HSC Service Manual

Order Number EK-HSCMA-SV-002

Digital Equipment Corporation
Maynard, Massachusetts

Second Edition, December 1989

DEC	DIBOL	UNIBUS
DEC/CMS	EduSystem	VAX
DEC/MMS	IAS	VAXcluster
DECnet	MASSBUS	VMS
DECsystem–10	PDP	VT
DECSYSTEM–20	PDT	
DECUS	RSTS	
DECwriter	RSX	digital™

4,475,212	4,434,487	4,413,339
4,468,035	4,543,626	4,592,072
4,241,399	4,338,663	4,349,871
4,450,572		and other patents pending

4,475,212	4,434,487	4,413,339
4,468,035	4,543,626	4,592,072
4,241,399	4,338,663	4,349,871
4,450,572		and other patents pending

This document was prepared using VAX DOCUMENT, Version 1.1

Contents

3 Removal and Replacement Procedures

4 Initialization Procedures

5 Device Integrity Tests

8 Troubleshooting Techniques

A Internal Cabling Diagrams

B Exception Codes and Messages

C Generic Error Log Fields

D Interpretation of Status Code Bytes

E Revision Matrix Charts

Examples

Figures

Tables

About This Manual

This manual contains servicing information and procedures for the HSC70, HSC50 (modified), HSC50, and HSC40 subsystems. In this manual, HSC refers to the HSC70 and HSC40 models. HSC50 refers to the HSC50 and HSC50 (modified) models. Individual model names are used only when the information is model specific.

This manual describes HSC 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 service personnel with HSC maintenance. Operational theory is included wherever such background is helpful to service personnel.

Installation procedures, most HSC utilities, and detailed 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 below.

Audience

This manual is intended for use by Level 1 Digital Field Service Engineers and other personnel in maintaining the components of the HSC controller subsystem.

Scope

This manual is divided into the following chapters:

1. General Information
2. Controls and Indicators
3. Removal and Replacement Procedures
4. Initialization Procedures
5. Device Integrity Tests
6. Offline Diagnostics
7. Utilities
8. Troubleshooting Techniques
9. Appendixes:
    A. Internal Cabling Diagrams
    B. Exception Codes and Messages
    C. Generic Error Log Fields
    D. Interpretation of Status Code Bytes
    E. Revision Matrix Charts

Related Documentation

Documents related to the HSC are available under the following titles and part numbers:

- *HSC User Guide* (AA-GMEAA-TK)
- *HSC Installation Manual* (EK-HSCMN-IN)
- *HSC70 Illustrated Parts Breakdown* (EK-HSC70-IP)
- *HSC50 Illustrated Parts Breakdown* (EK-HSC50-IP)
- *HSC50 Device Integrity Tests User Documentation* (EK-IHSC5-UG)
- *HSC50 Offline Diagnostics User Documentation* (EK-OHS-UG)
- *HSC50 Utilities User Documentation* (EK-UHSC5-UG)
- *VT320 Owners Manual* (EK-VT320-UG)
- *VT320 Programmer Pocket Guide* (EK-VT320-HR)
- *VT320 Installation Guide* (EK-VT320-IN)
- *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)
- *Installing and Using the LA75 Printer* (EK-0LA75-UG)
- *LA75 Printer Programmer Reference Manual* (EK-0LA75-RM)
- *Star Coupler User Guide* (EK-SC008-UG)
- *CI780 User Guide* (EK-CI780-UG)
- *DECwriter Correspondent Technical Manual* (EK-CPL12-TM)
- *TU58 DECtape II User Guide* (EK-0TU58-UG)

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

The *HSC 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.

1 General Information

1.1 Introduction

This chapter includes general information about the Hierarchical Storage Controllers (HSC) mass storage server, including:

- Cabinet layout
- Software overview
- Subsystem block diagram
- Module descriptions
- Maintenance features
- Specifications

NOTE
In this manual "HSC" refers to the HSC70 and HSC40 models. "HSC50" refers to the HSC50 and HSC50 (modified) models. Individual model names are used only when the information is model-specific.

Table 1–1 shows the major differences between the various HSC models. Note that the HSC70 supports a combination of eight disk and tape data channels, the HSC50 supports a combination of six disk and tape data channels, and the HSC40 supports a combination of three 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.

Table 1–1 Differences Between HSC Models

HSC Contents	HSC70	HSC50 (Modified)	HSC50	HSC40
I/O control processor	L0111	L0105	L0105	L0111-YA
Memory	L0117	L0105	10105	L0117
Number of data channels (disk + tape)	8	6	6	3
Load devices	RX33	TU58	TU58	RX33
Power controller	30–24374	30–24374	70–19122	30–24374
Auxiliary power supply	Yes	Yes	No	No

Kit number HSC7X–AA/AB is available to upgrade the HSC40 to an HSC70.

The HSC controller subsystem can interface with multiple hosts using the computer interconnect (CI) bus. One CI bus is included with the subsystem. In case of bus failure, each CI bus consists of two paths (path A and path B). See Figure 1–1 for a sample five-node cluster configuration with two HSCs and three host computers. In this figure, all three hosts access both HSCs over the CI bus. Through dual-porting, both HSCs can access the tape formatter and the disks.

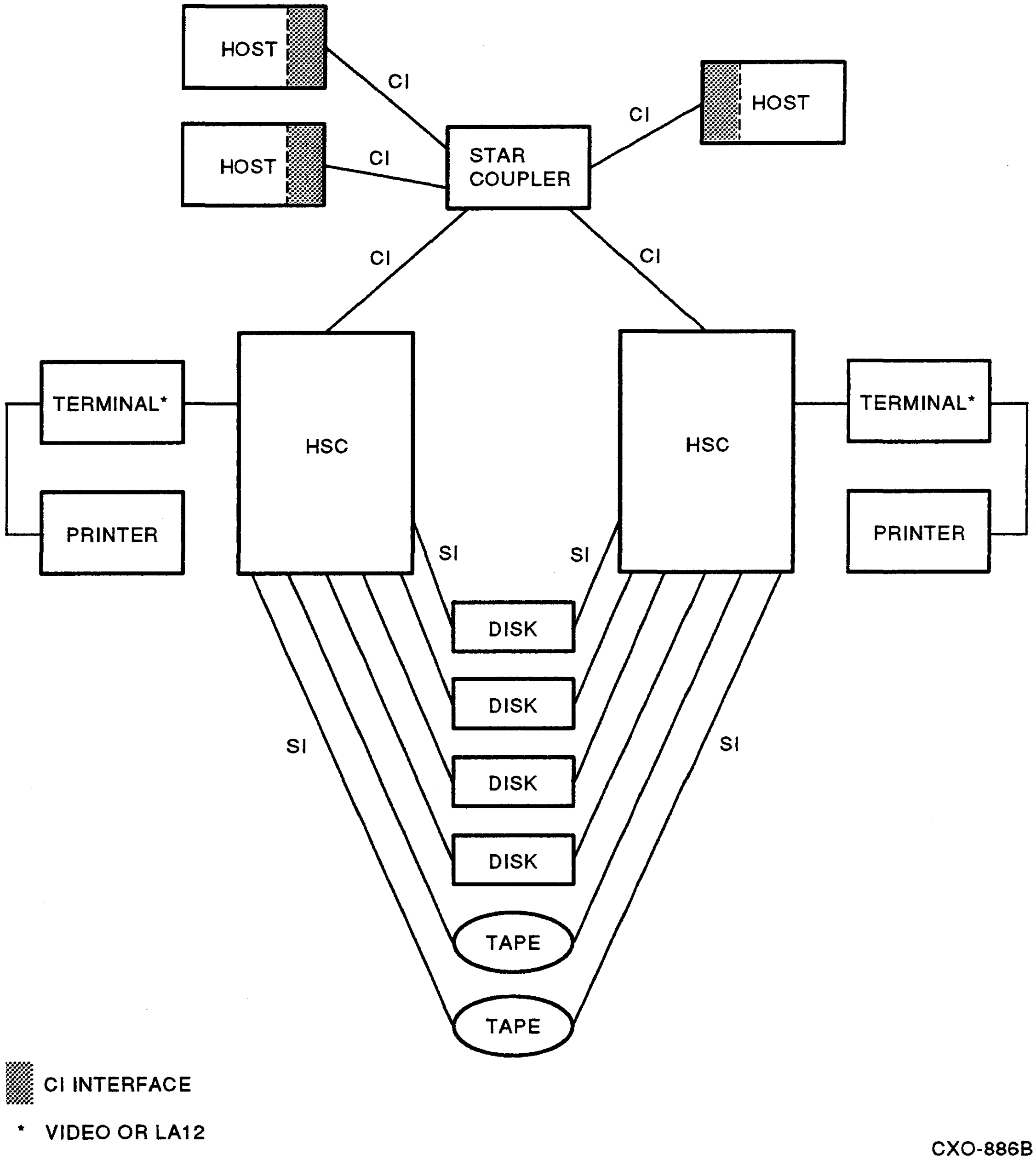


Figure 1–1 Redundant Cluster Configuration

1.2 HSC Cabinet Layout

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

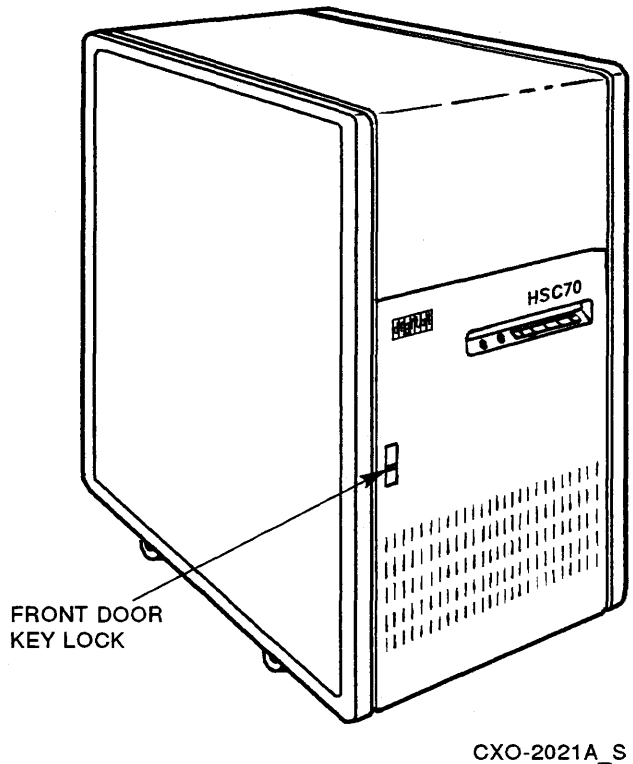


Figure 1–2 HSC Cabinet Front View

The front of the HSC cabinet contains the operator control panel (OCP) 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). Figure 1–3 shows the HSC cabinet with the front door open.

The upper right-hand portion of the cabinet houses two RX33 dual drives and connectors for the OCP.

The HSC70 contains two power supplies. The HSC40 contains one power supply. The power supplies are housed under the RX33 drives. 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.

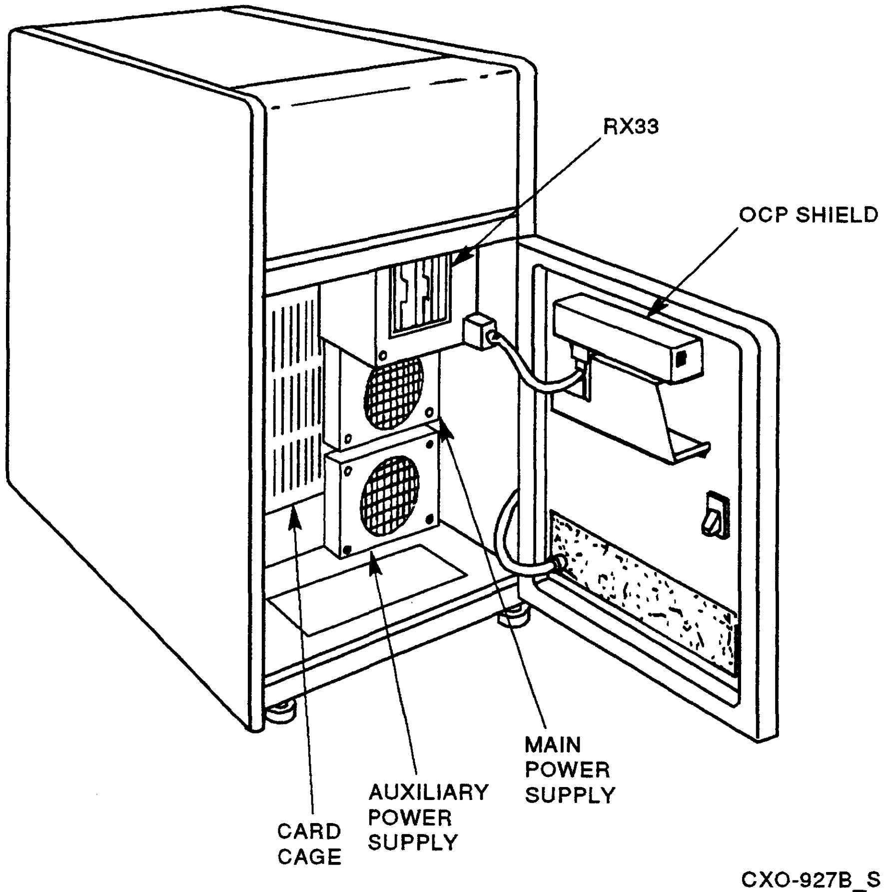
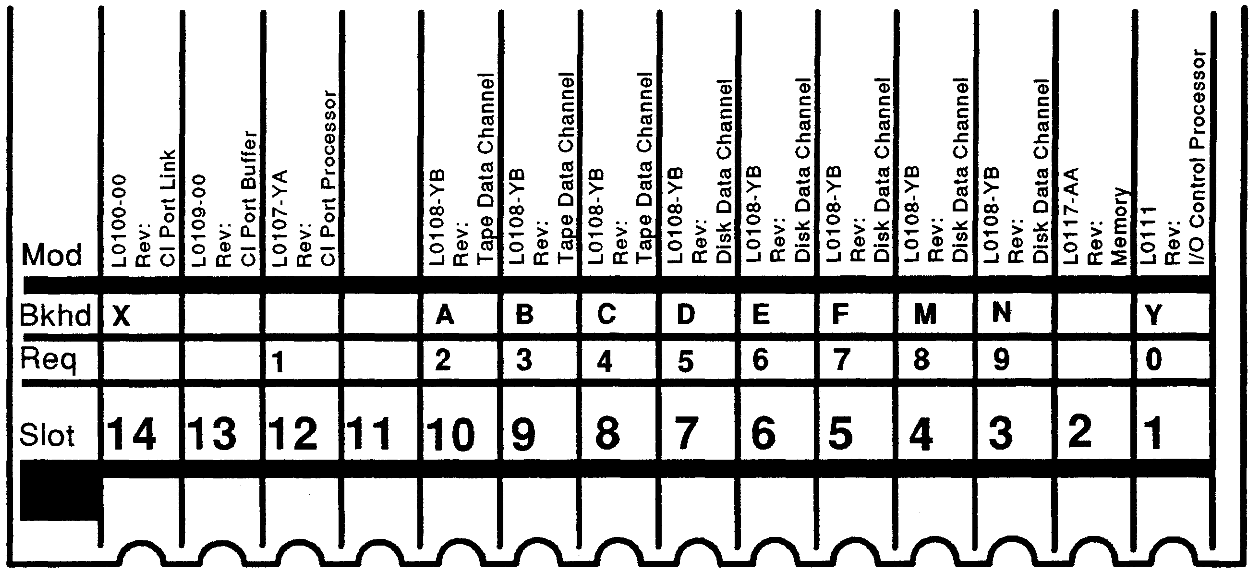


Figure 1–3 HSC Cabinet Inside Front View

A 14-slot card cage with a corresponding backplane provides housing for the L-series extended hex HSC logic modules. The card cage occupies the upper left corner of the cabinet. Above the card cage is a module utilization label indicating the slot location of each module. Figure 1–4 shows a typical HSC module utilization label. All unassigned slots in the backplane contain baffles.



CXO-889A

Figure 1–4 HSC Module Utilization Label Example

NOTE
Requester 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.

Open the cabinet rear door with a 5/32-inch hex key. A rear view of the cabinet with the back door open is shown in Figure 1–5. The backplane logic modules are cooled by a blower mounted behind the card cage. Air is drawn in through the front door louver, up through the modules, and exhausted through the larger duct at the rear.

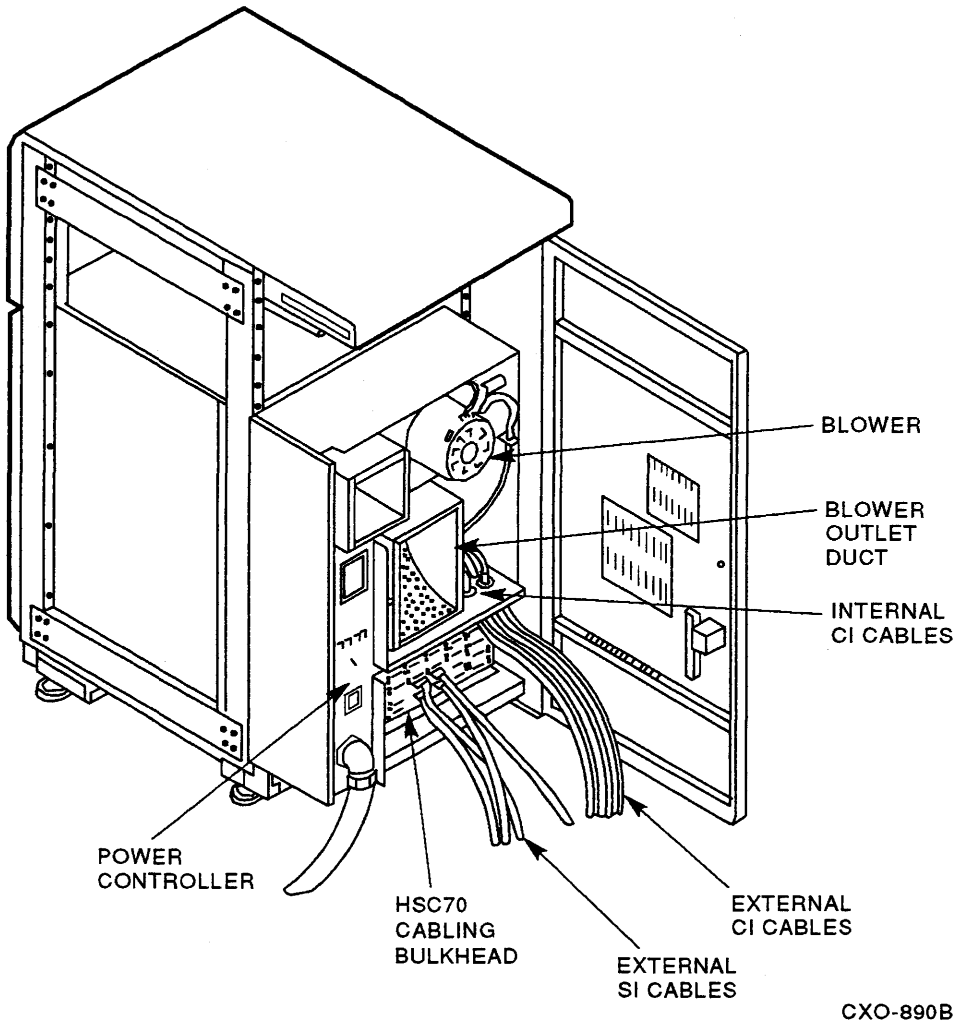


Figure 1–5 HSC Cabinet Inside Rear View

NOTE
Figure 1–5 shows the blower motor outlet duct for current models. Earlier models have a smaller blower motor outlet duct.

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

The power controller is located in the lower left-hand rear corner of the HSC. 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 HSC 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.

1.3 HSC50 Cabinet Layout

HSC50 logic and power systems are housed in a modified H9642 cross-products cabinet with both front and rear access. Figure 1–6 shows the front view of the cabinet.

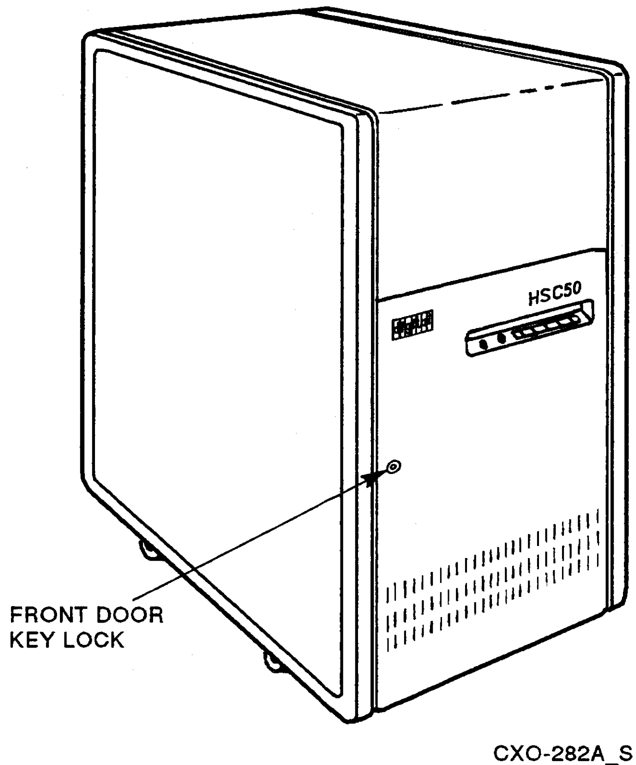


Figure 1–6 HSC50 Cabinet Front View

On the front of the cabinet are the OCP 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 (part number 12-14664). Figure 1–7 shows the inside front view of the HSC50. The back of the front door contains two TU58 drives and slots for tape storage.

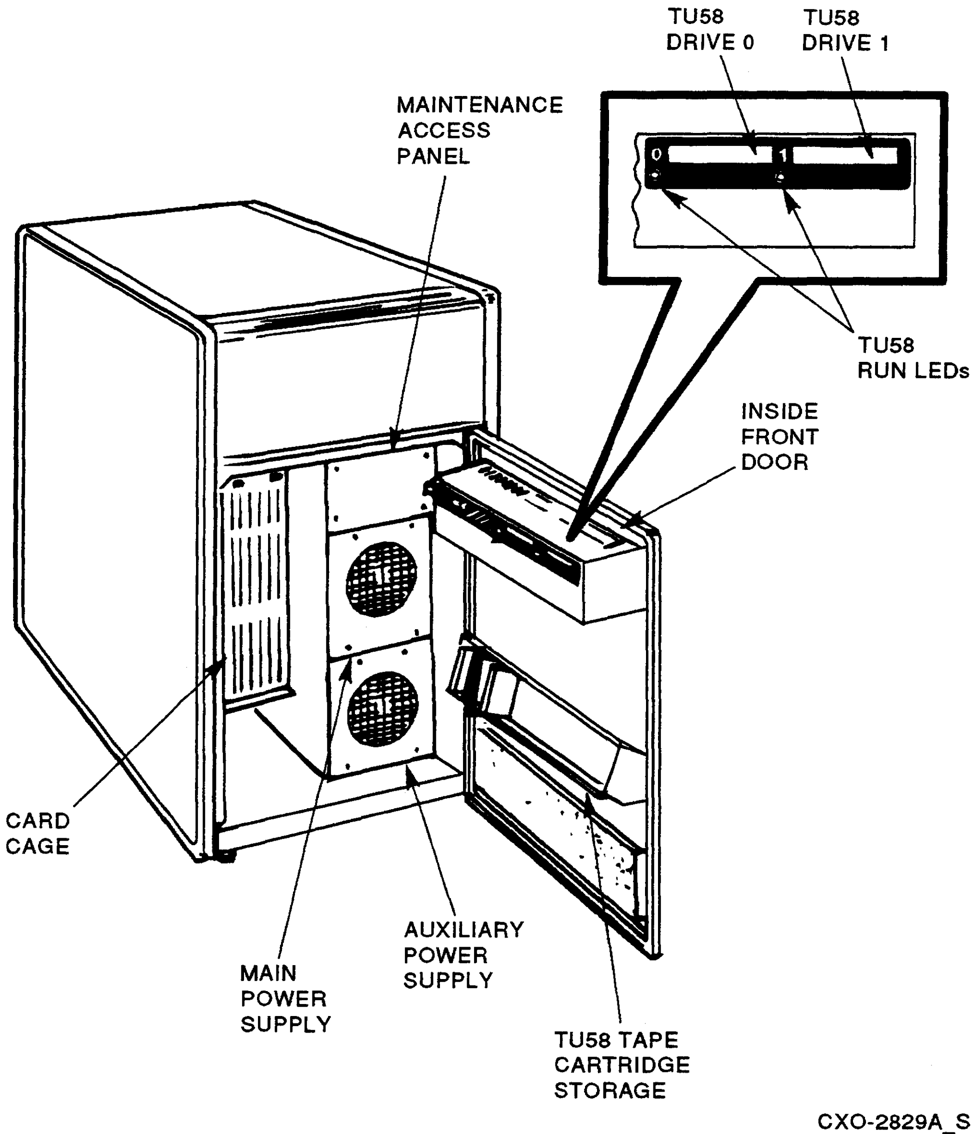


Figure 1–7 HSC50 Cabinet Inside Front View

A 14-slot card cage with a corresponding backplane provides housing for the HSC50 L-series extended hex logic modules. The card cage and backplane occupy the lower left corner of the cabinet. Above the card cage is a module utilization label indicating the slot location of each module (Figure 1–8). All unassigned slots contain baffles.

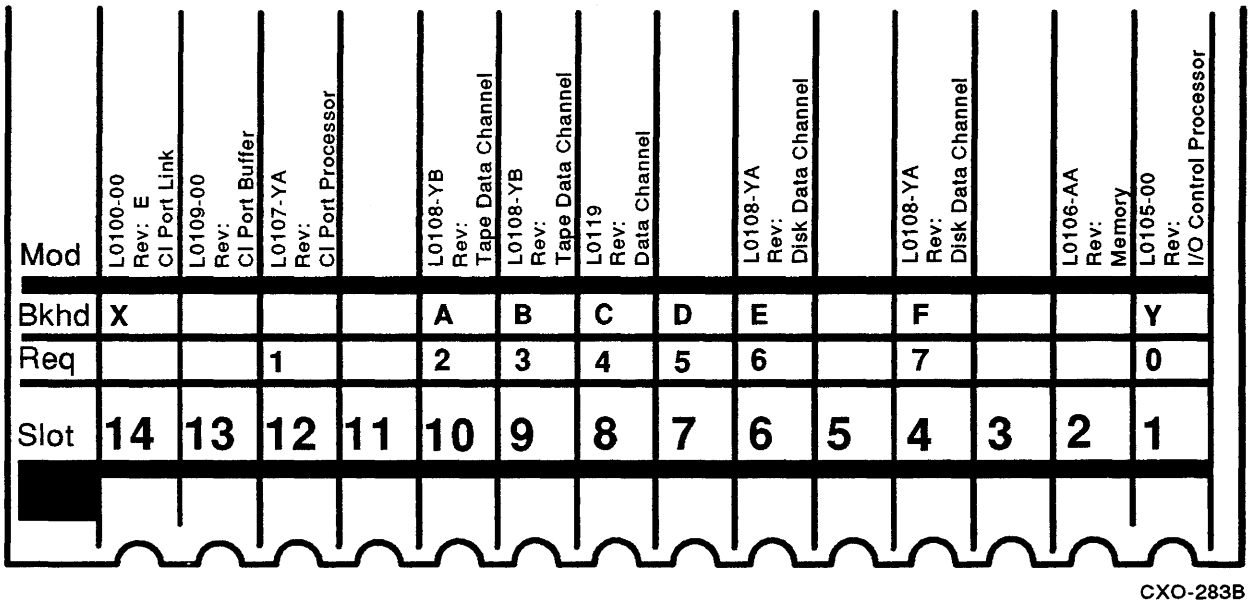


Figure 1–8 HSC50 Module Utilization Label Example

NOTE

Requester slots A through F, as shown in Figure 1–8, 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.

The upper right-hand portion of the cabinet houses the maintenance access panel. A dc power on/off switch and connectors for the TU58, the OCP, and the maintenance terminal port are located on this panel.

Power supply units are housed under the maintenance panel. A basic HSC50-AA/AB contains one power supply capable of providing power for three data channels. A fourth data channel requires the addition of an auxiliary power supply. 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 1–9 shows the back of the HSC50 cabinet. The rear door is opened with a 5/32-inch hex key.

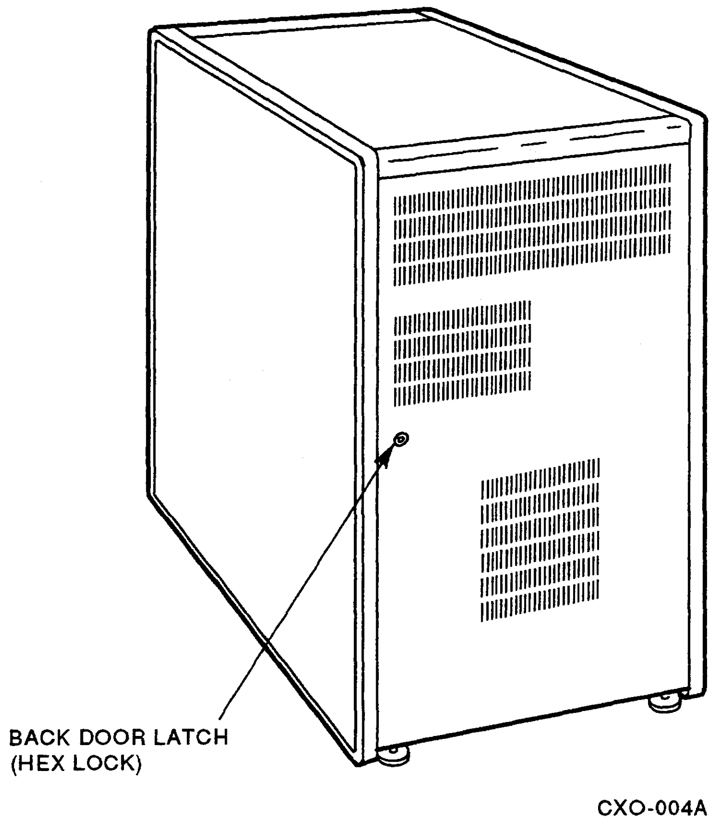


Figure 1–9 HSC50 Cabinet Rear View

Logic modules are cooled by a blower mounted behind the card cage as shown in Figure 1–10. Air is drawn in through the front door louver, up through the modules, and exhausted through the middle duct at the rear.

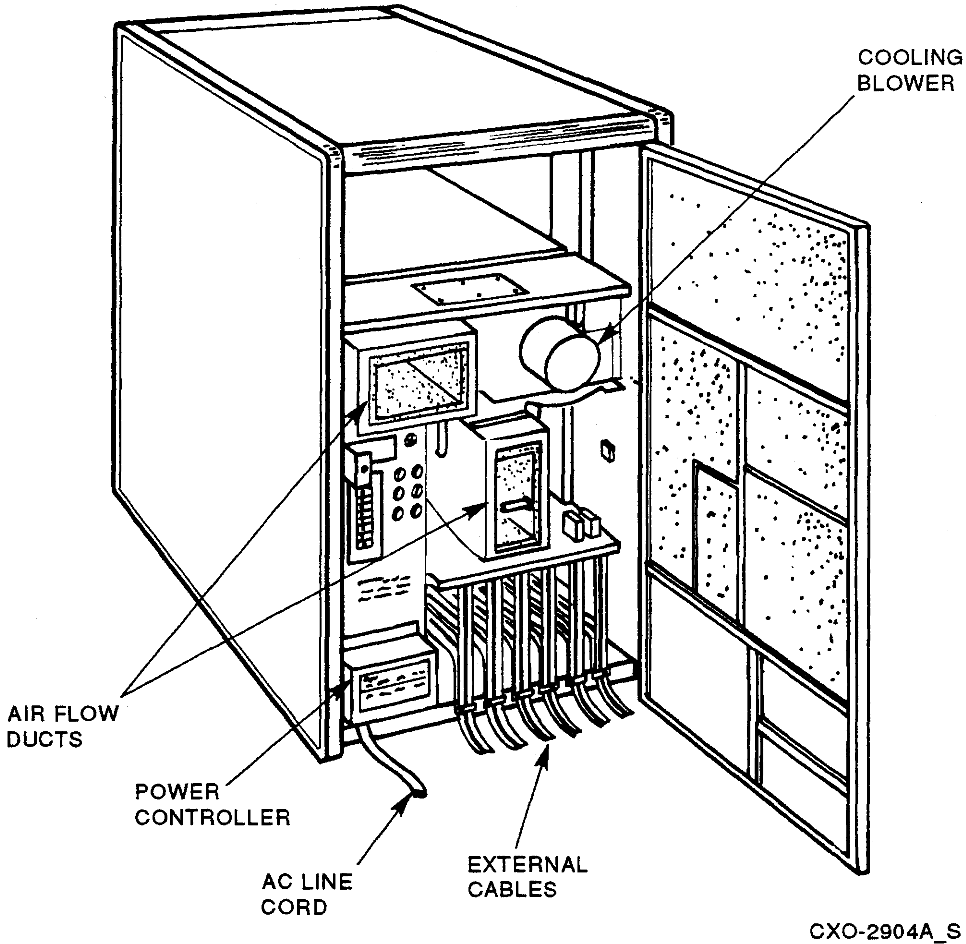


Figure 1–10 HSC50 Cabinet Inside Rear View

Two levels of cable connections are found in the HSC50: backplane to bulkhead and bulkhead to outside the cabinet. All connections to the logic modules are made through the backplane. All cables attach with press-on connectors to the backplane.

The power controller is in the lower left-hand rear corner of the HSC50. Also at the rear of the HSC50, the power control bus and delayed output line are connected to noise isolation filters.

Exterior CI, SDI, and STI buses are shielded up to the HSC50 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 and are attached with press-on connectors.

1.4 External Interfaces

Figure 1–11 shows the external hardware interface lines used by the HSC, and Figure 1–12 shows the external hardware interface lines used by the HSC50.

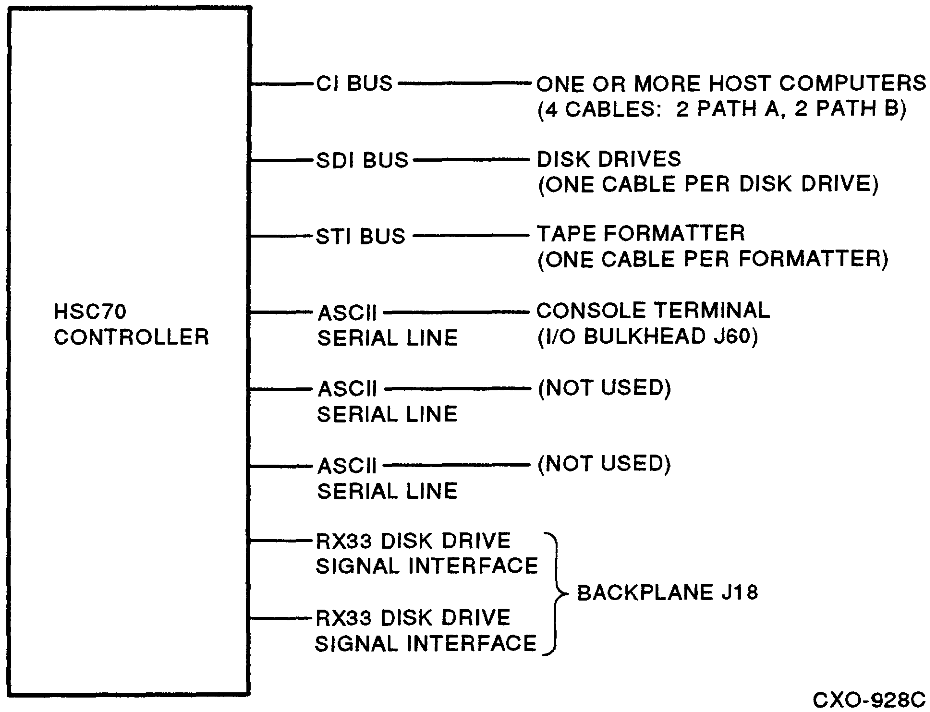


Figure 1–11 HSC External Interfaces

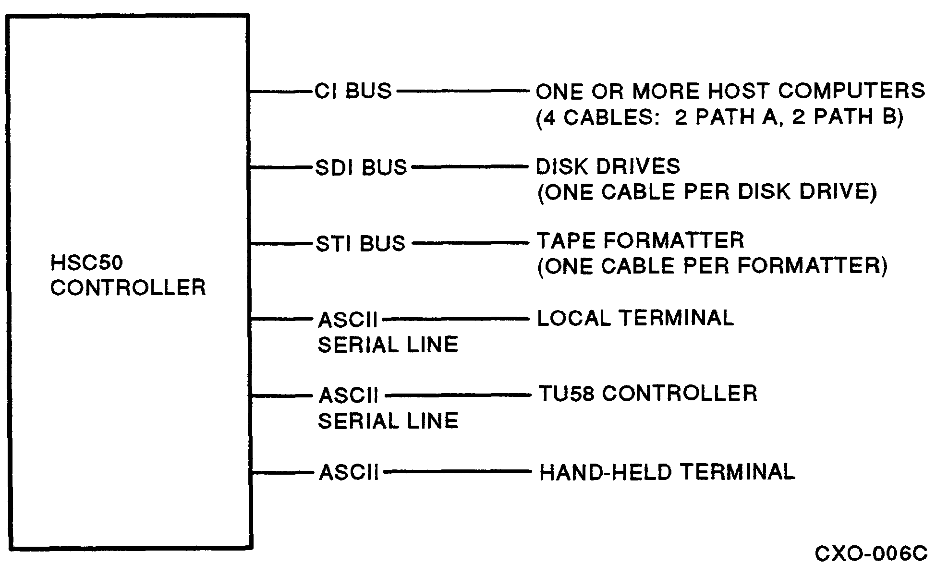


Figure 1–12 HSC50 External Interfaces

The external hardware interface lines perform the following functions:

Line	Function
CI bus	Four coaxial cables (BNCIA-xx): two-path (path A and path B) serial bus with a transmit and receive cable in each path. This is the communication path between system host(s) and the HSC.
SDI bus	Four shielded wires for serial communication between the HSC and the disk drives (one SDI cable per drive per controller) (BC26V-xx).
STI bus	Four shielded wires for serial communication between the HSC and the tape formatter (one STI cable per formatter) (BC26V-xx).
ASCII serial line	RS-232-C cable for local console terminal communication with the I/O control processor module.
ASCII serial line	RS-232-C cable in the HSC50 to link the TU58 controller to the cabinet.
RX33 disk drive signal interface	Cable linking RX33 drives with the RX33 controller on the M.std2 module of the HSC.
ANSII hand held terminal	RS-232-C cable for hand held terminal communications with the I/O Control Processor module of the HSC50.

1.5 HSC Hardware Overview

The HSC is a multimicroprocessor subsystem with two shared memory structures: one for control and one for data. In addition, the HSC I/O control processor fetches its own instructions from a private (Program) memory. Figure 1–13 shows a subsystem block diagram. Each major block is a module unless otherwise specified.

