, the memory chip containing the failing bit for that address is probably defective.

  If the error occurs in many locations but only occurs in a particular nibble (4-bit field), one of the bus data transceivers for that nibble is probably defective.

  If the error occurs in many locations and the bits in error are randomly spaced throughout the word, the memory or bus timing is probably the problem.

If the error occurs in more than one location, but the addresses of the failing locations are similar, crosstalk may be present between the memory data and addressing lines. For example, all failing addresses end with either 2 or 6.

- **Error 008** - occurs in the Walking Ones test (Section 6.7.7). All locations in the memory under test were written with the pattern 000000. Then all locations were read to check that they all contained 000000. When the location specified in the error report was read, it did not contain 000000, where:

  MA = Address of the failing location
  EXP = Data pattern expected (000000)
  ACT = Data pattern actually found

  Because all locations were cleared to 00000000 before this error was detected, a dual-addressing problem is unlikely. More likely, a bit was picked up when the word was written or read.

  If the error occurs repeatedly but only in one location, the memory chip containing the bit in error for that address is probably marginal.

  If the error occurs in many locations but always occurs in a particular nibble (4-bit field), one of the bus data transceivers for that nibble is probably marginal.

  If errors occur in many locations, and the bits in error are randomly spaced throughout the words, the memory or bus timing is probably marginal.

- **Error 009** - occurs in the Walking Ones Test (Section 6.7.7). One location in the memory under test was written with the pattern 17777777 and all the other locations should contain the pattern 00000000. While reading to check that all other locations are clear, a location was found containing something other than 00000000, where:

  MA = Address of the failing location
  EXP = Data pattern expected (000000)
  ACT = Data pattern actually found

  This error is either a data error or a dual-addressing error. (The location was incorrectly addressed and written when some other location was being written.)

  At this step of the test a dual-addressing failure is possible if the ACTual data is 17777777. During this part of the test, one location in the memory was written to 17777777. When this write was performed, the failing

location may also have been addressed and written with the same data. When the test was checking that all other locations were clear, it found the second location with the pattern 17777777. If this is a true dual-addressing problem, the error is repeated on each pass of the test.

At this step of the test, a data error is probable if the ACTual data is NOT 17777777. Some clues to the possible causes of a data error follow.
If the error occurs repeatedly but only in a particular bit in a single location, the memory chip that contains the failing bit for that location is defective.

If errors occur in many locations but only occur in a particular nibble (4-bit field), one of the bus data transceivers for that nibble is probably marginal.

If errors occur in many locations and the bits in error are randomly spaced throughout the words, the memory or bus timing is probably marginal.

- **Error 010** - occurs in the Walking Ones test (Section 6.7.7). At this step of the test, one location in the memory under test was set to the pattern 17777777 and all other locations were cleared to 00000000. After checking that all other locations contain 00000000, the location that should contain 17777777 was read. It contained some other pattern. Because only read operations were performed after writing the 17777777, a dual-addressing problem is highly improbable.

  MA = Address of the failing location
  EXP = Data pattern expected (177777)
  ACT = Data pattern actually found

  If the error occurs repeatedly but only in a particular bit of a single location, the memory chip that holds that bit for the failing location is defective.

  If errors occur in many locations but only occur in a particular nibble (4-bit field), one of the bus data transceivers for that nibble is probably marginal.

  If errors occur in many locations, and the bits in error are randomly spaced throughout the words, the memory or bus timing is probably marginal.

  If errors occur in more than one location, but the addresses of the failing locations are similar, crosstalk may be present between the memory data and addressing lines. (For example, all failing addresses end in 2 or 4.)

- **Error 011** - indicates a parity trap occurred. The parity trap probably occurred in a location under test but may have been caused by Program memory where the memory test itself resides. The MA data in the error report indicates the address of the location causing the parity trap. After reporting the parity trap, the memory test continues if the parity error occurred in a memory location under test, where:

  MA = Address of the location causing the parity trap.
  VPC = Virtual PC of the memory test at the time the trap occurred. Reference this address in the listing to locate the area of the test where the error occurred.

  Because the data is lost when a parity trap occurs, no EXPected or ACTual data is displayed. To further localize the problem, disable parity errors and rerun the test. (Refer to Section 6.6.5.)

  If the original failure was in a data-bit position, the memory test detects and reports the error, displaying the EXPected and ACTual data. This helps trace the error to a particular address and/or bit position. If no further errors are detected after disabling parity errors, the original failure was in one of the parity bits for the address displayed in the parity trap report.

- **Error 012** - indicates a Non-Existent Memory trap occurred. An NXM error is caused when no memory responds to a particular address. The MA data in the error report identifies the address that produced the NXM trap. After reporting the error, the program attempts to restart testing from the beginning, where:

  MA = The address being tested at the time the NXM trap occurred.
  VPC = The PC of the memory test at the time the trap occurred. Reference this address in the listing to locate the area of the test where the error occurred.

  The most frequent cause of this error is specifying too large of a value for the Last Address to Test? prompt (trying to test beyond the end of the memory in your system).

  If this error occurs at a memory address that should be within your memory configuration, the memory in question is not supplying an ACK to the I/O control processor when the specified address is presented on the memory bus. The most probable point of failure is the logic on the memory module that compares addresses on the memory

bus with the range of addresses the module should respond to. The comparator itself could be faulty or the [C IN, C OUT], [D IN, D OUT], or [P IN, P OUT] lines on the backplane could be in error.

- **Error 013** - occurs in the Quick Verify test. This error may indicate a dual-addressing problem. The Quick Verify test consists of clearing the entire memory, then writing two patterns to each location and checking that the writes worked properly. Before writing the first pattern to each location, the contents of the location should be zero. Error 013 indicates a location contained something besides a zero before the first pattern was written.

  If the ACTual data in the error report is 031463(8) or 146314(8), a dual-addressing problem is probably the cause of the error. (When an address lower in memory was written with a test pattern, the failing location was also written with the same pattern.) Dual-addressing problems are normally caused by shorts between memory address bits.

  If the ACTual data is other than 031463(8) or 146314(8), the problem probably is caused by a memory bit or bits stuck in the one state. The first pattern written is 146314(8). The second pattern written is the ones complement of the first pattern, 031463(8).

- **Error 014** - occurs in the Quick Verify test. The MA in the error report shows the failing address. The ACTual data shows the bit or bits that failed.

- **Error 015** - occurs when an NXM trap occurs as the memory under test is initially being cleared. The last address to test (operator-supplied) exceeds the amount of memory actually installed in the HSC or part of the memory under test is not responding. If the NXM occurs at an address that should respond, use CTRL/C or CTRL/Y to return to the Offline Loader. Use the Loader's REPEAT EXAMINE (address that caused trap) to set up a scope loop for isolating the problem.

- **Error 016 - Cache Parity Trap, VPC = xxxxxx** - indicates the J11 took a trap through the parity error vector during the run of the diagnostic, and the error was determined to be from the cache. The virtual PC at the time of the trap is printed.

6.7.7 Offline Memory Test Summaries

The following list describes the three algorithms used by the Offline Memory Test:

- **Test 000 - Quick Verify Test** - quickly detects stuck bits and dual-addressing problems. The algorithm used by the Quick Verify test is as follows:

  Write 00000000 to each location of the memory

  FOR i = First to Last address

  IF < location i does not contain zero > THEN < display error >

  Write test pattern to location i (146314(8))

  IF < location i does not contain pattern > THEN < display error >

  Write complement of pattern to location i (031463(8))

  IF < location i does not contain complement > THEN < display error >

  NEXT i

- **Test 001 - Moving Inversions Test** - detects data and addressing problems in dynamic semiconductor memories.

  The Moving Inversions Algorithm performs the following:

  1. Writes 00000000 in each location of the memory.

  2. Reads all locations in order from lowest to highest. After reading a location and checking for a zero, rewrites the same location with a single one in the least-significant bit. Then rereads the location and verifies the write worked correctly.

  3. Again read all locations in order from lowest to highest. Check that each location contains the data previously written. Rewrite the data found with a single additional one bit. Rereads it to verify the write operation worked properly.

  4. Repeats step 3 until the test pattern consists of a word containing all ones (pattern 17777777).

  5. Repeats step 3 but this time substitute a single extra zero each time instead of a one.

  6. Continues step 5 until the test pattern consists of a word of all zeros (pattern 00000000).

7. Repeats steps 1 through 6 but this time starts at the highest memory address each time and works down to the lowest. This works each memory location from all zeros to all ones and back to all zeros.

8. Clears all memory to 00000000.

- **Test 002 - Walking Ones Test** - is an algorithm that stresses semiconductor memories and is effective in locating timing problems on the memory module or on the bus.

  The Walking Ones Algorithm performs the following:

  1. Writes all memory to zeros (pattern=00000000).

  2. Checks all memory for zeros. Declare Error 008 if not zero.

  3. Sets TESTADDRESS = first address to test.

  4. Writes 17777777 to contents of TESTADDRESS.

  5. Checks all other locations = 00000000. Declare an Error 009 if not equal to 00000000.

  6. Checks that TESTADDRESS contains 17777777. Declare an Error 010 if not equal to 17777777.

  7. Writes 00000000 to contents of TESTADDRESS.

  8. IF <TESTADDRESS = last address to test>
     THEN <done testing>
     ELSE <add 2 to TESTADDRESS, GOTO step 4>

6.8 RX33 OFFLINE EXERCISER

OFLRXE is a combined hardware diagnostic and exerciser for the HSC70 M.std2/RX33 subsystem. Diagnosis of the DMA hardware and diskette controller are provided, as well as a read/write exerciser to provide exercise for the actual drive portion of the subsystem. OFLRXE is a stand-alone diagnostic running under the Offline Diagnostic Loader. This loader provides terminal I/O service, time keeping, string conversions, and interrupt handling. OFLRXE is an 8 Kword program of which approximately half is control code and half is mapped for data buffer transfers.

6.8.1 RX33 Offline Exerciser System Requirements

This test must have a J11 P.ioj module, and a M.std2 memory/controller board. At least one RX33 drive must be present. One scratch diskette is needed for each drive to be tested (maximum of two).

Testing of the entire J11 chip set and the J11 cache is assumed if it is turned on. Two tested 4 Kword partitions of memory are required by OFLRXE.

6.8.2 RX33 Offline Exerciser Operating Instructions

If the HSC70 is not booted and loaded, refer to Section 6.1.2, Section 6.1.3, and Section 6.2. If the HSC70 is already booted and displaying the Offline Loader prompt (ODL>), proceed as follows:

At the ODL> prompt, invoke the Offline RX33 diagnostic by typing **TEST RX** followed by a carriage return. This loads the Offline RX33 diagnostic (OFLRXE) from the media, and transfers control to the diagnostic. At the start, the diagnostic should print out the following string:

HSC70 Offline RX33 Exerciser Vxxx

where:

Vxxx is a 3-digit version/edit number.

NOTE

If you are unable to boot from drive 0, move the diskette to drive 1, try again, or use a backup copy of the Offline Diagnostics diskette.

The Offline RX33 Exerciser terminates on CTRL/C, CTRL/Y, or on expiration of the allotted time. The program also terminates on fatal errors.

6.8.3 RX33 Offline Exerciser Parameter Entry

Following are the three user-modifiable parameters for this test:

1. Drive selection is prompted for by the program in the following manner:

   Test drive n (Y/N) [Y] ?

   Where n is drive number (0 or 1). The default is yes. The prompt repeats for each available diskette on the HSC70.

2. Operator is asked if initial write operation should be performed:

   Perform initial write on this drive (Y/N) [Y]

   The default is yes. This lays down a background pattern on the entire disk in preparation for the random read/write exerciser. Selecting this option adds 10 minutes of test time per drive.

   As soon as you have answered the previous prompts, the program directs you to place a scratch diskette in the selected drive:

   Insert a scratch diskette in the drive, type a carriage return to continue.

   At this point, insert your scratch diskette. The random read/write exercise takes place over the entire surface of the diskette, so be sure the diskette is a scratch one only to be used for the exercise.

3. Run time of the exerciser is user-selectable and is prompted for by the program as follows:

   # of minutes to exercise (D) [30] ?

   Enter a number between 1 and 32767. The default, if the user types just a carriage return, is 30 minutes. This 30 minutes starts after the initial patterning of the disk (if selected) so the total test time with two drives, and initial patterning is amount of time plus 20 minutes. A value of 1440 minutes gives a 24-hour run time for burn-in purposes. The 30-minute default is sufficient for installation use and repair verification.

At the end of the amount of time allotted for the exerciser, the program will prompt the user by printing:

Reuse parameters (Y/N) [Y] ?

Answering this prompt with a Y allows you to run the diagnostic again with the same parameters. Answering with a N takes you back through the parameter entry questions again.

6.8.4 RX33 Offline Exerciser Progress Reports

The Offline RX33 Exerciser does not run in a conventional pass sense. There are no pass completed messages. Instead some informational messages are printed indicating what the exerciser is currently doing. At the end of the initial write test (if selected), the exerciser prints:

Initial write completed on drive 000n

Where n is the drive number, 0 or 1.

When the exerciser begins the random read/write phase of the testing, the following message is printed:

Beginning random exerciser

The random exerciser is now in progress. It runs for the amount of time requested by the user. When the requested time has expired, the program prints the following string:

Exerciser completed.

The program then returns to the parameter entry routine.

The program also has a user-requested status report available. If at any time the user types CTRL/T on the console, the program responds:

Number of sectors transferred = xxxxxxxxxx, yyyyyy errors.

Where xxxxxxxx is a 16-digit number of sectors successfully transferred, and yyyyyy is a 6-digit cumulative number of errors detected.

6.8.5 RX33 Offline Exerciser Error Information

A generic message format for all offline diagnostic errors is found in Section 6.1.5. The following section contains information on specific errors associated with RX33 Offline Exerciser. A typical RX33 Offline Exerciser error message is:

```
OFLRXE>52:22 T 008 E 010 D 001
 SEEK error detected during positioning operation
 LBN  = 004356
 Track = 000114
 Sector =000007
 Surface = 00000
```

Soft errors, such as seek errors, can build up to a point where diagnostic defines them as fatal and terminates on a fatal error. The internal bias for soft errors is currently set to 20. When this number is exceeded, the Exerciser determines the errors are fatal and terminates.

6.8.5.1 Specific RX33 Offline Exerciser Error Messages

The following is a list of errors associated with test failures:

- **Error 00 - Parity trap, VPC = xxxxxx** - (Applicable to all tests) - occurs at any time during execution of the

diagnostic. The virtual PC on the stack is printed to help identify the program area where the error occurred. Both the content of the error address register and the virtual PC are displayed as optional lines. This error terminates the test. The diagnostic returns to the Reuse parameters prompt.

- **Error 01 - NXM Trap, VPC = xxxxxx** (Applicable to all tests) - causes the diagnostic to return to the Reuse parameters prompt. Additional data, such as the virtual PC of the instruction which caused the trap, and the physical address contained in the error address register are printed as optional lines.

- **Error 02 - Bit Stuck in Register** (Applicable to Test 1) - indicates a stuck-at fault is present in one of the RX33 control registers. The register address and the expected and actual data are printed as optional lines in the error message. If the error is in the low byte, the problem is the diskette controller chip. If the error is in the high byte, the problem is with the MAR register at that address. If more than one register show the same bit(s) in error, the problem is probably in the bus transceivers.

- **Error 03 - Interrupt Occurred Without Enable Set** (Applicable to Test 2) - indicates there is a stuck-at fault in the register, or the etch going into the DC003 interrupt control chip. The interrupt enable bit, <13> of the CSR, does not disable interrupts.

- **Error 04 - RX33 Interrupt Occurred at Wrong Priority** (Applicable to Test 2) - indicates the RX33 interrupt occurred with the priority at five or greater. The virtual PC where the interrupt occurred is printed out as an optional line. Using the listing of the program, you can determine the priority at the time of the interrupt.

- **Error 05 - Unexpected Interrupt from RX33** (Applicable to all tests) - indicates an unexpected interrupt. An interrupt that occurs at any time when a command to the RX33 is not in progress is defined as unexpected. The virtual PC where the interrupt occurred is printed as an optional line.

- **Error 06 - Track 0 Did Not Set After RECALIBRATE Command** (Applicable to Test 5) - indicates the track 0 status bit (bit 2 of the CSR) did not set upon completion of a RECALIBRATE command. The drive may not be sending the signal, or the cable to the drive may be faulty.

- **Error 07 - RX33 Did Not Interrupt as Expected** (Applicable to Test 2) - indicates an expected interrupt never occurred. The interrupt control chip (DC003) may be at fault, or the diskette controller chip interrupt signal is stuck at 1. The J11 may be unable to recognize interrupts from the diskette controller, or the backplane etches carrying interrupt control signals are open.

- **Error 10 - Seek Error Detected During Positioning Operation** (Applicable to Tests 5, 6, 7, and 8) - indicates a seek error status (bit 4 of the CSR) was set after a SEEK or RECALIBRATE command. The problem may be in the diskette controller chip or the diskette. If the errors are occurring mostly in test 5 starting with track 0, the problem is probably fundamental; the controller cannot read the diskette at all. If the errors occur in a random fashion, the problem is probably the diskette.

- **Error 11 - Current Track Register Incorrect** (Applicable to Tests 5 and 6) - indicates the values in the track register of the diskette controller chip are not as expected after a given operation. This problem probably is in the diskette controller chip.

- **Error 12 - CRC Error in Header Detected During Position Verify** (Applicable to Tests 5, 6, 7, and 8) - detects a CRC error when reading a header during a position verify. This error occurs when a valid header has been found and read, but the CRC at the end is incorrect. This is probably the diskette. If the controller is able to detect the address and data marks that precede a header (so that it knows that a header is being read), the data separation logic is probably working.

- **Error 13 - Processor Type is Not J11** (Applicable to Test 0) - does not contain the value which defines a J11. This error causes the diagnostic to terminate.

- **Error 14 - Drive Under Test is Not Ready** (Applicable to Tests 5, 6, 7, and 8) - indicates the diskette drive is sending NOT READY status to the controller. The door may open on the drive, or no diskette is inserted. If these conditions are not the cause of the fault, the ready signal from the drive may be stuck.

- **Error 15 - Last Command Did Not Complete** (Applicable to Tests 5, 6, 7, and 8) - indicates the last command issued to the diskette controller never interrupted to show completion. This error points to the diskette chip, since it occurs after the interrupt logic has already been tested.

- **Error 16 - RX33 Header Does Not Compare** (Applicable to Tests 7 and 8) - The header information written in the data area of a sector is not what it should be for that sector. Each sector has a unique header consisting of track, sector, and side, written as part of the data in that sector. This error happens when an undetected positioning error has occurred, either during the read, or the write of the sector involved. The LBN, track, sector, and side are displayed as optional lines.

- **Error 17 - Record Not Found During Read** (could also say Write) (Applicable to Tests 7 and 8) - indicates the controller was unable to find that sector on the current track when attempting to read or write a given sector. Either a misposition occurred, or that sector is unreadable. Because this error occurs after basic read capability has been tested, the most probable culprit is the diskette, with the diskette chip being the next most probable. The LBN, track, sector, and side are displayed as optional lines.

- **Error 20 - CRC error in Data During Read** (could also say Write) (Applicable to Tests 7 and 8) - indicates the controller detected a CRC error when reading the desired sector. If the error occurs multiple times in a row for a given sector, the problem is most likely the diskette (or the drive it is installed in). Single errors when an LBN has this error only once are soft errors. The LBN, track, sector, and side information is printed as optional lines.

- **Error 21 - Lost Data Detected During Read** (could also say Write) (Applicable to Tests 7 and 8) - indicates the DMA logic did not service an I/O request of the diskette controller chip in time. There are probably problems in the DMA logic, or stuck-at faults exist in the etch between the controller chip and the DMA logic.

- **Error 23 - Invalid Pattern Code in Buffer** (Applicable to Test 8) - indicates the data word, defined as the pattern code, read from the diskette does not match any of the possible patterns used. It is unlikely the data was read incorrectly from the diskette and not detected as a CRC error. Usually this error occurs when a diskette is not written with the initial data pattern. The LBN, track, sector, and side are displayed as optional lines.

- **Error 24 - Drive is Write-Protected** (Applicable to Tests 7 and 8) - indicates the drive is sending write protect status. Either the interface is bad, or the drive is in error (assuming you don't have a write-protected diskette in the drive). This error terminates the diagnostic, as you cannot write on a write-protected diskette.

- **Error 25 - CRC Error in Header During Read** (could also say Write) (Applicable to Tests 7 and 8) - indicates the controller detected bad CRC in the header it was reading as part of a data transfer command. This is probably a diskette error. The LBN, track, sector, and side are displayed as optional lines.

- **Error 26 - Data Incorrect After DMA TEST MODE command** (Applicable to Tests 3 and 4) - indicates the memory content after a DMA test mode command was not correct. There are either stuck-at faults in the DMA registers, or the transfer did not happen at all (that is, the memory is unchanged). This is a fundamental error in the diskette logic, and the diagnostic terminates after detecting it.

- **Error 27 - Data Compare Error** (Applicable to Tests 7 and 8) - indicates a manual check of data read by the diskette turned up an error. Either the transfer did not complete, an intermittent error occurred in the data or address path, or what was written on the disk was written incorrectly. The LBN, track, sector, and side are displayed as optional lines.

- **Error 30 - RX33 Detected Parity Error During Read** (could also say Write) (Applicable to Tests 7 and 8) - indicates the RX33 detected a parity error when doing a DMA read from memory. Either program memory is bad, or the parity logic on the controller is in error.

- **Error 31 - RX33 Detected NXM During Read** (could also say Write) (Applicable to Tests 7 and 8) - indicates the RX33 detected a NXM during a DMA operation. Either the DMA address was loaded wrong and pointed to a nonexistent location, or the handshake logic on the M.std2 board is in error.

- **Error 32 - RX33 MAR Value Incorrect After DMA Transfer** (Applicable to Test 3) - indicates the value of the MAR address counters was in error after a DMA test operation. The problem is probably in the counters or the etch associated with them. The EXPected and ACTual data are printed out as optional lines.

- **Error 33 - a Parity Error was not Forced in Main Memory** (Applicable to Test 4) - indicates a write to program

memory with bad parity (bit 11 of the CSR) set did not result in bad parity in memory. There is either a stuck-at fault in the parity logic or the operation never wrote memory in the first place.

- **Error 34 - Parity Error Did Not Set in CSR** (Applicable to Test 4) - indicates a DMA read of a location with known bad parity did not set the parity error bit (bit 15 of the CSR). Either the data was never read, or there is a stuck-at fault in the parity logic.

- **Error 35 - NXM Did Not Set in CSR** (Applicable to Test 4) - indicates a DMA read of a location expected to give a NXM did not set NXM in the CSR. Look for stuck-at faults in the NXM detection logic.

- **Error 36 - Parity Error Set Along With NXM in CSR** (Applicable to Test 4) - indicates both the parity error and the NXM error set simultaneously in the CSR. On a NXM error, the parity error should not set. Check for stuck-at faults in the NXM/parity error logic.

- **Error 37 - Cache Parity Error, VPC = xxxxxx** (Applicable to all tests.) - indicates the J11 took a trap through the parity error vector, a cache error during the run of the diagnostic. The virtual PC at the time of the trap is printed.

6.8.6 RX33 Offline Exerciser Test Summaries

The following is a summary of RX33 offline tests:

1. **Test 1 - RX33 Controller Registers** - performs stuck-at testing on the RX33 controller registers at 177400, 177402, 177404, and 177406. A simple walking one's test is performed on each register, except for the CSR register at 177400 which only has the high byte tested.

2. **Test 2 - Interrupt Hardware** - exercises the interrupt hardware on the M.std2. The interrupts generated are also tested for the correct priority when they occur.

3. **Test 3 - DMA Logic and Counters** - checks out all of the DMA handshake signals, the data path, and the address path. A special DMA test mode in the controller is used to perform one read or write to/from each memory location loaded in the DMA address registers. Correct incrementing action from the counters is checked. The actual data loaded to memory on a DMA write is checked as well.

4. **Test 4 - Parity Logic** - also uses DMA test mode in addition to the force bad parity function (bit 11 of the CSR) to prove parity errors can be detected, and correct parity is written to memory by the DMA control logic. NXM action is also lumped into this test. Correct handling of NXM errors is checked as well as correct reporting by the error bit in the CSR.

5. **Test 5 - Verify Track Counters and Registers** - uses the step function of the diskette controller chip to verify all cases of the track counter bits internal to the diskette controller chip work as advertised. Step functions are performed for each power of two in the diskette track register, (step four times, step eight times more, etc.). The verify option is set on each step command so the diskette controller reads headers on each track to verify position.

6. **Test 6 - Oscillating Seek Test** - performs an oscillating seek test using the algorithm:

```
oscillating seek test
 begin
       incnt = 0
       outcnt = 124
       while incnt<> outcnt do
         begin
          seek outcnt;
          CHECKSTATUS;
          If outcnt <> rxtrk then error 11
          outcnt =outcnt-1;
          seek incnt;
          CHECKSTATUS;
          if   incnt <> rxtrk  then error 11
          incnt =incnt + 1;
         end;
  end  { oscillating seek test. }
```

   In this manner, all seeks are performed in both directions with all seek counts between <0:77>. Verification is performed on each track to check the step logic.

7. **Test 7 - Sequential Read/Write Test** - performs the basic patterning of the diskette with a background pattern. This test is user-selected. If selected, this test writes each LBN on the RX33 diskette in ascending order with a unique pattern consisting of the track, sector, and side of that LBN, and then an incrementing-byte pattern for the remainder of the 512-byte sector. Each LBN so written is then read back, and each word compared to the data that was written. This test takes about 10 minutes per drive.

8. **Test 8 - Random Reads/Writes** - does random reads and writes to the selected drives. If both drives are selected for test, operations on each drive are performed in groups of five.

   This test runs until the allotted time for the exercise expires, or the user terminates the test with a CTRL/C. The mechanism of this test is as follows:

   A random number is generated. The value of this number determines if the operation is a read or a write, and which LBN is used.

   If the command is a read, the appropriate LBN is read from the disk. The header bytes (0:5) of the data read are then compared against the values expected. The pattern number bytes (6:7) are then compared against a list to see which pattern should be used to compare the rest of the buffer (10:512).
   If the command is a write, other bits of the random number are used to select one of four different patterns to write to the disk. A buffer is then set up with the correct header bytes for the LBN to be written and the correct background data pattern. This buffer is then written out on the diskette.

   Descriptions of the data patterns used are found in the following section.

6.8.7 RX33 Offline Exerciser Data Patterns

Four unique data patterns were selected to give maximum delta of frequency with the MFM (modified frequency modulation) encoding used on the RX33. These patterns are as follows:

PATTERN NUMBER	PATTERN VALUE
177400	Incrementing by bytes starting at 2404
11111	1000101110001011 binary, 105613 octal
22222	0011001100110011 binary, 031463 octal
33333	0011000010010001 binary, 030221 octal
44444	0000101110001011 binary, 005613 octal

6.9 OFFLINE REFRESH TEST

The Offline Memory Refresh test finds memory problems related to refresh. Patterns are written to memory and then checked after waiting one minute. Three separate patterns are used to test each memory bit (including parity bits) in both the one and zero states. All three HSC memories are tested (Program, Control, and Data), although only the Program and Control memories require refreshing. Tests of Data memory are included because some static RAM failures resemble refresh problems.

The refresh test can find problems in the memories not detected by the normal memory tests. The refresh test is not intended to be run on memories that fail the normal memory tests.

6.9.1 Offline Refresh Test System Requirements

The following hardware is required to run this diagnostic:

- I/O control processor module with HSC70 Boot ROMs
- At least one memory module that passes the Offline Memory test and/or the Offline K/P memory test
- RX33 controller with at least one working drive
- Terminal connected to I/O control processor console interface

This test assumes the HSC memories pass both the Offline Memory Test and the Offline K/P Memory Test. In addition, the test assumes the memories are working except for the refresh circuitry.

6.9.2 Offline Refresh Test Operating Instructions

If the HSC70 is not booted and loaded, refer to Section 6.1.2, Section 6.1.3, and Section 6.2. If the HSC70 is already booted and displaying the Offline Loader prompt (ODL>), proceed as follows:

1. Type **TEST REFRESH** in response to the prompt ODL>.
2. The refresh test indicates it is loaded properly by displaying the following:

   HSC OFL Memory Refresh Test

3. The refresh test now prompts for parameters. Refer to Section 6.9.3 for test parameter entries.

6.9.3 Offline Refresh Test Parameter Entry

This section describes the prompts for the Offline Refresh Test parameters.

NOTE

For any of the Offline Refresh Test prompts, use the DELete key to delete mistyped parameters before the terminating carriage return is typed. If you note an error in a parameter already terminated with a carriage return, type a CTRL/C to return to the initial prompt and re-enter all parameters.

The Offline Memory Refresh Test first prompts:

of passes to perform (D) [1] ?

Enter a decimal number between 1 and 2,147,483,647 (omitting commas) to specify the number of times the refresh test should be repeated. (Entering a 0, or just a carriage return results in one pass.) After selection of the number of passes the test begins. The test can be aborted at any time by typing a CTRL/C. Each pass of the test requires three minutes to complete.

After the refresh test completes, the following prompt is issued:

Reuse parameters (Y/N) [Y] ?

Answering this prompt with a carriage return, or a Y followed by a carriage return, repeats the test using the same parameters. Answering the prompt with a N followed by a carriage return causes a prompt for new parameters.

6.9.4 Offline Refresh Test Progress Reports

Each time the refresh test completes one full pass, an end-of-pass report is displayed. Each pass of the test requires three minutes to complete. The end-of-pass message is displayed as follows:

End of Pass nnnnnn, xxxxxx Errors, yyyyyy Total Errors

The Pass count nnnnnn is a decimal count of the number of complete passes made. The Errors count (xxxxxx) indicates the number of errors detected on the current pass. The Total Errors count (yyyyyy) indicates the number of errors detected during the passes completed so far.

6.9.5 Offline Refresh Test Error Information

All error messages produced by the refresh test conform to the HSC Diagnostic Error message format (refer to Section 6.1.5). Following is a typical Offline Refresh error message.

```
ORFT>hh:mm T aaa E bbb
<Text describing error>
 MA -xxxxxxxx
 EXP-yyyyyy
 ACT-zzzzzz
```

where:

```
 MA = Address of the failing location
EXP = EXPected data
ACT = ACTual data
```

6.9.5.1 Offline Refresh Test Error Messages - The following list describes the nature of the failure indicated by each error number:

- **Error 01** - indicates the test detected a parity error when reading the pattern from the indicated location. The expected and actual data are included in the error report. This error indicates a data bit or parity bit was not refreshed (assuming the memory in question passed the Offline Memory Test). If the expected and actual data are the same, one of the parity bits was not refreshed.

- **Error 02** - indicates the test detected a data compare error when reading the pattern from the indicated location. The expected and actual data are displayed in the error report. Note, a parity error did not occur so more than one bit must have failed to refresh.

- **Error 03** - indicates the I/O control processor detected a parity error. The 22-bit address of the location that caused the trap is displayed as the MA data in the error report, where:

  ```
   MA = Address causing the parity trap.
  VPC = Virtual PC of the memory test at the time the
        trap occurred. Reference this address in the
        listing to locate the area of the test where
        the error occurred.
  ```

  Because the data is lost when a parity trap occurs, no EXPected or ACTual data can be displayed. The parity error occurred within the program itself not within the memory being tested. After the trap is reported, the program attempts to restart the test from the beginning.

- **Error 04** - indicates the I/O control processor detected a Non-Existent Memory trap. An NXM error is caused when no memory responds to a particular address. The MA data in the error report indicates the address that produced the NXM trap. After the trap is reported, the program

attempts to restart the test from the beginning. (The MA and VPC fields have the same meanings as those in Error 03.)

If this error is at a memory address that should be in your memory configuration, the memory in question is not supplying an ACK to the I/O control processor when the specified address is presented on the memory bus. The most probable point of failure is the logic on the memory module that compares addresses on the memory bus with the range of addresses to which the module should respond. The comparator itself could be faulty or the [C IN, C OUT], [D IN, D OUT] or [P IN, P OUT] lines on the backplane could be in error.

- **Error 05** - Cache Parity Trap, VCP = xxxxxx - Indicates the J11 took a trap through the parity error vector during the run of the diagnostic. This is a cache error. The virtual PC at the time of the trap is printed.

6.9.6 Offline Memory Refresh Test Summaries

The following are the test summaries for the Offline Memory Refresh Test:

- **Test 01 - Pattern 17777777** - fills the memories with the pattern 177777. This sets all data bits and also sets the upper and lower byte parity bits. The entire Control and Data memories are filled with the pattern. All of Program memory not occupied by the Refresh test and the Offline Loader is also filled with the pattern. After filling the memories, the program delays for one minute, then each memory location is read and checked for the pattern. Any errors detected are reported on the terminal.

- **Test 02 - Pattern 000000** - fills the memories with the pattern 000000. This clears all data bits and sets the upper and lower byte parity bits. The entire Control and Data memories are filled with pattern. All Program memory not occupied by the Refresh test and the Offline Loader is also filled with the pattern. After filling the memories, the program delays for one minute, then each memory location is read and checked for the pattern. Any errors detected are reported on the terminal.

- Test 03 - Pattern 100001 - fills the memories with the pattern 100001. This sets data bits 0 and 15 and clears data bits 1 through 14. Both parity bits are also cleared. The entire Control and Data memories are filled with the pattern. All of Program memory not occupied by the Refresh test and the Offline Loader is also filled with the pattern. After filling the memories, the program delays for one minute, then each memory location is read and checked for the pattern. Any errors detected are reported on the terminal.

6.10 OFFLINE OPERATOR CONTROL PANEL TEST

The Offline Operator Control Panel (OCP) test checks the operation of the HSC lamps and switches. Testing includes the five OCP lamps and switches, the State LED, and the Secure/Enable switch, and the Enable LED.

This section includes troubleshooting procedures for localizing faults detected by this test.

6.10.1 Offline Operator Control Panel Test System Requirements

The following hardware is required to run this test:

- I/O control processor module with HSC70 Boot ROMs
- At least one memory module
- RX33 controller with at least one working drive
- Terminal connected to I/O control processor console interface
- Operator Control Panel

Due to the sequence of tests that precede this test, you can assume the I/O control processor module, Program memory, and RX33 are tested and working.

6.10.2 Operator Control Panel Test Operating Instructions

If the HSC70 is not booted and loaded, refer to Section 6.1.2, Section 6.1.3, and Section 6.2. If the HSC70 is already booted and displaying the Offline Loader prompt (ODL>), proceed as follows:

Type **TEST OCP** in response to the ODL> prompt. The RX33 drive in motion LED should be ON.

The test indicates it is loaded properly by displaying the following message:

The test then prompts for parameters. Refer to Section 6.10.3 for test parameter entry.

6.10.3 Offline Operator Control Panel Test Parameter Entry

The test first checks the position of the Secure/Enable switch, via a bit in the I/O control processor Control and Status Register (address 17770040). If the switch is in the SECURE position, the following prompt is issued. Otherwise, the test skips to the next prompt:

Put Secure/Enable switch into ENABLE position

If the Secure/Enable switch is in the ENABLE position and the above prompt is issued anyway, a problem is indicated with the bit in the I/O control processor CSR that monitors the Secure/Enable switch. Refer to the troubleshooting procedures in Section 6.10.6. The program waits until the Secure/Enable switch is changed to the ENABLE position and issues the following message:

(Enable LED should be lit, State LED should be blinking)

Check to verify the Enable LED is lit and the OCP State LED is blinking. There are two State LEDs, one is to the left of the Init switch on the HSC OCP, the other is located on the I/O control processor module (the fourth LED from the bottom of the rightmost module in the HSC card cage). If either LED is not blinking, refer to the troubleshooting procedures in Section 6.10.6. The test next prompts for a lamp test:

Press Fault (all OCP lamps should light) (Y/N) [Y] ?

Press the Fault lamp and observe that all OCP lamps light. If none of the lamps light, a problem may be present in the lamp test logic on the OCP assembly. If all lamps light properly, type a carriage return to continue the test. If the lamp test fails, replace the OCP.

Next, the program checks that all OCP switches are OFF (out position). If any switch bits in the I/O control processor Switch/Display register read as ones (ON), the program lights the lamps for those switches and prompts:

Put all lit switches in OFF (out) position (Y/N) [Y] ?

If the Fault or Init lamps are lit (nonlocking switches), a problem exists with the wiring in those switches or with their respective bits in the Switch/Display register. Replace the OCP.

Otherwise press all lit switches to release their locks and type a carriage return. If the message repeats, and one or more lamps remain lit even though the switches are OFF (out position), refer to the troubleshooting procedures in Section 6.10.6.

The program then tests each of the OCP switches, one at a time. A switch lights and the following prompt is displayed:

Press and release the lit switch

Press the switch that is lit. The program allows about one second for the switch to be released after it is pressed and then continues to the next prompt. If the program fails to respond when a switch is pressed, refer to the troubleshooting procedures in Section 6.10.6. For those switches that lock in the ON position (Online switch and the two unmarked switches), the program prompts:

Press and release the lit switch again

Press the switch again to return it to the OFF (out) position. If the Online switch or either of the unmarked switches fails to lock in the ON position, the switch is defective, and the OCP should be replaced.

After the OCP switch tests are complete, several features of the Secure/Enable switch are tested. The program begins these tests by prompting:

Put Secure/Enable switch into SECURE position

The program waits until the Secure/Enable switch is in the proper position before continuing. If the program fails to respond when the switch is moved to the SECURE position, refer to the troubleshooting procedures in Section 6.10.6. When the program detects the switch is in the SECURE position, it prompts with:

(Enable LED should turn off)

Ensure the Enable LED is off. If this LED fails to turn off when the switch is in the SECURE position, a short or wiring problem is probable.

Next, the program prompts:

Press Init (HSC should not re-boot) (Y/N) [Y] ?

Press the Init switch. When the Secure/Enable switch is in the SECURE position, pressing the Init switch should have no effect. (Do not press any other switch or an error message results.) If the HSC70 starts to perform a bootstrap, (Init lamp turns on and green LED on I/O control processor turns off), the Secure/Enable switch is not disabling the action of the Init switch. After pressing the Init switch, type a carriage return to continue.

The test responds with the following prompt:

Press terminal BREAK key (HSC should not halt) (Y/N) [Y] ?

Press the BREAK key as directed. When in SECURE mode, the BREAK key should not cause the J11 processor to halt (enter ODT.) If the terminal displays the @ character when BREAK is pressed, the Secure/Enable switch is not disabling the action of the BREAK key. Refer to the troubleshooting procedures in Section 6.10.6. After pressing the BREAK key, type a carriage return to continue the test. The final prompt of the test is:

Put Secure/Enable switch into ENABLE position.

The test waits until the Secure/Enable switch is returned to the ENABLE position. At that point the test terminates and returns to the Offline Loader.

The test may be aborted at any time by typing a CTRL/C.

6.10.4 Offline Operator Control Panel Test Error Information

All error messages produced by this test conform to the HSC Diagnostic Error message format. Refer to Section 6.1.5. Listed below is a typical Offline Operator Control Panel Test error message format:

```
OOCP>hh:mm T aaa E bbb
    <Text describing error>
     MA -xxxxxxxx
     EXP-yyyyyy
     ACT-zzzzzz
```

6.10.4.1 Offline Operator Control Panel Test Error Messages -

The following list describes the nature of the failure indicated by each error number:

- **Error 000 - Wrong Bit Set** - occurs when the test detects a switch bit set in the I/O control processor Switch/Display register other than the switch bit being tested. This error can be caused by:
  - The operator pressing the wrong switch.
  - A short causing an additional switch bit to set along with the expected bit.
  - A wiring error causing the wrong bit to set when a switch is pressed.

The MA (Media Address) field of the error report gives the address of the I/O control processor Switch/Display register. The EXPected and ACTual data in the error report show the switch bit the program expected to find set and the bit or bits that actually were set.

If the EXPected data and the ACTual data each consist of only one bit, the failure was either caused by the operator pressing the wrong switch or by a wiring error. If the ACTual data consists of two or more set bits, a short between switches is likely. Refer to the troubleshooting procedures in Section 6.10.6.

- **Error 001 - Bit Set When Init Pressed** - occurs when the Init switch is pressed while the HSC is in the SECURE mode (Test 008). This error can be caused by one of the following:

  - Pressing some switch other than the Init switch.

  - Pressing the Init switch, causing a switch bit in the I/O control processor Switch/Display register to set.

  The MA (Media Address) field of the error report gives the address of the I/O control processor Switch Display register. The EXPected data is always zero (no bit is expected to set). The ACTual data shows the bit or bits that read as a 1 when the Init switch was pressed. Refer to the troubleshooting procedures in Section 6.10.6.

6.10.5 Offline Operator Control Panel Test Summaries

The following sections summarize Test 000 through Test 009:

- **Test 000 - Observe Enable and State LEDS** - is performed by the operator, because the program cannot tell whether the Enable or State LEDs are lit. If the Enable LED is off, a wiring problem may be the cause (LED not connected to power/ground source) or the LED itself may be faulty.

  If the State LED on the OCP fails to blink, check the State LED on the I/O control processor module (fourth LED from the bottom of the rightmost module in the HSC70 card cage). If neither State LED is blinking, the problem is probably caused by the bit in the I/O control processor CSR register that controls the State LED. (Refer to Section 6.10.6.4.) If one of the State LEDs is blinking but the other is not, the nonblinking LED is probably wired wrong or is faulty.

- **Test 001 - Lamp Test via Fault Switch** - performs an automatic lamp test. When the Fault switch is pressed, all lamps should light and remain lit until the switch is released.

  If none of the lamps light when the Fault switch is pressed, the problem is probably in the lamp test circuitry on the OCP assembly. It is possible all lamps are defective or they are not installed. Replace the OCP.

  If some lamps light when Fault is pressed but others do not, replace the OCP.

- **Test 002 - Check All Switches OFF** - reads the I/O control processor Switch/Display register to see if any of the switch bits read as ON (switch bit is a one). If the bit for any switch reads as ON, the corresponding lamp is lit, and the program prompts to turn off any switch that is lit. The program will not proceed until all switch bits read as OFF.

  If a lamp remains ON, even though the corresponding switch is OFF (out position), the switch is either wired incorrectly or the bit in the I/O control processor Switch/Display register for that switch is faulty. Refer to Section 6.10.6.1 to localize the problem.

- **Test 003 - Fault Switch** - directs pressing the lit switch. The program then monitors the switch bits in the I/O control processor Switch/Display register and waits for the Fault switch bit to set. If any other switch bit sets, an error is reported and the program terminates.

  If pressing the Fault switch has no effect, one of the following could be the cause:

  - Fault switch is broken.
  - Fault switch is not wired properly.
  - Fault switch bit in the I/O control processor CSR cannot be set.

  Refer to the troubleshooting procedures in Section 6.10.6.

  If pressing the Fault switch results in an error message, refer to Section 6.10.4.1.

- **Test 004 - Online Switch** - directs pressing the lit switch. The program then monitors the switch bits in the I/O control processor Switch/Display register and waits for the Online switch bit to set. If any other switch bit sets, an error is reported and the program is terminated.

  After the Online switch bit sets in the I/O control processor Switch/Display register, the program directs you to press the lit switch again returning it to the OFF (out) position. Then the program waits until the switch bit reads OFF (0) before proceeding to the next test.

  If pressing the Online switch has no effect, one of the following could be the cause:

  - Online switch is broken.
  - Online switch is not properly wired.
  - Online switch bit in the I/O control processor CSR cannot be set.

  Refer to the troubleshooting procedures in Section 6.10.6.

  If pressing the Online switch results in an error message, refer to Section 6.10.4.1.

- **Test 005 - First Unmarked Switch** - directs pressing the lit switch. The program then monitors the switch bits in the I/O control processor Switch/Display register and waits for the first unmarked switch bit to set. If any other switch bit sets, an error is reported and the program is terminated.

  After the first unmarked switch bit sets in the I/O control processor Switch/Display register, the program directs you to press the lit switch again returning it to the OFF (out) position. Then the program waits until the switch bit reads OFF (0) before proceeding to the next test.

  If pressing the first unmarked switch has no effect, one of the following could be the cause:

  - First unmarked switch is broken.
  - First unmarked switch is not wired properly.
  - First unmarked switch bit in the I/O control processor CSR cannot be set.

Refer to the troubleshooting procedures in Section 6.10.6.

If pressing the first unmarked switch results in an error message, refer to Section 6.10.4.1.

- **Test 006 - Second Unmarked Switch** - directs pressing the lit switch. The program then monitors the switch bits in the I/O control processor Switch/Display register and waits for the second unmarked switch bit to set. If any other switch bit sets, an error is reported and the program terminates.
  After the second unmarked switch bit sets in the I/O control processor Switch/Display register, the program directs you to press the lit switch again, returning it to the OFF (out) position. Then the program waits until the switch bit reads OFF (0) before proceeding to the next test.

  If pressing the second unmarked switch has no effect, one of the following could be the cause:

  - Second unmarked switch is broken.
  - Second unmarked switch is not properly wired.
  - Second unmarked switch bit in the I/O control processor CSR can not be set.

  Refer to the troubleshooting procedures in Section 6.10.6.

  If pressing the second unmarked switch results in an error message, refer to Section 6.10.4.1.

- **Test 007 - Enable LED Off** - begins with a prompt to put the Secure/Enable switch into the SECURE position. The program waits until bit 15 of the I/O control processor Control and Status register reads as a zero, indicating the switch is in the SECURE position. Then the program tells the operator to observe the Enable LED is OFF.

  If the Enable LED fails to turn off when the switch is in the SECURE position, replace the OCP.

- **Test 008 - Init Switch in Secure Mode** - checks that the Init switch has no effect when the Secure/Enable switch is in the SECURE position. You are prompted to press the Init switch while the program monitors the switch bits in the I/O control processor Switch/Display register. Monitoring ensures that pressing the Init switch does not cause any switch bits to set.

If pressing the Init switch causes the HSC70 to reboot, the SECURE position of the Secure/Enable switch is not disabling the Init switch.

Replace the OCP.

If pressing the Init switch causes one of the switch bits in the Switch/Display register to set, an error message is displayed. Refer to Section 6.10.4.1 for further information.

- **Test 009 - BREAK Key in SECURE Mode** - checks the terminal BREAK key has no effect when the Secure/Enable switch is in the SECURE position. (Normally the BREAK key causes the I/O control processor J11 CPU to halt and enter ODT.) You are prompted to press the BREAK key and to observe the HSC70 does not halt.

  If pressing the BREAK key causes the terminal to print an @ symbol, the SECURE position of the Secure/Enable switch is not disabling BREAK from halting the J11 CPU.

6.10.6 Offline OCP Registers And Displays Via ODT

The following paragraphs and layouts are included to assist you with troubleshooting.

6.10.6.1 Offline OCP Test Switch Check Via ODT -

To check the operation of an HSC70 switch, follow this procedure:

1. With the Secure/Enable switch in the ENABLE position, press the terminal BREAK key.

   The I/O control processor J11 CPU should halt and display an @ symbol.

2. Type: **17770042/**

   The contents of address 17770042 (the I/O control processor Switch Display register) are displayed in octal. Refer to the layout of the Switch Display register in Figure 6-1 to locate the switch bits. Each bit is in the 1 state when the associated switch is ON (pressed in).

3. Type a carriage return.

4. You may now type a slash (/) to re-examine the Switch Display register.

5. To restart the Offline Loader (or the diagnostic that was interrupted), type a carriage return, then type a **P** followed by another carriage return.

Using this method, the switch bits of the Switch/Display register can be monitored when various switches are in the ON or OFF position.

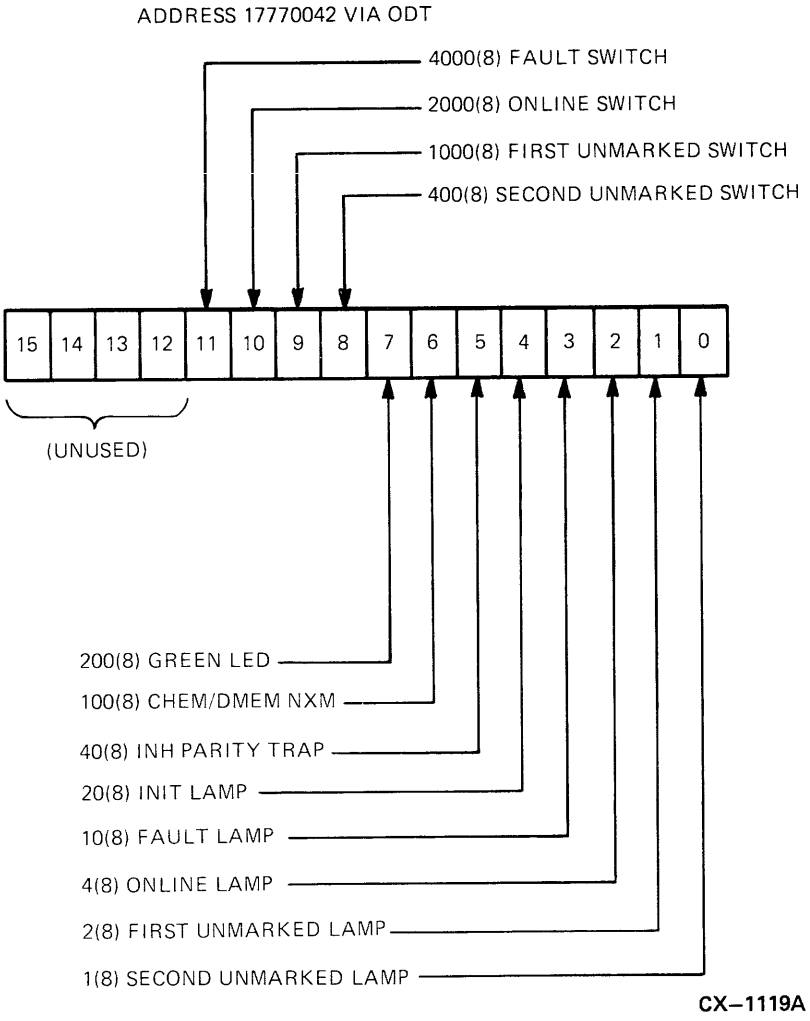


Figure 6-1 P.ioj Switch Display Register Layout

6.10.6.2 Offline OCP Test Lamp Bit Check Via ODT - To check the operation of the lamp control bits in the I/O control processor Switch/Display register, use the following method:

1. With the Secure/Enable switch in the ENABLE position, press the terminal BREAK key.

   The I/O control processor J11 CPU should halt and display an @ symbol.

2. Type 17770042/

   The contents of the Switch/Display register are displayed in octal.

3. Use Figure 6-1 to locate the bits controlling the OCP lamps.

   When a lamp bit is set, the corresponding lamp should be lit.

4. To light a lamp, type the octal value that corresponds to the proper lamp, then type a carriage return. The lamp should light.

5. Type / to re-examine the contents of the Switch/Display register.

6. Type a carriage return to restart the Offline Loader (or the diagnostic that was interrupted), then type a P followed by another carriage return.

Using this method, various lamps can be manually enabled or disabled.

6.10.6.3 Offline OCP Test Secure/Enable Switch Check Via ODT - To manually check the operation of the Secure/Enable bit in the I/O control processor Control and Status register, use the following procedure. Using this method, the Secure/Enable bit in the I/O control processor CSR can be checked with the Secure/Enable switch in both positions.

1. With the Secure/Enable switch in the ENABLE position, press the terminal BREAK key. (If the HSC70 is stuck in the SECURE mode, this method cannot be used, because BREAK is disabled.)

2. The I/O control processor J11 CPU halts and displays an @ symbol.

3. Type 17770040/

4. The content of the I/O control processor Control and Status register is displayed in octal. Refer to Figure 6-2 to identify the various bits of this register.

   When the Secure/Enable switch is in the ENABLE position, the contents of the register should be 1xxxxx. When in the SECURE position, the contents should be 0xxxxx.

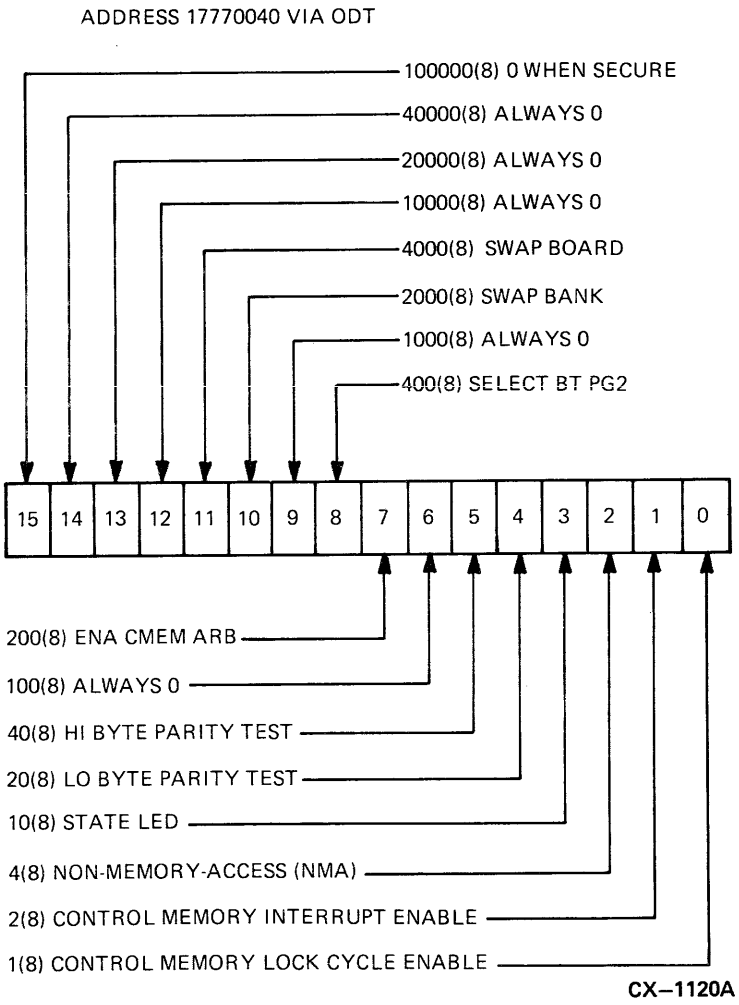


Figure 6-2 P.ioj Control and Status Register Layout

5. Type a carriage return and a / (slash) to re-examine the register,

6. Type a carriage return, then type a **P** followed by another carriage return to restart the Offline Loader (or the diagnostic that was interrupted).

6.10.6.4 Offline OCP Test State LED Check Via ODT - There are two State LEDs in the HSC70. One is on the OCP, far left. The other State LED is on the I/O control processor module (rightmost module in the HSC70 card cage, fourth LED from the bottom of the module). Both LEDs are controlled by a bit in the I/O control processor Control and Status register. (Refer to Figure 6-2 for a layout of this register.) To manually control the State LED, use the following procedure:

1. With the Secure/Enable switch in the ENABLE position, press the terminal BREAK key.

   The I/O control processor J11 CPU should halt and display an @ symbol.

2. Type 17770040/

   The contents of the Control and Status register are then displayed in octal.

3. Use Figure 6-2 to find the octal value corresponding to the State LED.

4. To light the State LED, type the octal value corresponding to the State LED, followed by a carriage return. To extinguish the State LED, put a zero in the same bit position and press a carriage return.

   CAUTION

   Bit 7 of the I/O control processor CSR must be set to allow the HSC70 Ks to access Control memory. The setting of other bits in the CSR can result in strange side-effects. Be careful not to set any bits except the State LED bit and leave bit 7 set when you are done.

5. Type a slash (/) to re-examine the contents of the I/O control processor CSR.

6. To restart the Offline Loader (or the diagnostic that was interrupted), type a carriage return, then type a P followed by another carriage return.

CHAPTER 7
UTILITIES

7.1 INTRODUCTION

This chapter contains the information required to run three of the offline utilities: DKUTIL (Disk Utility), FORMAT, VERIFY, RXMFT (RXFORMAT Utility), and VTDPY (Video Terminal Display Utility). Topics include initiating the utility, using commands, and interpreting error messages. These HSCS70 utilities are interactive and therefore are prompt-oriented. Note that prompt information displayed in square brackets is the default.

For information on the other HSC utilities, refer to the *HSC User Guide*. Utilities described in that manual include:

- SETSHO
- BACKUP Package
- DKCOPY
- PATCH
- COPY

7.2 OFFLINE DISK UTILITY (DKUTIL)

DKUTIL is a general utility for displaying disk structures and disk data. Unlike other utilities, DKUTIL is a command language interpreter. Initially, the user is prompted for the unit number of the appropriate disk. The program then goes into command mode prompting for a command, executing it, and then prompting for another. Execution is terminated by CTRL C, CTRL Y, CTRL Z, or the EXIT command.