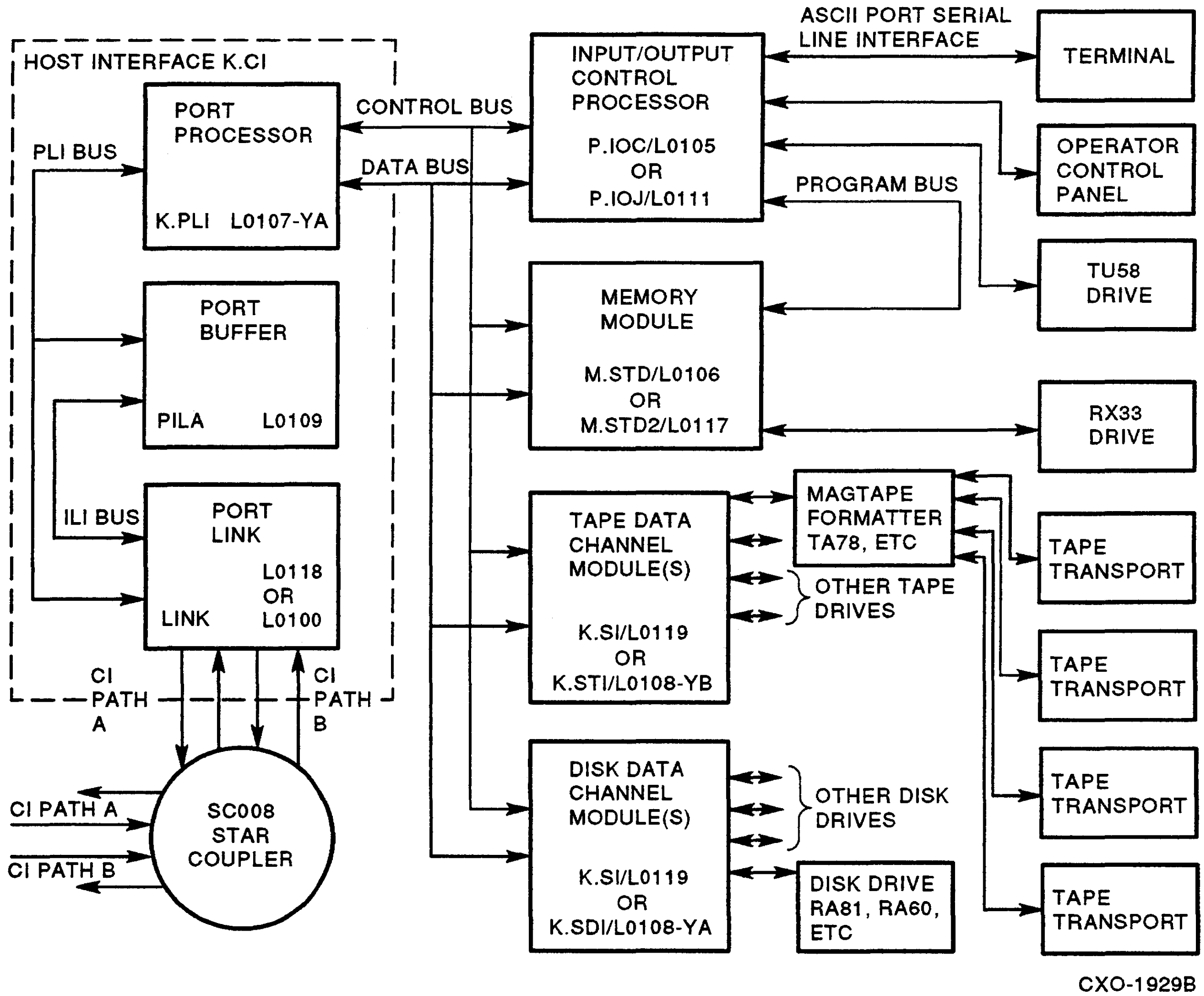


Figure 1–13 HSC Subsystem Block Diagram

References to logic modules by their engineering terms appear throughout HSC documentation, as well as on diagnostic printouts. Refer to Table 1–2 for a cross-reference of HSC module names.

Table 1–2 HSC Module Names

Module Name	Engineering Name	Module Designation
Port link	LINK or Interprocessor LINK Interface (ILI)	L0100, Rev E2 or L0118
Port buffer	PILA	L0109
Port processor	K.pli	L0107

Table 1–2 (Cont.) HSC Module Names

Module Name	Engineering Name	Module Designation
Disk data channel or Data channel	K.sdi or K.si	L0108-YA or L0119-YA
Tape data channel or Data channel	K.sti or K.si	L0108-YB or L0119-YA
Input/output control processor	P.ioj P.ioj P.ioc	L0111 (HSC70) L0111-YA (HSC40) L0105 (HSC50)
Memory	M.std2 M.std	L0117 (HSC) L0106 (HSC50)
Host interface	K.ci	Consists of: Port link (LINK or ILI), Port buffer (PILA), and Port processor (K.pli) modules

1.5.1 Port Link Module (LINK) Functions

The port link module (L0100-E2 or L0118) is a part of the host interface module set (K.ci). With all configuration switches and jumpers in default positions, the L0118 is functionally identical with the L0100-E2. The location and default positions are described in Chapter 3. The port link module 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. Also generates the 32-bit CRC character during the transmission of a packet. An incorrect CRC means either errors were induced by noise or a packet collision occurred.
- **ACK/NACK 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 NACK 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 or the node address is incorrect.
- **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 through line drivers/receivers directly to the CI coaxial cables. On the HSC interior side, the port link module interfaces to the port buffer module through a set of interconnect link 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.

1.5.2 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 HSC internal memory buses. It also interfaces to the port link (CI link) module through the ILI signals and the port processor module through port/link interface (PLI) signals.

1.5.3 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 HSCcontrolmemory and notifies the correct server process of incoming messages
- Moves data packets to/from HSC data memory

The port processor module interfaces to three buses:

- PLI bus interfaces the port buffer and port link modules
- Control memory bus interfaces HSCcontrolmemory
- Data memory bus interfaces HSC data memory

1.5.4 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 HSC data memory and the disk drives

- Checks the error detection code (EDC) and generates or checks the error correction code (ECC) during read/write operations.

Commands and responses pass between the disk data channel microprocessor and other internal HSC 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 or tape data channel modules are possible in the HSC70. The HSC50 supports combinations of up to six disk data channel or tape data channel modules and the HSC40 supports combinations of up to three disk data channel or tape data channel modules. Configuration guidelines are found in the *HSC Installation Manual* (EK-HSCMN-IN).

1.5.5 Tape Data Channel Module (K.sti) 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 an error detection code (EDC) for each 512 bytes during a write operation. The tape formatter generates and sends an EDC every 512 bytes during a read operation.

Commands and responses pass between the tape data channel microprocessor and other internal HSC processes through control memory. The tape data channel module interfaces to thecontrolmemory bus and to the data memory bus.

1.5.6 Data Channel Module (K.si) Functions

Data channel module (L0119-YA) is an interface between the HSC and the standard disk interface (SDI) or standard tape interface (STI) bus and is a direct replacement for the K.sdi or K.sti data channel modules. The K.si is configured for disk or tape interface when the HSC is initialized (see Chapter 4). The K.si functions are the same as the functions for the K.sdi or K.sti.

1.5.7 I/O Control Processor Module (P.ioj/c) Functions

The HSC70 P.ioj module (L0111) and the HSC40 P.ioj module (L0111-YA) use a PDP-11 ISP (J-11) processor. The HSC50 and HSC50 (modified) both use the P.ioc module (L0105), with a PDP-11 ISP (F-11) processor. Both contain memory management and memory interfacing logic. These processors execute their respective HSC internal software. The input/output (I/O) control processor modules also contain the following functional blocks:

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

The I/O control processor modules interface 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
- Console terminal RS-423 compatible signal levels (HSC only)
- TU58 tape drives (HSC50 only)
- Auxiliary terminal through an RS-232-C interface (HSC50 only)

1.5.8 Memory Module (M.std2) Functions

The HSC memory module (L0117) contains three separate and independent system memories, each residing on a different bus within the HSC. In addition, the memory module contains the RX33 diskette controller. The three memory systems and RX33 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.

  CAUTION
  The switch pack on the M.std2 module is factory set to calibrate the RX33 diskette controller. Do not change the setting of this switch pack; the switch settings are unique to each module and cannot be restored outside of the manufacturing environment.

- **RX33 diskette controller (K.rx)**—Resides on the Program bus and performs direct memory access word transfers when reading or writing data to/from the RX33 diskette.

Using physical addresses, the memory space allocations for the three memories are illustrated in Figure 1–14.

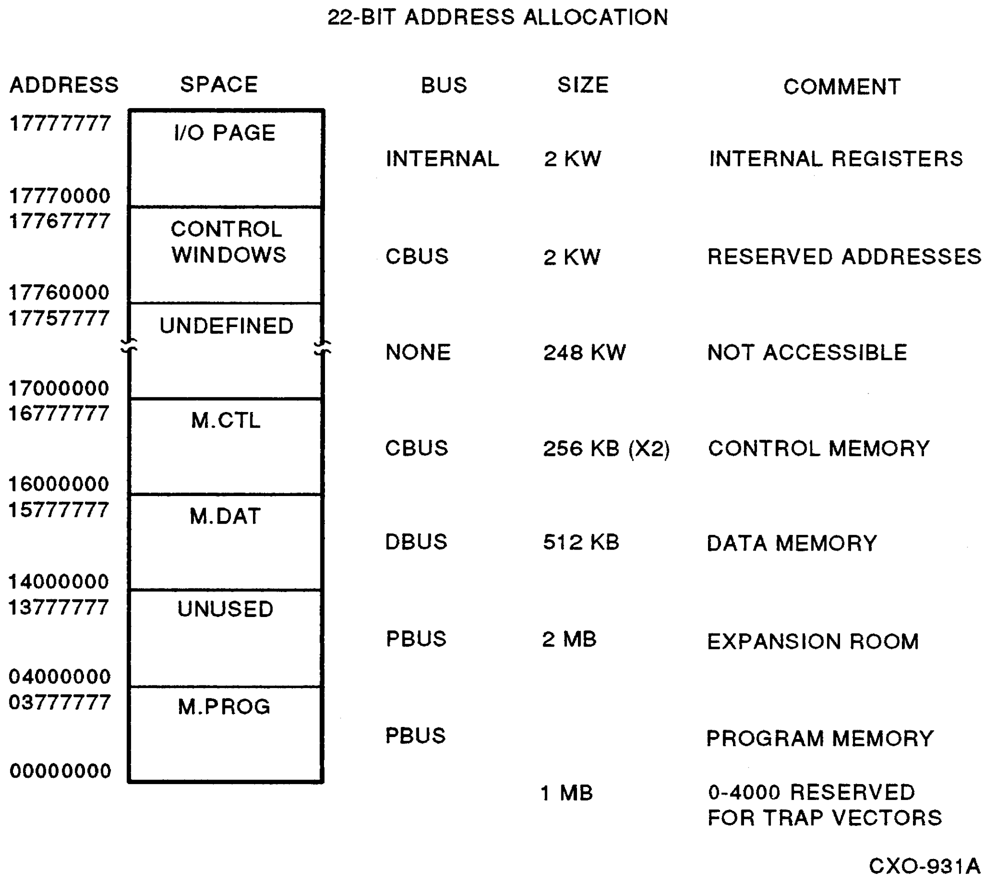


Figure 1–14 Memory Map (M.std2—L0117)

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.

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

1.5.9 Memory Module (M.std) Functions

The memory module (L0106) used in the HSC50 contains the following three independent and separate memories:

- 256 Kbytes of Program memory (M.prog)–This is space for the control program loaded from the TU58.
- 128 Kbytes of control memory (M.ctl)–This is space for the routines initiating data transfer action.
- 128 Kbytes of data memory (M.dat)–This is space to hold the data from/to a data channel module.

Using physical addresses, the memory space allocations for the three memories are illustrated in Figure 1–15.

22-BIT ADDRESS ALLOCATION

ADDRESS	SPACE	BUS	SIZE	COMMENT
17777777–17770000	I/O PAGE	INTERNAL	2 KW	INTERNAL REGISTERS
17767777–17760000	CONTROL WINDOWS	CBUS	2 KW	RESERVED ADDRESSES
17757777–17000000	UNDEFINED	NONE	248 KW	NOT ACCESSIBLE
16777777–16400000	UNUSED	CBUS	64 KW	EXPANSION ROOM
16377777–16000000	M.CTL	CBUS	64 KW	CONTROL MEMORY
15777777–14400000	UNUSED	DBUS	192 KW	EXPANSION ROOM
14377777–14000000	M.DAT	DBUS	64 KW	DATA MEMORY
13777777–01000000	UNUSED	PBUS	1.5 MW	EXPANSION ROOM
00777777–00000000	M.PROG	PBUS	128 KW	PROGRAM MEMORY 0-4000 RESERVED FOR TRAP VECTORS

CXO-338B

Figure 1–15 Memory Map (M.std—L0106)

The interface to the control memory is through the backplane control bus, and to the data memory through the backplane data bus. The interface to the I/O control processor local Program memory is through a set of backplane signals to the control memory module.

1.6 HSC Software Overview

The HSC subsystem uses internal software to perform various tasks and to interface with an operator through a dedicated terminal. The *HSC Software Release Notes* for your version of software describes the unique features of the software. These software release notes are shipped with each HSC and with updates of the software. The major HSC software modules are shown at a block level in Figure 1–16.

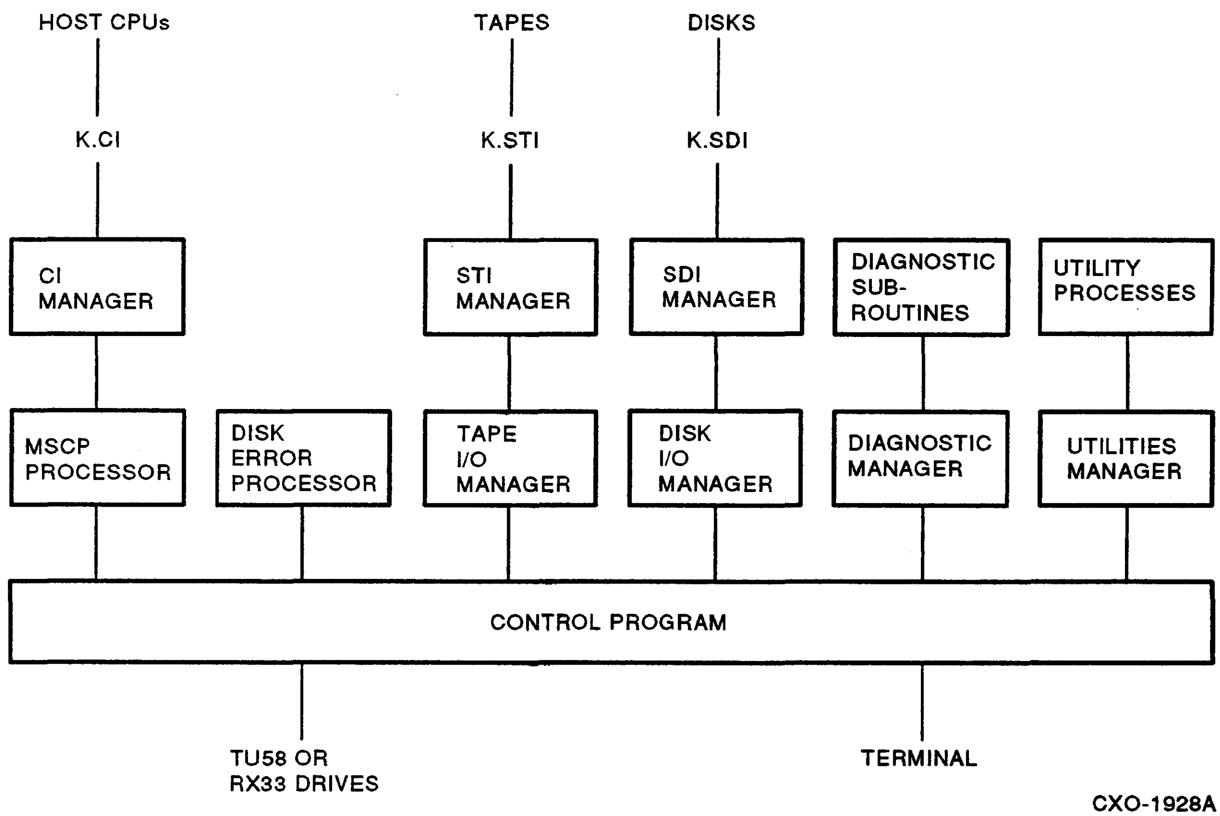


Figure 1–16 HSC Internal Software

The **HSC control program** is found on the system diskette for the HSC and the system tape for the HSC50. This module is the lowest level manager of the subsystem, provides a set of subroutines and services shared by all HSC processes. The HSC control program performs the following functions:

- Interprets incoming utility requests
- Sets up the appropriate subsystem environment for operation of the requested utility
- Invokes the utility process
- Returns the subsystem to its normal environment upon completion of the utility execution
- Initializes and reinitializes the subsystem
- Executes all auxiliary terminal I/O
- Schedules processes (both functional and diagnostic) for execution by the P.ioc/j
- Provides a set of system services and system subroutines to HSC processes
- Manages the RX33 local storage media (HSC only)
- Manages the TU58 local storage media (HSC50 only)

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

The **MSCP class server** validates, interprets, and routes incoming MSCP commands and dispatches MSCP completion acknowledgments. The following are part of the MSCP class server:

- The **SDI manager** handles the SDI protocol, responds to attention conditions, and manages the on-line/off-line status of the disk drives.
- The **disk I/O manager** translates logical disk addresses into drive-specific physical addresses, organizes the data-transfer structures for disk operations, and manages the physical positioning of the disk heads.

The **CI manager** handles virtual circuit and server connection activities.

The **disk error processor** responds to all detected error conditions. It reports errors to the diagnostic manager and attempts to recover from errors, such as ECC, bad block replacement, and retries. 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.

The **TMSCP class server** sets up the data transfer structures for tape operations and manages the physical positioning of the tape. The **STI manager** is the part of the TMSCP class server that handles the STI protocol, responds to attention conditions, and manages the on-line/off-line status of the tape drives.

The **diagnostic manager** handles all diagnostic requests, error reporting, and error logging. It also provides decision-making and diagnostic-sequencing functions, and can access a large set of resource-specific diagnostic subroutines.

The **diagnostic subroutines** run under the control of the diagnostic manager and are classified as device integrity tests.

The **utility processes** perform volume-management functions such as formatting, disk-to-disk copy, disk-to-tape copy, tape-to-disk restore. They also handle miscellaneous operations required for modifying subsystem parameters, such as COPY, PATCH, and error dump, and are used in analyzing subsystem problems.

1.7 HSC Maintenance Strategy

Maintenance of the HSC is accomplished with field replaceable units (FRUs). Procedures for removal and replacement are described in Chapter 3. Do 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 HSC-resident off-line diagnostics are available to support and verify corrective maintenance decisions.

1.7.1 Maintenance Features

The following features assist in troubleshooting the HSC:

- Self-contained and self-initiated diagnostics—On initialization, various levels of diagnostics execute in the HSC. Read-only memory (ROM) diagnostics test each microprocessor in the disk and tape data channels, port processor, and I/O processor modules. Pressing the HSC Init button starts all internal ROM diagnostics.
- Operator control panel fault code display—The OCP or the console terminal displays any failures. If further diagnostics are needed, use the terminal to initiate diagnostics stored on the system boot media or the off-line diagnostic media (RX33 diskettes for the HSC or TU58 tapes for the HSC50).

- Console terminal—After initialization, the operator can use the console terminal to run on-line device integrity tests (see Chapter 5) or off-line diagnostic tests (see Chapter 6). Also, certain resource failure detections can initiate tests automatically.
- Module LED indicators—All logic modules have at least one LED to indicate board status. See Chapter 2 for the location of these LEDs.

The HSC subsystem allows logical assignment of a disk drive or tape formatter to the diagnostics. Device integrity tests allow drive diagnosis, even though other active drives are connected to the HSC.

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

Requestor-detected data memory errors cause an initiation of the in-line 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. If the same buffer is sent to test twice, it is retired from service, even though no failure is found.

1.8 Specifications

Figure 1–17 lists the HSC physical and environmental specifications.

DESCRIPTION	OPTION DESIGNATION
HSC MASS STORAGE SERVER	HSCXX-AA = 60 HZ, 120/180 V
	HSCXX-AB = 50 HZ, 380/415 V

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 AND TOLERANCE	PHASE	STEADY-STATE CURRENT (RMS)	POWER CONSUMPTION (MAX)
120/208	104-128/180-222	60 HZ $\pm$ 1	3	17	2250 WATTS
380/415	331-443	50 HZ $\pm$ 1	3	9	2245 WATTS

POWER (AC)

AMP (MAX) BY PHASE			
120 V	PHASE A = 1	380 V	PHASE A = 1
	PHASE B = 12		PHASE B = 7
	PHASE C = 12		PHASE C = 7
	NEUTRAL = 17		NEUTRAL = 9

POWER (AC)

PLUG TYPE	POWER CORD LENGTH	INTERRUPT TOLERANCE	APPARENT POWER (KVA)
NEMA - L21 - 30P	15 FT (4.5 M)	4 MS (MIN)	3.4 (KVA)
HUBBELL - 520 P6	15 FT (4.5 M)	4 MS (MIN)	3.4 (KVA)

POWER (AC)

HSC OPTION	INRUSH CURRENT	SURGE DURATION
HSCXX - AA	70 AMPS/PHASE	16 MS
HSCXX - AB	70 AMPS/PHASE	20 MS

DEVICE ENVIRONMENT

TEMPERATURE		RELATIVE HUMIDITY		RATE OF CHANGE		HEAT DISSIPATION	
OPERATING*	STORAGE	OPERATING	STORAGE	TEMP	HUMIDITY	60 HZ	50 HZ
59 - 90° F	.40 - 151° F	20 - 80%	5 - 95%	20° F/HR	20%/HR	7676 BTU/HR	8078 KJ/HR
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	30,000 FT	210	5.92	N/A
2.4 KM	9.1 KM			

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

CXO-2023A

Figure 1–17 HSC Specifications

2 Controls and Indicators

2.1 Introduction

This chapter describes the following controls and indicators located in five areas of the HSC and HSC50:

- HSC
  1. Operator control panel (OCP)
  2. Inside front door
  3. RX33 disk drives
  4. Logic modules
  5. Power controller
- HSC50
  1. Operator control panel (OCP)
  2. Inside front door (TU58 tape drives)
  3. Maintenance access panel
  4. Logic modules
  5. Power controllers (60 Hz and 50 Hz)

2.2 Operator Control Panel (OCP)

Figure 2–1 illustrates the controls and indicators on the OCP.

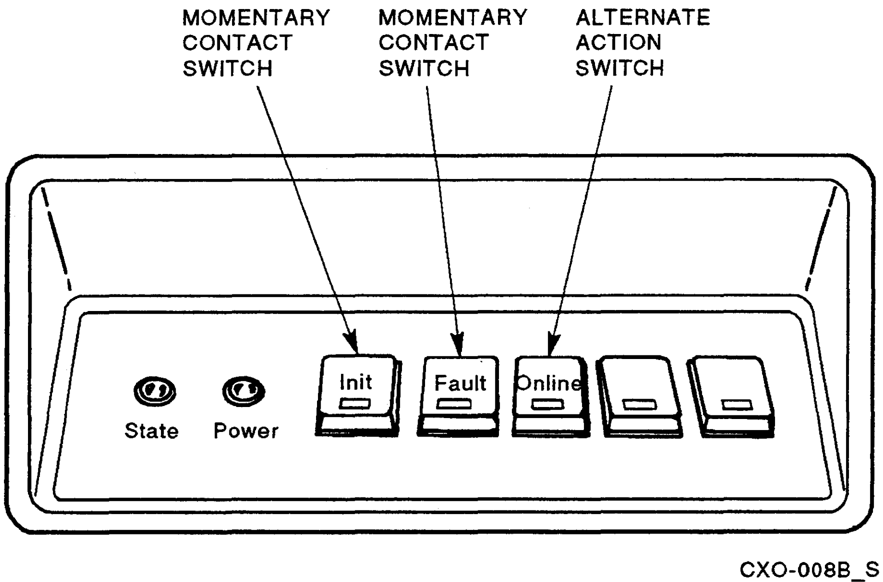


Figure 2–1 Operator Control Panel

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

- **State and Init indicators**—Describes the state of the HSC. 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. Refer to the bootstrap flowchart in Chapter 8 for details on these phases.
- **Init switch**—Pushing the Init switch causes the HSC to start its initialization routine. The Secure/Enable switch must be in the enable position for this switch to be operational. Holding the Init switch in causes the console terminal to loop back.
- **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 on the HSC70, L0111-YA on the HSC40, or L0105 on the HSC50 and HSC50 (modified)), which constantly monitors the +5, +12, and –5.2 voltages. The power indicator also is 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
  An on power indicator does *not* mean these voltages are within specification (±5 percent).

- **Fault indicator and switch**—Comes on when the HSC logic detects a fault. The Fault switch is also 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 blank switches function as an error display. If the fault code is a hard fatal error, the fault code blinks on and off until the HSC is powered down or the Fault 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 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 does *not* continue operation until the failure is remedied.

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

- **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 HSC logic in the available state when pushed to the in position and allows a host to establish a virtual circuit with the HSC. 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 HSC 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**—Forms the lowest two bits of a five-bit fault code.

2.3 HSC Inside Front Controls and Indicators

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

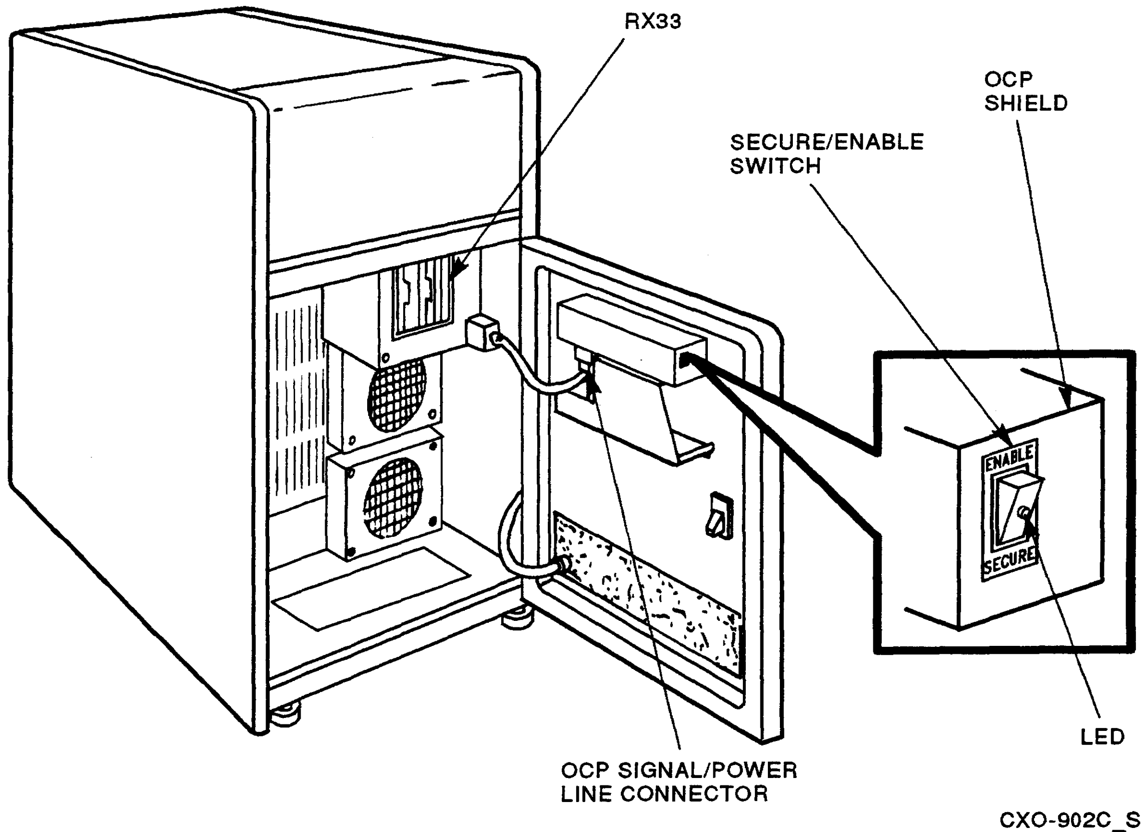


Figure 2–2 Controls/Indicators Inside Front Door

The following list describes the controls and indicators found on the HSC inside front door:

- **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 key 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.

- **RX33 LEDs**—When lit, indicates 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 (Figure 2–3).

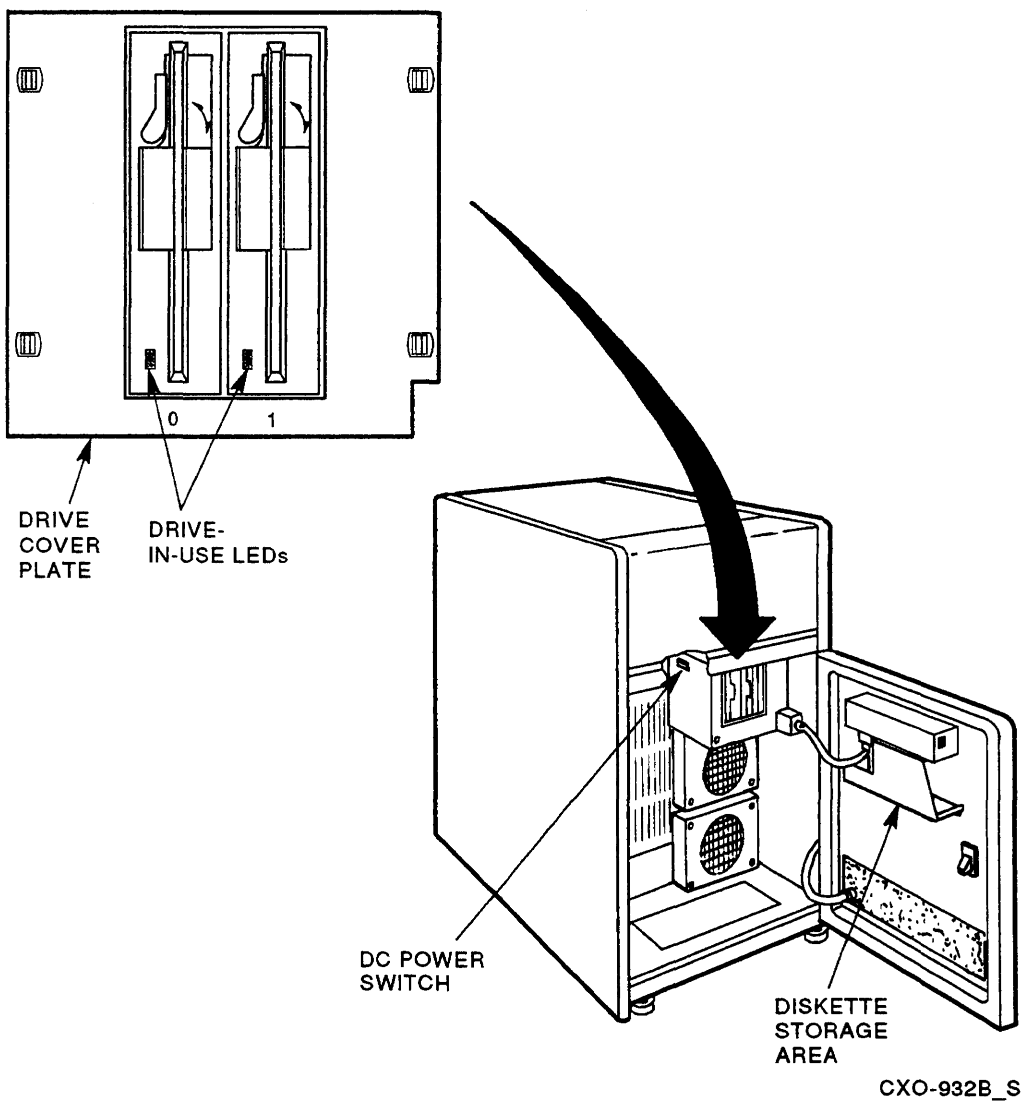


Figure 2–3 RX33 and dc Power Switch

- **dc power switch**—Located on the left side of the RX33 housing (Figure 2–3). When the dc power switch is in the 0 position, the HSC is without dc power. Moving the switch to the 1 position restores dc power.

2.4 HSC50 Inside Front Door Controls and Indicators

Figure 2–4 shows the controls and indicators on the inside of the front door.

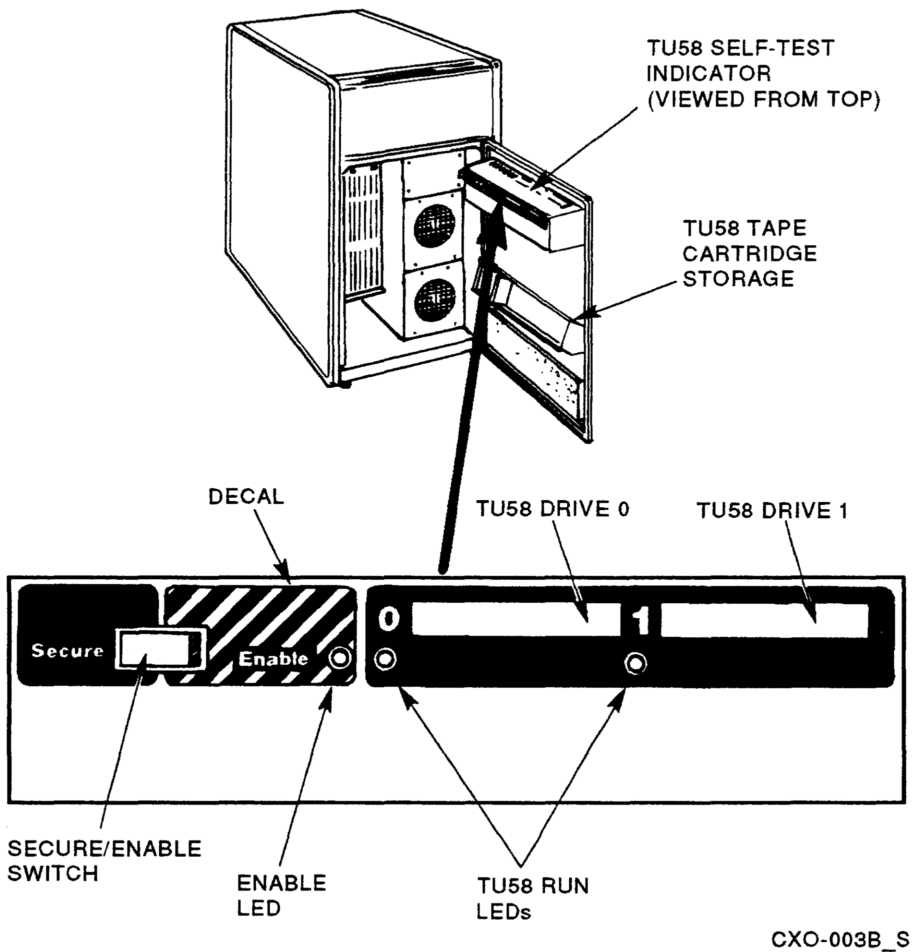


Figure 2–4 HSC50 Controls/Indicators Inside Front Door

The following list describes the controls and indicators found on the inside of the front door of the HSC50:

- **Secure/Enable switch**—With the Secure/Enable switch in the secure position, the Init switch is disabled from the OCP. Also, the SET utility program cannot run and the break key 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**—An illuminated Enable LED indicates the Secure/Enable switch is in the enable position (all switches can be used). When the Enable indicator is off, the OCP is secure.

- **TU58 Run indicators**—When a TU58 Run indicator is on, the TU58 is currently moving tape. Data loss can occur if the tape is removed while this indicator is on. If the indicator is off, tape is not in motion.
- **TU58 Self-Test indicator**—The TU58 Self-Test indicator is found on the TU58 controller module (Figure 2–4). The controller module is located inside the TU58 housing with the drive mechanics. Observe the Self-Test indicator by looking down through the TU58 housing vents. When this indicator is on, the TU58 controller has successfully completed self-diagnostics.

2.5 HSC50 Maintenance Access Panel Controls and Connectors

Removing the maintenance access panel cover reveals the dc power switch and several connectors available for HSC50 maintenance (Figure 2–5).

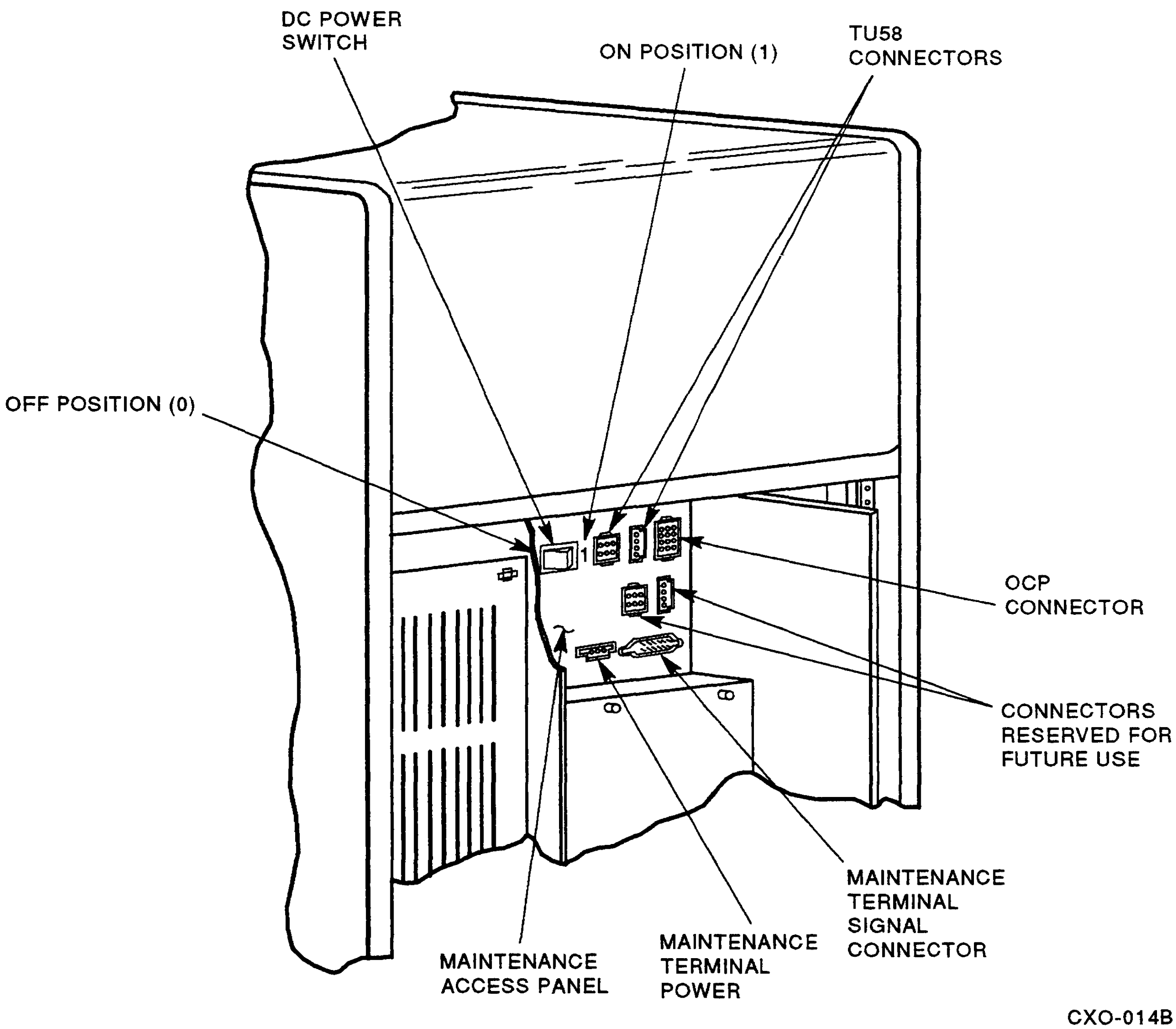


Figure 2–5 HSC50 Maintenance Access Panel

2.5.1 HSC50 dc Power Switch

When the dc power switch is in the 0 position, the HSC50 is without dc power. Moving the switch to the 1 position restores dc power.

2.5.2 HSC50 Maintenance Panel Connectors

Two of the connectors in the maintenance access panel are used to connect the maintenance terminal to the HSC50. One connector supplies power to the maintenance terminal and the other is the signal connector. Additional connectors are:

- OCP connector
- TU58 connectors
- Connectors reserved for future use

2.6 Module Indicators

All logic modules have at least one LED to indicate board status. Figure 2–6 shows the locations of these LEDs and the module utilization label. Additionally, three of these logic modules contain specific switches.