7.2.1 DKUTIL Initialization

DKUTIL is initiated via the standard CRONIC command syntax, **RUN DKUTIL.UTL**. The program prompts for the unit number of the disk to examine:

```
DKUTIL-Q Enter unit number (U) [D0]?
```

Reply with the appropriate unit number. The first block of the Format Control table (FCT) is read, if possible, and dumped in a format similar to a VERIFY printout. The unit is brought online with the ignore media format error modifier so drives improperly or not completely formatted can be examined. If the FCT cannot be read or the mode is invalid, the program prompts for the sector size:

```
DKUTIL-Q Enter sector size (512/576) [512]?
```

The program places the unit in diagnostic mode to access the DBN area. After the initial prompts, DKUTIL goes into command mode and prompts for a command.

```
DKUTIL>
```

Comment lines can be entered by prefixing them with an exclamation point (!). Entering a CTRL Z terminates the program. Commands are executed immediately and take only the time necessary to print the results. A CTRL Y or CTRL C at any time aborts the program and releases the drive.

7.2.2 DKUTIL Command Syntax

The DKUTIL commands are:

- DEFAULT
- DISPLAY
- DUMP
- EXIT
- GET
- POP
- PUSH
- REVECTOR
- SET

Any initial substring recognizes commands, command options, and modifiers. For example, DUMP can entered as DUM, DU, or D. In cases where the initial substring can indicate one of several commands, the match depends on an order based on history and expected frequency of usage. Thus, D specifies DUMP, DI specifies DISPLAY, and DE specifies DEFAULT. In the following descriptions, only the part of the command or command in **bold print** must be specified.

Some command options take optional parameters which, if omitted, default.

7.2.3 DKUTIL Command Modifiers

Modifiers, specified only for commands that allow them, can occur anywhere after the command itself. They are preceded by a slash (one slash for each modifier). The following are equivalent:

```
DUMP/NOEDC RBN 0
DUMP /NOEDC RBN 0
DUMP RBN/NOEDC 0
DUMP RBN 0/NOEDC
DUMP RBN 0 /NOEDC
```

Modifiers are processed left to right and applied to the current default modifiers. The DUMP command is the exception. The default modifiers for DUMP can be changed via the DEFAULT command. In the following descriptions, only the portion of the modifier in **bold print** needs to be specified. The initial default modifiers for DUMP are /DATA, /EDC, and /IFERROR.

7.2.4 DKUTIL Sample Session

The following is a sample session using DKUTIL. Command input is indicated in **bold print**.

```
HSC> RUN DX0:DKUTIL

DKUTIL-Q Enter unit number (U) [D0]?D133

   Serial Number:        0000000004
   Mode:                 512
   First Formatted:      17-Nov-1858 00:35:47.48
   Date Formatted:       04-Apr-1984 00:05:09.20
   Format Instance:      6
   FCT:                  VALID

DKUTIL> DIS/F FCT

            Factory Control Table for D133 (RA80)

   Serial Number:        0000000004
   Mode:                 512
   First Formatted:      17-Nov-1858 00:35:47.48
   Date Formatted:       04-Apr-1984 00:05:09.20
   Format Instance:      6
   FCT:                  VALID

   Bad PBNs in FCT:      1 (512), 0 (576)

   Scratch Area Offset: 63
   Size (Not Last):      417
   Size (Last):          289

   Flags:                000000
   Format Version:       0
```

```
PBNs in 512 Byte Subtable (04) 244865 (LBN 237213),

DKUTIL> REV 1000
ERROR-W Bad Block Replacement (Success) at 04-Apr-1984 17:47:24.20
        Command Ref #    00000000
        RA80 Unit #      133.
        Err Seq #        6.
        Error Flags      80
        Event            0014
        Replace Flags    A400
        LBN              1000.
        Old RBN          32.
        New RBN          33.
        Cause Event      004A
ERROR-I End of error.

DKUTIL> DIS/F RCT


           Revector Control Table for D133 (RA80)

   Serial Number:        0000000004
   Flags:                000000

   LBN Being Replaced:   1000 (000000 001750)
   Replacement RBN:      33 (060000 000041)
   Bad RBN:              32 (060000 000040)

   Cache ID:             0000000000
   Cache Incarnation:    0
   Incarnation Date:     17-Nov-1858 00:00:00.00

 Bad RBN:      32,   1000 *->    33,  25512 -->   822,
 139512 -->  4500,

   RCT Statistics:          1 Bad RBNs,
                            3 Bad LBNs,
                            2 Primary Revectors,
                            1 Non-Primary Revectors,
                            0 Probationary RBNs.

DKUTIL> DEF/NODATA
DKUTIL> DUMP LBN 1000

Buffer for LBN 1000 (000000 001750), MSCP Status: 000000

 Error Summary = header compare

 Original Error Bits  = 004000     BN = 1000 (000000 001750)
 Error Recovery Flags = 000        ECC Symbols Corrected = 0,0
 Error Retry Counts   = 0,1,0      Error Recovery Command = 000

   Header = 001750 030000 001750 030000 001750 030000 001750 030000
```

```
        EDC = 000105    Calculated EDC Difference = 000000

        ECC = 000000 000000 000000 000000 000000 000000
              000000 000000 000000 000003 000000 000000

DKUTIL> DIS CHAR LBN 1000

        Characteristics for LBN 1000 (000000 001750)

 Cylinder 1, Group 0, Track 4, Position 8

 PBN 1032 (000000 002010)

 Primary RBN 32 (060000 000040) in RCT Block 3 at Offset 128

DKUTIL> DIS CHAR DISK

        Drive Characteristics for D133

 Type:          RA80 (576 byte mode allowed)

 Media:         FIXED

 Cylinders:     275 LBN, 2 XBN, 2 DBN

 Geometry:      14 tracks/group, 2 groups/cylinder, 28 tracks/cylinder
                31 LBNs/track, 1 RBNs/track, 32 sectors/track, 32 XBNs
                896 XBNs/cylinder, 868 LBNs/cylinder, 28 RBNs/cylinder

 Group Offset:  16 (LBN), 16 (XBN)

 LBNs:          237212 (host), 238700 (total)

 RBNs:          7700

 XBNs:          1792

 DBNs:          1344 (read/write), 448 (read only)

 PBNs:          249984

 RCT:           465 (size), 63 (non-pad), 4 (copies)

 FCT:           480 (size), 63 (non-pad), 4 (copies)

 SDI Version:   3

 Transfer Rate: 97

 Timeouts:      3 (short), 7 (long)

 Retry Limit:   5
```

```
 Error Recover: 0 command levels

 ECC Threshold: 2 symbols

 Revision:      10 (microcode), 0 (hardware)

 Drive ID:      0A7A00000000

 Drive Type ID: 1

 DBN RO Groups: 1

 Preamble Size: 11 (data), 4 (header)

DKUTIL> DUMP RCT BLOCK 3

                          RCT Block 3, Copy 1

      Buffer for LBN 237214 (000003 117236), MSCP Status: 000000

Data = 000000 000000 000000 000000 000000 000000 000000 000000
  +16  000000 000000 000000 000000 000000 000000 000000 000000
  +32  000000 000000 000000 000000 000000 000000 000000 000000
  +48  000000 000000 000000 000000 000000 000000 000000 000000
  +64  000000 000000 000000 000000 000000 000000 000000 000000
  +80  000000 000000 000000 000000 000000 000000 000000 000000
  +96  000000 000000 000000 000000 000000 000000 000000 000000
 +112  000000 000000 000000 000000 000000 000000 000000 000000
 +128  000000 040000 001750 030000 000000 000000 000000 000000
 +144  000000 000000 000000 000000 000000 000000 000000 000000
 +160  000000 000000 000000 000000 000000 000000 000000 000000
 +176  000000 000000 000000 000000 000000 000000 000000 000000
 +192  000000 000000 000000 000000 000000 000000 000000 000000
 +208  000000 000000 000000 000000 000000 000000 000000 000000
 +224  000000 000000 000000 000000 000000 000000 000000 000000
 +240  000000 000000 000000 000000 000000 000000 000000 000000
 +256  000000 000000 000000 000000 000000 000000 000000 000000
 +272  000000 000000 000000 000000 000000 000000 000000 000000
 +288  000000 000000 000000 000000 000000 000000 000000 000000
 +304  000000 000000 000000 000000 000000 000000 000000 000000
 +320  000000 000000 000000 000000 000000 000000 000000 000000
 +336  000000 000000 000000 000000 000000 000000 000000 000000
 +352  000000 000000 000000 000000 000000 000000 000000 000000
 +368  000000 000000 000000 000000 000000 000000 000000 000000
 +384  000000 000000 000000 000000 000000 000000 000000 000000
 +400  000000 000000 000000 000000 000000 000000 000000 000000
 +416  000000 000000 000000 000000 000000 000000 000000 000000
 +432  000000 000000 000000 000000 000000 000000 000000 000000
 +448  000000 000000 000000 000000 000000 000000 000000 000000
 +464  000000 000000 000000 000000 000000 000000 000000 000000
 +480  000000 000000 000000 000000 000000 000000 000000 000000
 +496  000000 000000 000000 000000 000000 000000 000000 000000
```

```
 EDC = 023277     Calculated EDC Difference = 000000
DKUTIL> EXIT
```

7.2.5 DKUTIL Command Descriptions

Descriptions for the individual DKUTIL commands are found in Table 7-1. In the syntax and modifier specifications, the characters displayed in **bold print** are the minimum allowed abbreviations. Command options are shown by separate lines in the syntax specification. Parameters are indicated in the syntax by braces ({}) and lower case. Options indicated by brackets ([]) can be omitted.

Table 7-1 DKUTIL Command Summary

Command	Description
DEFAULT	Change default modifiers for DUMP command.
DISPLAY	Display characteristics, error history, RCT, or FCT.
DISPLAY ALL	
DISPLAY CHARACTERISTICS DBN {block}	
DISPLAY CHARACTERISTICS DISK	
DISPLAY CHARACTERISTICS LBN {block}	
DISPLAY CHARACTERISTICS PBN {block}	
DISPLAY CHARACTERISTICS RBN {block}	
DISPLAY CHARACTERISTICS XBN {block}	
DISPLAY ERRORS	
DISPLAY FCT	
DISPLAY RCT	
DUMP	Dump given block or table of blocks.
DUMP [BUFFER]	
DUMP DBN [{block}]	
DUMP FCT [BLOCK {number}] [COPY {copy}]	

Command	Description
DUMP LBN [{block}]	
DUMP RBN [{block}]	
DUMP RCT [BLOCK {number}] [COPY {copy}]	
DUMP XBN [{block}]	
EXIT	Terminate execution of the program.
GET	Change the current drive.
GET [{drive}]	
POP	Restore save buffer to current buffer.
PUSH	Save current buffer in save buffer.
REVECTOR	Force bad block replacement for the given LBN.
SET	Change various program parameters.
SET [SIZE {size}]	

7.2.5.1 **DKUTIL DEFAULT Command** - The DEFAULT command is outlined as follows:

Purpose: To change the default modifiers for the DUMP command.

Syntax: DEFAULT

Parameters: None

Modifiers: Shown in the following list.

> /IFERROR (NOIFERROR) (defaults ON) - dumps the error, header, and ECC fields in the buffer if an error occurs when reading the block. When this modifier is used in conjunction with the /RAW modifier, the error must occur on the reread of the block with the header code extracted from the first read.

/ERRORS (NOERRORS) (defaults OFF) - dumps the error fields in the buffer. If this modifier is specified, the error fields in the buffer are dumped.

/EDC (NOEDC) (defaults ON) - dumps the EDC and calculated EDC fields in the buffer.

/ECC (NOECC) (defaults OFF) - dumps the ECC fields in the buffer.

/DATA (NODATA) (defaults ON) - displays the data in the buffer unless the /NZ modifier was also specified.

/HEADERS (NOHEADERS) (defaults OFF) - displays the header fields in the buffer.

/ALL (NONE) (same as /ERRORS/EDC/ECC/DATA/HEADERS) - requests all fields be displayed. Its opposite, /NONE, requests no fields be displayed. When using the /NONE qualifier, only the MSCP status line prints.

/RAW (NORAW) - allows reading the original LBN that was revectored rather than the RBN that would be read without the /RAW qualifier. /RAW only affects revectored (primary or nonprimary) LBNs. If /IFERROR is in effect, this modifier also applies only to dumping a revectored LBN.

/NZ (NONZ) (defaults OFF) - prevents the data from being displayed if it is all zero. Instead, a single line indicating the data is zero is printed. It has no effect if the /DATA modifier is not specified or if it is defaulted OFF.

/BBR (NOBBR) (defaults OFF) - is usually inhibited when a block is accessed. If this modifier is specified, bad block replacement can occur. It only occurs, however, if the error recovery code detects the block being accessed as bad and the block is an LBN in the host area.

/ORIGINAL (NOORIGINAL) (defaults off) - saves the first data seen for display. When a block is accessed for dumping, the data is seen twice by the program if an error occurs. It is seen first just after the K detects the error and sends it to error recovery. It is seen again after error recovery takes place and the data has been corrected or reread. Usually, the data is saved for displaying when it is last seen.

Usage: The modifiers specified are applied to the current default modifiers for the DUMP command. The result becomes the new default.

Examples: DEFAULT/NONE, DE/A/OR/NZ, and DEF/RAW/NODATA

7.2.5.2 **DKUTIL DISPLAY Command** - The DISPLAY command is outlined as follows:

Purpose: To display the disk characteristics, the characteristics of a given block, the error history in the drive, the FCT, and/or the RCT.

Syntax:

DISPLAY ALL

DISPLAY CHARACTERISTICS DBN {block}

DISPLAY CHARACTERISTICS DISK

DISPLAY CHARACTERISTICS LBN {block}

DISPLAY CHARACTERISTICS PBN {block}

DISPLAY CHARACTERISTICS RBN {block}

DISPLAY CHARACTERISTICS XBN {block}

DISPLAY ERRORS

DISPLAY FCT

DISPLAY RCT

Parameters: Block is a number specifying the DBN, LBN, PBN, RBN, or XBN whose characteristics are displayed. The default radix is decimal, and can be changed to octal by prefixing the number with a zero.

Modifiers:

/FULL - displays all defined fields in xCT block 0. /FULL applies only to the RCT and FCT command options. For the RCT option, the bad block replacement and write back caching fields in RCT block 0 are only displayed if the appropriate flags in the flags field are set. These flags indicate they are currently in use (BBR or caching in progress). This modifier forces all fields to be displayed regardless of the flags settings. For the FCT option, the number of bad PBNs field is normally displayed only if the FCT is valid. Also, the scratch

area parameters, format version, and format flags are normally not displayed. This modifier forces all fields in FCT block 0 to be displayed.

/NOITEMS - does not display the individual items in the FCT or RCT. It applies only to the FCT and RCT command options. If given, only the block 0 information is displayed.

Usage:

DISPLAY ALL - displays FCT, RCT, and error history. Because the error history in the drive is dumped by this option, it should not be used for RA60 drives. Using the SDI command to read RA60 error history is illegal and causes the drive to become inoperative.

DISPLAY CHARACTERISTICS DISK - displays the drive type, media, cylinders, geometry, group offsets, numbers of LBNs, number of RBNs, number of XBNs, numbers of DBNs, number of PBNs, RCT parameters, FCT parameters, SDI version, transfer rate, SDI timeouts, SDI retry limit, error recovery command levels, ECC threshold, revision levels, drive ID, drive type ID, DBN Read/Only groups, and preamble sizes.

DISPLAY CHARACTERISTICS xBN {block} - displays the characteristics of the given block. For DBNs and XBNs, these are the block number in decimal and octal, cylinder, group, track, position, and PBN in decimal and octal. For RBNs, the RCT block number and offset are also displayed. For LBNs, the primary RBN number and its RCT block number and offset are also displayed. For PBNs, the display depends on the type of block: DBN, LBN, RBN, or XBN.

DISPLAY ERRORS - reads the error history in the drive. The error history in the drive is read from region 2, offset 0, and dumped in hexadecimal. This option should not be used for RA60 drives because it causes them to become inoperative.

DISPLAY FCT - displays the information in FCT block 0. Certain fields are not displayed unless the /FULL modifier is given. The list of bad PBNs is displayed unless the /NOITEMS modifier is given. For each item in the list, the header bits, PBN number, type (DBN, LBN, RBN, or XBN), and xBN number are displayed.

DISPLAY RCT - displays the information in RCT block 0. Certain fields are not displayed unless the /FULL modifier is given. The list of revectors, bad RBNs, and probationary RBNs are displayed unless the /NOITEMS

modifier is given. For bad and probationary RBNs, just the RBN number is displayed (in decimal). For revectors, the LBN number and RBN number to which it is revectored are displayed (in decimal). A primary revector is distinguished by the character sequence "-->". A nonprimary revector is distinguished by the character sequence "*->".

Examples: DISPLAY/FULL ALL, DI/F A, DI C D, DIS CHAR LBN 1000, DI/NOI RCT

7.2.5.3 **DKUTIL DUMP Command** - The DUMP command is outlined as follows:

Purpose: To dump the given block or table of blocks.

Syntax:

```
DUMP [BUFFER]

DUMP DBN [{block}]

DUMP FCT [BLOCK {number}] [COPY {copy}]

DUMP LBN [{block}]

DUMP RBN [{block}]

DUMP RCT [BLOCK {number}] [COPY {copy}]

DUMP XBN [{block}]
```

Parameters: Block is a number specifying the DBN, LBN, RBN, or XBN to be dumped. The default radix is decimal. It can be changed to octal by prefixing the number with a zero.

Number is the relative block number in the FCT or RCT to be dumped. The default radix is decimal and can be changed to octal by prefixing the number with a zero. The value must be in the range 1 through nonpad FCT or RCT size. That is, the first block is number 1 (not 0) and the block must lie in the nonpad area.

Copy specifies which copy of the given block in the FCT or RCT is to be dumped. The first copy is number 1. The value must not exceed the number of copies.

Modifiers:

/IFERROR (NOIFERROR) - (defaults ON) dumps the error, header, and ECC fields in the buffer when an error occurs while reading the block. When used in

conjunction with the /RAW modifier, the error must occur on the read of the LBN (reread) with the header code extracted from the RBN (first read). Refer to Section 7.2.5.1.

/ERRORS (NOERRORS) - (defaults OFF) dumps the error fields in the buffer.

/EDC (NOEDC) - (defaults ON) dumps the EDC and calculated EDC fields in the buffer.

/ECC (NOECC) - (defaults OFF) dumps the ECC fields in the buffer.

/DATA (NODATA) - (defaults ON) displays the data in the buffer unless the /NZ modifier was also specified.

/HEADERS (NOHEADERS) - (defaults OFF) displays the header fields in the buffer.

/ALL (NONE) - is the same as /ERRORS/EDC/ECC/DATA/HEADERS. It requests display of all fields. Its opposite, /NONE, requests display of no fields. When using the /NONE qualifier, only the MSCP status line prints.

/RAW (NORAW) - allows a read of the original revectored LBN (rather than the RBN that would be read without the /RAW qualifier). /RAW only affects revectored (primary or non-primary) LBNs. If in effect, the /IFERROR modifier also applies only to dumping a revectored LBN.

/NZ (NONZ) - prevents data from being displayed when it is all zeroes. Instead, a single line prints indicating the data is zeroes. /NZ has no effect unless the /DATA modifier is specified. It also has no effect if /DATA is not specified (or is defaulted OFF).

/BBR (NOBBR) - (defaults OFF) permits bad block replacement. Normally, bad block replacement is inhibited when a block is accessed. BBR occurs if the block being accessed is detected as bad by the error recovery code and is an LBN in the host area.

/ORIGINAL (NOORIGINAL) - saves the first data seen for display. When a block is accessed for dumping, the data is seen twice by the program when an error occurs. It is seen first just after the K detects the error and sends it to error recovery. It is seen again after error recovery takes place and the data has been corrected or reread. Normally, the data is saved for displaying when it is last seen.

Usage: DUMP [BUFFER] The current buffer is dumped subject to the given modifiers. If there is no current buffer, an error message is printed as follows:

DUMP xBN [{block}]

The specified DBN, LBN, RBN, or XBN is read in and dumped subject to the given modifiers. If the block number is not specified, it defaults to zero.

DUMP xCT [BLOCK {number}] [COPY {copy}]

If a BLOCK number is given, that block in the FCT or RCT is read in and dumped. If none is specified, every block in the nonpad area of the FCT or RCT is read in and dumped. If COPY is not specified, it defaults to copy 1.

Examples: DUMP RCT BLOCK 3 COPY 4, DU/NZ RCT C 2, DU LBN 1000, D F B 2, D X, D/DATA

7.2.5.4 DKUTIL EXIT Command - The EXIT command is outlined as follows:

Purpose: To terminate execution of the program.

Syntax: EXIT

Parameters: None

Modifiers: None

Usage: The current drive is released, all resources are returned, and the program exits.

Examples: EXIT, E

7.2.5.5 DKUTIL GET Command - The GET command is outlined as follows:

Purpose: To change the current drive.

Syntax: GET [{drive}]

Parameters: Drive is a valid drive unit specification of the form Dnnn. If this parameter is omitted, GET defaults to D000 (unit 0).

Modifiers:

/NOIMF - inhibits the reading of FCT block 0 to determine the mode and the reading and writing of RCT block 0 to verify the RCT is sane. If this modifier is specified, the IMF MSCP modifier is not used in the online mode and these actions take place. By default, a new drive is brought online with the IMF (MD.IMF) MSCP modifier.

/WP - brings the drive online with the MSCP SET WRITE PROTECT modifier (MD.SWP) and WRITE PROTECT unit flag (UF.WPS). The drive is then software or volume write-protected.

/NOWP - brings the drive online with the MSCP SET WRITE PROTECT modifier. The drive is not software write-protected.

Usage: The current drive is released. The new drive is acquired and then brought online with the requested modifiers and unit flags. If the drive is nonexistent, in use, or inoperative, the program prompts for another unit. The modifiers cannot be changed for this other unit. If the mode word in FCT block 0 is invalid or all copies of FCT block 0 are bad, the program prompts for the sector size to use.

Examples: GET D133, G/WP D64, G

7.2.5.6 DKUTIL POP Command - The POP command is outlined as follows:

Purpose: To restore the data in the current buffer from the save buffer.

Syntax: POP

Parameters: None

Modifiers: None

Usage: The data in the save buffer is restored to the current buffer. The data in the current buffer is lost.

Examples: POP, P

7.2.5.7 DKUTIL PUSH Command - The PUSH command is outlined as follows:

Purpose: To save the data in the current buffer in the save buffer.

Syntax: **PUSH**

Parameters: None

Modifiers: None

Usage: The data in the current buffer is saved in the save buffer. The data in the save buffer is lost.

Examples: PUSH, PU

7.2.5.8 DKUTIL REVECTOR Command - The REVECTOR command is outlined as follows:

Purpose: To force bad block replacement to occur for a given LBN.

Syntax: **REVECTOR** {block}

Parameters: Block is a number specifying the LBN to be replaced. The default radix is decimal. It can be changed to octal by prefixing the number with a zero.

Modifiers: None

Usage: The specified LBN is sent to the bad block replacement module to be revectored. If it is not a valid LBN or in the RCT, the revector fails, and an error message prints. Otherwise, the result of the replace attempt shows in the error log produced (if the appropriate level message level is enabled (INFO)). The data in the replacement RBN is read from the specified LBN.

Examples: REVECTOR 1000, R 100

7.2.5.9 DKUTIL SET Command - The SET command is outlined as follows:

Purpose: To change various program parameters.

Syntax: **SET** [SIZE {size}]

Parameters: The size parameter specifies the new sector size to be used for the current drive. It must be either 512 or 576.

Modifier: /SIZE {size}

The sector size is changed to the given value and the disk parameters are recomputed. This new sector size is used when doing I/O to the LBN area and is also reflected in the parameters printed by the DISPLAY CHARACTERISTICS DISK command.

Examples: SET SIZE 576, S S 512

7.2.6 DKUTIL Error Messages

Table 7-2 contains a list of error and information messages printed out by DKUTIL. These messages are arranged alphabetically.

7.2.6.1 DKUTIL Error Message Variables - Certain portions of the error messages are variable and are shown in **bold print**. The meanings of these variables are as follows:

n = a decimal number
par = BLOCK or COPY
parm = the part of the command in error (modifier, etc.)
status = MSCP status (an octal number)
text = the actual text in error
xBN = DBN, LBN, etc.
xCT = FCT or RCT

7.2.6.2 DKUTIL Error Message Severity Levels - DKUTIL error messages conform to the HSC utility error message format. In each case, the utility name at the start of the message is followed by a letter indicating the severity level of the message. These are defined as follows:

- E = Error
- F = Fatal
- I = Information
- S = Success

Table 7-2 DKUTIL Error Messages

Error Message	Explanation
DKUTIL-S CTRL/Y or CTRL/C Abort!	This termination message prints if you abort DKUTIL by typing CTRL-C or CTRL-Y.
DKUTIL-F Insufficient resources to RUN!	This message prints if DKUTIL cannot acquire the necessary resources to run or if the disk functional code is not loaded. The program terminates after this message is printed.
DKUTIL-F Drive went OFFLINE!	This message prints if the selected unit goes offline while DKUTIL is running. The program terminates after this message is printed.

Error Message	Explanation
DKUTIL-F I/O request was rejected!	This message prints if the diagnostic interface (DDUSUB) rejects a request to start an I/O operation. It indicates a bug in DKUTIL and should be reported to Field Service Support. The program terminates after this message is printed.
DKUTIL-E Illegal response to start-up question.	This error message prints if you enter an invalid response to a start-up question or to a prompt for the GET command. The program reprompts with the same question.
DKUTIL-E Nonexistant unit number.	This error message prints if the unit number entered does not correspond to any known unit. The program reprompts for the unit number.
DKUTIL-E Unit is not available.	This error message prints if the unit requested is unavailable. The unit may be in use by a host or another diagnostic or it may be inoperative. The program reprompts for another unit.
DKUTIL-E Cannot bring unit ONLINE.	This error message prints if the requested unit is available, but the ONLINE command failed. The unit is released, and the program reprompts for another unit.
DKUTIL-E Invalid decimal number.	This error message prints if you entered an invalid decimal number in a command line.
DKUTIL-E Invalid octal number.	This error message is printed if the user entered an invalid octal number in a command line.
DKUTIL-E Missing parameter.	This error message prints if a command line is entered with a required parameter missing.
DKUTIL-E There is no buffer to dump.	This error message prints if the DUMP BUFFER command is entered, and there is no current buffer. This can only happen if a drive has just been selected.

Error Message	Explanation
DKUTIL-E Missing modifier (only "/" was specified).	This error message prints if a command line is entered with a slash (/) followed by a blank or is entered at the end of the line. A modifier is expected, but is missing.
DKUTIL-E SDI command was unsuccessful.	This error message prints when an SDI command is rejected by the drive. A DISPLAY ERRORS command for a RA60 drive always generates this message.
DKUTIL-E n is an invalid par number; maximum is n.	This error message prints if an out-of-range number is entered for a BLOCK or COPY value for the DUMP command.
DKUTIL-E "text" is an invalid parm.	This generic error message prints when an invalid command, invalid command option, invalid modifier, invalid block type, or invalid SET option is specified in a command line.
DKUTIL-E Invalid block number for xBN space.	This error message is printed if the block number specified for a DISPLAY CHARACTERISTICS xBN command is out-of-range for the given space.
DKUTIL-E Copy n of xCT Block n (xBN n) is bad.	This error message prints when FCT or RCT blocks cannot be read correctly with error recovery. It occurs when the FCT or RCT is being read just after a drive has been selected. It also occurs when the DISPLAY FCT or DISPLAY RCT command is being used.
DKUTIL-E All copies of of xCT Block n are bad.	This error message prints when all copies of FCT or RCT blocks are bad. It occurs when the FCT or RCT is being read just after a drive has been selected. It also occurs when the DISPLAY FCT or DISPLAY RCT command is being used.

Error Message	Explanation
DKUTIL-E Invalid sector size; only 512 and 576 are legal.	This error message prints if the sector size entered for the SET SIZE command is other than 512 or 576.
DKUTIL-E Revector for LBN n failed, MSCP Status: status.	This error message prints if a revector (using the REVECTOR command) fails.

7.3 OFFLINE DISK VERIFIER UTILITY (VERIFY)

VERIFY is a utility that checks the integrity of the disk architectural structure. This utility is a tool designed for DIGITAL support personnel to check a disk to ensure it conforms to the DIGITAL Standard Disk format.

VERIFY has many messages that may print during the course of a disk structure verification. These messages have significance only when VERIFY reports the drive is bad. At the end of its run, VERIFY reports the drive is either OK or BAD.

NOTE

The VERIFY utility only reads the disk. It does not destroy user data and does not perform Bad Block Replacement.

The following steps describe the process by which this utility verifies a disk.

1. The first block of the Factory Control Table (FCT) is read to determine how the disk is formatted. The serial number, format mode, date first formatted, date last formatted, format instance, state of the FCT, number of bad PBNs, scratch area parameters (offset, size of not last, and size of last), flags, and format version are printed.

2. The first block of the Revector Control Table (RCT) is then read. The information in it is printed, including the serial number, flags, bad block replacement variables (LBN being replaced, replacement RBN, and bad RBN), and cache variables (ID, incarnation, and incarnation date).

3. All copies of the first two blocks in the RCT (used by bad block replacement) are read and compared. Discrepancies or bad blocks are reported.

4. All copies of the rest of the RCT are read and compared. Any discrepancies or bad blocks are reported. The information about revectors and bad RBNs is dumped. A summary of the number of bad blocks and revectors by type is printed.

5. All copies of FCT block 0 are read and compared, and bad blocks or discrepancies are reported.

6. All copies of the appropriate FCT subtable are read (if not null) and bad blocks or discrepancies are reported.

7. The list of bad PBNs is printed. Each entry is printed with the header bits, PBN number, and xBN number (in parentheses) as separate fields. If a bad PBN is found which should be in the RCT but is not, the xBN field is printed in brackets instead of parentheses. If any such PBNs are found, an error message indicating the total number is printed at the end of the bad PBN list.

8. After reading and dumping the FCT, a quick scan of DBN space is done. Every block is accessed only once. Counts of various detected errors are recorded for a summary printed at the end of the scan. If more than nine positioner errors are detected, a message is printed suggesting DBN space be reformatted. If more than nine EDC errors are detected, a message is printed suggesting the INITIAL WRITE option should be used when running ILEXER.

9. All LBN space up to the RCT and all RBNs are scanned. Any block with an error is reread five more times to determine the type of error. Information about bad blocks and revectors collected in this phase is compared with information collected from reading the RCT. During the scan, four error classes can be found:

   - Structure errors
   - Permanent recoverable errors
   - Permanent unrecoverable errors
   - Transient errors

   Structure and permanent unrecoverable errors are considered inconsistencies and are always reported. Permanent recoverable errors, usually ECC errors, are reported if requested. During the five rereads of a block with an error, a block read at least once with no detected error is considered to have a transient error. Transient errors are reported if you request them.

10. At the end of the scan, certain other errors are reported. Some errors can only be determined at that time by examining information collected during the scan.

11. Finally, a summary, by type, of the errors detected and certain other information is printed. If no inconsistencies were discovered, a message saying the drive is OK prints. Otherwise, the message indicates the number of inconsistencies.

7.3.1 VERIFY Initiation

VERIFY is initiated via the standard CRONIC command syntax, **RUN DX0:VERIFY,UTL**. The following prompt asks for the unit number of the disk to verify.

VERIFY-Q Enter unit number to verify (U) [D0]?

It then prompts to determine if the unit was recently formatted:

VERIFY-Q Was this unit just FORMATted (Y/N) [Y]?

This question is asked because certain errors are classed as inconsistencies only when the unit has not been subject to bad block replacement following the execution of FORMAT. The next prompt determines whether errors not considered inconsistencies should be reported:

VERIFY-Q Print informational (non-warning) messages (Y/N) [N]?

If you reply N to this question, only inconsistencies are reported. If your reply is Y, you are further prompted to decide whether transient errors should be reported:

VERIFY-Q Report transient errors by block (Y/N) [N]?

Regardless of the response to this question, the number of transient errors is printed in the final summary. The response to this question determines whether or not individual blocks with transient errors should be reported.

You can enter CTRL Z at any prompt for the remainder of the responses. CTRL Z forces the default response (in square brackets). Also, the responses to subsequent questions can be supplied at any question by typing them separated with commas. For example, if unit D133 (which was just formatted) is to be verified and all options are to be selected, the user could type **D133,,Y,Y** at the first prompt.

If the unit does not exist or cannot be accessed, you are notified and reprompted for another unit number. If the unit can be accessed, it is acquired and brought online. VERIFY runs to completion, unless aborted by CTRL Y or CTRL C.

7.3.2 VERIFY Sample Session

The following is a sample session using VERIFY. User input is in **bold print.**

```
HSC50> RUN DX0:VERIFY

VERIFY-Q Enter unit number to verify (U) [D0]?D133
VERIFY-Q Was this unit just FORMATted (Y/N) [Y]?
VERIFY-Q Print informational (non-warning) messages (Y/N) [N]?Y
VERIFY-Q Report transient errors by block (Y/N) [N]?Y
   ***    FCT Block 0 Information

   Serial Number:        0000000004
   Mode:                 512
   First Formatted:      17-Nov-1858 00:35:47.48
   Date Formatted:       10-Apr-1984 00:05:09.20
   Format Instance:      6
   FCT:                  VALID
   Bad PBNs in FCT:      1 (512), 0 (576)

   Scratch Area Offset: 63
   Size (Not Last):      417
   Size (Last):          289

   Flags:                000000
   Format Version:       0

   ***    RCT Block 0 Information

   Serial Number:        0000000004
   Flags:                000000

   LBN Being Replaced:   0 (000000 000000)
   Replacement RBN:      0 (060000 000000)
   Bad RBN:              0 (060000 000000)

   Cache ID:             0000000000
   Cache Incarnation:    0
   Incarnation Date:     17-Nov-1858 00:00:00.00

   ***     Revector Control Table for D133

VERIFY-I Copy 1 of RCT Block 2 (LBN 237213.) is bad.
  25512 -->   822, 139512 -->  4500,

   RCT Statistics:          0 Bad RBNs,
                            2 Bad LBNs,
                            2 Primary Revectors,
                            0 Non-primary Revectors,
                            0 Probationary RBNs,
                            1 Bad RCT Blocks,
                            1 Bad First Copy RCT Blocks.

   ***                    Factory Control Table for D133
```

```
                                PBNs in 512 Byte Subtable

  (04) 244865 (LBN 237213),

    ***                 Quick Scan of DBN Area

   Statistics:
                        0 total blocks with any error.

    ***                   Scan of LBN Area

VERIFY-I LBN 26003. has a 1 symbol correctable ECC error.
VERIFY-I RBN 2471. has a 1 symbol correctable ECC error.
VERIFY-I LBN 139962. has a 1 symbol correctable ECC error.

   Statistics:
                        3 total ECC symbols corrected,
                        3 blocks with 1 symbol ECC errors,
                        2 revectors verified,
                        5 total blocks with any error.

VERIFY-I Drive is OK.
```

The preceding example is the output of an actual session for an RA80 disk with one bad PBN in the FCT. Notice this PBN corresponds to copy 1 of RCT block 2. RCT block 2 is used to store the copy of the user data during bad block replacement. In its scan of the RCT, VERIFY noticed this block was bad and printed an informational message indicating that. If informational messages had been suppressed through use of the SETSHO utility, this information would show only in the summary of the RCT dump.

In the example, VERIFY also printed informational messages for the three blocks it found with solid 1-symbol correctable ECC errors. If informational messages had been suppressed, these messages would not have printed. However, the number of such blocks would show up in the summary statistics.

No transient errors were detected and, therefore, no count is reported in the summary statistics. Also note, although no messages were printed for them, the two revectors in the RCT were verified (as indicated in the summary statistics). Note the funny date for the Date Formatted field. This date is the default when no date is supplied by a host or a human during manufacturing format. If structure inconsistencies had been found, some of the following error messages would also print.

7.3.3 VERIFY Errors And Information Messages

This section describes error and information messages that may be printed out by VERIFY. Error messages are arranged alphabetically according to the actual message.

7.3.3.1 VERIFY Variable Output Fields - Error message fields with variable output print are in bold print. Definitions for these fields are:

xCT = FCT or RCT
n = a decimal number
n. = a decimal LBN, RBN, or XBN
xBN = LBN, RBN, or XBN
o = an octal number
t = type code: I or W
x = error: ECC, EDC, etc.

7.3.3.2 VERIFY Error Message Severity Levels - VERIFY error messages conform to the HSC utility error message format. In each case, the utility name at the start of the message is followed by a letter indicating severity level. These are defined as follows:

F = Fatal
I = Information
t = type: either W or I, depending on the error
W = Warning

7.3.3.3 VERIFY Fatal Error Messages - Following is a list of the error messages fatal to the VERIFY utility. The program terminates after printing one of these messages.

- **VERIFY-F All Copies of xCT Block n Are Bad!** - prints if all copies of some block in either the RCT or the FCT are bad. The program cannot continue to run because vital information is missing. In any case, it has verified that the unit is bad!

- **VERIFY-F Current System Sector Size is 512!** - prints if the mode field in FCT block 0 indicates the unit is formatted in 576-byte mode, but the system sector size is set to 512. In this case, VERIFY cannot run because it cannot read sectors 576 bytes long.

- **VERIFY-F Drive Went OFFLINE!** - prints if the unit selected goes offline while VERIFY is running.

- **VERIFY-F Insufficient Resources to Run!** - prints if VERIFY cannot acquire the necessary resources to run or the disk functional code is not loaded.

- **VERIFY-F I/O Request Was Rejected!** - prints if the diagnostic interface (DDUSUB) rejects a request to start an I/O operation. It is an indication of a bug in VERIFY and should be reported to Field Service Support.

- **VERIFY-F Mode is Bad or Format is in Progress on This Unit!** - prints if the mode field in FCT block 0 of the selected unit is not valid.

7.3.3.4 **VERIFY Information Messages** - The following messages are informational only.

- **VERIFY-I CTRL/Y or CTRL/C Abort!** - prints if the user aborts VERIFY by typing a CTRL Y or CTRL C.

- **VERIFY-I Drive is OK.** - is a termination message and prints at the end of VERIFY if no inconsistencies were discovered.

- **VERIFY-I There Were n Inconsistencies Found for This Drive.** - is a termination message and prints at the end of VERIFY if inconsistencies were discovered.

7.3.3.5 **VERIFY Warning Messages** - The following messages are warning messages. In many cases, they are true warnings; in other cases, they simply precede a reprompt.

- **VERIFY-W n Bad PBNs (in Brackets Above) Not in the RCT.** - prints if the LBN/RBN count is anything other than zero. After the RCT has been collected, the appropriate subtable of the FCT is read. The list of PBNs is printed. The RCT is searched for RBNs and LBNs corresponding to PBNs. They should be there! If they are not found, the LBN or RBN corresponding to the PBN is printed in brackets and counted.

- **VERIFY-W Cannot ONLINE unit** - message prints if the unit requested is available but the ONLINE command failed. The unit is released and the user is reprompted for another unit.

- **VERIFY-W Cannot Read Track with Starting xBN n** - prints if this access fails before the request is sent to the drive. It is usually caused by failing hardware. When VERIFY accesses LBN space or RBN space to check it, it reads all LBNs or RBNs on a track with one request. This operation is done with VERIFY processing all errors for each LBN or RBN.

- **VERIFY-W Copy n of xCT Block n (xBN n.) Does Not Compare** - prints whenever a block is found that does not compare to the first good one. All copies of every RCT or FCT block are read and compared to the first good copy read.

- **VERIFY-W Illegal Response to Start-up Question!** - prints if an invalid response is entered for a start-up question. The program reprompts with the same question.

- **VERIFY-W LBN n., a Non-Primary Revector, is Improper.** - prints if an LBN was not a nonprimary revector but was recorded in the RCT as such. When VERIFY reads an LBN with a header indicating it is a nonprimary revector, it looks it up in the collected RCT information and flags the fact it was found to be a nonprimary revector.

- **VERIFY-W LBN n., a Primary Revector, is Improper.** - prints if an LBN was not a primary revector but was recorded in the RCT as such. When VERIFY reads an LBN with a header indicating it is primarily revectored, it looks it up in the collected RCT information and flags the fact that it was found to be a primary revector.

- **VERIFY-W LBN n. Revectors to RBN n. Which is Bad.** - prints if VERIFY finds an RBN is good (can be read with error recovery) or only has a forced error (after error recovery). It looks it up in the collected RCT information. If found, VERIFY marks it as good. If, after the scan is finished, this flag is not set for an RBN revectored to, this message prints.

- **VERIFY-W Nonexistent Unit Number.** - prints if the unit number entered does not correspond to any known unit. The program reprompts for the unit number.

- **VERIFY-W Unit is Not Available.** - prints if the unit requested is unavailable. It may be in use by a host or another diagnostic. It may be inoperative. The program reprompts for another unit.

- **VERIFY-W xBN n. Has a Hard EDC Error.** - prints for LBNs and RBNs found to have a bad EDC (neither correct nor forced error). This error is classed as an inconsistency. Only a software error can result in a record with a bad EDC (unless the WRITE/BAD DKUTIL command is used).

- **VERIFY-W xBN n. is Bad but Not in the RCT.** - prints when VERIFY accesses a particular track for LBNs or RBNs only once. Any LBNs or RBNs where errors are detected in this initial pass are recorded. They are then read five more times, one LBN or RBN at a time. If errors are detected each time the LBN or RBN is accessed, and all of the errors are header errors, but the LBN or RBN is not recorded in the RCT, this error message is printed.

- **VERIFY-W xBN n. I/O Error in Access (MSCP Code: o).** - indicates a problem in the drive or the K. When this

message prints, it is an inconsistency. VERIFY provides its own error processing for records read where the K detects errors. This message prints if the return from the I/O operation is not SUCCESS (forced error, EDC error, or uncorrectable ECC error).

7.3.3.6 VERIFY Type Error Messages - A list of the type error messages produced by VERIFY follows. The t for type in these messages can stand for either I (Information) or W (Warning).

- **VERIFY-t LBN n. Has Corrupted Data (Forced Error).** - prints with t as a W if you answered Y to the prompt about FORMAT. However, if the unit has been subject to bad block replacement, this message is printed (if at all) with t as an I.

  Normally, all LBNs have a correct EDC indicating their data is good. However, a bad block replacement which occurs when the data could not be recovered produces a revectored LBN with a forced error flag. This indicates the data is probably bad. No such LBNs should exist just after FORMAT has run.

- **VERIFY-t RBN n. is Good but Not Used for a Revector.** - prints if a good RBN with a valid EDC is found in the verification pass but not recorded in the RCT as used. Unused RBNs on a disk are written with a forced error indication (the EDC is the complement of the proper EDC). No such records should exist just after FORMAT has been run. If you answered Y to the prompt about FORMAT, this message prints with t as a W. However, if the unit has been subject to bad block replacement, this message is printed (if at all) with t as an I.

- **VERIFY-t RBN n. Marked Bad in the RCT was Not Bad.** - prints with t as a W if you answered Y to the prompt about FORMAT. However, if the unit has been subject to bad block replacement, this message prints (if at all) with t as an I. When VERIFY reads a bad RBN (bad header or header code of bad), it looks it up in the collected RCT information and flags the fact it was indeed found to be bad. If any bad RBN recorded in the RCT is in fact all right, this flag is not set. No such RBNs should exist just after FORMAT has been run.

- **VERIFY-t xBN n. Has an Uncorrectable ECC Error.** - prints when VERIFY discovers an inconsistency. No LBN should have an uncorrectable ECC error; it should be revectored either by FORMAT or by bad block replacement. Thus, for an LBN, this error is considered an inconsistency. Also, FORMAT should have discovered all RBNs with uncorrectable ECC errors and marked them as

bad in the RCT. If an RBN is found with an uncorrectable ECC error, but that RBN is not in the RCT, it is also considered an inconsistency.

In both of these cases, this message is printed with t as a W. If an RBN is discovered with an uncorrectable ECC error marked bad in the RCT, this message prints (if at all) with t as an I.

7.3.3.7 **VERIFY Informational Messages** - Following are descriptions of the informational messages printed by VERIFY. Note, this type of message may or may not need informational messages enabled in order to print.

- **VERIFY-I Copy n of xCT Block n (xBN n.) is Bad.** - prints if informational messages are enabled for RCT or FCT blocks that cannot be read correctly with error recovery.

  NOTE

  Table is null or empty (no bad PBNs). This message is printed for null or empty FCTs whether or not informational messages are enabled.

- **VERIFY-I DBN Area Should Probably be Reformatted.** - prints whether or not informational messages are enabled. If more than nine DBNs were detected with positioner errors, this message prints after the DBN scan.

- **VERIFY-I INITIAL WRITE Should be Specified for ILEXER.** - prints whether or not informational messages are enabled. If more than nine DBns were detected with positoner errors, this message prints after the DBN scan.

- **VERIFY-I LBN n., a Primary, Has a Bad Header (is Non-Primary).** - prints if informational messages are enabled for LBNs recorded in the RCT as primary revectors but have garbled headers. Such a condition is abnormal but not erroneous.

- **VERIFY-I xBN n. Has a Transient (n Out of 6) x Error.** - prints if an LBN or RBN has been read six times with a least one error-free read when informational and transient error messages are enabled. The number of times out of six that errors were detected is indicated in the message.

- **VERIFY-I xBN n. Has a n Symbol Correctable ECC Error.** - prints for LBNs or RBNs with solid ECC errors (errors on all six accesses) that are correctable when informational messages are enabled. The highest number of symbols corrected on a seventh access is indicated in the message.

- **VERIFY-I xBN n. Has Solid Errors: x.** - prints for LBNs or RBNs with errors on all six accesses when informational messages are enabled. The errors included those other than ECC or EDC. The record is read a seventh time with error recovery to determine if the error is correctable. If it is not, a warning message prints.

7.4 OFFLINE DISK FORMATTER UTILITY (FORMAT)

FORMAT is the HSC70 utility used to format disks. It formats with either a 512- or 576-byte sector size. It can be used to format only the DBN area or to format both the LBN area and the DBN area.

CAUTION

The FORMAT utility destroys user data and can destroy the FCT if used by persons not familiar with DSA.

The DBN area is always formatted. If the user requests it, the LBN area is also formatted. When the LBN area is formatted, there are two modes of operation. In Best Guess mode, the XBN area is formatted, no FCT is used, and a null FCT is generated. In Reformat mode, the FCT on the disk is used and the XBN area is not formatted. If a Reformat is requested, but the FCT is null or clobbered, a modified Best Guess mode is used where only the LBN area is formatted. The main difference between Best Guess mode and Reformat mode is each track is reread at least three times during the check pass (Best Guess Mode) instead of once (Reformat Mode). If any error is detected, the track is reread 20 times instead of 3 times for Reformat mode.

CAUTION

Be careful when using CTRL C or CTRL Y to abort the FORMAT utility after formatting operations begin. Doing this may destroy the contents of the FCT and/or the RCT. The FORMAT utility should only be aborted under fatal-unrecoverable disk failure conditions.

7.4.1 FORMAT Initiation

FORMAT is initiated via the standard CRONIC command syntax, **RUN DX0:FORMAT.UTL.** Note the last field in the following prompts (shown in square brackets). This indicates the default for that prompt.

The program prompts for the unit number of the disk to format with the following:

FORMAT-Q Enter unit number to format (U) [D0]?

The next prompt determines whether the LBN (user data) area should formatted or only the DBN (diagnostic) area. If you answer Y to this prompt, user data is destroyed.

FORMAT-Q Format user data area (Y/N) [N]?

If you reply with N or a carriage return only (to obtain the default), the program starts executing and formatting the DBN area only. If you enter a Y, the program prompts for the sector size to use when formatting the disk:

FORMAT-Q Enter sector size to be used (512/576) [512]?

If you press carriage return only, the sector size used is 512 bytes. Otherwise, either 512 or 576 should be entered. The next prompt determines which format mode should be used:

FORMAT-Q Use existing bad block information (Y/N) [Y]?

If you enter an N, the program assumes a Best Guess mode format and is done depending on the response to the next question. If the user enters Y, a Reformat mode or modified Best Guess mode format is used, depending on the state of the FCT on the disk and the response to the next question. If N was entered in the previous question, the next question is:

FORMAT-Q FCT will be destroyed; are you sure (Y/N) [N]?

If you enter N, the program reprompts with the previous question. If you enter Y, the DBN, XBN, and LBN areas are formatted in Best Guess mode. This mode is seldom, if ever, used because the FCT contains important information about bad spots on the HDA which FORMAT does not always find using Best Guess mode. User data loss is highly probable when formatting in this mode.

If Y was entered for the previous prompt, the next question is:

FORMAT-Q Continue if bad block information is inaccessable (Y/N) [N]?

If you enter N, a Reformat mode is used if the FCT is valid. If it is not valid, the program aborts with an appropriate error message. If Y is entered, a Reformat mode is used if the FCT is

valid or a modified Best Guess mode is used if the FCT is null or clobbered. In either case, the XBN area is not formatted.

If the response to this prompt is Y or the response to the destroy FCT prompt is Y, the program prompts for a serial number:

FORMAT-Q Enter a non-zero serial number (D)?

This serial number is used if a Best Guess mode format is used or all copies of FCT block 0 are unreadable (in modified Best Guess mode). FORMAT allows a number of special options, not only for debugging purposes but also to increase data reliability. To determine if any of these options are desired, the program prompts with the following:

FORMAT-Q Do you want special options (Y/N) [N]?

If the response is N or a carriage return (the default of N), FORMAT starts processing.

If the response is Y, the following three special option prompts appear:

FORMAT-Q Revector blocks with 1 symbol ECC errors (Y/N) [N]?

Normally, blocks discovered during the check pass of formatting with 1-symbol ECC errors are not retired. The program assumes this level of error is tolerable. If the response to this prompt is Y, all blocks with solid (nontransient) ECC errors are retired. However, in all cases, blocks with 2-symbol (or more) ECC errors are always retired, regardless of the drive's ECC symbol threshold.

The second special option prompt is:

FORMAT-Q Revector blocks with transient errors (Y/N) [N]?

After a track is formatted, it is read either once (Reformat) or three times (Best Guess). If an error is detected, and the mode is Reformat, the track is read twice more. If any block not previously retired shows an error twice, it is retired, and the track is reformatted with this check pass done again. If no block had errors twice, the track is read 3 more times (Reformat) or 20 more times (Best Guess). Blocks which show an error only once during all of these reads are normally not retired. Such errors are considered tolerable transient errors. If the response to this prompt is Y, blocks that show any error are retired.

The third and final special option prompt is:

FORMAT-Q Report position of bad blocks (Y/N) [N]?

Blocks retired during the format process are reported with a single line printout. The type, block number, and cause are printed. If the response to this prompt is Y, the PBN number, cylinder, track, group, and position are also printed on a subsequent line.

The user can enter CTRL Z at any prompt to use the default for the remainder of the responses. Also, the responses to subsequent questions can be supplied at any question by typing the responses separated by commas. For example, if unit D133 has an FCT and is to be formatted in 512-byte mode with no special options, the user could type **D133,Y,,,,** at the first prompt.

7.4.2 FORMAT Sample Session

The following is a sample session using FORMAT. User input is in **bold print.**

```
HSC70> RUN DX0:FORMAT

FORMAT-Q Enter unit number to format (U) [D0]?D133
FORMAT-Q Format user data area (Y/N) [N]?Y
FORMAT-Q Enter sector size to be used (512/576) [512]?
FORMAT-Q Use existing bad block information (Y/N) [Y]?
FORMAT-Q Continue if bad block information is inaccessable (Y/N)
[N]?
FORMAT-Q Do you want special options (Y/N) [N]?Y
FORMAT-Q Revector blocks with 1 symbol ECC errors (Y/N) [N]?
FORMAT-Q Revector blocks with transient errors (Y/N) [N]?
FORMAT-Q Report position of bad blocks (Y/N) [N]?
FORMAT-S Format begun.
FORMAT-I    2 cylinders left in DBN space at 00:05:34.60.
FORMAT-I  275 cylinders left in LBN space at 00:05:39.60.
FORMAT-I Bad LBN 237213 (FCT), in the RCT area.
FORMAT-I  265 cylinders left in LBN space at 00:06:05.60.
FORMAT-I  255 cylinders left in LBN space at 00:06:31.40.
   .
   .
   .

FORMAT-I   25 cylinders left in LBN space at 00:16:36.20.
FORMAT-I   15 cylinders left in LBN space at 00:17:02.00.
FORMAT-I    5 cylinders left in LBN space at 00:07:28.40.
FORMAT-S Format completed.
FORMAT-I Stats:     0 Bad RBNs,
                    2 Revectored LBNs,
                    2 Primary Revectored LBNs,
                    0 Non-Primary Revectored LBNs,
                    1 Bad Blocks in RCT Area,
                    0 Bad Blocks in DBN Area,
                    0 Bad Blocks in XBN Area,
                    9 Blocks Retried on Check Pass.
FORMAT-I FCT was used successfully.
```

```
**************************************************************
*                                                            *
*   VERIFY must be RUN to complete FORMAT verification!      *
*                                                            *
**************************************************************
```

The preceding example is the output for an actual session for an RA80 disk with one bad PBN in the FCT. Notice the message which indicates it was retired because it was in the FCT, and it was in the RCT area. Note the informational message which is printed every 10 cylinders. This confirms that progress is actually being made and to show at what rate. Also, note the two LBNs which were retired because they had 2-symbol ECC errors; they became primary revectors. The error log messages were printed for them because, in the case of an RA80, two symbols are in excess of the ECC drive threshold.

Note, the final statistics indicate two LBNs were revectored and one bad LBN was found in the RCT area. The nine Blocks Retried on Check Pass include the two bad LBNs plus seven other blocks with transient errors only and therefore not retired. The bad block in the RCT was not retried in the check pass because it was known to be bad from the FCT. This would be true for any blocks retired due to their location in the FCT. The final message indicates an FCT was found and was successfully used.

Note, the message in the box which indicates VERIFY must be run to complete verification. This is an essential step and should not be skipped.

7.4.3 FORMAT Errors And Information Messages

This section describes the error and information messages printed by FORMAT. Error messages are arranged alphabetically according to the actual message.

7.4.3.1 FORMAT Error Message Variables - Variable output in the error and information messages is shown in **bold print.** These fields are formed as follows:

n = a decimal number
x = the way a block was found bad: FCT or check
xBN = a space: DBN, XBN, or LBN
hh = hours
mm = minutes
ss = seconds
xx = hundredths of a second

7.4.3.2 FORMAT Message Severity Levels - FORMAT error messages conform to the HSC utility error message format. In each case, the utility name at the start of the message is followed by a letter indicating severity level. These are defined as:

F = Fatal
I = Information
E = Error
S = Success
W = Warning

7.4.3.3 FORMAT Fatal Error Messages - This section describes the fatal error messages printed by FORMAT.

- **FORMAT-F Cannot Position to DBN Area!** - attempts to verify it has positioned the heads to the DBN area before it formats the disk unless FORMAT is running in Best Guess mode. FORMAT does this by reading the first sector of every track in the DBN read/write area until a sector is read without a header error. This fatal error message is printed if no such sector can be found.

- **FORMAT-F Current Maximum Sector Size is 512!** - prints if the user requests a 576-byte sector size but the system sector size is set to 512. In this case, FORMAT cannot run because I/O cannot be done with sectors that are 576 bytes long.

- **FORMAT-F DBN/XBN Format Error (Drive FORMAT Command Failed)!** - prints if a FORMAT command fails for five retries when formatting the DBN or XBN area.

- **FORMAT-F Drive Does Not Support 576 Mode on This Media!** - prints if the user requests a 576-byte sector size for a drive that does not support it.

- **FORMAT-F Drive is Write Protected!** - prints if the requested drive is hardware write-protected and therefore cannot be formatted.

- **FORMAT-F FCT Does Not Have Enough Good Copies of Each Block!** - prints if any block in the FCT does not have two good copies.

- **FORMAT-F FCT is Improper!** - prints if one or more PBNs are remaining to be processed. When the program finishes formatting the LBN area, it checks to see if all PBNs in the FCT have been processed. It usually indicates an FCT where some PBNs are out of order.

- **FORMAT-F FCT Nonexistent!** - prints if the FCT is null or clobbered, and the user has instructed the program not to continue.

- **FORMAT-F FCT Read Error!** - prints if all copies of some given block of the FCT cannot be successfully read.