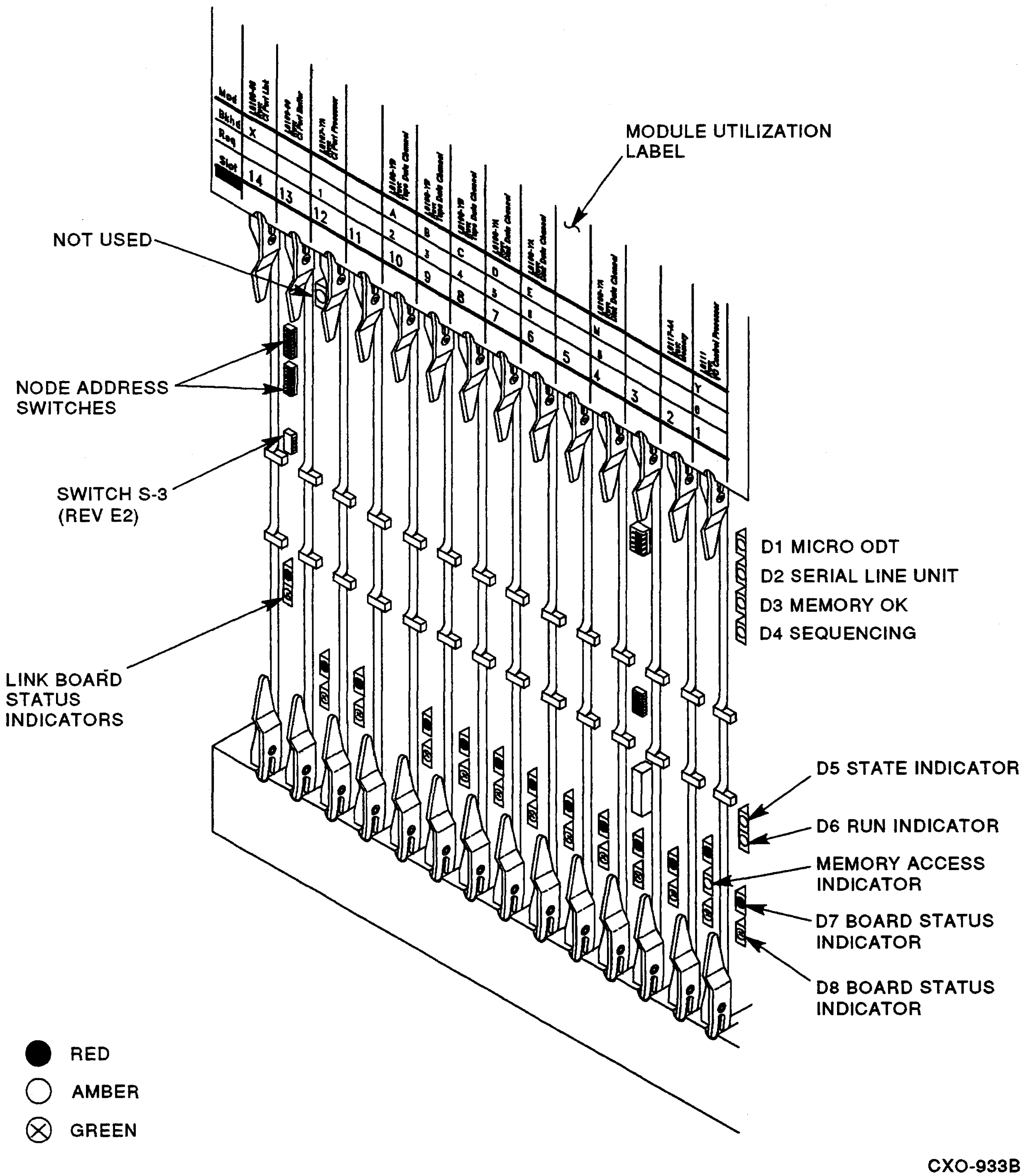


Figure 2–6 Module LED Indicators

NOTE
Figure 2–6 and Figure 2–7 shows a typical HSC module configuration. The disk and tape data channel module combinations vary as follows between the HSC models:

The HSC70 supports up to 8 disk and tape data channel module combinations.

The HSC50 supports up to 6 disk and tape data channel module combinations.

The HSC40 supports up to 3 disk and tape data channel module combinations.

Figure 2–7 shows the HSC slot location for each of the modules.

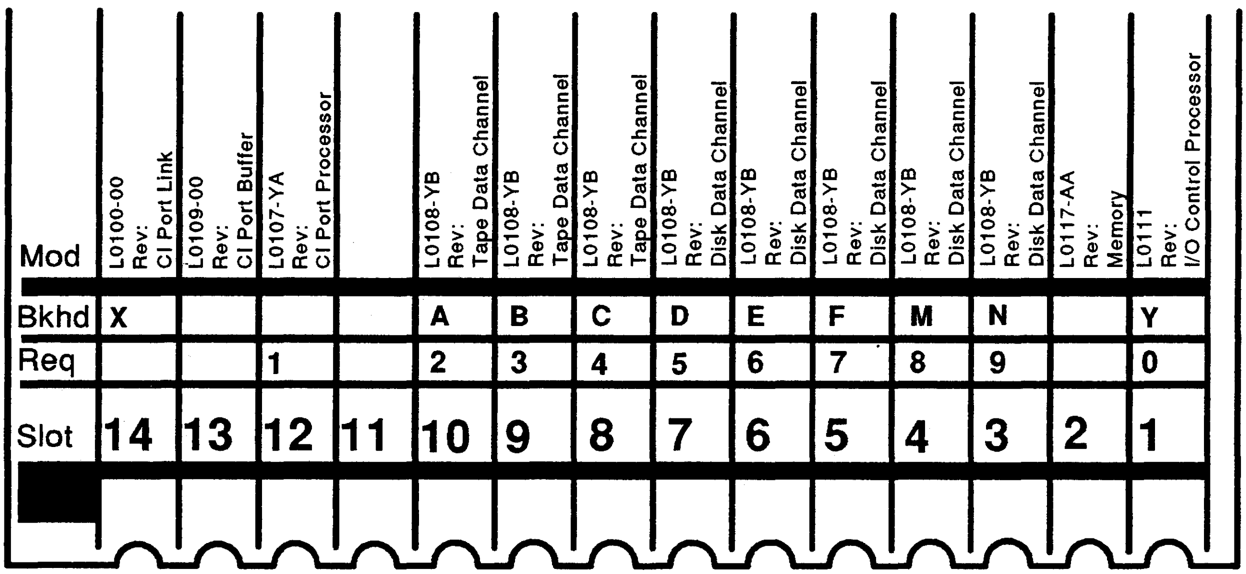


Figure 2–7 HSC Module Utilization Label Example

Table 2–1 shows the functions of the various module LEDs.

Table 2–1 Functions of Logic Module LEDs

Module	Color	Function
LINK (L0100-E2) LINK (L0118)	Green	Board status—Indicates the node is either transmitting or receiving; dims or brightens relative to the amount of local CI activity.
	Red	Board status—Indicates the module is in the internal maintenance mode.
PILA (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 when the HSC is on (used only for engineering test purposes).
K.pli (L0107-YA)	Green	Board status—Indicates the operating software is running and that self-test module microdiagnostics have completed successfully.

Table 2–1 (Cont.) Functions of Logic Module LEDs

Module	Color	Function
	Red	Board status—Indicates an inoperable module, except during initialization, when it comes on during module testing.
K.sdi (L0108-YA) K.sti (L0108-YB) K.si (L0119-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.
K.si only	Amber (eight LEDs)	D1—Off for PROM load, on for RAM load. D2 through D8—Upper register #2 contents. When a microinstruction parity error is detected, the module clocks are inhibited, which stops the module. The bit content of the upper error register #2 is displayed on the LEDs. See Figure 2–6 for the location of the LEDs.
M.std (L0106)	Green	Board status—Indicates memory cycles are operating.
M.std2 (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.
P.ioc (L0105)	Amber	State indicator (top LED)—Mirrors the OCP State indicator.
	Amber	Run indicator (bottom LED)—Pulses at the on-board microprocessor run rate.
	Red	Board status—Indicates an inoperable module except during initialization when it comes on during module testing.
	Green	Board status—Indicates the module has passed all applicable diagnostics.
P.ioj (L0111 or L0111-YA)	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.
	D8 green	Board status—Indicates the module has passed all applicable diagnostics.

2.7 Module Switches

Specific switches are found on LINK (L0100-E2 or L0118), port processor (L0107), and port buffer (L0109) modules as follows:

- **CI port LINK module (L0100-E2/L0118)**—Refer to Figure 2–8 for the CI node address switches mounted on the L0100-E2 or L0118 module.

NOTE
Memory module M.std2 (L0117) contains a switch pack. These switches are factory set to calibrate the RX33 diskette controller. Do not change the setting of this switch pack; the switch settings are unique to each module and cannot be restored outside of the manufacturing environment.

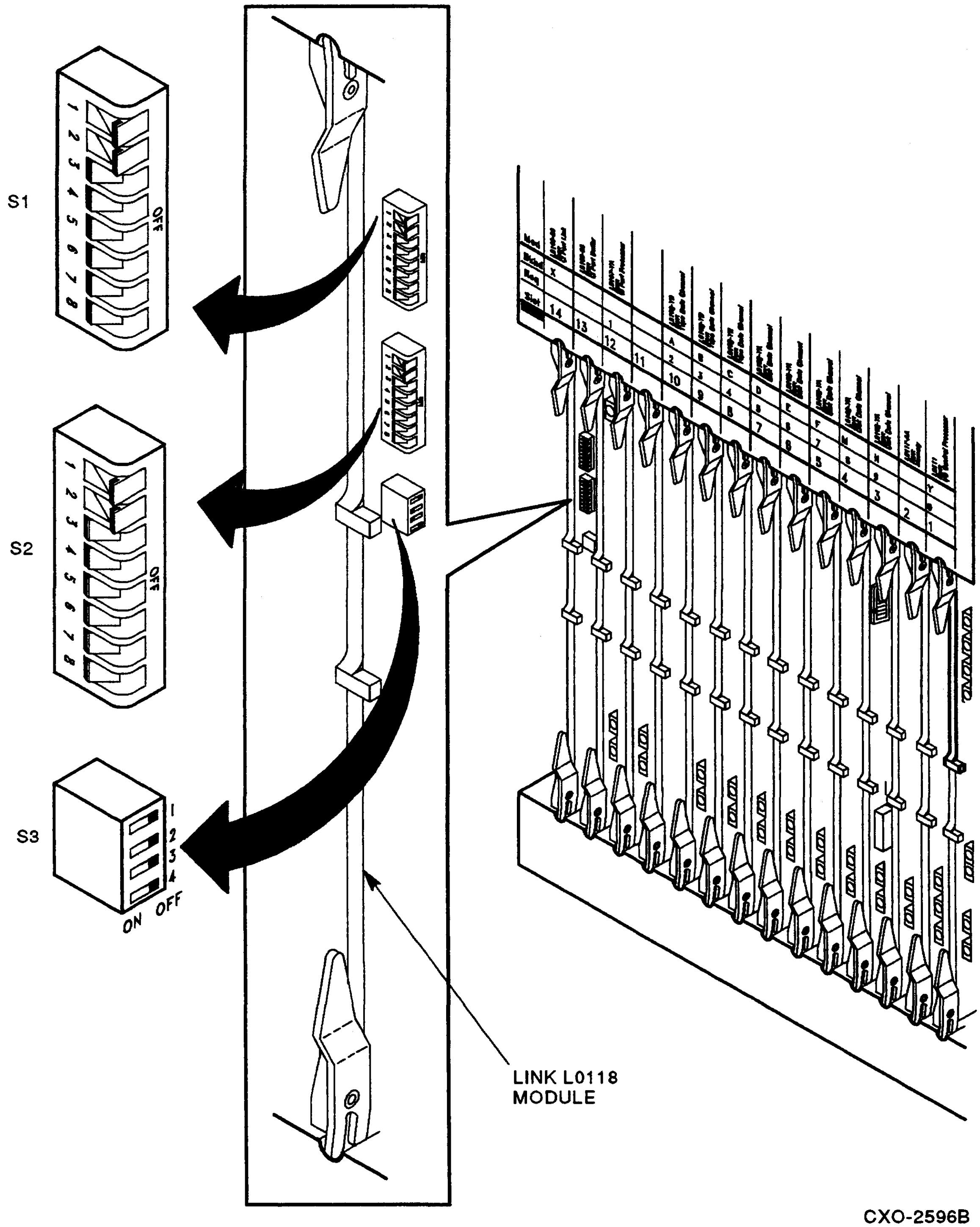


Figure 2–8 L0118 Module (DIP) Switches

Switches must be identically set to avoid CI addressing errors. See Chapter 3 for switch positions of S1, S2, and S3.

- **CI port processor and CI port buffer modules (L0107 and L0109)**—Both the L0107 and L0109 modules have dual in-line pack (DIP) switches to indicate the hardware revision level. DIP switch positions should not be changed, except as directed by a Field Change Order (FCO).

  Figure 2–9 shows the location of the L0107 switches.

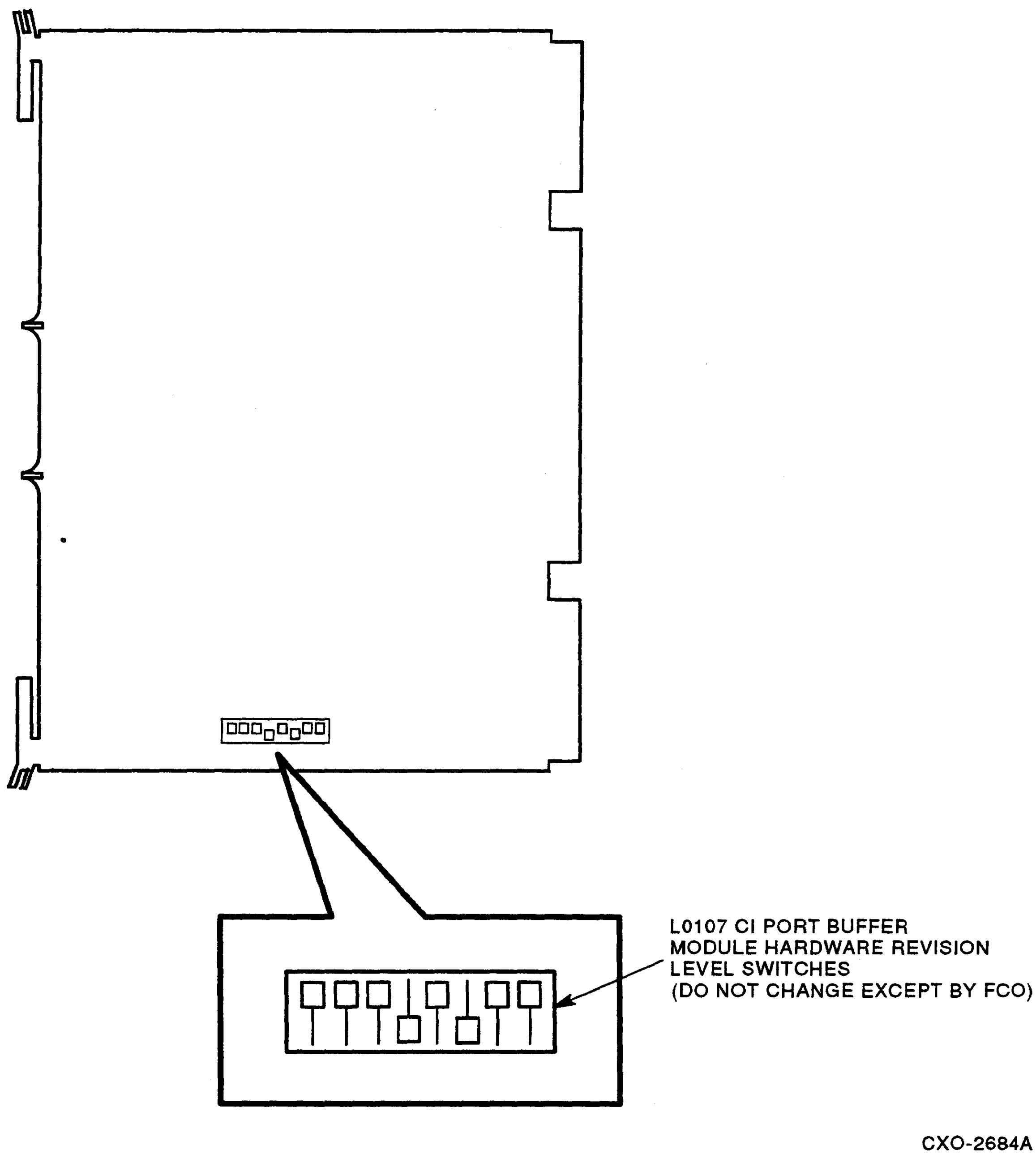


Figure 2–9 L0107 Module (DIP) Switches

Figure 2–10 shows the location of the L0109 switches.

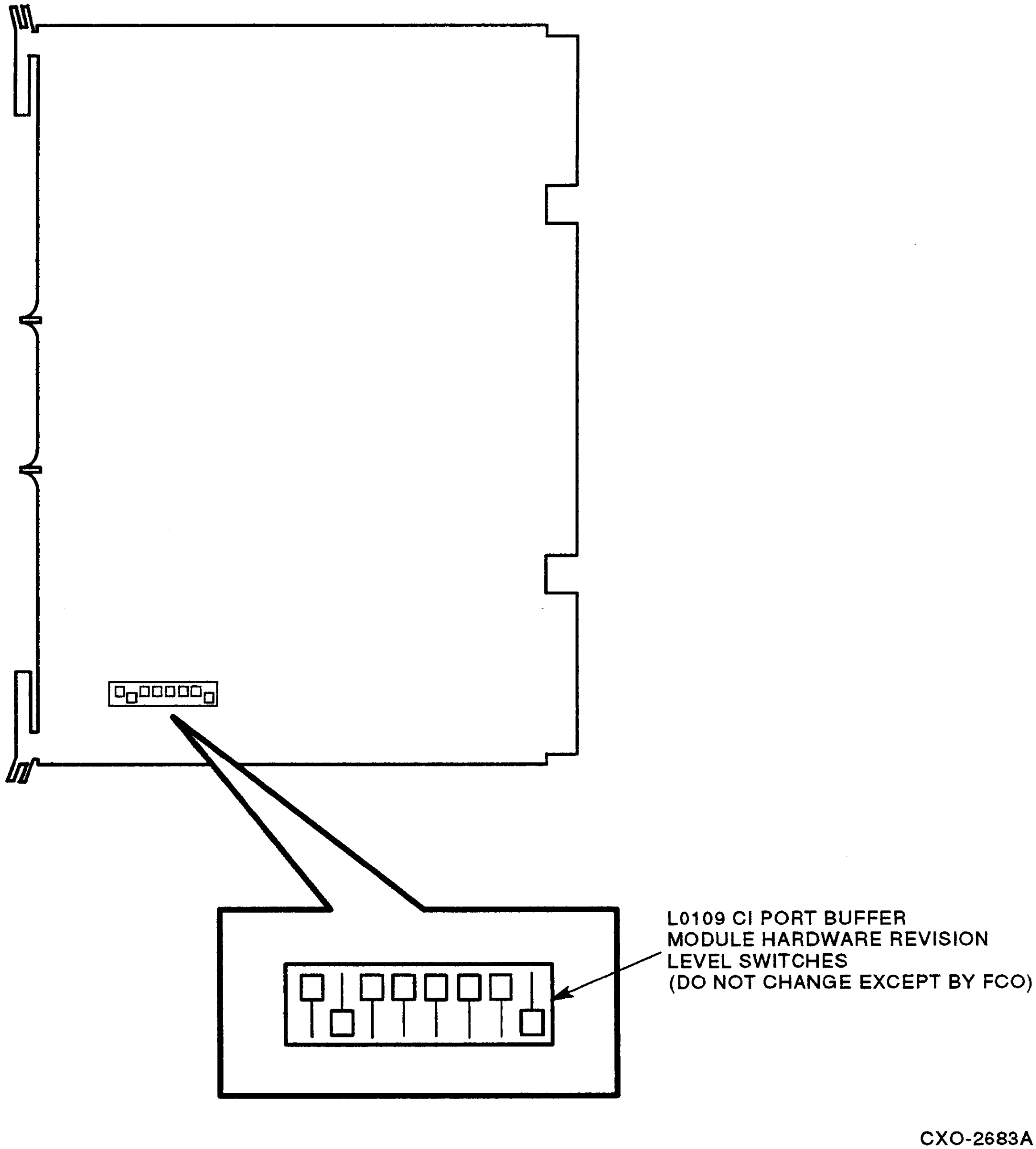


Figure 2–10 L0109 Module (DIP) Switches

- **K.si (L0119-YA) data channel switchpack** – Figure 2–11 shows the location of the K.si module switches.

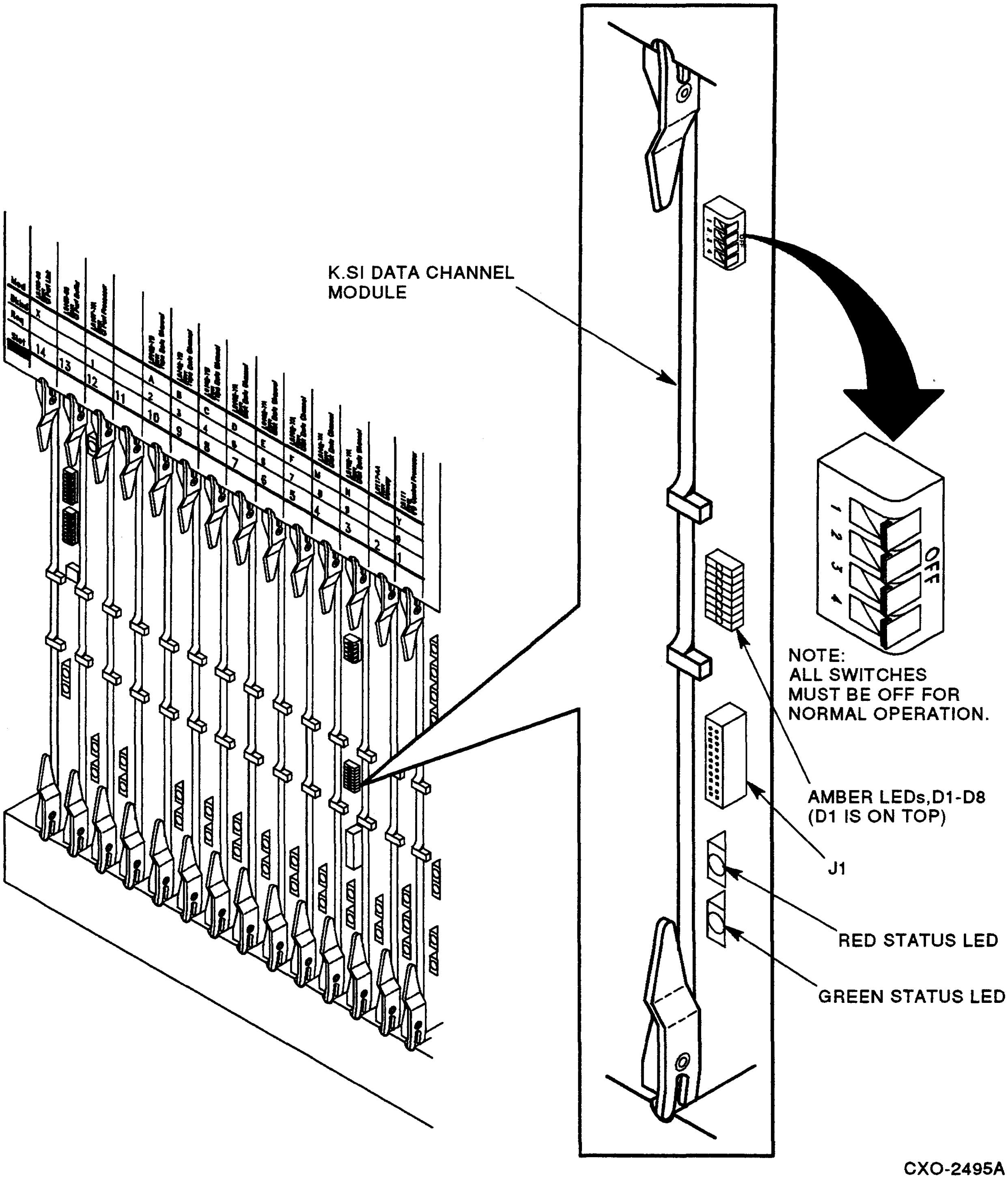


Figure 2–11 K.si Module (L0119-YA) Switches

- **P.ioj (L0111/L0111-YA)**—The P.ioj 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 used only when a default HSC SDS-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.

2.8 881 Power Controller

The 881 power controller is a general-purpose, three-phase controller that controls and distributes ac power to various ac devices (power supplies, fans, blower motor, and so forth) packaged within an HSC and HSC50. 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
- Power Control bus inputs
- Power Control bus delayed output (to allow sequencing of other controllers)

2.8.1 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 2–12).

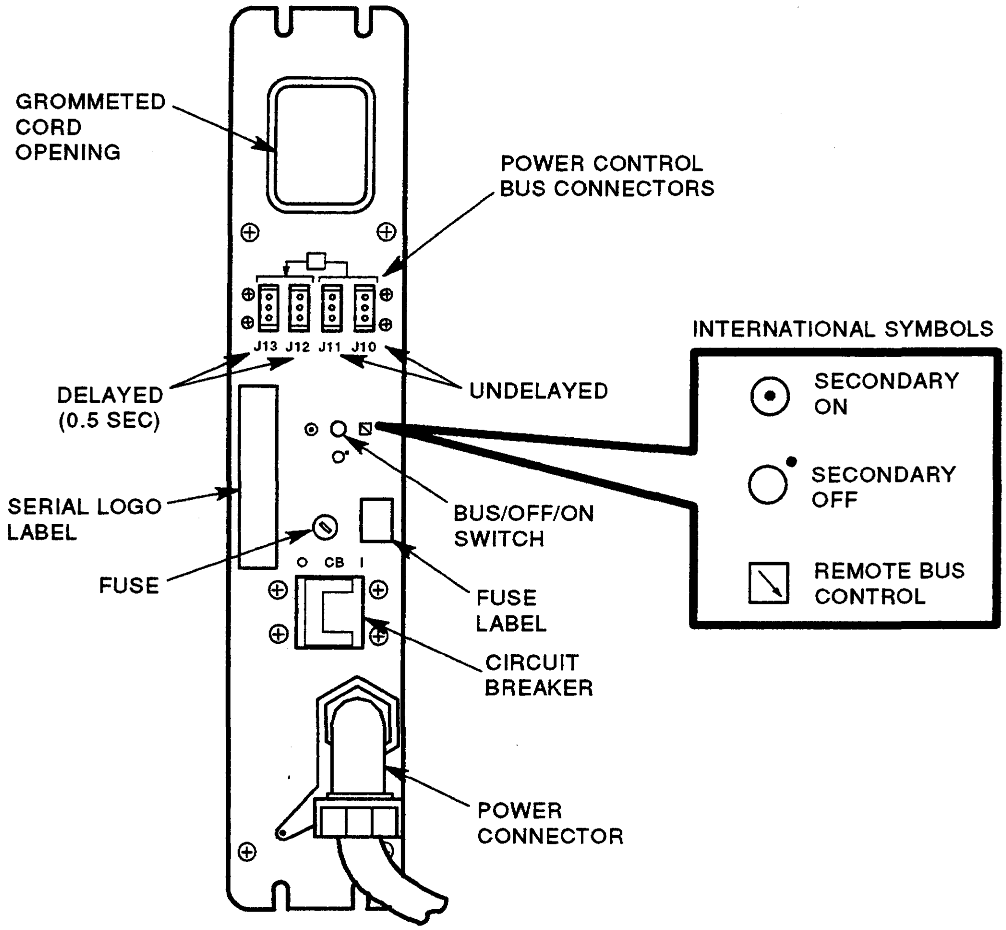


Figure 2–12 881 Power Controller—Front Panel Controls

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.
- **Power Control bus connections**—Used if Control bus connections to another cabinet are required. Power Control bus connectors are J10, J11, J12, and J13. Connectors J10 and J11 are not delayed. Connectors J12 and J13 are delayed.
- **BUS/OFF/ON switch**—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 power control bus is not used.

- **TOTAL OFF connector**—A two-pin male connector on the rear panel of the power controller (Figure 2–13). It removes power from the HSC whenever the air flow sensor detects system air-flow loss or an over temperature condition. To reset the TOTAL OFF, cycle the circuit breaker off and then back on again.

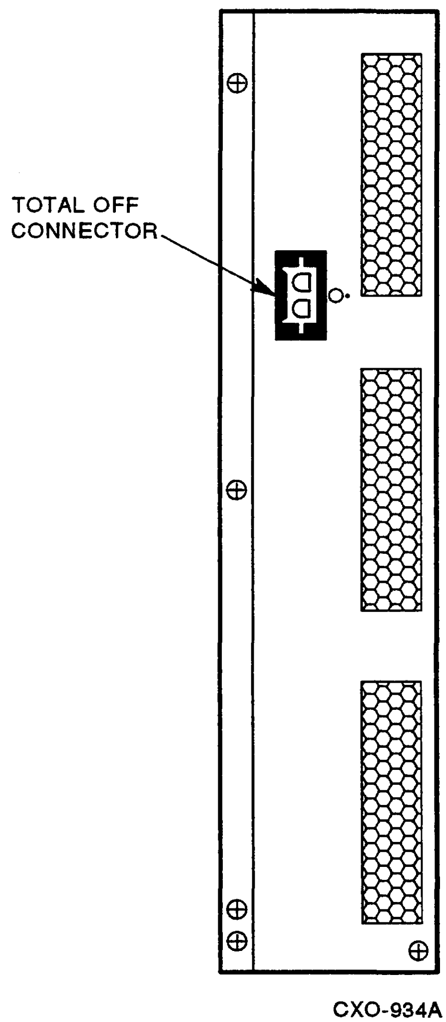


Figure 2–13 881 Rear Panel

2.9 HSC50 Power Controller

The 60 Hz power controller is shown in Figure 2–14 and Figure 2–16. For the 50 Hz unit, refer to Figure 2–15 and Figure 2–17. A physical description of the power controller follows.

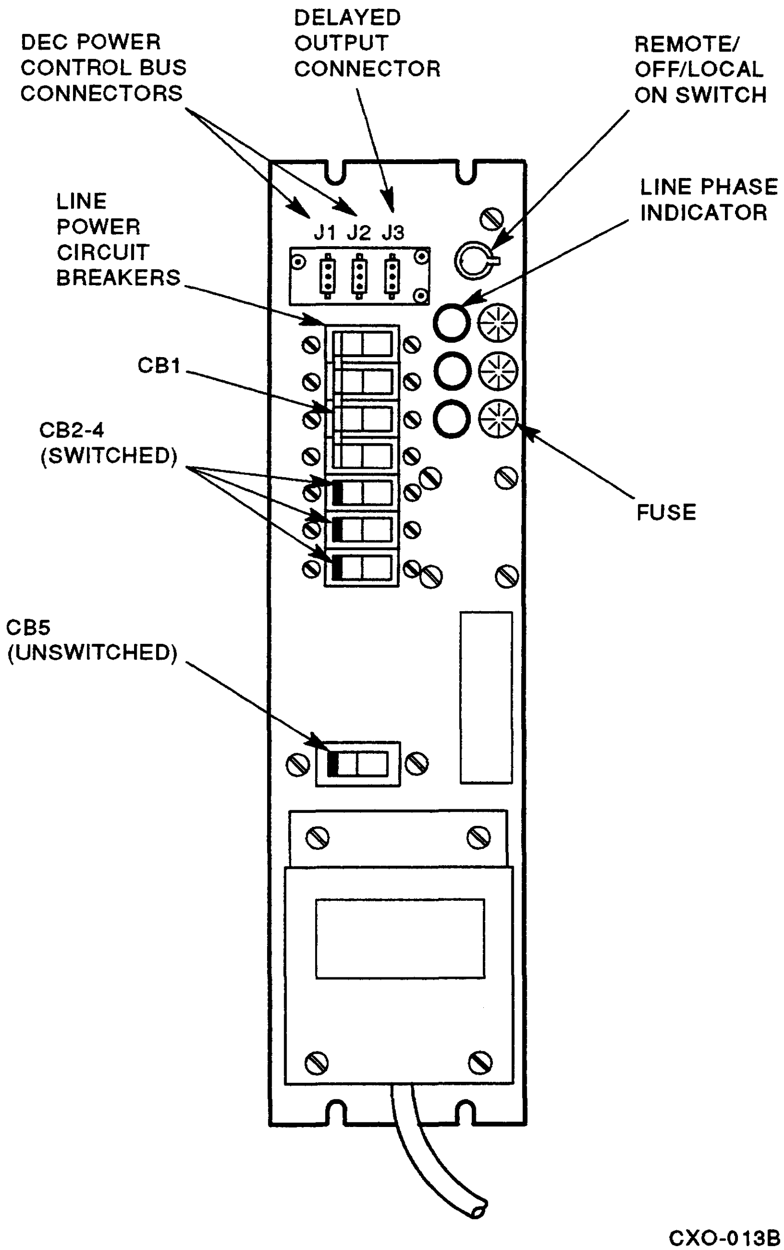


Figure 2–14 HSC50 Power Controller (60 Hz)—Front View

2.9.1 Line Phase Indicators

Three Line Phase indicators display the status of incoming line power. If any phase drops, the indicator for that phase goes off.

2.9.2 Fuses

The three line phases are fused to protect the HSC50 circuitry. These fuses are located beside the Line Phase indicators as shown in Figure 2–14 and Figure 2–15

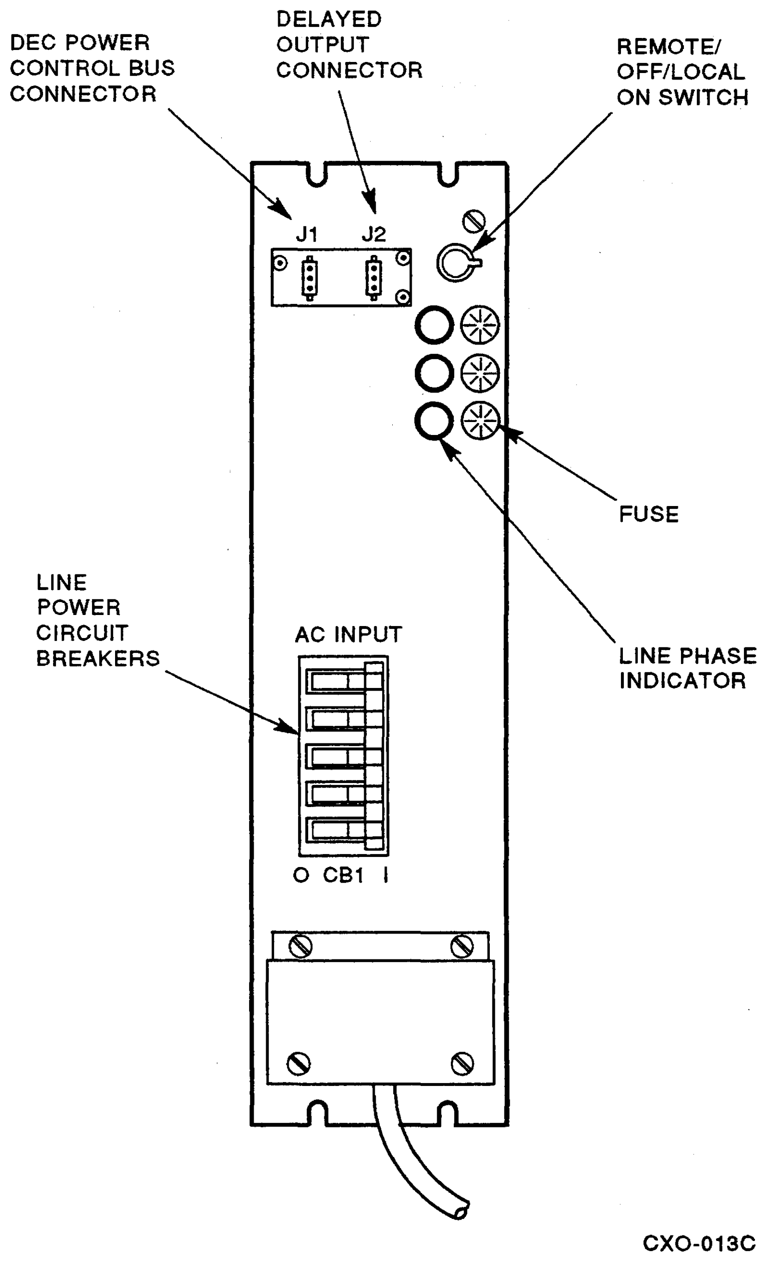


Figure 2–15 HSC50 Power Controller (50 Hz)—Front View

2.9.3 Remote/Off/Local On Switch

When this switch is in the off position, the power controller does not route ac line power to the switched or unswitched outlets.

With the switch in the Local On position, ac power is routed to the power controller switched or unswitched outlets.

When the switch is in the Remote position, the routing of ac power is dependent upon the Power Control bus signals.

2.9.4 Circuit Breakers (60 Hz)

There are five power controller circuit breakers which perform the following functions:

- CB1—Protects from incoming power surges
- CB2-4—Protects the switched outlets (refer to Figure 2–13)
- CB5—Protects the unswitched outlets

2.9.5 Circuit Breakers (50 Hz)

The 50 Hz unit contains one circuit breaker (Figure 2–15). CB1 on this unit protects all circuits.

2.9.6 Power Controller (60 Hz)—Rear View

The switched outlets in Figure 2–16 are protected by CB2-4 (refer to Figure 2–14) and the bottom (unswitched) by CB5. Both the bottom and top outlets are currently unused.

A three-pin male connector (J8) is located on the back of the power controller (Figure 2–16). It removes power from the HSC whenever the air flow sensor detects system air-flow loss or an over temperature condition. To reset the TOTAL OFF, cycle the circuit breaker off and then back on again.

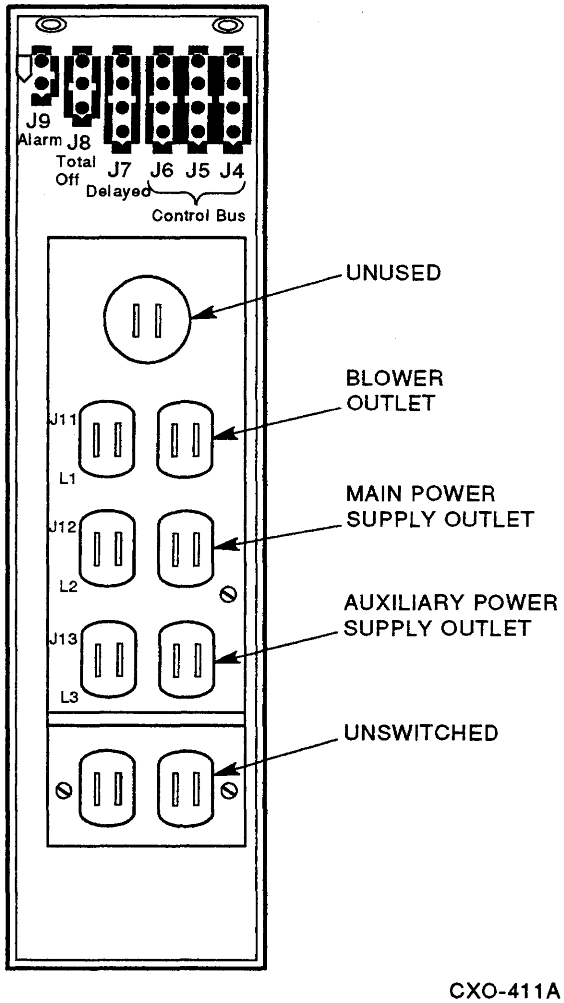


Figure 2–16 HSC50 Power Controller (60 Hz)—Rear View

2.9.7 Power Controller (50 Hz)—Rear View

Outlets in Figure 2–17 are protected by CB1 (refer to Figure 2–15). Connector J3, shown at the top of the 50 Hz power controller rear view, connects the air flow sensor.