- **FORMAT-F FCT Write Error!** - prints if all copies of some given block of the FCT cannot be successfully written.

- **FORMAT-F Formatter Initialization Error!** - prints if FORMAT cannot acquire enough data buffers or control blocks to start formatting, or if the disk functional code is not loaded.

- **FORMAT-F GET STATUS Failure!** - prints if the unit requested is not available or cannot be brought online.

- **FORMAT-F LBN Format Error (Drive FORMAT Command Failed)!** - prints if a FORMAT command fails for five retries when formatting the LBN area.

- **Nonexistent Unit Number!** - prints if the unit requested does not exist.

- **FORMAT-F RCT Does Not Have Enough Good Copies of Each Block!** - prints if any block in the RCT does not have two good copies.

- **FORMAT-F RCT is Full!** - prints if so many bad blocks are encountered the RCT overflows.

- **FORMAT-F RCT Read Error!** - prints if all copies of some given block of the RCT cannot be successfully read.

- **FORMAT-F RCT Write Error!** - prints if all copies of some given block of the RCT cannot be successfully written.

- **FORMAT-F SDI Receive Error!** - prints if a track cannot be read at all after it has been formatted.

- **FORMAT-F Too Many Bad RBNs Found Before RCT was formatted.** - prints if more RBNs than can be recorded in memory are encountered before the RCT area has been formatted.

- **FORMAT-F Unsuccessful SDI Command!** - prints if the drive fails to respond to an SDI command. FORMAT issues SEEK, RECALIBRATE, and DRIVE CLEAR SDI commands.

7.4.3.4 FORMAT Warning Message - The FORMAT utility prints only one warning message.

- **FORMAT-W WARNING: Possible Head Addressing Problem.** - prints if no sector was successfully read from one or more tracks in the XBN area. Note that all cylinders are checked. This is a simple check for a bad head.

7.4.3.5 FORMAT Information Messages - Following are the informational messages printed by FORMAT:

- **FORMAT-I Bad LBN n (x), a Non-Primary Revector.** - prints for LBNs retired by being revectored to some RBN other than the primary RBN; they are marked in the RCT as nonprimaries. They are formatted with a header code of nonprimary or with a headeer code of bad if their header area is bad.

- **FORMAT-I Bad LBN n (x), a Primary Revector to RBN n.** - prints for LBNs retired by being revectored to the first RBN on the same track; they are marked in the RCT as primaries. They are formatted with a header code of primary.

- **FORMAT-I Bad LBN n (x), in the RCT Area.** - prints for retired LBNs in the RCT area. They are formatted with a header code of bad.

- **FORMAT-I Bad RBN n (x).** - prints for retired RBNs. They are marked bad in the RCT and are formatted with a header code of bad.

- **Cylinder n, Group n, Track n, Position n, PBN n.** - prints following the preceeding four messages, if the user requested the special option to print bad block position.

- **FORMAT-I CTRL/Y or CTRL/C Abort!** - is an informational message and prints if the user aborts FORMAT by typing a CTRL Y or CTRL C. Note, this probably leaves the disk in an unusable state if the format has begun.

- **FORMAT-I FCT was Not Used.** - prints if a null or clobbered FCT was found on the disk or generated at the request of the user (Best Guess mode).

- **FORMAT-I FCT was Used Successfully.** - prints if a valid FCT was found on the disk and used.

- **FORMAT-I n Cylinders Left in xBN Space at hh:mm:ss.xx.** - is an informational message and prints after every 10 cylinders are formatted in order to record the progress of the FORMAT program.

- **FORMAT-I Only DBN Area Formatted (n Bad DBNs).** - prints if the user requested formatting of the DBN area only. It prints after the format of the DBN area is completed. After this message prints, the program terminates.

7.4.3.6 **FORMAT Error Messages** - Following are the error messages printed by FORMAT:

- **FORMAT-E Illegal Response to Start-up Question!** - prints if an invalid input is supplied for a start-up question. The program reprompts with the same question.

- **FORMAT-E Nondefaultable Parameter.** - prints if the user enters only a carriage return, requesting the default for the only nondefaultable parameter (the serial number). The program reprompts for the serial number.

7.4.3.7 **FORMAT Success Messages** - Following are the FORMAT success messages:

- **FORMAT-S Format Completed.** - prints after the format process is done, and all verification tests are complete.

- **FORMAT-S Format Begun.** - prints when FORMAT actually begins formatting the disk.

7.5 RXFORMAT UTILITY

The RXFMT utility program allows the user to format and verify RX33 diskettes. These are 5 1/4-inch, two-sided, double-density diskettes available from DIGITAL. This utility is used only to format diskettes for the HSC70. The program should complete in less than five minutes.

7.5.1 RXFORMAT Initiation

To run the RXFMT utility, select an HSC70. At the KMON prompt, type:

HSC>RUN dev:RXFMT

where dev is the name of the drive containing the RXFORMAT utility.

The program prompts the user from beginning to end. As with all HSC prompts, material contained in square brackets is the default. To accept the default, press the RETURN key. With square brackets that do not contain material, you have to supply the value and then press the RETURN key.

To abort the utility, type CTRL/Y or CTRL/C at any point. However, note this action leaves your diskette in an unknown state.

After the RUN command is input, the utility prompts:

RXFMT-Q Unit to format []?

RXFMT allows the user to select either drive to run the program.

Following is an example of a typical RXFMT session:

```
HSC70>RUN DXl:RXFMT

RXFMT-Q Unit to format []?DXl:
RXFMT-Q Ready to start formatting (Y or N) []?Y

RXFMT-I Formatting track 0, side 0, LBN 0
RXFMT-I Formatting track 8, side 0, LBN 240
RXFMT-I Formatting track 16,side 0, LBN 480
RXFMT-I Formatting track 24,side 0, LBN 720
RXFMT-I Formatting track 32,side 0, LBN 960
RXFMT-I Formatting track 40,side 0, LBN 1200
RXFMT-I Formatting track 48,side 0, LBN 1440
RXFMT-I Formatting track 56,side 0, LBN 1680
RXFMT-I Formatting track 64,side 0, LBN 1920
RXFMT-I Formatting track 72,side 0, LBN 2160
RXFMT-S Format successfully completed.
RXRD-I Reading track 0, side 0, LBN 0
RXRD-I Reading track 8, side 0, LBN 240
RXRD-I Reading track 16,side 0, LBN 480
RXRD-I Reading track 24,side 0, LBN 720
RXRD-I Reading track 32,side 0, LBN 960
RXRD-I Reading track 40,side 0, LBN 1200
RXRD-I Reading track 48,side 0, LBN 1440
RXRD-I Reading track 56,side 0, LBN 1680
RXRD-I Reading track 64,side 0, LBN 1920
RXRD-I Reading track 72,side 0, LBN 2160
RXFMT-I Program Exit
```

7.5.2 RXFORMAT Error Messages

Error messages possible while running RXFORMAT follow:

- **RXFMT-E Requested unit is unavailable.** - The unit specified in the command line is unavailable.

- **RXFMT-F Aborting.** - RXFMT tries to format and verify 10 times. If no progress is made, RXFMT issues an error message and the program exits. This message is also displayed after the user types either a CTRL/Y or a CTRL/C. Try a different diskette. If the problem persists, report it to appropriate personnel.

- **RXFMT-F Error comparing track.** - RXFMT detected an inconsistency. The data read from the diskette in the verify pass did not match what was written. Retry.

- **RXFMT-F Error formatting track.** - This could be caused by a bad diskette or a hardware problem. Retry. If the problem still persists, try a different diskette.

- **RXFMT-F Error reading track.** - This error could be caused by a bad diskette or a hardware problem. RXFMT tries to verify the formatting 10 times. If no progress is made, the program exits. Run program again. If problem persists, use a different diskette. If the problem still persists, report it to Field Service Support.

- **RXFMT-F Unable to allocate sufficient mapped memory.** - Not enough blocks in Program memory are available to use as buffer space. Try again later.

- **RXFMT-F Unable to allocate sufficient XFRBs.** - The common pool did not contain enough memory to allocate an XFRB, required for RXFMT using load media. This is a transient condition. Try again later.

- **RXFMT-W About to format diskette in boot device.** - RXFMT warns the user the utility is about to format the diskette in the boot device. The user must be very cautious when running RXFMT. As a result, RXFMT not only asks whether reformatting should start, but also outputs this warning message.

- **RXFMT-I Formatting track, side, LBN.** - RXFMT did not encounter any problem while formatting previous track, and simply reports.

- **RXFMT-I Please specify a valid unit.** - The user must specify the unit-id, either DX0: or DX1:.

- **RXFMT-I Program Exit.** - The program is finished and is exiting.

- **RXFMT-I Reading Track, side, LBN.** - RXFMT did not encounter any problems while verifying previous track.

- **RXFMT-S Format successfully completed.** - RXFMT completed without any errors or interruptions.

- **RXFMT-Q Unit to format []?** - RXFMT asks which unit the user will use to format the diskette.

- **RXFMT-Q Ready to start formatting (Y or N) []?** - RXFMT asks if the user is ready to format the diskette. Ensure the diskette is loaded into the correct drive.

7.6 VIDEO TERMINAL DISPLAY (VTDPY)

VTDPY is a utility for gathering system statistics. This utility displays, on a continuing basis, activity within the HSC. VTDPY can display system throughput, AVAILABLE or ONLINE status of disk and tape drives, and utilities running on other terminals. This utility also indicates which nodes have virtual circuits, connections, and multiple connections to the HSC.

NOTE

Do not run VTDPY using the command SET HOST/HSC through the Diagnostic Utility Program (DUP). DUP cannot manage VTDPY because too much optional interrupt data is produced.

This utility requires a video terminal and does not display on an LA12. Either a VT100 or a VT220, set at 9600 baud, must be attached to the EIA port on the HSC to run VTDPY.

To run VTDPY, enter at the prompt

HSC> **RUN dev:VTDPY (update-interval)**

In this command, **update-interval** is in seconds, anywhere from 2 to 420. If you do not provide this update interval, VTDPY prompts:

VTDPY-Q Interval (secs) ?

If your response is outside the allowable range, VTDPY displays an error message. The higher the number for the update-interval, the smaller the performance impact on the HSC.

VTDPY terminates after the user enters a CTRL/Y or a CTRL/C. The screen is cleared upon termination.

7.6.1 VTDPY Error Messages

This utility has only two error messages, as follows:

Message: VTDPY-E Illegal Interval Value (2 to 420 seconds)

Explanation: The user has entered an update-interval ouside the range permitted. VTDPY reprompts for the update-interval.

User Action: Re-enter a value within the correct range.

Message: VTDPY-F Insufficient Common Pool

Explanation: This message indicates insufficient memory to run VTDPY.

User Action: Try again later.

7.6.2 VTDPY Display Example

An example of a VTDPY screen display follows:

```
HSC70 V3.00 C3P0  Id 0000000000DD On 14-Apr-1986 12:28:13.12   UP: 113.49
 42.9% Idle     39 Work Requests/Sec      40 Sectors/Sec       0 Records/Sec

    Free Lists              Process Pr St  Time%        Disk Status
  Ctrl Blks    2269 +          Kernel       16.4%                1111111111
  SLCB/DCB       32 +        4 VTDPY  11 Rn 19.2%     +1234567890123456789
  Buffers       889 +       24 SYSDEV  1 Bl          0....................
                            50 DEMON  11 Bl         20A.A..........A......
    Pool Sizes              52 PDEMON  7 Bl         40..........A.A.A.....
  SYSCOM       1800 +       54 PSCHED 13 Rn 42.9%   60.AA.......0..A..0...
  Kernel       6504 +       72 DISK    9 Rn 16.0%   80....................
  Program    821120 +      110 ECC     6 Bl         100...................
  Control     32436 +      120 TAPE    8 Bl         120...................
                           122 TTRASH  7 Bl         140...................
Data B/W used:    .0%      124 HOST    4 Bl    .9%  160...................
                           126 POLLER  5 Bl         180.................A.
    Host Status            130 SCSDIR  5 Bl    .9%  200A..................
  0123456789012345         146 DPOUT  10 Bl         220...................
  ..MMMMM..MMMM...         150 DP2OUT 10 Bl         240...................
  ..^^^BA..^^BA...         152 DUP     9 Bl
```

NOTE

A true video display contains solid diamond symbols on the bottom line, indicated in this example as a caret (^).

7.6.2.1 VTDPY Display Explanation - The previous display example constantly changes as different processes run in the HSC. These changes are made automatically with the exception of the fields relating to HSC memory. Memory statistics are updated only by typing a CTRL/W. The major fields are explained as follows:

```
HSC70 V3.00 C3P0  Id 0000000000DD On 14-Apr-1986 12:28:13.12   UP:
113.49
```

The top line, reading from left to right, shows the HSC model number (HSC70), the baselevel of the operating software (V3.00), the system name (C3P0), the system ID (Id) any hexadecimal number unique to the cluster, in this case 0000DD), time, and date. The last number on the right indicates the hours and minutes the HSC has been running since the last boot or reboot.

```
42.9% Idle   39 Work Requests/Sec   40 Sectors/Sec   0 Records/Sec
```

This second line in the display shows the percentage of current P.io idle time, average number of work requests (i.e., MSCP and TMSCP) per second, number of disk data sectors transferred per second, and number of tape data records transferred per second. These numbers are normalized to match the update interval.

```
    Free Lists
  Ctrl Blks    2269 +
  SLCB/DCB       32 +
  Buffers       889 +

    Pool Sizes
  SYSCOM       1800 +
  Kernel       6504 +
  Program    821120 +
  Control     32436 +
```

This field represents the quantity of available memory and memory structures. The sizes are usually followed by plus signs. If followed by minus signs, the system is in memory deficit. Extremely prolonged memory deficit results in HSC slowdown and could eventually result in an HSC crash.

Data B/W used: .0%

This display shows the percentage of data bandwidth used. This is an instantaneous display and may often show 0% when the HSC is busy because the sampling interval missed the instantaneous bandwidth usage.

```
   Host Status
0123456789012345
..MMMMM..MMMM...
..^^^BA..^^BA...
```

NOTE

A true video display contains solid diamond symbols on the bottom line, indicated in this example as a caret (^).

This field indicates host status. The line below Host Status shows the node number (in the range 0 through 15) of the hosts in the cluster. If no letter appears under this node number in the next line, that node number is not a currently active host. If a **V** appears on that line, a Virtual Circuit only is open and no connection is present (host usually in the transitional state). A **C** on this line indicates one connection to that host and an **M** indicates multiple connections. Because each host can make a separate connection to each of the Disk, Tape, and DUP servers, this field frequently shows multiple connections.

The bottom line of this field contains CI path status information and each position can contain either a diamond symbol, an A, or a B. The meanings are as follows:

- A diamond symbol equals normal operation in any position with a connection.
- An **A** or B indicates only one CI path is operational. If an A is displayed, Path A is running, but Path B is not; if a B is displayed, Path B is running, but Path A is not. Either letter probably indicates a hardware problem.

```
   Process Pr St  Time%
     Kernel       16.4%
  4 VTDPY  11 Rn  19.2%
 24 SYSDEV  1 Bl
 50 DEMON  11 Bl
 52 PDEMON  7 Bl
 54 PSCHED 13 Rn  42.9%
 72 DISK    9 Rn  16.0%
110 ECC     6 Bl
120 TAPE    8 Bl
122 TTRASH  7 Bl
124 HOST    4 Bl    .9%
126 POLLER  5 Bl
130 SCSDIR  5 Bl    .9%
146 DPOUT  10 Bl
150 DP2OUT 10 Bl
152 DUP     9 Bl
```

The headings in this display (from left to right) mean the following:

- The first column with numbers is the process number.
- The **Process** column shows the name of the process running at the time.
- The **Pr** column shows the priority of the process.
- The **St** column shows the status of the process, either running **Rn** or blocked **Bl**.
- The **Time%** column is the percentage of P.io time each currently-running process is using.

Certain process names in the first column under **Kernel** (the operating system) are defined as follows:

- In this case, VTDPY. However, it could be another utility (in which case the priority number would change also).
- SYSDEV is the load device driver.
- DEMON indicates demand and automatic diagnostics are running.
- PDEMON indicates periodic diagnostics are running.
- PSCHED is the scheduler for periodic diagnostics. This is the HSC idle loop.

- DISK is the disk server.
- ECC is the error correction code process and is always displayed when disk I/O is indicated.
- TAPE is the tape server.
- TTRASH is always displayed when the tape server is active. It is the process that sends tape error logs to the host.
- HOST is the process that interfaces to the host. It is always present.
- POLLER polls for the host process and is always present when a connection is present.
- SCSDIR processes directory requests from the host.
- DPOUT and DP2OUT are the I/O from two different DUP processes.
- DUP is the Diagnostic and Utility Protocol server.

Note, not all processes are necessarily shown. Because of limited space on the screen, the display of some processes may be truncated and the CPU time percentages may not total 100 percent.

```
        Disk Status
             1111111111
   +1234567890123456789
  0.....................
 20A.A..........A......
 40..........A.A.A.....
 60.AA.......O..A..O...
 80.....................
100.....................
120.....................
140.....................
160.....................
180...................A.
200A....................
220.....................
240.....................
```

The last area in the display can indicate either Disk Status or Tape Status. This rightmost field fluctuates between the two displays whenever both device types are connected to the HSC.

The line immediately under Disk Status indicates the following unit numbers are augmented by 10 from the base number in the leftmost column. To find the identification number of the disk indicated by a single letter in this field, count from the left.

For instance, on the 20s line, the third A would be disk unit number 33.

A letter anywhere in the field has a particular meaning for the particular disk unit identified, as follows:

- An **A** indicates the drive is available but not mounted.
- An **O** indicates Online status. The drive is in use by a host, an HSC utility, or an HSC diagnostic.
- **A D** indicates the HSC is connected to duplicate units (two or more drives with the same unit number).
- **A U** indicates the drive went into an undefined state.

The letters and method of determining tape drive ID number are the same when tape status is displayed. However, one additional letter can be shown, an **F**, indicating no tape is mounted on the tape drive.

CHAPTER 8
TROUBLESHOOTING TECHNIQUES

8.1 INTRODUCTION

This chapter describes the types of errors occurring during HSC70 boot and operation. The major divisions are initialization errors and system type errors. Initialization errors occur while the HSC70 is trying to boot. System type errors occur while the HSC70 is running functional code. System type errors may be reported to a host node and possibly the HSC console device. Some system errors may result in the HSC70 crashing and rebooting. System errors include MSCP, TMSCP, BBR, and out-of-band errors.

8.2 HOW TO USE THIS CHAPTER

Initialization error indications are found in the Operator Control Panel (OCP) fault codes and the module LEDS. In addition, the bootstrap diagnostics may produce error messages printed out to the console. Read Section 8.3 for an understanding of initialization errors that do not produce a message. All errors displayed as English messages are shown in the index to this manual. Section 8.3 divides initialization errors into three types:

- o OCP fault codes
- o Module LEDS
- o Boot diagnostic messages

HSC console error messages for system type errors are described in this chapter and are organized into the following sections:

- o MSCP/TMSCP errors--Section 8.4.1
  - Controller errors--Section 8.4.2.4
  - MSCP SDI errors--Section 8.4.2.5
  - Disk Transfer errors--Section 8.4.2.6

- o BBR errors--Section 8.4.3
- o TMSCP errors--Section 8.4.4
  - STI Communication or Command Errors -- Section 8.4.4.1
  - STI Formatter Error Log -- Section 8.4.4.2
  - STI Drive Error Log -- Section 8.4.4.3
- o Out-of-Band errors -- Section 8.4.5
  - HOST-X -- CI errors -- Section 8.4.5.1
  - SYSDEV-X -- Load Device errors -- Section 8.4.5.2
  - DISK-X -- Disk Functional errors -- Section 8.4.5.3
  - TAPE-X -- Tape Functional errors -- Section 8.4.5.4
  - SINI-X -- Miscellaneous errors -- Section 8.4.5.5

Each message description includes the following:

- o Actual error message
- o Error message severity level
- o Message description
- o Field service action
- o Possible FRUs

8.3 INITIALIZATION ERROR INDICATIONS

Initialization errors are indicated by:

- o OCP fault code displays
- o Module LEDs
- o Boot diagnostic messages

8.3.1 OCP Fault Code Displays

OCP fault codes are divided into two categories, hard fault codes and soft fault codes. Soft fault codes are also called nonfatal fault codes. Soft faults impede HSC70 operation, but the fault does not hinder the boot process. Hard fault codes are fatal to the HSC and prevent further operation of the HSC subsystem until the condition is remedied.

Figure 8-1 shows the possible displays available on the OCP in the event of errors during initialization or operation. For detailed interpretations of these fault codes, refer to Chapter 1.

DESCRIPTION	HEX	OCT	BINARY	OCP INDICATORS				
				INIT	FAULT	ONLINE		
PORT PROCESSOR MODULE FAILURE†	01	01	0 0001					■
DISK DATA CHANNEL MODULE FAILURE†	02	02	0 0010				■	
TAPE DATA CHANNEL MODULE FAILURE†	03	03	0 0011				■	■
INSTRUCTION CACHE PROBLEM IN I/O CONTROL PROCESSOR*	08	10	0 1000		■			
HOST INTERFACE ERROR*	09	11	0 1001		■			■
DATA CHANNEL ERROR*	0A	12	0 1010		■		■	
I/O CONTROL PROCESSOR MODULE FAILURE	11	21	1 0001	■				■
MEMORY MODULE FAILURE	12	22	1 0010	■			■	
BOOT DEVICE FAILURE**	13	23	1 0011	■			■	■
PORT LINK MODULE FAILURE	15	25	1 0101	■		■		■
MISSING FILES REQUIRED	16	26	1 0110	■		■	■	
NO WORKING K.SDI, K.STI, OR K.CI	18	30	1 1000	■	■			
REBOOT DURING BOOT	19	31	1 1001	■	■			■
SOFTWARE DETECTED INCONSISTENCY	1A	32	1 1010	■	■		■	

† INCORRECT VERSION OF MICROCODE.

* THESE ARE THE SO-CALLED SOFT OR NON-FATAL ERRORS.

**POSSIBLE MEMORY MODULE/CONTROLLER ON HSC70

CX-905B

Figure 8-1 Operator Control Panel Fault Codes

8.3.2 Module LEDs

HSC70 modules contain LEDs used as state indicators for each module. These LEDs are described in the following tables. Figure 2-1 in Chapter 2 shows the locations of the module LEDs.

8.3.2.1 P.ioj LEDs - Table 8-1 shows the L0111 (I/O Control Processor module) LEDs and their functions.

Table 8-1 L0111-0 (P.ioj) LEDs

Led	Color	Meaning
D1	Yellow	Micro-ODT ON when J-11 executing micro-ODT
D2	Yellow	SLU OK Serial Line Unit output of UART
D3	Yellow	MEM OK Turned OFF as J-11 successfully accesses Program memory
D4	Yellow	SEQ Lamp Turned OFF as J-11 verifies proper functioning of its sequencers for control store
D5	Yellow	State Lamp Blinks in parallel with OCP State Lamp (under software control)
D6	Yellow	Fetch Lamp Blinks once for every PDP-11 instruction fetched (J-11 run LED)
D7	Red	Diagnostic/Testing Failure Initially on for powerup; turned off upon successful completion of J-11 initialization diagnostics
D8	Green	Diagnostics Passed Turned on upon successful completion of J-11 initialization diagnostics

8.3.2.2 Power-Up Sequence Of I/O Control Processor LEDs - This section defines the power-up sequence of the LEDs shown in Table 8-1. First, LED numbers D8 and D7 are used to indicate whether the P.ioj module has successfully completed all of its initialization diagnostics. The module powers up with the red (D7) LED ON and the green (D8) LED OFF. D1 through D4 (yellow)

are initially on. As soon as the J-11 starts operating, D1 (micro-ODT LED) turns off.

Several microcode steps later, D4 (sequence LED) is turned off indicating the J-11 is sequencing and succeeded in reaching this point in its microcode. The J-11 performs several program memory operations and, if successful, turns off D3, (memory OK LED). Finally, the J-11 accesses the console terminal port of the UART (universal asynchronous receiver/transmitter) and turns off D2 (SLU or serial line unit LED).

Upon successful completion of the boot time initialization diagnostics, D8 (module OK LED) turns on, and D7 (module failure LED) turns OFF. The J-11 then proceeds to the software initialization programs.

In addition to being initially ON, the D1 (micro-ODT run LED) is on any time the J-11 is executing micro-ODT. D6 (the fetch LED, sometimes referred to as the run LED) blinks once for every PDP-11 instruction fetch cycle. When the J-11 is running, D6 is illuminated at half-brilliance compared to the other yellow LEDs.

8.3.2.3 Memory Module LEDs - Table 8-2 shows the L0117-0 (M.std2) module LEDs and their functions. These LEDs are controlled by a bit in the RX33 FDC MAR02 register. The green LED is set to ON by the P.ioj boot/ROM self-test diagnostics after the RX33 has passed its self-tests, and Program memory has found 8 Kwords to load INIPIO/OFLPIO.

NOTE

The entire LED package on the M.std2 is called D2. All three LEDs are contained in the D2 package.

Table 8-2 L0117-0 (M.std2) LEDs

LED	COLOR	MEANING
D2	Red	Mod Not OK
D2	Green	Mod OK
D2	Yellow	Memory Active Indicates access activity to any of the three memories on this module

8.3.2.4 Data Channel LEDs - Table 8-3 shows the L0108-YA and -YB (K.sdi and K.sti) module LEDs and their functions with the system software.

Table 8-3 L0108-YA/YB (K.sdi/K.sti) LEDs

Led	Color	Meaning
D3	Red	Module Failure Indicates a module microdiagnostic failed to successfully complete, or this module is still under initialization by the subsystem.
D4	Green	Module OK Turned on by the Init/Func Flag signal in the K functional microcode. The green LED comes on after successful initialization or while the data channel is running functional microcode.

8.3.2.5 Host Interface LEDs - Table 8-4 shows the three modules in the K.ci set, their LEDs, and the functions of the LEDs with the system software.

Table 8-4 K.ci (LINK, PILA, K.pli) LEDs

Module	LED	Color	Meaning
K.pli	D2	Red	ON when P.io has booted or rebooted, but K.pli module has not yet passed its self-test.
K.pli	D1	Green	ON when K.pli has passed its self-test.
PILA	D2	Red	ON when PILA module has not yet passed the test performed by the K.pli.
PILA	D1	Green	ON when the PILA module has passed the test performed by the K.pli. LED is controlled by the port processor.

Module	LED	Color	Meaning
PILA	D3	Yellow	(Not found on all etch rev modules.) ON when K.pli is asserting init. When init is true, both the red and the green PILA LEDs are forced OFF.
LINK	D998	Green	ON when local activity is present on the LINK module (whenever the LINK module detects a message directed to its node or when it detects an outgoing message).
LINK	D999	Red	ON during the CI maintenance loop test.

8.3.3 Communication Errors

It is possible for the HSC70 to complete its initialization and not report the fact on the local console terminal (VT220). This is an indication of a failure in the serial communication path between the UART chip on the P.ioj (L0111-0) and the local console terminal.

As a method of testing this serial path, the HSC70 echoes the characters typed on the local console terminal as if the terminal were in local mode. Use the following procedure to test the serial path:

1. Place the Secure/Enable switch in the ENABLE position
2. With power on, push in and hold the OCP Init switch.
3. Type a series of characters on the terminal keyboard.
4. Check to see if the series of characters echoed correctly on the terminal.

NOTE

When the Init switch is released, the HSC reboots.

If this procedure fails to echo characters typed at the keyboard, the failure is either a terminal/P.ioj baud-rate mismatch (default is 9600), a P.ioj module failure, or a problem within the terminal-cabling subsystem. Ensure the terminal set-up parameters are correct. Refer to the *HSC70 Installation Manual* (EK-HSC70-IN-001) for the proper terminal configuration. Refer to the *VT220 Owner's Manual* (EK-VT220-UG-001) for problem-solving techniques related to the VT220.

8.3.4 Requestor Status For Nonfailing Requestors

When a requestor successfully completes all internal microdiagnostics, bits 0 through 5 contain the following codes defining module types.

- Code 001 represents a properly-functioning host interface module set (K.ci).
- Code 002 represents a properly-functioning disk data channel module (K.sdi).
- Code 203 represents a properly-functioning tape data channel module (K.sti).
- Code 377 indicates the requestor slot does not contain a module.

NOTE

When a module fails internal microdiagnostics or its functional code, the status byte reflects the failure. See Appendix D for a complete list of K.ci-, K.sdi-, and K.sti-detected failures.

8.3.5 Boot Flowchart

The HSC70 Boot Flowchart (Figure 8-2) maps the entire boot sequence. The flowchart calls out useful visual milestones that aid in troubleshooting the problems which can occur during initialization.

The flowchart has three main divisions:

1. Information on activity common to both the system and offline diskettes is contained in boxes A through O.
2. Information on activity specific to the system diskette is contained in boxes SA through SJ.
3. Information on activity specific to the Offline diskette is contained in boxes OA through OG.

The flowchart begins when one of the following occurs:

- Init button pushed
- Powerup has started
- Other software caused reboot

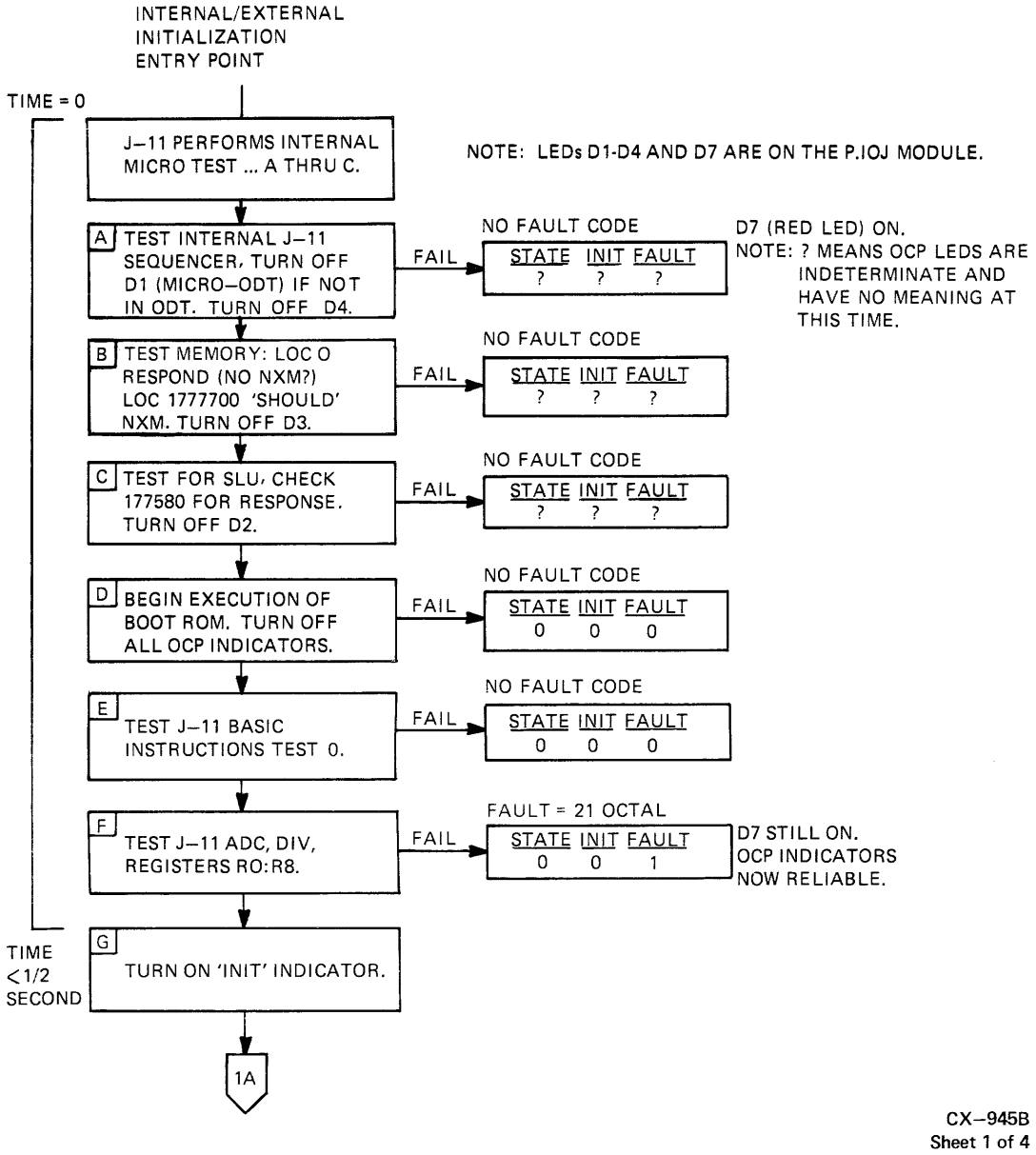


Figure 8-2 HSC70 Boot Flowchart (1 of 4)

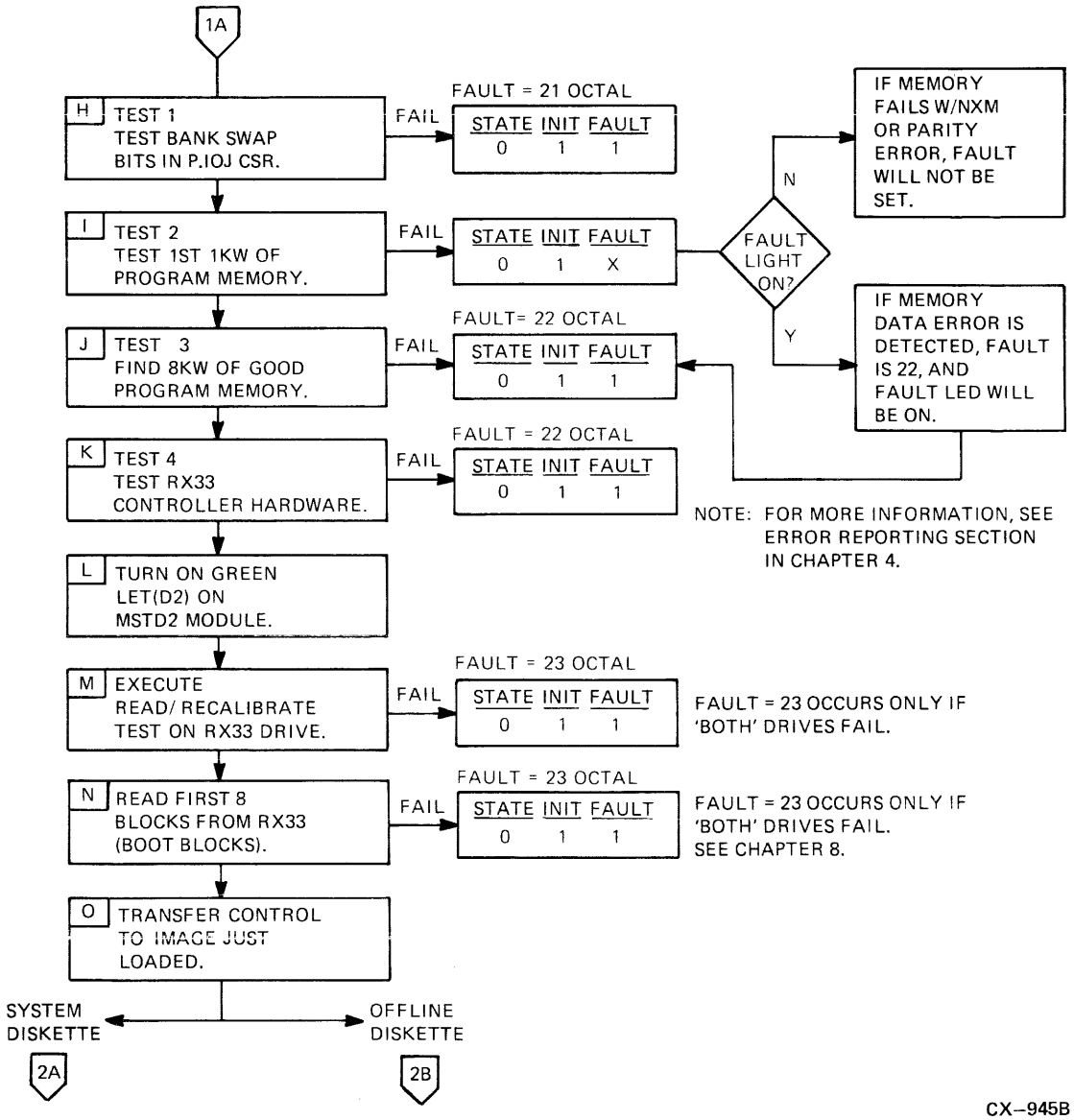


Figure 8-3 HSC70 Boot Flowchart (2 of 4)

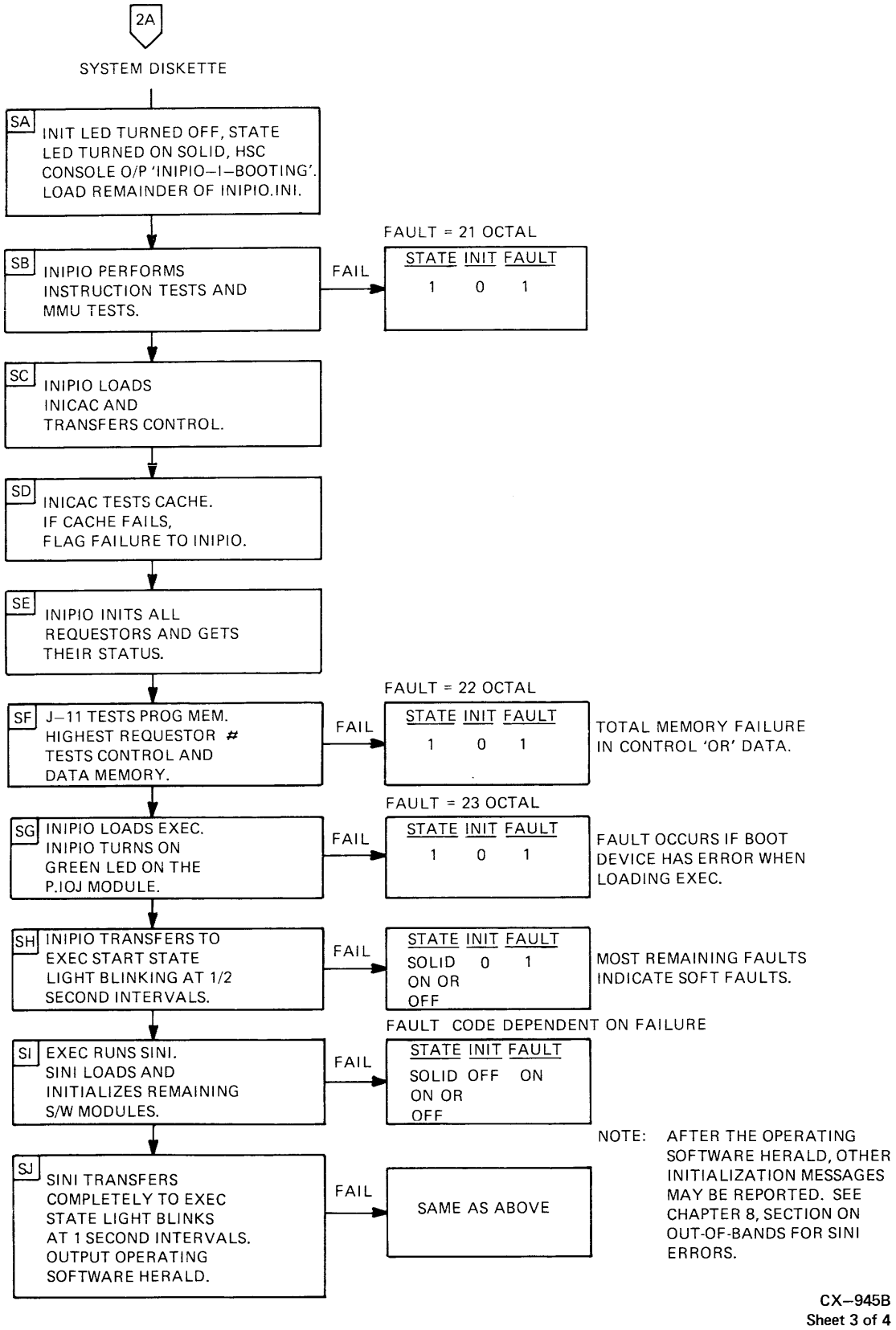
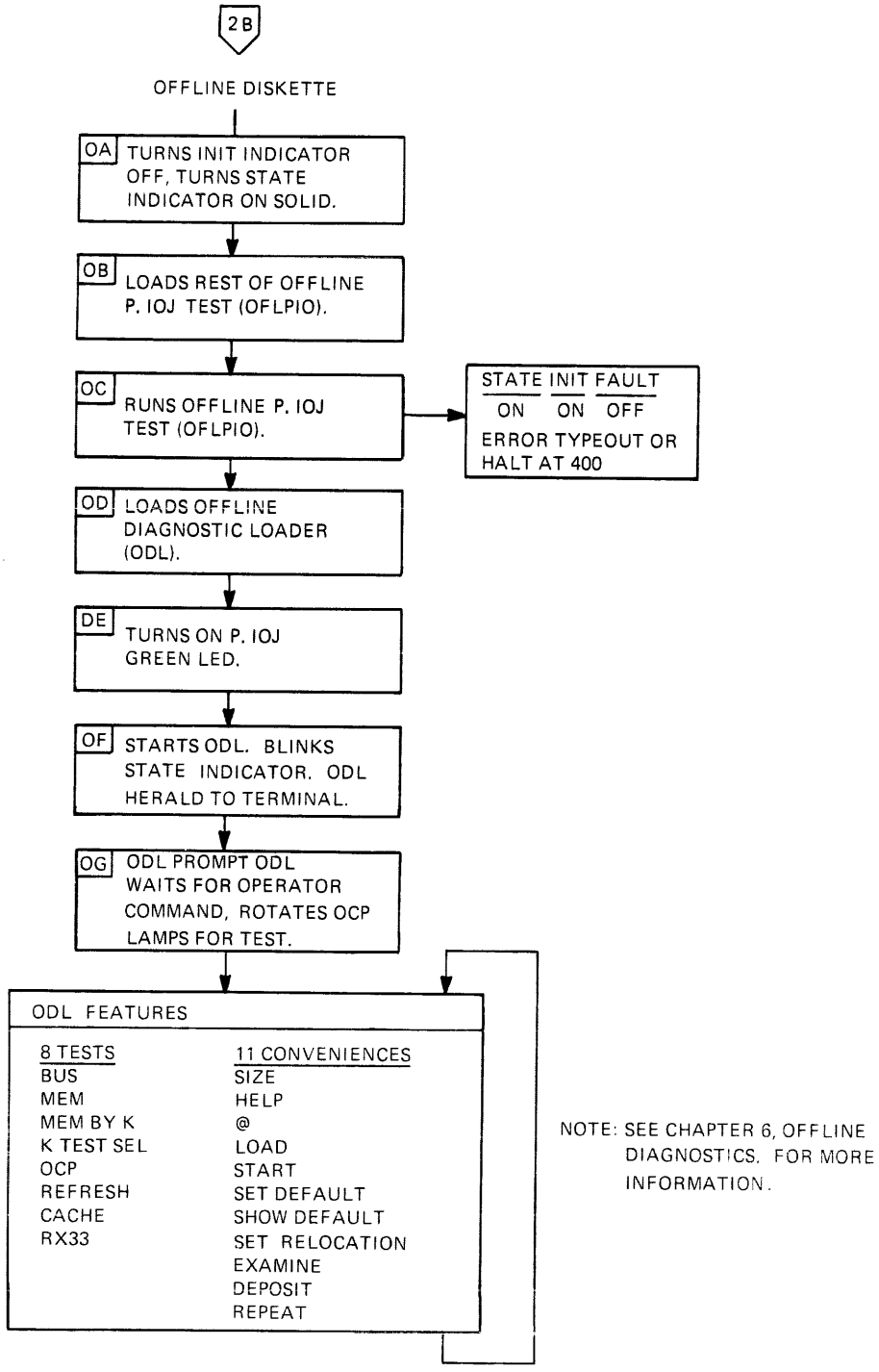


Figure 8-4 HSC70 Boot Flowchart (3 of 4)



NOTE 1: FIRST PORTION OF THE OFFLINE P.IOJ TEST (OFLPIO) WAS LOADED WITH THE PREVIOUS LOAD OF EIGHT BOOT BLOCKS.

NOTE 2: FOR DETAILED INFORMATION ON THE OFFLINE P.IOJ TEST AND ERROR REPORTS, REFER TO CHAPTER 6, OFFLINE DIAGNOSTICS.

CX–945B
Sheet 4 of 4

Figure 8-5 HSC70 Boot Flowchart (4 of 4)

8.3.6 Boot Diagnostic Indications

The HSC70 can pass boot diagnostics with a failing requestor. Although the HSC70 passed the boot, the failure associated with the requestor is considered an initialization error.

Following is an example of an error message displayed when a requestor fails on initialization of the operating software. The HSC70 has passed most of the initialization/boot diagnostics, but a requestor has failed.

```
SINI-E ERROR SEQUENCE 2. AT 20-SEPT-1985 00:00:02.80
    REQUESTOR 2 FAILED INIT DIAGS, STATUS = 107
```

The requestor with the red LED on is the failing requestor. In this case, the diagnostic identifies requestor 2 as failing its internal self-test number 7. Additionally, the Fault indicator turns on, and a soft fault code of octal 12 is displayed on the OCP after the Fault switch is pressed.

See Chapter 4 for more information on errors indicated by the OCP.

8.4 SOFTWARE ERROR MESSAGES

Software error messages are classified into three categories:

1. MSCP/TMSCP errors
2. Bad Block Replacement errors (BBR)
3. Out-of-Band errors

8.4.1 Mass Storage Control Protocol Errors

The Mass Storage Control Protocol (MSCP/TMSCP) errors printed out at the console terminal and reported to a host can be one of the following types:

- Controller Errors
- SDI Errors
- Disk Transfer Errors
- STI Communication Errors
- STI Formatter Errors
- STI Drive Errors

8.4.2 MSCP/TMSCP Error Format, Description, And Flags

Error formats, descriptions of the fields within the error format, and error flags are nearly identical for MSCP and TMSCP errors. Differences are noted where they exist.

8.4.2.1 MSCP/TMSCP Error Format - Example 8-1 shows an error format generic to all MSCP/TMSCP errors. Some errors may contain optional lines with additional information.

Example 8-1 MSCP/TMSCP Error Message Format

```
ERROR-X Text of message at (date) (time)
        Command Ref #   xxxxxxxx
        Err Seq #       x.
        Error Flags     xx
        Event           xxxx
        (Optional line)
        (Optional line)
        (Optional line)
ERROR-I End of error.
```

8.4.2.2 MSCP/TMSCP Error Message Fields - Table 8-5 describes the various fields found in an MSCP/TMSCP error message. These are common fields to all error messages of this type.

Table 8-5 MSCP/TMSCP Error Message Field Description

Field	Description
ERROR-E	The E is a code indicating the severity level of an error. Other codes are: Q for inquiry, I for informational, F for fatal, W for warning, and S for success. Note: Only severity levels E and Q require user action. Information following the severity level code is a textual version of the error message describing the event code, followed by the the date and time.
Command Ref #	This number (in hexadecimal) is the MSCP/TMSCP command number which caused the reported error. It is zero if the error does not correspond to a specific outstanding command. This number is normally assigned by the issuing host CPU.
Err Seq #	This number (in decimal) is a sequential number which counts error log messages since the MSCP/TMSCP server established a connection with the host. It is zero if the MSCP/TMSCP server does not implement error log sequence numbers.
Format Type	This field is found only in TMSCP error messages. This number, in bit format, is the formatted density in bits per linear inch of tape.

Field	Description
Error Flags	This number (in hexadecimal) indicates bit flags, collectively called error log message flags, used to report various attributes of the error. Refer to Table 8-6.
Event	This number (in hexadecimal) identifies the specific error or event being reported by this error log message. This code consists of a 5-bit major event code and an 11-bit subcode. The event codes and their meanings are listed in Appendix D.
Error-I	The I indicates the severity level of the end of error message is informational.

8.4.2.3 MSCP/TMSCP Error Flags - Table 8-6 defines the MSCP/TMSCP error flags.

Table 8-6 MSCP/TMSCP Error Flags

Bit Number	Bit Mask Hex.	Format Description
7	80	If set, the operation causing this error log message has successfully completed. The error log message summarizes the retry sequence necessary to successfully complete the operation.
6	40	If set, the retry sequence for this operation continues. This error log message reports the unsuccessful completion of one or more retries.
5	20	(MSCP-specific) If set, the identified logical block number (LBN) needs replacement.
4	10	(MSCP-specific) If set, the reported error occurred during a disk access initiated by the controller bad block replacement process.
0	1	If set, the error log sequence number has been reset by the MSCP server since the last error log message sent to the receiving class driver.

8.4.2.4 MSCP/TMSCP Controller Errors - Example 8-2 is a printout of a typical controller error.

Example 8-2 Controller Error Message Example

```
ERROR-E Data memory error (NXM or parity) at 5-Mar_1985 12:52:14.43
        Command Ref #   1C430008
        Err Seq #       1.
        Error Flags     41
        Event           012A
        Buffer Addr     143611
        Source Req.     0.
        Detecting Req.  3.
ERROR-I End of error.
```

NOTE

The direction of data transfer may be deduced from the types of requestors identified in the Source Requestor and Detecting Requestor field of the error message. In this example, the source requestor (the P.ioj) filled the buffer and requestor 3 is reading it.

This section lists controller and compare errors together because their format and fields are the same. These errors contain three optional fields in addition to those described in Table 8-5. The controller/compare specific fields are shown in Table 8-7. The actual descriptions for these errors follow in Section 8.4.2.4.1.

Table 8-7 MSCP/TMSCP Controller Error Message Field Description

Field	Description
Buffer Addr	This number (in octal) is the starting address of the HSC data buffer where the error occurred.
Source Req.	This number (in decimal) is the requestor that orginally filled the buffer with data.
Detecting Req.	This number (in decimal) is the requestor that detected the error.

8.4.2.4.1 Controller Error List - The following is an alphabetical listing and an explanation of the controller errors.

Compare Error

Message Error Level: E

Message Description: A compare error occurred during a Read-compare or a Write-compare operation. For the Read-compare operation, the HSC again obtains the data from the unit or shadow set and compares it with data obtained from host memory. If the data is not the same, a compare error results. For the Write-compare operation, the controller obtains data from each destination and compares it with data again obtained from host memory. If the data is not the same, a compare error results.

Field Service Action: Isolate the FRU by moving the disk or tape unit to another data channel and retrying the exact failing operation. Also, check the HSC data memory buffer address for repetition. If failure occurs on multiple physical units across multiple data channels and HSC data memory buffer address is not repetitive, investigate a possible K.ci problem.

Possible FRUs:

1. Isolated disk (or tape) unit
2. Data channel
3. M.std2
4. K.ci module set.

Data Bus Overrun

Message Error Level: E

Message Description: The HSC attempted to perform too many concurrent transfers, causing one or more of them to fail due to a data overrun or underrun. For example, data is sent to a bus by a data producer and then removed from the bus by a data consumer. If the producer sends data to the bus more quickly then the consumer can remove it, a data overrun occurs. If the consumer removes data more quickly than the producer can send it, a data underrun occurs.

Field Service Action: Determine which module is the data producer and which module is the consumer for a given error. Use the requestor number for assistance.

If the problem persists after replacing the suspect module(s), an HSC software problem should be investigated.

Possible FRUs: Source or detecting requestor modules.

Data Memory Error (NXM or Parity)

Message Error Level: E

Message Description: The HSC detected an error in internal Data memory. The error was either a parity error, detected via a parity generator/checker (data only - not address) on the requestor module, or a nonresponding address (the requestor did not receive a DACK from the memory module).

Field Service Action: Determine if this error is repetitive; if so, the problem is probably the M.std2 module. However, it may be a data bus problem caused by a number of things, such as failing bus drivers/receivers on the indicated requestor modules.

Possible FRUs: M.std2 or a possible data bus problem.

EDC Error

Message Error Level: E

Message Description: The sector was read with correct or correctable ECC and invalid EDC. A fault probably exists in the ECC logic of either this controller or the controller that last wrote the sector. Look at the source and detecting requestor fields in the error message to determine which requestor detected the error and the direction of the transfer (read or write).

Field Service Action: Determine if other errors indicate a problem with the data path circuitry on the indicated requestor modules.

Possible FRUs:

1. K.sdi
2. M.std2, if an address parity error on Data memory occurs, as this is checked by the EDC field.

Internal Consistency Error

Message Error Level: E

Message Description: A high-level check detected an inconsistent data structure. For example, a reserved field contained a nonzero value, or the value in a field was outside its valid range. This error is most likely caused by the requestor microcode or hardware.

Field Service Action: If the error is repetitive, check for consistent requestor numbers in detecting requestor field of error. Determine if any other surrounding error reports indicate a possible internal memory error.

Possible FRUs:

1. FRU noted in the detecting requestor field
2. M.std2 memory module.

SERDES Overrun

Message Error Level: E

Message Description: This error is either a SERDES overrun or underrun error. Either the drive is too fast for the controller, or a controller hardware fault prevented controller microcode from keeping up with data transfer to or from the drive.

Field Service Action: Determine if other errors have occurred that may indicate a K.sdi problem. Move the offending drive to another requestor. If the problem persists, test the drive further.

Possible FRUs: K.sdi module

PLI Receive Buffer Parity Error

Message Error Level: E

Message Description: When the data from the packet in a receive buffer on the PILA module was transferred to the K.pli module, a parity error was detected on the bus. In this case, parity is generated by the LINK module (L0100) and checked by the K.pli module (L0107). The PILA module stores the data without checking or generating parity.

Field Service Action: If failure is persistent and is accompanied by K.ci level 7 K interrupt HSC crashes, analyze K.ci module status code for more detailed information. Run Offline Test K diagnostic to test K.ci. Any error report should more clearly indicate the specific K.ci module failure. For very intermittent failures follow sequence of possible FRUs.

Possible FRUs:

1. PILA
2. K.pli
3. LINK.

PLI Transmit Buffer Parity Error

Message Error Level: E

Message Description: When data was being transferred from the K.pli to the PILA transmit buffer, a parity error was detected on the bus. In this case, parity is generated by the K.pli module and checked by the LINK module. The PILA module stores the data without checking or generating parity.

Field Service Action: If failure is persistent and is accompanied by K.ci level 7 K interrupt HSC crashes, analyze K.ci module status code for more detailed information. Run Offline Test K diagnostic to test K.ci. Any error report should more clearly indicate specific K.ci module failure. For very intermittent failures follow sequence of possible FRUs.

Possible FRUs:

1. PILA
2. LINK
3. K.pli

8.4.2.5 MSCP SDI Errors - The SDI type errors total 15. Example 8-3 shows a typical SDI error message. Table 8-8 describes the fields specific to SDI errors. Tables 8-9, 8-10, 8-11 and 8-12 further define the fields in Table 8-8. For the remaining fields, refer to Table 8-5.

Example 8-3 SDI Error Printout

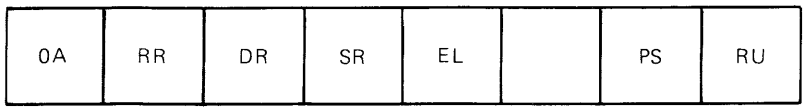
```
ERROR-E Drive Detected Error at 5-Mar_1985 12:52:14.43
        Command Ref #   00000000
        RA81 unit #     124.
        Err Seq #       4.
        Error Flags     40
        Event           00EB
        Request         1B
        Mode            00
        Error           80
        Controller      00
        Retry/fail      00
        Extended Status 88
                        00
                        03
                        00
                        07
                        4B
                        1A
        Requestor #     6.
        Drive port #    2.
ERROR-I End of error.
```

Table 8-8 SDI Error Printout Field Description

Field	Description
Request	This number (in hexadecimal) is a byte describing the state of the drive. Figure 8-6 shows the bits of this byte field, and Table 8-9 describes the bits. In this example, the 1B indicates: o RUN/STOP switch in o Port switch in o Log information in extended area o Spindle ready
Mode	This number (in hexadecimal) is a byte describing the mode of the unit. Figure 8-7 shows the bits of this byte field, and Table 8-10 describes the bits. In this example, the 00 indicates: o No subunits are write protected. o The disk is in 512-byte sector format.

Field	Description
Error	This number (in hexadecimal) is a byte describing the errors in the unit. Figure 8-8 shows the bits of this byte field, and Table 8-11 describes the bits. In this example, the 80 indicates a drive error has occurred, and the drive FAULT lamp may be on.
Controller	This number (in hexadecimal) is a byte describing the subunits with attention available messages suppressed in the controller and a status code indicating various states of drive operation. Figure 8-9 shows the bits of this byte field, and Table 8-12 describes the bits. In this example, the 00 indicates: o No subunits with attention available message suppressed in the controller o Drive normal operation
Retry/fail	This number (in hexadecimal) is a byte containing one of two types of information depending upon the status of the DF bit in the Error field. The DF bit describes the drive initialization process. The DF bit is a zero if the drive initialization was successful. In this case, the Retry/fail field contains the retry count from the previous operation. For example, a Seek operation required 14 retries to be successful. If a GET STATUS command is initiated, the Retry/fail field contains the number 14. The DF bit set indicates the drive initialization failed, and therefore, the Retry/fail contains a specific drive error code. This error code is defined in the appropriate drive service manual. In this example, 00 indicates no retry count exists for the previous operation. (The DF bit is zero in the Error field.)

Field	Description
Extended Status	These bytes, in hexadecimal, contain the extended status of the particular drive. (In this example it is an RA81.) Refer to the appropriate drive service manual for the meaning of these bytes. In this example, the extended status is: o 88 - Controller command functional code last executed by the drive. (In this case, a GET SUBUNIT CHARACTERISTICS command.) o 00 - Interface error status bits which are all reset. o 03 - Low-order cylinder address bits of the last Seek operation. o 00 - High-order cylinder address bits of the last Seek operation. o 07 - The present group address. o 4B - Error code (index pulse error) displayed by the drive LEDs during the execution of a drive-resident diagnostic. o 1A - Error code (Servo fine positioning error) displayed on the operator control panel of the RA81.
Requestor #	This number, in decimal, is the number of the requestor connected to the drive.
Drive port #	This number, in decimal, is the number of the port on the requestor. (The ports are numbered 0 through 3.)

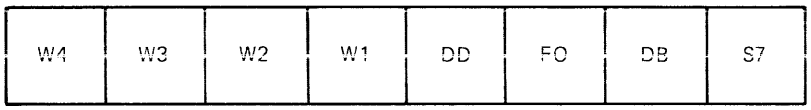


CX-1121A

Figure 8-6 Request Byte Field

Table 8-9 Request Byte Field Description

Bits	Description
OA	A logical one in this position indicates the drive is unavailable to the controller. A logical zero indicates the drive is available to the controller.
RR	A logical one in this position indicates the drive requires an internal readjustment. Some drives do not use this bit.
DR	A logical one in this position indicates a request is outstanding to load a diagnostic in the drive microprocessor memory. A logical zero indicates no diagnostic is being requested of the host system.
SR	A logical one in this position indicates the drive spindle is up to speed. A logical zero indicates the drive spindle is not up to speed.
EL	A logical one in this position indicates usable information in the extended status area. A logical zero indicates no information is available in the extended status area.
PS	A logical one in this bit position indicates the drive port select switch for this controller is pushed in (selected). A logical zero indicates the switch is out.
RU	A logical one in this position indicates the RUN/STOP switch is pushed in (RUN). A logical zero indicates the switch is out (STOP).



CX-1122A

Figure 8-7 Mode Byte Field

Table 8-10 Mode Byte Field Description

Bits	Description
W4-W1	Logical ones in any of these four bit positions represent the write-protect status for the subunit. (For example, a 0001 indicates subunit 0 within the selected drive is write-protected.)
DD	A logical one in this position indicates the drive was disabled by a controller error routine or diagnostic. The Fault light is on when this bit is set. A logical zero indicates the drive is enabled for communication to a controller.
FO	A logical one in this position indicates the drive can be formatted.
DB	A logical one in this position indicates the diagnostic cylinders on the drive can be accessed.
S7	A logical one in this position indicates the 576-byte sector format is selected. A logical zero indicates that the 512-byte sector format is selected.

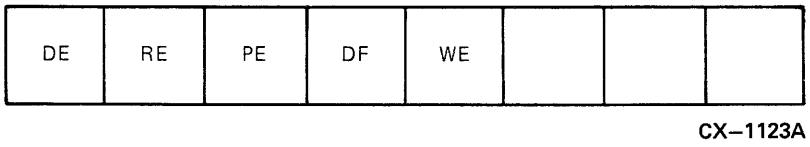
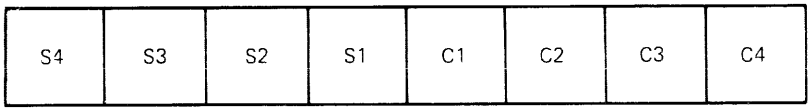


Figure 8-8 Error Byte Field

Table 8-11 Error Byte Field Description

Bits	Description
DE	A logical one in this position indicates a drive error has occurred and the drive FAULT lamp may be on.
RE	A logical one in this position indicates an error occurred in the transmission of a command between the drive and the controller. The error could be a checksum error or an incorrectly formatted command string.
PE	A logical one in this position indicates improper command codes or parameters were issued to the drive.
DF	A logical one in this position indicates a failure in the initialization routine of the drive.
WE	A logical one in this position indicates a write-lock error has occurred.



CX–1124A

Figure 8-9 Controller Byte Field

Table 8-12 Controller Byte Field Description

Bits	Description
S4-S1	This is a 4-bit representation of the subunits with Attention Available messages suppressed in the controller. The rightmost bit position represents subunit 0. The leftmost bit position represents subunit 2. If one of the bits is set, it indicates the controller is not to interrupt the host CPU with an Attention Available message when the specified subunit raises its available real-time drive status line to the controller. The S4 through S1 bits reflect the results of a CHANGE CONTROLLER FLAGS command in which Attention Available messages are not desired for certain subunits.
C4-C1	This is a 4-bit drive status code indicating various states of drive operation. At the present time, only three codes are valid: o 0000 - Drive normal operation o 1000 - Drive is offline because it is under the control of a diagnostic o 1001 - Drive is offline due to another drive having the same unit identifier (for example, serial number, drive type, class).

Following is an alphabetical listing of SDI type errors with an explanation of each.

NOTE

When the HSC marks the drive as inoperative, it places the drive in a state of Unit-Offline with a substate of unit-inoperative relative to this HSC.

Controller-Detected Transmission or Time Out Error

Message Error Level: E

Message Description: The controller detected an invalid framing code or a checksum error in a Level 2 response from the SDI drive.

Field Service Action: Determine if this error is occurring on more than one drive which may indicate a K.sdi problem. However, if it is occurring on only one drive, the SDI cable or the drive may be at fault. Refer to the appropriate drive service manual for assistance with drive FRUs.

Possible FRUs:

1. SDI cable
2. Drive SDI interface module
3. K.sdi module
4. SDI transition bulkheads

Drive Clock Dropout

Message Error Level: E

Message Description: Either data or state clock was missing when it should have been present. This is detected by the requestors connected to this SDI drive.

Field Service Action: Determine if this error is occurring on more than one drive which may indicate a K.sdi problem. However, if it is occurring on only one drive, the SDI cable or the drive may be at fault. If other errors surround or precede this one, those errors may have sequentially triggered this error. Refer to the appropriate drive service manual for assistance with drive FRUs.

Possible FRUs:

1. SDI cables
2. Drive SDI interface module
3. K.sdi module
4. SDI transition bulkheads

Drive Inoperative

Message Error Level: E

Message Description: The drive is generating so many unrecoverable errors that it appears inoperative. Once the HSC reports the drive as inoperative, the drive state clocks must transition to return the drive to an operational state.

Field Service Action: Refer to the drive service manual. Run ILDISK to help isolate failure between HSC and drive.

Possible FRUs:

1. Drive modules (Refer to Drive service manual.)
2. K.sdi module
3. SDI cables

Drive-Detected Error

Message Error Level: E

Message Description: The controller received a GET STATUS command or unsuccessful response with EL set, or the controller received a response with the D flag set and does not support automatic diagnosis for that SDI drive type.

Field Service Action: Determine if the drive has a hard fault (fault light on, and an error code in the drive microprocessor LEDs). Refer to the drive service manual for assistance with drive internal diagnostics and LED error codes. Decode remaining error message bytes for more detailed error information. If error message decoding does not clearly indicate a drive error, move the drive to another requestor (or requestor port) to help isolate failure between HSC and drive.

Possible FRUs:

1. Drive modules (Refer to drive service manual.)
2. SDI cables
3. SDI bulkheads

Drive-Requested Error Log (EL Bit Set)

Message Error Level: E

Message Description: The controller requested a drive error log because the drive returned a status message with the EL bit set in the request byte field.

Field Service Action: Determine what drive-detected error (previous error description) caused the drive to request a drive error log by finding the error in the error log report. Also decode remaining fields in the drive status response of this error message and any preceding errors on the unit.

Possible FRUs: Drive modules (Refer to the drive service manual.)

Lost Read/Write Ready

Message Error Level: E

Message Description: Read/Write Ready drops when the controller attempts to initiate a transfer or at the completion of a transfer with Read/Write Ready previously asserted. This usually results from a drive-detected transfer error, where additional error log messages containing the drive-detected error subcode may be generated.

Field Service Action: Look for surrounding drive-detected errors and/or associated disk transfer error log. Move suspect drive to another port or data channel to help isolate failure, as this error may be caused by any of several communication components.

Possible FRUs:

1. Drive modules (Refer to drive service manual.)
2. K.sdi module
3. SDI cables
4. SDI transition bulkheads

Lost Receiver Ready

Message Error Level: E

Message Description: Receiver Ready was negated when the controller attempted to initiate an SDI disk transfer or did not assert at the completion of a transfer. This includes all cases of the controller timeout expiring for a transfer operation (LEVEL 1 REAL TIME command).

Field Service Action: Look for a probable drive error or a possible SDI cable problem. Move suspect drive to another port or data channel to help isolate failure, as this error may be caused by any of several communication components.

Possible FRUs:

1. Drive modules (Refer to drive service manual.)
2. K.sdi module
3. SDI cables
4. SDI transition bulkheads

Position or Unintelligible Header Error

Message Error Level: E

Message Description: The drive reported a Seek operation was successful by returning successful status in response to the INITIATE SEEK SDI command and asserting R/W Ready when on the desired cylinder. However, the controller determined the drive had positioned itself to an incorrect cylinder. The header read from the drive is consistent (three out of four header copies are identical) but does not match the desired target header value. The error is considered recoverable if the Error Flags bit indicates success or a subsequent replacement succeeds.

Field Service Action: The drive Servo system or media is probably at fault in this case. If one is available, move the drive to a different requestor. A drive failure is indicated if the failure persists on the new requestor.

Possible FRUs:

1. Drive modules (Refer to drive service manual.)
2. K.sdi module

Pulse or Parity Error

Message Error Level: E

Message Description: The controller detected a pulse error on either the SDI drive state or data line, or the controller detected a parity error in a drive state frame. The HSC does an SDI GET STATUS command, reports any errors from it, and then clears those errors, if possible. After this, the HSC retries the original command up to two more times before considering the error unrecoverable.

Field Service Action: If the error is reported on more than one drive, a K.sdi problem is indicated. If the error is reported on only one drive, an SDI cable or drive problem is indicated.

Possible FRUs:

1. Drive modules (Refer to drive service manual.)
2. SDI cable
3. SDI transition bulkhead
4. K.sdi Module

SI Clock Resumption Failed After INIT

Message Error Level: E

Message Description: The drive clock did not resume following a controller attempt to initialize the SDI drive. This implies the drive encountered a fatal initialization error. Closely examine error logs for surrounding disk errors, as this error may be the result of a previously-reported drive error.

Field Service Action: Determine if this drive has encountered any other related problems which may be found in an appropriate error log report. Also, this error may be due to an SDI cable problem.

Possible FRUs:

1. Drive modules (Refer to drive service manual.)
2. SDI cable

SI Clock Persisted After INIT

Message Error Level: E

Message Description: The drive clock did not cease following a controller attempt to initialize the SDI drive. This implies the drive did not recognize the initialization attempt. This error condition causes the HSC to retry the INIT command eight more times before marking the drive inoperative.

Field Service Action: Determine if this drive has encountered any other related problems which may be entered in an appropriate error log report. Also, this error may be due to an SDI cable problem. Closely examine error logs for surrounding disk errors, as the error may be a result of a previously-reported drive error.

Possible FRUs:

1. Drive modules (Refer to drive service manual.)
2. SDI cable

SI Command Timeout

Message Error Level: E

Message Description: The controller timeout expired for either a Level 2 exchange or the assertion of READ/WRITE READY after an Initiate Seek. The HSC retries the command three more times, reinitializing the SDI drive each time. If the error persists on a single SDI level 2 exchange, the drive is marked inoperative.

Field Service Action: Determine if this drive has encountered any other related problems which may be found in an appropriate error log report. Also, this error may be due to an SDI cable problem. Closely examine error logs for surrounding disk errors, as the error may be a result of a previously-reported drive error.

Possible FRUs:

1. Drive modules (Refer to drive service manual.)
2. SDI cable

Ensure the drive and all HSC modules are at the latest revision levels.

SI Receiver Ready Collision

Message Error Level: E

Message Description: This error occurs when the drive fails to follow the SDI protocol during SDI command/reception. For example, the controller sends the drive a command, asserts Controller Receiver Ready, and waits for the SDI response. The following lists the possible drive operations that lead to this error:

1. The drive fails to deassert Drive Receiver Ready. In this case, the drive indicates it did not receive the command.

2. The drive deasserts Drive Receiver Ready and then reasserts it before sending a proper SDI response. In this case, the drive believes it has sent a response and is indicating so by re-asserting Drive Receiver Ready, yet the controller has never received the response.

The HSC K.sdi detects this error. The HSC functional code does an SDI GET STATUS command and clears the drive of any errors found. The original command is then retried. This cycle is repeated twice before the drive is initialized by the HSC, and the entire operation is done two more times. If the failure persists, the drive is marked inoperative.

Field Service Action: Determine if this drive has encountered any other related problems which may be found in an appropriate error log report. Also, this error may be due to an SDI cable or SDI transceiver/encoder/decoder problem. Closely examine error logs for surrounding disk errors, as this error may be the result of a previously-reported drive error.

Possible FRUs:

1. Drive modules (Refer to drive service manual.)
2. SDI cable
3. K.sdi module

SI Response Length or Opcode Error

Message Error Level: E

Message Description: A Level 2 response from the drive had correct framing codes and checksum but was not a valid response within the constraints of the SI protocol. The response had an invalid opcode, was an improper length, or was not a possible response in the context of the exchange.

The HSC K.sdi detects this error. The HSC functional code does an SDI GET STATUS command and clears the drive of any errors found. The original command is then retried. This cycle is repeated twice before the drive is initialized by the HSC, and the entire operation is done two more times. If the failure persists, the drive is marked inoperative.

Field Service Action: Determine if the drive has experienced other similar errors. Closely examine error logs for surrounding disk errors, as this error may be the result of a previously-reported drive error.

Possible FRUs:

1. Drive modules (Refer to drive service manual.)
2. K.sdi module

SI Response Overflow

Message Error Level: E

Message Description: A drive sent back more frames than the reception buffer could hold. This can be caused by a hung drive microdiagnostic or a malfunctioning K.sdi.

Field Service Action: Determine if the drive is failing in other ways, indicating a drive problem. If not, the K.sdi may be the more likely cause.

Possible FRUs:

1. Drive modules (Refer to drive service manual.)
2. K.sdi module

8.4.2.6 Disk Transfer Errors - Disk transfer errors are either data or media format type errors. Example 8-4 shows an example Disk Transfer error printout, and Table 8-13 describes the various fields of the printout.

Example 8-4 Disk Transfer Error Printout

```
ERROR-E SEVEN Symbol ECC Error at 27-Mar-1985 12:15:15.00
        Command Ref #       50400015
        RA81 unit #         120.
        Err Seq #           9.
        Error Flags         E0
        Event               01C8
        Recovery level      0.
        Recovery count      0.
        LBN                 426978
        Orig err flags      100020
        Recovery Flags      000003
        Lvl A retry cnt     1.
        Lvl B retry cnt     0.
        Buffer addrs        143022
        Source Req.         5.
        Detecting Req.      5.
Error-I End of error.
```

Table 8-13 describes the fields in a disk transfer error message not described in Table 8-5. Unless otherwise specified, all fields in this table are shown in decimal numbers. These fields are specific to an RA81 disk and may not be the same for other RAXX type drives.

Table 8-13 Disk Transfer Error Printout Field Description

Field	Description
RA81 unit #	This is the number of the unit the error log message relates to, or is zero if the message does not relate to a specific unit. In this example, the RA81 indicates the drive is an RA81 and is unit 120.
Recovery level	This number indicates the error recovery level used for the most recent transfer attempt by the unit. In this example, the 0 indicates it used error recovery level 0. An RA81 only has a recovery level of 0 (recalibration).
Recovery count	This number indicates the number of times the recovery level was tried. In this example, the 0 indicates the recovery level was not retried.
LBN	This number indicates the logical block number. In this example, the LBN is 426978.

Field	Description
Orig err flags	This number (octal) indicates the original errors associated with this error. Table 8-14 describes the bits associated with this field. In this example, the 100020 indicates: o ECC Error o EDC error
Recovery flags	This number (octal) indicates the recovery flags the software processes should take to recover from this error. Table 8-15 describes the bits associated with this field. In this example, the 000003 indicates: o An LBN should be replaced. o The current error should be logged on the console and to the host if a connection is present.
Lvl A retry cnt	This number indicates the number of times the HSC attempted the Level A recovery routines. These routines are those not requiring any exhaustive SI exchanges as part of the recovery sequence. In this example, the 1 indicates the ECC error correction was completed in the HSC without going over the SI.
Lvl B retry cnt	This number indicates the number of times the HSC attempted the Level B recovery routines. These routines require extensive SDI exchanges as part of the recovery sequence. In this example, the 0 indicates no Level B recovery was attempted.
Buffer addrs	This number (octal) is the address of the HSC internal data buffer associated with this error. In this example, the buffer address is 143022.
Source Req.	This number is the requestor that filled the buffer with data. In this example, the 5 indicates the source requestor was requestor number 5. A requestor of 1 in this field would indicate a disk Write operation. All other values would indicate a disk Read operation.
Detecting Req.	This number is the requestor that detected that error. In this example, the 5 indicates requestor number 5 detected the ECC error.

Table 8-14 shows definitions of the original error flags and Table 8-15 defines the recovery flags.

Table 8-14 Original Error Flags Field Description

Bits	Mask (Octal)	Definition
15	100000	ECC error
14	040000	SERDES overrun error
13	020000	SDI RESPONSE/DATA line pulse error
12 and 11	014000	Suspected position error - low header mismatch
12	010000	Header sync timeout
11	004000	Header compare error - compare-64 performed
10	002000	Data sync timeout
09	001000	Drive clock timeout
08	000400	SDI STATE line pulse error
07	000200	Data Bus overrun
06	000100	Data Memory parity error
05	000040	Data Memory NXM
04	000020	EDC error
03 and 02	000014	READ/WRITE READY down at end of sector
03	000010	Lost READ/WRITE READY before transfer began
02	000004	Lost RECEIVER READY before transfer began
01	000002	Forced error (EDC = ones complement of correct EDC)
00	000001	Drive inoperative

Table 8-15 Recovery Flags Field Definition

Bit	Mask (Octal)	Definition
05	000040	Indicates the error count reported by the ILEXER should be updated
04	000020	Indicates an error log message has already been generated for the current error
03	000010	Indicates an entry for the desired logical block number was found
02	000004	Indicates revectoring and replacement should be suppressed
01	000002	Indicates the current error should be logged on the console and to the host if a connection is present
00	0000001	Indicates the logical block should be replaced

The following is an alphabetical listing of the disk transfer errors with an explanation of each error.

Data Synch Not Found

Message Error Level: E

Message Description: This error occurs when the SERDES 16 does not detect the SYNC character (26BC hex) immediately preceding read data from the disk drive. The K.sdi has already read a valid header and is awaiting the Data SYNC character.

Field Service Action: Determine if additional errors occur from this drive to indicate a drive or media error. If not, the problem is probably the K.sdi module.

Possible FRUs:

1. Drive modules (Refer to drive service manual.)
2. K.sdi module
3. SDI interface

ECC Errors

Message Error Level: E

Message Description: The following description covers all of the ECC error types:

- Uncorrectable ECC Error
- One Symbol ECC Error
- Two Symbol ECC Error
- Three Symbol ECC Error
- Four Symbol ECC Error
- Five Symbol ECC Error
- Six Symbol ECC Error
- Seven Symbol ECC Error
- Eight Symbol ECC Error

ECC errors occur when the data read from the disk does not agree with the data written. When data is written to the disk, an ECC is calculated (by the R-S GEN) and appended to the end of the sector. When the data is subsequently read from the sector, the ECC is revalidated. The two possible results are:

1. The data error falls within the ECC error correction capability (less than nine 10-bit symbols in error) and data correction is performed. In this case, no data errors are shown.

2. The data error does not fall within the error correction capability of the ECC, and the error is retried according to drive dependent parameters. If all of the retries fail, an uncorrectable ECC error occurred, and a bad block is reported via an end packet.

NOTE

An uncorrectable ECC error can also occur if the Suppress Error Correction modifier is chosen and the transfer encounters any type of ECC error.

Field Service Action: Determine if the ECC errors are just normal occurrences or if a very large number of blocks is being replaced. The latter indicates the drive may have a read path problem.

Possible FRUs:

1. Drive modules (Refer to drive service manual.)
2. K.sdi module

Forced Error

Message Error Level: E

Message Description: The sector was written with a Force Error modifier indicating this is a replaced image and the original data could not be read correctly using retries and the ECC algorithms.

Field Service Action: Backup the media.

Possible FRUs: None

Header Error

Message Error Level: E

Message Description: The subsystem reads an invalid or inconsistent header for the requested sector. The header is considered invalid if all of the following are true:

- The header is consistent (three out of four copies match).
- Two out of four of the low-word header values match the desired target header low-word value.
- The high-word header values do not match the respective target header values.

For recoverable errors, this code implies a retry of the transfer read a valid header. For unrecoverable errors, this code implies the subsystem attempted nonprimary revectoring and determined the requested sector is not revectored. Causes of an invalid header include header missync, header sync timeout, and an unreadable header.

Field Service Action: Determine if this error is repetitive on this unit indicating a deteriorating media.

Possible FRUs:

1. Drive modules (Refer to drive service manual.)
2. K.sdi module

RCT Corrupted Error

Message Error Level: E

Message Description: The RCT search algorithm encountered an invalid RCT entry. The subcode may be returned under the following conditions:

- o During replacement of a block
- o During nonprimary revectoring of a block
- o When bringing a unit online

Field Service Action: Determine if this error is repetitive for this unit possibly indicating a defective media or drive read path failure.

Possible FRUs: Drive modules (Refer to drive service manual.)

8.4.3 Bad Block Replacement Errors (BBR)

Another type of error displayed on the console terminal is for a bad block replacement request. The bad block replacement request is a result of the one of the following errors:

- o Data sync timeout
- o ECC symbol error above the threshold
- o Header compare error
- o Header sync timeout
- o Loss of R/W Ready at end of read from disk (SERDES read)
- o Uncorrectable ECC

Example 8-5 shows a bad block replacement message. This message reports completion, successful or unsuccessful, of a bad block replacement attempt. A message is generated regardless of the success or failure of the replacement attempt. Refer to Table 8-16 for a definition of the fields in the message explicit to this type of message. Table 8-17 describes replace flag bits. Fields generic to all MSCP/TMSCP error messages are described in Table 8-5.

Example 8-5 Bad Block Replacement Error Printout

```
ERROR-W Bad Block Replacement (Success) at 18-Dec-1985 18:05:37.1
        Command Ref #   B8590012
        RA60 Unit #     251
        Err Seq #       2
        Error Flags     80
        Event           0014
        Replace Flags   8000
        LBN             205
        Old RBN         0
        New RBN         5
        Cause Event     00E8
ERROR-I End of error
```

Table 8-16 defines BBR error fields not previously described in Table 8-5. The replace flags are defined in Table 8-17.

Table 8-16 Bad Block Replacement Error Printout Field Definition

Field	Description
Replace Flags	This number, in hexadecimal, indicates bit flags used to report in detail the outcome of the bad block replacement attempt. (Refer to Table 8-17). In this example, the 8000 indicates the block was verified as bad.
LBN	This number, in decimal, is the logical block number that is the target of the replacement. In this example, the LBN is 205.
Old RBN	This number, in decimal, indicates the RBN the bad LBN was formerly replaced with, or zero if it was not formerly replaced. In this example, the 0 indicates it was not formerly replaced.
New RBN	This number, in decimal, indicates the RBN the bad LBN was replaced with, or is zero if no actual replacement was attempted. In this example the new RBN is 5.
Cause Event	This number, in hexadecimal, is the event code from the original error that caused the replacement to be attempted. The number is zero if that event code not available. (Refer to Appendix C.) In this example, the 00E8 indicates an uncorrectable ECC error caused the bad block replacement.

Table 8-17 Replace Flags Bit Description

Bit Number	Bit Mask Hex.	Replace Flag Bit Definition
15	8000	Replacement Attempted This bit is set if the suspect bad block indeed tested bad during the initial stages of the replacement process. If not set, the suspect block did not check bad and no replacement was completed.
14	4000	Forced error The data from the suspect bad block could not be corrected or obtained without error. The Forced Error Indicator will be written to the replacement block along with the bad data from the block that was replaced. The user data from the bad block is read with a forced error when accessed. If this condition occurs frequently on a specific drive, then a closer analysis of the drive for possible problems would be recommended.
13	2000	Nonprimary revector This bit is set if the replacement process was accomplished and required putting the bad blocks data into a replacement block that is not the bad blocks primary RBN.
12	1000	Replace command failure This bit is set during the replacement process if the status coming back from the execution of the MSCP REPLACE command is not successful. If this occurs, the drive should not be used until it can be reformatted.
11	800	RCT inconsistent This bit is set if the Replacement Control Tables are not usable. The drive should not be used until it can be reformatted.
10	400	Bad replacement block This bit is set if the bad block reported is a replacement block. The replacement block can be replaced just like any LBN.

The following is an alphabetic listing of the BBR errors with an explanation of each error.

Bad Block Replacement (Block OK)

Message Error Level: Warning.

Message Description: Block tested OK - not replaced.

Field Service Action: Monitor drive for the frequency of these reports. If frequency increases, troubleshoot the error that triggers BBR.

Possible FRUs: Refer to **Cause Event** error message field in Table 8-16.

Bad Block Replacement (Drive Inoperative)

Message Error Level: Warning

Message Description: Replacement failure--drive access failure. One or more transfers specified by the replacement algorithm failed. If necessary and possible, write-protect the drive and perform a volume backup immediately.

Field Service Action: Drive should be tested further. Move the drive to another K.sdi (or to just another K.sdi port) if available. If the problem persists, failure is most likely in the drive.

Possible FRUs: Drive module (Refer to drive service manual.)

Bad Block Replacement (RCT Inconsistent)

Message Error Level: Warning

Message Description: Replacement failure--the RCT table is not usable.

Field Service Action: Drive media should not be used until replaced or verified as good. If necessary, write-protect this drive, and have the customer perform a volume backup immediately. Further testing of drive may be necessary.

Possible FRUs: Drive module (Refer to drive service manual.)

Bad Block Replacement (REPLACE Failed)

Message Error Level: Warning

Message Description: Replacement failure - REPLACE command or its analogue failed. The status returned from the replacement process indicates the command was not successful.

Field Service Action: Drive media should not be used until it is replaced or verified as good. If necessary, write-protect this drive and have the customer perform a volume backup immediately. Further testing of drive may be necessary.

Possible FRUs: Drive module (Refer to drive service manual.)

Bad Block Replacement (Success)

Message Error Level: Warning

Message Description: The bad block was successfully replaced.

Field Service Action: Monitor drive for the frequency of these reports. If frequency increases, troubleshoot the error triggering BBR.

Possible FRUs: Refer to Cause Event error message field in Table 8-16.

8.4.4 TMSCP-Specific Errors

The Tape Mass Storage Control Protocol (TMSCP) error messages printed out at the console terminal are one of the following types:

- STI Communication or Command Errors
- STI Formatter Error Log Errors
- STI Drive Error Log Errors
- Controller Errors (Section 8.4.1)

8.4.4.1 STI Communication Or Command Errors - The following is an example of the console printout of an STI communication or command Error.

Example 8-6 shows the printout and Table 8-18 explains the fields additional to those defined in Table 8-5.

Example 8-6 STI Communication or Command Error Printout

```
ERROR-E Drive detected error at  6-Mar-1985 09:51:11.88
        Command Ref #    864E0004
        TA78 unit #      0
        Err Seq #        12
        Error Flags      40
        Event            00EB
        Position         13026
        GSS Text         02 00 00 00
                         05 00 00 00 00 00 00 00
Error-I End of error
```

Table 8-18 STI Communication or Command Error Printout Field Description

Field	Description
Event	The number, in hexadecimal, identifies the specific error or event reported by this error log message. The event codes and their meanings are shown in Appendix C. In this example, the 00EB means drive detected error.
Position	This is the last known tape position the formatter received. This is given in gap counts from BOT. In this example, the number 13026 means 13026 gaps from BOT.
GSS Text	The GSS Text field is the response received by the HSC from the formatter when the HSC issues the GET SUMMARY STATUS (GSS) and TOPOLOGY commands. The GSS text in this example is 02 00 00 00 05 00 00 00 00 00 00 00. This means Level 2 protocol error, Speed Management Enabled, Zero Threshold. See Section 8.4.4.5 for details on field definitions and bit decoding.

8.4.4.2 STI Formatter Error Log - The following is an example of the console printout of an STI Formatter Error Log. Example 8-7 shows the printout, and Table 8-19 explains the fields not previously defined in Table 8-5.

Example 8-7 STI Formatter Error Log Printout

```
ERROR-E Tape Formatter Requested Error Log at 30-Jan-1986 11:20:09.31
        Command Ref #    43900012
        TA81 unit #      95
        Err Seq #        47
        Format Type      08
        Error Flags      40
        Event            FF6C
        Position         1057
        Formatter E Log  40 00 00 81 00 00 00
                         01 98 72 00 00 00 00
                         C4 48 00 00

ERROR-I End of error.
```

Table 8-19 STI Formatter Error Log Field Description

Field	Description
Position	The last known tape position the formatter received. This is given in gap counts from BOT. In this example, the number 1057 means 1057 gaps from BOT.
Formatter E Log	See Table 8-20.

Table 8-20 Formatter E Log

BYTE No.	BYTE DATA	DESCRIPTION
1	40	Formatter error
2	00	
3	00	
4	81	Data pulse parity error during data transfer

The information contained in these fields is product specific. Refer to the appropriate drive manual for a description of the remainder of the bytes.

8.4.4.3 STI Drive Error Log - The following is an example of a console printout of an STI Drive Error Log. Example 8-8 shows the printout, and Table 8-21 explains the fields additional to those defined in Table 8-5. Table 8-22 describes GEDS Text field, and Table 8-23 describes the Drive Error Log field.

Example 8-8 STI Drive Error Log Printout

```
ERROR-E Tape Drive Requested Error Log at 5-Mar-1985 14:43:31.15
        Command Ref #   D6300023
        TA78 unit #     528
        Err Seq #       210
        Error Flags     40
        Event           FF6B
        Position        1
        GEDS Text       7D 04 5000 01000000
        Drive Error Log 00 00 00 00 50 3B 04 0D
                        46 FF 07 FF 00 00 00 00
                        81 00 00 00 FF 22 04 C4
                        00 00 80 FF 17 94 00 08
                        00 00 D9 FF FF FF FF FF
                        FF 47 E6 E0 00 16 25 97
                        A2 00 00
ERROR-I End of error
```

Table 8-21 STI Drive Error Log Field Description

Field	Description
Position	The last known tape position where the HSC believes the tape drive is upon successful completion of all outstanding commands. This is given in gap counts from BOT. In this example the number 1 means 1 gap from BOT.
GEDS Text	See Table 8-22
Drive Error Log	See Table 8-23

See also Section 8.4.4.4 for field definitions and bit decoding.

Table 8-22 GEDS Text

Byte No.	Byte DATA	Description
1	7D	125 IPS tape drive
2	04	6250 BPI GCR encoding
3	50	
4	00	MSCP unit number = 80.
5	01	
6	00	
7	00	
8	00	GAP count = 1.

The information shown in Table 8-23 is product specific to the TA78. See the TA78 service manual for details.

Table 8-23 STI Drive Error Log

Byte No.	Byte Data	Description
1	00	No SOFT error
2	00	No SOFT error
3	00	
4	00	
5	50	Last STI level 2 cmd = 50(X)
6	3B	
7	04	Error ID number = 50. Operational error fault number indicates possible cause general area = unknown fault number
8	0D	RMC write fail bits
9	46	Statistics select clock stopped STATUS VALID
10	FF	NON-BOT cmd sts is ok
11	07	Last cmd sent to M8953 via "RCMD" = normal NON-BOT read
12	FF	Read channel AMTIE sts (CH 7:0)
13	00	
14	00	Read channel illegal sts (CH 7:0)
15	00	
16	00	End mark for read channels 7:0
17	81	Weak amplitude on parity bit ECC corrected output (parity bit)
18	00	Read channel PE postamble detect
19	00	Data from read channels to ECC
20	00	CRC checker output bits
21	FF	Corrected data (ECC to CRC)

Byte No.	Byte Data	Description
22	22	2-TRK ECC performed on data "AMTIE" during data of record
23	04	Channel 0 tie bus 2
24	C4	Channel 3 tie bus 3
25	00	
26	00	
27	80	Tie bus = 0F(X)
28	FF	Tape unit bus line AMTIE 7:0
29	17	AMTIE parity READ parity WCS parity Tape unit present
30	94	TU bus line read data 7:0
31	00	
32	08	"CRC" to "WMC DR" bus
33	00	Tape unit selected = 0.
34	00	Tape unit selected = 0.
35	D9	R/W Data, intermediate DRD bus
36	FF	
37	FF	Byte count = 65535.
38	FF	
39	FF	PAD counter = 65535.
40	FF	
41	FF	Unknown error code
42	47	"DR MBD" parity error
43	E6	"PE" write parity error POWER OK

Byte No.	Byte Data	Description
44	E0	Online READY ON READY
45	00	
46	16	Position 0 (normal) 125 IPS tape drive
47	25	
48	97	Tape unit serial #2597.
49	A2	AMTIE threshold field = 2. READ ENABLE WRITE BIT 4
50	00	
51	00	

8.4.4.4 Breakdown Of GEDS Text Field - The following is an example of a tape drive related error message printed on the HSC70 terminal.

Example 8-9 Tape Drive Related Error Message

```
ERROR-W Tape Drive Requested Error Log at 15-Aug-1984 18:43:05.80
        Command Ref      00001D8E
        TA78 unit        20.
        Err Seq          1.
        Error Flags      40
        Event            FF6B
        Position         2.
        GEDS Text        7D 02 0014 00000002
        Drive Error Log  00 00 00 00 C5 38 04 04
                         46 FF 07 FF 00 00 00 00
                         81 00 00 21 FF B0 00 04
                         00 00 80 FF 17 DE 00 08
                         00 00 21 FF FF 00 00 99
                         99 47 F4 E8 00 56 85 19
                         A2 0A 80 FF 17 DE
```

Both the GEDS Text and Drive Error Log portions of this message result from a GET EXTENDED DRIVE STATUS command to the drive from the HSC70. The Drive Error Log portion can be interpreted by referencing the service manual for the appropriate tape drive. (The preceding example is for a TA78 drive.)

Following is a breakdown of the information contained in the GEDS Text field. The leftmost byte is referenced as the First Byte and the rightmost byte as the Eighth Byte.

Bytes in the GEDS Text field are described as follows:

First Byte = Speed: Currently sets speed of the drive; it is an integer value (in hex) in inches per second (IPS) rounded down to the nearest integer. For a totally variable speed drive, the speed returned is the lower bound on the range of permissible speeds. In the example shown, this field contains a value of 7D which corresponds to 125 IPS.

Second Byte = Density: This is the current operating density of the tape unit. Only one bit is set to indicate the current operating density.

04 = 6250 BPI 02 = 1600 BPI 01 = 800 BPI

Third and Fourth Bytes = Unit Number: These bytes contain the drive unit number (hex).

Fifth through Eighth Bytes = Gap Count: The formatter's gap count is from the beginning of the tape to where the tape drive is. The contents of this field may differ from the Position field in this error message. The HSC's gap count is contained in The Position field at the end of successful completion of all outstanding commands.

8.4.4.5 Breakdown Of GSS Text Field - Following is another example of a tape drive related error message printed at the HSC70 console.

```
ERROR-E Drive detected error at 18-Aug-1984 12:05:34.82
        Command Ref     0346003
        TA78 unit       3.
        Err Seq         7.
        Error Flags     40
        Event           00EB
        Position        0.
        GSS Text        02 20 00 00
                        28 00 00 00 00 00 14 00
ERROR-I End of error.
```

The HSC70 received the GSS Text field form of this error message from the tape formatter when the HSC70 issues the GET SUMMARY STATUS (GSS) and TOPOLOGY commands. The field is also the unsuccessful response for all Level 2 commands. Following is a breakdown of this response and an interpretation of bits contained in it.

AF	A3	A2	A1	A0	OA	P	D	SUMMARY MODE BYTE 1
FE	TE	PE	DF					SUMMARY ERROR BYTE
				PB	EL	RP	RT	FD	SUMMARY MODE BYTE 2
C1	C2	C3	C4	C5	C6	C7	C8	CONTROLLER BYTE
TM	EOT	BOT	WL	OL	AV	MR	EL	DRIVE 0 MODE BYTE
DE	LP	PL	EX	DTE	SME	DI	ZT	DRIVE 0 ERROR BYTE
TM	EOT	BOT	WL	OL	AV	MR	EL	DRIVE 1 MODE BYTE
DE	LP	PL	EX	DTE	SME	DI	ZT	DRIVE 1 ERROR BYTE
TM	EOT	BOT	WL	OL	AV	MR	EL	DRIVE 2 MODE BYTE
DE	LP	PL	EX	DTE	SME	DI	ZT	DRIVE 2 ERROR BYTE
TM	EOT	BOT	WL	OL	AV	MR	EL	DRIVE 3 MODE BYTE
DE	LP	PL	EX	DTE	SME	DI	ZT	DRIVE 3 ERROR BYTE

AF : Formatter Attention Asserted

A3 : Drive 3 Attention Asserted

A2 : Drive 2 Attention Asserted

A1 : Drive 1 Attention Asserted

A0 : Drive 0 Attention Asserted

AV : Drive Available to Formatter

BOT: Beginning of Tape

Cn : Controller Flags (C1 - C8) - currently not implemented

DE : Drive Error - asserted when any drive error not covered by other status bits is detected.

DF : Formatter Diagnostic Failed

DI : Diagnostic Mode - when set, instructs the formatter to use special internal algorithms to report imperfect performance.

D : Diagnostic Requested - asserted when the formatter is requesting permission to execute a diagnostic.

DTE: Data Transfer Error - asserted when any error occurs which prevents a data transfer from completing successfully.

EL : Error Logging Request - asserted by either the drive or formatter when error logging information is available.

EOT: End of Tape - asserted when the tape is positioned at or past the end of tape marker.

EX : Exception Condition - asserted whenever the formatter encounters TM, BOT, or EOT during a data transfer operation or when EL is raised during a data transfer.

FD : Retry Bit - Failure / Direction - is asserted during error recovery to indicate the direction of a retry or to indicate a failing operation. If RP = 0 and RT = 1, then FD = direction to transfer. FD = 0 means transfer in the same direction as original operation; FD = 1 means transfer in the opposite direction of original operation. If RP = 1 and RT = 0, then FD indicates success or failure of operation. FD = 0 means the retry sequence succeeded; FD = 1 means the retry sequence failed.

FE : Formatter Error - asserted on formatter errors not covered by the TE, PE, or DF bits. These errors include fatal errors that may turn on the drive fault indicator.

LP : Lengthy Operation in Progress - asserted when a rewind operation (including the optional data security erase portion of a rewind) is in progress.

MR : Maintenance Mode Request - asserted when the drive is put into maintenance mode. On the TA78, this is accomplished via a thumbwheel switch on the operator panel.

OA : Formatter Online or Available (for the TOPOLOGY command).

OL : Drive Online to Formatter.

PB : Active Port Button - PB = 0 if formatter is connected to the controller through port A; PB = 1 if formatter is connected to the controller through port B.

PE : Level 2 Protocol Error - asserted when a protocol error is detected while processing a Level 2 command.

PL : Position Lost - asserted when the formatter is not certain of the current tape position.

P : Port Switch - asserted when the port switch is enabled.

RP : Request Position - used by the formatter along with RT to inform the controller of the next step in the error recovery sequence.

- Retryable RP = 1, RT = 1
- Transfer RP = 0, RT = 1
- Done RP = 1, RT = 0
- No Error RP = 0, RT = 0

RT : Request Transfer - refer to the explanation for RP.

SME: Speed Management Enabled - asserted whenever the formatter may change the current operating speed of a particular drive at any time (provided the changing of the drive operating speed is transparent to the controller).

TE : Transmission Error - used by the formatter to report Level 0 and Level 1 STI errors. The formatter only reports Level 0 real-time state parity errors and Write/Cmd Data Line pulse errors when a transfer is in progress. Level 1 errors are framing errors, checksum errors, inappropriate value in data field of real-time command, or a real-time command occurring in an invalid context.

TM : Tape Mark

WL : Write Locked

ZT : Zero Threshold - instructs the formatter to change all error thresholds from their default values to zero.

A list of the tape errors and their meaning follows:

NOTE

Always verify proper dc voltage levels if the indicated possible FRUs do not rectify failure.

Acknowledge Not Asserted At Start Of Transfer

Message Error Level: Error

Message Description: The HSC is ready to start a transfer by sending the formatter a Level 1 command and the formatter does not have ACKNOWLEDGE asserted.

Field Service Action: Check the formatter. This error may indicate a formatter STI communications error, or if preceded by tape transport errors, may be a result of a transport failure.

Possible FRUs:

1. Formatter
2. K.sti module
3. STI cable set

Buffer EDC Error

Message Error Level: Error

Message Description: The K.sti detected an EDC error on the data buffer it read from memory on a Write operation.

Field Service Action: Test the data path from tape formatter to HSC data memory.

Possible FRUs:

1. Formatter
2. M.std2 module
3. K.sti module
4. K.ci module

Cannot Clear Formatter Errors

Message Error Level: Error

Message Description: Issued a clear bit three times and cannot clear the error.

Field Service Action: Check the formatter

Possible FRUs:

1. Formatter
2. STI cable set
3. K.sti module

Cannot Clear Drive Errors

Message Error Level: Error

Message Description: Issued a clear bit three times and cannot clear the bit.

Field Service Action: Check the formatter and drive. Further analysis of tape drive error log may be necessary.

Possible FRUs:

1. Drive modules (Refer to drive service manual.)
2. Formatter
3. STI cable set
4. K.sti module

Controller Detected Position Lost

Message Error Level: Error

Message Description: Information contained in the response from the formatter to the HSC POSITION command did not match the expected tape drive position.

Field Service Action: Check the formatter. If the error persists, run the Inline Tape (ILTAPE) diagnostic to help isolate to the FRU.

Possible FRUs: Formatter

Controller Transfer Retry Limit Exceeded

Message Error Level: Error

Message Description: The controller failed to perform the command within the limit of allowable retries.

Field Service Action: Check formatter, drive

Possible FRUs:

1. Drive modules (Refer to drive service manual.)
2. Formatter

Could Not Complete Online Sequence

Message Error Level: Error

Message Description: Could not complete on-line sequence due to a condition in the drive.

Field Service Action: Check the formatter and drive.

Possible FRUs:

1. Drive modules (Refer to drive service manual.)
2. Formatter

Could Not Get Extended Drive Status

Message Error Level: Error

Message Description: Issued the Get Status command and the drive did not respond with the extended drive status.

Field Service Action: Check the formatter.

Possible FRUs: Formatter

Could Not Get Formatter Summary Status During Transfer Error Recovery

Message Error Level: Error

Message Description: Issued the command and the formatter did not respond with the formatter summary.

Field Service Action: Check the formatter.

Possible FRUs: Formatter

Could Not Get Formatter Summary Status While Trying To Restore Tape Position

Message Error Level: Error

Message Description: Issued the command and the formatter did not respond with the formatter summary status.

Field Service Action: Check the formatter.

Possible FRUs: Formatter

Could Not Position For Formatter Retry

Message Error Level: Error

Message Description: The HSC issued a command for data recovery with position required, and the drive could not complete the command.

Field Service Action: Check the media, drive, formatter.

Possible FRUs:

1. Drive modules (Refer to drive service manual.)
2. Media
3. Formatter

Could Not Set Byte Count

Message Error Level: Error

Message Description: Issued command to set byte count and could not complete command.

Field Service Action: Check the formatter.

Possible FRUs: Formatter

Could Not Set Unit Characteristics

Message Error Level: Error

Message Description: Issued command to set unit characteristics and could not complete command.

Field Service Action: Check the formatter.

Possible FRUs: Formatter

Data Ready Timeout

Message Error Level: Error

Message Description: The controller did not detect DATA READY from the formatter within 5 ms after sending it a Level 1 command.

Field Service Action: Check the STI path.

Possible FRUs:

1. STI cable set
2. K.sti module
3. Formatter

Data Overflow Due To Pipeline Error

Message Error Level: Error

Message Description: No data buffers in HSC data memory were available when the K.sti needed one during a data transfer.

Field Service Action: Intermittent errors may indicate excessive error recovery simultaneously occurring elsewhere in the subsystem. Retry operation. Persistent failures may indicate a tape data channel error during a read operation or a K.ci problem during a tape write operation.

Possible FRUs:

1. M.std2 module
2. K.sti module
3. K.ci module

Erase Command Failed

Message Error Level: Error

Message Description: Issued erase command and command failed.

Field Service Action: Check the formatter.

Possible FRUs: Formatter

Erase Gap Command Failed

Message Error Level: Error

Message Description: Issued erase gap command and command failed.

Field Service Action: Check the formatter.

Possible FRUs: Formatter

Formatter And HSC Disagree On Tape Position

Message Error Level: Error

Message Description: The formatter and the HSC disagree on position of the tape.

Field Service Action: Check the formatter.

Possible FRUs:

1. Tape drive module
2. Formatter
3. K.sti module

Formatter Detected Position Lost

Message Error Level: Error

Message Description: The formatter lost track of tape position.

Field Service Action: Check media, drive, formatter.

Possible FRUs:

1. Drive modules (Refer to drive service manual.)
2. Formatter
3. Media

Formatter Requested Error Log

Message Error Level: Error

Message Description: The formatter detected an error and set the EL bit to request an error log be taken.

Field Service Action: Check the formatter.

Possible FRUs: Formatter

Formatter Retry Sequence Exhausted

Message Error Level: Error

Message Description: The formatter failed to complete a command within the retry limit.

Field Service Action: Check the media, drive, formatter.

Possible FRUs:

1. Drive modules (Refer to drive service manual.)
2. Formatter
3. Media

Host Requested Retry Suppression On A Formatter Detected Error

Message Error Level: Error

Message Description: The formatter detected an error and the host issued a command to suppress the retry of the command that failed.

Field Service Action: Check the formatter.

Possible FRUs: Formatter

Host Requested Retry Suppression On A K.sti Detected Error

Message Error Level: Error

Message Description: An error was detected in the K.sti and the host issued a command to suppress the retry of the command that failed.

Field Service Action: Check the K.sti.

Possible FRUs: K.sti module

Lower Processor Error

Message Error Level: Error

Message Description: A bit was set in the lower processor error register. Bits included in the lower processor error register are Data Bus NXM, Data SERDES Overrun, Data Bus Overrun, Data Bus Parity Error, Data Pulse Missing, and Sync Real Time Parity Error.

Field Service Action: Check the K.sti.

Possible FRUs: K.sti module

Lower Processor Timeout

Message Error Level: Error

Message Description: The upper processor in the K.sti detected the lower processor had stopped and restarted it.

Field Service Action: Check the K.sti.

Possible FRUs: K.sti module

Receiver Ready Not Asserted At Start Of Transfer

Message Error Level: Error

Message Description: The HSC is ready to start a transfer by sending the formatter a Level 1 command and the formatter does not have Receiver Ready asserted.

Field Service Action: Check the formatter, cable, K.sti.

Possible FRUs:

1. Formatter
2. Cable
3. K.sti module

Record EDC Error

Message Error Level: Error

Message Description: On a read from tape operation, the EDC calculated by the K.sti did not match the EDC generated by the tape formatter.

Field Service Action: Check the formatter, cable, K.sti.

Possible FRUs:

1. Formatter
2. Cable
3. K.sti module

Retry Limit Exceeded While Attempting To Restore Tape Position

Message Error Level: Error

Message Description: A command was issued to restore the tape position, and the command failed in the limit of retries.

Field Service Action: Check the formatter.

Possible FRUs: Formatter

Reverse Retry Currently Not Supported

Message Error Level: Error

Message Description: Reverse Retry requests from the formatter are currently not supported by HSC.

Field Service Action: None

Possible FRUs: None

Rewind Failure

Message Error Level: Error

Message Description: A command for a rewind was issued, and the command failed (the controller received an unsuccessful response from the formatter).

Field Service Action: Check the drive, formatter.

Possible FRUs:

1. Drive modules (Refer to drive service manual.)
2. Formatter

Tape Drive Requested Error Log

Message Error Level: Warning

Message Description: The drive detected an error condition and set the EL bit for an error log to be taken.

Field Service Action: Check the drive.

Possible FRUs: Drive modules (Refer to drive service manual.)

Topology Command Failed

Message Error Level: Error

Message Description: A topology command was issued and the command failed.

Field Service Action: Check the formatter.

Possible FRUs: Formatter

Unable To Position To Before LEOT

Message Error Level: Error

Message Description: The command to position the tape was issued before LEOT and could not do the command.

Field Service Action: Check the drive.

Possible FRUs: Drive module (Refer to drive service manual.)

Unknown K.tape Error

Message Error Level: Error

Message Description: The ER bit was set but was undefined.

Field Service Action: Check the formatter.

Possible FRUs: Formatter

Word Rate Clock Timeout

Message Error Level: Error

Message Description: The K.sti detected the loss of clocks from a drive during a transfer.

Field Service Action: Check the formatter, cable.

Possible FRUs:

1. Formatter
2. Cable

8.4.5 Out-of-Band Errors

The out-of-band errors are those not conforming to a specific template format, as the MSCP and TMSCP errors do. The method of reporting differs for individual errors.

NOTE

The HSC70 operating software allows the setting of different levels of error reporting for out-of-band type errors using the SETSHO utility. These message error levels are Informational, Warning, Fatal, Error, and Success. The identifiers for the out-of-band errors are followed by an I, W, F, E or S, depending on the SETSHO value. The X in the following list represents the message error level.

Out-of-band errors are further classified into five categories. They are:

1. CI Errors - identified by HOST-X identifier printed prior to message

2. Load Device Errors - identified by SYSDEV-X identifier prior to message

3. Disk Functional Errors - identified by DISK-X identifier prior to message

4. Tape Functional Errors - identified by TAPE-X identifier prior to message

5. Miscellaneous (Software Inconsistencies) - identified by SINI-X identifier prior to message

NOTE

Some out-of-band errors report microcode-detected error status codes within the printout. Refer to Appendix D for a full list of all K.ci, K.sti, and K.sdi microcode-detected errors.

8.4.5.1 CI Errors - The following list shows each CI-detected out-of-band error message and gives the error level, a message description, field service action, and possible FRUs. The messages are displayed in alphabetical order. When replacing indicated FRUs, always verify correct dc voltage levels before and after replacing a module.

Date/Time set by node nn

Message Error Level: Informational

Message description: The HSC received either a START or STACK (Start Acknowledge) message over the CI, and the date and time was not set.

Field Service Action: None. This is a normal message as part of establishing a VC between a host and an HSC.

Possible FRUs: None

VC open with node nn

Message Error Level: Informational

Message Description: A virtual circuit (VC) has been established with the given node. The first time a VC is established to an HSC causes the ONLINE lamp on the HSC operator control panel to light.

Field Service Action: None is required; this message is for informational purposes only.

Possible FRUs: None

Node nn Cables have gone from uncrossed to crossed

Message Error Level: Warning

Message Description: This message occurs when an IDRSP (ID Response) packet is received by an HSC in response to an IDREQ (ID Request) message. Upon receiving an IDRSP packet, the HSC checks two bits in the IDRSP message that indicate which path the sending node used. If these two bits do not indicate the same path the HSC received the message on, this error occurs.

Field Service Action: Determine if the problem is broken hardware in the HSC CI interface, broken hardware in the host CI interface, or if the CI cables are crossed. Before replacing any modules or cables, determine if the HSC is encountering crossed paths to multiple nodes in the cluster or only to a particular node. If the HSC is encountering crossed paths to all nodes, the problem is probably in the HSC or the cables. If it is encountering the problem to only one node, it is likely a problem with that host node's CI module set or the cables running from the host to the star coupler.

Possible FRUs:

1. Cables physically connected wrong at HSC, Star Coupler, or host CI
2. Any of the three K.ci modules in the HSC (L0100, L0109, L0107)
3. Host CI module set
4. Duplicate node address settings

Node nn Cables have gone from crossed to uncrossed

Message Error Level: Error

Message Description: This message occurs only when check for a crossed path finds a previously crossed path no longer crossed. More detail is covered in the description of preceding error message:

Node nn Cables have gone from uncrossed to crossed

Field Service Action: Note, if both the "uncrossed to crossed" and "crossed to uncrossed" messages are occurring, it is most likely an indication of failing hardware, not a cable problem. See the **Field Service Action** for the previous message for more detail.

Possible FRUs:

1. CI cables, if a single message is displayed
2. K.ci module set, if both messages are displayed

Node nn Path (A or B) has gone from good to bad

Message Error Level: Warning

Message Description: K.ci microcode detects a hard (nonrecoverable) transmission error on a previously good path. Examples of hard transmission errors are:

1. Transmit Buffer Parity Error
2. Unrecoverable NAK
3. Unrecoverable NORSP
4. Transmitter Attention Timeout

Determining the reason for failure using the error message is not possible.

Field Service Action: Before replacing any FRU, determine if the message is occurring because of problems with one host or problems with multiple hosts. If the problem involves one host, it is most likely in the star coupler's host side. If the problem involves multiple hosts, it is most likely on the star coupler's HSC side. Also, if the message occurs on both paths to a host, that host may have been powered down, stopped, or may have crashed. Examine the host console log and the error log to determine if something did happen to the host.

Determining which error caused the bad path is not possible except with the Transmit Buffer Parity Error (XBUF PE) which prints as an MSCP type message.

Possible FRUs:

1. CI cable
2. Host
3. CI interface hardware in the host

Node nn Path n has gone from bad to good

Message Error Level: Warning

Message Description:

A disconnected CI cable has been reconnected, or an intermittent hardware or cable problem is indicated. More detail is found in the description of previous error message:

Node nn Path (A or B) has gone from good to bad

This message also occurs if an open VC node path was previously found to be bad. During this polling cycle the node sends out IDREQ (ID Request) packets to all nodes and receives successful IDRSP ID Response messages.

Field Service Action: If the cable was reconnected, there is no further action. Otherwise, replace the possible FRUs.

Possible FRUs:

1. CI cable
2. Host
3. CI interface hardware in the host

K.ci exception detected, code = nnn

Message Error Level: Warning

Message Description: The code here is the contents of KH$FLG (the second word in the K.ci control area). Below is a breakdown of the bits contained in this word:

1. 000001 KHF$PD - Path(s) disabled by K.ci due to Xmit error or VC breakage due to other K.ci-detected error.

2. 000002 KHF$EQ - Item(s) placed on error queue (KH$EQ).

3. 000004 KHF$BL - Data memory error during BMB list operation.

4. 000010 KHF$UP - Unreceivable packet. K.ci stopped (causes a crash).

5. 000100 KHF$NH - Sequenced message received while reserved-to-receive queue was empty.

6. 040000 KHF$PD - Set by diagnostics to disable interrupts.

Field Service Action: Compare the code from the printout to the previous list, and determine whether the error code points to an HSC70 module or to the host.

Possible FRUs:

1. Status 1: K.pli module
2. Status 4: M.std2 module
3. Status 10: PILA module, Host K.ci set
4. Status 100: Host K.ci set

VC closed with node nn due to unexpected disconnect

Message Error Level: Warning

Message Description: The HSC receives a DISCONNECTREQ packet, and the following conditions exist inside the HSC.