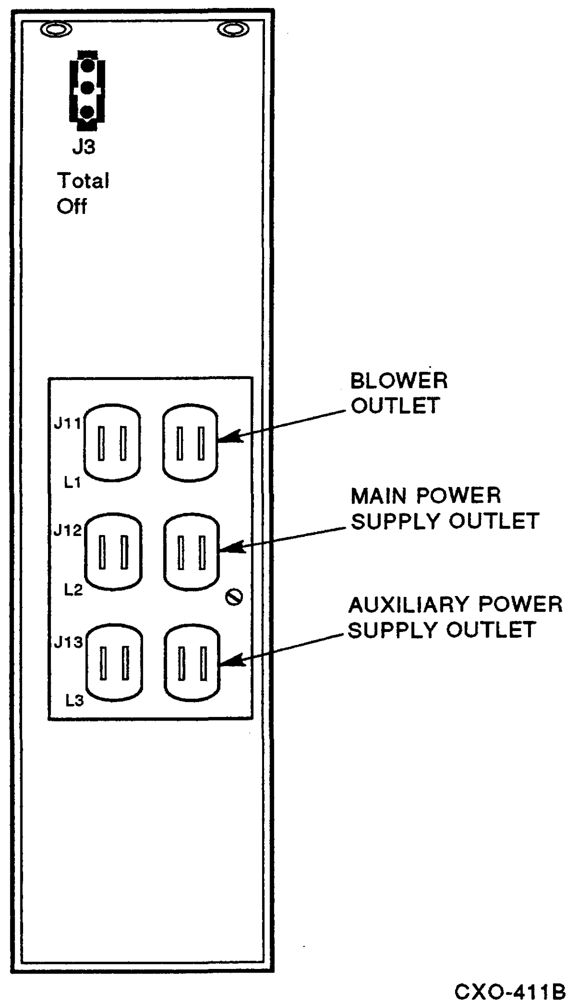


Figure 2–17 HSC50 Power Controller (50 Hz)—Rear View

3
Removal and Replacement Procedures

3.1 Introduction

This chapter emphasizes conditions that must be met when replacing field replaceable units (FRUs), including the following information:

- Safety precautions
- HSC failover
- FRU overviews
- Jumper configurations
- Switch configurations
- Test sequence to perform after FRU replacement

Observe the safety and electrostatic discharge (ESD) precautions in this section before starting removal and replacement procedures.

This chapter covers the following replaceable HSC subunits and modules:

Modules:

Port link module (LINK)
Port buffer module (PILA)
Port processor module (K.pli)
Disk data channel module (K.sdi)
Tape data channel module (K.sti)
Data channel module (K.si)
I/O control processor module (P.ioj and P.ioc)
Memory module (M.std2 and M.std)

Subunits:

RX33 disk drive
TU58 tape drive
Operator control panel
Air flow sensor
Blower
Power controller
Main power supply
Auxiliary power supply

3.2 Safety Precautions

Because hazardous voltages exist inside the HSC, service must be performed only by qualified people. Serious bodily injury or equipment damage can result from improper servicing. Observe the following safety steps before servicing the HSC:

1. Turn off the dc and ac power to the HSC before removing or installing internal parts or cables.
2. To ensure absolute safety, disconnect the ac plug from its receptacle after removing power from the HSC.
3. Remove and replace heavy subunits with care.
4. Use the Velostat (anti-static) kit (part number 29-11762) strap provided when removing and replacing logic modules.

3.3 Taking the HSC Off Line for Maintenance

This section describes how to take an HSC off line for performing maintenance.

3.3.1 Single HSC in a Cluster and Clusters Running ULTRIX/UNIX

If there is only one HSC in the cluster, or if the cluster is running the ULTRIX/UNIX operating system, use the following procedure:

1. Notify the system manager that the HSC is being taken off line and the drives attached to it will not be available.
2. Dismount the drives connected to the HSC or shut down the system.
3. Place the Off-line switch in the out position.
4. Take the HSC off line with *one* of the following methods:
   - Turn off the dc and ac power to the HSC.
   - Press the Init switch to reboot the HSC.

3.3.2 Multiple HSCs in a Cluster

Most VMS system clusters have primary and secondary paths established between the host and drives. If the HSC is the primary path, failover to the secondary path occurs and the drives remain available to the host. Use the following procedure when taking an HSC off line from a cluster with multiple HSCs:

1. Notify the system manager that you are taking the HSC off line.
2. Use the SETSHO command SHOW TAPE to determine which tape drives are on line to the HSC. Dismount these tape drives using the appropriate VMS commands.
3. Determine which disk drives are on line to the HSC undergoing maintenance with the SETSHO command SHOW DISK. The on-line drives must be either be failed over to the alternate HSC or dismounted.
4. Dismount single-ported disk drives using the appropriate VMS commands.
5. On all dual-ported disk drives, make sure that both A and B port select switches on the drives are pressed in. De-select the active port on the drive by pressing and releasing the illuminated port select switch.
6. Verify that the illuminated switch turns off and the alternate port switch illuminates. Actual failover time depends on the server timeout and may require 1 minute or more depending on activity. When the alternate port light illuminates, failover to the other HSC is successful.

7. Use the SETSHO command SHOW DISK to verify that all on-line drives have failed over to the alternate HSC.
8. If failover did not occur, reselect the ports on the drive and check the connections between the drives and the HSC.
9. After all tapes and disks have been failed over, take the HSC off line using *one* of the following methods:
   - Turn off the dc and ac power to the HSC, or
   - Place the Off-line switch in the out position, or
   - Press the Init switch to reboot the HSC.

3.4 Removing and Replacing Field Replaceable Units

The following sections describe procedures for removing and replacing field replaceable units (FRUs) in an HSC or HSC50:

3.4.1 Removing HSC Power

Before removing/replacing an FRU, turn off the ac power from the HSC. Following are the methods for removing dc and ac power from the HSC:

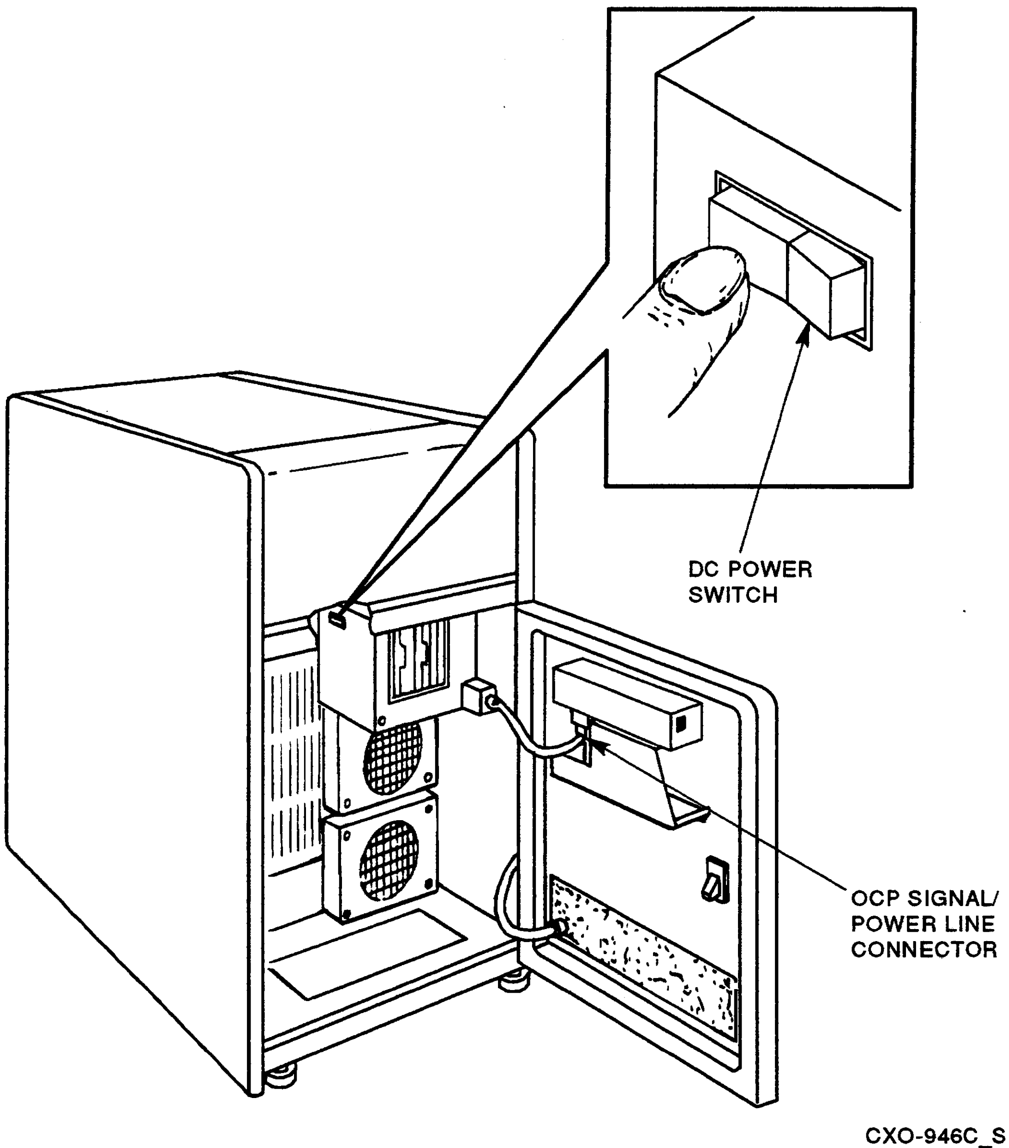


Figure 3–1 HSC DC Power Switch

1. Set the dc power switch on the side of the RX33 housing to the off position (Figure 3–1).

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.

2. Place the main power switch CB1 on the power controller in the off position (Figure 3–2). To ensure safety precautions, unplug the ac power plug from the ac socket.

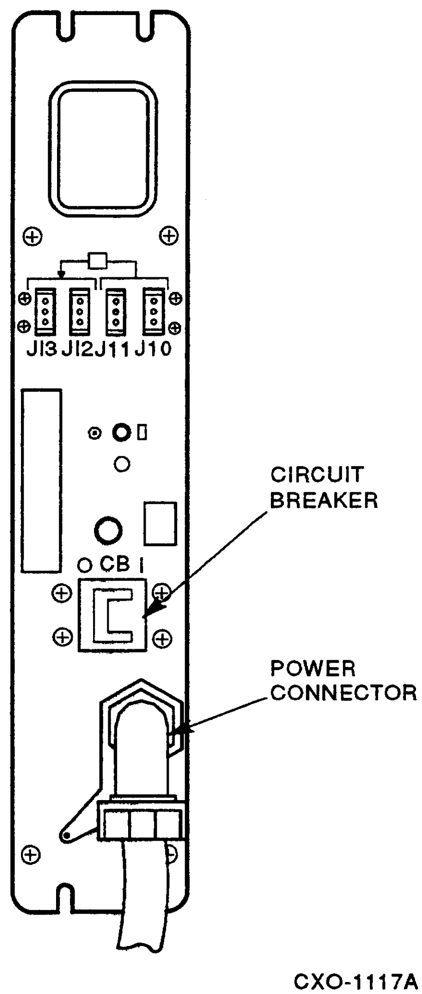


Figure 3–2 HSC 881 Power Controller Circuit Breaker

3.4.2 Removing HSC50 Power

Following are the steps for removing dc and ac power from the HSC50:

1. Set the dc power switch on the maintenance access panel to the off position (Figure 3–3).

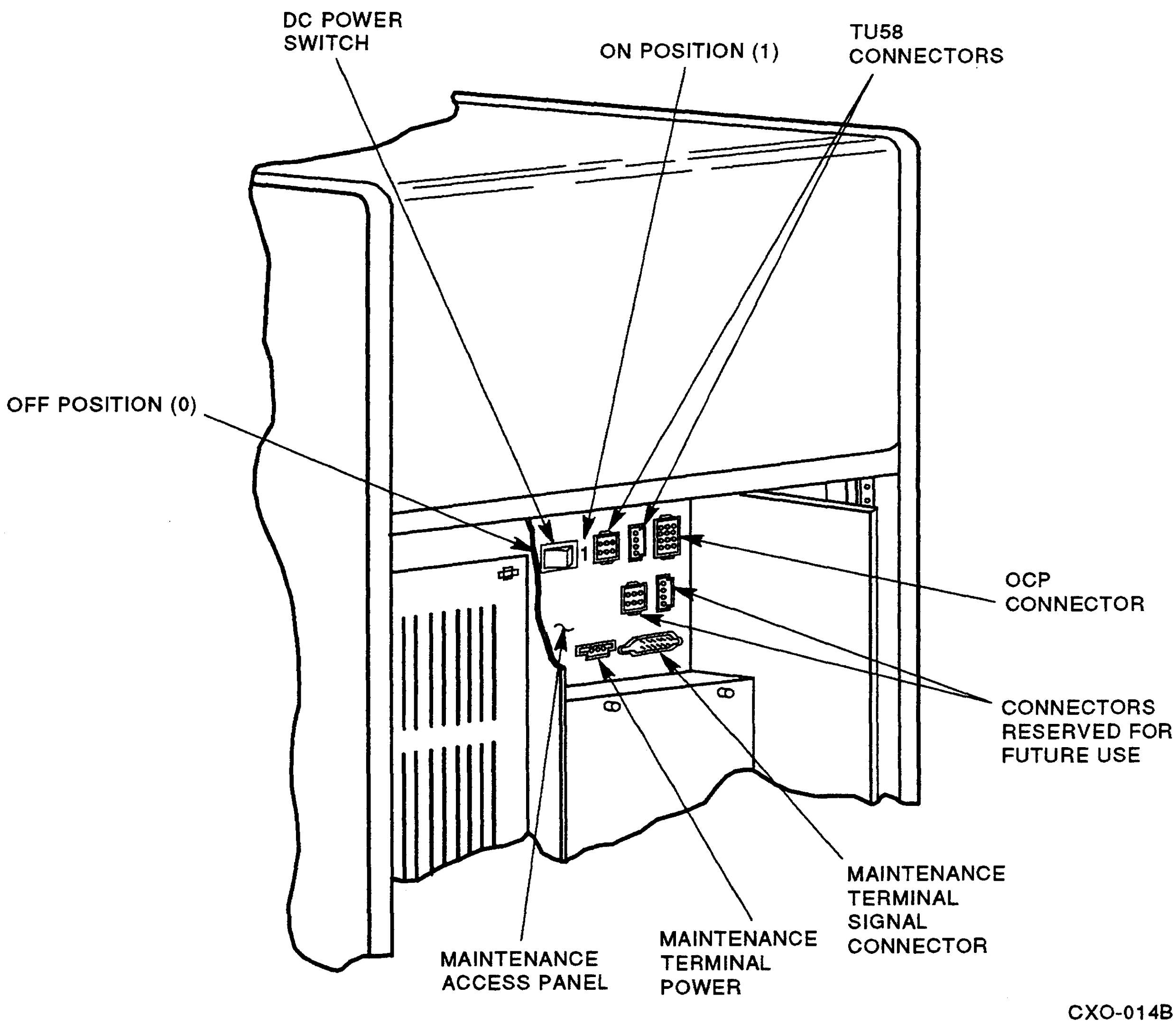


Figure 3–3 HSC50 DC Power Switch

2. Place the main power switch CB1 on the power controller in the off position (Figure 3–2) for the HSC50 (modified) and the line power circuit breakers (Figure 3–4) for the HSC50. To ensure safety precautions, unplug the ac power plug from the ac socket.

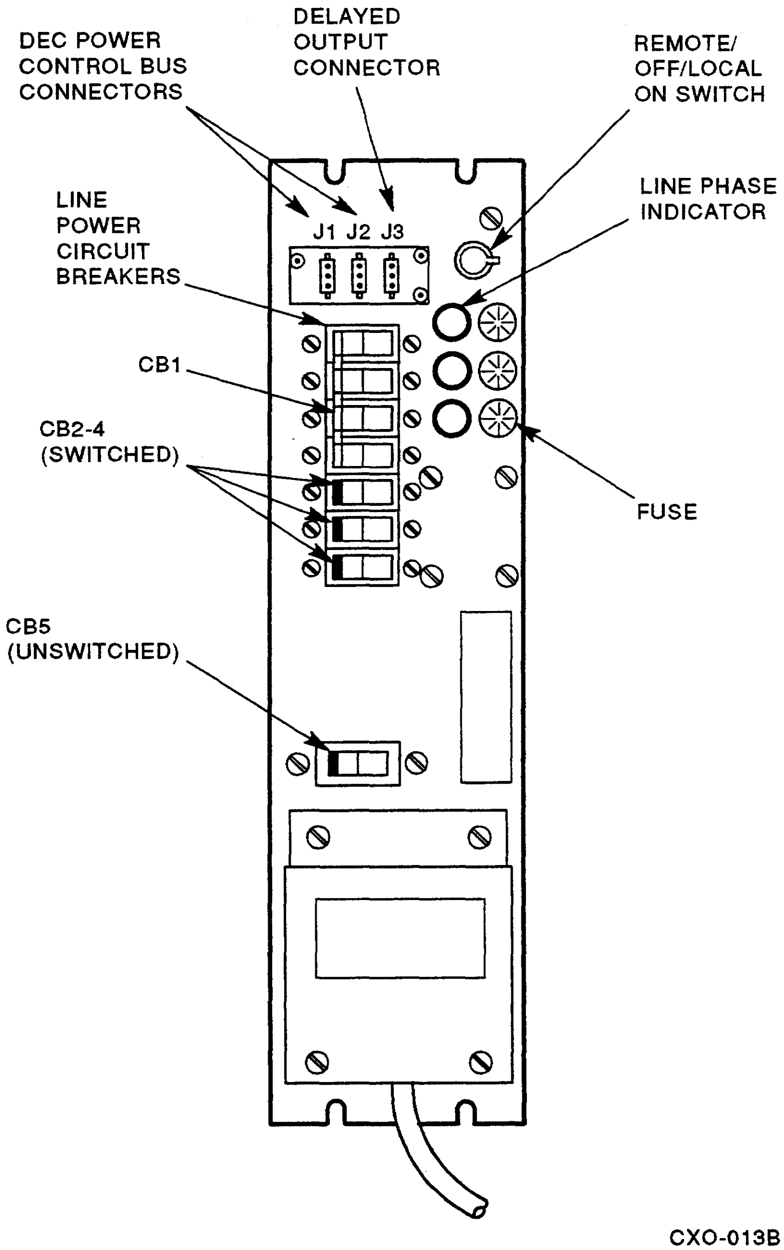


Figure 3–4 HSC50 Line Power Circuit Breakers

3.4.3 Removing Field Replaceable Units

Figure 3–5 shows the sequence for removing field replaceable units (FRUs) in the HSC:

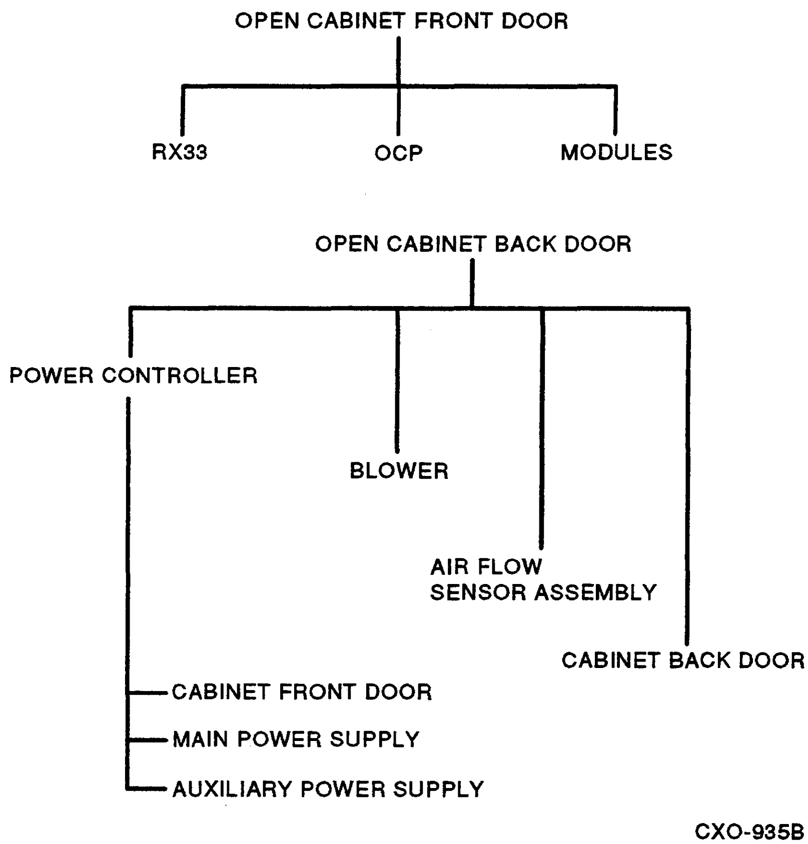


Figure 3–5 HSC FRU Removal Sequence

Figure 3–6 shows the FRU removal sequence for an HSC50.

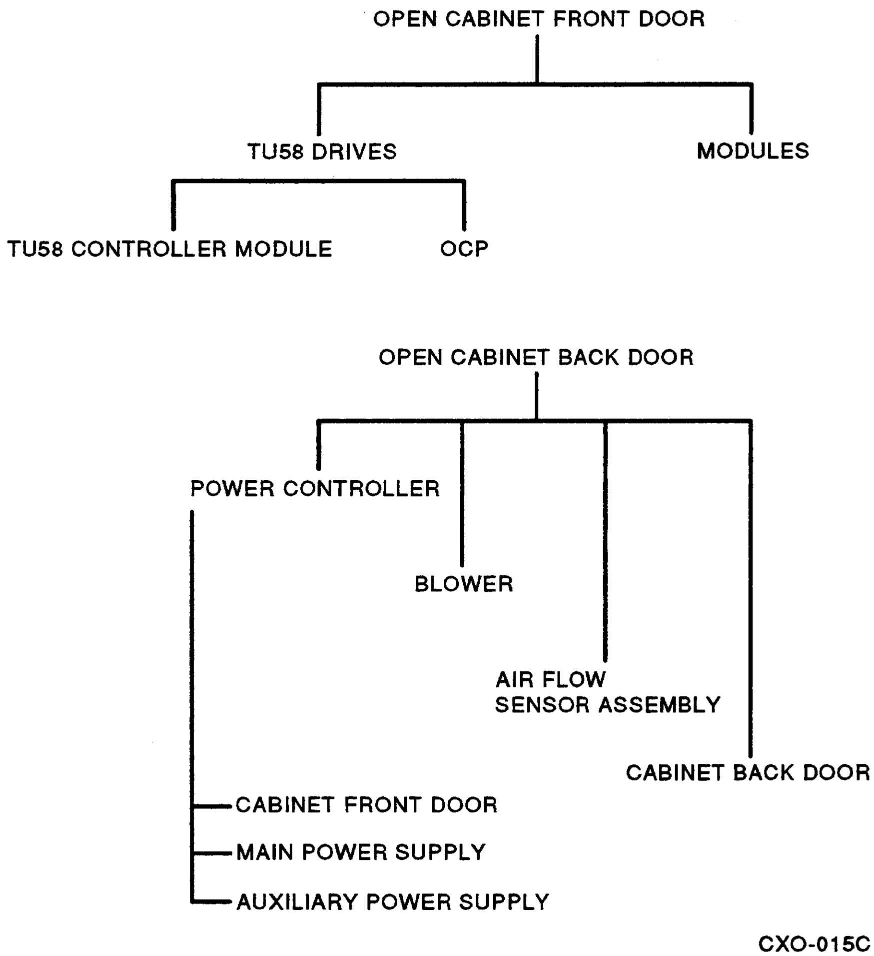


Figure 3–6 HSC50 FRU Removal Sequence

3.4.4 Removing the HSC Cabinet Front Door

The FRUs accessed through the front door include the RX33 drives, the operator control panel (OCP), and the logic modules. 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 HSC power by setting the dc power switch to the 0 position.
3. Disconnect the ground wire from the door.
4. Disconnect the OCP signal/power line connector at the bottom of the OCP shield (Figure 3–7).

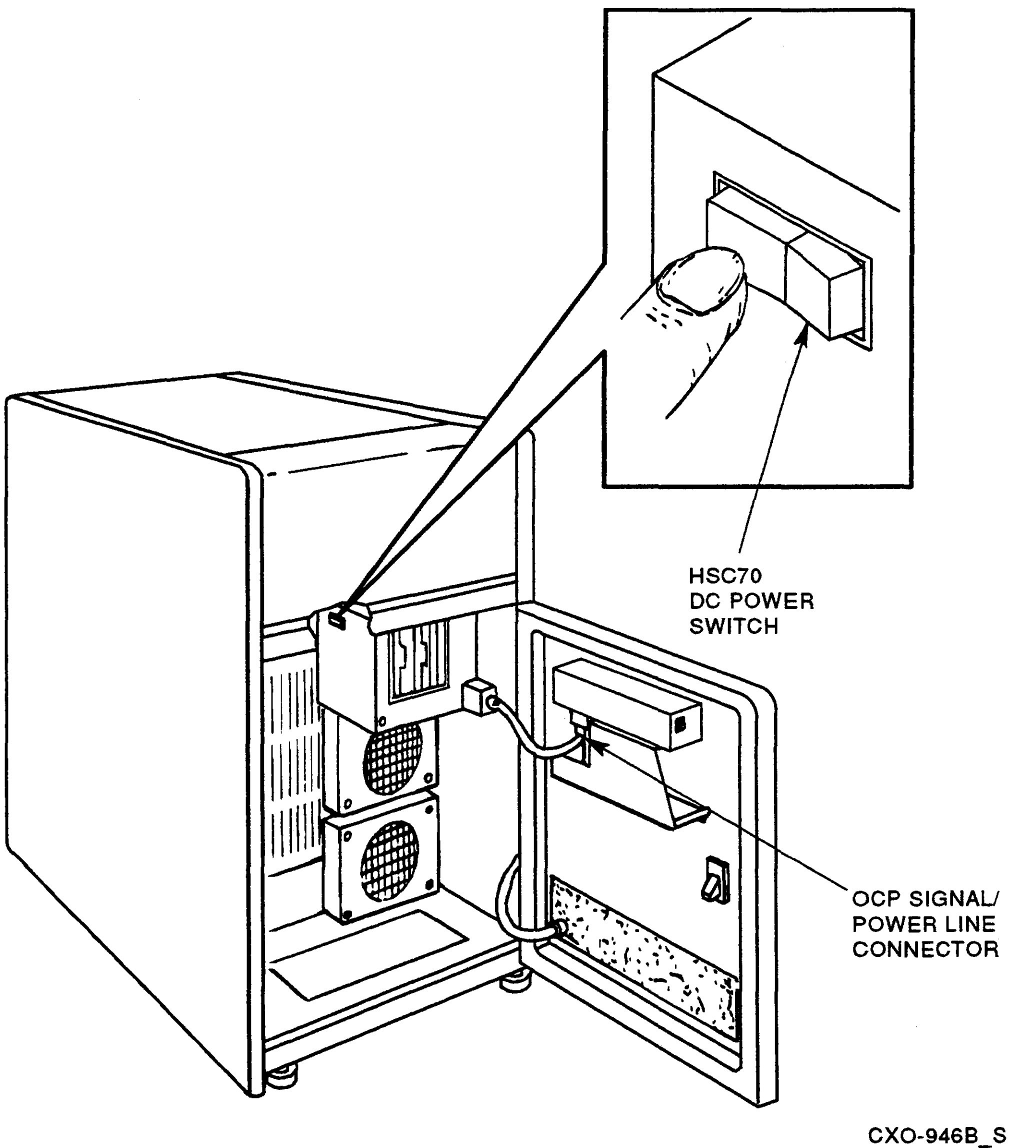


Figure 3–7 HSC OCP Signal/Power Line Connector

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

Reverse the removal procedure to replace the front door.

3.4.5 Removing the HSC50 Cabinet Front Door

The FRUs accessed through the front door include the TU58 drives, the operator control panel (OCP), and the logic modules. To remove the front door, use the following procedure:

1. Open the cabinet front door by turning the key clockwise.

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

2. Disconnect the ground wire from the door.
3. Remove the maintenance access panel cover by loosening the four captive screws.

   NOTE
   Some HSC50s have a hinged maintenance access panel with only one captive screw.

4. Remove HSC50 power by setting the dc power switch to the 0 position.
5. Remove the plastic cable duct cover.
6. Disconnect the cables from the maintenance access panel connectors shown in Figure 3–8.

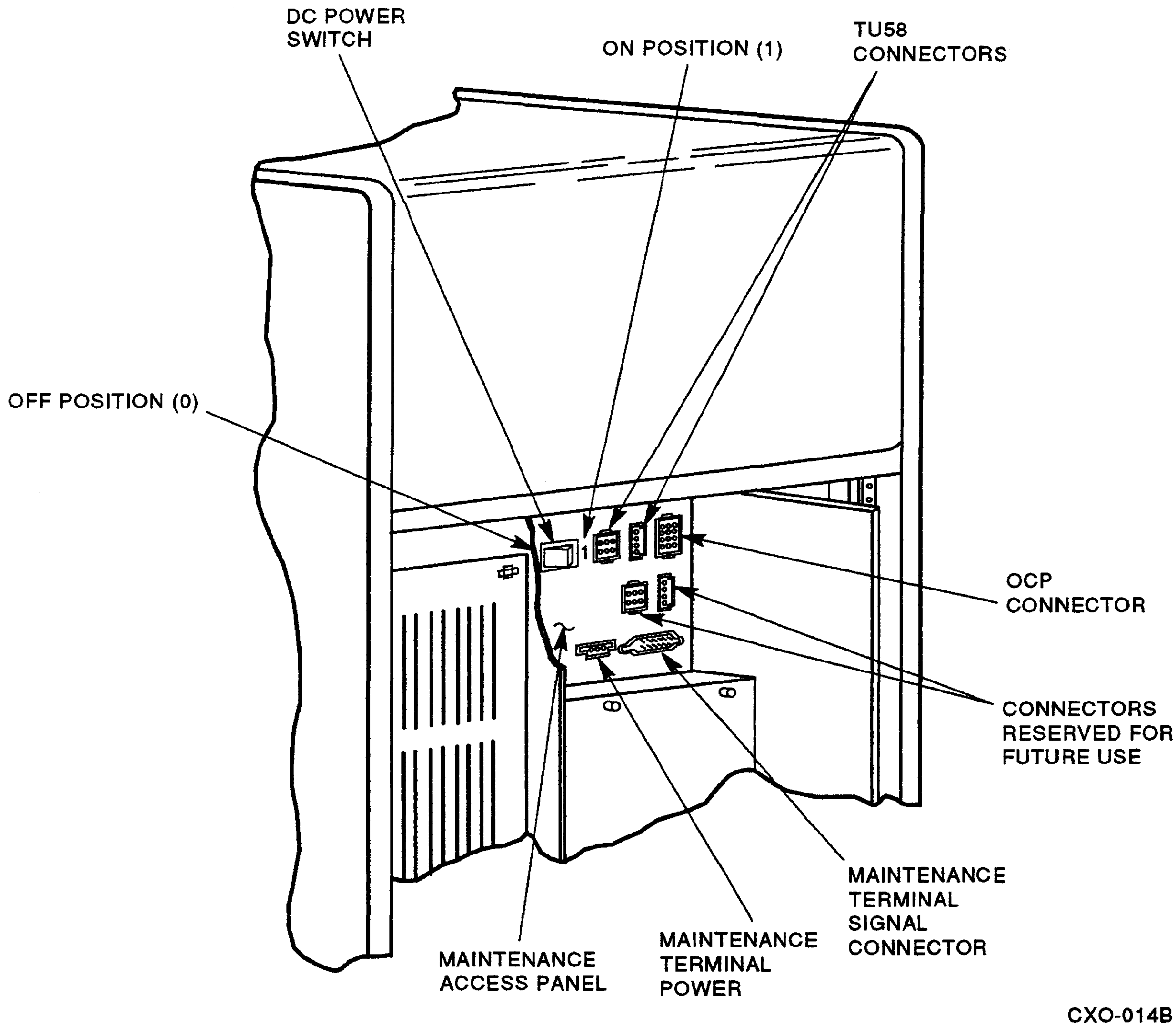


Figure 3–8 HSC50 Maintenance Access Panel Connectors

7. Pull down on the spring-loaded rod on the top hinge inside the cabinet to disengage the door, then lift the door off its bottom pin.

Reverse the removal procedure to replace the front door.

3.4.6 Removing the HSC Cabinet Back Door

The FRUs accessed through 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 to disengage the door, then lift the door off its bottom pin.

Reverse the removal procedure to replace the back door.

3.5 Removing and Replacing Modules

This section contains procedures for removing and replacing modules. Refer to Figure 3–9 for the HSC and Figure 3–10 for the HSC50 when removing the card cage cover.

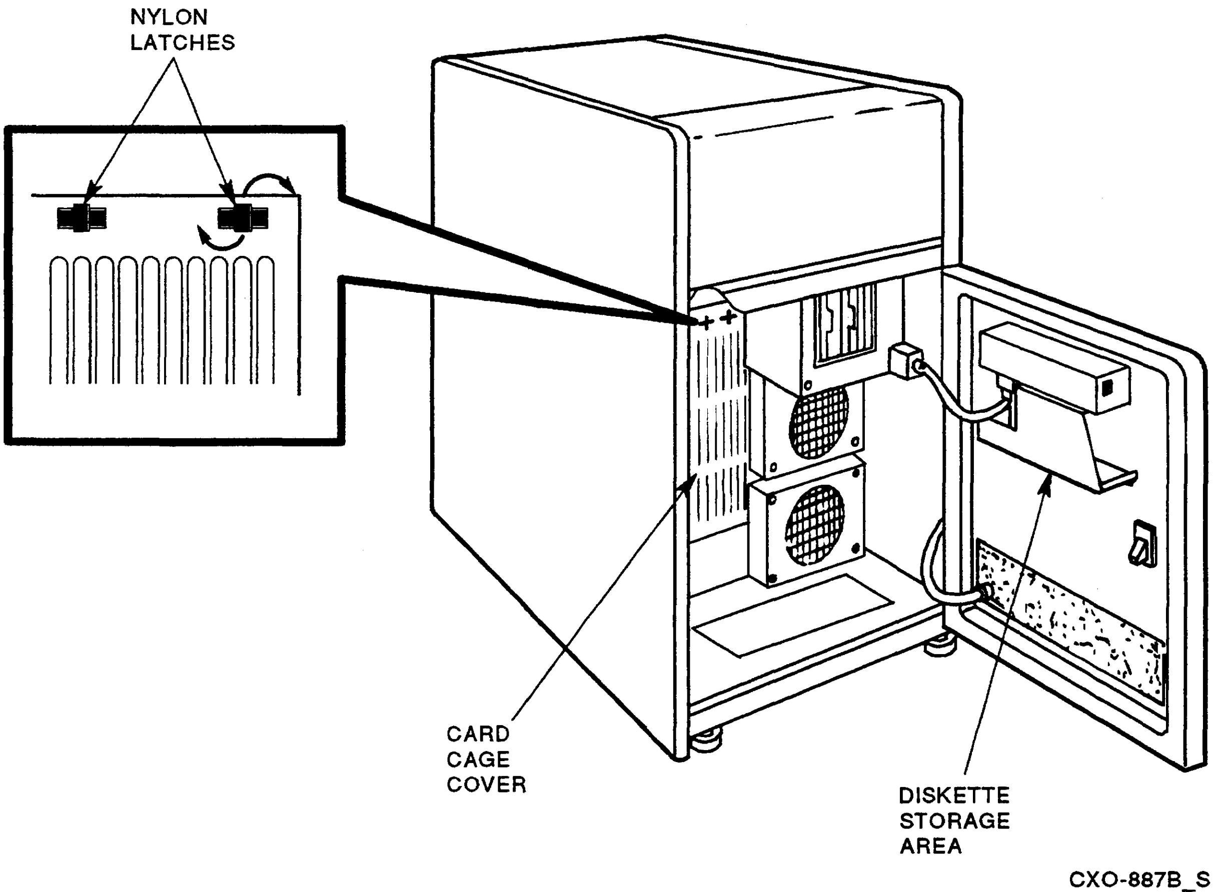


Figure 3–9 HSC Card Cage Cover Removal

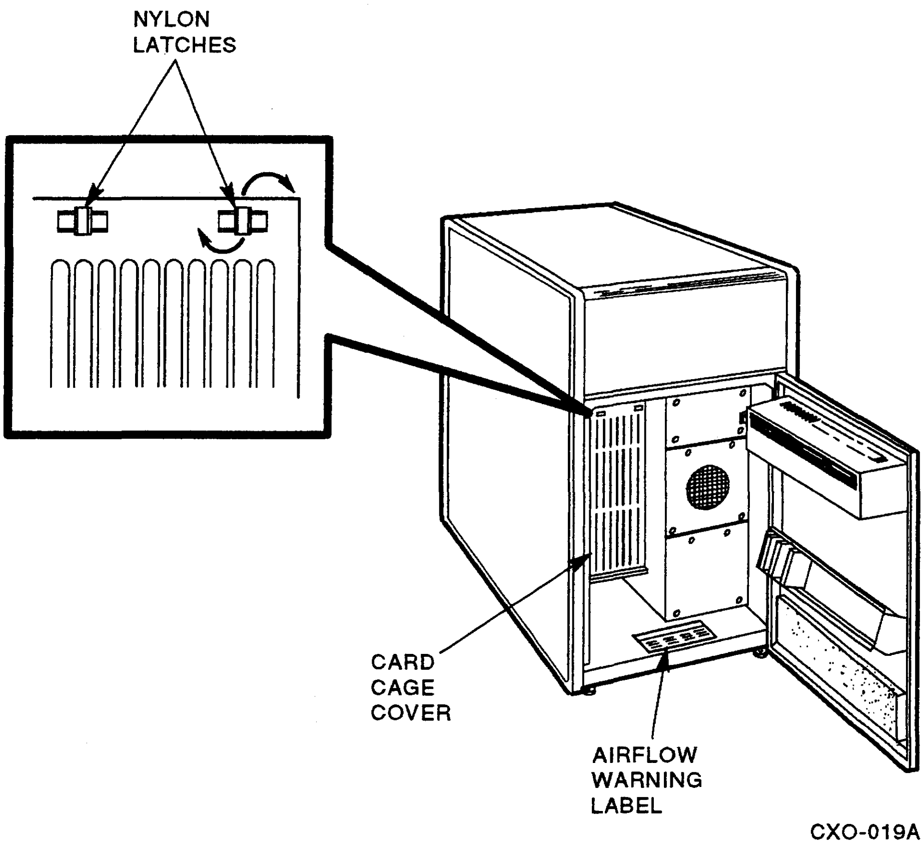


Figure 3–10 HSC50 Card Cage Cover Removal

WARNING:
Because hazardous voltages exist inside the HSC, service must only be performed by qualified people. Bodily injury or equipment damage can result from improper servicing procedures.

A Velostat (anti-static) kit (part number 29-11762) *must* be used during module removal/replacement.

3.5.1 Removing and Replacing the Port Link Module (LINK)

The port link module (LINK) is a part of the host interface module set (K.ci). With all configuration switches and jumpers in default positions, the L0118 is functionally identical with the L0100 or L0100-E2.

3.5.1.1 Removing the LINK Module

Use the following procedure to remove the LINK module. Observe safety and ESD precautions before starting the module removal procedure.

1. Notify users that the HSC is being taken off line and the drives attached to it will not be available.
2. Dismount or failover any drives connected to the HSC or shut down the system.
3. Set the dc power switch to the 0 (off) position.

   On the HSC, the dc power switch is located on the side of the RX33 housing.

On the HSC50, the dc power switch is located on the maintenance access panel.

4. Turn the two nylon latches on the card cage cover one-quarter turn.
5. Pull the HSC card cage cover up and out.
6. Locate the LINK module in slot number 14 of the card cage. This can be verified by the module utilization label.
7. Move the door latch plate attached to the left side of the cabinet frame away from the module removal path. In the HSC cabinet, the latch plate is swivel mounted. Lift the plate slightly and press it flat against the cabinet frame. Remove the LINK module.

3.5.1.2 Setting the Replacement LINK Module Switches

S1 and S2 are the node address switches on the LINK module. The node address switches on the replacement module must be set identically to the switch settings on the removed LINK module. The L0100-E2 and L0118 LINK module also have an additional switch pack (S3).

The switch configurations and significance are as follows:

S1/S2 — Node number
S3-1 — Cluster size (GT15), OFF for 16 or less nodes (default); ON for 17 or more nodes
S3-2, S3-3 — Delta time/quiet slot (default) = always OFF
S3-4 — 10 ticks = always ON

Figure 3–11 shows the LINK (L0100) module node address switches.

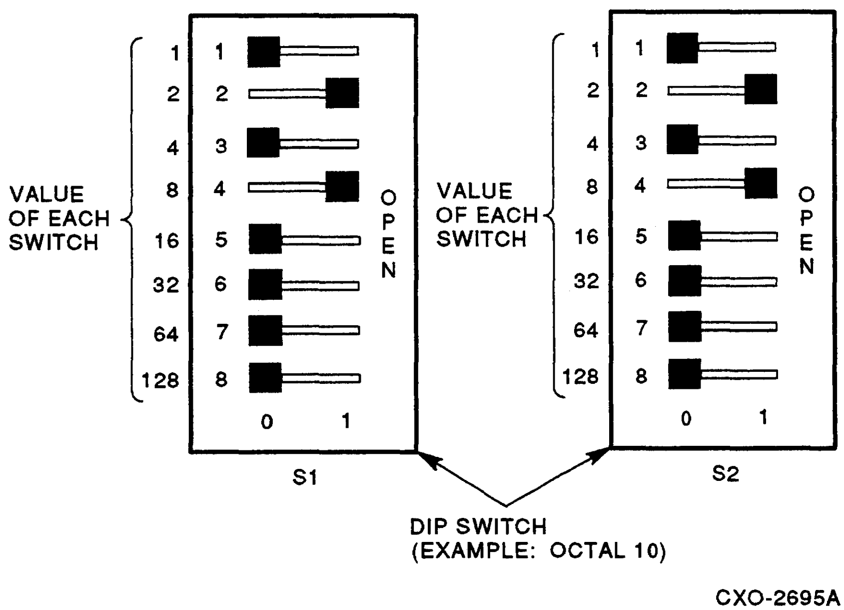


Figure 3–11 L0100 Node Address Switches

Figure 3–12 shows the LINK (L0100-E2/L0118) module node address switches.

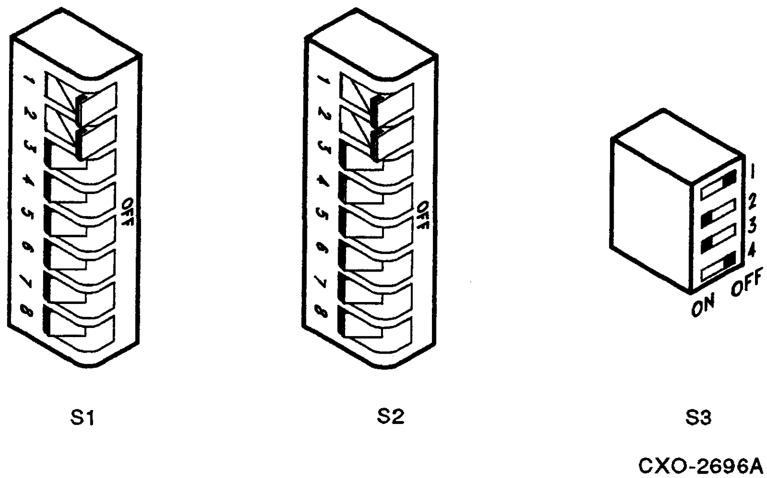


Figure 3–12 L0100-E2/L0118 Node Address Switches

3.5.1.3 Setting the Replacement LINK Module Jumpers

The LINK jumper configurations are as follows:

W1—Extender head, Default = Not used
W2—Active hub, Default = Not used
W3—Extender ACK timeout, Default = Not used
W4—Cluster size (GT32), Default = Less than 32

Figure 3–13 shows the LINK (L0100) jumpers.

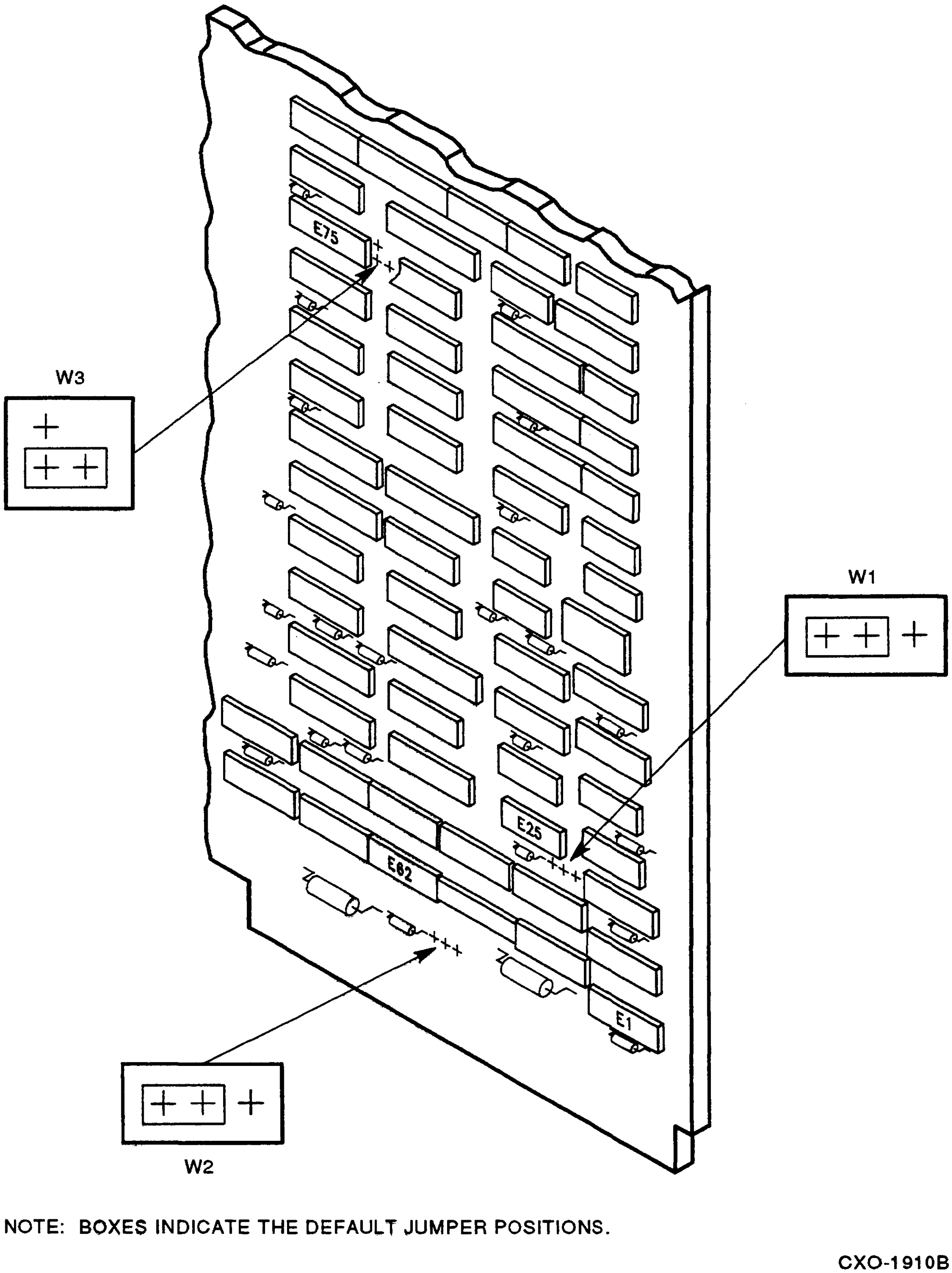


Figure 3–13 L0100 Jumper Configuration

Figure 3–14 shows the LINK (L0118-B1) jumpers.

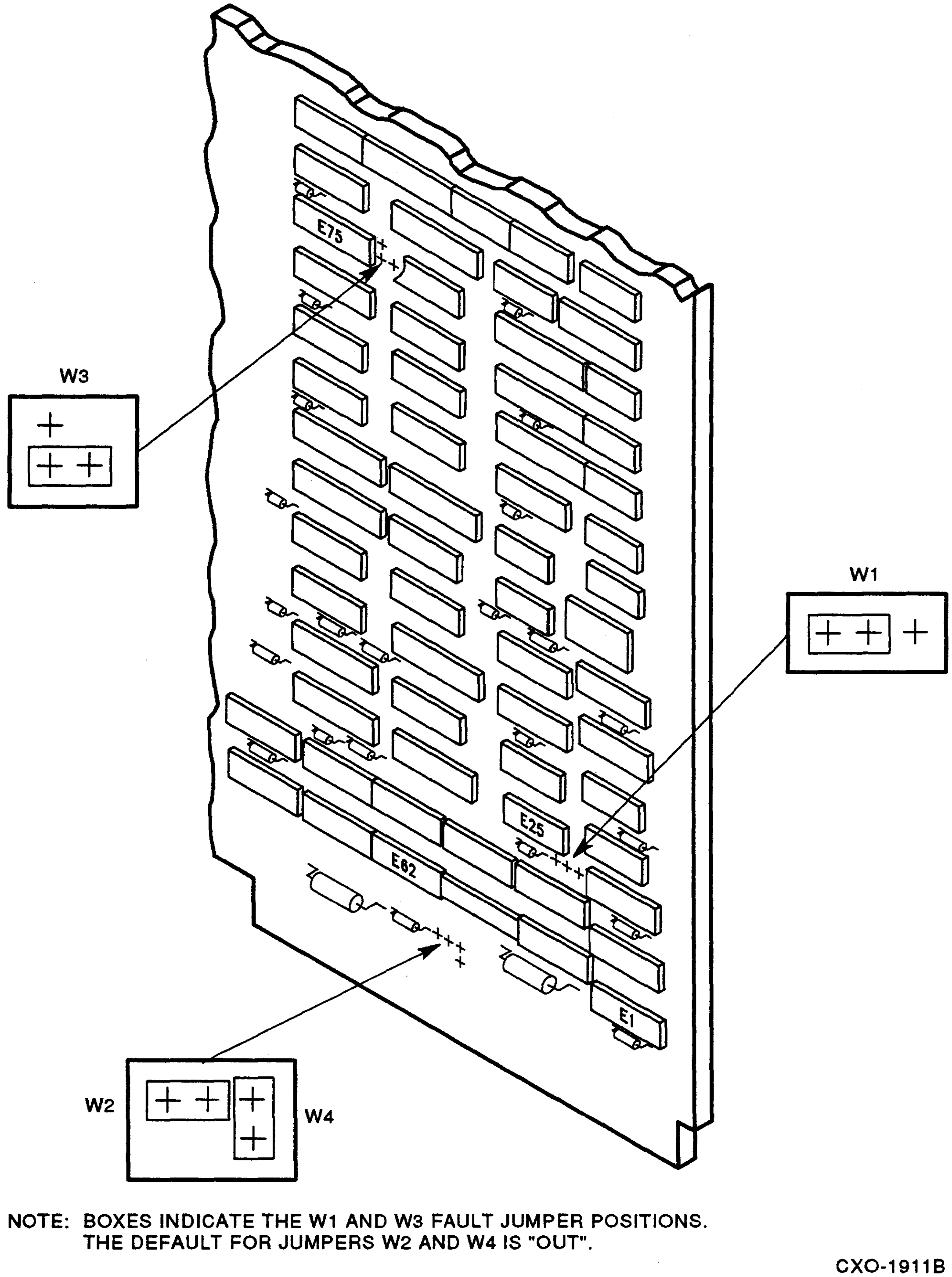


Figure 3–14 L0118-B1 Jumper Configuration

Figure 3–15 shows the LINK (L0118-B2) jumpers.

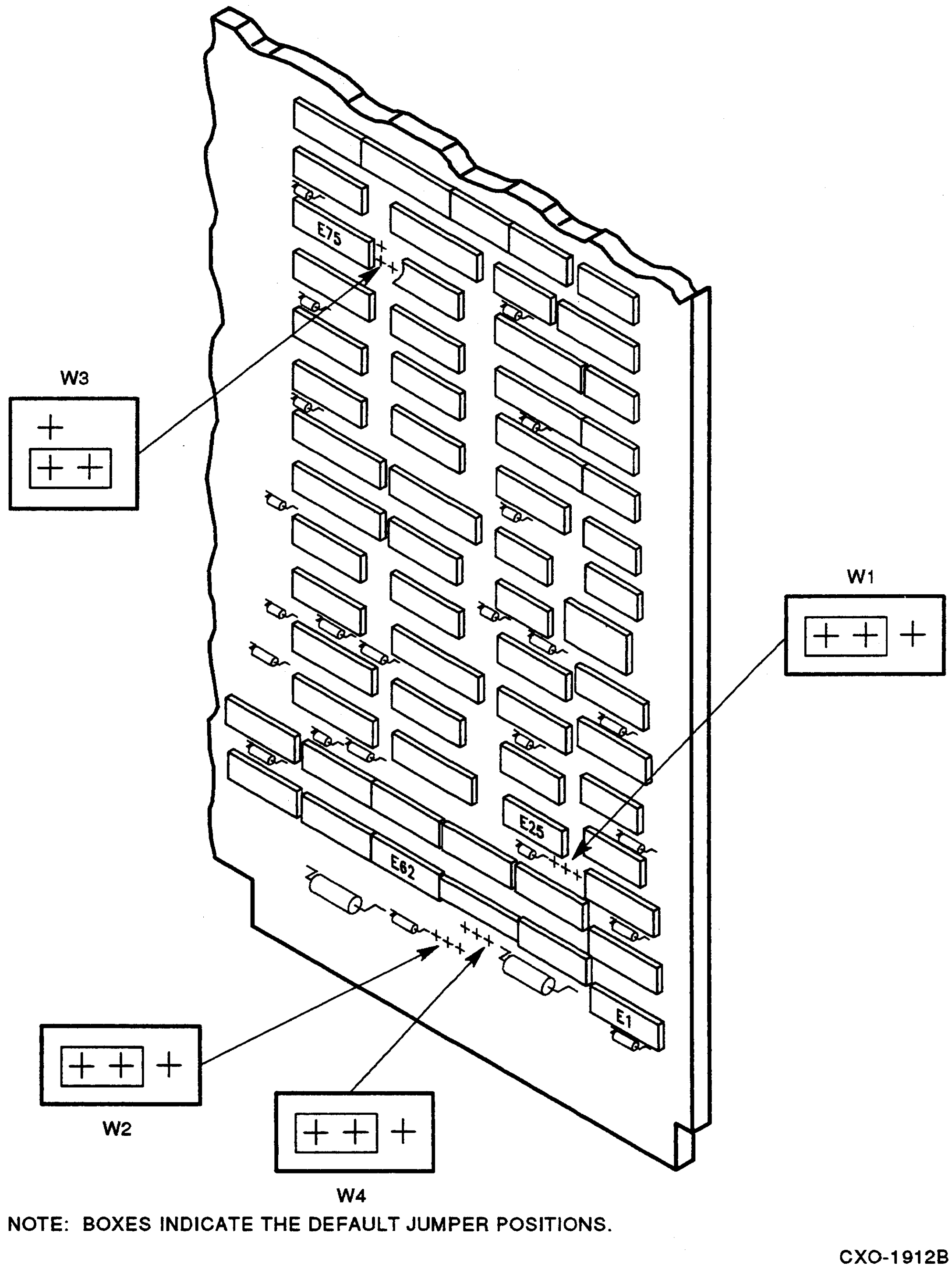


Figure 3–15 L0118-B2 Jumper Configuration

3.5.1.4 Replacing the LINK Module

This section provides the LINK module replacement procedure. Observe safety and ESD precautions before starting the module replacement procedure.

1. Install the LINK module in slot number 14 of the card cage. This can be verified by the module utilization label.
2. Move the door latch plate attached to the left side of the cabinet frame away from the module installation path. In the HSC cabinet, the latch plate is swivel mounted. Lift the plate slightly and press it flat against the cabinet frame. Install the LINK module and return the plate to its locked position.
3. Pull the card cage cover down and in.
4. Turn the two nylon latches on the module cover plate one-quarter turn.
5. Set the dc power switch to the 1 (on) position.

   On the HSC, the dc power switch is located on the side of the RX33 housing.

   On the HSC50, the dc power switch is located on the maintenance access panel.

3.5.1.5 Testing the LINK Module

Perform the following tests to verify correct LINK operation as part of the K.ci host interface module set.

1. Boot the HSC with the off-line diagnostic media. Refer to Chapter 6 for boot procedures.

   NOTE
   The off-line diskette must be write protected. Place a write-protect tab over the diskette write-enable notch.

2. Refer to Chapter 6 for test descriptions and procedures and perform the following tests:
   - Off-line bus interaction test
   - Off-line K test selector
   - Off-line K/P memory test
3. Place the Secure/Enable switch in the secure position.
4. Boot the HSC with the system media by pressing and releasing the Init switch.
5. Bring the HSC on line by pressing and releasing the Online switch.
6. Use the SETSHO command SHOW VIRTUAL_CIRCUITS to ensure that both A and B paths are present to all hosts.
7. Use the SETSHO command SHOW CI to verify the absence of the RTNDAT/DISC datagram.

3.5.2 Removing and Replacing Port Buffer Module (PILA)

The port buffer module (PILA) is a part of the host interface module set (K.ci). The PILA interfaces with the port link module through a set of interconnect link signals.

3.5.2.1 Removing the PILA Module

Use the following procedure to remove the PILA module. Observe safety and ESD precautions before starting the module removal procedure.

1. Notify users that the HSC is being taken off line and the drives attached to it will not be available.
2. Dismount or failover any drives connected to the HSC or shut down the system.

3. Set the dc power switch to the 0 (off) position.

   On the HSC, the dc power switch is located on the side of the RX33 housing.

   On the HSC50, the dc power switch is located on the maintenance access panel.

4. Turn the two nylon latches on the module cover plate one-quarter turn.
5. Pull the card cage cover up and out.
6. Locate the PILA in slot number 13 of the card cage. This can be verified by the module utilization label.
7. Remove the PILA module.

3.5.2.2 Setting the Replacement PILA Module Switches

The PILA module has factory-set dual in-line pack (DIP) switches to indicate the hardware revision level. Do not change DIP switch positions, except as directed by a field change order (FCO).

Figure 3–16 shows the PILA module switch.

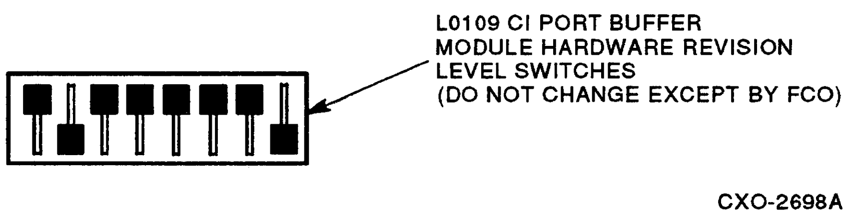


Figure 3–16 L0109 Hardware Rev Level Switch

3.5.2.3 Replacing the PILA Module

This section provides the PILA module replacement procedure. Observe safety and ESD precautions before starting the module replacement procedure.

1. Install the PILA module in slot number 13 of the card cage. This can be verified by the module utilization label.
2. Pull the card cage cover down and in.
3. Turn the two nylon latches on the module cover plate one-quarter turn.
4. Set the dc power switch to the 1 (on) position.

   On the HSC, the dc power switch is located on the side of the RX33 housing.

   On the HSC50, the dc power switch is located on the maintenance access panel.

3.5.2.4 Testing the PILA Module

Perform the following tests to verify correct PILA operation as part of the K.ci host interface module set.

1. Boot the HSC with the off-line diagnostic media. Refer to Chapter 6 for boot procedures.

   NOTE
   The off-line diskette must be write protected. Place a write-protect tab over the diskette write-enable notch.

2. Refer to Chapter 6 for test descriptions and procedures and run the following tests:
   - Off-line bus interaction test

- Off-line K test selector test
- Off-line K/P memory test

3. Place the Secure/Enable switch in the secure position.
4. Boot the HSC with the system media by pressing and releasing the Init switch.
5. Bring the HSC on line by pressing and releasing the Online switch.
6. Use the SETSHO command SHOW VIRTUAL_CIRCUITS to ensure that both A and B paths are present to all hosts.

3.5.3 Removing and Replacing the Port Processor Module (K.pli)

The port processor module (K.pli) is a part of the host interface module set (K.ci). The K.pli interfaces with the port link module indirectly through the post buffer module.

3.5.3.1 Removing the K.pli Module

Use the following procedure to remove the K.pli module. Observe safety and ESD precautions before starting the module removal procedure.

1. Notify users that the HSC is being taken off line and the drives attached to it will not be available.
2. Dismount or failover any drives connected to the HSC or shut down the system.
3. Set the dc power switch to the 0 (off) position.

   On the HSC, the dc power switch is located on the side of the RX33 housing.

   On the HSC50, the dc power switch is located on the maintenance access panel.
4. Turn the two nylon latches on the module cover plate one-quarter turn.
5. Pull the HSC card cage cover up and out.
6. Locate the K.pli module in slot number 12 of the card cage. This can be verified by the module utilization label.
7. Remove the K.pli module.

3.5.3.2 Setting the Replacement K.pli Module Switches

The K.pli module has factory-set dual in-line pack (DIP) switches to indicate the hardware revision level. Do not change DIP switch positions, except as directed by a field change order (FCO).

Figure 3–17 shows the K.pli module switch.

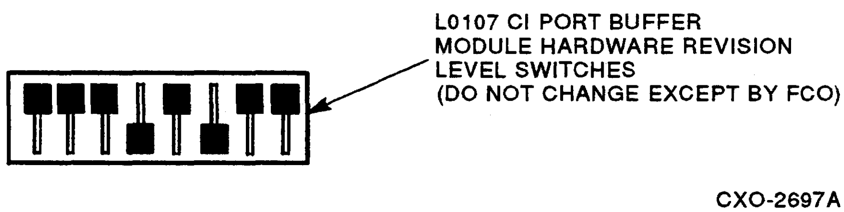


Figure 3–17 L0107 Hardware Rev Level Switch

3.5.3.3 Replacing the K.pli Module

This section provides the K.pli module replacement procedure. Observe safety and ESD precautions before starting the module replacement procedure.

1. Install the K.pli module in slot number 12 of the card cage. This can be verified by the module utilization label.
2. Replace the K.pli module.
3. Pull the card cage cover down and in.
4. Turn the two nylon latches on the module cover plate one-quarter turn.
5. Set the dc power switch to the 1 (on) position.

   On the HSC, the dc power switch is located on the side of the RX33 housing.

   On the HSC50, the dc power switch is located on the maintenance access panel.

3.5.3.4 Testing the K.pli Module

Run the following tests to verify correct K.pli operation as part of the K.ci host interface module set.

1. Boot the HSC with the off-line diagnostic media. Refer to Chapter 6 for boot procedures.

   NOTE
   The off-line diskette must be write protected. Place a write-protect tab over the diskette write-enable notch.

2. Refer to Chapter 6 for test descriptions and procedures and run the following tests:
   - Off-line bus interaction test
   - Off-line K test selector test
   - Off-line K/P memory test
3. Place the Secure/Enable switch in the secure position.
4. Boot the HSC with the system media by pressing and releasing the Init switch.
5. Bring the HSC on line by pressing and releasing the Online switch.
6. Use the SETSHO command SHOW VIRTUAL_CIRCUITS to and ensure both A and B paths are present to all hosts.

3.5.4 Removing and Replacing the Disk Data Channel Module (K.sdi)

The K.sdi data channel interfaces between the HSC and the standard disk interface (SDI). K.sdi operation is controlled by an on-board microprocessor with a local programmed read-only memory (PROM). Commands and responses pass between the K.sdi microprocessor and other internal HSC processes through Control memory.

3.5.4.1 Removing the K.sdi Module

Use the following procedure to remove the K.sdi module. Observe safety and ESD precautions before starting the module removal procedure.

1. Notify users that the HSC is being taken off line and the drives attached to it will not be available.
2. Dismount or failover any drives connected to the HSC.
3. Set the dc power switch to the 0 (off) position.

   On the HSC, the dc power switch is located on the side of the RX33 housing.

On the HSC50, the dc power switch is located on the maintenance access panel.

4. Turn the two nylon latches on the module cover plate one-quarter turn.
5. Pull the card cage cover up and out.
6. Check the module utilization label above the card cage for the location of the K.sdi module. The module slots are numbered from right to left when viewed from the front.
7. Remove the K.sdi module.

3.5.4.2 Replacing the K.sdi Module

This section provides the K.sdi module replacement procedure. Observe safety and ESD precautions before starting the module replacement procedure.

1. Check the module utilization label above the card cage for the location of the module. The module slots are numbered from right to left when viewed from the front.
2. Replace the K.sdi module.
3. Pull the card cage cover down and in.
4. Turn the two nylon latches on the module cover plate one-quarter turn.
5. Set the dc power switch to the 1 (on) position.

   On the HSC, the dc power switch is located on the side of the RX33 housing.

   On the HSC50, the dc power switch is located on the maintenance access panel.

3.5.4.3 Testing the K.sdi Module

Perform the following tests to verify correct K.sdi operation:

1. Boot the HSC with the off-line diagnostic media. Refer to Chapter 6 for boot procedures.

   NOTE
   The off-line diskette must be write protected. Place a write-protect tab over the diskette write-enable notch.

2. Refer to Chapter 6 for test descriptions and procedures and run the following tests:
   - Off-line bus interaction test
   - Off-line K test selector test
   - Off-line K/P memory test
3. Place the Secure/Enable switch in the secure position.
4. Boot the HSC with the system media by pressing and releasing the Init switch.
5. Bring the HSC on line by pressing and releasing the the Online switch.
6. Use the SETSHO command SHOW VIRTUAL_CIRCUITS to ensure both A and B paths are present to all hosts.
7. Use the SETSHO command SHOW DISK to verify that all applicable drives are present.
8. Perform the disk drive integrity test ILDISK. Refer to Chapter 5 for test description and procedure.

3.5.5 Tape Data Channel Module (K.sti)

The K.sti is an interface between the HSC and the standard tape interface (STI). K.sti operation is controlled by an on board microprocessor with a local programmed read-only memory (PROM). Commands and responses pass between the K.sti microprocessor and other internal HSC processes through control memory.

3.5.5.1 Removing the K.sti Module

Use the following procedure to remove the K.sti module. Observe safety and ESD precautions before starting the module removal procedure.

1. Notify users that the HSC is being taken off line and the drives attached to it will not be available.
2. Dismount or failover any drives connected to the HSC.
3. Set the dc power switch to the 0 (off) position.

   On the HSC, the dc power switch is located on the side of the RX33 housing.

   On the HSC50, the dc power switch is located on the maintenance access panel.
4. Turn the two nylon latches on the module cover plate one-quarter turn.
5. Pull the card cage cover up and out.
6. Check the module utilization label above the card cage for the location of the module. The module slots are numbered from right to left when viewed from the front.
7. Remove the K.sti module.

3.5.5.2 Replacing the K.sti Module

This section provides the K.sti module replacement procedure. Observe safety and ESD precautions before starting the module replacement procedure.

1. Check the module utilization label above the card cage for the location of the module. The module slots are numbered from right to left when viewed from the front.
2. Replace the K.sti module.
3. Pull the card cage cover down and in.
4. Turn the two nylon latches on the module cover plate one-quarter turn.
5. Set the dc power switch to the 1 (on) position.

   On the HSC, the dc power switch is located on the side of the RX33 housing.

   On the HSC50, the dc power switch is located on the maintenance access panel.

3.5.5.3 Testing the K.sti Module

Perform the following to verify correct K.sti operation:

1. Boot the HSC with the off-line diagnostic media. Refer to Chapter 6 for boot procedures.

   NOTE
   The off-line diskette must be write protected. Place a write-protect tab over the diskette write-enable notch.
2. Refer to Chapter 6 for test descriptions and procedures and run the following tests:
   - Off-line bus interaction test
   - Off-line K test selector test

- Off-line K/P memory test

3. Place the Secure/Enable switch in the secure position.
4. Boot the HSC with the system media by pressing and releasing the Init switch.
5. Bring the HSC on line by pressing and releasing the the Online switch.
6. Use the SETSHO command SHOW VIRTUAL_CIRCUITS to ensure both A and B paths are present to all hosts.
7. Run the tape drive integrity test ILTAPE. Refer to Chapter 5 for test description and procedure.

3.5.6 Removing and Replacing the Data Channel Module (K.si)

The K.si (L0119-YA) data channel module is a direct replacement for the K.sdi (L0108-YA) disk data channel or K.sti (L0108-YB) tape data channel modules.

NOTE
The K.si data channel initializes only with HSC Version 3.90 or higher software. Do not use the K.si module with HSC system software with a version level lower than 3.90. Versions lower than 3.90 will cause initialization failure.

3.5.6.1 Removing the K.si Module

Use the following procedure to remove the K.si module. Observe safety and ESD precautions before starting the module removal procedure.

1. Notify users that the HSC is being taken off line and the drives attached to it will not be available.
2. Dismount or failover any drives connected to the HSC.
3. Set the dc power switch to the 0 (off) position.

   On the HSC, the dc power switch is located on the side of the RX33 housing.

   On the HSC50, the dc power switch is located on the maintenance access panel.
4. Turn the two nylon latches on the module cover plate one-quarter turn.
5. Pull the card cage cover up and out.
6. Check the module utilization label above the card cage for the location of the module. The module slots are numbered from right to left when viewed from the front.
7. Remove the K.si, K.sdi, or K.sti module.

3.5.6.2 Setting the Replacement K.si Module Switches

The K.si has a switchpack containing four switches. Table 3–1 describes the names and functions of the switches. SW1 is on the top of the switchpack.

NOTE
The four switches must be in the OFF position to prevent errors during initialization and normal operation.

Table 3–1 K.si Switchpack Options

Switch Number	Switch Name	Function Description	Normal Position
SW1	MFG	Provides loop on error and single port external loop.	Off
SW2	Burn in	Continuous loop. Assumes external loop and clock.	Off
SW3	Ext loop	Loops on all ports; for manufacturing or field use.	Off
SW4	Ext clock	Substitutes external clock; for manufacturing use only.	Off

Figure 3–18 shows the K.si module switchpack.

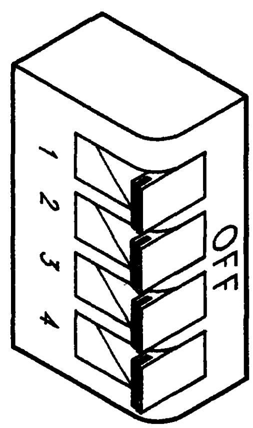


Figure 3–18 K.si Switchpack

3.5.6.3 Configuration of Requestors While Replacing the K.si Module

To obtain maximum performance from the HSC, physically configure the tape and disk requestors according to the following guidelines:

- Each K.si, K.sdi, or K.si module in the HSC backplane has a requestor priority number according to its backplane slot. You can greatly enhance the performance of the HSC if the attached devices are arranged according to their transfer speeds.
- Attach the slower devices to the lower priority requestors and the faster devices to the higher priority requestors. Do this while you have the HSC shut down for K.si installation.
- With the introduction of faster devices, such as the TA90 tape drive and RA90 disk drive, backplane configuration has become especially important. Failure to configure the devices properly could result in data bus overrun, EDC errors and, significant performance loss.

The following table shows the device relative speeds:

Table 3–2 Physical Configuration

Device	Relative Speed
TA90 tape drive	Fastest
RA90 disk drive	|
RA82 disk drive	|
RA81 disk drive	|
RA60 disk drive	|
RA70 disk drive	|
RA80 disk drive	|
All STI tape drives except TA90	Slowest

Requestor priority in the HSC ascends from Requestor 2 (lowest priority) through the highest requestor number on the HSC. Note that requestor priority levels are all relative; that is, the individual requestor numbers have no intrinsic speed characteristic. When configuring, it is recommended that you leave blank slots where possible to eliminate reconfiguration when adding requestors in the future.

NOTE
The lowest priority slot is located next to the K.ci module and the highest priority slot is located next to the P.ioj/c module.

3.5.6.4 Replacing the K.si Module

This section provides the K.si module replacement procedure. Observe safety and ESD precautions before starting the module replacement procedure.

1. Refer to Table 3–2 and verify that the requestors are arranged according to the transfer speeds of their attached devices.
2. Check the module utilization label above the card cage for the location of the module. The module slots are numbered from right to left when viewed from the front.
3. Replace an existing K.si, K.sdi, or K.sti module with the K.si. If you are installing a new requestor, install the K.si in the card cage. Attach the label accompanying the K.si module to the appropriate location of the module utilization label chart.
4. If you are planning to run the recommended external loop test, disconnect the SDI cables to the bulkhead of the requestor slot for the installed K.si. Install the loopback connectors (part number 70-22953-01), and refer to the section on K.si external loop tests.
5. Pull the card cage cover down and in.
6. Turn the two nylon latches on the module cover plate one-quarter turn.
7. Set the dc power switch to the 1 (on) position.

   On the HSC, the dc power switch is located on the side of the RX33 housing.

   On the HSC50, the dc power switch is located on the maintenance access panel.

3.5.6.5 K.si Module External Loop Test

Run the K.si External Loop Test before you initialize and configure the K.si to ensure that there are no hidden problems with the installation. Loopback connectors (part number 70-22953-01) are required for this test. These connectors are not included with the HSC.

1. Set SW3 of the K.si switch pack to the ON position. (SW1 is at the top of the switch pack.) All other switches on the switch pack must be set to OFF.
2. If you have not already done so, open the rear door and remove any SDI cables from the bulkhead of the requestor slot being tested. The K.si in slot 6, bulkhead E, is being tested in the example in this section.
3. Install four loopback connectors to the bulkhead.
4. Set the dc power switch to 1 to restore dc power to the card cage.
5. Press and release the Init switch on the operator control panel (OCP) to run the automatic loopback test as part of the powerup diagnostics. The following system messages appear:

```
INIPIO-I Booting...

    HSCxx Version V3.90 11-Feb-1988 17:00:41    System HSC009
```

6. If the HSC shows an error on boot, check that the loopback connectors are seated and that the K.si modules are fully seated in the backplane.
7. Remove the system media and insert the off-line diagnostic media into the load device.
8. Press and release the Init switch on the OCP.

   The load device drive-in-use LED should light within a few seconds, indicating the bootstrap is loading the off-line diagnostic loader to program memory.

   The off-line diagnostic loader indicates it has been loaded properly by displaying the following:

```
HSC OFL Diagnostic Loader, Version Vnnn
Radix=Octal,Data Length=Word,Reloc=00000000
ODL>
```

   The off-line loader is now ready to accept commands.

9. Issue the TEST K command and run microdiagnostics 11, 12, 13, and 14. The following example shows the diagnostics run for requestor 6. Perform these tests for each requestor installed in your system. The number of passes shown is for this example; you may perform more or less passes.

```
ODL> TEST K [RETURN]
```

   The test responds with:

```
HSC OFL K Test Selector
Requestor # of K (1 thru 9) [] ? 6 [RETURN]
Test # (1 thru 17) (O) [] ? 11 [RETURN]
# of passes to perform (D) [1] ? 10 [RETURN]
    End of Pass # 0000001, 00000 Errors, 00000 Total Errors
      .
      .
      .
    End of Pass # 0000010, 00000 Errors, 00000 Total Errors
```

```
Re-use parameters (Y/N) [Y] ?  N [RETURN]
Requestor # of K (1 thru 9) [] ? 6 [RETURN]
Test # (1 thru 17) (O) [] ? 12 [RETURN]
# of passes to perform (D) [1] ? 10 [RETURN]

    End of Pass # 0000001, 00000 Errors, 00000 Total Errors
      .
      .
      .
    End of Pass # 0000010, 00000 Errors, 00000 Total Errors

Re-use parameters (Y/N) [Y] ?  N [RETURN]
Requestor # of K (1 thru 9) [] ? 6 [RETURN]
Test # (1 thru 17) (O) [] ? 13 [RETURN]
# of passes to perform (D) [1] ? 5 [RETURN]

    End of Pass # 0000001, 00000 Errors, 00000 Total Errors
    End of Pass # 0000002, 00000 Errors, 00000 Total Errors
    End of Pass # 0000003, 00000 Errors, 00000 Total Errors
    End of Pass # 0000004, 00000 Errors, 00000 Total Errors
    End of Pass # 0000005, 00000 Errors, 00000 Total Errors

Re-use parameters (Y/N) [Y] ?  N [RETURN]
Requestor # of K (1 thru 9) [] ? 6 [RETURN]
Test # (1 thru 17) (O) [] ? 14 [RETURN]
# of passes to perform (D) [1] ? 5 [RETURN]

    End of Pass # 0000001, 00000 Errors, 00000 Total Errors
    End of Pass # 0000002, 00000 Errors, 00000 Total Errors
    End of Pass # 0000003, 00000 Errors, 00000 Total Errors
    End of Pass # 0000004, 00000 Errors, 00000 Total Errors
    End of Pass # 0000005, 00000 Errors, 00000 Total Errors

Re-use parameters (Y/N) [Y] ? N [RETURN]
```

The following output is an example of a loopback test failure message:

```
Requestor # of K (1 thru 9) [] ? 6 [RETURN]
Test # (1 thru 17) (O) [] ? 11 [RETURN]
# of passes to perform (D) [1] ? 5 [RETURN]

    OKTS>00:01 T#000 E#008 U-000
    O  K Timed-Out During Init
      K-Status = 370
         End of Pass # 0000001, 00001 Errors, 00001 Total Errors
```

10. If you receive a failure message, check that the loopback connectors are securely installed. Remove the K.si module and try it in a different slot. If these steps fail, replace the K.si. Be sure to run the tests again after fixing the fault.
11. Set SW3 of the K.si switchpack to the OFF position.
12. Install or replace all SDI cables on the HSC bulkhead.

3.5.6.6 Initializing the K.si Module

The K.si is by default a disk data channel after initial installation or on first boot of the HSC software. This default can result in a mismatch between the K.si configuration and the devices attached to the K.si. The mismatch results in a series of device errors printed on the HSC console, leading to the mismatched device(s) being declared inoperative by the HSC.

This mismatch can happen under the following conditions:

- After initial installation of the K.si and the attached devices are tape drives.
- After initial installation of HSC software.

- When replacing tape formatters with disk drives after the K.si has been configured for tape formatters.
- When replacing disk drives with tape formatters after the K.si has been configured for disk drives.

3.5.6.7 Correcting K.si Module Configuration Problems

To correct configuration problems, connect the data channel to the proper drive or use the following procedure to reconfigure the K.si:

1. Press the Init switch on the OCP. The HSC prints the following message to signify that initialization has started:

```
INIPIO-I Booting...

    HSCxx Version 3.90 11-Feb-1988 17:00:41    System HSC009
```

NOTE
The term HSCxx refers to the HSC model that is receiving the K.si module.