- o A connection is not open.
- o The HSC is not in the DISCONNECTSENT state. (The DISCONNECTSENT state indicates the HSC also sent a DISCONNECTREQ packet.)

Field Service Action: Verify no other nodes in the cluster failed and caused sending an unexpected disconnect to the HSC. If failure persists, the K.ci module set may be causing this error. Run Offline Test K diagnostic to test K.ci. If no failure, verify no duplicate node addresses exist in this cluster (L0100 node address switches).

Possible FRUs: K.pli module.

VC closed with node nn due to disconnect timeout

Message Error Level: Warning

Message Description: A second disconnect call for the same connection block has been received by the CI manager.

Field Service Action: Verify other cluster nodes have not failed or have CI port problems. If the problem persists, run Offline Test K diagnostic to test K.ci. If no failures exist, verify Set parameters are valid, use backup copy of the HSC code and replace FRUs indicated.

Possible FRUs: Host K.ci module set

VC closed with node nn due to request from K.ci

Message Error Level: Warning

Message Description: The K.ci microcode has detected both CI paths have gone from good to bad during polling. More details are found under the description for error message:

Node nn Path n has gone from good to bad

Field Service Action: See the descriptions and field service action for the following error messages:

- Node nn Path (A or B) has gone from good to bad
- Node nn Path (A or B) has gone from bad to good

Possible FRUs:

1. K.ci hardware interface in HSC
2. CI cables

VC closed with node nn due to START received

Message Error Level: Warning

Message Description: A START message is received over the CI to an already open virtual circuit (VC).

Field Service Action: Check for two HSC70s with the same ID (not node address) on the cluster. This happens when new HSC70 is installed on the cluster and is given existing ID.

Possible FRUs: CI cables

No control block available to satisfy HMB request.

Message Error Level: Warning

Message Description: The CIMGR tried to allocate an HMB (Host Memory Block) from the free control block queue when none were available. If a significant amount of control memory was removed from use due to errors detected during boot, this message occurs. Otherwise, it may indicate an internal HSC software problem where control blocks in HSC memory are taken by some service and never returned to the list of free control blocks.

Field Service Action: Type in the SHOW MEMORY command for HSC50 software version V300 and later and HSC70 software version V100 and later to determine how much control memory is being used. Compare the amount of control memory shown on the SHOW MEMORY printout to the amount contained in the HSC. If more than 10% has been disabled from use, replace the memory module. For HSC50 software before V300, run the offline memory test on control memory to determine if excessive solid failures are causing removal of a large amount of memory. If memory amount is adequate, the problem may be caused by a software or microcode problem within the HSC.

Possible FRUs:

1. M.std2 module
2. Software

HML$ER set - HM$ERR = nn

Message Error Level: Warning

Message Description: A HMB (host memory block) operation resulted in an error. A breakdown of HMB error word (HM$ERR) bits follow:

1. 000002 HME$BM - Insufficient BMBs to receive message.

2. 000004 HME$NC - Sequenced message received over a connection with "0" in credit field.

3. 000010 HME$NC - Sequenced message received over a connection with credit field >"1". Excess has been added to CB$EM.

4. 000020 HME$OV - Oversize message received (>1096. bytes).

5. 000040 HME$DN - Data memory NXM during BMB operation.

6. 000100 HME$DP - Data memory parity error in BMB operation.

7. 000200 HME$DO - Data memory overrun during BMB operation.

8. 000400 HME$FP - Reception buffer parity error in packet "header". Message not receivable.

9. 001000 HME$PL - Reception buffer parity error in "body" of message.

10. 002000 HME$CN - Transmission not attempted because connection not valid.

11. 004000 HME$VC - Transmission not attempted because VC closed or connection invalid.

12. 010000 HME$TE - Transmission attempted but failed (no ACK).

13. 020000 HME$TP - Transmission failed due to transmission buffer parity error.

14. 040000 HME$HC - Packet inconsistent with K.ci context received from host.

15. 100000 HME$IC - Illegal control function opcode.

Field Service Action: Compare the displayed code to the previous list and determine where the problem lies. For example, a code of 000040 indicates a failure in the M.std2 module, and a code of 002000 indicates a problem in the K.ci module set.

Possible FRUs:

1. PILA module
2. K.pli module
3. M.std2 module

Bad dispatch state in CB...

Message Error Level: Warning

Message Description: The CI manager sends a SCS control message and finds an invalid dispatch state in the control block. The CI manager then uses the dispatch state to determine where to send the proper control message. If this is the only known problem, a software problem could exist within the HSC. Otherwise, the problem could be caused by a Control Bus addressing problem with the K.pli, M.std2, or P.ioj modules.

Field Service Action: Replace the following FRUs.

Possible FRUs:

1. K.pli
2. M.std2
3. P.ioj

K.ci loopback microcode loaded

Message Error Level: Error

Message Description: The CIMGR detected K.ci loopback microcode was loaded during initialization. When this message occurs a problem with the K.pli (L0107) module most likely exists.

Field Service Action: Replace the following FRUs.

Possible FRUs: K.pli module

Resource lost to K.ci -- xxx xxx HMBs

Message Error Level: Error

Message Description: A control memory HMB (host message block) data structure was lost. HMBs were expected in the sequence message ready to receive queue (.KHSRR), but none were found.

Field Service Action: Report the error, with frequency of occurrence, to support. Also, note sequence of events that reproduce this failure. This message indicates a software bug. Verify dc power levels are correct.

Possible FRUs:

1. Software
2. Dc power

8.4.5.2 Load Device Errors - Detected errors from the RX33 load device are classified into the out-of-band error category. The following is an example printout of a detected RX33 error.

```
SYSDEV-S Seq 104. at  6-JAN-1986 10:12:00.76
        DX1: LBN 1488. (49,0,02), Status 001
        Seek 000, 000000
        Tran 003, 021404
        T.O. 000
        87 3 1485 -7680 1 49 1 4
```

The -S following the SYSDEV prompt and before the Seq. number indicates the severity level. The RX33 has three severity levels:

1. Success (S) : two or less errors during a command/retry
2. Informational (I) : more than two errors
3. Error (E) : unrecoverable error

The status field is most important and is a direct indication of the error. Following is a list of the RX33 status codes:

- 000 :success
- 001 :success with retries
- 002 :S/W version mismatch (driver vs. operating code)
- 200 :command aborted via a CTRL/Y or exception operation
- 201 :illegal file name
- 202 :file not found
- 203 :file is not in a loadable image format
- 204 :insufficient memory to load image
- 205 :no free partition to load image into
- 206 :unit is S/W disabled
- 365 :unit is write protected
- 367 :no media mounted
- 375 :EOF detected during read or write
- 376 :hard disk error, other than the following:
  - 370 :bad unit number
  - 357 :data check error
  - 343 :motor broken (would not spin up)
  - 340 :Uncorrectable seek error (desired cylinder not found)
  - 311 :bad record (LBN) number (not on media)
  - 272 :parity error in controller on M.std2 module

The failing floppy disk drive is indicated by DX1:. The logical block number where the failure occurred is displayed by LBN 1488. The three numbers in parentheses, separated by commas after the

logical block number, indicate in order they are shown, the cylinder, the media surface, and the drive sector.

The Seek entry's first group of zeros shows the retry count for seek/recal errors or the number of times the command was issued but not completed. The second group of zeros shows an inclusive OR of the control and status registers CSR bits set during seek error retries. The important bit in a seek error is bit 4.

The Tran (transfers) entry's first group of zeros shows the retry count for read, write, and format errors, or the number of times the command was issued and not completed. The second group of zeros shows an inclusive OR of the CSR bits set during read, write, and format error retries. A breakdown of the upper CSR bits is shown in Figure 8-10. The status of the lower CSR bits is shown in Table 8-24.

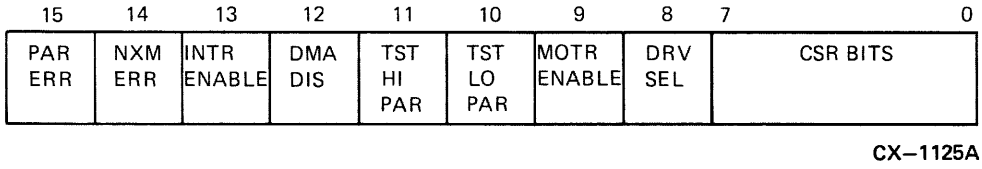


Figure 8-10 RX33 Floppy Controller CSR Breakdown

Table 8-24 Status Register Summary

BIT	ALL TYPE I COMMANDS	READ ADDRESS	READ SECTOR	READ TRACK	WRITE SECTOR	WRITE TRACK
S7	Not Ready	Not Ready	Not ready	Not Ready	Not Ready	Not Ready
S6	Write Protect	0	0	0	Write Protect	Write Protect
S5	Head Loaded	0	Record Type	0	0	0
S4	Seek Error	RNF	RNF	0	RNF	0
S3	CRC Error	CRC Error	CRC Error	0	CRC Error	0
S2	Track 0	Lost Data	Lost Data	Lost Data	Lost Data	Lost Data
S1	Index Pulse	DRQ	DRQ	DRQ	DRQ	DRQ
S0	Busy	Busy	Busy	Busy	Busy	Busy

The T.O. entry line is a timeout recording for each command type. This counter reflects the total number of timeouts for the command in error. All commands (read, write, recal, spinup, and format track) time out in one second.

The last line in the error message is more complicated to breakdown. The breakdown of the last line is as follows:

```
87  3  1485  -7680  1  49  1  4

|   |    |       |   |   |  |  |__sector number
|   |    |       |   |   |  |___surface number
|   |    |       |   |   |_____cylinder number
|   |    |       |   |_______unit number
|   |    |       |_________byte count negative implies write)
|   |    |_____________LBN
|   |_________________success count
|_____________________err count number
```

Most information in the error printout is reiterated in the last line. Starting from the right, sector, surface, cylinder number, and unit number are displayed as in the main body of the error

message. The byte count has an indicator for write and read commands; the negative indicates a write operation. The LBN in this field is the starting LBN for this transfer. The LBN in the main message body is the failing LBN. The success count and error count are for informational purposes.

8.4.5.3 Disk Functional Errors - Although most disk drive related errors are MSCP errors, several disk functional errors fall into the out-of-band error category. They are identified by the DISK-E identifier printed on the terminal display prior to the error.

The message, message description, field service action, and probable FRUs for the disk functional out-of-band errors follow in alphabetical order.

Aborting Error Recovery Due to Excessive RECALS
Disk Unit xx
Requestor xx
Port xx

Message Error Level: Error

Message Description: For each transfer, a counter detects the number of recals attempted. If the count exceeds number of recals attempted, this message is printed. Recovery from an error is not possible because of excessive recals.

Field Service Action: Refer to drive service manual to determine reasons for persistent positioning failures.

Possible FRUs: Drive unit

Aborting Error Recovery Due to Excessive Timeouts

Message Error Level: Error

Message Description: The HSC detects several timeouts on the disk drive. All error recovery attempts will be aborted.

Field Service Action: Replace the following FRUs. Further testing may be necessary.

Possible FRUs:

1. Drive module (Refer to drive service manual.)
2. K.sdi module

Attention Condition serviced for ONLINE disk unit xxx

Message Error Level: Information

Message Description: A condition change in the drive needs servicing. A Get Status exchange is invoked to the drive.

Field Service Action: Refer to the console printed Get Status response.

Possible FRUs: Drive modules (Refer to drive service manual.)

ATN. message sent to Node xx, for Unit xx

Message Error Level: Information

Message Description: The attention message has been sent. This message corresponds to the previous message.

Field Service Action: None

Possible FRUs: None

Clock dropout from ONLINE disk unit xx

Message Error Level: Error

Message Description: The online disk has lost its real-time state clock.

Field Service Action: Check the path between the K.sdi and the disk drive that was reported. Determine if the problem is in the HSC or the disk drive. Other disk error reports may precede this message and provide more detail about this error condition.

Possible FRUs:

1. Drive modules (Refer to drive service manual.)
2. SI cable
3. K.sdi module

Deferred ATN. message for Node xx, Unit xx

Message Error Level: Information

Message Description: A attention message is delayed in process.

Field Service Action: None

Possible FRUs: None

Disk unit xx ready to transfer.!
Retrieval failure or subsystem deadlock probable.

Message Error Level: Information

Message Description: Necessary resources would not do the transfer.

1. Out of buffers
2. K.sdi ready to die

Field Service Action: Check data transfer path. This error may indicate too many utilities or inline diagnostics running simultaneously. The problem might also be an HSC software problem.

Possible FRUs: K.sdi

Disk Unit xx (Requestor xx, Port xx) being initialized
DCB addr: xxxxxx

Message Error Level: Information

Message Description: A disk is being initialized and identified.

Field Service Action: None

Possible FRUs: None

Disk unit xxx.(Requestor xx.,Port xx.) declared inoperative intervention required.

Message Error Level: Error

Message Description: The K.sdi sent a nondata transfer command over to the disk three times and received the same error back three times. The HSC ignores the disk until it detects some intervention. An example is to deport the port button to drop the state clock.

Field Service Action: Examine previous error reports to help resolve failure. Toggle port switch on drive.

Possible FRUs: Drive modules (Refer to drive service manual.)

DRAT/SEEK timeout, disk unit xxx

Message Error Level: Information

Message Description: A stimulus resulting in error recovery code action is the expiration of the DRAT/SEEK timer for the drive. A DRAT represents data transfer action with the drive, whereas the SEEK timer represents position requests to the drive.

Each drive has a timer (set to three times the SDI drive short timeout value) allocated on its behalf at subsystem initialization time. This timer, called the DRAT/SEEK timer, is active whenever data transfer activity to the drive is outstanding.

When the disk transfer code queues transfer work to K.sdi on behalf of a previously idle drive, the timer starts. When it adds transfer work to a drive that already has transfer work, the timer restarts. When it detects the completion of the last DRAT queued to the drive, the timer stops. Thus, the timer is running only as long as transfer work is outstanding. A timer may expire for several reasons:

1. The drive has detected a drive error and has lowered Read/Write Ready.

2. The drive has stopped sending clock signals.

3. Another element in the subsystem that should have supplied resources to the disk transfer operation in a reasonable time did not.

Field Service Action: Check out the drive.

Possible FRUs: Drive modules (Refer to drive service manual.)

DRIVE CLEAR attempt on disk unit xx (Requestor xx, Port xx)
DCB addr: xxxxxx Error count xxxxxx

Message Error Level: Information

Message Description: The drive had some previous error and now is attempting to clear that error.

Field Service Action: Examine the host error log to determine what error the drive is trying to clear.

Possible FRUs: Drive

Duplicate disk unit xx

Message Error Level: Information

Message Description: Disk unit numbers are duplicated within the system.

Field Service Action: Locate the duplicate disks and change the plug number on one.

Possible FRUs: Drive modules (Refer to drive service manual.)

FRB error: K.ci, lst LBN xx buffers, FE$SUM xx

Message Error Level: Information

Message Description: A fragment request block arrives to the error process. Example: A Revector.

Field Service Action: If excessive, reformat drive.

Possible FRUs: Drive modules (Refer to drive service manual.)

FRB error: K.sdi, Unit xx, first LBN xxx, buffers, FE$SUM

Message Error Level: Information

Message Description: A fragment request block arrives to the error process. Example: A Revector.

Field Service Action: If excessive, reformat drive.

Possible FRUs: Drive modules (Refer to drive service manual.)

Illegal bit change in status from disk unit xxx
EL bit forced on so status logged.

Message Error Level: Error

Message Description: An unsupported bit was received in status returned from disk unit.

Field Service Action: Check drive and version of software in HSC.

Possible FRUs:

1. Drive module (Refer to drive service manual.)
2. Version of software.

K.sdi in slot xx failed its init diagnostics, status = xxx

Message Error Level: Error

Message Description: A requestor fails during boot. The displayed K.sdi has failed with the displayed status. This message is only displayed at the end of the boot procedure.

Field Service Action: Record the status for module repair purposes.

Possible FRUs: The K.sdi displayed

LBN Restored with Forced Error in RESTOR Operation!
Disk Unit xx LBN xx
Tape Unit xx

Message Error Level: Warning

Message Description: An error was detected in the LBN data during backup. A forced error bit was set in the LBN.

Field Service Action: If excessive, reformat drive.

Possible FRUs: Drive modules (Refer to drive service manual.)

Positioner error on disk unit xxx. DRAT addr:xxx
Desired hdr (lo,hi):xxx xxx
Actual hdr (lo,hi):xxx xxx

Message Error Level: Information

Message Description: The drive positioned the heads in the wrong place.

Field Service Action: Check drive modules and K.sdi module.

Possible FRUs:

1. Drive modules (Refer to drive service manual.)
2. K.sdi module

Premature LP flag in RTNDAT sequence from host node xx

Message Error Level: Warning

Message Description: A violation of packet protocol; the last packet flag was set before all data was received from a host.

Field Service Action: If the problem is transient, monitor error for repetitive node numbers as this may indicate a host CI problem. If problem is persistent across all cluster nodes, test the K.ci.

Possible FRUs:

1. K.ci modules
2. CI cables

SDI exchange retry on disk unit xxx (Requestor xx Port xx) DCB addr xx Error count xx

Message Error Level: Information

Message Description: Retry the SDI command on the drive.

Field Service Action: None

Possible FRUs: None

Unexpected AVAILABLE signal from ONLINE disk unit xx

Message Error Level: Information

Message Description: The HSC believes the disk is already online; therefore the disk should not be asserting available.

Field Service Action: Determine why the disk drive is asserting the Available signal.

Possible FRUs: Drive modules (Refer to drive service manual.)

Unrecoverable error on disk unit xx. Drive appears inoperative intervention required.

Message Error Level: Error

Message Description: An error log message from the drive caused this message, or the drive may be offline.

Field Service Action: Check error log and drive.

Possible FRUs: Drive modules (Refer to drive service manual.)

Unsuccessful SEEK initiation, disk unit xxx. DCB addr: xxx

Message Error Level: Information

Message Description: The dialog control block sent the seek exchange, and it was rejected or lost.

Field Service Action: Check drive.

Possible FRUs: Drive modules (Refer to the drive service manual.)

VC closed due to timeout of RTNDAT/CNT from host node xx

Message Error Level: Information

Message Description: The host issued a request over the CI, and the response timed out.

Field Service Action: Determine if the problem lies in the HSC K.ci module set or the host CI module.

Possible FRUs:

1. K.ci module set in the HSC
2. CI module set in the host

8.4.5.4 Tape Functional Errors - Although most tape errors are covered under TMSCP errors, certain tape functional errors are classified in the out-of-band error category. They are identified by the TAPE-E identifier printed prior to the error printout on the local console terminal.

The following shows each tape functional detected out-of-band error message, a message description, field service action, and probable FRUs.

Data Error Flagged in Backup Record
Disk Unit xx LBN xx
Tape unit xx

Message Error Level: Warning

Message Description: During a backup, a data error was encountered. During the BBR, the record was written with a forced error bit set.

Field Service Action: Check BBR history on source drive.

Possible FRUs:

1. Disk unit
2. Media

Insufficient Control Memory for K.sti in Requestor xx

Message Error Level: Error

Message Description: Not enough Control Memory left in pool to allocate a control block. A certain amount of Control Memory is needed to set up control blocks. Enough memory has not been found to set up control blocks to turn the K.sti functional code on.

Field Service Action: Use HSC SETSHO utility to show available HSC memory (control, data, and program). If less than 87.5% of available control memory is usable, replace M.std2 module. Run Offline TEST MEM by K diagnostic and test control memory.

Possible FRUs:

1. M.std2 module
2. P.ioj module
3. Software

Insufficient Private Memory remaining for TMSCP Server

Message Error Level: Error

Message Description: In the SCT, a parameter determines the maximum number of supported tape formatters. During initialization, all the working K.sti modules are counted and a calculation is done showing the maximum number of possible formatters. These two parameters are compared. Based on the comparison, a certain amount of Private memory is allocated for the TMSCP Server. If that allocated portion of Private memory is not enough, this message is displayed.

Field Service Action: Use HSC SETSHO utility to show available HSC program memory. If less than 87.5% of available program memory is usable, replace M.std2. Run Offline TEST MEM or TEST REFRESH to test program memory.

Possible FRUs:

1. M.std2 module
2. P.ioj module
3. Software

K.sti in Requestor xx has microcode incompatible with this TMSCP Server

Message Error Level: Error

Message Description: The data structure version within the microcode version residing on the K.sti module is a lower version than the TMSCP Server can support.

Field Service Action: Ensure the version of microcode on the K.sti module is up to current revision. If not, replace the microcode or replace the K.sti module with a K.sti module of the current revision.

Possible FRUs: K.sti module

No Tape Drive Structures available for Requestor xx Port xx Unit xx Increase Structures via SET MAXTAPE command

Message Error Level: Error

Message Description: An additional tape drive has been added to an existing tape formatter, but the tape structures set up in initialization have been exceeded.

Field Service Action: Use the SET/SHO utility to increase to the number of tape structures with the SET MAXTAPE command.

Possible FRUs: None

No Tape Formatter Structures available for Requestor xx Port xx Increase structures via SET MAXFORMATTERS command

Message Error Level: Error

Message Description: An additional tape formatter has been added to the HSC70, but enough Tape Formatter Structures are not available to service this additional tape formatter. Tape Formatter Structures are set up during initialization.

Field Service Action: Use the SET/SHO utility to set the structure level higher to compensate for the additional tape formatter with the SET MAXFORMATTERS command.

Possible FRUs: None

No usable K.sti boards were found by the TMSCP Server

Message Error Level: Error

Message Description: The TMSCP server polled the HSC and found no working K.sti modules. This message does not appear frequently because the K.sti normally fails its initialization diagnostics and displays the error message.

Field Service Action: Check for a failed initialization diagnostic error message prior to this message. This prior message displays the failed requestor slot and failing status.

Possible FRUs: The K.sti(s) displaying the failing status is the FRU.

Requestor xx has failed initialization diagnostics with status = xx

Message Error Level: Error

Message Description: The requestor in slot xx has failed initialization diagnostics with the displayed status. The message indicates the failed K.sti module.

Field Service Action: Refer to the section on status codes to determine what the displayed status indicates the failure to be.

Possible FRUs: K.sti module in the indicated slot

Tape unit number xx connected to Requestor xx Port xx Ceased to exist while Online

Message Error Level: Error

Message Description: This message is similar to the previous error message except in this case, the HSC70 was using the tape drive to do data transfers when the tape drive went Offline.

Field Service Action: Check to see if a breaker has blown. The tape drive may be diagnostic mode also making the tape drive go Offline.

Possible FRUs:

1. Tape drive
2. Tape formatter
3. SI cable

Tape unit number xx connected to Requestor xx Port xx Dropped state clock while Online

Message Error Level: Error

Message Description: The formatter supplies the state clock over the SI cable. The state bits are encoded on this state clock waveform such as AVAILABLE and ATTENTION. As long as the K.sti is receiving a state clock, the SI cable must still be plugged in, and the formatter must be operating correctly. Dropping state clock is equivalent to disconnecting the SI cable from the HSC70.

Field Service Action: First isolate the problem to the HSC70, SI cable or tape unit. Next, try replacing or swapping the K.sti module exhibiting the failure. If the problem is not solved, try a known good tape unit.

Possible FRUs:

1. SI cable
2. Tape unit
3. K.sti module

Tape Formatter connected to Requestor xx Port xx Has been declared Inoperative. Intervention required

Message Error Level: Error

Message Description: The K.sti has sent a nondata transfer command over the SI cable to the displayed tape formatter three times and has received back the same error three times. The HSC70 then ignores the tape formatter until it detects some intervention such as a change in the state clock.

Field Service Action: Replace the possible FRUs. Deasserting the tape drives port switches, recycling power, unplugging the SI cable or any action causing the state clock to come and go is considered an intervention. The HSC70 will not attempt to communicate with the failing tape formatter until it detects this change in state clock. Examine any previous error reports for more specific data regarding this error message.

Possible FRUs:

1. Tape formatter
2. STI cabling
3. K.sti module

Tape unit number xx connected to Requestor xx Port xx Is not asserting Available when it should be

Message Error Level: Error

Message Description: The formatter is not online and is not asserting its Available signal to the HSC70. The HSC70 does not detect the Available signal and displays this message on the local console terminal.

Field Service Action: First isolate the problem to either the HSC70, the SI cable, or the tape unit. Next, try replacing or swapping the K.sti module exhibiting the failure. If the problem is not solved, try a known good tape unit.

Possible FRUs:

1. SI cable
2. Tape unit
3. K.sti module

Tape unit number xx connected to Requestor xx Port xx Went Available without request

Message Error Level: Error

Message Description: When the formatter is online, Available is not normally asserted to the HSC70. When the formatter is online and doing I/O and an Available is asserted, the HSC70 detects this as an error. A formatter does not need to send Available unless the K.sti requests it.

Field Service Action: First isolate the error to the formatter or to the active K.sti.

Possible FRUs:

1. K.sti
2. Formatter
3. SI cable

Tape unit number xx connected to Requestor xx Port xx Went Offline without request

Message Error Level: Error

Message Description: The formatter lost contact with one of the tape drives. The HSC70 detected this loss of a tape drive and printed this message.

Field Service Action: Check to see if breaker has blown. The tape drive may be in diagnostic mode also making the tape drive go Offline.

Possible FRUs:

1. Tape drive
2. Tape formatter
3. SI cable

TMSCP fatal initialization error - TMSCP functionality not available

Message Error Level: Error

Message Description: Something went wrong during initialization with the tape functional code (TFUNC). A routine was called up to initialize some part of the functional code, and that part failed to initialize. Typically, some other message is displayed prior to this message giving more detail on the error.

Field Service Action: Take action depending on the previous message.

Possible FRUs: Dependent on the previously displayed error message.

TMSCP Server operation limited by insufficient Private Memory Use the SET MAX command to reduce Private Memory requirements.

Message Error Level: Error

Message Description: This message appears before the message

Insufficient Private Memory remaining for TMSCP server

and indicates the same problem. Private memory has insufficient space to hold the necessary structures the TMSCP Server needs as dictated by the number of K.sti modules and the number of tape formatters on the HSC70.

Field Service Action: Use HSC SETSHO utility to decrease maximum number of tape formatters for which the HSC should reserve memory structures.

Possible FRUs:

1. M.std2
2. P.ioj
3. Software

TTRASH fatal initialization error.

Message Error Level: Error

Message Description: This message is similar to the message,

TMSCP fatal initialization error - TMSCP functionality not available

except the process failing to initialize is TTRASH instead of the tape functional process (TFUNCT).

Field Service Action: Check for previous error reports displaying a more specific reason for this error report. If earlier error messages do not exist, reboot HSC using backup HSC software copy.

Possible FRUs:

1. M.std2 module
2. Software

WARNING K.sti microcode too low for large transfers.

Message Error Level: Warning

Message Description: The amount of microcode I/O the K.sti can accommodate is restricted. The code still attempts to do transfers, but a warning has been issued.

Field Service Action: Check the microcode version level to ensure the proper revision.

Possible FRUs: Change the level of K.sti microcode to a supported version, or change the K.sti with the out-of-date code.

8.4.5.5 **Miscellaneous Errors** - Miscellaneous errors are identified by the SINI-E identifier printed on the local console terminal. Many of these messages are one or two line messages, but some have several lines of informational text and result from subsystem exceptions. Subsystem exceptions detect inconsistencies in the operating software. These SINI errors are discussed in more detail in this section.

The following describes each message text, gives field service action, and lists the probable FRUs associated with the SINI out-of-band errors.

Booted from drive 1. Drive 0 Error (text)

Message Error Level: Informational

Message Description: The System diskette was booted from Drive 1 of the RX33. Normal boot is from Drive 0.

Field Service Action: None

Possible FRUs: RX33 Drive 0

Cache disabled due to failure

Message Error Level: Error

Message Description: SINI looks back at the Cache diagnostic and senses the Cache is disabled due to cache failure or manually disabled in the diagnostic. This error also shows as a soft fault code on the OCP.

Field Service Action: Load the Offline Cache diagnostic and answer the prompt asking to disable or enable Cache with an enable. Reboot the System diskette and check if the original message is displayed again.

Possible FRUs:

1. P.ioj module
2. M.std2 module

Hard transfer error loading (file) xx

Message Error Level: Error

Message Description: The P.ioj detected a hard error while loading a file from the System diskette into Program memory. The particular files that can produce this error are DUP and MIRROR.The xx field is the error status value from the device driver.

Field Service Action: Load the file from the other disk drive; load the back-up diskette.

Possible FRUs:

1. Diskette
2. RX33

Hard transfer error writing SCT xx

Message Error Level: Error

Message Description: The HSC detected an error while attempting to write the SCT. The xx designates the octal byte that is the error status value returned from the device driver.

Field Service Action: Make sure the drive is not write protected; try the back-up diskette; try the other disk drive.

Possible FRUs:

1. Diskette
2. RX33

Host Clear from CI node

Message Error Level: Error

Message Description: The host cannot function with the HSC70 for some reason such as a nonresponse within a certain amount of time or too many errors on the CI.

Field Service Action: Check the HSC70 console messages and the error logs of the systems connected to the HSC70.

Possible FRUs:

1. HSC70
2. HSC70 operating software
3. System software

Host interface (K.ci) failed INIT diags, status = xxx

Message Error Level: Error

Message Description: The failing status indicates which module in the K.ci set has failed. A soft fault code is generated and may be examined by pressing the Fault button on the OCP.

Field Service Action: Determine which is the failing module by comparing the failing status value to the values in Appendix D. This comparison will point more directly to the failing module.

Possible FRUs:

1. Link module
2. PILA module
3. K.pli module

Host interface (K.ci) is required but not present

Message Error Level: Error

Message Description: A K.ci module set is absent, or the failure in the K.ci module set was so severe upon initialization, the initialization diagnostics did not run.

Field Service Action: Check for the presence of a K.ci module set. If missing, install the K.ci module set. If K.ci module set is present, determine which module is failing by running Offline diagnostics. This error generates a soft fault and is examined by pressing the Fault button on the OCP.

Possible FRUs: See list below and next error message (**Last soft init resulted from unknown cause**)

1. K.pli module
2. K.ci module set (any one of the three modules in the set)

Last soft init resulted from unknown cause

Message Error Level: Error

Message Description: Software has a list of known reasons for reboot (Trap thru 134, Trap thru 250, CRASH$, SET/SHO, etc.). If no reason for reboot is apparent, the software may have failed to detect where the error came from.

Field Service Action: Check the HSC70 console error messages and the system error logs on all the systems connected to the HSC70. This error indicates a probable software problem.

Possible FRUs: Dependent upon the information obtained from the error logs.

Less than 87.5% of program memory is available
Less than 87.5% of control memory is available
Less than 87.5% of data memory is available.

Message Error Level: Error

Message Description: These three messages are a result of the P.ioj polling the memories on initialization and finding an insufficient amount of working memory in either one. Any combination of the three messages may appear.

Field Service Action: The error printout determines which memory is failing.

Possible FRUs: M.std2 module

P.ioj running with memory bank or board swap enabled

Message Error Level: Error

Message Description: Upon initialization an error was detected in the low address space of private memory. The P.ioj asserted the SWAP BANK signal, and the second bank of private memory was enabled. The P.ioj and memory combination can still function under limited capabilities.

Field Service Action: Exchange the M.std2 module. The HSC70 still functions with limited capabilities.

Possible FRUs: M.std2 module

Requestor xx failed INIT diags, status = xxx

Message Error Level: Error

Message Description: The data channel in the displayed requestor has failed initialization diagnostics with the displayed status.

Field Service Action: Determine which data channel is in the displayed requestor slot. Make note of the status value for module repair. Replace the failing data channel.

Possible FRUs: The data channel (K.sdi or K.sti) exhibiting the failing status

SCT read or verification error. Using template SCT.

Message Error Level: Error

Message Description: An error was detected by the P.ioj as it attempted to read the System Configuration Table (SCT) or as it attempted to verify the SCT. The reason this error message occurred because a new, previously uninitialized system diskette was booted. The default settings from SYSCOM are used instead of the SCT from the load media. The second sentence in this message indicates the SCT is new as derived from the template SCT settings set in the factory.

Field Service Action: Reinstall the old system diskette and do a SHO SYSTEM. Install the new diskette exhibiting the error and set all system diskette fields to the old values using the SET command. Reboot the HSC70 to validate these values and ensure system continuity.

Possible FRUs: System diskette

The following alphabetical list of the SINI out-of-band errors consist of informational text. These SINI errors result from subsystem exceptions. A detected inconsistency in the operating software causes a subsystem exception and results in an HSC crash.

Level 7 K interrupt (Trap thru 134)

process yyy
PC xxx
Status xxx xxx xxx xxx xxx xxx xxx xxx xxx

Message Error Level: Error

Message Description: A level 7 K interrupt occurs when one or more requestors detect a fatal error condition while executing functional code. The requestor, upon detecting the error, generates a level 7 K interrupt to the P.ioj. The P.ioj traps through location 134 causing a reboot. The requestor status and the failing requestors status value are displayed for all requestors on the last line of the printout.

Field Service Action: In some cases, the error printout shows a failing requestor when the real problem is in the M.std2 module.

Wait for two or more failures of this type to determine if the real problem is the M.std2 module. If the M.std2 is at fault, the same requestor is not displayed twice as the failing requestor. Refer to Appendix D for failing status values and their meanings. Check the status line message to determine the failing requestor status. Change the requestor exhibiting the failing status if the same requestor is displayed more than once.

Possible FRUs:

1. Requestor displaying a continuous failing status value
2. M.std2 module

MMU (Trap thru 250)

Process yyy
PC xxxxxx
PSW xxxxxx
MMSR0 xxxxxx
MMSR1 xxxxxx
MMSR2 xxxxxx

Message Error Level: Error

Message Description: A failure was detected in the memory management unit on the P.ioj. The active process is displayed as well as the bit assignments for the memory management status registers.

Field Service Action: Examine the MMSR registers to determine the failure in the MMU.

Possible FRUs: P.ioj module.

NXM (Trap thru 4)
process yyy
PC xxx
PSW xxx
Low err reg xxx
Hi err reg xxx
WBUSR xxx

Message Error Level: Error

Message Description: For the J-11:

- o A memory location did not respond within the specified timeout period.
- o A stack overflow occurred.
- o An odd address access was attempted for example, a byte access instead of a word.
- o A halt was executed in user mode.

Field Service Action: Determine which memory is failing by examining the low and high error address registers for module repair.

Possible FRUs:

1. M.std2 module.
2. P.ioj module

Parameter change
process yyy
PC xxx
PSW xxx
Reason xxx

Message Error Level: Informational

Message Description: A parameter has been changed via the SET/SHO utility.

Field Service Action: None

Possible FRUs: None

Parity Error (Trap thru 114)
process yyy
PC xx
PSW xx
Lo err add xxxxxx
Hi err add xxxxxx
WBUSR

Message Error Level: Error

Message Description: This message covers parity errors in memory and in cache. In the case of a memory parity error, the address of the failing memory is latched into the low error address register. In the case of a cache parity error, the address is not latched into the low error address register. Instead, the address of the low error address register is displayed in the error printout.

Field Service Action: Determine if the error occurred in memory or in cache memory by reading the contents of the low error address displayed in the error printout. If the contents is the address of the low error address register (170024), the error is in cache memory. If the error is in cache, the probable FRU is the P.ioj.

Possible FRUs:

1. P.ioj
2. M.std2

Reserved Instruction (Trap thru 10)
From process yyyy
PC xxx
PSW xxx

Message Error Level: Error

Message Description: The P.ioj detected an opcode resulting in the execution of an invalid instruction. The process indicated is the process that executed the nonexistent instruction.

Field Service Action: Determine what process was active for module repair.

Possible FRUs:

1. P.ioj module.
2. M.std2 module
3. Software

Software inconsistency
Process yyy
PC xxxxxx
PSW xxxxxx
Stack dump xxxxxx xxxxxx xxxxxx

Message Error Level: Error

Message Description: During operation, the operating software performs numerous consistency checks. When one of these consistency checks fails, the HSC70 crashes and reboots. The active process is displayed, as well as the stack dump.

Field Service Action: None

Possible FRUs: None

The previous SINI error messages are a result of the operating software performing a consistency check which failed. When consistency checks fail, the HSC70 performs a soft initialization causing it to crash and reboot. This is known as a subsystem exception. Upon successful completion of the reboot, the subsystem exception printout displays the contents of several HSC70 registers as well as the status of all requestors. As a result of the subsystem exception, the SINI error message is printed. This message tells why the last soft init happened.

The actual sequence of events for a SINI-E out-of-band error printout is as follows:

1. When the HSC70 detects an unrecoverable problem, a soft init or crash occurs. A system dump is performed under the heading SUBSYSTEM EXCEPTION. The HSC70 then reboots.

2. When the HSC70 reboots, a message indicating the HSC70 has rebooted, followed by the multiline SINI message, gives the reason for the last soft init (crash).

3. The same message is written on the system diskette and can be examined with the SHO EXCEPTION command. A host error message log is also filed in host memory as an HSC datagram, storing the out-of-band error SINI message.

8.4.6 Traps

The four traps described in the following sections (Trap Thru 4, Trap Thru 10, Trap Thru 114, and Trap Thru 134) are the same as are found in the 1170 CPU.

8.4.6.1 NXM (Trap Thru 4) - If the error registers in the NXM printout equal 170024 000077, the error is not a nonexistent memory error. Instead, it is a stack overflow or some illegal instruction. When the error register is any number other than 170024 000077, the number represents the unresponsive address. The nonexistent memory trap produces a subsystem exception printout similar to the example in Section 8.4.6.5.1.

If the error register equals 16xxxx, the Window Bus register equals the Control memory address causing the NXM error. If the failing address is in Control memory and shows an NXM error, it is definitely a hardware problem. Otherwise, it can be either a software or a hardware problem.

8.4.6.2 Reserved Instruction (Trap Thru 10) - The subsystem exception message for this trap indicates on the User Pc: the vector number is 10 and identifies the trap as ILOP (an illegal opcode). Refer to the (PC-6) to (PC): field in the example (Section 8.4.6.5.1). With a trap thru 10, the first line is the field; the third word from the left is the instruction causing the trap. If this is a valid PDP-11 instruction, it is definitely a hardware problem. Otherwise, the program may not executing in the right place indicating the problem could be either hardware or software.

8.4.6.3 Parity Error (Trap Thru 114) - This error, caused by hardware, does not crash the HSC but causes a reboot and SINI error message. The error message shows the last reboot caused by the trap through 114 and the address that caused the trap.

Determine if the error occurred in memory or in cache memory by reading the contents of the low error address displayed in the error printout. If the content is the address of the low error address register (170024), the error is in cache memory. Any other address indicates the error is in memory.

In the following example printout, note the low error address and the high error address fields. When these fields contain the exact addresses as shown in this example, the error is from the P.ioj cache.

```
SINI-E  Seq 1. at 17-Nov-1858 00:00:01.60
          Parity Error (Trap Thru 114)
        Process PSCHED
        PC  111022
        PSW 140000
        Lo err adr 170024
        Hi err adr 000077
        WBUSR 020633
```

8.4.6.4 Level 7 K Interrupt (Trap Through 134) - A level 7 K interrupt, detected by hardware or microcode, occurs when one or more requestors detect a fatal error condition while executing functional code.
The microcode-detected errors causing level 7 K interrupts result from a microcode consistency check failure in either K.sdi, K.sti or K.ci microcode.
Requestor hardware detected errors are the result of errors detected on the control bus.
K.ci hardware detected errors are a result of errors detected on the control bus, scratchpad RAM parity errors/data bus parity errors or host clears, or control bus NXMs (not related to data transfers).
The requestor, upon detecting the error, generates a level 7 interrupt to the P.ioj. The P.ioj traps through location 134 causing a reboot.

8.4.6.5 Control Bus Error Conditions (Hardware Detected) - The hardware detected control bus errors causing level 7 K interrupts are:

- Control Bus Error - The requestor was in the process of executing a control bus cycle and received CERR L (control bus error low) from the P.ioj.
  The P.ioj had detected an illegal control bus cycle type.

- Control Bus Parity Error - The requestor detected bad parity on the data it read off the control bus.

- Control Bus NXM - The requestor tried to reference control memory and did not receive an acknowledgment (CACK L) from the M.std2 within the timeout period.

8.4.6.5.1 Level 7 K Interrupt Printout - An example of a detected level 7 K Interrupt follows:

```
   SUBSYSTEM EXCEPTION *-        V# 250          HSC  LONDON
       at 25 Oct 1985  00:08:46.64           0 23:23:21.40

User    PC:  110574 caused by  (134   ) Kint
PSW: 140011

PSCHED active  PCB addr = 054536

R0-R5:
000000  000024  000000  000000  000000  000000

Kernel SP: 000774

Kernel Stack

005046  000004  052744  045412  001012  000000  045644  000000
052136  000000  047260  000000  051300  000000  054742  000000

User SP: 000774

User Stack:
150042  147502  147516  000000  102146  000000  000000  000000
000000  000000  000000  000000  000000  000000  000000  000000

KPAR(0-7):
000440  000640  001040  1577770  001440  001240  000240  177600

KPDR(0-7):
077506  077506  077506  077406  077506  077406  077506  077506

UPAR(0-7):
000000  000000  000000  000000  002204  001240  000240  177600

UPDR(0-7):
077406  077406  077406  077406  063406  077406  077406  000116

MMSR(0-2): 000017  000000  037260

Window Index Reg: 000026

Window Bus Reg: 001431

WADR(0-7):

160004  161004  162004  163004  164004  165004  166004  167004

Translated WADR(0-7):
001401  001401  001401  001401  001401  001401  001401  001401

Error Regs: 170024  000077

Status of Requestors(1-9):
```

```
000001  000377  000377  000377  000377  000175  000377  000377  000377

(PC-6) to (PC):
013737  141020  110560  013701
Control area for slot #000006
Control area address: 017660:
Register area contents:
000000
000000
000011
021154
102557
000770
000000
000000
017650
000000
057502
005317
002224
001000
000000
000671
000000
143444
107001
001000
005317
002212
000671
001000
000000
000000
000000
040506
000010
000374
043520
005400
001000

Booting
INIPIO-I Booting
```

Requestor 6 has failed with a status of 175. Refer Appendix D to determine if the failure was a control bus error.

At this time the HSC70 reboots. A message is displayed on the local console
terminal stating the HSC70 has rebooted.

```
^*HSC Version 200 29-Sept-1985 23:17:28   System LONDON\*
```

The actual SINI error message is printed on the local console terminal
after the HSC70 has rebooted.

```
SINI-E  Error sequence 1. at 17-Nov-1858 00:00:03.00
        Last soft init caused by level 7 K interrupt
        From process PSCHED
        PC  110574
        Status: 001 377 377 377 377 175 377 377 377
```

The resulting 134 trap information is printed on the local console terminal. The PSCHED statement indicates PSCHED was the active process when the error occurred. The status statement shows requestor 6 failed with a status of 175. Also, three lines after the status line is a message line indicating the control area for slot six and slot six control address. This indicates requestor six is the failing requestor. The INIPIO-I Booting statement indicates the HSC70 is attempting to reboot.

When the HSC70 completes the initialization, the Last Soft Init caused by Level 7 K interrupt failure is printed on the local console terminal identified by SINI-E. The active process at time of failure is identified. In this case, the active process was PSCHED. If the failure is a hard failure, the following message may also be displayed on the local console terminal.

```
SINI-E ERROR SEQUENCE 1. AT 25-OCT-1858 00:00:02.80
    REQUESTOR 6 FAILED INIT DIAGS, STATUS 107
```

This message is also considered an out-of-band error.

8.4.6.6 MMU (Trap Thru 250)

Following is an sample printout of a detected Memory Management Unit (MMU) failure.

```
**SUBSYSTEM EXCEPTION**    V# Y10B          HSC70 LAYER
      at 12-DEC-1985 13:43:40.05    up   2 19:24:07.40

User  PC:  004747  caused by (250 ) MMU
PSW: 140000

SETSHO active, PCB addr = 104116

R0-R5:
000320  000001  100000  100212  000266  000002

Kernel SP: 000774
```

```
Kernel Stack:
005046  000004  053314  045762  001012  000000  046214  000000
047022  000000  047426  000000  052052  000000  051042  000000

User SP: 000226

User Stack:
040314  021356  033552  021356  021246  000040  017440  017440
020040  020037  020037  000330  101000  027113  000144  060542

KPAR(0-7):
000440  000640  001040  001440  002040  001240  000240  177600

KPDR(0-7):
077506  077506  077506  077506  077506  077506  077506  077506

UPAR(0-7):
007074  007274  006410  000000  002240  001240  000240  177600

UPDR(0-7):
077506  077406  013406  077406  077406  077506  077506  000116

MMSR(0-2): 040145  000000  004743

Window index reg: 000002

Window bus reg: 001407

WADR(0-7):

160000  161004  162440  163000  164004  165004  166220  167034

Translated WADR(0-7):
000000  001401  067510  040000  001401  001401  010444  001407

Error rags: 170024  000077

Status of requestors(1-9):
000001  000002  000002  000002  000203  000203  000203  000377  000377

(PC-6) to (PC):
027441  067516  051040  071545
```

Because the trap is a memory management trap, look first at the register contents of MMSR0 (memory management status register 0). Refer to Figure 8-11 for a breakdown of the bits in MMSR0.

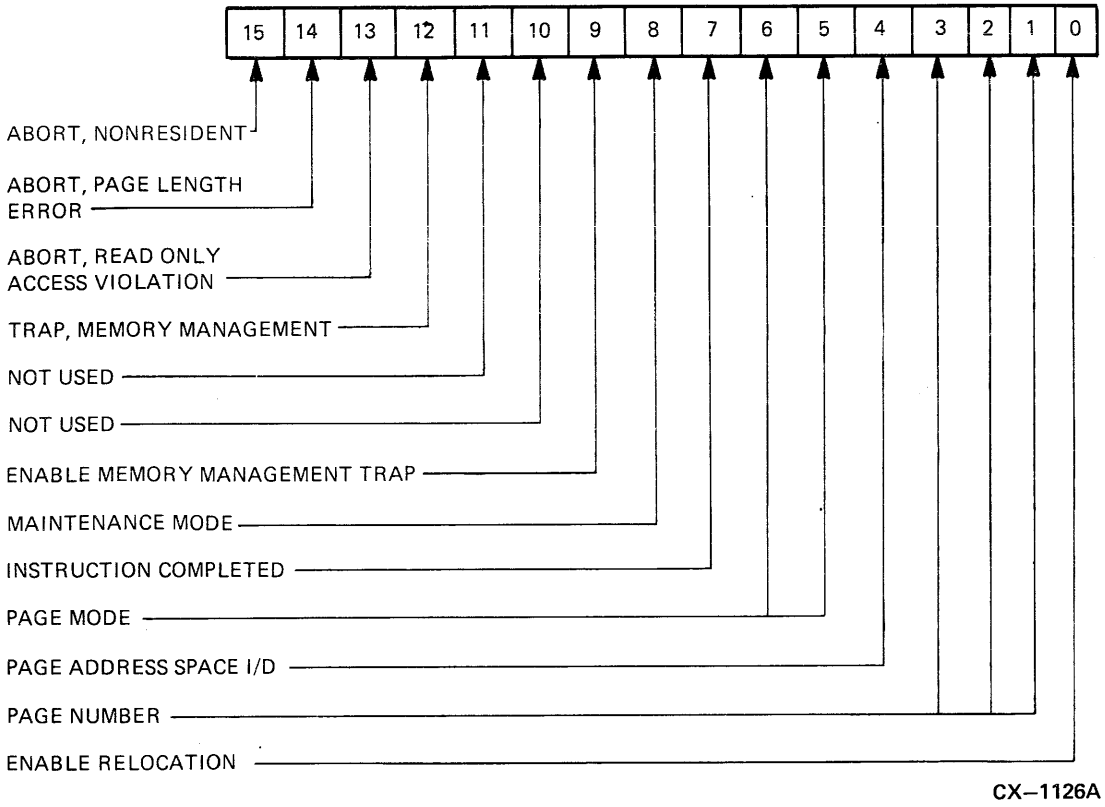


Figure 8-11 MMSR0 Bit Breakdown

Look at the printout lines for MMSR (0-2). Compare the bits set in MMSR0 to the bit breakdown in Figure 8-11. The example indicates a page length violation on page 2. The page length error bit is set, and the page number 2 bit is set.

Next, check the PSW line and determine the mode the HSC70 reported this error in. A 140000 in the PSW means user mode, a 000000 in the PSW means kernel mode. Also, above the PSW line the word user or kernel appears to identify the mode. Our example shows user mode is active. Therefore, the next register contents of any value are the UPAR and UPDR. If the active mode had been kernel, the important registers would have been the KPDR and KPAR registers.

The first group of numbers under the UPAR(0-7) line is for page zero, the second for page one, the third for page two, and so forth. The third group of numbers in the example are for page two, the violated page. Note the difference in UPDR contents on page two versus the UPDR contents on other pages. The UPDR contents on other pages all start with 077 designating a full page of memory to be allocated for that page. The UPDR contents on page 2 starts with a 013. indicating a failure also.

Two possible problems cause this error:

1. Memory Management Unit on the P.ioj
2. Software

If the error occurred in page 0, the problem is a hardware problem. Replace the P.ioj. Otherwise, let the error recur and see if different pages are affected.

Software Inconsistency (Trap thru 20) is reported similar to trap. A subsystem exception is dumped on the local console terminal with the trap vector reported being a Trap thru 20, (AT). An example printout and explanation are found in Appendix B.

The subsystem exception is followed by the HSC70 reboot. Upon successful reboot, the following message is displayed.

```
HSC70 Version Y10B   16-Jan-1986  15:30:20.20  System MASTER
```

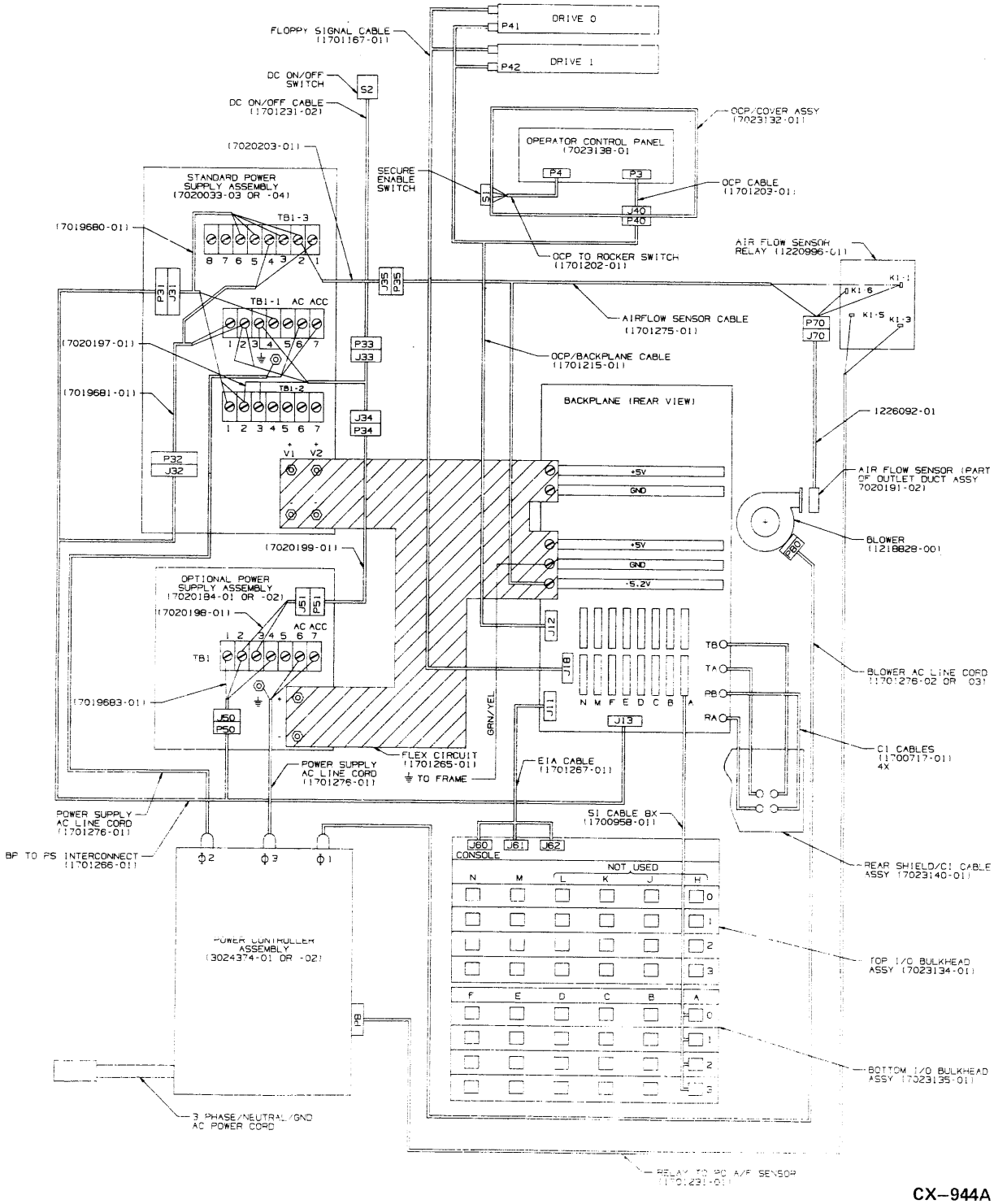
Then the SINI error resulting from the detected subsystem exception is printed.

```
SINI-E Sequence 1. at 16-Jan-1986  00:00:11.20
 Last soft init caused by software inconsistency
 From process HOST
 PC  007044
 PSW 140001
 Stack dump: 000016  006401  015476
```

APPENDIX A
INTERNAL CABLING DIAGRAM

A.1 HSC70 INTERNAL CABLING

Figure A-1 is a diagram of the internal HSC70 cabling.