2. After initialization, use the SETSHO command SHOW REQUESTORS to show the status of the requestors. In the following example, the K.si modules are in requestors 2, 3, 4, 6, and 8; note that these modules show up as K.sdi modules.

```
CTRL/Y
HSCxx> SHOW REQUESTORS RETURN
Req    Status      Type           Version            Next Microcode Load
0     Enabled     P.ioc
1     Enabled     K.ci     MC- 43 DS- 2 Pila-0 K.pli-32
2     Enabled     K.sdi    MC- 2  DS- 4
3     Enabled     K.sdi    MC- 2  DS- 4
4     Enabled     K.sdi    MC- 2  DS- 4
5     Enabled     Empty
6     Enabled     K.sdi    MC- 2  DS- 4
7     Enabled     Empty
8     Enabled     K.sdi    MC-  2 DS- 4
9     Enabled     Empty
SETSHO-I Program Exit
```

3. Change the configuration of requestors 2 and 8 to tape data channels. Enter reconfiguring commands as shown, then re-initialize the system:

NOTE
Ensure the system load media is write enabled.

```
CTRL/Y
HSCxx> RUN SETSHO RETURN
SETSHO> ENABLE REBOOT RETURN
SETSHO-S The HSC will reboot on exit.
SETSHO> SET REQUESTOR 2/TYPE=TAPE RETURN
SETSHO> SET REQUESTOR 8/TYPE=TAPE RETURN
SETSHO> EXIT RETURN
SETSHO-Q Rebooting HSC, type Y to continue, CTRL/Y to abort: Y

INIPIO-I Booting...

    HSCxx Version 3.90 11-Feb-1988 17:00:41    System HSC009
```

This configuration is retained on the boot media to ensure that the K.si module comes up in the proper data channel configuration when the HSC is rebooted. If you have a K.si configured as a tape data channel and you attach disk drives, you can reconfigure it using the above procedure and specifying TYPE=DISK.

4. Use the SHOW REQUESTORS command again to current configuration of the K.si modules:

```
CTRL/Y
HSCxx> SHOW REQUESTORS [RETURN]
Req   Status      Type          Version            Next Microcode Load
0    Enabled    P.ioc
1    Enabled    K.ci     MC- 43 DS- 2 Pila-0 K.pli-32
2    Enabled    K.sti    MC- 2  DS- 3
3    Enabled    K.sdi    MC- 2  DS- 4
4    Enabled    K.sdi    MC- 2  DS- 4
5    Enabled    Empty
6    Enabled    K.sdi    MC-  2 DS- 4
7    Enabled    Empty
8    Enabled    K.sti    MC-2   DS- 3
9    Enabled    Empty
SETSHO-I Program Exit
```

3.5.6.8 K.si Module New Boot Microcode

When changing to a new boot media, you must power down the HSC to clear the old microcode from the K.si. A power-down interval of about 15 seconds is adequate. After powering up again, reboot and use the following procedure:

1. Use the SETSHO command SHOW REQUESTORS to check the existing configuration and verify necessary changes.

```
SET REQUESTOR n/TYPE=xxxx

where:

n is the requestor number
xxxx is the requestor type (DISK or TAPE)
```

2. Repeat the command SET REQUESTOR for each K.si in the HSC to set the HSC configuration on the new boot media.

Table 3–3 describes the conditions that determine if new K.si microcode is loaded.

Table 3–3 K.si New Microcode Load Conditions

Action	New Microcode Load
Using the ENABLE REBOOT and EXIT commands after using the SET REQUESTOR n/TYPE=xxxx command.	Yes.
Booting the system after a total power failure or after powering down the HSC.	Yes.
Changing requestor modules. (This action requires powering down.)	Yes.
Using the SET REQUESTOR n/TYPE=xxxx command.	No.
Changing to new boot media.	No—The new boot media must be updated with the SETSHO command SET REQUESTOR n/TYPE=xxxx and the HSC must be rebooted.

Table 3–3 (Cont.) K.si New Microcode Load Conditions

Action	New Microcode Load
Using the SET SCT CLEAR command.	No—The new boot media must be updated with the SETSHO command SET REQUESTOR n/TYPE=xxxx and the HSC must be rebooted.
Holding in the Fault button while pushing in the Init switch. (This action clears the SCT.)	No—The boot media must be updated with the SETSHO command SET REQUESTOR n/TYPE=xxxx and the HSC must be rebooted.

3.5.6.9 Testing the K.si Module (After Initialization)

The K.si module is one of the K requestor interface modules. Perform the following to verify correct K.si operation:

1. Boot the HSC with the off-line diagnostic media. Refer to Chapter 6 for boot procedures.

   NOTE
   The off-line diskette must be write protected. Place a write-protect tab over the diskette write-enable notch.

2. Refer to Chapter 6 for test descriptions and procedures and run the following tests:
   - Off-line bus interaction test
   - Off-line K test selector test
   - Off-line K/P memory test
3. Place the Secure/Enable switch in the secure position.
4. Boot the HSC with the system media by pressing and releasing the Init switch.
5. Bring the HSC on line by pressing and releasing the the Online switch.
6. Use the SETSHO command SHOW VIRTUAL_CIRCUITS to ensure both A and B paths are present to all hosts.
7. If the K.si is configured as a disk data channel, run the disk drive integrity test ILDISK. Refer to Chapter 5 for test description and procedure.
8. If the K.si is configured as a tape data channel, run the tape drive integrity test ILTAPE. Refer to Chapter 5 for test description and procedure.

3.5.7 Removing and Replacing the I/O Control Processor Module (P.ioj/c)

The P.ioj module (L0111/L0111-YA) uses a PDP-11 ISP (J-11) processor. The P.ioc module (L0105) uses a PDP-11 ISP (F-11) processor. Both contain memory management and memory interfacing logic. These processors execute their respective HSC internal software.

3.5.7.1 Removing the P.ioj/c Module

Use the following procedure to remove the P.ioj/c module. Observe safety and ESD precautions before starting the module removal procedure.

1. Press CTRL/Y to get the HSC> prompt. Use the SETSHO command SHOW SYSTEM and save the printout for reference.
2. Notify users that the HSC is being taken off line and the drives attached to it will not be available.
3. Dismount or failover any drives connected to the HSC.

4. Set the dc power switch to the 0 (off) position.

   On the HSC, the dc power switch is located on the side of the RX33 housing.

   On the HSC50, the dc power switch is located on the maintenance access panel.
5. Turn the two nylon latches on the module cover plate one-quarter turn.
6. Pull the card cage cover up and out.
7. Locate the P.ioj/c module in slot number 1 of the card cage. This can be verified by the module utilization label.
8. Remove the P.ioj/c module.

3.5.7.2 Setting the Replacement P.ioj/c Module Jumpers

The P.ioj/c modules have factory-set jumpers. Each module has a unique serial number that matches the pattern of the jumpers. *Do not reconfigure these jumpers.*

The P.ioc (L0105) also has a baud rate jumper. When the P.ioc is replaced, set the baud rate jumper W3 to match the console terminal or duplicate the jumper configuration of the P.ioc being replaced.

Figure 3–19 shows the P.ioc module jumper location.

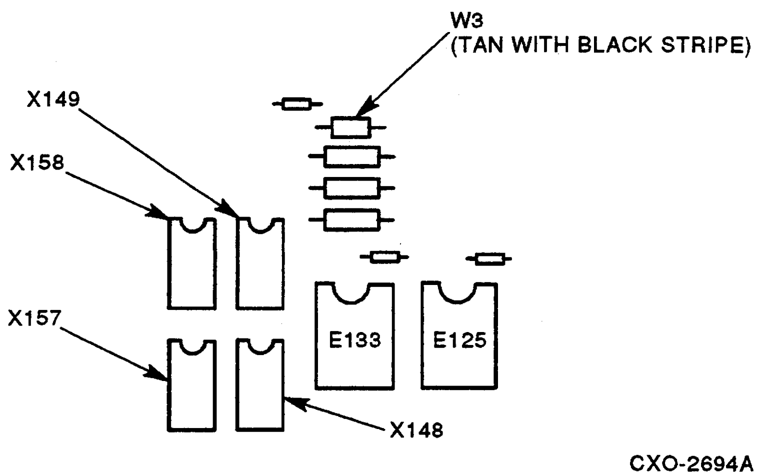


Figure 3–19 L0105 Baud Rate Jumper

Set the baud rate jumper as follows:

For 9600 baud, leave jumper W3 intact.
For 300 baud, cut one elbow angle on the W3 lead and spread slightly to avoid contact with the cut edges.

3.5.7.3 Replacing the P.ioj/c Module

This section provides P.ioj/c module replacement procedure. Observe safety and ESD precautions before starting module replacement procedures.

1. Install the P.ioj/c module in slot number 1 of the card cage. This can be verified by the module utilization label.
2. Pull the card cage cover down and in.
3. Turn the two nylon latches on the module cover plate one-quarter turn.
4. Set the dc power switch to the 1 (on) position.

On the HSC, the dc power switch is located on the side of the RX33 housing.

On the HSC50, the dc power switch is located on the maintenance access panel.

NOTE
Once VAX/VMS recognizes an HSC, the HSC's Online indicator may show that the HSC is alternately going on line and off line. This is because there is a discrepancy between the nodename or ID of the HSC and the one recognized by the host for that HSC. The HSC will be allowed to function only if the old nodename is equal to the new nodename and the old ID is equal to the new ID.

5. Press CTRL/Y to get the HSC> prompt, and issue the SETSHO command SHOW SYSTEM. Compare this printout with the one saved during removal.

   If the printout for the Nodename/ID is different, use the SETSHO commands SET NAME or SET ID to change the Nodename/ID so it is the same as in the saved printout.

3.5.7.4 Testing the P.ioj/c Module

When booting the HSC with the off-line diagnostic media, the P.ioj/c ROM bootstrap verifies the basic integrity of the P.ioj/c module.

Perform the following to verify correct P.ioj operation:

1. Boot the HSC with the off-line diagnostic media. Refer to Chapter 6 for boot procedures.

   NOTE
   The off-line diskette must be write protected. Place a write-protect tab over the diskette write-enable notch.

2. Refer to Chapter 6 for test descriptions and procedures and run the following tests:
   - Off-line cache test (P.ioj only)
   - Off-line bus interaction test
   - Off-line K test selector test
   - Off-line K/P memory test
3. Place the Secure/Enable switch in the secure position.
4. Boot the HSC with the system media by pressing and releasing the Init switch.
5. Bring the HSC on line by pressing and releasing the the Online switch.
6. Use the SETSHO command SHOW VIRTUAL_CIRCUITS to ensure both A and B paths are present to all hosts.

3.5.8 Removing and Replacing the HSC Memory Module (M.std2)

Memory module M.std2 (L0117) is used in the HSC only. It contains three independent systems memories, each residing on a different bus in the HSC. In addition, the memory module contains the RX33 diskette controller.

3.5.8.1 Removing the M.std2 Module

Use the following procedure to remove the M.std2 module. Observe safety and ESD precautions before starting the module removal procedure.

1. Notify users that the HSC is being taken off line and the drives attached to it will not be available.
2. Dismount or failover any drives connected to the HSC.

3. Set the dc power switch to the 0 (off) position.

   On the HSC, the dc power switch is located on the side of the RX33 housing.

4. Turn the two nylon latches on the module cover plate one-quarter turn.
5. Pull the card cage cover up and out.
6. Locate the M.std2 module in slot number 2 of the card cage. This can be verified by the module utilization label.
7. Remove the M.std2 module.

3.5.8.2 Replacing the M.std2 Module

This section provides the M.std2 module replacement procedure. Observe safety and ESD precautions before starting the module replacement procedure.

CAUTION
The switch pack on the M.std2 module is factory set to calibrate the RX33 diskette controller. Do not change the setting of this switch pack; the switch settings are unique to each module and cannot be restored outside of the manufacturing environment.

1. Install the M.std2 module in slot number 2 of the card cage. This can be verified by the module utilization label.
2. Replace the M.std2 module.
3. Pull the card cage cover down and in.
4. Turn the two nylon latches on the module cover plate one-quarter turn.
5. Set the dc power switch to the 1 (on) position.

   On the HSC, the dc power switch is located on the side of the RX33 housing.

6. Press CTRL/C to get the HSC> prompt.
7. Issue the SET MEMORY ENABLE ALL command.
8. After the HSC reboots, type the command SHOW MEMORY. Check that the available memory is equal to the maximum memory, except for 32 (decimal) words, which are disabled for lock functionality.

3.5.8.3 Testing the M.std2 Module

Perform the following to verify correct M.std2 operation:

1. Boot the HSC with the off-line diagnostic media. Refer to Chapter 6 for boot procedures.

   NOTE
   The off-line diskette must be write protected. Place a write-protect tab over the diskette write-enable notch.

2. Refer to Chapter 6 for test descriptions and procedures and run the following tests:
   - Off-line bus interaction test
   - Off-line K/P memory test
   - Off-line memory test
   - Off-line refresh test
3. Place the Secure/Enable switch in the secure position.
4. Boot the HSC with the system media by pressing and releasing the Init switch.

5. Bring the HSC on line by pressing and releasing the the Online switch.
6. Use the SETSHO command SHOW VIRTUAL_CIRCUITS and ensure both A and B paths are present to all hosts.
7. Run the memory integrity test ILMEMY and refer to Chapter 5 for test description and procedure.

3.5.9 Removing and Replacing the HSC50 Memory Module (M.std)

The HSC50 memory module (L0106) contains three separate and independent systems memories, each residing on a different bus within the HSC50.

3.5.9.1 Removing the M.std Module

Use the following procedure to remove the M.std module. Observe safety and ESD precautions before starting the module removal procedure.

1. Notify users that the HSC is being taken off line and the drives attached to it will not be available.
2. Dismount or failover any drives connected to the HSC.
3. Set the dc power switch to the 0 (off) position.

   On the HSC50, the dc power switch is located on the maintenance access panel.
4. Turn the two nylon latches on the module cover plate one-quarter turn.
5. Pull the card cage cover up and out.
6. Locate the M.std module in slot number 2 of the card cage. This can be verified by the module utilization label.
7. Remove the M.std module.

3.5.9.2 Replacing the M.std Module

This section provides the M.std module replacement procedure. Observe safety and ESD precautions before starting the module replacement procedure.

1. Install the M.std module in slot number 2 of the card cage. This can be verified by the module utilization label.
2. Replace the M.std module.
3. Pull the card cage cover down and in.
4. Turn the two nylon latches on the module cover plate one-quarter turn.
5. Set the dc power switch to the 1 (on) position.

   On the HSC50, the dc power switch is located on the maintenance access panel.
6. Press CTRL/C to get the HSC> prompt.
7. Issue SET MEMORY ENABLE ALL command.
8. After the HSC50 reboots, type the command SHOW MEMORY. Check that the available memory is equal to the maximum memory, except for 32 (decimal) words, which are disabled for lock functionality.

3.5.9.3 Testing the M.std Module

Run the following tests to verify correct M.std operation.

1. Boot the HSC with the off-line diagnostic media. Refer to Chapter 6 for boot procedures.

   NOTE
   The off-line diskette must be write protected. Place a write-protect tab over the diskette write-enable notch.

2. Refer to the Chapter 6 for test descriptions and procedures and run the following tests:
   - Off-line bus interaction test
   - Off-line K/P memory test
   - Off-line memory test
   - Off-line refresh test
3. Place the Secure/Enable switch in the secure position.
4. Boot the HSC with the system media by pressing and releasing the Init switch.
5. Bring the HSC on line by pressing and releasing the the Online switch.
6. Use the SETSHO command SHOW VIRTUAL_CIRCUITS to ensure both A and B paths are present to all hosts.
7. Run the memory integrity test ILMEMY and refer to Chapter 5 for test description and procedure.

3.6 Removing and Replacing Subunits

This section contains procedures for removing and replacing subunits.

WARNING:
Because hazardous voltages exist inside the HSC, service must be performed only by qualified people. Bodily injury or equipment damage can result from improper servicing procedures.

3.6.1 Removing and Replacing the RX33 Disk Drive

Two RX33 disk drives are used to load the HSC software or off-line diagnostics. The RX33 disk drives are mounted in the HSC cabinet. A cover plate ensures proper air flow and cooling. When removing and replacing the RX33, avoid snagging the cables attached to the rear of the drive. After replacing an RX33, always replace the cover plate.

3.6.1.1 Removing the RX33 Disk Drive

Use the following procedure to remove the RX33 disk drive:

1. Notify users that the HSC is being taken off line and the drives attached to it will not be available.
2. Dismount or failover any drives connected to the HSC.

3. Turn off the dc power switch, located on the side of the RX33 housing (Figure 3–20).

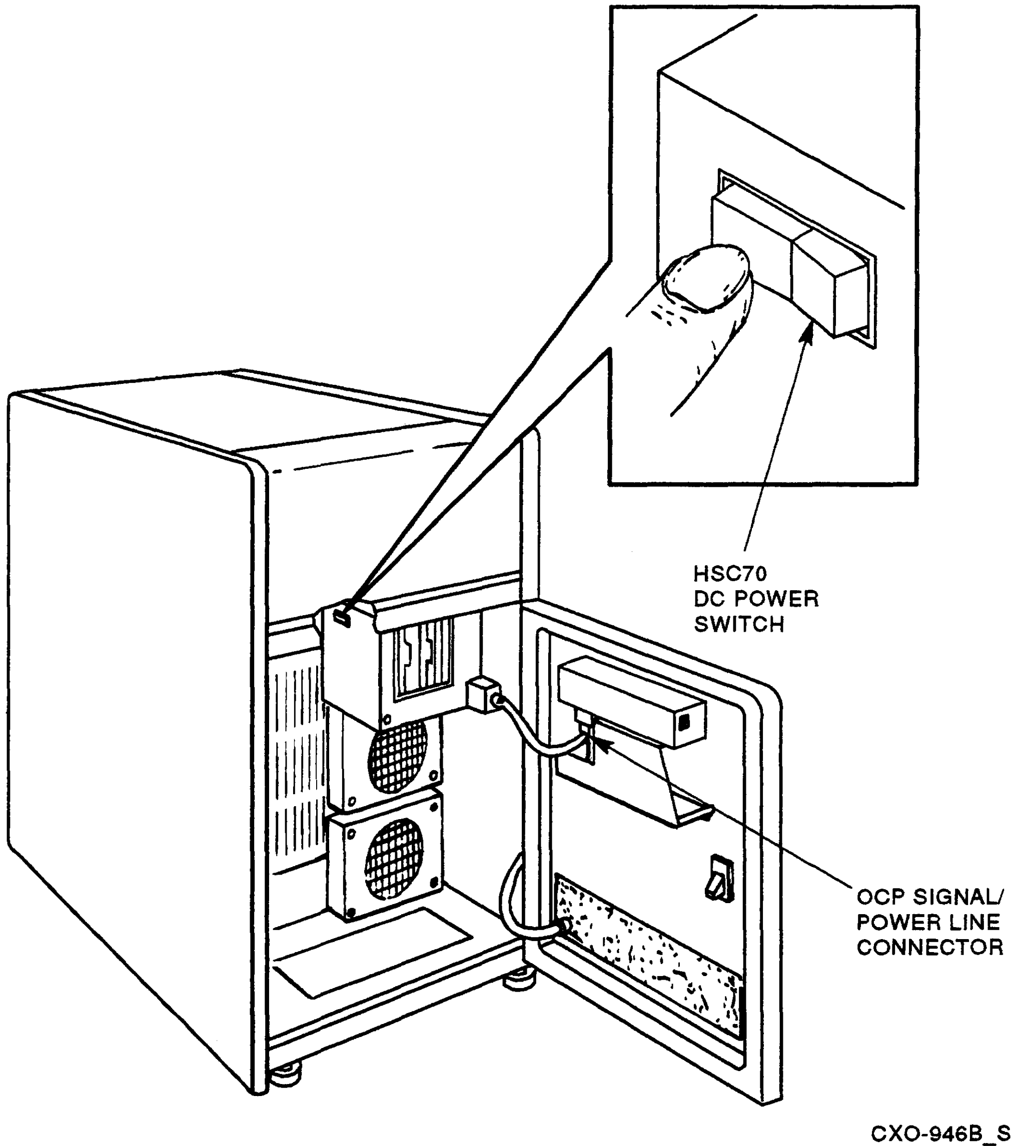


Figure 3–20 HSC DC Power Switch

4. Rotate the four fasteners on the RX33 cover plate one-quarter turn and remove the cover plate (Figure 3–21).

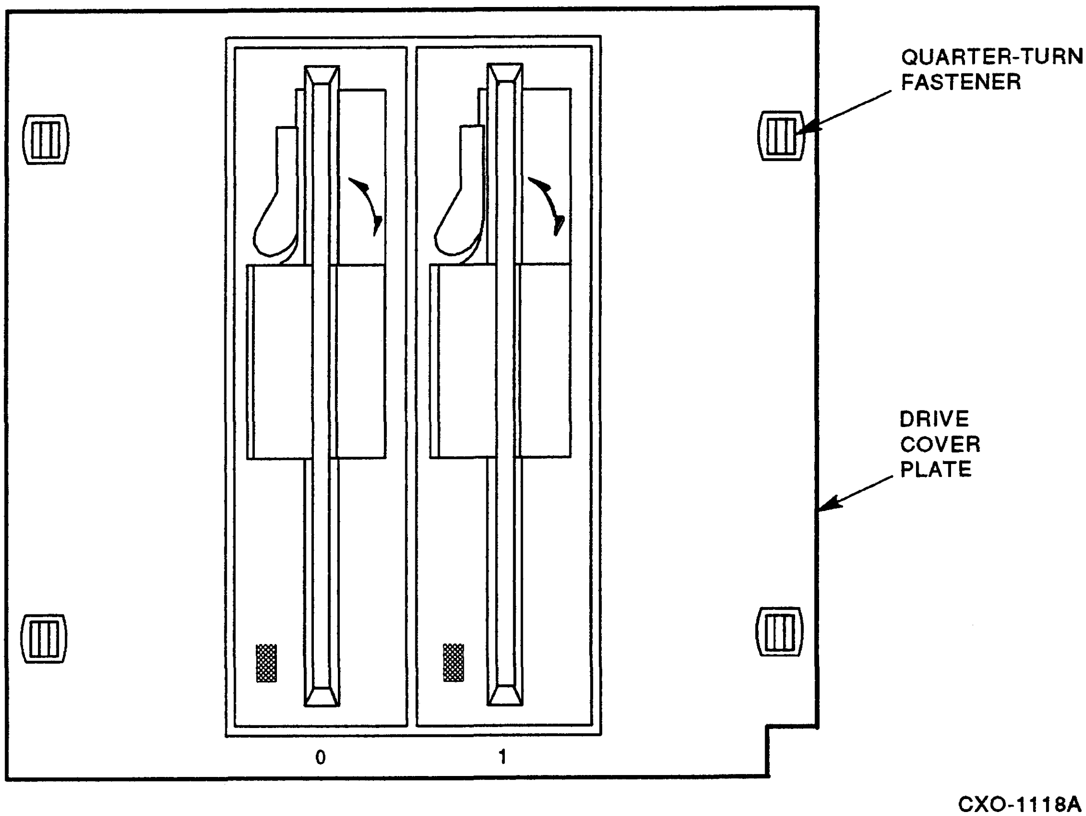


Figure 3–21 Removing the RX33 Cover Plate

5. 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.

6. Carefully slide the drive assembly out until the housing is cleared.
7. Support the drive assembly with one hand and remove the flat ribbon cables and power cables from the rear of the drives.
8. Determine whether drive 0 or drive 1 should be replaced.

9. Loosen the captive screws on the drive to be replaced and remove the drive from the drive assembly (Figure 3–22).

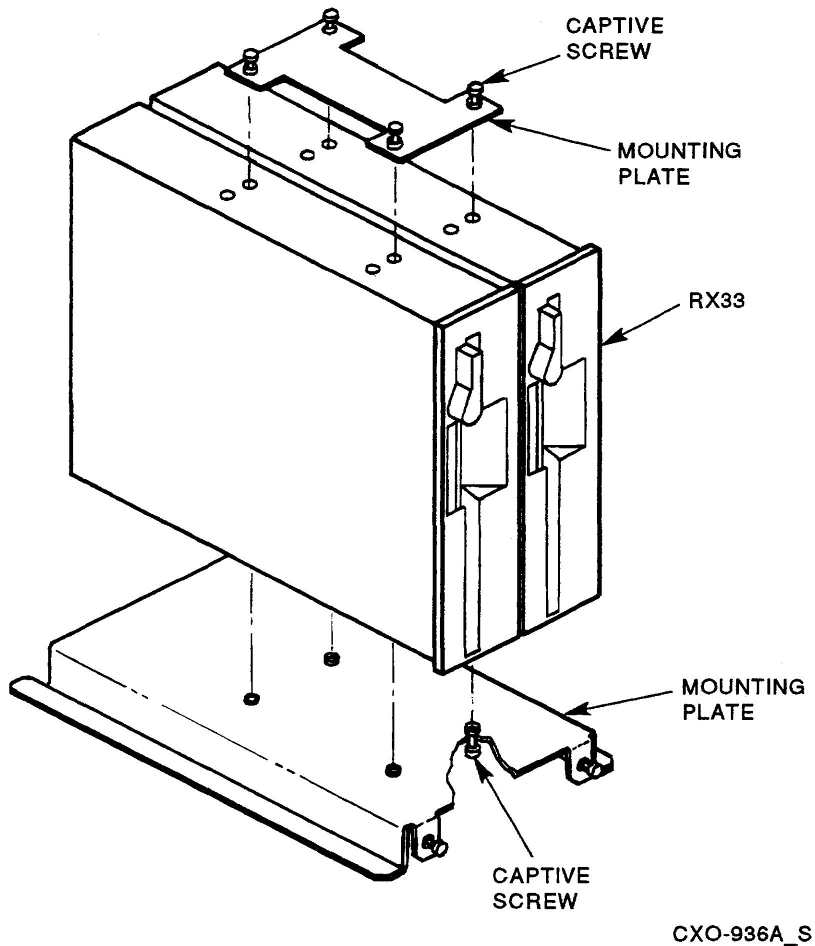


Figure 3–22 RX33 Disk Drive Removal

3.6.1.2 Setting the RX33 Disk Drive Jumpers

Replacement RX33 drives are not configured for the HSC. Two identical jumpers (part number 12-18783-00) must be added. If no extra jumpers are available, remove the jumpers from the defective drive. Correct jumper configuration is necessary for the operation of the replacement RX33 drive.

If replacing drive 0, be sure to insert jumper DS0. If replacing drive 1, be sure to insert jumper DS1.

The RX33 module may be revision A1 or A3. Table 3–4 shows the jumper differences and configurations for both revisions when the drive is used in an HSC.

Table 3–4 RX33 Jumper Description

Rev A1 Name	Rev A3 Name	Description	Status In or Out
FG	FG	Frame ground	In
HG		Hi gain	In
LG		Lo gain	Out
I	SI	Speed, mode 1 Dual speed	Out
II	II	Speed, mode 2 360 RPM only	In
DS0	D0	Drive select 0	In to select drive 0
DS1	D1	Drive select 1	In to select drive 1
DS2	D2	Drive select 2	Out
DS3	D3	Drive select 3	Out
U1	U0	Selects mode of operation for loading the heads and lighting the bezel LED (See note)	In
U2	U1	See U1/U0 above	In
HL	HL	Not applicable to HSC use	Out
IU	IU	Not applicable to HSC use	Out
	ML	Motor enable	Out
RE		Recalibration	Out
DC	DC1	Disk changed on pin 34	Out
	DC2	Factory setting	In
	DC3	Not applicable to HSC use	Out
	DC4	Not applicable to HSC use	Out
RY	RY	Ready on pin 34	In

NOTE

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

Figure 3–23 shows the jumper locations for RX33 with a revision A1 module.

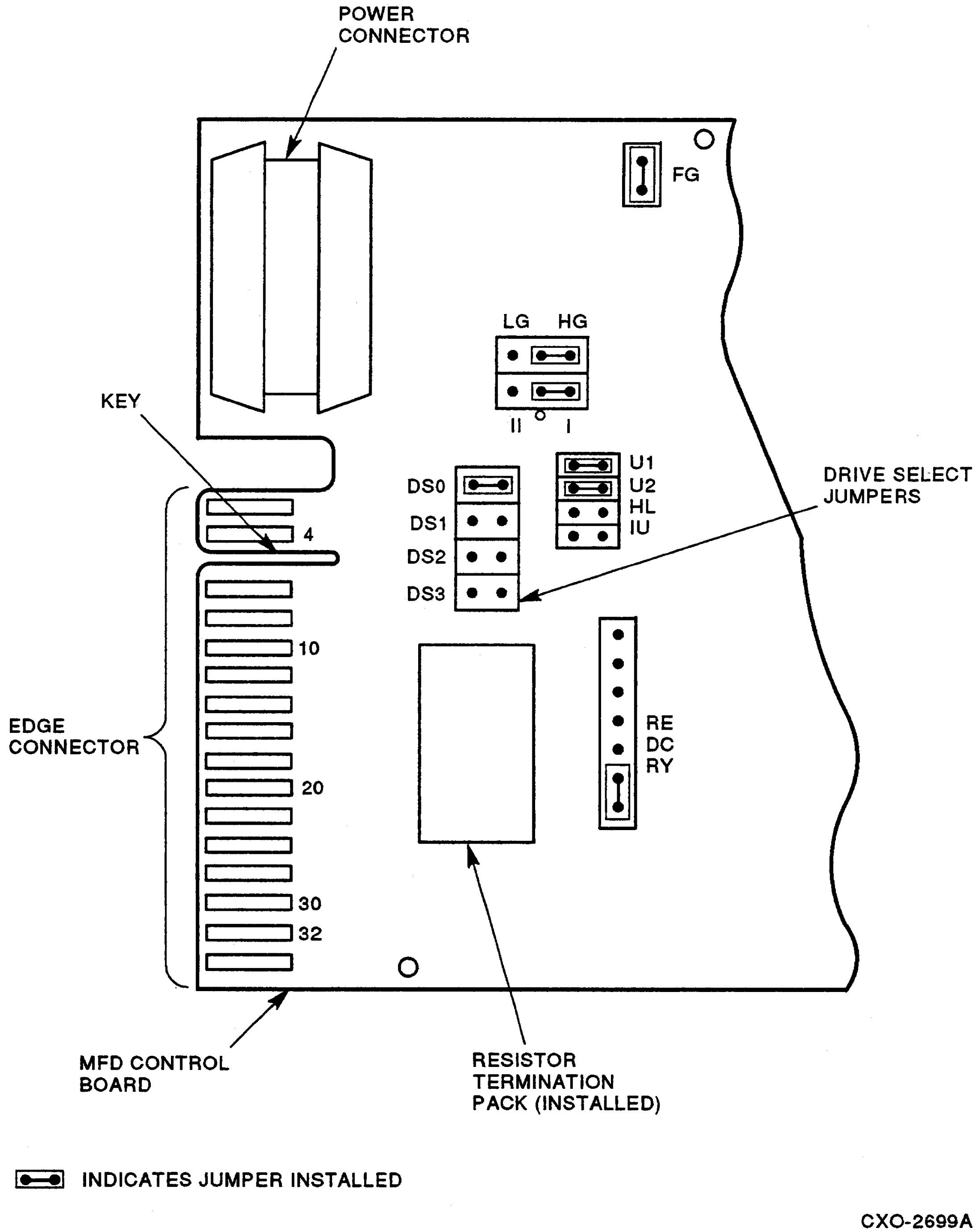


Figure 3–23 Revision A1 Jumper Configurations

Figure 3–24 shows the jumper locations for RX33 with a revision A3 module.

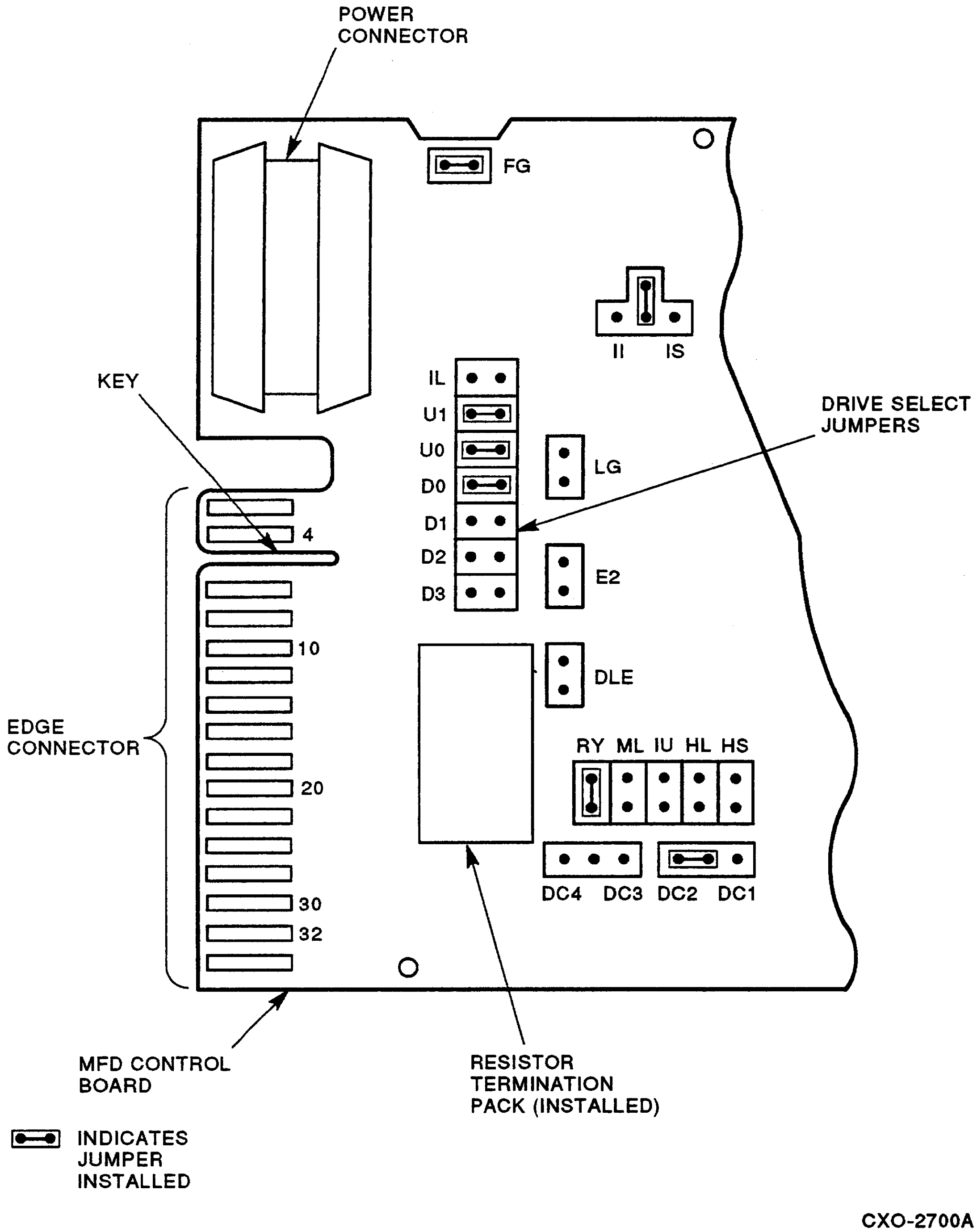


Figure 3–24 Revision A3 Jumper Configurations

3.6.1.3 Replacing the RX33 Disk Drive

Use the following procedure to replace the RX33 disk drive assembly:

1. Replace the drive in the drive assembly and tighten the drive captive screws.
2. Support the drive assembly with one hand and attach the flat ribbon cables and power cables to the rear of the drives.
3. Carefully slide the drive assembly into the drive housing.
4. Replace the cover plate and tighten the captive screws holding the drive assembly.
5. Replace the ac plug in the wall socket and place the main power switch on the power controller in the on position.
6. The drive-in-use LED lights while the drive is accessed and extinguishes after the drive access is completed or after the drive motor stops.

3.6.1.4 Testing the RX33 Disk Drive

After replacing the RX33, use the following procedure to test the drive:

1. Boot the HSC with the off-line diagnostic media. Refer to Chapter 6 for boot procedures.

   NOTE
   The off-line diskette must be write protected. Place a write-protect tab over the diskette write-enable notch.

2. Refer to Chapter 6 for a test description and procedure and run the RX33 off-line exerciser OFLRXE.
3. Place the Secure/Enable switch in the secure position.
4. Boot the HSC with the system media by pressing and releasing the Init switch.
5. Bring the HSC on line by pressing and releasing the the Online switch.
6. Use the SETSHO command SHOW VIRTUAL_CIRCUITS to ensure both A and B paths are present to all hosts.
7. Refer to Chapter 5 for test description and procedure and run the RX33 device integrity test ILRX33.

3.6.2 Removing and Replacing the TU58 Tape Drive

Two TU58 tape drives are used to load the HSC50 software or off-line diagnostics. The TU58 tape drives are mounted on the rear of the HSC50 front door.

CAUTION
When servicing the TU58, avoid bending the tachometer disk mounted on the drive motor shaft. If the disk is bent but not creased, it may be straightened. If it cannot be straightened or if it is creased, the TU58 must be replaced. The disk should not rub against the optical sensor block or dangling wires.

3.6.2.1 Removing the TU58 Tape Drive

Use the following procedure to remove the TU58 tape drive:

1. Notify users that the HSC is being taken off line and the drives attached to it will not be available.
2. Dismount or failover any drives connected to the HSC.
3. Remove the maintenance access panel cover by loosening the four captive screws.