CX-944A
Sheet 1 of 5

Figure A-1 HSC70 Internal Cabling (1 of 5)

WIRE TABLE 1226092-01 A/F SENSOR				
COLOR	FROM	TO	SIGNAL	REMARKS
RED	A1-+	J70-1		
BLACK	A1-GND	J70-3		
WHITE	A1-LOAD	J70-2		

WIRE TABLE 1701202-01 OCP TO ROCKER SWITCH				
COLOR	FROM	TO	SIGNAL	REMARKS
	P4-01		NO CONNECTION	SPARE
	P4-02		NO CONNECTION	KEYING PLUG
RED	P4-03	S1-3	+5 VOLT	
BLK	P4-04	S1-6	GND (+5 VOLT)	
	P4-05		NO CONNECTION	SPARE
YEL	P4-06	S1-4	GND	
YEL	P4-07	S1-5	TERM ENABLE	
	P4-08		NO CONNECTION	SPARE
WHT	P4-09	S1-1	INIT SWL	
WHT	P4-10	S1-2	INIT L	

WIRE TABLE 1701203-01 OCP CABLE				
COLOR	FROM	TO	OCP SIGNAL	REMARKS
YELLOW	J40-1	P3-1	STATE LAMP L	
YEL/ORG	J40-2	P3-2	POWER LAMP L	
YEL/BLU	J40-3	P3-4	LAMP ENA 0 L	
YEL/GRN	J40-4	P3-3	TERM ENA L	
YEL/BLK	J40-5	P3-6	LAMP ENA 2 L	
YEL/VIO	J40-6	P3-5	LAMP ENA 1 L	
YEL/GRY	J40-7	P3-8	LAMP ENA 4 L	
YEL/WHT	J40-8	P3-7	LAMP ENA 3 L	
YEL/RED	J40-9	P3-10	PANEL SWITCH 1 L	
YEL/BRN	J40-10	P3-9	PANEL SWITCH 0 L	
YEL/BLK/GRY	J40-11	P3-12	PANEL SWITCH 3 L	
YEL/GRN/ORG	J40-12	P3-11	PANEL SWITCH 2 L	
YEL/RED/WHT	J40-13	P3-15	BDCOKH (INIT L)	
BLACK	J40-14	P3-14	GND	
RED	J40-15	P3-16	+5V	
		P3-20		KEYING PLUG (OCP)

WIRE TABLE 1701215-01 OCP/BACKPLANE				
YELLOW	J12-1	P40-01	STATE LAMP L	
YELLOW/ORG	J12-2	P40-02	POWER LAMP L	
YELLOW/BLUE	J12-3	P40-03	LAMP ENA 0 L	
YELLOW/GRN	J12-4	P40-04	TERM ENA L	
YELLOW/BLACK	J12-5	P40-05	LAMP ENA 2 L	
YELLOW/VIOLET	J12-6	P40-06	LAMP ENA 1 L	
YELLOW/GRAY	J12-7	P40-07	LAMP ENA 4 L	
YELLOW/WHITE	J12-8	P40-08	LAMP ENA 3 L	
YELLOW/RED	J12-9	P40-09	PANEL SWITCH 1 L	
YELLOW/BRN	J12-10	P40-10	PANEL SWITCH 0 L	
YELL/BLK/GRY	J12-11	P40-11	PANEL SWITCH 3 L	
YELL/GRN/ORG	J12-12	P40-12	PANEL SWITCH 2 L	
YELL/RED/WHT	J12-13	P40-13	BDCOK H(INT L)	
BLACK	J12-14	P40-14	GROUND	
RED	J12-16	P40-15	+5 VOLTS	
RED	J12-19	P41-04	+5 VOLTS	
RED	J12-20	P42-04	+5 VOLTS	
BLACK	J12-21	P41-02	GROUND	
BLACK	J12-22	P41-03	GROUND	
BLACK	J12-23	P42-02	GROUND	
BLACK	J12-24	P42-03	GROUND	
VIOLET	J12-25	P41-01	+12 VOLTS	
VIOLET	J12-26	P42-01	+12 VOLTS	

CX-944A
Sheet 2 of 5

Figure A-1 HSC70 Internal Cabling (2 of 5)

WIRE TABLE 1701231-01 RELAY TO PC A/F SENSOR				
COLOR	FROM	TO	SIGNAL	REMARKS
WHITE	K1-3	P8-1	TRIP	
WHITE	K1-5	P8-2	RETURN	

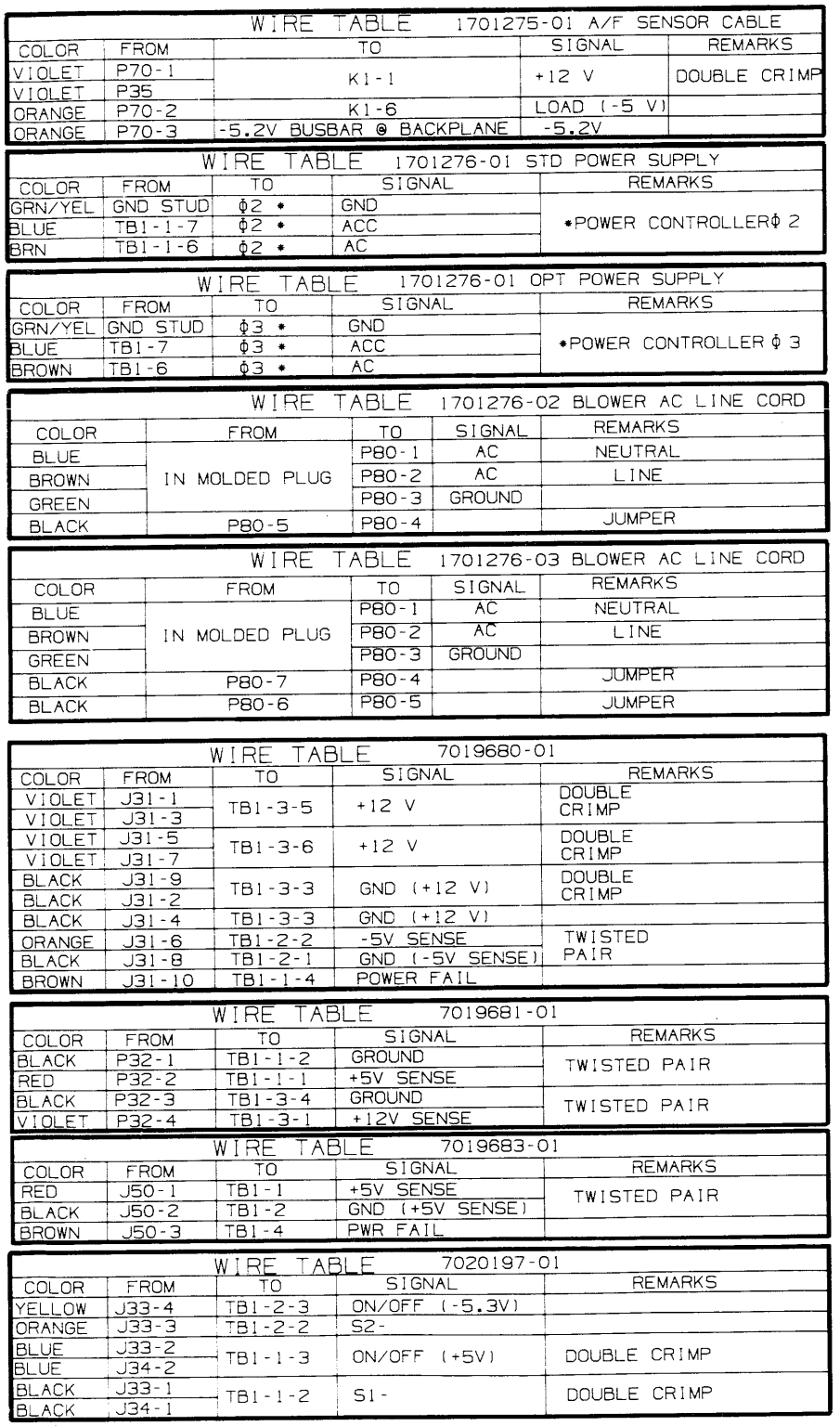
WIRE TABLE 1701231-02 DC ON/OFF				
COLOR	FROM	TO	SIGNAL	REMARKS
YELLOW	S2-2	P33-4	ON/OFF (-5.2V)	
ORANGE	S2-1	P33-3	S2-	
BLUE	S2-4	P33-2	ON/OFF (+5.0V)	
BLACK	S2-3	P33-1	S1-	

WIRE TABLE 1701266-01 BP TO PS					
COLOR	FROM	TO	SIGNAL	REMARKS	
VIOLET	J13-1	P31-1	+12V		STANDARD POWER SUPPLY
VIOLET	J13-2	P31-3	+12V		
VIOLET	J13-3	P31-5	+12V		
VIOLET	J13-4	P31-7	+12V		
BLK	J13-5	P31-9	GND(+12V)		
BLK	J13-6	P31-2	GND(+12V)		
BLK	J13-7	P31-4	GND(+12V)	DOUBLE CRIMPED	
BLK	J13-8		GND(+12V)		
ORANGE	J13-9	P31-6	-5.2V SENSE	TWISTED PAIR	
BLK	J13-10	P31-8	GND(-5V SENSE)		
BRN	J13-11	P31-10	POWER FAIL L		
BLK	J13-14	J32-1	GND(+5V SENSE)	TWISTED PAIR	
RED	J13-13	J32-2	+5V SENSE		
BLK	J13-16	J32-3	GND(+12V SENSE)	TWISTED PAIR	
VIOLET	J13-15	J32-4	+12V SENSE		
BLK	J13-17	P50-2	GND(+5V SENSE)	TWISTED PAIR	OPTIONAL POWER SUPPLY
RED	J13-18	P50-1	+5V SENSE		
BRN	J13-20	P50-3	POWER FAIL L		

WIRE TABLE 1701267-01 EIA				
COLOR	FROM	TO	BACKPLANE SIGNAL	REMARKS
WHITE	J11-1	J60-20	HSC RDY+	
WHITE/BLK	J11-2	J60-6	TERM PRES L	
WHITE/BLU	J11-3	J60-1	TERM XMT-	
WHITE/ORG	J11-4	J60-2	TERM XMT+	
WHITE/RED	J11-5	J60-3	TERM RCV+	
WHITE/VIO	J11-6	J60-7	TERM RCV-	
WHITE	J11-9	J61-20	HSC RDY+	
WHITE/BLK	J11-10	J61-6	AUX1 PRES L	
WHITE/BLU	J11-11	J61-1	AUX1 XMT-	
WHITE/ORG	J11-12	J61-2	AUX1 XMT+	
WHITE/RED	J11-13	J61-3	AUX1 RCV+	
WHITE/VIO	J11-14	J61-7	AUX1 RCV-	
WHITE	J11-17	J62-20	HSC RDY+	
WHITE/BLK	J11-18	J62-6	AUX2 PRES L	
WHITE/BLU	J11-19	J62-1	AUX2 XMT-	
WHITE/ORG	J11-20	J62-2	AUX2 XMT+	
WHITE/RED	J11-21	J62-3	AUX2 RCV+	
WHITE/VIO	J11-22	J62-7	AUX2 RCV-	

CX-944A
Sheet 3 of 5

Figure A-1 HSC70 Internal Cabling (3 of 5)



WIRE TABLE 1701275-01 A/F SENSOR CABLE

COLOR	FROM	TO	SIGNAL	REMARKS
VIOLET	P70-1	K1-1	+12 V	DOUBLE CRIMP
VIOLET	P35			
ORANGE	P70-2	K1-6	LOAD (-5 V)	
ORANGE	P70-3	-5.2V BUSBAR @ BACKPLANE	-5.2V	

WIRE TABLE 1701276-01 STD POWER SUPPLY

COLOR	FROM	TO	SIGNAL	REMARKS
GRN/YEL	GND STUD	ϕ2 *	GND	*POWER CONTROLLERϕ 2
BLUE	TB1-1-7	ϕ2 *	ACC	
BRN	TB1-1-6	ϕ2 *	AC	

WIRE TABLE 1701276-01 OPT POWER SUPPLY

COLOR	FROM	TO	SIGNAL	REMARKS
GRN/YEL	GND STUD	ϕ3 *	GND	*POWER CONTROLLER ϕ 3
BLUE	TB1-7	ϕ3 *	ACC	
BROWN	TB1-6	ϕ3 *	AC	

WIRE TABLE 1701276-02 BLOWER AC LINE CORD

COLOR	FROM	TO	SIGNAL	REMARKS
BLUE	IN MOLDED PLUG	P80-1	AC	NEUTRAL
BROWN		P80-2	AC	LINE
GREEN		P80-3	GROUND	
BLACK	P80-5	P80-4		JUMPER

WIRE TABLE 1701276-03 BLOWER AC LINE CORD

COLOR	FROM	TO	SIGNAL	REMARKS
BLUE	IN MOLDED PLUG	P80-1	AC	NEUTRAL
BROWN		P80-2	AC	LINE
GREEN		P80-3	GROUND	
BLACK	P80-7	P80-4		JUMPER
BLACK	P80-6	P80-5		JUMPER

WIRE TABLE 7019680-01

COLOR	FROM	TO	SIGNAL	REMARKS
VIOLET	J31-1	TB1-3-5	+12 V	DOUBLE CRIMP
VIOLET	J31-3			
VIOLET	J31-5	TB1-3-6	+12 V	DOUBLE CRIMP
VIOLET	J31-7			
BLACK	J31-9	TB1-3-3	GND (+12 V)	DOUBLE CRIMP
BLACK	J31-2			
BLACK	J31-4	TB1-3-3	GND (+12 V)	
ORANGE	J31-6	TB1-2-2	-5V SENSE	TWISTED PAIR
BLACK	J31-8	TB1-2-1	GND (-5V SENSE)	
BROWN	J31-10	TB1-1-4	POWER FAIL	

WIRE TABLE 7019681-01

COLOR	FROM	TO	SIGNAL	REMARKS
BLACK	P32-1	TB1-1-2	GROUND	TWISTED PAIR
RED	P32-2	TB1-1-1	+5V SENSE	
BLACK	P32-3	TB1-3-4	GROUND	TWISTED PAIR
VIOLET	P32-4	TB1-3-1	+12V SENSE	

WIRE TABLE 7019683-01

COLOR	FROM	TO	SIGNAL	REMARKS
RED	J50-1	TB1-1	+5V SENSE	TWISTED PAIR
BLACK	J50-2	TB1-2	GND (+5V SENSE)	
BROWN	J50-3	TB1-4	PWR FAIL	

WIRE TABLE 7020197-01

COLOR	FROM	TO	SIGNAL	REMARKS
YELLOW	J33-4	TB1-2-3	ON/OFF (-5.3V)	
ORANGE	J33-3	TB1-2-2	S2-	
BLUE	J33-2	TB1-1-3	ON/OFF (+5V)	DOUBLE CRIMP
BLUE	J34-2			
BLACK	J33-1	TB1-1-2	S1-	DOUBLE CRIMP
BLACK	J34-1			

CX-944A
Sheet 4 of 5

Figure A-1 HSC70 Internal Cabling (4 of 5)

WIRE TABLE 7020198-01				
COLOR	FROM	TO	SIGNAL	REMARKS
BLUE	J51-2	TB1-3	ON/OFF +5V	
BLACK	J51-1	TB1-2	S-	

WIRE TABLE 7020199-01				
COLOR	FROM	TO	SIGNAL	REMARKS
BLACK	P34-1	P51-1	S-	
BLUE	P34-2	P51-2	ON/OFF (+5V)	

WIRE TABLE 7020203-01				
COLOR	FROM	TO	SIGNAL	REMARKS
VIOLET	J35	TB1-3-2	+12 V	

CX-944A
Sheet 5 of 5

Figure A-1 HSC70 Internal Cabling (5 of 5)

APPENDIX B
EXCEPTION CODES AND MESSAGES

This appendix describes all known HSC exception (crash) codes caused entirely or in part by software inconsistencies. For ease of reference, these codes are arranged in numerical order (octal radix). Each message contains the code number, the meaning of the crash, the facility causing it, an explanation, and user action. Note the code number but not the text appears on hardcopy printouts.

B.1 Overview

In order to determine which exception code caused a particular crash, refer to the crash dump printed out at your terminal. The following HSC7Ø crash dump example shows you where to look.

Examples

```
-*  SUBSYSTEM EXCEPTION  *-      V1Ø0            HSC7Ø  HSCØØ1 1
         at 17-Nov-1858 Ø0:13:34.2Ø         up     Ø ØØ:13:34.2Ø
User 2  Pc: Ø15Ø66 caused by (2Ø      ) IOT 3
PSW:  14ØØØ1
DEMON 4 active, PCB addr = Ø54214
RØ-R5:
ØØØØØ5   ØØØØØØ   Ø23ØØ4   147602   16ØØ2Ø   154752
Kernel SP: ØØØ774
Kernel Stack:
ØØ5Ø45 5 ØØØØØ4   Ø53336   Ø46ØØ4   ØØ1Ø12   ØØØØØØ   Ø46236   ØØØØØØ
Ø47Ø44   ØØØØØØ   Ø47450   ØØØØØØ   Ø52Ø74   ØØØØØØ   Ø55334   ØØØØØØ
User SP: 154734
User Stack:
ØØ2Ø13 6 1Ø4262   14Ø31Ø   1Ø225Ø   ØØØØ34   Ø35Ø64   ØØ43Ø5   ØØØØØØ
ØØØØØØ   ØØØØØ3   ØØØØØ1   ØØØØØ4   ØØØØØØ   ØØ2445   ØØØØØØ   ØØØØØØ

KPAR(Ø-7):
    .
    .
    .
Booting
INIPIO-I  Booting...
```

1 This line calls out a crash and indicates the HSC7Ø is at software version number V1ØØ. The last field is the assigned node name (set with SET NAME).

2 Mode of the crash. This can be either Kernel or User. It indicates in which processor mode the crash occurred.

3 This three-letter mnemonic indicates the crash is a software inconsistency. Any other combination of letters, such as NXM (Non-Existent Memory) would designate a crash outside the scope of this appendix. Hardware exceptions are defined in Appendix D.

4 The initial name on this line identifies the process active at the time of the crash. It is valid only during user-mode crashes. This name can be used as a cross-check when you look up your crash description.

5 If the mode notation is Kernel, you would read the first word of the Kernel Stack for your crash code.

6 Because the mode notation in this example indicated User, check the User Stack for your crash code number. This code is always the first word on the stack (in this case 002013).

The crash codes are listed numerically in this appendix (Section B.3). Consult them for explanations and suggested user action.

The following SINI-E error example appears immediately upon reboot after a subsystem exception. Information contained in this error message is a condensation of the crash dump.

Examples

```
SINI-E  Seq 1. at 17-Nov-1858 00:00:02.00
        Software inconsistency 1
        Process DEMON  2
        PC  000002
        PSW 140001
        Stack dump:  002013  104262  140310
```

1 This line defines the cause of the crash.

2 This line and the following three lines duplicate the applicable information in the crash dump.

In each of the exception descriptions in this appendix, Facility indicates the process(es) running at the time the crash occurred. The first name listed is the major process. The second is the module of the process that generated the exception. This may be a subprocess of the main process or simply a different code module. A large number of these messages request submission of

an SPR (Software Performance Report). This process is described in the following section.

B.2 SPR SUBMISSION

Include with the SPR the crash dump message and any other hard-copy information needed. Your customer will contact you or the Customer Support Center if a crash dump appears with one of the exception messages described in this appendix. The HSC User Guide gives the customer a short explanation about the exception condition. This appendix shows the same messages, but provides more detailed information needed to analyze the crash.

In many cases, the HSC User Guide tells your customer to submit a Software Performance Report (SPR). The SPR should be submitted only after you decide a hardware condition did not cause the error, and you suspect a software problem.

In some cases, not all of the necessary information which must accompany the SPR is contained in the crash dump (the message printed on the console when the HSC detects an exception). This appendix lists the additional information you or the customer must gather after the HSC has printed its crash dump message.

After this additional information is known, the Customer Support Center may be able to assist you over the telephone. If an SPR is necessary, your customer must include all the information listed for the specific exception code.

After two or three similar exception messages occur and you determine the customer should submit the SPR, look up the exception message in this appendix. If a data structure (for instance, HMB or PCB) should be included with the SPR, set the ODT parameter, causing the HSC to enter ODT after an exception. If data structures are not requested in the applicable exception code, you do not need to enter ODT.

To set the ODT parameter type:

```
CTRL Y
HSC>  RUN SETSHO
SETSHO>  SET ODT DUMP_BPT
SETSHO>  EXIT
SETSHO-Q Rebooting HSC; Y to continue, CTRL/Y to abort:?  Y
```

The HSC then reboots with the new parameter ODT DUMP_BPT set. When the next exception occurs, the HSC prints the exception message, followed by an asterisk (*) prompt. Instruct your customer to call you or the Customer Support Center when the next crash occurs.

NOTE

If you instruct the customer to call the Customer Support Center for assistance, inform the Center of the problem. Also, let them know your customer will need help gathering information related to the software error.

When another crash occurs, check the appropriate exception code in this appendix for information needed to analyze the crash. Include all the requested information with the SPR.

Data structures needed with the SPR must be formatted. These data structures are addressed by a register or the contents of another structure's field. To format the necessary data structure(s), substitute the **x** in Table B-1 with the pointer from the specified register or location. Substitute only the x and type the rest of the line exactly as you see it in the table, except for the information in parentheses. The number of = signs designates which memory the data structure is in (= indicates Program memory, == indicates Control memory, and === indicates Data memory).

Table B-1 Obtaining Data Structure Information

Data Structure Needed	Type this at the * prompt
CB	x==CB$
Counter	x=C. (and) x==C.
DCB	x==DC$DISK (or) x==DC$TAPE (if tape path problem)
DDCB	x=DD$
FRB	x==F$
HCB	x=HC$
HMB	x==HM$ (command packet) x==HM$CPY (BACKUP) x==HM$DATA (with BMBs) x==HM$QUIET (diagnostic) x==HM$XFR (used while work is outstanding) x==HM$VC (used to alter VC state)
K Control Area	x==KG$
PCB	x=Z.
SLCB	x=SL$
TDCB	x=TD.
TFCB	x=TF.
TTCB	x=TT$
XFRB	x==X.

After the information is complete, the customer should fill out the SPR and submit it together with all hardcopy as instructed on the SPR form.

B.3 EXCEPTION MESSAGES

001001 ($CKERSTK)
Execution of Kernel Stack

Facility: EXEC, EXEC

Explanation: The HSC executive executed stack space.

User Action: Submit an SPR with a dump. You may reboot immediately.

001002 ($CPUM1)
Previous mode not user

Facility: EXEC, EXEC

Explanation: During context switch of user processes, the previous mode (as indicated by the PSW) was not user mode.

User Action: Submit a Software Performance Report (SPR) with a dump. R5 points to PCB (Process Control Block).

001003 ($CEXPCB)
EXEC PCB was scheduled

Facility: EXEC, EXEC

Explanation: During process scheduling, the EXEC PCB (Process Control Block) was scheduled. This dummy PCB is used only for loading the process and should never be scheduled.

User Action: Submit an SPR with a dump. R2 points to PCB.

001004 ($CDEBCAC)
Cache setting in PDR is in incorrect state

Facility: EXEC, EXEC

Explanation: This software inconsistency should not appear under normal circumstances. A PDR (Page Descriptor Register) directed to program memory does not have "disable cache" set. A PDR directed to data memory does have "disable cache" set.

User Action: Submit an SPR with a dump. R0 points to PDR.

001005 ($CPUM2)
Previous mode not user

Facility: EXEC, EXEC

Explanation: During context switch of user processes, the previous mode (as indicated by the PSW) was not user mode.

User Action: Submit an SPR with a dump.

001006 ($CCB4)
Spurious Interrupt from K at Control Bus Level 4

Facility: EXEC, EXEC

Explanation: This software inconsistency should not appear under normal circumstances. One of the Ks interrupted the P.ioc at level 4 (an element should be on the level 4 interrupt queue), yet upon examination, no elements were shown on the queue.

User Action: Save any dump before rebooting. Submit an SPR. If this crash continues to occur, escalate the problem to Field Service support.

001007 ($CCB5)
Spurious Interrupt from K at Control Bus Level 5

Facility: EXEC, EXEC

Explanation: This software inconsistency should not appear under normal circumstances. One of the Ks interrupted the P.ioc at level 5 (an element should be on the level 5 interrupt queue), yet upon examination, no elements were shown on the queue.

User Action: Submit an SPR. If this crash continues to occur, escalate the problem to Field Service support.

001010 ($CDC1)
Downcount failed

Facility: EXEC, EXEC

Explanation: This software inconsistency should not appear under normal circumstances. During processing of the level 5 interrupt queue, a down-count operation on a counter (down counted by 1) failed.

User Action: Submit an SPR with a dump. R1 points to counter.

001011 ($CDC2)
Downcount failed

Facility: EXEC, EXEC

Explanation: This software inconsistency should not appear under normal circumstances. During processing of the level 5 interrupt queue, a down-count operation on a counter (down counted by 1) failed.

User Action: Submit an SPR with a dump. R1 points to counter.

001012 ($CACQ)
Acquire on Semaphore with address of 0

Facility: EXEC, EXEC

Explanation: This software inconsistency should not appear under normal circumstances. The ACQ$P System Service was called with a Semaphore address of 0.

User Action: The process specified as active is the offender. Submit an SPR with a dump.

001013 ($CAML)
Acquire Multiple on Semaphore with address of 0

Facility: EXEC, EXEC

Explanation: This software inconsistency should not appear under normal circumstances. The AMLT$P System Service was called with a Semaphore address of 0.

User Action: The process specified as active is the offender. Submit an SPR with a dump.

001014 ($CRLP)
Release on Semaphore with address of 0

Facility: EXEC, EXEC

Explanation: This software inconsistency should not appear under normal circumstances. The REL$P System Service was called with a Semaphore address of 0.

User Action: The process specified as active is the offender. Submit an SPR with a dump.

ØØ1Ø15 ($CRRTI)
RRTI$ on Semaphore with address of Ø

Facility: EXEC, EXEC

Explanation: This software inconsistency should not appear under normal circumstances. The RRTI$P System Service was called with a Semaphore address of Ø.

User Action: The process specified as active is the offender. Submit an SPR with a dump.

ØØ1Ø16 ($CRTI1)
RRTI$ on Semaphore with address of Ø

Facility: EXEC, EXEC

Explanation: This software inconsistency should not appear under normal circumstances. The RRTI$P System Service was called with a Semaphore address of Ø.

User Action: The process specified as active is the offender. Submit an SPR with a dump.

ØØ1Ø17 ($CRTI2)
RRTI$ on Semaphore with address of Ø

Facility: EXEC, EXEC

Explanation: This software inconsistency should not appear under normal circumstances. The RRTI$P System Service was called with a Semaphore address of Ø.

User Action: The process specified as active is the offender. Submit an SPR with a dump.

001020 ($CRCPP)
Receive/Dequeue from Queue with address of 0

Facility: EXEC, EXEC

Explanation: This software inconsistency should not appear under normal circumstances. One of the RCV$P FROM$P or DEQ$P FROM$P System Services was called with a Queue Head address of 0.

User Action: The process specified as active is the offender. Submit an SPR with a dump.

001021 ($CRCCP)
Receive/Dequeue from Queue with address of 0

Facility: EXEC, EXEC

Explanation: This software inconsistency should not appear under normal circumstances. One of the RCV$C FROM$P or DEQ$C FROM$P System Services was called with a Queue Head address of 0.

User Action: The process specified as active is the offender. Submit an SPR with a dump.

001022 ($CRCCV)
Receive/Dequeue from Queue with address of 0

Facility: EXEC, EXEC

Explanation: This software inconsistency should not appear under normal circumstances. One of the RCV$C FROM$P, DEQ$C FROM$P, RCV$C FROM$W, or DEQ$C FROM$W System Services was called with a Queue Head address of 0.

User Action: The process specified as active is the offender. Submit an SPR with a dump.

ØØ1Ø23 ($CRMPP)
Receive/Dequeue Multiple from Queue with address of Ø

Facility: EXEC, EXEC

Explanation: This software inconsistency should not appear under normal circumstances. One of the RMLT$P FROM$P, or DMLT$P FROM$P System Services was called with a Queue Head address of Ø.

User Action: The process specified as active is the offender. Submit an SPR with a dump.

ØØ1Ø24 ($CRMCP)
Receive/Dequeue Multiple from Queue with address of Ø

Facility: EXEC, EXEC

Explanation: This software inconsistency should not appear under normal circumstances. One of the RMLT$C FROM$P, or DMLT$C FROM$P System Services was called with a Queue Head address of Ø.

User Action: The process specified as active is the offender. Submit an SPR with a dump.

ØØ1Ø25 ($CRMCV)
Receive/Dequeue Multiple from Queue with address of Ø

Facility: EXEC, EXEC

Explanation: This software inconsistency should not appear under normal circumstances. One of the RMLT$C FROM$P, DMLT$C FROM$P, RMLT$C FROM$W, or DMLT$C FROM$W System Services was called with a Queue Head address of Ø.

User Action: The process specified as active is the offender. Submit an SPR with a dump.

001026 ($CRAMCV)
Receive All-Maybe from Queue with address of 0

Facility: EXEC, EXEC

Explanation: This software inconsistency should not appear under normal circumstances. One of the RCAM$C FROM$P, or RCAM$C FROM$W System Services was called Queue Head address of 0.

User Action: The process specified as active is the offender. Submit an SPR with a dump.

001027 ($CSPP)
Send/Enqueue to Queue with address of 0

Facility: EXEC, EXEC

Explanation: This software inconsistency should not appear under normal circumstances. One of the SEND$P TO$P or ENQ$P TO$P System Services was called with a Queue Head address of 0.

User Action: The process specified as active is the offender. Submit an SPR with a dump.

001030 ($CSCP)
Send/Enqueue to Queue with address of 0

Facility: EXEC, EXEC

Explanation: This software inconsistency should not appear under normal circumstances. One of the SEND$C TO$P or ENQ$C TO$P System Services was called with a Queue Head address of 0.

User Action: The process specified as active is the offender. Submit an SPR with a dump.

ØØ1Ø31 ($CSCV)
Send/Enqueue to Queue with address of Ø

Facility: EXEC, EXEC

Explanation: This software inconsistency should not appear under normal circumstances. One of the SEND$C TO$P, ENQ$C TO$P, SEND$C TO$W or ENQ$C TO$W System Services was called with a Queue Head address of Ø.

User Action: The process specified as active is the offender. Submit an SPR with a dump.

ØØ1Ø32 ($CSHPP)
Send-/Enqueue-to-Head to Queue with address of Ø

Facility: EXEC, EXEC

Explanation: This software inconsistency should not appear under normal circumstances. One of the SNDH$P TO$P or ENQH$P TO$P System Services was called with a Queue Head address of Ø.

User Action: The process specified as active is the offender. Submit an SPR with a dump.

ØØ1Ø33 ($CSHCP)
Send-/Enqueue-to-Head to Queue with address of Ø

Facility: EXEC, EXEC

Explanation: This software inconsistency should not appear under normal circumstances. One of the SNDH$C TO$P, ENQH$C TO$P, SNDH$C TO$P, or ENQH$C TO$P System Services was called with a Queue Head address of Ø.

User Action: The process specified as active is the offender. Submit an SPR with a dump.

001034 ($CIHPP)
Insert at Head to Queue with address of 0

Facility: EXEC, EXEC

Explanation: This software inconsistency should not appear under normal circumstances. The INSH$P TO$P System Service was called with a Queue Head address of 0.

User Action: The process specified as active is the offender. Submit an SPR with a dump.

001035 ($CIHCP)
Insert at Head to Queue with address of 0

Facility: EXEC, EXEC

Explanation: This software inconsistency should not appear under normal circumstances. The INSH$C TO$P System Service was called with a Queue Head address of 0.

User Action: The process specified as active is the offender. Submit an SPR with a dump.

001036 ($CUPCV)
Upcount to Counter with address of 0

Facility: EXEC, EXEC

Explanation: This software inconsistency should not appear under normal circumstances. The UPC$ System Service was called with a Queue Head address of 0.

User Action: The process specified as active is the offender. Submit an SPR with a dump.

001037 ($CDWCV)
Downcount to Counter with address of 0

Facility: EXEC, EXEC

Explanation: This software inconsistency should not appear under normal circumstances. The DWNC$ System Service was called with a Queue Head address of 0.

User Action: The process specified as active is the offender. Submit an SPR with a dump.

001040
Set Timer operation to Timer with address of 0

Facility: EXEC, EXEC

Explanation: This software inconsistency should not appear under normal circumstances. The SETTM$ System Service was called with a Queue Head address of 0.

User Action: The process specified as active is the offender. Submit an SPR with a dump.

001041 ($CSNZ1)
Release of Semaphore with address of 0

Facility: EXEC, EXEC

Explanation: This software inconsistency should not appear under normal circumstances. During some circumstances, a semaphore will require a down count, without subsequent scheduling considerations. This typically happens when a process enters hibernation or exits. During the implicit release operation, the Semaphore had an address of 0.

User Action: Submit an SPR with a dump.

ØØ1Ø42 ($CTOVR)
Time-of-day overflowed

Facility: EXEC, EXEC

Explanation: During update of current time of day, the Executive detected an overflow. This can happen if a node on the CI sets a bogus time to the HSC.

User Action: Examine previous console printouts to verify accurate date and time fields. If accurate, submit an SPR with the console crash report. If inaccurate, set the HSC outband error level to **INFO**. Then verify console report of date and time set by a host node on the next HSC reboot. If a host node problem is **NOT** indicated, escalate the problem to Field Service support.

ØØ1Ø43 ($CPWFL)
Power Failure

Facility: EXEC, EXEC

Explanation: After a power failure indication on the P.ioc, CRONIC will wait five seconds for power to diminish fully enough to stop the processor. If the processor is still operating five seconds after a power failure indication, CRONIC concludes that the powerfail indication was bogus.

User Action: Verify the dc voltages are correct. If so, and the problem persists, notify Field Service support.

ØØ12Ø1 ($CNOHIBER)
Process on Recoverable List not Hibernating

Facility: EXEC, EXECLOAD

Explanation: When requested to load a utility or diagnostic, the Loader first examined the Recoverable Memory List of cached programs to determine whether a program might be loaded from memory instead of from the load device. When the program was indeed found on the Recoverable Memory List, its state was not Hibernate State. This software inconsistency should not be seen under normal circumstances.

User Action: Submit an SPR with a dump noting previous activity with the program requested. R3 points to PCB (Process Control Block) for process to restart.

ØØ12Ø2 ($CIMAGE)
Memory extent encroaches defined area

Facility: EXEC, EXECLOAD

Explanation: The process to be loaded specified required additional memory for buffer space, as specified on the LFHEADER (Loadable File Header) directive. When the additional memory was allocated and mapped to the process, it had encroached upon the loaded area. This software inconsistency should not appear under normal circumstances.

User Action: Submit an SPR with a dump. RØ points to XFRB (Extended Function Request Block) for loading the image. R4 points to CH$ (Canonical File Header).

ØØ12Ø3 ($CNOPROC)
No code parent process loaded

Facility: EXEC, EXECLOAD

Explanation: When a process was loaded, its PCB (Process Control Block) specified it should execute and share code associated with another process. When attempting to locate the code parent, the loader found the parent was not loaded. This software inconsistency should not appear under normal circumstances.

User Action: Submit an SPR with a dump. R2 equals Process Number of code parent. R3 points to code child's PCB.

ØØ12Ø4 ($CALLOCATE)
Insufficient Kernel Pool

Facility: EXEC, EXECLOAD

Explanation: When attempting to allocate either a PCB (Process Control Block--Z.) or an Address Descriptor (A.) structure from Kernel Pool for a new process, Kernel Pool was inadequate to support the additional structures.

User Action: Submit an SPR with a dump.

ØØ12Ø5 ($CLFAO)
FAO overrun

Facility: EXEC, EXECLOAD

Explanation: When formatting a module version mismatch message, the string returned from FAO was too large for the buffer. This software inconsistency should not appear under normal circumstances.

User Action: Submit an SPR with a dump. If possible, send a copy of the load medium.

001401 ($CBUSY)
Performed receive when already busy with request

Facility: EXEC, EXECRDWR

Explanation: The READ$/WRITE$ service is single-threaded, handling only one request at a time. The service, while in its exception routine, was already busy with one request while a RCV$P operation was performed.

User Action: Submit an SPR with a dump.

001402 ($CNOLOADED)
Requested driver not loaded

Facility: EXEC, EXECRDWR

Explanation: A process within the HSC specified a READ$ or WRITE$ operation with a DDCB (Device Control Block) for a device not configured on that model. For example, a program specified a transfer for a TU58 on an HSC70 model. Because the device is not configured on the system, the driver is not loaded.

User Action: Submit an SPR with a dump, describing activity on the HSC at the time of the exception. The process listed as active may be the READ$/WRITE$ service, and not the process which performed the offending request. R3 points to XFRB (Extended Function Request Block). R4 points to DDCB. R5 equals CSR for device.

001403 ($CDDCB)
Invalid DDCB specified

Facility: EXEC, EXECRDWR

Explanation: A request to the READ$/WRITE$ service specified a DDCB (Device Control Block) that was invalid (or specified an invalid device type in the DD$TYPE field).

User Action: Submit an SPR with a dump, describing activity on the HSC at at the time of the exception. The process listed as active may be the READ$/WRITE$ service, and not the process which performed the offending request. R3 points to XFRB (Extended Function Request Block). R4 points to DDCB. R5 equals CSR for device. R0 equals Device Type.

001501
Software Inconsistency - Motor not Running

Facility: EXEC, EXECRX33

Explanation: The motor was not running when the Motor Shutdown Timer expired.

User Action: Submit an SPR with a crash dump.

001502
Software Inconsistency - Non-RX33 command requested

Facility: EXEC, EXECRX33

Explanation: An XFRB (CRONIC transfer request) was received by the RX33 driver, but specified a DDCB (Device Control Block) for a non-RX33 device. R4 points to DDCB, R5 points to XFRB (Extended Function Request Block).

User Action: Submit an SPR with a crash dump.

001503
Software Inconsistency - Invalid Unit Number

Facility: EXEC, EXECRX33

Explanation: The DDCB (Device Control Block) specified an RX33 device, but the unit requested was not 0 or 1. R5 points to XFRB (Extended Function Request Block).

User Action: Submit an SPR with a crash dump.

001504
Software Inconsistency - Zero byte count transfer

Facility: EXEC, EXECRX33

Explanation: A transfer was requested with a zero byte count.

User Action: Submit an SPR with the crash dump. R2 equals byte count, R5 points to XFRB (Extended Function Request Block).

001505
Software Inconsistency - Invalid byte count

Facility: EXEC, EXECRX33

Explanation: A transfer was requested with a byte count that was not a multiple of 512 (sector size). R2 equals byte count, R5 points to XFRB (Extended Function Request Block).

User Action: Submit an SPR with a crash dump.

001506
Software Inconsistency - Invalid internal byte count

Facility: EXEC, EXECRX33

Explanation: Remaining byte count of a partially completed transfer was not a multiple of 512 (sector size). The original (requested) byte count was a multiple of 512. R2 equals byte count, R5 points to XFRB (Extended Function Request Block).

User Action: Submit an SPR with a crash dump.

001507
Software/Hardware Inconsistency - RX33 hardware registers are incorrect

Facility: EXEC, EXECRX33

Explanation: RX33 hardware signaled successful completion of an I/O operation, but the hardware registers (current sector, current track, or memory address register) did not contain the expected values.

User Action: The most probable candidates are M.std2 and the RX33 drives. If the problem persists, submit an SPR with crash dumps.

001510
Software Inconsistency - Invalid Head Select

Facility: EXEC, EXECRX33

Explanation: Software attempted to select a head other than 0 or 1. R0 equals head select.

User Action: Submit an SPR with a crash dump.

001511
Software Inconsistency - Memory Management

Facility: EXEC, EXECRX33

Explanation: Relocation is not enabled in the memory management hardware. (Bit 0 not set in MMR0.)

User Action: Submit an SPR with a crash dump.

001512
Software Inconsistency - Invalid Virtual Address

Facility: EXEC, EXECRX33

Explanation: The virtual address passed in the XFRB is not in page 4. R5 points to XFRB (Extended Function Request Block).

User Action: Submit an SPR with a crash dump.

ØØ1513
Software/Hardware Inconsistency - Unexpected Interrupt from RX33

Facility: EXEC, EXECRX33

Explanation: An unexpected interrupt was received from the RX33 controller. This condition is not detected until a command is about to be issued (i.e., the crash does not happen when the interrupt is detected).

User Action: If problem persists, submit an SPR with crash dumps. Further testing of the subsystem (load device area) may be necessary.

ØØ1514
Software Inconsistency - Invalid Internal Unit Number

Facility: EXEC, EXECRX33

Explanation: The unit number index value is not Ø or 2. This unit number index value is contained in R4.

User Action: Submit an SPR with a crash dump.

ØØ1515
Software/Hardware Inconsistency - Non-Existent Memory

Facility: EXEC, EXECRX33

Explanation: RX33 controller returned an NXM error.

User Action: Further testing of the HSC subsystem (load device area) may be necessary. If problem persists, submit an SPR with crash dumps.

ØØ16Ø1 ($CPAG1)
TYPE$ crosses page boundaries

Facility: EXEC, EXECTT

Explanation: A process requested a TYPE$ System Service (or an ACPT$ Service with a prompt) specifying a buffer which crosses a memory management page boundary. This is a restriction of the driver.

User Action: Submit an SPR with a dump, describing activity at the time of the exception. RØ equals size of print string. R1 points to String Buffer. R4 points to TTCB (Device Control Block). R5 points to XFRB (Extended Function Request Block).

ØØ16Ø2 ($CPAG2)
ACPT$ crosses page boundary

Facility: EXEC, EXECTT

Explanation: A process requested an ACPT$ System Service specifying a buffer which crosses a memory management page boundary. This is a restriction of the driver.

User Action: Submit an SPR with a dump, describing activity at the time of the exception. R4 points to TTCB (Device Control Block). R5 points to XFRB (Extended Function Request Block).

ØØ16Ø3 ($CNOPCB)
PCB not found on run queue

Facility: EXEC, EXECTT

Explanation: When a process attached to a terminal is excepted by a keyboard command, the exception manager of terminal service first performs an EXCPT$ on the Terminal Service and load device driver. To prevent the attached process from running while the drivers potentially run down any activity, the PCB (Process Control Block) for the active process is removed from the run queue. When searching the run queue specified in the

Z.RUNQ field of the PCB, the PCB itself was not found. This is a software inconsistency.

User Action: Submit an SPR with a dump. R4 points to attached PCB.

001701 ($CPAGE)
READ$ or WRITE$ crossed page boundary

Facility: EXEC, EXECTU58

Explanation: A request to the TU58 driver specified a buffer which crossed a memory management page boundary. This is a restriction of the driver.

User Action: Submit an SPR with a dump, describing activity at the time of the exception. The process listed as active may be the READ$/WRITE$ service and not the process which initiated the offending request.

002001
Exception routine invoked for unknown reason

Facility: DEMON

Explanation: Demon's exception routine was activated, but not for CTRL Y, CTRL C, or a diagnostic timeout. A software problem is the most likely cause of this crash.

User Action: Submit an SPR with the crash dump. If a certain sequence of HSC operations induced this crash, include a description of that sequence.

002002
Insufficient free memory to allocate a program stack

Facility: DEMON

Explanation: When DEMON was initialized, it could not allocate enough free program memory for use as a stack. A failing memory module is the most likely cause of the problem.

User Action: A failing memory module is the most likely cause of the problem. If no hardware problem is found, submit an SPR and the crash dump. If a certain sequence of operations causes this crash, include a description of that sequence.

002003
DEMON was initiated when there was no diagnostic to run

Facility: DEMON

Explanation: DEMON did a receive on its work queue and received a nondiagnostic request. A software problem is the most probable cause of this crash.

User Action: Submit an SPR and the crash dump. If a certain sequence of HSC operations induced this crash, include a description of that sequence.

002004
Failure in periodic control or data memory test

Facility: DEMON, PRMEMY

Explanation: One of the periodic control or data memory interface tests detected a failure. Failures in these tests are fatal, and the HSC must reboot after displaying a message describing the failure. A failing P.ioc module is the most probable cause of this crash.

User Action: Further testing of the HSC memory and P.ioj may be necessary.

002005
Failure in periodic K.sdi or K.sti test

Facility: DEMON, PRKSDI, PRKSTI

Explanation: The periodic K.sdi test or the periodic K.sti test detected a failure. Failures in either test are fatal, and the HSC must reboot after displaying a message which describes the type of error and the requestor number of the failed module. A failing K.sdi or K.sti module is the most probable cause of this crash.

User Action: The requestor number of the probable failing module is displayed in the error message preceding the crash. Further testing of HSC data channels and HSC internal buses may be necessary.

002006
ILDISK received illegal queue address

Facility: DEMON, ILDISK

Explanation: ILDISK requested exclusive access to a drive's state area. The acquire operation should return the control memory address of the Attention/Available Service Queue for the specified drive. The address returned was zero, an illegal address for a queue. A software problem is the most likely cause of this crash.

User Action: Submit an SPR and the crash dump. If a certain sequence of HSC operations induced this crash, include a description of that sequence. Also note if the problem occurs only when a particular disk drive is tested.

002007
ILDISK received illegal buffer descriptor

Facility: DEMON, ILDISK

Explanation: ILDISK received a buffer descriptor from the free buffer queue. A consistency check on the buffer descriptor failed because the descriptor indicated the buffer was not in the HSC's buffer memory. A software problem is the most likely cause of this crash.

User Action: Submit an SPR which includes the crash dump information. If a certain sequence of HSC operations induced this crash, include a description of that sequence. Also note if the problem occurs only when a particular disk drive is tested.

002010
ILDISK detected inconsistency in exception routine

Facility: DEMON, ILDISK

Explanation: ILDISK's internal flags indicated exclusive ownership of a drive's state area, but the address of the K.sdi control area was not available. When ILDISK has exclusive ownership of a drive state area, the address of the K.sdi control area should always be available. A software problem is the most likely cause of this crash.

User Action: Submit an SPR and the crash dump. If a certain sequence of HSC operations induced this crash, include a description of that sequence. Also note if the problem occurs only when a particular disk drive is tested.

002011
An ILEXER disk I/O request failed to complete

Facility: DEMON, ILEXER

Explanation: ILEXER attempted to abort all outstanding disk I/O requests. After waiting two minutes, the program found one or more I/O requests uncomplete. The HSC is crashed and rebooted because ILEXER cannot exit with a request outstanding. A faulty disk drive is the most likely cause of this problem.

User Action: Submit an SPR and the crash dump. If a certain sequence of HSC operations induced this crash, include a description of that sequence. Also note if the problem occurs only when a particular disk drive is tested. Further testing of suspect disk and associated requestor(s) may be necessary.

002012
An ILEXER tape I/O request failed to complete

Facility: DEMON

Explanation: ILEXER attempted to abort all outstanding tape I/O requests. After waiting two minutes, the program found one or more I/O requests uncomplete. The HSC is crashed and rebooted because ILEXER cannot exit with a request still outstanding. A faulty tape drive or formatter is the most likely cause of this problem. This crash could also be caused by the K.sti clocks stopping due to a hardware error (such as an Instruction Parity error).

User Action: Submit an SPR and the crash dump. If a certain sequence of HSC operations induced this crash, include a description of that sequence. Also note if the problem occurs only when a particular tape drive or formatter is tested. Further testing of suspect tape subsystem and associated requestor(s) may be necessary.

002013
ILTAPE was supplied an illegal requestor number

Facility: DEMON, ILTAPE

Explanation: ILTAPE was automatically initated to test a particular formatter. One of the parameters supplied to ILTAPE is the requestor number of the K.sti connected to the formatter. ILTAPE checked the specified requestor and found it was not a K.sti. A software problem is the most likely cause of this crash.

User Action: Submit an SPR and the crash dump. Also include a summary of any tape error messages immediately preceding the crash. If a certain sequence of HSC operations induced this crash, include a description of that sequence. Also note if the problem occurs only when a particular tape drive or formatter is used.

002014
ILTAPE timed-out waiting for drive state area

Facility: DEMON, ILTAPE

Explanation: In order to test a tape formatter, ILTAPE must acquire exclusive access to the drive state area for that formatter. When ILTAPE requests exclusive access to a drive state area, the request should complete within 60 seconds. Failure to complete indicates a problem with the tape server.

User Action: Submit an SPR and the crash dump. Also include a summary of any tape error messages immediately preceding the crash. If a certain sequence of HSC operations induced this crash, include a description of that sequence. Also note if the problem occurs only when a particular tape drive or formatter is used.

002015
ILTAPE detected inconsistency after a command failure

Facility: DEMON, ILTAPE

Explanation: ILTAPE issued a command to the HSC tape diagnostic interface. The command failed. In the process of preparing an error message, ILTAPE found the command opcode was illegal, an unknown value. A software problem is the most likely cause of this crash.

User Action: Submit an SPR and the crash dump information. Also include a summary of any tape error messages immediately preceding the crash. If a certain sequence of HSC operations induced this crash, include a description of that sequence. Also note if the problem occurs only when a particular tape drive or formatter is used.

002016
ILTAPE detected inconsistency while restoring a TACB

Facility: DEMON, ILTAPE

Explanation: ILTAPE maintains a table of available Tape Access Control Blocks (TACBs). When a particular TACB is in use by the program, the associated table entry is zeroed. When finished with a TACB, ILTAPE stores the address of that TACB into one of the table entries which contains a zero. While trying to return a TACB to the table, ILTAPE discovered all table entries are nonzero implying no TACBs were in use. A software problem is the most probable cause of this crash.

User Action: Submit an SPR and the crash dump. Also include a summary of any tape error messages immediately preceding the crash. If a certain sequence of HSC operations induced this crash, include a description of that sequence. Also note if the problem occurs only when a particular tape drive or formatter is used.

002017
ILTAPE detected inconsistency in exception routine

Facility: DEMON, ILTAPE

Explanation: ILTAPE's internal flags indicated exclusive ownership of a drive's state area, but the address of the K.sdi control area was not available. When ILTAPE has exclusive ownership of a drive state area, the address of the K.sti control area should always be available. A software problem is the most likely cause of this crash.

User Action: Submit an SPR which includes the crash dump information. If a certain sequence of HSC operations induced this crash, include a description of that sequence. Also note if the problem occurs only when a particular tape drive is tested.

003001 ($CFMTTYP)
Illegal format type specified

Facility: CERF

Explanation: An illegal format type was specified in an error message to CERF.

User Action: Submit an SPR with a dump. R4 equals Format Type.

003002 ($CFAO1)
Output length too long

Facility: CERF

Explanation: When processing an MSCP error message, the FAO output of the text string was too long for CERF's buffer.

User Action: Submit an SPR with a dump. R1 equals number of bytes output.

003003 ($CFAO2)
Output length too long

Facility: CERF

Explanation: When processing an out of band message, the FAO output of the text string was too long for CERF's buffer.

User Action: Submit an SPR with a dump. R1 equals number of bytes output.

004001
No structure to ONLINE disk to connection

Facility: DISK, MSCX

Explanation: When an MSCP ONLINE command was issued to bring a disk online to a connection, there was no structure to record the necessary information. Since the initialization code allocates enough structures to bring every disk online to every connection, this crash indicates either memory corruption or mismanagement of the free pool for this structure.

User Action: Submit an SPR with the crash dump. Specify the number of hosts in the cluster.

004002
BMB reserved but not found

Facility: DISK, many

Explanation: A Big Memory Buffer (BMB) was reserved via a system function but not found when the table of BMBs was searched. This indicates memory corruption or mismanagement of the BMB pool.

User Action: Submit an SPR with the crash dump. Specify which process was running.

004003
DUCB address zero in K Control Area

Facility: DISK, SDI

Explanation: A disk attention condition sent a control area to a disk subprocess. The subprocess found a zero in the location which should have contained the DUCB address. This indicates an invalid structure address was passed to the process (possibly due to memory corruption), the structure was corrupted, or it was not initialized properly.

User Action: Submit an SPR with the crash dump.

004004
Invalid action byte in Connect Block

Facility: DISK, SDI

Explanation: The subprocess within the disk path which processes requests from the CI manager received a Connect Block with an invalid action byte. This indicates an invalid structure was passed to the process, the structure was passed at the improper time, or that memory was corrupted.

User Action: Submit an SPR with the crash dump. Note the contents of user register 2 in the crash dump.

004005
Datagram received from a connection

Facility: DISK, MSCP

Explanation: The main MSCP command server process received a nonsequenced message from some connection. This may indicate memory corruption or improper message reception. It may also indicate an improper structure was passed to the process. Host software may have improperly sent such a message.

User Action: Submit an SPR with the crash dump. Note all levels of host software running in the cluster.

004006
MSCP message size exceeded maximum

Facility: DISK, MSCP

Explanation: The main MSCP command server process received a sequenced message with a length greater than the MSCP 36-byte maximum from some connection. This may indicate memory corruption or improper message reception. It may also indicate an improper structure was passed to the process. Host software may have improperly sent such a message.

User Action: Submit an SPR with the crash dump. Note all levels of host software running in the cluster.

004007
Invalid error signaled by K.ci

Facility: DISK, MSCP

Explanation: An MSCP command packet with invalid error bits set was received by the main MSCP command server from the K.ci. This may indicate memory corruption or improper message reception. It may also indicate an improper structure was passed to the process. Host software may have sent an improper message.

User Action: Submit an SPR with the crash dump. Note all levels of host software running in the cluster and the revision level of the K.ci microcode.

004010
Server queue on work queue with no items

Facility: DISK, many

Explanation: The main disk process received a subprocess work queue with no items from the main work queue. This indicates either memory corruption or improper manipulation of items on the subprocess work queue. An invalid structure may have been queued to the main work queue.

User Action: Submit an SPR with the crash dump. Note the current process running.

004011
Invalid module number in subprocess work queue

Facility: DISK, many

Explanation: The main disk process received a subprocess work queue containing an invalid module number. This indicates memory corruption or an invalid structure was queued to the main work queue.

User Action: Submit an SPR with the crash dump. Note the current process running.

004012
SLCB not available when needed

Facility: DISK, SDI

Explanation: A Short Lifetime Control Block (SLCB) was needed by the disk path but one was not available. Because many processes and subprocesses require SLCBs, this is unlikely except under extreme load circumstances. The number SLCBs allocated by default should be sufficient to avoid this crash.

User Action: Submit an SPR with the crash dump. Note the configuration of the HSC and the number of disk and tape drives online at the time of the crash.

004013
State change to ONLINE requested via gatekeeper

Facility: DISK, SDI

Explanation: The state change processor within the sequential command gatekeeper received a Disk Unit Control Block extension with the current state set to online. This crash indicates an improper use of the state change mechanism.

User Action: Submit an SPR with the crash dump.

004014
Inconsistent drive state detected

Facility: DISK, SDI

Explanation: The state change processor within the sequential command gatekeeper received a Disk Unit Control Block extension different than the current state. This crash indicates an improper use of the state change mechanism.

User Action: Submit an SPR with the crash dump.

004015
Improper state change for shadow member

Facility: DISK, SDI

Explanation: The sequential gatekeeper mechanism suspends activity for shadow units before allowing a state change. This crash indicates the mechanism failed to operate properly.

User Action: Submit an SPR with the crash dump.

004016
Shadow unit not found in Disk Unit Table

Facility: DISK, MSCP

Explanation: The subroutine SUREM could not find the shadow unit in the Disk Unit Table. This crash indicates improper sequencing of actions to remove a shadow unit. The most probable cause is multiple calls on SUREM for the same unit.

User Action: Submit an SPR with the crash dump.

004017
Invalid diagnostic HMB

Facility: DISK, MSCP

Explanation: The diagnostic interface within the disk path received an HMB with a nonzero length field in the HM$LOF word. This indicates an invalid request from some diagnostic or improper routing of the HMB by the disk path.

User Action: Submit an SPR with the crash dump. List any utilities or diagnostics running at the time of the crash.

004020
Too many seek blocks requested by diagnostic

Facility: DISK, MSCP

Explanation: A diagnostic or utility requested an excessive number of seek blocks for transfers during initialization.

User Action: Submit an SPR with the crash dump. List any utilities or diagnostics running at the time of the crash.

004021
Diagnostic release of disk unit while online

Facility: DISK, MSCP

Explanation: A diagnostic or utility attempted to release a disk unit while it was still online.

User Action: Submit an SPR with the crash dump. Specify the utilities or diagnostics running at the time of the crash.

004022
Diagnostic release of HCB while units still online

Facility: DISK, MSCP

Explanation: A diagnostic or utility attempted to release a Host Control Block (HCB) which keeps records of online units, while some disk units were online via that HCB.

User Action: Submit an SPR with the crash dump. Specify the utilities or diagnostics executing at the time of the crash.

004023
DRAT/Seek timer not allocated for disk unit

Facility: DISK, ERROR

Explanation: The disk path initialization code discovered a disk unit without a DRAT/Seek timer allocated (address of zero). This is an initialization inconsistency, possibly due to an improper load of the disk path.

User Action: Submit an SPR with the crash dump. Specify the configuration of the HSC which crashed.

004024
Not enough mapped memory to initialize disk path

Facility: DISK, ERROR

Explanation: The disk path initialization routine could not allocate enough program memory to perform error recovery. The most probable cause is insufficient available memory.

User Action: Determine the amount of available program (P.ioj) memory. If it is lower than the minimum amount, replace the memory module. If the memory appears to be sufficient, submit an SPR with the crash dump. Note the actual amount of available memory by executing the SHOW ALL command. If no hardware problem exists, submit an SPR with a printout of SHOW ALL command results.

004025
Error identification table overwritten

Facility: DISK, ERROR

Explanation: This crash can only occur if the disk error identification table was overwritten or a wild branch was taken. The most probable cause is a bad load.

User Action: If this crash occurs immediately after a boot, try rebooting with a backup copy of the HSC software. Otherwise, submit an SPR with the crash dump.

004026
Invalid error bit value found during error recovery

Facility: DISK, ERROR

Explanation: The bit value describing a K.sdi error was not valid for a given stage of the error recovery. The most probable cause is a design error within the error recovery code. It is possible, although unlikely, the cause is a malfunctioning K.sdi.

User Action: If this error appears to recur from the same K.sdi, replace the K.sdi. If no hardware problem exists, submit an SPR with the crash dump.

004027
Invalid disk characteristics for operation

Facility: DISK, ERROR

Explanation: An arithmetic operation to compute some disk parameter caused an overflow or produced a result outside the allowed range. The most probable cause is a design error within the error recovery code. It is also possible, although unlikely, a disk supplied invalid characteristics to the HSC.

User Action: If possible, get the number of the requestor involved from the last error log printed on the console or from the system error log. Further testing of the disk and attached requestor(s) may be necessary. If this error appears to recur from the same unit, repair the unit. If no hardware problem exists, submit an SPR with the crash dump.

004030
S bit not set in FRB error state

Facility: DISK, ERROR

Explanation: The S bit in the K control area port subarea for a drive in FRB error state was not set as expected. This logical inconsistency indicates improper manipulation of the port state. The most probable cause is a design error within the error recovery code. It is also possible, although unlikely, a K.sdi is malfunctioning.

User Action: If possible, get the number of the requestor involved from the last error log printed on the console or from the system error log. Further testing of suspected requestor may be necessary. If this error appears to recur from the same K.sdi, replace the K.sdi. If no hardware problem exists, submit an SPR with the crash dump.

004031
DT$ERQ not zero in FRB error state

Facility: DISK, ERROR

Explanation: The FRB error queue in the DRAT being processed by error recovery was not zero as expected. This logical inconsistency indicates improper manipulation of the port state. The most probable cause is a design error within the error recovery code. It is also possible, although unlikely, a K.sdi is malfunctioning.

User Action: If possible, get the number of the requestor involved from the last error log printed on the console or from the system error log. Further testing of suspected requestor may be necessary. If this error appears to recur from the same K.sdi, replace the K.sdi. If no hardware problem exists, submit an SPR with the crash dump.

004032
Unable to get to FRB error state

Facility: DISK, ERROR

Explanation: Error recovery was unable to place a port in FRB error state in order to perform an error recovery operation. This crash can occur in an extremely unlikely compound error situation. The most probable cause, however, is a design error within the error recovery code.

User Action: Reboot the HSC. If this error persists, submit an SPR with the crash dump.

004033
Non-ECC/EDC errors remaining after ECC correction

Facility: DISK, ERROR

Explanation: ECC error correction should take place after all other errors except EDC have been corrected. This crash occurs because other error bits are set after ECC correction. The most probable cause is a design error within the error recovery code.

User Action: Submit an SPR with the crash dump.

004034
Level B retry in wrong state

Facility: DISK, ERROR

Explanation: This crash occurs because a level B retry operation is attempted without the drive port being in FRB error state. The only cause is a design error within the error recovery code.

User Action: Submit an SPR with the crash dump.

004035
Level C retry in wrong state

Facility: DISK, ERROR

Explanation: This crash occurs because a level C retry operation is being attempted without the drive port being in FRB error state. The only cause is a design error within the error recovery code.

User Action: Submit an SPR with the crash dump.

004036
DCB state is busy with empty DCB queue

Facility: DISK, ERROR

Explanation: The drive state indicator in the K control area indicates a K.sdi is processing a DCB, but the DCB queue is empty. The most probable cause of this crash is a design error in the error recovery code. It is also possible, but unlikely, that the K.sdi is malfunctioning.

User Action: If possible, get the number of the requestor involved from the last error log printed on the console or from the system error log. Further testing of suspect requestor may be necessary. If this error appears to recur from the same K.sdi, replace the K.sdi. If no hardware problem exists, submit an SPR with the crash dump.

004037
Invalid error queue address in route

Facility: DISK, ERROR

Explanation: When attempting to route an FRB to an error queue, the error queue address in a route descriptor was invalid. The most likely cause of this crash is a corrupted route descriptor probably due to a logic error in the error recovery code.

User Action: Submit an SPR with the crash dump.

004040
Undefined error bit in error word from K

Facility: DISK, ERROR

Explanation: The error recovery routine IDENTIFY found an undefined bit in the error word stored by either a K.sdi or K.ci. The most probable cause of this crash is a logic error within the error recovery code. It is also possible but unlikely a K is malfunctioning.

User Action: If possible, get the number of the requestor involved from the last error log printed on the console or from the system error log. Further testing of suspect requestor may be necessary. If this error appears to recur from the same K.sdi, replace the K.sdi. If no hardware problem exists, submit an SPR with the crash dump.

004041
No buffer found in FRB when expected

Facility: DISK, ERROR

Explanation: The error recovery routine MAPBUF attempted to map a buffer but found the buffer address to be zero. The only cause of this crash is a design error within the error recovery code.

User Action: Submit an SPR with the crash dump.

004042
FRB not in error state for level D I/O operation

Facility: DISK, ERROR

Explanation: A call to the error recovery subroutine LVLDIO was made without the port being in FRB error state. The only cause of this logical inconsistency is a design error within the error recovery code.

User Action: Submit an SPR with the crash dump.

004043
Stack too deep to save in thread block

Facility: DISK, ERROR

Explanation: A call to the error recovery subroutine LVLDIO was made with too many items on the stack to save in a thread block. The only cause of this logical inconsistency is a design error within the error recovery code.

User Action: Submit an SPR with the crash dump.

004044
Buffer not found for specified error

Facility: DISK, ERROR

Explanation: A call to the error recovery subroutine RCDHMX specified a buffer which was not found in the list of buffers for the specified FRB. The only cause of this logical inconsistency is a design error within the error recovery code.

User Action: Submit an SPR with the crash dump.

004045
Parent downcount failed

Facility: DISK, ERROR

Explanation: A downcount of the parent HMB failed during routing of an FRB in the error recovery subroutine RETIRE. This crash is caused by improper manipulation of the parent counter by some process or overwritten memory.

User Action: Submit an SPR with the crash dump.

004046
DRAT not found for FRB retirement

Facility: DISK, ERROR

Explanation: The error recovery subroutine RETIRE could not locate the DRAT for down counting while attempting to retire an FRB by simulating route completion. This crash is caused by either a logic inconsistency within error recovery or overwritten memory. It is also possible, although unlikely, it is caused by a malfunctioning K.sdi.

User Action: If possible, get the number of the requestor involved from the last error log printed on the console or from the system error log. Further testing of requestors and HSC internal buses may be necessary. If this error appears to recur from the same K.sdi, replace the K.sdi. If no hardware problem exists, submit an SPR with the crash dump.

004047
Sectors/track field in K control area is zero

Facility: DISK, ERROR

Explanation: The error recovery subroutine RETIRE found the sectors/track field in the K control area to be zero while attempting to retire an FRB by simulating route completion. This crash is caused by either a logic inconsistency within error recovery or overwritten memory. It is also possible, although unlikely, it is caused by a malfunctioning K.sdi.

User Action: If possible, get the number of the requestor involved from the last error log printed on the console or from the system error log. If this error appears to recur from the same K.sdi, replace it. If no hardward problem exists, submit an SPR with the crash dump.

004050
DRAT queue not empty for shadow copy

Facility: DISK, MSCP

Explanation: After obtaining exclusive use of a drive, the shadow copy code found a DRAT queue for that drive was not empty. This crash can only be caused by a design error within the MSCP command processing.

User Action: Submit an SPR with the crash dump.

004051
Inconsistent result for repair operation

Facility: DISK, MSCP

Explanation: An impossible combination of results was found at the end of a shadow repair operation. This crash can only be caused by a design error within the shadow repair code.

User Action: Submit an SPR with the crash dump.

004052
Known drive not found in the Disk Unit Table

Facility: DISK, MSCP

Explanation: While attempting to remove a known disk unit from the Disk Unit Table, the unit was not found in that table. This crash can only be caused by a design error within the MSCP command processing.

User Action: Submit an SPR with the crash dump. Note any utilities or diagnostics running at the time of the crash.

004053
Invalid block number for transfer operation

Facility: DISK, MSCP

Explanation: All MSCP transfer commands are prechecked for valid parameters. This applies to most diagnostic transfers as well. This crash indicates an invalid block number somehow slipped past the checks. It indicates a design error within the disk path transfer processing or a corrupted Disk Unit Control Block.

User Action: Submit an SPR with the crash dump. Note any utilities or diagnostics running at the time of the crash.

004054
Unexpected compare failure following write

Facility: DISK, ERROR

Explanation: The RCT.MWRITE routine writes, reads back, and one at a time compares a block of data to all copies of that block in the RCT. This crash indicates a block was read back with no errors detected; however, it did not compare with the original data written. This indicates data was delivered incorrectly by the K.sdi without any error indications. It is possible, but unlikely, the failure is due to a legitimate undetected error.

User Action: If possible, get the number of the requestor involved from the last error log printed on the console or from the system error log. If this error appears to recur from the same unit, repair the unit. If no hardware problem exists, submit an SPR with the crash dump.

004055
Attempt to enable drive interrupt already enabled

Facility: DISK, many

Explanation: The ARM subroutine was called to enable the interrupt for drive state changes when the interrupt was already enabled. The only possible cause for this crash is a design error.

User Action: Submit an SPR with the crash dump. Note the process running at the time of the crash.

004056
Attempt to enable drive interrupt with pending state change

Facility: DISK, many

Explanation: The ARM subroutine was called to enable the interrupt for drive state changes while a drive state change was being processed. The only possible cause for this crash is a design error.

User Action: Submit an SPR with the crash dump. Note the process running at the time of the crash.

004057
State change requested for available but inoperative drive

Facility: DISK, many

Explanation: The SCHSQM subroutine was called to schedule a state change operation for an available but inoperative drive. The only possible cause for this crash is a design error.

User Action: Submit an SPR with the crash dump. Note the process running at the time of the crash.

004060
Attempt to down count DRAT already at zero

Facility: DISK, many

Explanation: A call was made to the DWNCDT subroutine to down count a DRAT when the count was already zero. The only possible cause for this crash is a design error.

User Action: Submit an SPR with the crash dump. Note the process running at the time of the crash.

004061
Thread block count not initialized

Facility: DISK, SDI

Explanation: During initialization, the routine which allocates thread blocks discovered the number of threads to be allocated was set to zero. This was probably caused by the failure of a previous initialization routine to initialize this count word. This inconsistency may indicate an improper load.

User Action: Reboot the HSC. If the failure persists, submit an SPR with the crash dump.

004062
Thread block area too small

Facility: DISK, SDI

Explanation: During initialization, the routine which carves up thread blocks found the area too small to allocate all the thread blocks required. This inconsistency may indicate an improper load.

User Action: Reboot the HSC with a backup copy of the HSC system software. If the failure persists, submit an SPR with the crash dump.

004063
Seek DCB without Clear D Bit flag set

Facility: DISK, SDI

Explanation: A SEEK DCB failed because the Clear D Bit flag was not set as expected. The DCB was not a SEEK DCB or the DCB was improperly set up. The only possible cause is a design error within the DCB processing code.

User Action: Submit an SPR with the crash dump.

004064
DRAT/SEEK timer running with SEEK DCB queued

Facility: DISK, SDI

Explanation: During processing a failed SEEK DCB, the DRAT/SEEK timer was not running as expected. The only possible cause is a design error within the DCB processing code.

User Action: Submit an SPR with the crash dump.

004065
D Bit set for port with SEEK DCB being processed

Facility: DISK, SDI

Explanation: During processing of a failed SEEK DCB, the D (Process DRAT) bit was set for the port to which the SEEK DCB had been queued. The only possible cause is a design error within the DCB processing code.

User Action: Submit an SPR with the crash dump.

004066
State changed during SDI ONLINE

Facility: DISK, SDI

Explanation: After completing an SDI ONLINE command, either the state was not AVAILABLE or a state change was pending. Because state changes are inhibited during the SDI ONLINE, this is a logical inconsistency. The only possible cause is a design error within the SDI manager.

User Action: Submit an SPR with the crash dump.

004067
SDI WRITE MEMORY command not implemented

Facility: DISK, SDI

Explanation: The SDI WRITE MEMORY command cannot be issued in the current implementation of the SDI manager. This crash indicates some process attempted to issue an SDI WRITE MEMORY command.

User Action: Submit an SPR with the crash dump. Note the diagnostics or utilities running at the time of the crash.

004070
Nonzero status for SUCCESSful DCB

Facility: DISK, SDI

Explanation: A DCB completed with a status of SUCCESS, but the error word indicated errors anyway. The most probable cause is a design error within DCB processing. It is possible, although unlikely, the cause is a malfunctioning K.sdi. This is a logical inconsistency.

User Action: If possible, get the number of the requestor involved from the last error log printed on the console or from the system error log. If this error appears to recur from the same K.sdi, replace the K.sdi. If no hardware problem exists, submit an SPR with the crash dump.

004071
D bit set in DCB error state

Facility: DISK, SDI

Explanation: During processing of a DCB, the D (Process DRAT) bit was set for the port to which the DCB had been queued. This is a logical inconsistency. The only possible cause is a design error within DCB processing.

User Action: Submit an SPR with the crash dump.

004072
DCB state is busy with empty DCB queue

Facility: DISK, SDI

Explanation: The drive state indicator in the K control indicates a DCB is being processed by the K.sdi but the DCB queue is empty. The most probable cause of this crash is a design error within DCB processing. It is also possible, but unlikely, the K.sdi is malfunctioning.

User Action: If possible, get the number of the requestor involved from the last error log printed on the console or from the system error log. Further testing of requestors, HSC internal buses, and memory subsystem may be necessary. If this error appears to recur from the same K.sdi, replace it. If no hardware problem exists, submit an SPR with the crash dump.

004073
K.sdi is not responding

Facility: DISK, SDI

Explanation: A K.sdi failed to process an immediate DCB within one second. The most probable cause is a broken K.sdi.

User Action: If possible, get the number of the requestor involved from the last error log printed on the console or from the system error log. If the error persists, replace the K.sdi.

004074
DCB state is blocked after QUIESCE or DCBSTS DCB

Facility: DISK, SDI

Explanation: The drive state indicator in the K control indicates DCB activity is blocked. This should not be possible after a QUIESCE or DCBSTS DCB. The most probable cause of this crash is a design error within DCB processing. It is also possible, but unlikely, the K.sdi is malfunctioning.

User Action: If possible, get the number of the requestor involved from the last error log printed on the console or from the system error log. Further testing of requestors, HSC internal buses, and memory subsystem may be necessary. If this error appears to recur from the same K.sdi, replace the K.sdi.

004075
Call to DCBOPR from process other than DISK

Facility: DISK, many

Explanation: The DCBOPR routine may only be called from the DISK process. This crash indicates a call was made from some other process.

User Action: Submit an SPR with the crash dump. Note the process running at the time of the crash.

004076
Port not in DCB error state for error DCB

Facility: DISK, SDI

Explanation: The DCBOPR routine received an error DCB, but the port was not in DCB error state as expected. This logical inconsistency can only be the result of a design error within DCB processing.

User Action: Submit an SPR with the crash dump.

004077
Match enable not set for DIALOG DCB

Facility: DISK, SDI

Explanation: The DCBOPR routine received a DCB with an improper combination of request bits set. This logical inconsistency can only be the result of a design error within DCB processing.

User Action: Submit an SPR with the crash dump.

004100
No thread block for operation

Facility: DISK, SDI

Explanation: The DCBWAIT routine was called. Insufficient thread block was available to suspend the process. This logical inconsistency can only be the result of a design error within DCB processing.

User Action: Submit an SPR with the crash dump.

004101
Stack too deep to suspend process in thread block

Facility: DISK, SDI

Explanation: The DCBWAIT routine was called with too many words on the stack to suspend the process in a thread block. This logical inconsistency can only be the result of a design error within DCB processing.

User Action: Submit an SPR with the crash dump.

004102
Thread block pointer corrupted in DCB

Facility: DISK, SDI

Explanation: A DCB was returned from a K.sdi with a corrupted thread block pointer. The most probable cause of this crash is a design error within DCB processing. It is also possible but unlikely the cause is a malfunctioning K.sdi.

User Action: If possible, get the number of the requestor involved from the last error log printed on the console or from the system error log. If this error appears to recur from the same K.sdi, replace it. If no hardware problem exists, submit an SPR with the crash dump.

004103
Insufficient pool to allocate a timer

Facility: DISK, SDI

Explanation: This crash indicates too much memory has been allocated from common pool. It can be caused by any process.

User Action: Submit an SPR with the crash dump. Specify the diagnostics or utilities running at the time of the crash and if there were DUP connections.

004104
DCB received with no errors and no frames

Facility: DISK, SDI

Explanation: A DCB was received from K.sdi with no frames in the response and no error indications. It can be caused by a design error within DCB processing. It is probably caused by a malfunctioning K.sdi.

User Action: If possible, get the number of the requestor involved from the last error log printed on the console or from the system error log. If this error appears to recur from the

same K.sdi, replace it. If no hardware problem exists, submit an SPR with the crash dump.

004105
Element in deferred seek queue with no FRB

Facility: DISK, MSCP

Explanation: An element was found in the deferred seek queue for a disk unit having no FRBs. The only possible cause is a design error within seek queue element processing.

User Action: Submit an SPR with the crash dump.

005001
ECC self-test string too big for FAO

Facility: ECC

Explanation: A self-test string generated for the ECC process was too big to print with the FAO buffer allocated. This crash can only occur if the self-test code is present and enabled. The self-test code is not enabled for distributed base levels.

User Action: Submit an SPR with the crash dump.

005002
No ECC errors to correct

Facility: ECC

Explanation: An FRB with no errors was sent to the ECC process. This logical inconsistency can only occur due to a design error within error recovery.

User Action: Submit an SPR with the crash dump.

005003
Can't allocate XFRB to print self-test messages

Facility: ECC

Explanation: The ECC process failed to allocate an XFRB (Extended Function Request Block) for printing messages during self-test. This crash can only occur if the self-test code is present and enabled (not tru for distributed base levels).

User Action: Submit an SPR with the crash dump.

005004
ECC found more than a 10-bit symbol error

Facility: ECC

Explanation: The ECC process was sent a buffer with more than a 10-bit symbol error. Error recovery processing should never pass on such a buffer. This logical inconsistency can only occur due to a design error within error recovery.

User Action: Submit an SPR with the crash dump.

006000
This class of crashes is for tape path software inconsistency errors

Facility: TAPE, TFxxxx

Explanation: A software inconsistency error occurred.

User Action: Submit an SPR with the crash dump. Specify the utilities or diagnostics active at the time of the crash.

006001
An STI GET LINE STATUS failed

Facility: TAPE, TFATNAVL

Explanation: When issued to the tape data channel, the STI command GET LINE STATUS returned with a failure. This command should not return failure when issued to a working tape data channel. General register 5 points to the windowed K control area for the tape data channel in question. Offset KG$SLT points to the tape requestor in question.

User Action: Investigate the tape data channel in question.

006002
Received an interrupt from an unknown tape data channel

Facility: TAPE, TFATNAVL

Explanation: Received an interrupt from an unknown tape data channel. This is a software inconsistency. General register 1 points to the windowed tape data channel control area for the tape data channel in question. General register 2 contains the tape data channel slot number the interrupt was received from.

User Action: Submit an SPR with the crash dump.

006003
Received an illegal Connection Block (CB) from the CIMGR

Facility: TAPE, TFCI

Explanation: A Connection Block (CB) with an illegal opcode was sent to the tape path. General register 1 points to the windowed address of the Connection Block (CB) in question. General register 2 contains the opcode in question.

User Action: Submit an SPR with the crash dump. Include the Connection Block (CB) structure.

ØØ6ØØ4
An illegal diagnostic opcode was received

Facility: TAPE, TFDIAG

Explanation: A diagnostic Host Message Block (HMB) with an illegal opcode was sent to the tape diagnostic interface. General register 3 points to the windowed diagnostic Host Message Block (HMB). General register 1 contains the opcode in question.

User Action: Submit an SPR with the crash dump. Specify the utilities or diagnostics active at the time of the crash. Include the Host Message Block (HMB) structure.

ØØ6ØØ5
Diagnostics trying to acquire assigned drive state area

Facility: TAPE, TFDIAG

Explanation: Diagnostics are trying to acquire previously-assigned drive state area. General register 3 points to the windowed control memory address of the Host Message Block (HMB). General register 2 points to the Tape Formatter Control Block (TFCB).

User Action: Submit an SPR with the crash dump. Specify the diagnostics or utilities active at the time of the crash. Include the Host Message Block (HMB) and Tape Formatter Control Block (TFCB) structure.

006006
Inconsistencies during drive state area acquisition

Facility: TAPE, TFDIAG

Explanation: The software context word (SFW) is not equal to the Tape Formatter Control Block (TFCB) address and/or DIALOG list head is nonzero when diagnostics are trying to acquire the drive state area. General register 0 points to the windowed K control area. General register 2 points to the Tape Formatter Control Block (TFCB).

User Action: Submit an SPR with the crash dump. Indicate the utilities or diagnostics active at the time of the crash. Include the Tape Formatter Control Block (TFCB) structure.

006007
No Block Header supplied by BACKUP

Facility: TAPE, TFDIAG

Explanation: BACKUP did not supply the initial Block Header buffer descriptor. General register 3 points to the windowed Host Message Block (HMB) address. General register 5 should point to the buffer descriptor and, in this case, should be 0.

User Action: Submit an SPR with the crash dump. Include details of the BACKUP operation. Include the Host Message Block (HMB) structure.

ØØ6Ø1Ø
No buffers supplied in BACKUP operation

Facility: TAPE, TFDIAG

Explanation: No disk data block buffers were supplied in Host Message Block (HMB) for BACKUP operation. General register 3 points to the windowed control memory address of the Host Message Block (HMB) in question. General register Ø should point to the buffer descriptor list for the BACKUP operation (in this case does so).

User Action: Submit an SPR with crash dump. Include details of BACKUP operation. Include the Host Message Block (HMB) structure.

ØØ6Ø11
Could not allocate a XFRB

Facility: TAPE, TFLIB

Explanation: Could not allocate a XFRB (Extended Function Control Block) through ALOCB for print routine.

User Action: Submit an SPR with the crash dump.

ØØ6Ø12
Required CIMGR functionality not yet implemented

Facility: TAPE, TFMSCP

Explanation: The host sent the tape server a command packet with an opcode that was not a sequenced message. General register 5 is the opcode received. General register 3 is the windowed control memory address of the command packet received (Host Message Block (HMB)).

User Action: Submit an SPR with crash dump. Indicate the host software version . Include the Host Message Block (HMB) (command packet) structure.

006013
Required CIMGR functionality not yet implemented

Facility: TAPE, TFMSCP

Explanation: The tape server received a host command packet longer than allowed (36. bytes). General register 4 is the size of command packet received. General register 3 is the windowed control memory address of the command packet in question.

User Action: Submit an SPR with the crash dump. Indicate the host software version. Include the Host Message Block (HMB) (command packet) structure.

006014
Required CIMGR functionality not yet implemented

Facility: TAPE, TFMSCP

Explanation: The tape server received a host command packet with a status that currently is not executed. General register 3 points to the windowed control memory address of the command packet in question. Offset HM$ERR is the field in question.

User Action: Submit an SPR with the crash dump. Indicate the host software version. Further testing of HSC hardware may be necessary. Investigate K.ci. Include the Host Message Block (HMB) (command packet) structure. If no hardware problem exists, submit an SPR.

006015
Could not find correct Tape Drive Control Block (TDCB) pointer

Facility: TAPE, TFSEQUEN

Explanation: A call to remove a host's access to a drive resulted in searching the current chain of Tape Drive Control Blocks (TDCB) in that host's HCB. Inability to find the correct Tape Drive Control Block (TDCB) pointer resulted in this message. General register 4 points to the Tape Drive Control Block (TDCB) that is trying to have host access removed. General register 3 points to the windowed control memory address of the Host Message Block (HMB). Offset HM$CTX in the Host Message Block (HMB) points to the Host Disk Block (HDB). Offset HDB.TDCB in the HDB points to the Tape Drive Control Block (TDCB).

User Action: Submit an SPR with the crash dump. Include the Host Message Block (HMB), Tape Drive Control Block (TDCB), Host Disk Block (HDB) structures.

006016
Unable to allocate an HDB

Facility: TAPE, TFSEQUEN

Explanation: An attempt to add a host access (requiring allocation of a Host Disk Block (HDB)) failed for lack of resources.

User Action: Submit an SPR with the crash dump.

006017
Tape formatter does not support allowed densities

Facility: TAPE, TFSEQUEN

Explanation: The tape formatter does not support a density the HSC supports. General register 4 points to the Tape Drive Control Block (TDCB) for the drive in question.

User Action: Submit an SPR with the crash dump. Include the host software version and the tape formatter revision. Also include the Tape Drive Control Block (TDCB) structure, host software version, and tape formatter revision.

006020
An invalid density is set in the Tape Drive Control Block (TDCB)

Facility: TAPE, TFSEQUEN

Explanation: An invalid density was set in the Tape Drive Control Block (TDCB). This should not happen. General register 4 points to the Tape Drive Control Block (TDCB) in question.

User Action: Submit an SPR with the crash dump. Submit an SPR with crash dump and Tape Drive Control Block (TDCB) structure.

006021
Read reverse emulation not flagged

Facility: TAPE, TFSEQUEN

Explanation: The tape server entered the read reverse emulation code without read reverse emulation being flagged in the Tape Drive Control Block (TDCB) at offset TD.FLAGS bit TDF.RREVEM. General register 3 points to the windowed control memory address of the Host Message Block (HMB). General register 4 points to the Tape Drive Control Block (TDCB) for drive in question.

General register 2 points to the Tape Formatter Control Block (TFCB) for formatter in question.

User Action: Submit an SPR with the crash dump. Include the following structures: Host Message Block (HMB), Tape Drive Control Block (TDCB), Tape Formatter Control Block (TFCB).

ØØ6Ø22
Route pointer for read reverse emulation zero

Facility: TAPE, TFSEQUEN

Explanation: The tape server entered the read reverse emulation code without having the route pointer set in the Host Message Block (HMB). General register 3 points to the windowed control memory address of the Host Message Block (HMB) in question.

User Action: Submit an SPR with crash dump and the Host Message Block (HMB) structure.

ØØ6Ø23
Requested transfer larger than 64Kb

Facility: TAPE, TFSEQUEN

Explanation: The requested transfer size for a read reverse is larger than 64 Kb. This should not happen. General register 3 points to the windowed control memory address of the Host Message Block (HMB) in question and offset HP..BC indicates the transfer size requested.

User Action: Submit an SPR with the crash dump. Include the Host Message Block (HMB) structure.

006024
Read reverse emulation not flagged

Facility: TAPE, TFSEQUEN

Explanation: The tape server entered the read reverse emulation short retry code without read reverse emulation being flagged in the Tape Drive Control Block (TDCB) at offset TD.FLAGS bit TDF.RREVEM. General register 3 points to the windowed control memory address of the Host Message Block (HMB). General register 4 points to the Tape Drive Control Block (TDCB) for drive in question. General register 2 points to the Tape Formatter Control Block (TFCB) for formatter in question.

User Action: Submit an SPR with the crash dump. Include the following structures: Host Message Block (HMB), Tape Drive Control Block (TDCB), Tape Formatter Control Block (TFCB).

006025
Read reverse emulation not flagged

Facility: TAPE, TFSEQUEN

Explanation: The tape server entered the read reverse emulation long retry code without read reverse emulation being flagged in the Tape Drive Control Block (TDCB) at offset TD.FLAGS bit TDF.RREVEM. General register 3 points to the windowed control memory address of the Host Message Block (HMB). General register 4 points to the Tape Drive Control Block (TDCB) for drive in question. General register 2 points to the Tape Formatter Control Block (TFCB) for formatter in question.

User Action: Submit an SPR with the crash dump. Include the following structures: Host Message Block (HMB), Tape Drive Control Block (TDCB), Tape Formatter Control Block (TFCB).

006026
KT$SEM is equal to zero

Facility: TAPE, TFSEQUEN

Explanation: The K control area offset KT$SEM is zero. This should not happen. General register 3 points to the K control area in question.

User Action: Submit an SPR with the crash dump. Include the K control area structure.

006027
The thread stack is not initialized

Facility: TAPE, TFSERVER

Explanation: The thread stack is not initialized to 52525(8) for a process suspend. This should not happen.

User Action: Submit an SPR with the crash dump.

006030
The thread stack is not initialized

Facility: TAPE, TFSERVER

Explanation: The thread stack is not initialized to 52525(8) for a process resume. This should not happen.

User Action: Submit an SPR with the crash dump.

006031
No available stacks

Facility: TAPE, TFSERVER

Explanation: There are no available stacks for a process trying to suspend.

User Action: Submit an SPR with the crash dump.

ØØ6Ø32
The thread stack is not initialized

Facility: TAPE, TFSERVER

Explanation: The thread stack is not initialized to 52525(8) for a process suspend. This should not happen.

User Action: Submit an SPR with the crash dump.

ØØ6Ø33
Top of user stack for a resume is not set to server return

Facility: TAPE, TFSERVER

Explanation: The top of the user stack on a process resume is not set to the routine server return. This is a software inconsistency.

User Action: Submit an SPR with the crash dump.

ØØ6Ø34
Stack not valid for a process resume

Facility: TAPE, TFSERVER

Explanation: The stack being returned on a process resume is not valid. This is a software inconsistency, caused by the stack not being set 52525(8).

User Action: Submit an SPR with the crash dump.

ØØ6Ø35
Wrong port state for Dialogue Control Block (DCB)

Facility: TAPE, TFSTI

Explanation: The Dialogue Control Block (DCB) is in the wrong port state for attempted operation. The port should be in DCB error state. General register 4 points to the Dialogue Control Block (DCB) in question. General register 2 points to the Tape Formatter Control Block (TFCB).

User Action: Submit an SPR with the crash dump. Include the structures Tape Formatter Control Block (TFCB) and Dialogue Control Block (DCB).

ØØ6Ø36
Wrong port state

Facility: TAPE, TFSTI

Explanation: An error recovery Dialogue Control Block (DCB) operation is attempted when the TRB is not in error state. Error state is determined by bit TFF.DE being set in offset TF.FLAGS of the Tape Formatter Control Block (TFCB). This is a software inconsistency. General register 4 points to the Dialogue Control Block (DCB) in question. General register 2 points to the Tape Formatter Control Block (TFCB) for the drive in question.

User Action: Submit an SPR with the crash dump. Include the following structures: Dialogue Control Block (DCB), Tape Formatter Control Block (TFCB).

ØØ6Ø37
Tape data channel not idle

Facility: TAPE, TFSTI

Explanation: The tape data channel should be idle when queuing this Dialogue Control Block (DCB) to the idle Dialogue Control Block (DCB) list. General register Ø points to the K control area in question. General register 4 points to the Dialogue Control Block (DCB).

User Action: Further testing of the K.sti may be necessary. Investigate tape data channel in requestor slot indicated by the field KG$SLT of the K control area. If no hardware problem exists, submit an SPR. Include the following structures: K control area, and Dialogue Control Block (DCB).

ØØ6Ø4Ø
No stack available to suspend with

Facility: TAPE, TFSTI

Explanation: No stack available for suspending a process. General register 2 points to the Tape Formatter Control Block (TFCB). General register 5 points to the K control area. General register 4 points to the Dialogue Control Block (DCB).

User Action: Submit an SPR with the crash dump and the following structures: Tape Formatter Control Block (TFCB), Dialogue Control Block (DCB), and K control area.

ØØ6Ø41
Dialogue Control Block (DCB) operation timed out

Facility: TAPE, TFSTI

Explanation: A Dialogue Control Block (DCB) operation timed out. This usually indicates a problem in the tape data channel. The tape requestor slot in question is given as the second word on the stack.

User Action: If no hardware problem exists, submit an SPR.

006042
Invalid context

Facility: TAPE, TFSTI

Explanation: A Dialogue Control Block (DCB) operation is being attempted from a context other than the TAPE server. This is a software inconsistency.

User Action: Submit an SPR with the crash dump.

006043
Buffer descriptor address missing

Facility: TAPE, TXREVERSE

Explanation: The next address is missing from the linked list of buffer descriptors. General register 5 points to the Fragment Request Block (FRB) in question. Offset F$BFHD points to the buffer descriptor list in question.

User Action: Submit an SPR with the crash dump. Include the Fragment Request Block (FRB) structure.

006044
Unexpected Fragment Request Block (FRB) error received

Facility: TAPE, TFERR

Explanation: An error was received from a software station rather than from a hardware station. General register 5 points to the fragment request block (FRB) in error.

User Action: Submit an SPR with the crash dump. Include the FRB.

006045
Unknown Fragment Request Block (FRB) error received

Facility: TAPE, TFERR

Explanation: An unidentifiable error is flagged in a fragment request block (FRB).

User Action: Submit an SPR with the crash dump.

006046
K.ci did not return a Fragment Request Block

Facility: TAPE, TFERR

Explanation: Transfer Request Blocks (TRB) have associated Fragment Request Blocks (FRB) that point to data buffers. When a TRB is received in error, the FRBs must be deallocated. If an FRB is held by K.ci and not returned within 20 seconds, this crash occurs.

User Action: If no hardware problem exists, submit an SPR with the crash dump. If the problem reoccurs, investigate the K.ci.

006047
Illegal downcount occurred on a Host Message Block (HMB) chain

Facility: TAPE, TFERR

Explanation: Whenever Transfer Request Blocks (TRB) are purged from the K.sti input queue, the associated Host Message Block (HMB) must not be returned to the host as an end message. This catching mechanism relies on a change of HMBs with associated counters. This is a software consistency check to ensure control memory is not corrupted by the end of the chain. General register 5 points to the HMB.

User Action: Submit an SPR with the crash dump. Include the HMB.

006050
Sequence number corruption occurred

Facility: TAPE, TFERR

Explanation: Error recovery ensures against a deadlock on K.sti by preventing a Transfer Request Block (TRB) from waiting for a diagnostic control block (DCB) that will never execute. Such a deadlock can only occur from a software inconsistency.

User Action: Submit an SPR with the crash dump.

007000
This class of crashes includes CIMGR software consistency errors

Facility: CIMGR, any

Explanation: A software inconsistency error occurred.

User Action: Submit an SPR with the crash dump. Specify the utilities or diagnostics active at the time of the crash.

007001
Received a sequence message without a credit

Facility: CIMGR, CIDIRECT

Explanation: The SCS$DIRECT process received a sequence message in a Host Message Block (HMB) flagged by the K.ci as not having a credit for the connection. General register 1 has the address of the HMB in error.

User Action: Submit an SPR with the crash dump. Include the HMB.

007002
Failed to acquire a control block from K.ci

Facility: CIMGR, CIMISCPRC

Explanation: The POLLER process was not able to obtain a control block from K.ci to resend a timed-out STACK datagram.

User Action: Further testing of the HSC subsystem may be necessary. Investigate the available control memory. If no hardware problem exists, submit an SPR with the crash dump.

007003
K.ci is hung

Facility: CIMGR, CIMISCPRC

Explanation: During the polling interval the POLLER ensures that K.ci is still running. This trap indicates it is not.

User Action: Further testing of the HSC subsystem may be necessary. Investigate the K.ci boards. If no hardware problem exists, submit an SPR with the crash dump.

007004
K.ci detected an unrecoverable error and stopped

Facility: CIMGR, CIMISCPRC

Explanation: K.ci sent its control area to the CIMGR exception process. This is done whenever K.ci has detected a non-recoverable hardware error.

User Action: Further testing of the HSC subsystem may be necessary. Investigate the K.ci boards and data memory. If no hardware problem exists, submit an SPR with the crash dump.

007005
K.ci patch status check failed

Facility: CIMGR, CIMISCPRC

Explanation: K.ci did not respond to a path status check within eight seconds.

User Action: Investigate the K.ci boards. Further testing of the HSC subsystem may be necessary. If no hardware problem exists, submit an SPR with the crash dump.

007006
System name is corrupted

Facility: CIMGR, CIROOT

Explanation: During initialization, the CIMGR discovered the System name was corrupted in the SCT.

User Action: Release the Online button on the HSC (out). Reboot the HSC by holding the Fault button down until the State light blinks. This will bypass using the SCT on the boot device. Run SETSHO to reset system name and ID, then reboot HSC one more time before pushing in the Online button on the front panel.

007007
HMB received with wrong number of BMBs

Facility: CIMGR, CISCS

Explanation: A Host Message Block (HMB) was received with the wrong number of Big Message Blocks (BMBs). A START or ID packet was received from K.ci without the proper number of associated data memory blocks. General register 0 points the HMB.

User Action: If no hardware problem exists, submit an SPR with the crash dump. Investigate the K.ci boards.

007010
Inconsistent connection state

Facility: CIMGR, CISCS

Explanation: An illegal state transition was attempted on a connection. This is a software problem. General register 2 points to the Connection Block (CB).

User Action: Submit an SPR with the crash dump. Include the CB.

007011
Connection incarnation inconsistent

Facility: CIMGR, CISCS

Explanation: While a connection is in the process of opening, the incarnation of that connection is flagged as formative. The final step of opening the connection is to remove the flag. This crash indicates the flag was prematurely removed indicating a state inconsistency for the connection. General register 2 points to the Connection Block (CB).

User Action: Submit an SPR with the crash dump. Include the CB.

007012
Connection incarnation mismatch

Facility: CIMGR, CISCS

Explanation: The incarnation of an opening connection is kept in both the Connection Block (CB) and the Connection Block vector table. As a connection opens a check is made to ensure these incarnations agree. A disagreement indicates dangling reference to an old carnation of the connection. Register 2 points to the Connection Block (CB).

User Action: Submit an SPR with the crash dump. Include the CB.

007013
Inconsistent connection state due to a VC closure

Facility: CIMGR, CISCS

Explanation: An illegal state transition was attempted on a connection. The state transition was initiated by a VC closure. General register 2 points to the Connection Block (CB).

User Action: Submit an SPR with the crash dump. Include the CB.

007014
Unable to retrieve resource from K.ci during a disconnect

Facility: CIMGR, CISCS

Explanation: During a disconnect, the CIMGR was unable to retrieve the resources associated with the credits on that connection from K.ci.

User Action: Submit an SPR with the crash dump.

007015
K.ci did not respond to notification of a VC closure

Facility: CIMGR, CISUBRS

Explanation: The CIMGR informs K.ci when it marks a VC as closed. It then allows the K.ci eight seconds to respond to the notification. This crash occurs if the response times out.

User Action: If no hardware problem exists, submit an SPR with the crash dump. Investigate the K.ci.

007016
Illegal attempt to deallocate a Connection Block (CB)

Facility: CIMGR, CISUBRS

Explanation: An attempt was made to deallocate a Connection Block (CB) without breaking the connection. General register 2 points to the CB.

User Action: Submit an SPR with the crash dump. Include the CB.

007017
Attempt to deallocate a Connection Block without an incarnation

Facility: CIMGR, CISUBRS

Explanation: A Connection Block (CB) did not have a valid incarnation at the time it was deallocated. This crash indicates a software inconsistency.

User Action: Submit an SPR with the crash dump. Include the CB.

007020
Failure to retrieve SCS resources from K.ci

Facility: CIMGR, CISUBRS

Explanation: When trying to allocate resources for use across a virtual circuit (VC), the count of data memory resources was incorrect. The Host Message Block (HMB) for serializing VC traffic must have two Big Message Blocks (BMB). General register 0 points to the HMB.

User Action: Submit an SPR with the crash dump. Include the HMB.

007021
The count of waiters for virtual circuit resources went negative

Facility: CIMGR, CISUBRS

Explanation: While processing the list of waiters for transmission resources for a virtual circuit (VC), a nonempty list was detected to indicate a negative number of waiters. This is strictly a software inconsistency. General register 1 points to the system block (SB).

User Action: Submit an SPR with the crash dump. Include the SB.

012001
Can't Find Connection Block

Facility: DUP

Explanation: When DUP receives an HMB, DUP tries to find a reference to the Connection Block (referred to by HM$CTX in the HMB) in the DG$ structures (DUP Context Control Blocks). DUP was unable to find a reference to the Connection Block, even though it searched every DG$ structure.

User Action: Submit an SPR with the exception dump or startup message indicating the contents of the stack.

012002
Illegal BMB Count

Facility: DUP

Explanation: The HMB (MSCP packet carrier) has an illegal number of Big Message Buffers (BMBs) allocated. DUP allows only one. The HMB is invalid.

User Action: Submit an SPR with the exception dump or startup message indicating the contents of the stack. The second word of the stack contains the windowed address of the HMB. The third

word of the stack contains the value in HM$CN -- the count of the number of BMBs.

012003
Illegal HMB Opcode

Facility: DUP

Explanation: The opcode specified in the HM$LOF field of the HMB was not equal to HML$RM. (Received sequence message over connection; HML$RM=000000.) HMB opcodes must indicate the HMB is for a sequenced message.

User Action: Submit an SPR with the exception dump or startup message indicating the contents of the stack. The second word of the stack contains the illegal opcode.

012004
Illegal HMB Error

Facility: DUP

Explanation: The error specified in the HM$ERR field of the HMB was not equal to 0, HME$EC, or HME$NC. (Extra credits received; HME$EC=10.) (No credits received; HME$NC=4.)

User Action: Submit an SPR with the exception dump or startup message indicating the contents of the stack. The second word of the stack contains the value in the HM$ERR field.

012021
Invalid Connection Block

Facility: DUP

Explanation: The DUP process received a Connection Block with an invalid value in the CB$ACT field. The CB$ACT field contains the action value (action to be performed by the DUP server).

User Action: Submit an SPR with the exception dump or startup message indicating the contents of the stack. The second word on the stack contains the contents of the CB$ACT field.

Ø12Ø24
Bad Down Count

Facility: DUP

Explanation: DUP initiates return of the endpacket to the host by down counting the reference counter in the related control block. The down-count action should return 1. If the downcount does not decrement the reference counter to 1, DUP crashes the HSC.

User Action: Submit an SPR with the exception dump or start-up message indicating the contents of the stack. The second word on the stack is the value of the counter following the downcount.

Ø12Ø36
Connection Broken

Facility: DUP

Explanation: While DUP was preparing to send a message to the K.ci, the connection to the host was broken. The connection was broken after DUP did an extensive check to ensure the connection existed. DUP detected the connection break the second time because the DG$CB field was set to Ø.

User Action: Submit an SPR. This is is an internal consistency check and should never be seen.

Ø42ØØ1
FAO message buffer overflow

Facility: DIRECT

Explanation: The progam DIRECT was attempting to output the end message, but the length of that message was longer than the allotted FAO output buffer.

User Action: Submit an SPR with the crash dump.

043001
Wrong HMB received when trying to bring source online

Facility: DKCOPY

Explanation: This is crash $CDKCOPY+SRC_ONL_HMB. An HMB (Host Message Block) was sent to the disk server requesting the source unit be brought online in a shadow set. When the completion queue of this HMB was checked, it pointed to a different (incorrect) HMB.

User Action: Submit an SPR with the dump. Top of stack equals crash code. Second word points to previous HMB.

043002
Bad downcount when trying to bring source online

Facility: DKCOPY

Explanation: This is crash $CDKCOPY+SRC_ONL_CNT. When an MSCP end message was to be sent over a connection to a host, a counter keeping track of the transaction (decrementing by one) failed to operate properly. This occurred after the disk server was asked to bring the source unit online in a shadow set.

User Action: Submit an SPR with the dump. Top of stack equals crash code. Second word points to counter.

043003
Wrong HMB received when trying to issue GCS to target unit

Facility: DKCOPY

Explanation: This is crash $CDKCOPY+TGT_GCS_HMB. An HMB (Host Message Block) was sent to the disk server requesting a GCS (GET COMMAND STATUS) command be sent to the target unit. When the completion queue of this HMB was checked, pointed to a different (incorrect) HMB.

User Action: Submit an SPR with the dump. Top of stack equals crash code. Second word points to previous HMB.

043004
Bad downcount when trying to issue GCS to target unit

Facility: DKCOPY

Explanation: This is crash $CDKCOPY+TGT_GCS_CNT. When an MSCP end message was to be sent over a connection to a host, a counter keeping track of the transaction (decrementing by one) failed to operate properly. This occurred after the disk server was asked to send a GCS (GET COMMAND STATUS) command to the target unit.

User Action: Submit an SPR with the dump. Top of stack equals crash code. Second word points to counter.

043005
Bad downcount when trying to bring target unit online

Facility: DKCOPY

Explanation: This is crash $CDKCOPY+TGT_ONL_CNT. When an MSCP end message was to be sent over a connection to a host, a counter keeping track of the transaction (decrementing by one) failed to operate properly. This occurred after the disk server was asked to bring the target unit online into the shadow set.

User Action: Submit an SPR with the dump. Top of stack equals crash code. Second word points to counter.

043006
Bad downcount when trying to issue abort command to target unit

Facility: DKCOPY

Explanation: This is crash $CDKCOPY+TGT_ABO_CNT. When an MSCP end message was to be sent over a connection to a host, a counter keeping track of the transaction (decrementing by one) failed

to operate properly. This occurred after the disk server had been asked to abort an online command to the target unit.

User Action: Submit an SPR with the dump. Top of stack equals crash code. Second word points to counter.

043007
Wrong HMB received after issuing AVL command to shadow unit

Facility: DKCOPY

Explanation: This is crash $CDKCOPY+SHA_AVL_HMB. An HMB (Host Message Block) was sent to the disk server requesting the shadow unit used to facilitate the copy operation be made available. When the completion queue of this HMB was checked, it pointed to a different (incorrect) HMB.

User Action: Submit an SPR with the dump. Top of stack equals crash code. Second word points to previous HMB.

043010
Bad downcount when trying to issue AVL command to shadow unit

Facility: DKCOPY

Explanation: This is crash $CDKCOPY+SHA_AVL_CNT. When an MSCP end message was to be sent over a connection to a host, a counter keeping track of the transaction (decrementing by one) failed to operate properly. This occurred after the disk server was asked to send the shadow unit available.

User Action: Submit an SPR with the dump. Top of stack equals crash code. Second word points to counter.

051001
An XFRB was not acquired to print messages

Facility: SETSHO,SSMAIN

Explanation: This is crash $CSETSHO+NOXFRB. An XFRB (Extended Function Request Block) was not acquired by the SETSHO main routine. A crash was initiated because the lack of this item prevented communication between the HSC and the console.

User Action: Submit an SPR with the dump.

051002
Failed to properly send HMB to K.ci

Facility: SETSHO,SSMAIN

Explanation: This is crash $CSETSHO+CIHMB. An HMB (Host Memory Block) was sent to the K.ci (the hardware that handles communication between the hosts and the HSC). A crash was initiated because confirmation of the HMB was not recieved from the K.ci within the required time.

User Action: Submit an SPR with the dump.

51003
Too many characters intended for console printout

Facility: SETSHO,SSMAIN

Explanation: This is crash $SETSHO+PNTOVF. In this case, Formatted ASCII Ouput (FAO) was called and it generated more characters than the buffer size allocated would allow. The maximum is 510 characters.

User Action: Submit an SPR with the dump. R1 points to string size.

051004
The SCT (System Control Table) crossed a page boundary

Facility: SETSHO,SSMAIN

Explanation: This is crash $SETSHO+SCTXPG. The SCT must remain on one page in memory. It typically indicates an incorrect amount of padding was placed at the end of the file SSDATA.MAC.

User Action: Submit an SPR with the dump.

051101
Failed in sending HMB to disk server for SET Dn [NO]HOST

Facility: SETSHO,SET

Explanation: This is crash $CSETSHO+SETDSK. An HMB (Host Memory Block) was sent to the disk server in order to SET a disk drive HOST or NOHOST. The crash was initiated because the confirmation of this command was not received within the required time.

User Action: Submit an SPR with the dump.

051102
Failed in sending HMB to tape server for SET Tn [NO]HOST

Facility: SETSHO,SET

Explanation: This is crash $CSETSHO+SETTAP. An HMB (Host Memory Block) was sent to the tape server in order to SET a tape drive HOST or NOHOST. The crash was initiated because the confirmation of this command was not received within the required time.

User Action: Submit an SPR with the dump.

051201
Failed in sending HMB to disk server for SHOW Dn

Facility: SETSHO,SHOW

Explanation: This is crash $CSETSHO+SHODSK. An HMB (Host Memory Block) was sent to the disk server in order to SHOW a specific disk drive. The crash was initiated because the confirmation of this command was not received within the required time.

User Action: Submit an SPR with the dump.

051202
Failed in sending HMB to tape server for SHOW Tn

Facility: SETSHO,SET

Explanation: This is crash $CSETSHO+SHOTAP. An HMB (Host Memory Block) was sent to the tape server in order to SHOW a specific tape drive. The crash was initiated because the confirmation of this command was not received within the required time.

User Action: Submit an SPR with the dump.

051203
SCT crash context table contained too many characters

Facility: SETSHO,SHOW

Explanation: This is crash $SETSHO+CSHOVF. The SCT crash context table contained too many characters. In this case FAO was called and it generated more characters than the buffer size would allow. The maximum is 510 characters.

User Action: Submit an SPR with the dump. R1 points to string size.

052001 ($CDWMATH)
Double word math not consistent

Facility: SINI

Explanation: During calculation and allocation of control blocks (allocate in quantities of double-word), the count of words in control blocks was not a double-word multiple.

User Action: Submit an SPR with a dump. R0 points to Memory Descriptor (MD).

052002 ($CDIV10)
Divide operation set overflow

Facility: SINI

Explanation: During allocation of control blocks (set as 80 percent of available control memory), a divide operation set the PSW Overflow bit.

User Action: Submit an SPR with a dump.

052003 ($CMUL8)
Multiply operation set overflow

Facility: SINI

Explanation: During allocation of control blocks (set as 80 percent of available control memory), a divide operation set the PSW Overflow bit.

User Action: Submit an SPR with a dump.

061001
XCALL stack corrupted

Facility: DIAGINT

Explanation: The DDUSUB transfer routines use a stack allocated from common pool for XCALLs (cross-address space calls) from the disk server. The low word of this stack is initialized to a special value which should never change. This crash occurs when the routine DDUTIO is called. The low word of the stack contains a different value than the initialization value. The most probable cause is corruption by the process running.

User Action: Submit an SPR with the crash dump. Note the diagnostics or utilities running at the time of the crash.

062001 ($CNOWINDOW)
Process does not have windows declared

Facility: SUBLIB, ERTYP

Explanation: A process which requested an out of band error log be issued via the ERTYP$ service in SUBLIB does not have windows declared in its PCB (Process Control Block) declaration. A Window set is required to use this service.

User Action: Submit an SPR with a dump.

APPENDIX C
GENERIC ERROR LOG FIELDS

C.1 GENERIC ERROR LOG FIELDS

Some fields described on HSC console message printouts are generic, regardless of error type. These fields are described in the following table. Error Flags and MSCP/TMSCP Event Codes are covered in more depth in separate tables in this appendix.

Table C-1 Generic Error Log Fields

Field	Description
ERROR-X	The X represents the severity level of the error message. Severity levels are E for error, S for success, W for warning, I for informational, and F for fatal. What follows is the English version of the error message describing the event code, the date and time.
Command Ref #	This number, in hexadecimal, is the MSCP command number that caused the error reported, or zero if the error does not correspond to a specific outstanding command.
Err Seq #	This number, in decimal, is the sequence number of this error log message since the last time the MSCP server lost context, or zero if the MSCP server does not implement error log sequence numbers.
Error Flags	This number, in hexadecimal, indicates bit flags, collectively called error log message flags, used to report various attributes of the error. See Table C-2 for a description of the error flags.

Field	Description
Event	This number, in hexadecimal, identifies the specific error or event being reported by this error log message. This code consists of a five-bit major event code and an 11-bit subcode. The event codes and what they mean are listed in Table C-3.

Table C-2 Error Flags

Bit Number	Bit Mask Hex.	Format Description
7	80	If set, the operation causing this error log message has successfully completed. The error log message summarizes the retry sequence necessary to successfully complete the operation.
6	40	If set, the retry sequence for this operation continues. This error log message reports the unsuccessful completion of one or more retries.
5	20	(MSCP-specific) If set, the identified logical block number (LBN) needs replacement.
4	10	(MSCP-specific) If set, the reported error occurred during a disk access initiated by the controller bad block replacement process.
0	1	If set, the error log sequence number has been reset by the MSCP server since the last error log message sent to the receiving class driver.

C.2 MSCP/TMSCP EVENT CODES

The following table is a sequential list of all known MSCP and TMSCP event codes. Each event code cross references to an error description. The first column is the event code number in hexadecimal. The second column references the class of error. The third column is the expanded description that matches the event code.

Table C-3 MSCP/TMSCP Event Codes

Event Code Hex	Class	Description
0000	Success	Normal
0001	Invalid Command	Invalid message length Other invalid command subcode values should be referenced as follows. Note, this is combined with the status code: offset*256.+code Offset is the command message offset symbol for the field in error and code is the symbol for the Invalid Command status code.
0002	Command Aborted	Command Aborted
0003	Unit Offline	Unit unknown or online to another controller.
0004	Unit Available	Unit Available
0007	Compare Error	Data compare error Data compare error resulted from COMPARE CONTROLLER DATA or COMPARE HOST DATA command.
0008	Data	Disk - Sector was written with Force Error modifier. Tape - Long gap encountered.
0009	Host Buffer	Host buffer access error--cause not available The controller was unable to access a host buffer to perform a transfer, but has no visibility into the cause of the error.
000A	Controller	Reserved for host--command timeout expired.

Event Code Hex	Class	Description
000C	Shadow Set Status Has Changed	Shadow set status has changed
000D	BOT Encountered	BOT Encountered
000E	Tape Mark Encountered	Tape mark encountered
0010	Record Data Truncated	Record data truncated, data transfer operation
0013	LEOT Detected	LEOT detected
0014	Bad Block Replacement	Bad block successfully replaced
0016	Access Denied	Access denied
0020	Success	Spindown ignored
0023	Unit Offline	Disk - No volume mounted or drive disabled via RUN/STOP switch. Unit is in known substate. Tape - No media mounted or disabled via switch setting
0026	Unit Available	No members in shadow set
0029	Host Buffer	Odd transfer address
002A	Controller	SERDES overrun or underrun error Either the drive is too fast for the controller, or a controller hardware fault has prevented controller microcode from being able to keep up with data transfer to or from the drive.
002B	Disk Drive	Drive command timeout For SI drives, the controller timeout expired for either a Level 2 exchange or the assertion of Read/Write Ready after an Initiate Seek.

Event Code Hex	Class	Description
0034	Bad Block Replacement	Block verified OK - not a bad block
0035	Invalid Parameter	Invalid key length The key length is to short for the specified key type.
0040	Success	Still connected
0043	Unit Offline	Unit is inoperative For SI drives, the controller has marked the drive inoperative due to an unrecoverable error in a previous Level 2 exchange, the drive Cl flag is set, or the drive has a duplicate unit identifier.
0044	Unit Available	Shadow set copy in progress
0048	Disk Data	Invalid header The subsystem read an invalid or inconsistent header for the requested sector. For recoverable errors, this code implies a retry of the transfer read a valid header. For unrecoverable errors, this code implies the subsystem attempted non primary revectoring and determined the requested sector was not revectored. (As an example, the RCT indicates the sector is not revectored). Causes of an invalid header include header missync, header sync timeout, and an unreadable header.
0049	Host Buffer	Odd byte count
004A	Controller	EDC Error The sector was read with correct or correctable ECC and an invalid EDC. A fault probably exists in the ECC logic of either this controller or the controller that last wrote the sector.

Event Code Hex	Class	Description
004B	Disk Drive	Controller-detected transmission error For SI drives, the controller detected an invalid framing code or a checksum error in a Level 2 response from the drive.
0054	Bad Block Replacement	Replacement failure-- REPLACE command or its analogue failed
0055	Invalid Parameter	Invalid key type The controller does not implement the specified key type.
0068	Disk Data	Data Sync not found (Data Sync timeout)
0069	Host Buffer	Non-existent memory error
006A	Controller	Inconsistent internal control structure A high-level check detected an inconsistent data structure. For example, a reserved field contained a nonzero value, or the value in a field was outside its valid range. This error almost always implies the existence of a microcode problem.
006B	Disk Drive	Positioner error (misseek) The drive reported a seek operation was successful, but the controller determined the drive had positioned itself to an incorrect cylinder.
0074	Bad Block Replacement	Replacement failure-- inconsistent RCT
0075	Invalid Parameter	Invalid key value A checksum or similar indicates the key value is internally inconsistent.
0080	Success	Duplicate unit number
0083	Unit Offline	Duplicate unit number

Event Code Hex	Class	Description
0085	Media Format	Characteristics or protection mismatch for shadow member
0088	Disk Data	Correctable error in ECC field A transfer encountered a correctable error where only the ECC field was affected. All data bits were correct, but a portion of the ECC field was incorrect. The severity of the error (the number of symbols in error) is unknown. If the number of symbols in error is known, an n Symbol ECC Error subcode should be returned instead.
0089	Host Buffer	Host memory parity error
008A	Controller	Internal EDC error A low-level check detected an inconsistent data structure. For example, a microcode-implemented checksum or vertical parity (hardware parity is horizontal) associated with internal sector data was inconsistent. This error usually implies a fault in the memory addressing logic of one or more controller processing elements. It can also result from a double bit error or other error exceeding the error detection capability of the controller hardware memory checking circuitry.
008B	Disk Drive	Lost Read/Write Ready during or between transfers For SI drives, Read/Write Ready drops when the controller attempts to initiate a transfer or at the completion of a transfer with Read/Write Ready previously asserted. This usually results from a drive-detected transfer error, where additional error log messages containing the drive-detected error subcode may be generated.