4. Turn off the dc power switch (Figure 3–25).

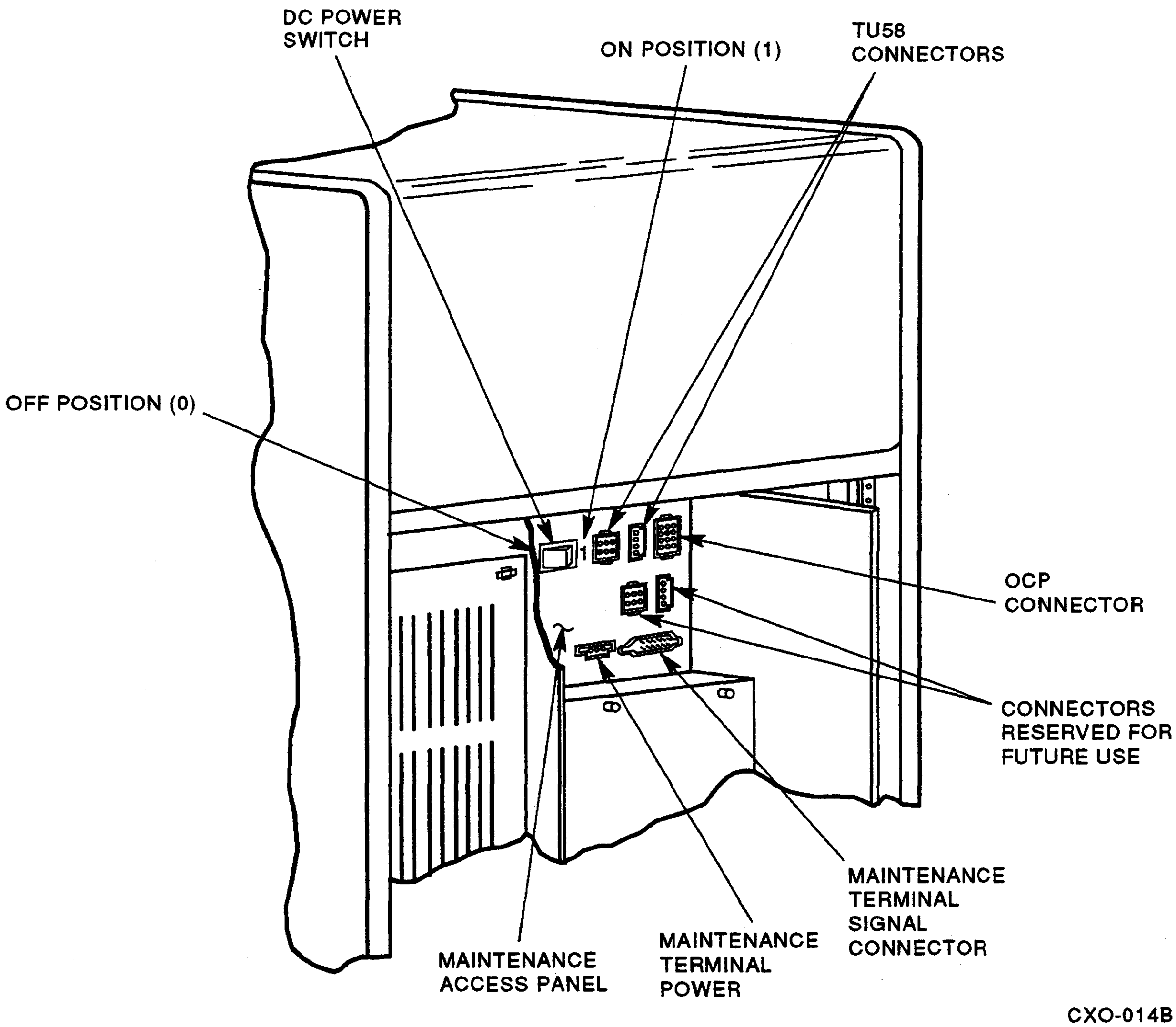


Figure 3–25 HSC50 DC Power Switch

5. Remove the two locknuts on the bottom of the TU58 bezel assembly (Figure 3–26).

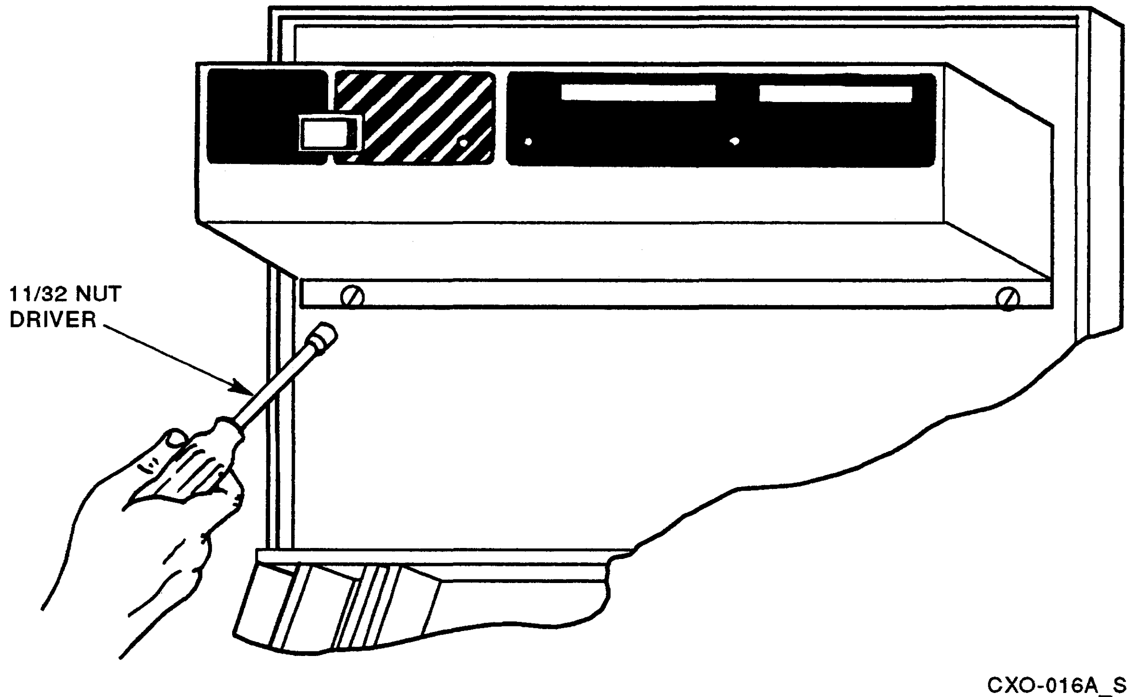


Figure 3–26 Removing the HSC50 TU58 Bezel Assembly

6. Push the bezel assembly up about 1 inch to clear the mounting hooks from their slots.
7. Pull the bezel assembly back 3 to 4 inches from the door for clearance.

8. Support the bezel assembly with one hand and disconnect J3 and J4 from the OCP (Figure 3–27).

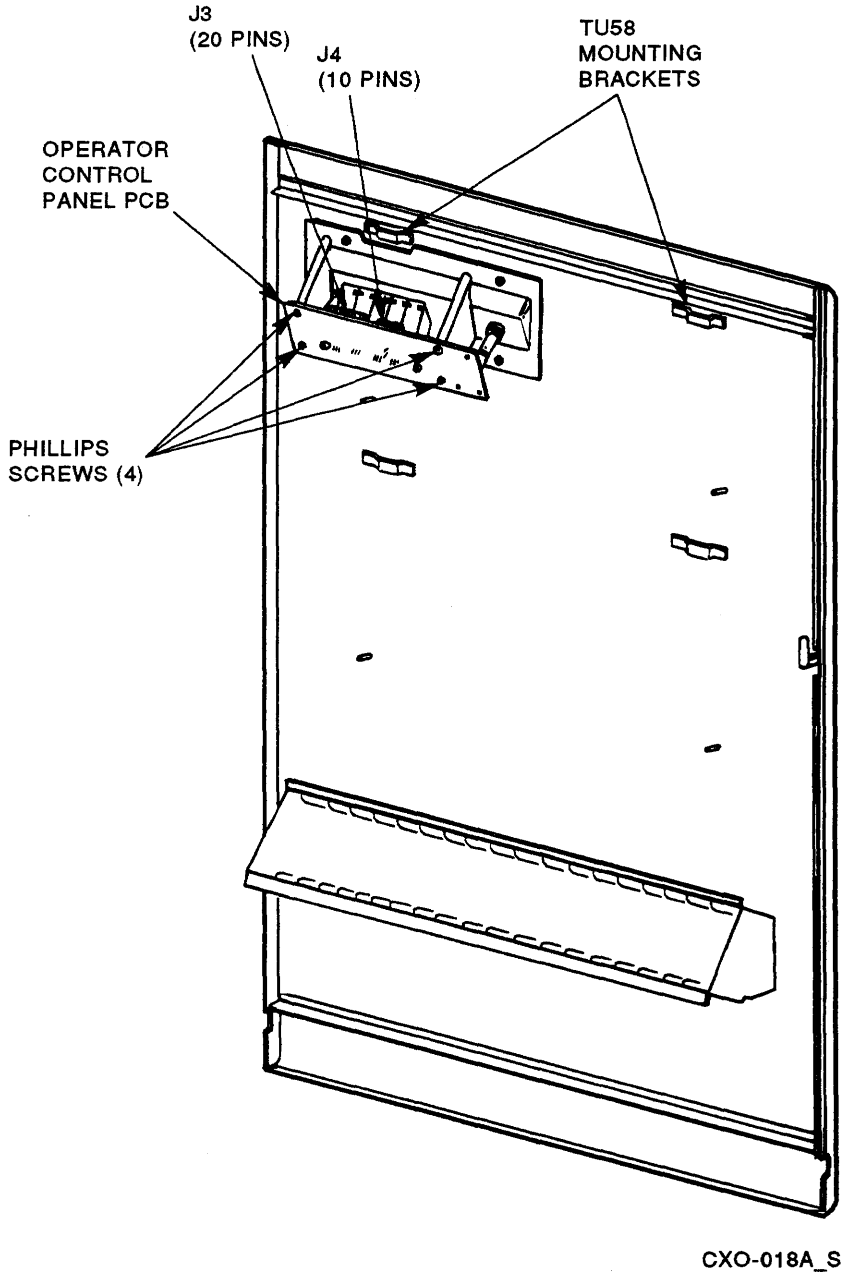


Figure 3–27 Disconnecting the HSC50 OCP Cables

9. Disconnect the cables from the TU58 controller module (Figure 3–28).

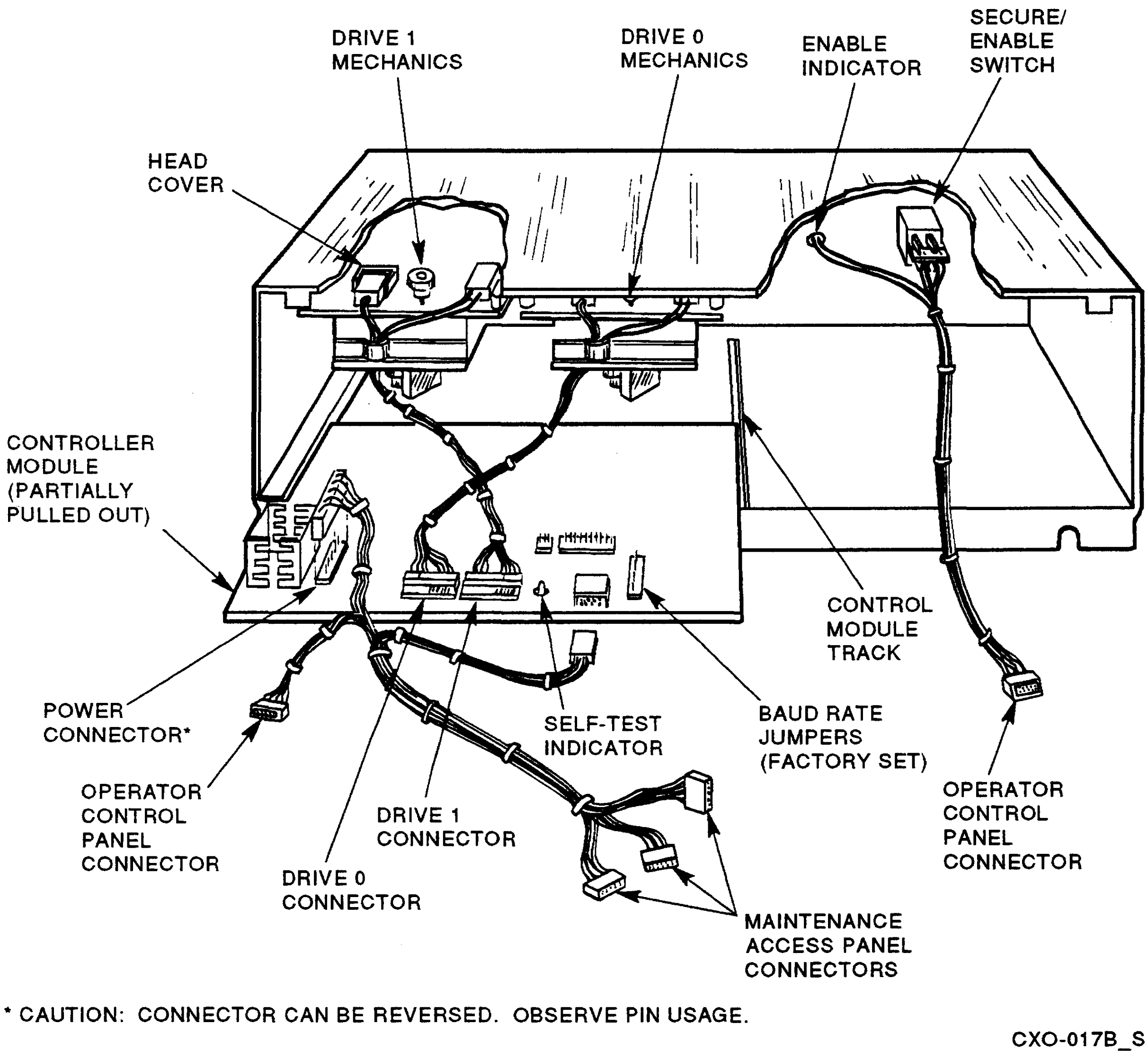


Figure 3–28 Disconnecting the HSC50 TU58 Controller Cables

NOTE
The head cover connector shown upper left in Figure 3–28 should be removed during operation.

10. Slide the TU58 controller module out of the plastic guides.

3.6.2.2 Setting the TU58 Tape Drive Jumpers

The TU58 baud rate jumpers are factory set. Ensure the baud rate jumper setting on the new module is the same as on the module being replaced.

Figure 3–29 shows the TU58 jumper location.

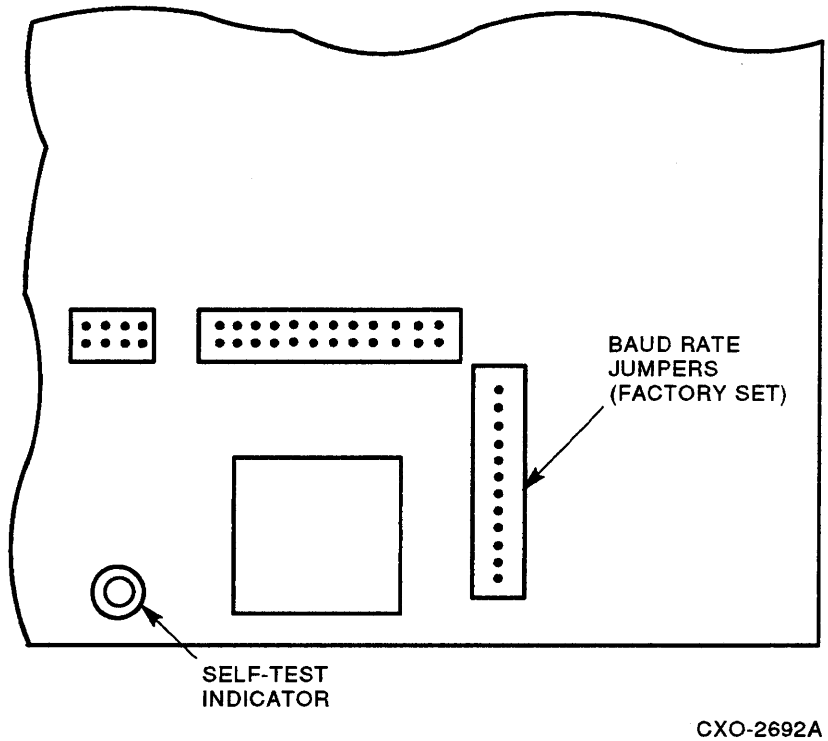


Figure 3–29 TU58 Baud Rate Jumpers

3.6.2.3 Replacing the TU58 Tape Drive

Following is the procedure for replacing the TU58 tape drive:

CAUTION

When servicing the TU58, avoid bending the tachometer disk mounted on the drive motor shaft. If the disk is bent but not creased, it may be straightened. If it cannot be straightened or if it is creased, the TU58 must be replaced. The disk should not rub against the optical sensor block or dangling wires.

1. Slide the controller module into the housing on the plastic guides.
2. Connect the cables to the TU58 controller module.
3. Support the bezel assembly with one hand and connect J3 and J4 to the OCP.
4. Attach the bezel assembly to the mounting hooks.
5. Replace the two locknuts on the bottom of the TU58 bezel assembly.
6. Replace the ac plug in the wall socket and place the main power switch on the power controller in the on position.

3.6.2.4 Testing the TU58 Tape Drive

After replacing the TU58, use the following procedure to test the drive:

1. Place the Secure/Enable switch in the secure position.
2. Boot the HSC with the system media by pressing and releasing the Init switch.
3. Bring the HSC on line by pressing and releasing the the Online switch.
4. Use the SETSHO command SHOW VIRTUAL_CIRCUITS and ensure both A and B paths are present to all hosts.
5. Run the TU58 device integrity test ILTU58. Refer to Chapter 5 for a description and procedure for this test.

3.6.3 Removing and Replacing the HSC Operator Control Panel (OCP)

If any OCP lamp fails, replace the entire OCP.

3.6.3.1 Removing the HSC OCP

Use the following procedure to remove the HSC OCP:

1. Open the front door by turning the key clockwise and lifting the latch.
2. Turn off the dc power switch located on the side of RX33 housing.
3. Remove the four Kepnuts securing the OCP shield to the studs on the front door.
4. Remove the OCP shield.

5. Remove the four screws securing the OCP to the shield (Figure 3–30).

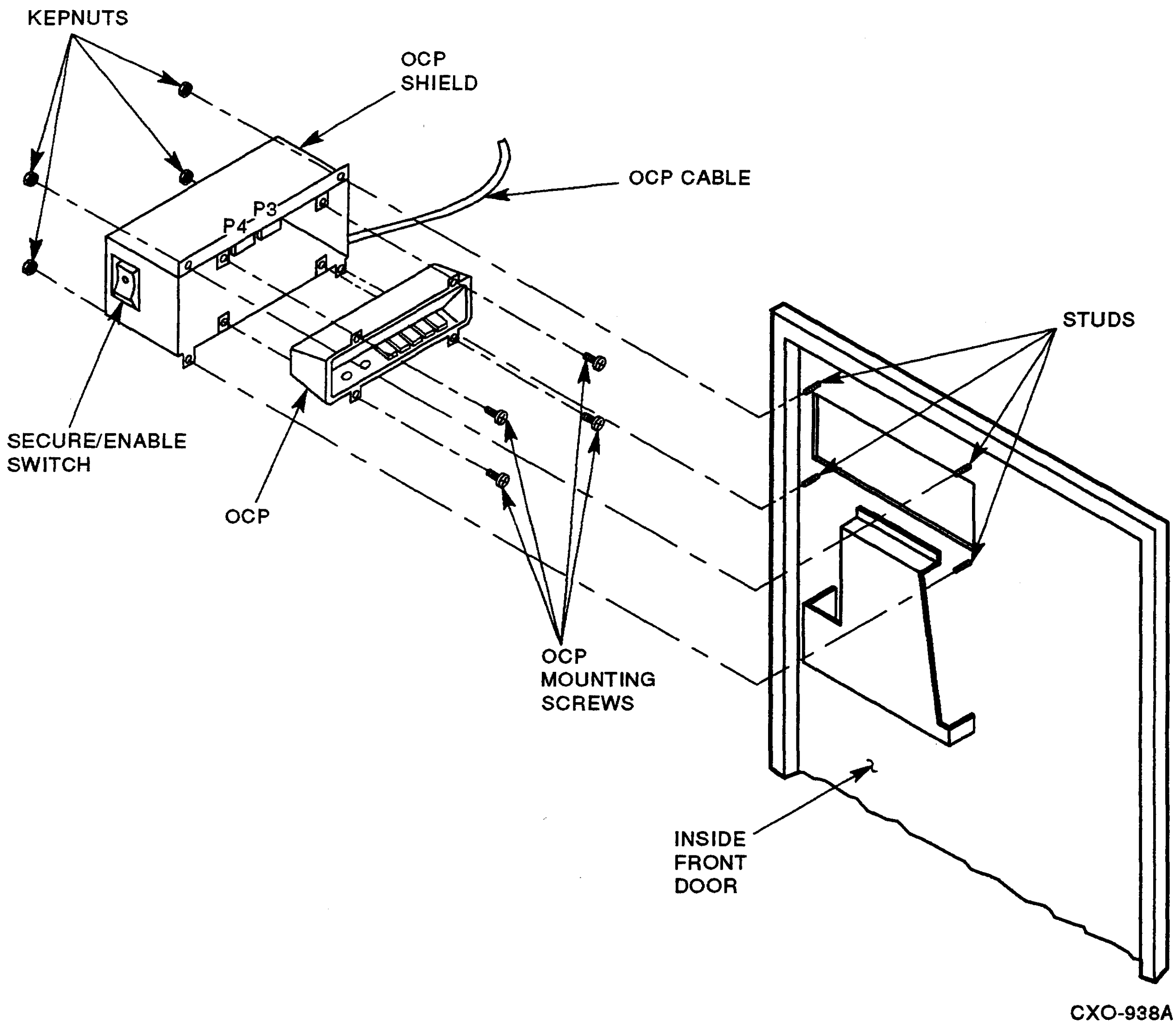


Figure 3–30 Removing the HSC OCP

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.

3.6.3.2 Replacing the HSC OCP

Following is the procedure for replacing the OCP:

1. Replace the two connectors from the printed circuit board on the OCP.
2. Secure the OCP to the shield using the four screws that were removed.
3. Replace the OCP shield.
4. Replace the four Kepnuts securing the OCP shield to the studs on the front door.
5. Turn on the dc power switch located on the side of RX33 housing.

6. Close and secure the front door by turning the key counter-clockwise.

3.6.3.3 Testing the HSC OCP

After replacement, use the following procedure to test the OCP:

1. Boot the HSC with the off-line diagnostic media. Refer to Chapter 6 for boot procedures.

   NOTE
   The off-line diskette must be write protected. Place a write-protect tab over the diskette write-enable notch.

2. Run the off-line OCP test. Refer to Chapter 6 for a test description and procedures.
3. Place the Secure/Enable switch in the secure position.
4. Boot the HSC with the system media by pressing and releasing the Init switch.
5. Bring the HSC on line by pressing and releasing the the Online switch.
6. Use the SETSHO command SHOW VIRTUAL_CIRCUITS to ensure both A and B paths are present to all hosts.

3.6.4 Removing and Replacing the HSC50 Operator Control Panel (OCP)

OCP indicators are not field replaceable. If any lamp fails, replace the entire OCP.

3.6.4.1 Removing the HSC50 OCP

Use the following procedure to replace the HSC50 OCP:

1. Open the front door by turning the key clockwise.
2. Remove dc power.
3. Remove the TU58s (Section 3.6.2).
4. Remove J3 and J4 from the OCP (Figure 3–31).
5. Remove the four screws from the OCP (Figure 3–31).
6. Carefully pull out the OCP, allowing for indicator and switch clearance.

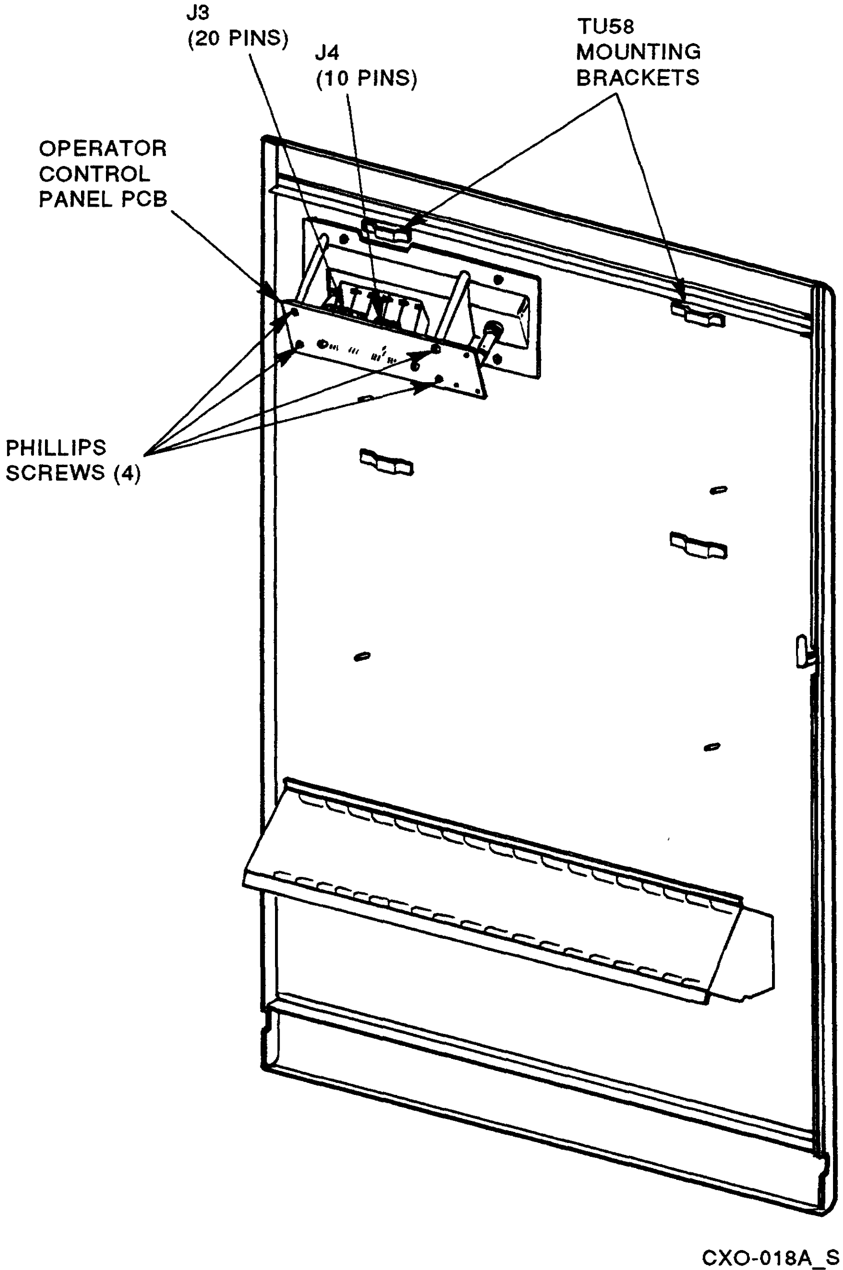


Figure 3–31 Removing the HSC50 OCP

3.6.4.2 Replacing the HSC50 OCP

Following is the procedure for replacing the OCP:

1. Carefully replace the OCP, allowing for indicator and switch clearance.
2. Replace the four OCP screws (refer to Figure 3–31).
3. Connect J3 and J4 to the OCP (refer to Figure 3–31).
4. Replace the TU58s (Section 3.6.2).
5. Turn on the dc power.
6. Close the front door by turning the key counterclockwise.

3.6.4.3 Testing the HSC50 OCP

After replacement, use the following procedure to test the OCP:

1. Boot the HSC with the off-line diagnostic media. Refer to Chapter 6 for boot procedures.

   NOTE
   The off-line diskette must be write protected. Place a write-protect tab over the diskette write-enable notch.

2. Run the off-line OCP test. Refer to Chapter 6 for a test description and procedure.
3. Place the Secure/Enable switch in the secure position.
4. Boot the HSC with the system media by pressing and releasing the Init switch.
5. Bring the HSC on line by pressing and releasing the the Online switch.
6. Use the SETSHO command SHOW VIRTUAL_CIRCUITS to ensure both A and B paths are present to all hosts.

3.6.5 Removing and Replacing the HSC Airflow Sensor Assembly

Use the following procedure to remove and replace the HSC Airflow Sensor:

1. Open the back door using a 5/32-inch hex wrench.
2. Turn off the ac circuit breaker (CB1) on the 881 power controller (Figure 3–32).
3. Disconnect J70 (Figure 3–33).
4. Remove the Phillips head screw that holds the mounting clamp to the duct (Figure 3–33).

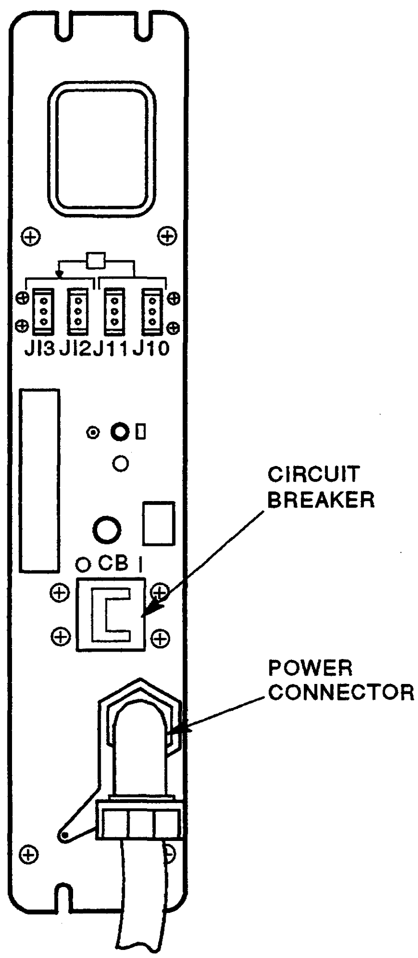


Figure 3–32 881 Power Controller Circuit Breaker

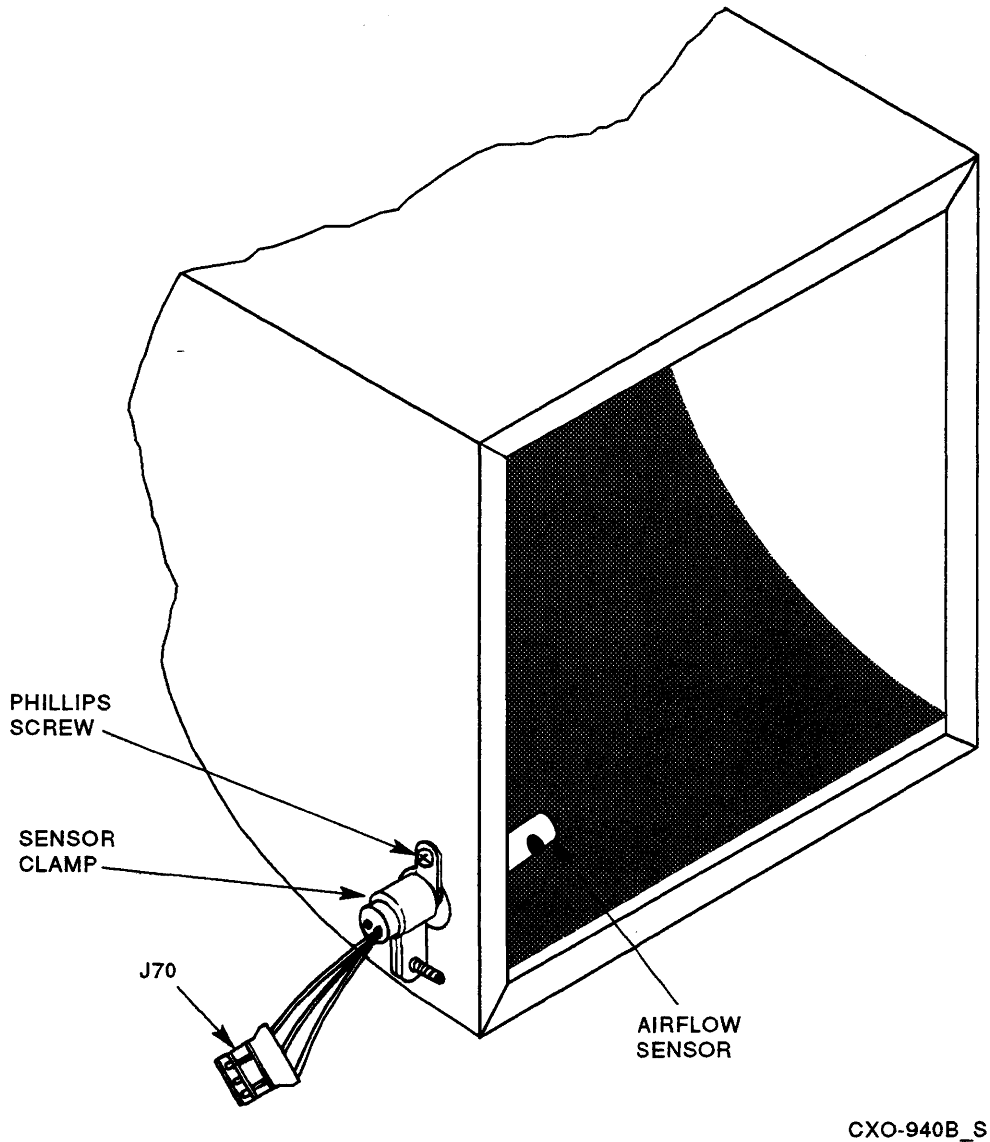


Figure 3–33 Removing and Replacing the HSC Airflow Sensor Assembly

5. Slide the sensor assembly out of the duct.

6. Reverse the removal procedure to replace the airflow sensor assembly. Align the slots in the airflow sensor tip horizontally with the floor. After turning on ac power to the HSC, test the new airflow sensor for proper operation by blocking the flow of air.

3.6.6 Removing and Replacing the HSC50 Airflow Sensor Assembly

Use the following procedure to remove and replace the HSC50 airflow sensor assembly:

1. Open the back door using a 5/32-inch hex wrench.
2. Turn off the ac circuit breaker (CB1) on the HSC50 power controller (Figure 3–34).

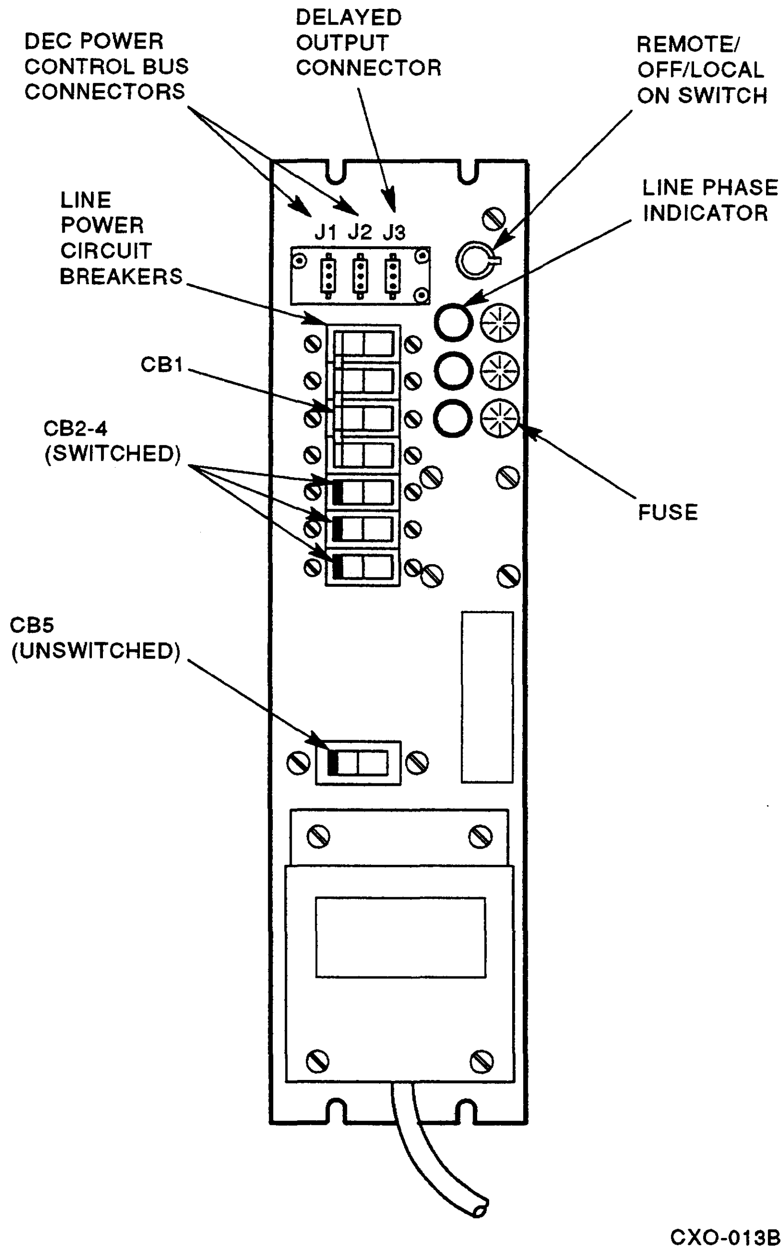


Figure 3–34 HSC50 Power Controller Circuit Breaker

3. Disconnect J70 (Figure 3–35).
4. Remove the Phillips head screw that holds the mounting clamp to the duct (Figure 3–35).

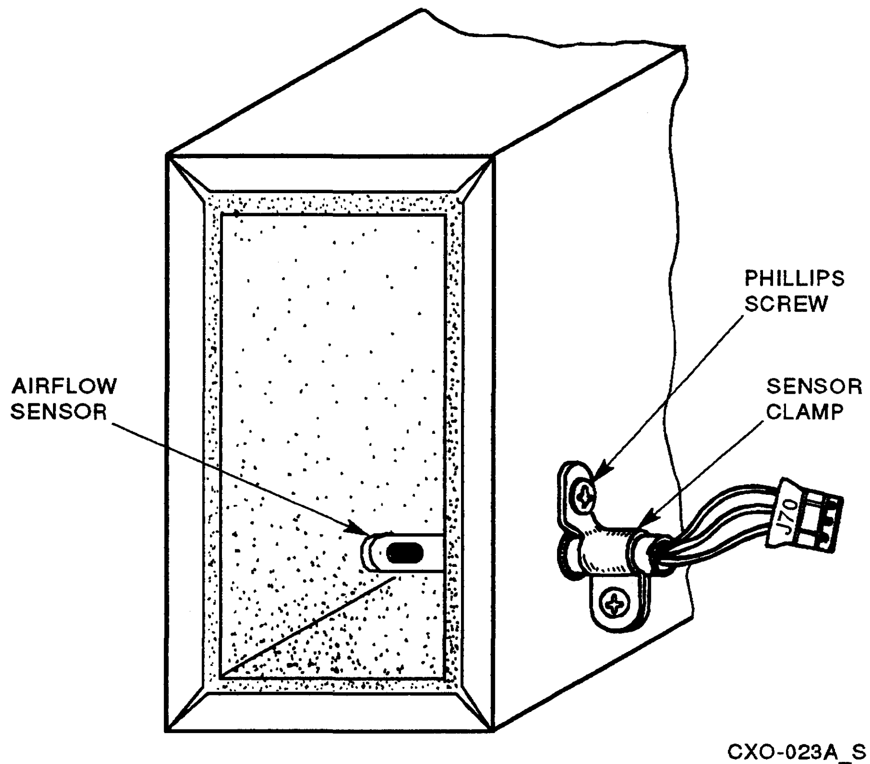


Figure 3–35 Removing and Replacing the HSC50 Airflow Sensor Assembly

5. Slide the sensor assembly out of the duct.
6. Reverse the removal procedure to replace the airflow sensor assembly. Align the slots in the airflow sensor tip horizontally with the floor. Ensure sensor operability by blocking the flow of air. Pinching the sensor should trip CB1.

3.6.7 Removing and Replacing the HSC Blower

The blower, which provides forced air cooling for the cabinet, is removed and replaced with the following procedure:

1. Open the back door using a 5/32-inch hex wrench.
2. Turn off the ac circuit breaker (CB1) on the 881 power controller (Figure 3–36).

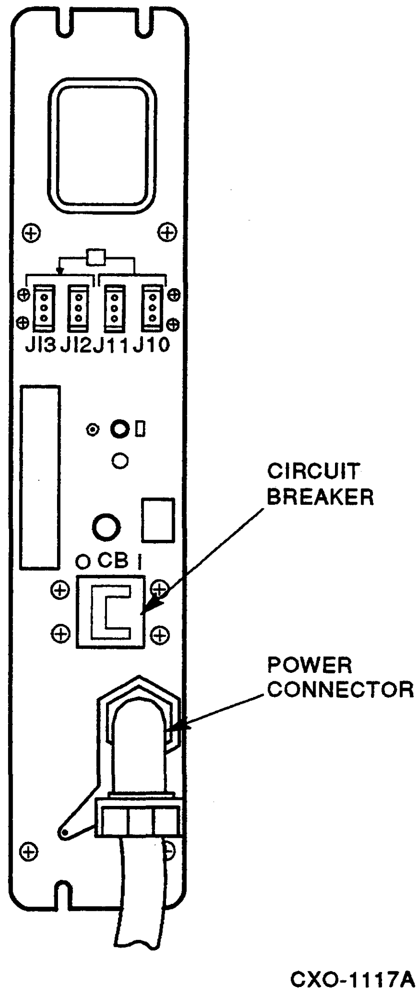


Figure 3–36 881 Power Controller Circuit Breaker

3. Disconnect the blower power connector (Figure 3–37).
4. Disconnect the airflow sensor power connector (J70) to allow removal of the exhaust duct (Figure 3–37).
5. Remove the exhaust duct from the bottom of the blower by lifting up the quick release latches on each side of the duct (Figure 3–37).