Event Code Hex	Class	Description
0094	Bad Block Replacement	Replacement failure-- drive access failure One or more transfers specified by the replacement algorithm failed.
00A5	Disk Media	Disk not formatted with 512-byte sectors The disk FCT indicates it is formatted with 576-byte sectors, although both the controller and the drive support only 512-byte sectors.
00A9	Host Buffer	Invalid page table entry See Unibus/Q-bus Storage Systems Port Specifications for additional detail.
00AA	Controller	LESI Adapter Card parity error on input (adapter to controller)
00AB	Disk Drive	Drive clock dropout For SI drives, either data or state clock was missing when it should have been present. This is usually detected by means of a timeout.
00B4	Bad Block Replacement	Replacement failure, no replacement block available Replacement was attempted for a bad block, but a replacement block could not be allocated. For example, the volume's RCT is full.
00C5	Disk Media	Disk not formatted or FCT corrupted The disk FCT indicates the disk is not formatted in either 512- or 576-byte mode.

Event Code Hex	Class	Description
00C9	Host Buffer	Invalid buffer name The key in the buffer name does not match the key in the buffer descriptor, the B bit in the buffer descriptor is clear, or the index into the buffer descriptor table is too large.
00CA	Controller	LESI Adapter Card parity error on output (controller to adapter)
00CB	Disk Drive	Lost receiver ready for transfer For SI drives, Receiver Ready was negated when the controller attempted to initiate a transfer or did not assert at the completion of a transfer. This includes all cases of the controller timeout expiring for a transfer operation (Level 1 real time command).
00D4	Bad Block Replacement	Replacement failure, recursion failure Two successive RBNs were bad.
00E8	Data	Disk - Uncorrectable ECC Error A transfer without the Suppress Error Correction modifier encountered an ECC error exceeding the correction capability of the subsystem error correction ted.algorithms or a transfer with the Suppress Error Correction modifier encountered an ECC error of any severity. Tape - Unrecoverable read error
00E9	Host Buffer	Buffer length violation The number of bytes requested in the MSCP command exceeds the buffer length as specified in the buffer descriptor.
00EA	Controller	LESI Adapter Card "cable in place" not asserted.

Event Code Hex	Class	Description
00EB	Disk Drive	Drive-detected error For SI drives, the controller received a Get Status or unsuccessful response with EL set or the controller received a response with the DR flag set, and it does not support automatic diagnosis for that drive type.
0100	Success	Already online
0103	Unit Offline	Unit disabled by Field Service or diagnostic For SI drive, the drive DD flag is set.
0105	Disk Media	RCT corrupted The RCT search algorithm encountered an invalid RCT entry. The subcode may be returned under the following conditions: during replacement of a block, revectoring a faulty block, and when a unit is brought online.
0106	Write Protected	Unit is data safety write protected
0108	Disk Data	One-Symbol ECC Error
0109	Host Buffer	Access control violation The access mode specified in the buffer descriptor is protected against the PROT field in the PTE.
010A	Controller	Controller overrun or underrun The controller attempted to perform too many concurrent transfers, causing one or more of them to fail due to a data overrun or underrun.

Event Code Hex	Class	Description
010B	Disk Drive	Controller-detected pulse or state parity error For SI drives, the controller detected a pulse error on either the state or data line, or the controller detected a parity error in a state frame.
0125	Disk Media	No replacement block available Replacement of a faulty block was attempted, but a replacement block could not be allocated (i.e. the RCT is full). This subcode may be returned during actual replacement and when an interrupted replacement is completed as part of bringing a unit online.
0128	Disk Data	Two-Symbol ECC Error
012A	Controller	Controller memory error The controller detected an error in an internal memory, such as a parity error or nonresponding address. This subcode applies only to errors not affecting the ability of the HSC70 to properly generate End and Error Log messages. Errors affecting End and Error Log messages are not reported via MSCP. For most controllers, this subcode is returned only for controller memory errors in data or buffer memory and noncritical control structures. If the controller has several such memories, the specific memory involved is reported as part of the error address in the error log message.
012B	Disk Drive	Drive-requested error log (EL bit set)
0148	Disk Data	Three-Symbol ECC Error
014A	Controller	PLI reception buffer parity error

Event Code Hex	Class	Description
014B	Disk Drive	Controller-detected protocol error For SI drives, a Level 2 response from the drive had correct framing codes and checksum but was not a valid response within the constraints of the SI protocol. The response had an invalid opcode, was an improper length,error. or was not a possible response in the context of the exchange.
0168	Disk Data	Four-Symbol ECC Error
016A	Controller	PLI transmission buffer parity error
016B	Disk Drive	Drive failed initialization For SI drives, the drive clock did not resume following a controller attempt to initialize the drive. This implies the drive encountered a fatal initialization error.
0188	Disk Data	Five-Symbol ECC Error
018B	Disk Drive	Drive ignored initialization For SI drives, the drive clock did not cease following a controller attempt to initialize the drive. This implies the drive did not recognize the initialization attempt.
01A8	Disk Data	Six-Symbol ECC Error
01AB	Disk Drive	Receiver Ready collision For SI drives, the controller attempted to assert its Receiver Ready when the Receiver Ready of the drive was still asserted.
01C8	Disk Data	Seven-Symbol ECC Error

Event Code Hex	Class	Description
01CB	Disk Drive	Response overflow A drive sent back more frames than the reception buffer could hold. This can be caused by a hung drive microdiagnostic or a malfunctioning K.sdi.
01E8	Disk Data	Eight-Symbol ECC Error A transfer encountered a correctable ECC error with the specified number of ECC symbols in error. The number of symbols in error roughly corresponds to the severity of the error.
0200	Success	Still online
0203	Unit Offline	Exclusive use
0208	Disk Data	Nine-Symbol ECC Error.
0220	Success	Still Online/Unload ignored
0228	Disk Data	Ten-Symbol ECC Error.
0248	Disk Data	Eleven-Symbol ECC Error.
0268	Disk Data	Twelve-Symbol ECC Error.
0288	Disk Data	Thirteen-Symbol ECC Error.
02A8	Disk Data	Fourteen-Symbol ECC Error.
02C8	Disk Data	Fifteen-Symbol ECC Error.
0400	Success	Tape - EOT encountered
0404	Unit Available	Already in use
044B	Tape Drive	Drive error Controller retry limit exhausted.
0800	Success	Invalid RCT
1000	Success	Read only volume format

Event Code Hex	Class	Description
1006	Write Protected	Unit is software write protected
2006	Write Protected	Unit is hardware write protected
F3AA	Controller	Unknown K.tape error
FCAA	Controller	Word Rate Clock timeout The K.sti detected the loss of clocks from a drive during a transfer.
FCEA	Controller	Receiver Ready not asserted at start of transfer - The HSC70 is ready to start a transfer by sending the formatter a Level 1 command, and the formatter does not have Receiver Ready asserted.
FD2A	Controller	Data Ready timeout - This controller did not detect Data Ready from the formatter within 5 ms after sending it a Level 1 command.
FD6A	Controller	Acknowledge not asserted at start of transfer - The HSC70 is ready to start a transfer by sending the formatter a Level 1 command, and the formatter does not have Acknowledge asserted.
FDEC	Tape Formatter	Could not get extended drive status
FE0C	Tape Formatter	Could not get formatter summary status while trying to restore tape position
FE2A	Controller	Record EDC error - On a read from tape operation the EDC calculated by the K.STI did not match the EDC generated by the tape formatter
FE2B	Tape Drive	Could not set byte count
FE4B	Tape Drive	Could not write tape mark
FE6B	Tape Drive	Could not set unit characteristics
FE8A	Controller	Lower processor timeout - The upper processor in the K.sti detected the lower processor had stopped and restarted it.

Event Code Hex	Class	Description
FE8B	Tape Drive	Unable to position to before LEOT
FEAB	Tape Drive	Rewind failure
FECB	Tape Drive	Could not complete online sequence
FEEB	Tape Drive	Erase gap failed
FF0B	Tape Drive	ERASE command failed
FF0C	Tape Formatter	TOPOLOGY command failed
FF31	Tape Drive Position Lost	Retry limit exceeded while attempting to restore tape position
FF68	Tape Data	Formatter retry sequence exhausted
FF6A	Controller	Lower processor error
		A bit was set in the lower processor error register. Bits included in the lower processor error register are Data Bus NXM, Data SERDES Overrun Data Bus Overrun, Data Bus Par Err, Data Pulse Missing, and Sync Real Time Par Err.
FF6B	Tape Drive	Tape drive requested error log
FF6C	Tape Formatter	Formatter requested error log
FF71	Tape Drive Position Lost	Formatter-detected position lost
FF88	Tape Data	Controller transfer retry limit exceeded
FF8A	Controller	Buffer EDC error
		The K.sti detected an EDC error on the data buffer it read from memory on a Write operation.
FFA8	Tape Data	Host requested retry suppression on a K.sti-detected error

Event Code Hex	Class	Description
FFAA	Controller	Data overflow due to Pipeline error No data buffers in HSC70 data memory were available when the K.sti needed one during a data transfer
FFC8	Tape Data	Reverse retry currently not supported
FFCB	Tape Drive	Could not position for (formatter) retry.
FFCC	Tape Formatter	Cannot clear formatter errors
FFD1	Tape Drive Position Lost	Formatter and HSC70 disagree on tape position
FFE8	Tape Data	Host requested retry suppression on a formatter-detected error
FFEB	Tape Drive	Cannot clear drive errors
FFEC	Tape Formatter	Could not get formatter summary status during transfer error recovery
FFF1	Tape Drive Position Lost	Controller-detected position lost

APPENDIX D
INTERPRETATION OF STATUS BYTES

D.1 INTRODUCTION

This appendix lists all possible codes each K can generate after detecting a fatal error. Only K-detected errors are included.

When a K detects a fatal error, it puts a code in its status register and performs a Level 7 Control Bus Interrupt. This interrupt causes the HSC to trap through location 134 and crash. The crash message contains the status codes from all Ks in the **Status of Requestors (1-9):** field.

Figure D-1 shows a printout example from a K-detected error. In this case, as in many others, the crash was not caused by the K but was detected by the K forcing the crash. For additional explanations of the fields in the crash message, refer to Appendix B.

```
-* SUBSYSTEM EXCEPTION *-     V# Y10B       HSC70 HSC002
       at 18-Jan-1986 01:15:14.50  up   0 00:08:46.20

User   PC: 0027360 caused by (134    ) Kint
PSW: 140000

KBCTRL active, PCB addr = 102636

R0-R5:
024302   047632   000020   047626   0000000   141404

Kernel SP: 000774

Kernel Stack:

005046 000004 053354 046022 001012 050476 050476 000000
047062 047466 047466 000000 047264 000000 055352 000000

User SP: 023346

User Stack:
052525 052525 025252 025252 025252 025252 025252 025252
025252 025252 025252 025252 025252 025252 025252 025252

KPAR(0-7):
000440 000640 001040 001440 002040 001240 000240 177600

KPDR(0-7):
077506 077506 177506 077506 077406 077506 077506 077506

UPAR(0-7):
000440 000640 001040 001440 002040 001240 000240 177600

UPDR(0-7):
007406 007406 177406 007406 007406 007406 007406 100016

MMSR(0-2): 000017 000020 037654

Window index reg: 000015

Window Bus Reg: 140105

WADR(0-7):
160004 161004 162004 163004 164004 165004 166034 167034

Translated WADR(0-7):
001401 001401 001401 001401 001401 001401 001407 001607
```

Figure D-1 Subsystem Exception K-Detected Error (1 of 2)

```
Error regs: 170024  000077

Status of requestors (1-9):
000177 000002 000002 000377 000377 000377 000377 000377 000203

(PC-6) TO (PC):
104002 012600 000003 011505
Control area for slot #000001
Control area address: 022010
Register area contents:
000000
000000
100307
040003
104000
140143
100007
000552
000200
012002
000000
000533
104000
000401
022000
000000
000001
000003
004572
000003
017176
000003
000063
000150
000000
000000
000372
040003
002501
002431
000000
000000
000000
```

Figure D-1 Subsystem Exception K-Detected Error (2 of 2)

D.2 OVERVIEW

The purpose of this appendix is to aid Field Service in analyzing the K-detected failure codes through the use of the status code tables. This appendix contains one status code table for each type of K:

- Table D-1 describes the K.ci status codes and applies only to requestor number 1.
- Table D-2 describes K.sdi status codes.
- Table D-3 describes the K.sti status codes.

D.3 HOW TO USE THE STATUS CODE TABLES

First of all, using these tables requires information as to the type of requestor involved. In order to determine which requestor detected the error, check the **Status of requestors (1-9):** field in the crash message. This field shows the status register contents of all requestors present in the subsystem.

NOTE

The registers referred to in this appendix are not general registers, but the internal K registers. All status codes followed by an * are hardware-detected errors. More detailed information for these errors is found in the appropriate sequencer error register.

The normal operational status codes for requestors are 001 for a K.ci, 002 for a K.sdi, and 203 for a K.sti. A 377 means no requestor is in the slot. Any value other than a 001, 002, 203, or 377 means the K detected an error. A K.ci-detected error always shows in the far left position in the **Status of requestors (1-9):** field of the message. In any other position, the type of requestor must be determined.

Count over the **Status of requestors (1-9)** field to the status contents showing an error (this is the requestor number). Type **SHOW REQUESTOR** at the SETSHO> prompt to see whether the requestor detecting the error is a K.sdi or a K.sti. Find the number of the data channel that found the error in the displayed response. This display shows whether that requestor number is either an K.sdi or a K.sti.

NOTE

If the HSC is not operational or the requestor in question fails initialization self tests, check the module utilization label above the card cage to determine whether the involved requestor number is a K.sdi or a K.sti.

Tables in this appendix consider only the rightmost two octal characters in failure code. Use the appropriate table (dependent upon requestor type) to find the meaning of the status code.

D.4 EXAMPLE EXAMINATION

Notice the third line of the message states the crash was caused by (134) Kint. The 134 indicates a K detected a fatal problem and interrupted the P.ioj with a Level 7 interrupt.

In this crash, requestor number 1 (the K.ci) status shows a 000177. The K.ci detected a fatal condition. The two digits in the status code are 77 (from the 000177 failure code).

Table D-1 provides additional information regarding status code 77. The description of this error indicates the HSC received a HOST CLEAR command from a host node. The description for the 77 status also shows the node number of the host which sent the HOST CLEAR is found in R17.

To find R17, look at the **Register area contents:** field on the second page of the example. The first entry in the register area contents is always the Q register from the K. The Q register contains important information for some crashes. The second entry is R0. In the example, count **in octal** up to R17 (remember the first entry is the Q register). The contents of R17 are 000001. Many of the error descriptions in the following tables indicate additional information exists in one of these registers.

Notice other entries below R17 in the register area contents. In the K.sdi and K.sti register areas, these other entries are RAM0 through RAM17, and they sometimes contain important information. On the K.ci, these entries are not significant for troubleshooting crash messages.

NOTE

A statement, See Note., appears in several places in the following tables. In each table, this information appears on the last page.

Table D-1 K.ci Status Bytes

Status Code (octal)	Description
00	Two conditions cause failure of the 2911 sequencer test upon powerup or reinitialization. In one case, the requestor sent status back to the P.io while Init was asserted. In the other case the sequencer had already released the Init signal, but the sequencer failed to reach the point in its code where it could change the status bits. A common occurrence of this status code is from an HSC false power fail crash dump. In this type of crash dump (IOT through 20), all requestors present report a 00 status code.
01	2901 ALU test failed upon powerup or reinitialization.
02	Data Bus (DBUS) test failed upon powerup or reinitialization.
03	Control Bus (CBUS) test failed upon powerup or reinitialization.
04	CROM test failed upon powerup or reinitialization.
06	K.pli RAM test failed upon powerup or reinitialization.
07	PLI interface test failed upon powerup or reinitialization.
10	Packet buffer test failed upon powerup or reinitialization.
11	LINK board test failed upon powerup or reinitialization.
12	Control Bus/memory error occurred during a lock cycle while the K.ci was attempting to locate the K-Init packet in Control memory upon powerup or reinitialization.
13	K.ci could not find a properly formatted K-Init packet in Control memory after completing power-up/init diagnostics.

Status Code (octal)	Description
14	An error was detected by the upper (control) sequencer. While attempting to update the next buffer pointer in an FRB, the pointer was found to be zero (illegal). R11 contains the FRB address.
15 *	An error was detected by the upper (control) sequencer. (See note.)
16	An error was detected by the upper (control) sequencer. The control stream found a structure on its own work queue that is not an HMB or FRB. R11 contains the structure address.
17	An error was detected by the upper (control) sequencer. While constructing a slot (SNDDAT, REQDAT) from an FRB, the FRB address was found to be zero (illegal). R12 contains the slot address.
20 *	An error was detected by the upper (control) sequencer. (See note.)
21	An error was detected by the upper (control) sequencer. A buffer allocate request was initiated without sufficient buffers on the Allocated queue in the control area to satisfy the request. R11 contains the FRB address.
22	An error was detected by the upper (control) sequencer. The queue head for an allocated Send buffer was zero.
23 *	An error was detected by the upper (control) sequencer. (See note.)
24	An error was detected by the lower (control) sequencer. The lower sequencer encountered an inconsistent internal data structure. R2 contains the message slot address.
25	An error was detected by the lower (control) sequencer. During the RTNDAT routine, the lower sequencer finds a zero (illegal) FRB address.
26	An error was detected by the lower (control) sequencer. The lower sequencer has received a packet from a node with a node ID greater than 63. R7 contains the node number.

Status Code (octal)	Description
27	An error was detected by the lower (control) sequencer. This error occurs when the lower sequencer polling loop calls a routine which adds or removes Big Message Block (BMB) pointers to or from the BMB chain, if the queue that is supposed to contain these pointers is empty.
30	An error was detected by the lower (control) sequencer. This error occurs when the lower sequencer determines that BMBs need to be returned to the free BMB pool and during a consistency check finds no BMBs to return. R2 contains the message slot address.
31 *	An error was detected by the upper (control) sequencer. (See note.)
32	An error was detected by the upper (control) sequencer. While attempting to transmit over a connection, the upper sequencer found an incarnation number of zero (invalid) in the connection block structure. R11 contains the HMB address, R14 contains the CB address.
33 through 41 *	An error was detected by the upper (control) sequencer. (See note.)
42	An error was detected by the upper (control) sequencer. A hardware error was detected following a block move to Control memory. R10 contains the upper processor error register contents. R16 contains the last Control memory address in the block that was moved.
43 *	An error was detected by the upper (control) sequencer. A hardware error was detected following a block move out of Control memory. R10 contains the upper processor error register contents. R16 contains the last control memory address in the block that was moved.
44 *	An error was detected by the upper (control) sequencer. A hardware error was detected following a Control memory receive operation. R10 contains the upper processor error register contents. R16 contains the Control memory address of the item received. R17 contains the Control memory address of the queue head.

Status Code (octal)	Description
45 and 46 *	An error was detected by the upper (control) sequencer. (See note.)
47 *	An error was detected by the upper (control) sequencer. A hardware error was detected during a downcount operation. R10 contains the upper processor error register value. R17 contains the counter address.
50 *	An error was detected by the upper (control) sequencer. A hardware error was detected while de-queueing a Control memory item from a scratchpad list. R10 contains the upper processor error register contents. R11 contains the Control memory address of the item.
51 *	An error was detected by the upper (control) sequencer. A hardware error was detected while internalizing an FRB. R10 contains the contents of the upper processor error register, R11 contains the FRB address, R14 contains the CB address. The Q register contains the work queue index.
52	An error was detected by the upper (control) sequencer. Either a consistency problem was found with the scratchpad queue or an attempt was made to send to a queue at address zero (illegal address).
53 through 55 *	An error was detected by the upper (control) sequencer. (See note.)
56 through 71 *	An error was detected by the lower (control) sequencer.(See note.)
72 *	An error was detected by the lower (control) sequencer.This error occurs while the lower processor is trying to link a BMB on the BMS free chain. R10 contains the lower processor error register contents. R5 contains the BMB data memory address.
73 *	An error was detected by the lower (control) sequencer. A hardware error was detected during a BMB list operation. R10 contains the lower processor error register contents. R5 contains the BMB data memory address.

Status Code (octal)	Description
74 *	An error was detected by the lower (control) sequencer. A hardware error was detected during a BMB list operation. R10 contains the lower processor error register contents. R5 contains the BMB data memory address.
75 *	An error was detected by the lower (control) sequencer. (See note.)
76	An error was detected by the upper (control) sequencer. While copying data from an HMB to a message slot, the upper sequencer found the byte count of the HMB was larger than the slot capacity. R12 contains the slot address. R17 contains the text length.
77	An error was detected by the upper (control) sequencer. A host clear sequence has been received. R17 contains the address of the issuing node number.

Status Code (octal)	Description

NOTE

The sequencers access Control memory several time before checking for a hardware error. Thus, to help determine the particular cause of the error, the sequencer saves the contents of the error register present at the time of the error check in R10 (octal). The contents of R10 are visible within the crash dump and can help in narrowing the error possibilities. The following lists show the bits available from both the Upper and Lower processor error registers. Those bits marked with (*) may cause a crash.

Upper Processor Error Register:

Bit 0 = Even/Odd Bit Control Memory Address
Bits 3,2,1 = CCYCLE 2,1,0
* Bit 4 = Control Bus Error (Illegal Cycle)
* Bit 5 = Control Bus NXM
* Bit 6 = Control Data Parity Error
* Bit 7 = Instruction (CROM) Parity Error
* Bit 8 = Scratchpad Parity Error
Bit 9 = PLI Parity Error
Bits 10 through 15 indicate the K.ci hardware revision level

Lower Processor Error Register:

Bit 0 = Data Memory Address Bit 16
Bit 1 = Data Memory Address Bit 17
Bit 2 = Data Memory NMA
* Bit 5 = Data Bus NXM
* Bit 6 = Data Memory Parity Error
* Bit 7 = Data Memory Overrun
* Bit 8 = Scratchpad Parity Error
* Bit 9 = PLI Parity Error
Bits 10 through 15 indicate the K.ci hardware revision level

Table D-2 K.sdi Status Bytes

Status Code (octal)	Description
00	Two conditions cause failure of the 2911 sequencer test upon powerup or reinitialization. In one case, the requestor sent status back to the P.io while Init was asserted. In the other case, the P.io had already released the Init signal, but the sequencer failed to reach the point in its code where it could change the status bits. A common occurrence of this status code is from an HSC false power fail crash dump. In this type of crash dump (IOT through 20), all requestors present report a 00 status code.
01	2901 ALU test failed upon powerup or reinitialization.
02	Data Bus (DBUS) test failed upon powerup or reinitialization.
03	Control Bus (CBUS) test failed upon powerup or reinitialization.
04	PROM test failed upon powerup or reinitialization.
06	Scratchpad RAM test failed upon powerup or reinitialization.
07	R-S/Gen test failed upon powerup or reinitialization.
10	Partial SDI test failed upon powerup or reinitialization.
12	The K.sdi encountered a Control Bus/memory problem while searching for the K-Init packet in Control memory.
13	After completing power-up/init diagnostics, the K.sdi could not find a properly formatted K-Init packet in Control memory.
14	While trying to write the microcode version into the control area at address R7+44 (R7 is base address), the upper sequencer encountered a Control Bus error. R11 contains the contents of the upper error register. (See note.)

Status Code (octal)	Description
15	This error occurs if the upper processor tries to advance the buffer descriptor pointer if the old value of the pointer is zero (illegal).
16	While attempting to read the block number (LBN) from a buffer descriptor in Control memory, the upper processor encountered a hardware error. R11 contains the contents of the upper error register. (See note.)
17 through 30 *	The upper processor encountered an error while attempting to access Control memory. R11 contains the upper processor error register contents. (See note.)
31	This error occurs if, during transfer completion, a DRAT counter goes to zero and the DRAT list head in the control area is not locked and not equal to the current DRAT value.
32 through 42 *	The upper processor encountered an error while attempting to access Control memory. R11 contains the upper processor error register contents. (See note.)
43	This error occurs while processing an active DCB, if the dialogue state indicator is not locked (a value of 100000 is not in KS$DHD) and not valid (KS$IND does not contain the values 0, 1, 2, 3, OR4, or -1).
44	The upper processor encountered an error while attempting to access Control memory. R11 contains the upper processor error register contents. (See note.)
45	This error occurs if, after completing state 0 processing, the upper sequencer cannot find a valid DCB opcode. (No valid state is present to go to next.)
46 through 55 *	The upper processor encountered an error while attempting to access Control memory. R11 contains the upper processor error register contents. (See note.)
74 through 76	The upper processor attempted to downcount a counter that was already at zero.

Status Code (octal)	Description

NOTE

The upper sequencer accesses Control memory several times before checking for a Control Bus error. Thus, to help determine the particular cause of the error, the upper sequencer saves the contents of the error register present at the time of the error in R11 (octal). The contents of R11 are visible within the crash dump and may help in narrowing the error possibilities. The following list defines all the bits contained within the upper processor error register (value loaded in R11). Those bits that can possibly cause a crash are denoted with an asterisk (*).

Upper Processor Error Register:

Bit 0 = Even/Odd bit Control Memory Address
Bits 3,2,1 = CCYCLE 2,1,0
* Bit 4 = Control Bus Error (Illegal Cycle)
* Bit 5 = Control Bus NXM
* Bit 6 = Control Data Parity Error
* Bit 7 = Instruction (CROM) Parity Error
Bits 8 through 12 not used
* Bit 13 = Response Pulse Missing on SDI RD/RES Line
Bit 14 = Upper Processor RTC Clock Pulse Present
Bit 15 = Parity Error on RTDS Line

Table D-3 K.sti Status Bytes

Status Code (octal)	Description
00	Two conditions cause failure of the 2911 sequencer test upon powerup or reinitialization. In one case, the requestor sent status back to the P.io while Init was asserted. In the other case, the sequencer had already released the Init signal, but the sequencer failed to reach the point in its code where it could change the status bits. A common occurrence of this status code is from an HSC false power fail crash dump. In this type of crash dump (IOT through 20), all requestors present report a 00 status code.
01	2901 ALU test failed upon powerup or reinitialization.

Status Code (octal)	Description
02	Data Bus (DBUS) test failed upon powerup or reinitialization.
03	Control Bus(CBUS) test failed upon powerup or reinitialization.
04	PROM test failed upon powerup or reinitialization.
06	Scratchpad RAM test failed upon powerup or reinitialization.
07	SERDES test failed upon powerup or reinitialization.
10	Partial STI test failed upon powerup or reinitialization.
12	The K.sti encountered a Control Bus/memory problem while searching for the K-Init packet in Control memory.
13	After completing power-up/init diagnostics, the K.sti could not find a properly formatted K-Init packet in Control memory.
14 through 22 *	Control Bus error. (See note.)
23	During transfer completion, the buffer descriptor link word in the FRB was zero. RAM7 contains the lower processor status.
24 through 33 *	Control Bus error. (See note.)
34	The lower processor has timed out on a transfer operation and the upper processor cannot restart it.
35 and 36 *	Control Bus error. (See note.)
37	A software inconsistency. The STI state zero processing code was entered when the drive state indicator was not zero.
40	State zero processing is complete. However, the next state (such as Send Level 1 frame, or Get Drive Status) is not specified. Thus, the state is undefined.

Status Code (octal)	Description
41 through 43 *	Control Bus error. (See note.)
44	While setting up a transfer, the next buffer descriptor in the FRB was zero (no buffer was there).
74	Attempted to downcount a counter that was already zero. R14 contains the FRB. R16 contains the counter minus one. R17 contains the address of the counter structure.
75 and 76 *	Control Bus error. (See note.)

NOTE

The upper sequencer accesses Control memory several times before checking for a Control Bus error. Thus, to help determine the particular cause of the error, the upper sequencer saves the contents of the error register present at the time of the error in R11 (octal). The contents of R11 are visible within the crash dump and may help in narrowing the error possibilities. The following list defines all the bits contained within the upper processor error register (value loaded in R11). Those bits that can possibly cause a crash are denoted with an asterisk (*).

Upper Processor Error Register:

Bit 0 = Even/Odd bit Control Memory Address
Bits 3,2,1 = CCYCLE 2,1,0
* Bit 4 = Control Bus Error (Illegal Cycle)
* Bit 5 = Control Bus NXM
* Bit 6 = Control Data Parity Error
* Bit 7 = Instruction (CROM) Parity Error
Bits 8 through 12 not used
* Bit 13 = Response Pulse Missing On SDI RD/RES Line
Bit 14 = Upper Processor RTC Clock Pulse Present
Bit 15 = Parity Error On RTDS Line

APPENDIX E
HSC70 REVISION MATRIX CHART

E.1 INTRODUCTION

Figure E-1 shows the revision status of all applicable HSC70 FRUs. An HSC70 must have all the FRUs at a particular revision level in order to be supported. Initial release of HSC70-AA (120/208 VAC, 60Hz) and HSC70-AB (380-415 VAC, 50 Hz) including V100 software, are at revision A1.

HSC70 - AA/CA		REV	A1				B1												
NUMBER	DESCRIPTION		REVISIONS																
L0100 - 00	ILI (CI LINK)	B - ETCH	D1	→															
		C - ETCH	D1	→															
L0107 - YA	K.pli		C2	C3	→														
L0108 - YA (HSC5X - BA)	K.sdi	C - ETCH	C8	→															
		D - ETCH	C8	C9	C10	→													
		E - ETCH	C1	C2	C3	→													
		F - ETCH	C22				C22	C23											
L0108 - YB (HSC5X - CA)	K.sti	D - ETCH	C10	→															
		E - ETCH	C3	C4	→														
		F - ETCH	C23				C23	C24											
L0109 - 00	PILA		E1	E2	→														
L0111 - 00	P.ioj	C - ETCH	A1	→															
		D - ETCH	A2	→															
L0117 - AA	M.std2	A - ETCH	A2	→															
5417764 - 01	BACKPLANE	C - ETCH	A1				C1	D1											

CX–1271A
Sheet 1 of 4

Figure E-1 HSC70 Revision Matrix Chart (1 of 4)

HSC70 - AA/CA		REV	A1		B1														
NUMBER	DESCRIPTION	REVISIONS																	
70 - 20033 - 03	STD PS ASSY - 120VAC IN	C1	→																
70 - 20184 - 01	OPT PS ASSY - 120VAC IN	B2	→																
30 - 24374 - 01	881A PWR CNTR ASSY	B1	→																
70 - 23138 - 01	OCP ASSEMBLY	A2	→																
54 - 15286 - 01**	OCP	C	→																
70 - 23129 - 01	FLOPPY DRIVE BKT ASSY	A2	→																
30 - 24962 - 01	RX33 DRIVE	A1	→																
EK - HSC70 - IN - xxx	INSTALLATION MANUAL	001	→																
QX926 - H7	HSC70 SOFTWARE	V100		V300															
BL - FH74x - DE	HSC70 OFFLINE DIAGS	A	→																

**THIS BREAKDOWN IS FOR FIELD SERVICE INFORMATION ONLY.

CX–1271A
Sheet 2 of 4

Figure E-1 HSC70 Revision Matrix Chart (2 of 4)

HSC70 - AB/CB		REV	A1				B1												
NUMBER	DESCRIPTION		REVISIONS																
L0100 - 00	ILI (CI LINK)	B - ETCH	D1	→	→	→													
		C - ETCH	D1	→	→	→													
L0107 - YA	K.pli		C2	C3	→	→													
L0108 - YA (HSC5X - BA)	K.sdi	C - ETCH	C8	→	→	→													
		D - ETCH	C8	C9	C10	→													
		E - ETCH	C1	C2	C3	→													
		F - ETCH	C22				C22	C23											
L0108 - YB (HSC5X - CA)	K.sti	D - ETCH	C10	→	→	→													
		E - ETCH	C3	C4	→	→													
		F - ETCH	C23	→	→	→	C23	C24											
L0109 - 00	PILA		E1	E2	→	→													
L0111 - 00	P.ioj	C - ETCH	A1	→	→	→													
		D - ETCH	A2	→	→	→													
L0117 - AA	M.std2	A - ETCH	A2	→	→	→													
5417764 - 01	BACKPLANE	C - ETCH	A1				C1	D1											

CX–1271A
Sheet 3 of 4

Figure E-1 HSC70 Revision Matrix Chart (3 of 4)

HSC70 - AB/CB	REV	A1		B1															
NUMBER	DESCRIPTION	REVISIONS																	
70 - 20033 - 04	STD PS ASSY - 240VAC IN	C1	→																
70 - 20184 - 02	OPT PS ASSY - 240VAC IN	B2	→																
30 - 24374 - 02	881B PWR CNTR ASSY	B1	→																
70 - 23138 - 01	OCP ASSEMBLY	A2	→																
54 - 15286 - 01**	OCP	C	→																
70 - 23129 - 01	FLOPPY DRIVE BKT ASSY	A2	→																
30 - 24962 - 01	RX33 DRIVE	A1	→																
EK - HSC70 - IN - xxx	INSTALLATION MANUAL	001	→																
QX926 - H7	HSC70 SOFTWARE	V100		V300															
BL - FH74x - DE	HSC70 OFFLINE DIAGS	A	→																

**THIS BREAKDOWN IS FOR FIELD SERVICE INFORMATION ONLY.

CX–1271A
Sheet 4 of 4

Figure E-1 HSC70 Revision Matrix Chart (4 of 4)

-A-

AC power
  Removing, 3-1
ACK/NAK generation, 1-13
Address switches, node, 2-9
Airflow sensor assembly
  Removing, 3-15
Auxiliary power supply
  Removing, 3-21

-B-

BBR errors, 8-42
Block Diagram, 1-10 figure
Blower, 1-5
  Removing, 3-13
Booting procedures, 4-2
Booting the Offline Diagnostics diskette, 6-2
Booting the System diskette, 4-2

-C-

Cables
  Backplane to bulkhead, 1-6
  Bulkhead to outside, 1-6
  CI, 1-7
  CI bus, 1-7
  SDI, 1-7
  SDI bus, 1-7
  STI, 1-7
  STI bus, 1-7
Cache Test Descriptions, 6-29
Cache Test Parameters, 6-23
CI Bus
  Connecting to multiple hosts, 1-1
CI cables
  Port link module interfaces, 1-14
CI Errors, 8-71
CI manager, 1-9
Console Terminal
  Troubleshooting, 8-7
Console terminal connection), 4-1
Control Bus Error Conditions (Hardware Detected), 8-112
Control memory size, 1-16
Controls and indicators
  DC power switch, 2-5
  Enable indicator, 2-4
  Operator control panel, 2-1
  Secure/Enable switch, 2-3
Cooling, 1-2, 1-5

-D-

Data memory size, 1-16
DC power
  Removing, 3-3
DC Power Switch
  Location of, 3-4figure
DC power switch, 2-5
DEFAULT command, DKUTIL, 7-8
Defaults, utility prompts, 7-1
Diagnostic manager, 1-10
Diagnostic subroutines, 1-10
Disabling P.ioj parity errors, 6-60
Disk functional errors, 8-84
Disk I/O manager, 1-9
DISPLAY command, DKUTIL, 7-10
DKUTIL
  commands, 7-8
  error messages, 7-20
DKUTIL command descriptions, 7-7
DKUTIL command modifiers, 7-3
DKUTIL command syntax, 7-2
DKUTIL Initiation, 7-1
Documents
  ordering, 1-20
Door
  Back, opening, 3-5
  Front, opening, 3-4
DUMP command, DKUTIL, 7-12

-E-

Error classes, VERIFY, 7-21
Error message format, generic, 6-10
Error message severity levels, DKUTIL, 7-17
Error message severity levels, VERIFY, 7-25

Error message variables, DKUTIL, 7-17
Error message variables, FORMAT, 7-34
Error messages, FORMAT, 7-38
Error processor, 1-9
Errors
  Aborting Error Recovery Due to Excessive RECALS, 8-84
  Aborting Error Recovery Due to Excessive Timeouts, 8-84
  Acknowledge Not Asserted At Start Of Transfer, 8-58
  ATN. message sent to Node xx, for Unit xx, 8-85
  Attention Condition serviced for ONLINE disk unit xxx, 8-85
  Bad Block Replacement (Block OK), 8-45
  Bad Block Replacement (Drive Inoperative), 8-45
  Bad Block Replacement (RCT Inconsistant), 8-45
  Bad Block Replacement (REPLACE Failed), 8-46
  Bad Block Replacement (Success), 8-46
  Bad dispatch state in CB..., 8-80
  Booted from drive 1. Drive 0 Error (text), 8-101
  Buffer EDC Error, 8-59
  Cables have gone from uncrossed to crossed, 8-72
  Cache disabled due to failure, 8-102
  Cannot Clear Drive Errors, 8-59
  Cannot Clear Formatter Errors, 8-59
  Clock dropout from ONLINE disk unit xx, 8-85
  Compare Error, 8-17
  Controller Detected Position Lost, 8-60
  Controller Transfer Retry Limit Exceeded, 8-60
  Controller-Detected Transmission or Time Out Error, 8-28
  Could Not Complete Online Sequence, 8-60
Errors (Cont.)
  Could Not Get Extended Drive Status, 8-61
  Could Not Get Formatter Summary Status During Transfer Error Recovery, 8-61
  Could Not Get Formatter Summary Status While Trying To Restore Tape Position, 8-61
  Could Not Position For Formatter Retry, 8-62
  Could Not Set Byte Count, 8-62
  Could Not Set Unit Characteristics, 8-62
  Data Bus Overrun, 8-17
  Data Error Flagged in Backup Record, 8-93
  Data Memory Error (NXM or Parity), 8-18
  Data Overflow Due To Pipeline Error, 8-63
  Data Ready Timeout, 8-63
  Data Synch Not Found, 8-39
  Date/Time set by node nn, 8-71
  Deferred ATN. message for Node xx, Unit xx, 8-86
  Disk Unit xx (Requestor xx, Port xx) being initialized, 8-86
  Disk unit xx ready to transfer.!, 8-86
  Disk unit xxx.(requestor xx.,Port xx.) declared inoperative, 8-87
  DRAT/SEEK timeout, disk unit xxx, 8-87
  DRIVE CLEAR attempt on disk unit xx, 8-88
  Drive Clock Dropout, 8-28
  Drive Inoperative, 8-29
  Drive-Detected Error, 8-29
  Drive-Requested Error Log (EL Bit Set), 8-30
  Duplicate disk unit xx, 8-88
  ECC Errors, 8-40
    Eight Symbol, 8-40
    Five Symbol, 8-40
    Four Symbol, 8-40
    One Symbol, 8-40
    Seven Symbol, 8-40
    Six Symbol, 8-40
    Three Symbol, 8-40

Errors
ECC Errors (Cont.)
Two Symbol, 8-40
Uncorrectable, 8-40
EDC Error, 8-18
Erase Command Failed, 8-64
Erase Gap Command Failed, 8-64
Forced Error, 8-41
Formatter And HSC Disagree On Tape Position, 8-64
Formatter Detected Position Lost, 8-65
Formatter Requested Error Log, 8-65
Formatter Retry Sequence Exhausted, 8-65
FRB error: K.ci, lst LBN xx buffers, FE$SUM xx, 8-88
FRB error: K.sdi, Unit xx, first LBN xxx, buffers, FE$SUM, 8-89
Hard transfer error loading (file) xx, 8-102
Hard transfer error writing SCT xx, 8-103
Header Error, 8-41
HML$ER set - HM$ERR = nn, 8-78
Host Clear from CI node, 8-103
Host interface (K.ci) failed INIT diags, status = xxx, 8-104
Host interface (K.ci) is required but not present, 8-104
Host Requested Retry Suppression On A Formatter Detected Error, 8-66
Host Requested Retry Suppression On A K.sti Detected Error, 8-66
Illegal bit change in status from disk unit xxx, 8-89
Insufficient Control Memory for K.sti in Requestor xx, 8-93
Insufficient Private Memory remaining for TMSCP Server, 8-94
Internal Consistency Error, 8-19
K.ci exception detected, code = nnn, 8-75
Errors (Cont.)
K.ci loopback microcode loaded, 8-80
K.sdi in slot xx failed its init diagnostics, status = xxx, 8-89
K.sti in Requestor xx has microcode incompatable with this TMSCP Server, 8-94
Last soft init resulted from unknown cause, 8-105
LBN Restored with Forced Error in RESTOR Operation!, 8-90
Less than 87.5% of xx memory is available, 8-105
Level 7 K Interrupt (Trap thru 134), 8-112
Level 7 K interrupt trap thru 134, 8-107
Lost Read/Write Ready, 8-30
Lost Receiver Ready, 8-31
Lower Processor Error, 8-66
Lower Processor Timeout, 8-67
MMU (Trap thru 250), 8-115
MMU Trap thru 250, 8-107
No control block available to satisfy HMB request., 8-78
No Tape Drive Structures available for Requestor xx Port xx Unit xx Increase Structures via SET MAXTAPE command, 8-95
No Tape Formatter Structures available for Requestor xx Port xx Increase structures via SET MAXFORMATTERS command, 8-95
No usable K.sti boards were found by the TMSCP Server, 8-95
Node nn Cables have gone from crossed to uncrossed, 8-73
Node nn Path (A or B) has gone from good to bad, 8-73
Node nn Path n has gone from bad to good, 8-74
NXM (Trap thru 4), 8-111
NXM Trap thru 4, 8-108
P.ioj running with memory bank or board swap enabled, 8-105
Parameter change, 8-108

Errors (Cont.)
Parity Error Trap thru 114, 8-109, 8-111
PLI Receive Buffer Parity Error, 8-19
PLI Transmit Buffer Parity Error, 8-20
Position or Unintelligible Header Error, 8-31
Positioner error on disk unit xxx. DRAT addr:xxx, 8-90
Premature LP flag in RTNDAT sequence from host node xx, 8-90
Pulse or Parity Error, 8-32
RCT Corrupted Error, 8-42
Receiver Ready Not Asserted At Start Of Transfer, 8-67
Record EDC Error, 8-67
Requestor xx failed INIT diags, status = xxx, 8-106
Requestor xx has failed initialization dignostics with status = xx, 8-96
Reserved Instruction (Trap thru 10), 8-111
Reserved Instruction Trap thru 10, 8-109
Resource lost to K.ci -- xxx xxx HMBs, 8-80
Retry Limit Exceeded While Attempting To Restore Tape Position, 8-68
Reverse Retry Currently Not Supported, 8-68
Rewind Failure, 8-68
SCT read or verification error. Using template SCT., 8-106
SDI exchange retry on disk unit xxx, 8-91
SERDES Overrun, 8-19
SI Clock Persisted After INIT, 8-33
SI Clock Resumption Failed After INIT, 8-32
SI Command Timeout, 8-33
SI Receiver Ready Collision, 8-34
SI Response Length or Opcode Error, 8-35
SI Response Overflow, 8-35
Errors (Cont.)
Software Inconsistency (Trap thru 20), 8-118
Software inconsistency Trap thru 20, 8-110
Tape Drive Requested Error Log, 8-69
Tape Formatter declared inoperative, 8-97
Tape unit number xx connected to Requestor xx Port xx Ceased to exist while Online, 8-96
Tape unit number xx connected to Requestor xx Port xx dropped state clock, 8-96
Tape unit number xx connected to Requestor xx Port xx Is not asserting Available when it should be, 8-98
Tape unit number xx connected to Requestor xx Port xx Went Available without request, 8-98
Tape unit number xx connected to Requestor xx Port xx Went Offline without request, 8-99
TMSCP fatal initialization error - TMSCP functionality not available, 8-99
TMSCP Server operation limited by insuffcient Private Memory, 8-100
Topology Command Failed, 8-69
TTRASH fatal initialization error, 8-100
Unable To Position To Before LEOT, 8-69
Unexpected AVAILABLE signal from ONLINE disk unit xx, 8-91
Unknown K.tape Error, 8-70
Unrecoverable error on disk unit xx. Drive appears inoperative, 8-91
Unsuccessful SEEK initiation, disk unit xxx. DCB addr: xxx, 8-92
VC closed due to timeout of RTNDAT/CNF from host node xx, 8-92

Errors (Cont.)
VC closed with node nn due to disconnect timeout, 8-76
VC closed with node nn due to request from K.ci, 8-77
VC closed with node nn due to START received, 8-77
VC closed with node nn due to unexpected disconnect, 8-76
VC open with node nn, 8-72
WARNING K.sti microcode too low for large transfers., 8-101
Word Rate Clock Timeout, 8-70
Event Codes
MSCP, C-3
TMSCP, C-3
Exception codes and messages, B-1
EXIT Command, DKUTIL, 7-14
External interfaces, 1-7

-F-

Fatal error messages, FORMAT, 7-35
Fatal error messages, VERIFY, 7-25
Fault code interpretation, 4-4
Fault codes, 4-6
FORMAT, CAUTION, 7-30
FRU
Removal sequence, 3-4

-G-

GEDS Text Field
Breakdown of, 8-53
General information, 1-1
GET command, DKUTIL, 7-14
GSS Text Field
Breakdown of, 8-55

-H-

HSC70 control program, 1-8
HSC70 maintenance strategy, 1-17
HSC70 specifications, 1-19

-I-

ILDISK error messages
error 01 DDUSUB initialization failure, 5-13
ILDISK error messages (Cont.)
error 02 unit selected is not a disk., 5-13
error 03 drive unavailable., 5-14
error 04 unknown status from DDUSUB., 5-14
ILEXER data patterns, 5-60
ILMEMY error messages
error 000 tested twice with no error., 5-8
error 001 returned buffer to free buffer queue., 5-8
error 002 memory parity error., 5-8
error 003 memory data error., 5-8
Iltape Error Messages
error 1 - initialization failure, 5-39
error 1 - requested device is busy, 5-40
error 10 - load device write error - check if write locked, 5-40
error 11 - command failure, 5-40
error 12 - read memory byte count error, 5-40
error 13 - formatter diagnostic detected error, 5-40
error 14 - formatter diagnostic detected fatal error, 5-40
error 15 - RX33 read error, 5-41
error 16 - insufficient resources to acquire specified device, 5-41
error 17 - k microdiagnostic did not complete, 5-41
error 18 - k microdiagnostic reported error, 5-41
error 19 - dcb not returned, k failed for unknown reason, 5-41
error 2 - selected unit not a tape, 5-39
error 20 - error in DCB upon completion, 5-41
error 21 - unexpected item on drive service queue, 5-41

Iltape Error Messages (Cont.)
error 22 - state line clock not running, 5-41
error 23 - init did not stop state line clock, 5-41
error 24 - state line clock did not start up after init, 5-41
error 25 - formatter state not preserved across init, 5-42
error 26 - echo data error, 5-42
error 27 - receiver ready not set, 5-42
Error 28 - available set in online formatter, 5-42
error 29 - RX33 error - file not found, 5-42
error 3 - invalid requestor/port number, 5-39
error 30 - data compare error, 5-42
error 31 - edc error, 5-42
error 32 - invalid multiunit code from GUS command, 5-42
error 33 - insufficient resources to acquire timer, 5-42
error 34 - unit unknown or online to another controller, 5-43
error 4 - requestor not a k.sti, 5-40
error 5 - timeout acquiring drive service area, 5-40
error 6 - requested device unknown, 5-40
error 8 - unknown status from tape diagnostic, 5-40
error 9 - unable to release device, 5-40
Iltape Prompts
drive unit number (u) []?, 5-32
enter port number (0-3) []?, 5-33
enter requestor number (2-9) []?, 5-33
execute formatter diagnostics (yn) [y]?, 5-33
execute test of tape transport (yn) [n]?, 5-33
Iltape Prompts (Cont.)
is media mounted (yn) [n]?, 5-33
memory region number (h) [0]?, 5-33
ILTCOM Error Messages
error 1 - initialization failure, 5-50
error 10 - can't find end of bunch, 5-50
error 11 - data compare error, 5-50
error 12 - data EDC error, 5-51
error 2 - selected unit not a tape, 5-50
error 3 - command failure, 5-50
error 5 - specified unit not available, 5-50
error 6 - specified unit cannot be brought online, 5-50
error 7 - specified unit unknown, 5-50
error 8 - unknown status from TDUSUB, 5-50
error 9 - error releasing drive, 5-50
Information messages, FORMAT, 7-37
Information messages, VERIFY, 7-26
Informational messages, VERIFY, 7-29
Initialization error indications, 8-2
Initiating FORMAT, 7-31
Initiating VERIFY, 7-22
Inline Diagnostics
inline disk drive diagnostic test (ILDISK), 5-9
inline memory test (ILMEMY), 5-6
inline multidrive exerciser (ILEXER), 5-51
inline tape compatability test (ILTCOM), 5-44
inline tape test (ILTAPE), 5-31
Inline diagnostics generic error message format, 5-1
Inline diagnostics generic prompt syntax, 5-1
Inline RX33 Diagnostic Test (ILRX33), 5-2

Internal Software, 1-8 figure
Internal software, 1-8

-K-

K.pli
  See Port processor module
K.sdi
  See Logic modules
    Disk data channel
K.sti
  See Logic modules
    Tape data channel

-L-

Load device
  See RX33 disk drives
Load device errors, 8-81
Loader commands, 6-12
Loader DEPOSIT Command, 6-15
Loader EXAMINE Command, 6-14
Loader HELP Command, 6-12
Loader LOAD Command, 6-14
Loader SIZE Command, 6-13
Loader START Command, 6-14
Loader TEST Command, 6-13
Logic Modules
  LEDS, functions of, 2-8 table
  Port Buffer Module (PILA),
      functions of, 1-14
  Port processor module,
      interfaces, 1-14
Logic modules
  Card cage
    Module utilization label, 1-4
  Disk data channel, functions of,
      1-14
  I/O control processor module,
      functions, 1-15
  Indicators and switches, 2-6
  Memory module, functions, 1-16
  Port link module (LINK),
      functions of, 1-12
  Port processor module,
      functions, 1-14
  Removing, 3-11
  Switches, 2-9
    Port link buffer, 2-10
  Tape data channel, functions,
      1-15
Logic modules, descriptions, 1-11
    to 1-18

-M-

M.std2
  See Logic modules
    Memory module
Main power supply
  Removing, 3-18
Maintenance features, 1-18
Message severity levels, FORMAT,
    7-35
Microcode detected errors
  K.ci, D-11
  K.sdi, D-14
  K.sti, D-16
Miscellaneous errors, 8-101
Module indicators, 2-6
Module LEDS
  Data Channel LEDs, 8-6
  Host Interface LEDs, 8-6
  Memory module LEDs, 8-5
  P.ioj LEDs, 8-4
    Power up sequence, 8-4
Module Nomenclature, 1-12 table
Module Switches
  Module s/n switches, 2-10
      figure
Module switches
  Node address switches, 2-9
Module Utilization Label, 1-5
    figure
Moving Inversions aLgorithm, 6-42
MSCP errors, 8-13
  Controller error list, 8-16
  Disk Transfer Errors, 8-35
  SDI Errors, 8-20
MSCP processor, 1-9

-N-

Node address switches, 2-9

-O-

OCP Fault code displays, 8-2
Offline bus interaction test
    error messages
  Error 000 - Memory test error,
      6-39

Offline bus interaction test error messages (Cont.)
Error 001 - K timed-out during init., 6-39
Error 002 - K timed-out during test., 6-40
Error 003 - Parity trap., 6-40
Error 004 - NXM trap, 6-40
Error 005 - Memory test error (P.ioj detected)., 6-41
Error 010 - Cache parity trap, 6-41
Error 011 - RX33 drive not ready, 6-41
Error 012 - RX33 CRC error during seek., 6-41
Error 013 - RX33 track 0 not set on recalibrate., 6-42
Error 014 - RX33 seek timeout, 6-42
Error 015 - RX33 seek error., 6-42
Error 016 - RX33 read timeout., 6-42
Error 017 - RX33 CRC/RNF error on read command., 6-42
Offline cache cumulative soft error results, 6-24
Offline Cache Test Error Messages
error 00 - memory parity error, 6-25
error 01 - NXM trap, 6-25
error 02 - cache parity error, 6-25
error 03 - bit stuck in cache control register., 6-25
Offline Cache Test Error messages
Error 04 - Forced miss operation failed., 6-25
Offline cache test error messages
Error 05 - Forced miss with abort failed., 6-25
Error 06 - Expected cache hit did not occur., 6-25
Error 07 - Expected cache miss did not occur., 6-25
Error 10 - Value in hit/miss register incorrect., 6-26
Error 11 - Write byte operation caused cache update., 6-26
Error 12 - Write byte did not cause cache update., 6-26
Offline cache test error messages (Cont.)
Error 13 - Cache failed to flush successfully., 6-26
Error 14 - Access with force bypass did not cause invalidate., 6-26
Error 15 - Tag Parity error bit did not set., 6-26
Error 16 - Abort on cache parity error did not occur., 6-26
Error 17 - Unexpected parity trap during abort test., 6-26
Error 20 - Content of memory system error register incorrect., 6-27
Error 21 - Return PC wrong during abort/interrupt test., 6-27
Error 22 - Cache data parity bit(s) did not set., 6-27
Error 23 - Interrupt on parity error did not occur., 6-27
Error 24 - Expected NXM trap did not occur., 6-27
Error 25 - Parity error was not blocked by NXM., 6-27
Error 26 - Cache data miscompare on word operation., 6-27
Error 27 - Cache data miscompare on byte operation., 6-27
Error 30 - DMA write to memory did not cause cache to invalidate., 6-27
Error 31 - Instruction still completed during abort condition., 6-28
Error 32 - Load device error during DMA test., 6-28
Error 33 - PDR cache bypass failed., 6-28
Error 34 - Tag store address hit failure., 6-28
Error 35 - Tag store address miss failure., 6-28
Error 41 - Processor type is not J11., 6-28

Offline Diagnostics
  Offline Bus Interaction Test, 6-33
  offline cache test, 6-22
  offline diagnostic loader, 6-11
  Offline K Test, 6-43
  Offline K/P Memory Test, 6-57
  Offline Memory Test, 6-73
  Offline Operator Control Panel Test, 6-104
  Offline Refresh Test, 6-100
  RX33 Offline Exerciser, 6-89
Offline diagnostics
  P.ioj ROM Bootstrap, 6-2
Offlines common characteristics, 6-1
Offlines generic error message format, 6-10
Operator control panel, 2-2figure
  Blank indicators, 2-3
  Fault codes, 2-3
  Fault indicator and switch, 2-2
  Init switch, 2-2
  Lamp test, 2-3
  Online indicator, 2-3
  Online switch, 2-3
  Power indicator, 2-2
  Removing, 3-9
  State and Init indicators, 2-1
Out-of-band errors, 8-70
  Categories of, 8-70

-P-

P.ioc
  See logic modules
    I/O control processor module
Packaging, 1-2
Packet reception, 1-13
Packet transmission, 1-13
Power, 1-2
Power control bus
  See power controller, 1-6
Power Controller
  rotating the line cord elbow, 3-16
Power controller
  Bus/off/on switch, 2-12
  Circuit breaker, 2-12
  Delayed output line, 1-6
  Description of, 2-10
  Fuse, location of, 2-12
Power controller (Cont.)
  Noise isolation filters, 1-6
  Operating instructions, 2-10
  Power control bus, 1-6
  Power control bus connections, 2-12
  Removing, 3-16
  Total Off connector and power up, 2-12
Program memory size, 1-16
PUSH Command, DKUTIL, 7-15

-R-

Removing power, 3-1
Requestor Error Summary, 6-38
REVECTOR command, DKUTIL, 7-16
RX33 cover plate
  Removal, 3-5
RX33 Disk Drives
  jumper configurations, 3-8, 3-9
RX33 disk drives, as program load device, 2-5
RX33 Diskette controller location, 1-16
RX33 error code tables, 6-10
RX33 Exerciser Data Patterns, 6-99

-S-

Safety precautions, 3-1
SDI manager, 1-10
Secure/Enable switch, 2-3
SET Command, DKUTIL, 7-16
SINI errors
  See miscellaneous errors, 8-101
Software, 1-8, 1-9, 1-10, 1-15
Software Error Messages
  Categories of, 8-13
Software Release Notes, 1-1, 1-15
Status bytes interpretation
  K.ci, D-11
  K.sdi, D-14
  K.sti, D-16
STI bus
  Maximum number of tape formatters, 1-1
STI Communication or Command Errors, 8-46
STI manager, 1-9
Subsystem block diagram, 1-10

Success messages, FORMAT, 7-38
Switches, node address, 2-9
Symbolic addresses, DEPOSIT and EXAMINE commands, 6-15

-T-

Tape functional errors, 8-92
Tape I/O manager, 1-9
Terminal connection), 4-1
Troubleshooting, Cache, 6-28
Typical Bus Interaction Test error message, 6-37

-U-

Utilities manager, 1-10
Utility processes, 1-10

-V-

Variable output fields, VERIFY, 7-25
VERIFY process, 7-20
VERIFY Type error messages, VERIFY, 7-28
VERIFY/FORMAT, 7-34

-W-

Warning message, FORMAT, 7-37
Warning messages, VERIFY, 7-26

Digital Equipment Corporation • Colorado Springs, CO 80919