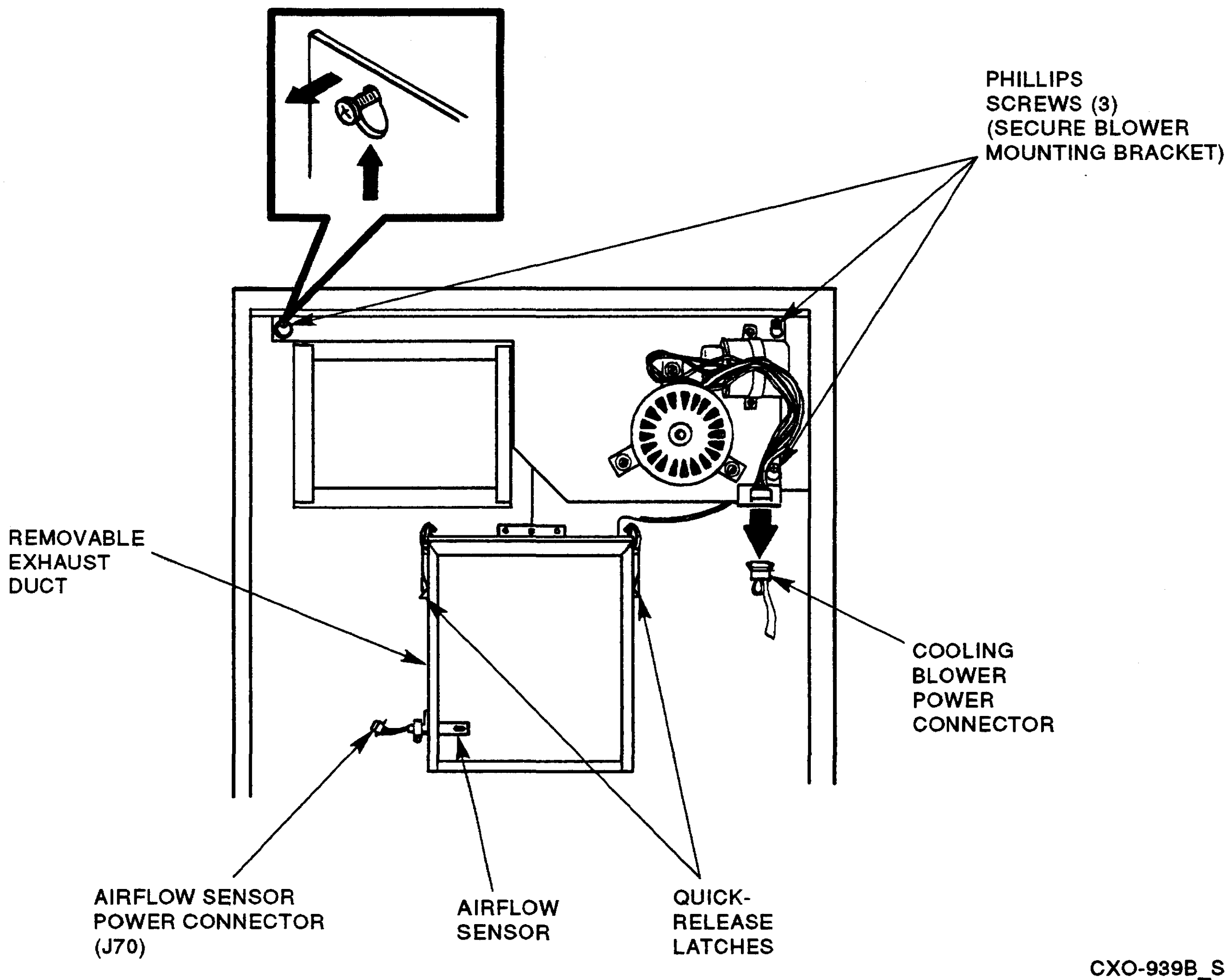


Figure 3–37 Removing and Replacing the HSC Main Cooling Blower

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.
8. Reverse the removal procedure to replace the cooling blower.

3.6.8 Removing and Replacing the HSC50 Blower

The blower, which provides forced air cooling for the cabinet, is removed and replaced with the following procedure:

1. Open the back door using a 5/32-inch hex wrench.
2. Turn off ac power (CB1 on the power controller) (Figure 3–38).

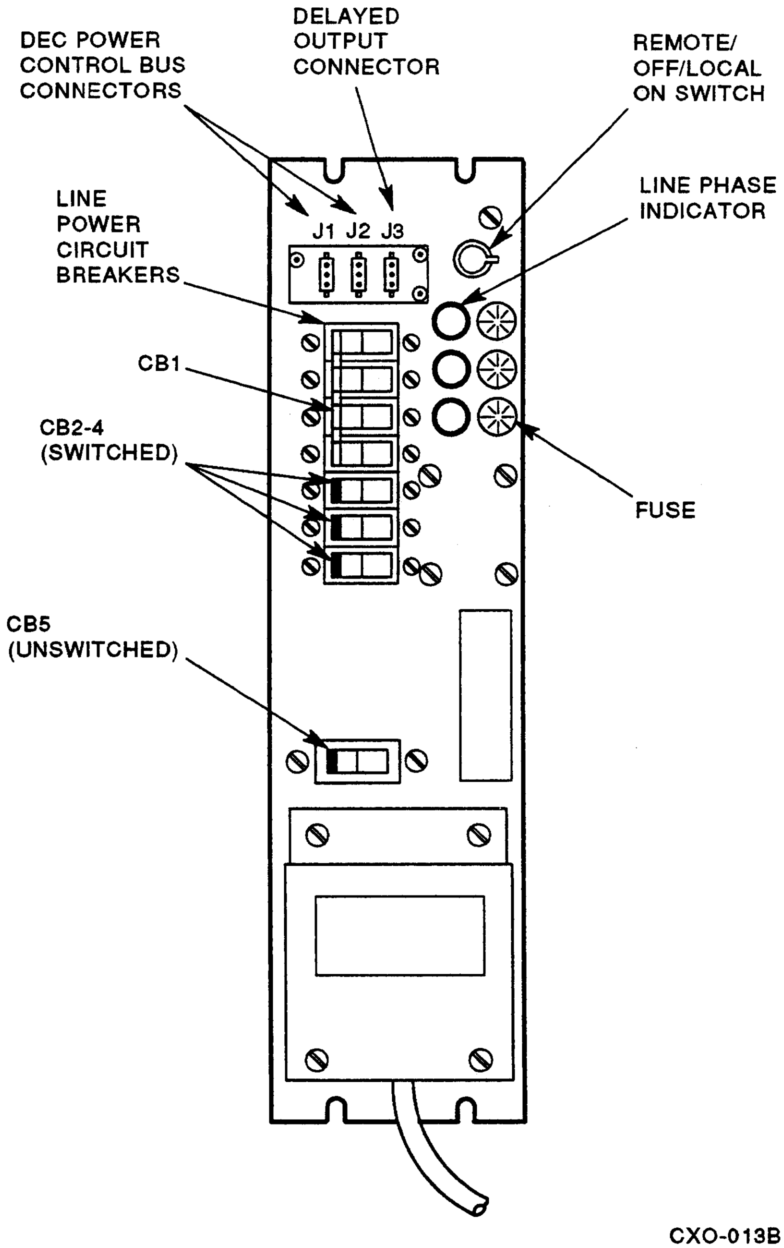


Figure 3–38 HSC50 Power Controller Circuit Breaker

3. Disconnect the blower power connector (Figure 3–39).
4. Disconnect the airflow sensor power connector (J70) to allow removal of the exhaust duct (Figure 3–39).
5. Remove the exhaust duct from the bottom of the blower by lifting up the quick release latches on each side of the duct (Figure 3–39).
6. Loosen, but do not remove, the three Phillips screws holding the blower mounting bracket to the cabinet (Figure 3–39).

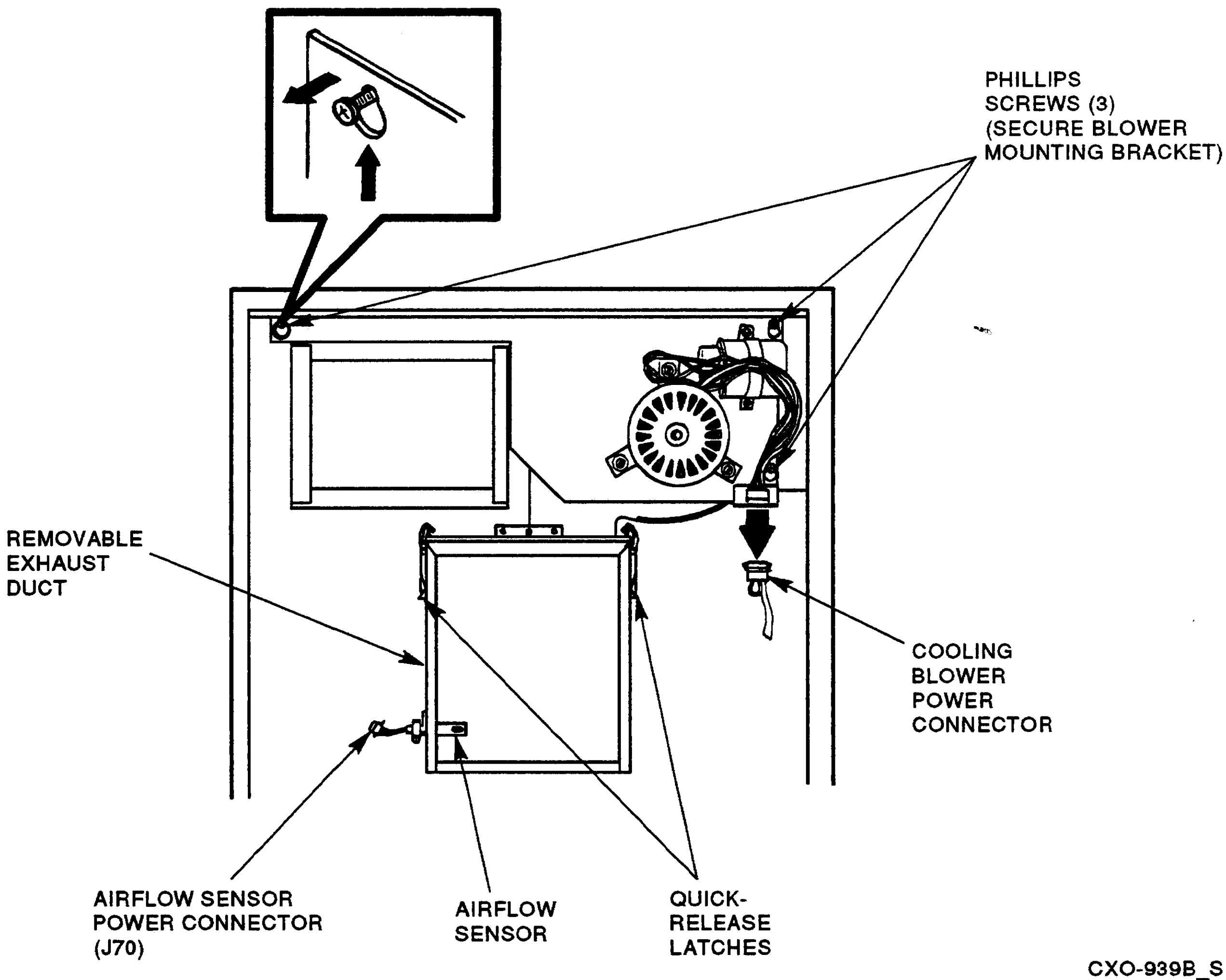


Figure 3–39 Removing and Replacing the HSC50 Blower

7. Lift the blower and bracket up and out of the cabinet.
8. Reverse the removal procedure to replace the blower.

3.6.9 Removing and Replacing the 881 Power Controller

The power controller must be removed to replace a power supply.

Use the following procedure to remove and replace the 881 power controller:

1. Open the back door using a 5/32-inch hex wrench.
2. Remove rear door latch to allow clearance for power controller removal.
3. Remove ac power by placing CB1 in the off position (Figure 3–40).

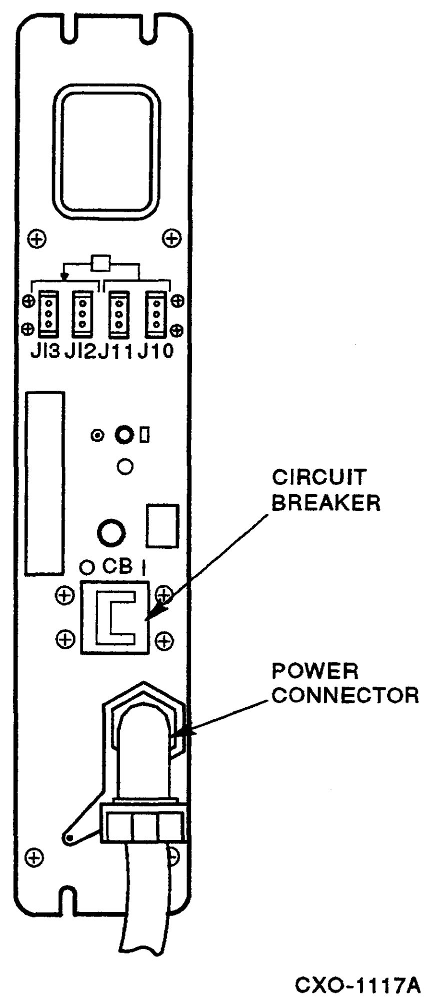


Figure 3–40 881 Power Controller Circuit Breaker

4. Unplug the power controller from the power source (Figure 3–41).
5. Remove the two top screws and then the two bottom screws securing the power controller to the cabinet (Figure 3–41). While removing the two bottom screws, push up on the power controller to take the weight off the screws.

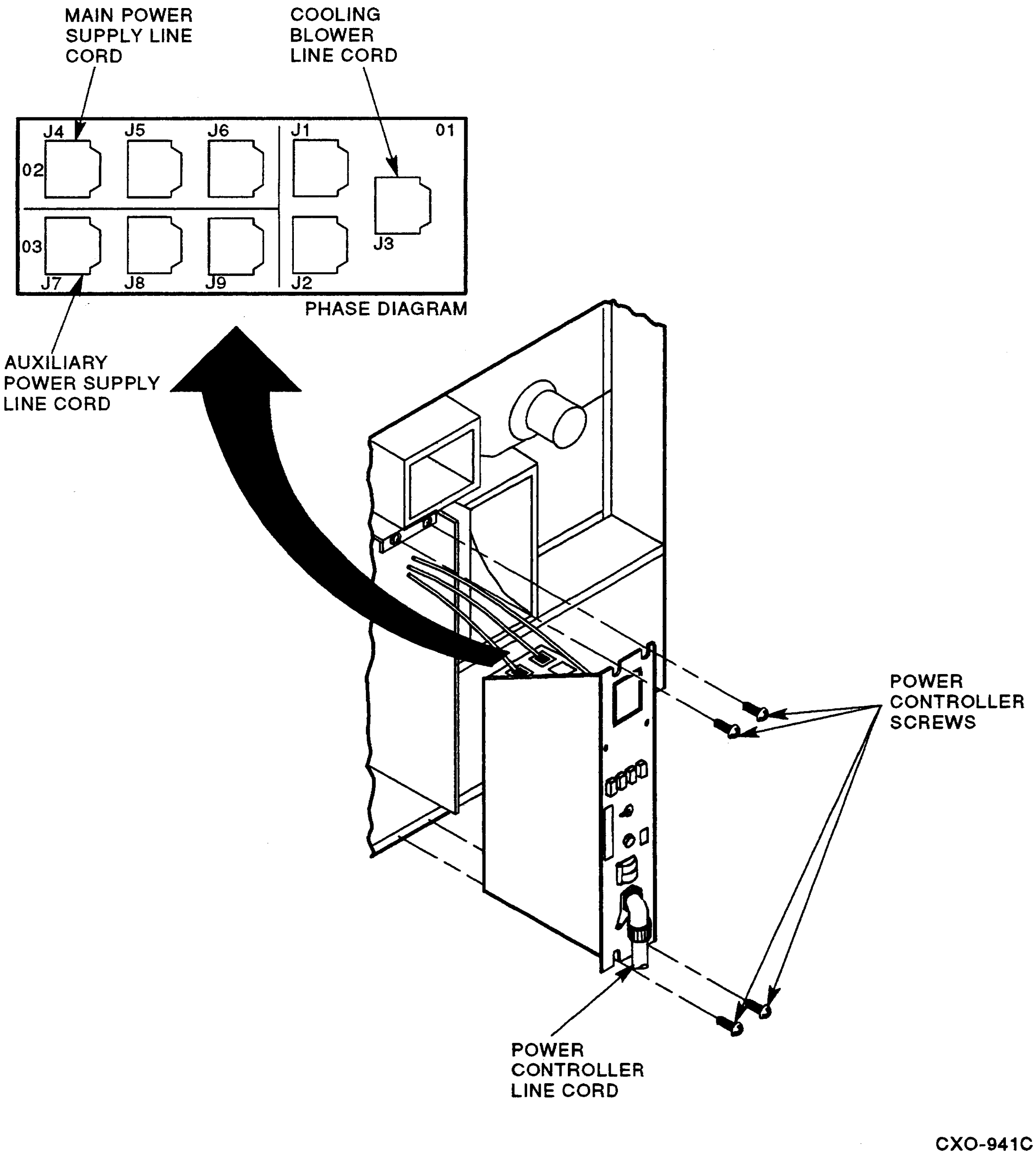


Figure 3–41 Removing and Replacing the 881 Power Controller

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 connectors J10, J11, J12, and J13 at the front of the power controller (Figure 3–40).
8. Disconnect the total off connector at the rear of the power controller (Figure 3–42).

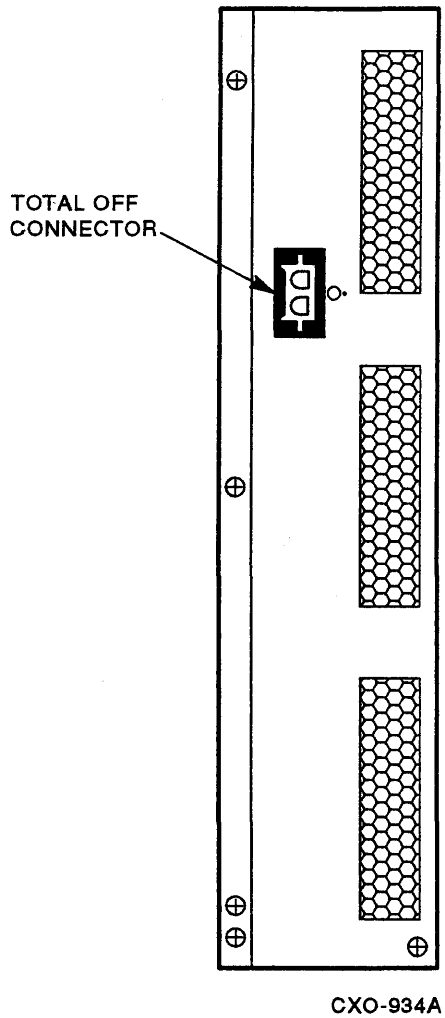


Figure 3–42 881 Total Off Connector

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 3–40, and replace the set screw in the other hole.

10. 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 3–41.

3.6.10 Removing and Replacing the HSC50 Power Controller

The HSC50 power controller must be removed to replace a power supply.

Use the following procedure to remove and replace the HSC50 power controller:

1. Open the back door using a 5/32-inch hex wrench.
2. Remove rear door latch to allow clearance for power controller removal.
3. Turn off ac power (CB1 on the power controller) (Figure 3–43).

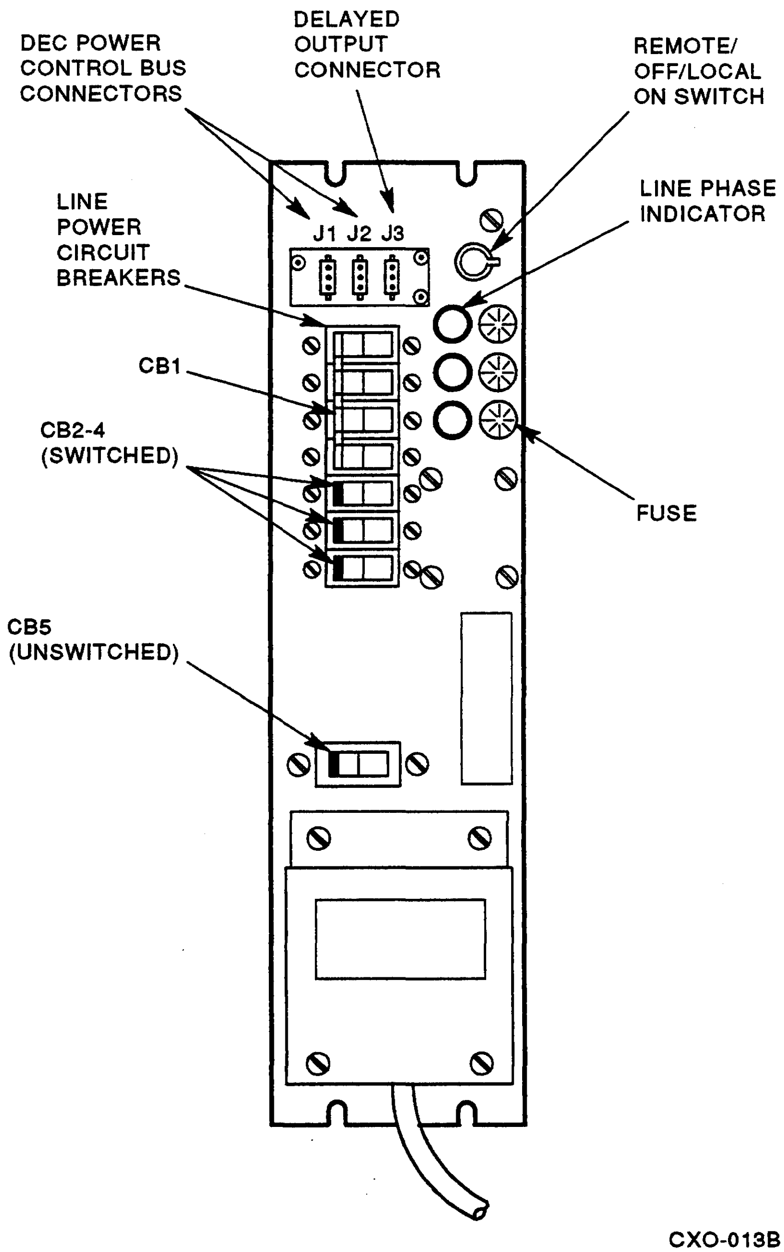


Figure 3–43 HSC50 Power Controller Circuit Breaker

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 3–44). While removing the two bottom screws, push up on the power controller to take the weight off the screws.

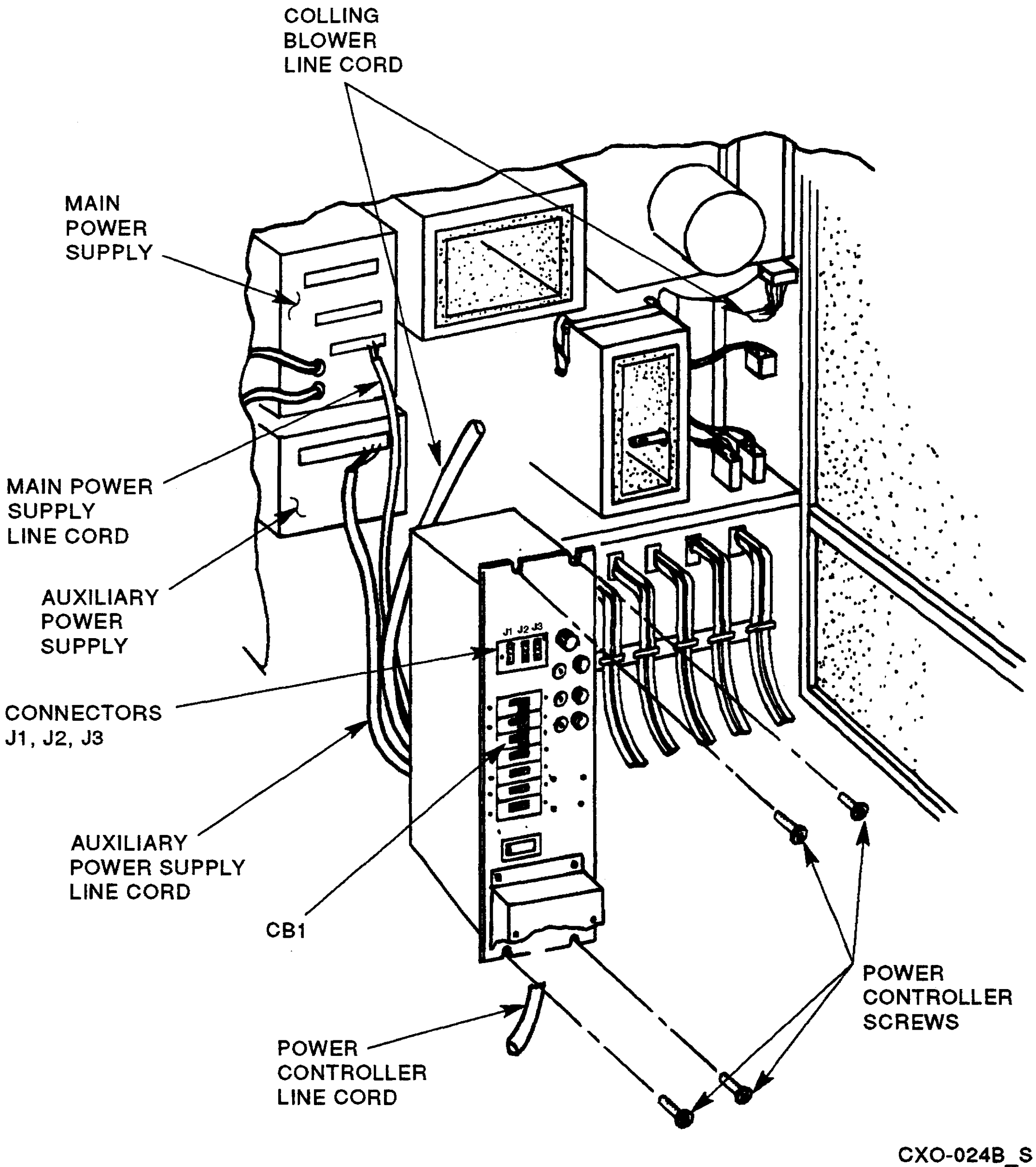


Figure 3–44 Removing and Replacing the HSC50 Power Controller

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 connectors J1, J2, and J3 at the front of the power controller (Figure 3–43).
8. Turn off ac power (CB1 on the power controller) (Figure 3–43).

9. Reverse the removal procedure to replace the power controller.

3.6.11 Removing and Replacing the HSC Main Power Supply

Use the following procedure to remove and replace the HSC main power supply:

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

1. Open the back door using a 5/32-inch hex wrench.
2. Turn off ac power (CB1 on the power controller) (Figure 3–45).

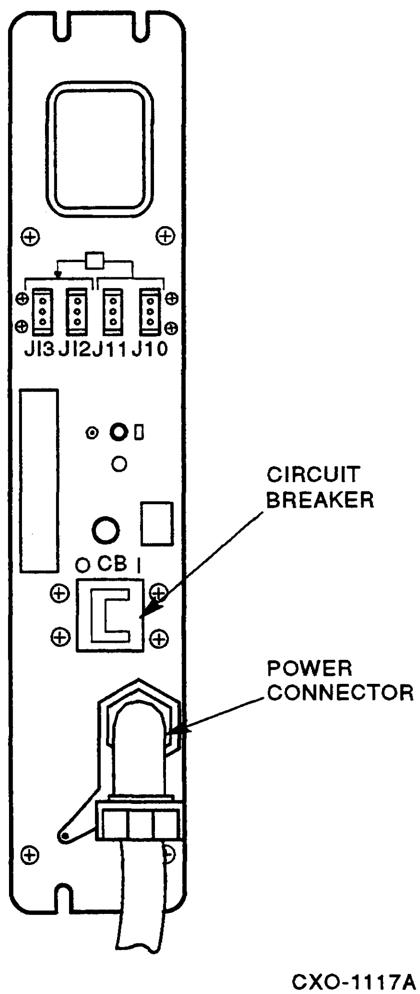


Figure 3–45 881 Power Controller Circuit Breaker

3. Unplug the power controller from the power source.
4. Remove the front door.
5. Remove the power controller (Section 3.6.9) to access the back of the power supply.

6. Unplug the main power supply line cord at the power controller.

   NOTE
   While performing 7 through 15, refer to (Figure 3–46).

7. Remove the nut from the –V1 stud (ground) on the back of the power supply.
8. Remove the nut from the +V1 stud (+5 volts) on the back of the power supply.
9. Remove the nut from the –V2 (ground) stud on the back of the power supply.
10. Remove the nut from the +V2 (–5.2 volts) stud.
11. Unplug J31 (+12 VDC output from the supply to backplane, power fail, and –5 volts sense line).
12. Unplug P32 (+12 VDC sense line and +5 VDC sense line).
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).

    Figure 3–46 shows the HSC main power supply test points.

WIRE LIST					
COLOR	POSITION	SIGNAL	COLOR	POSITION	SIGNAL
PURPLE	TBI-3-5	12 V	PURPLE	TB1-3-1	12 V SENSE
PURPLE	TBI-3-6	12 V	BLUE	TB1-2-7	ACC
BLACK	TBI-3-3	GND (12 V)	BROWN	TB1-2-6	AC
BLACK			GRN/YEL	TB1-2-5	GND
ORANGE	TBI-2-2	-5 V SENSE	YELLOW	TB1-2-3	ON/OFF (-5, 3 V)
BLACK	TBI-2-1	GND (-5 V SENSE)	ORANGE	TB1-2-2	-5 V SENSE (S2-)
BROWN	TBI-1-4	POWER FAIL	BLUE	TB1-1-3	ON/OFF 5 V
BLACK	TBI-1-2	GND (5 V SENSE)	BLACK	TB1-1-2	GND (5 V SENSE)
RED	TB1-1-1	5 V SENSE	PURPLE	TB1-3-2	12 V
BLACK	TB1-3-4	GND (12 V SENSE)			

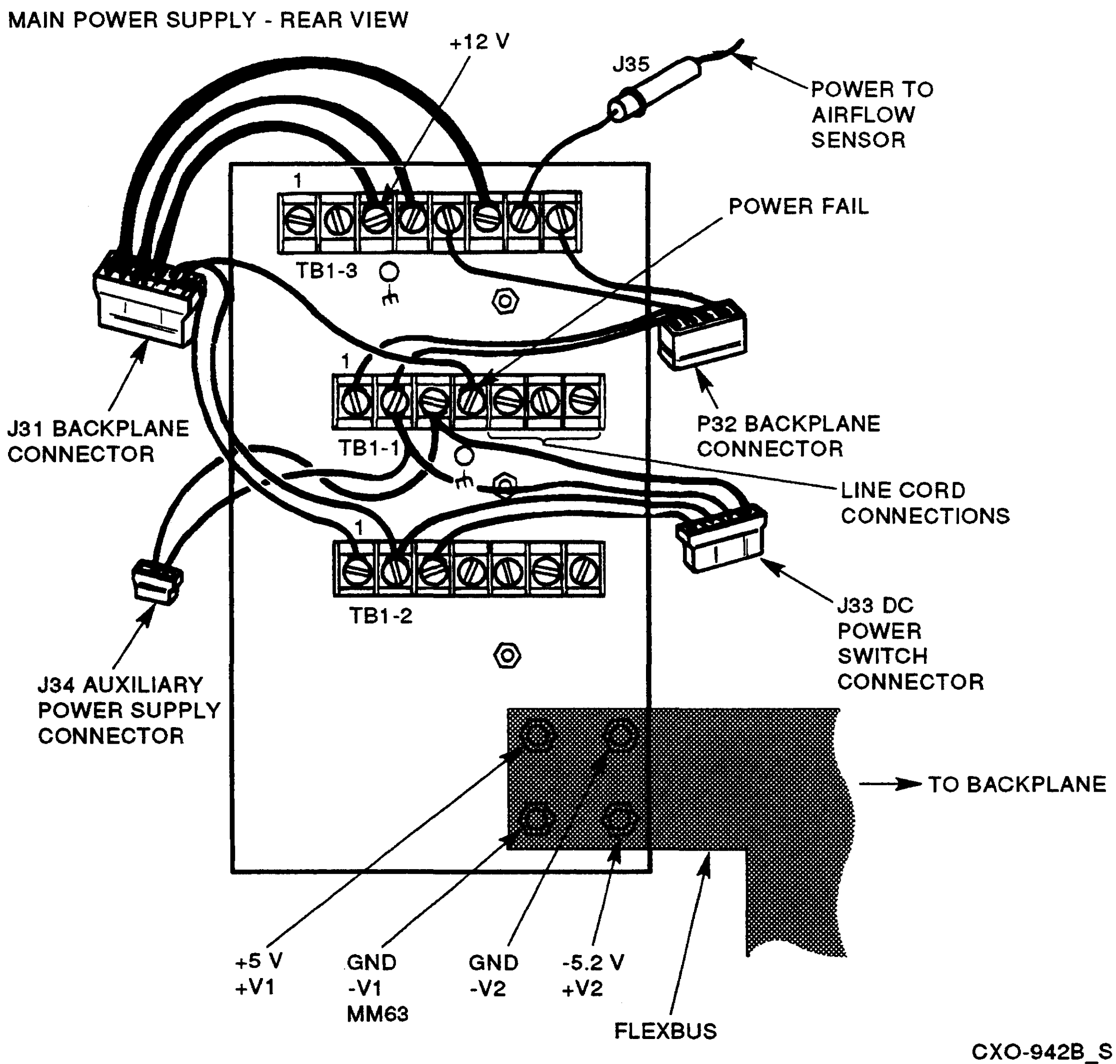


Figure 3–46 HSC Main Power Supply Cables and Test Points

16. Turn the four captive screws on the front of the power supply counterclockwise (Figure 3–47).

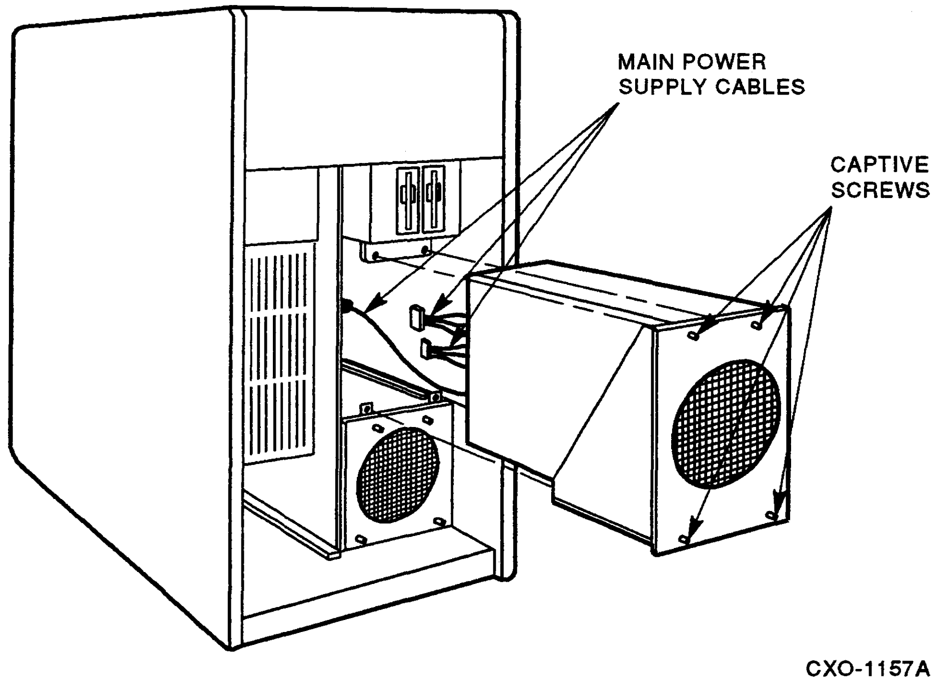


Figure 3–47 Removing and Replacing the HSC70 Main Power Supply

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.
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.

20. Reverse the removal procedure to replace the main power supply.

3.6.12 Removing and Replacing the HSC50 Main Power Supply

Use the following procedure to remove and replace the HSC50 main power supply:

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

1. Open the back door using a 5/32-inch hex wrench.
2. Turn off ac power (CB1 on the power controller) (Figure 3–48).

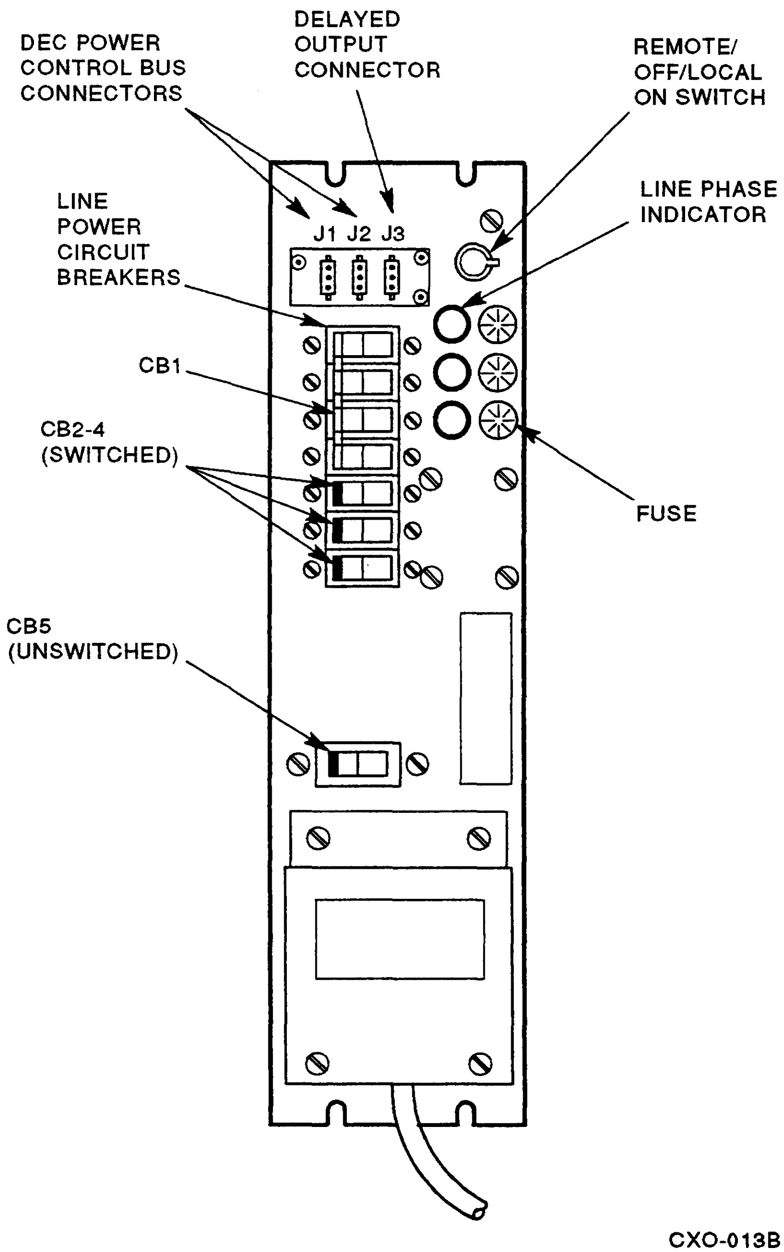


Figure 3–48 HSC50 Power Controller Circuit Breaker

3. Unplug the power controller from the power source.
4. Remove the front door.

5. Remove the power controller (Section 3.6.10) to access the back of the power supply.
6. Unplug the main power supply line cord at the power controller.

NOTE
While performing 6 through 14, refer to Figure 3–49, which shows the HSC50 main power supply test points.

7. Remove the nut from the –V1 stud (ground) on the back of the power supply (Figure 3–49).
8. Remove the nut from the +V1 stud (+5 volts) on the back of the power supply (Figure 3–49).
9. Remove the nut from the –V2 (ground) stud on the back of the power supply (Figure 3–49).
10. Remove the nut from the +V2 (–5.2 volts) stud on the back of the power supply (Figure 3–49).
11. Unplug J31 (+12 VDC output from the supply to backplane, power fail, and –5 volts sense line) (Figure 3–49).
12. Unplug P32 (+12 VDC sense line and +5 VDC sense line) (Figure 3–49). Ensure the P32 cable is free to be removed with the power supply.
13. Unplug J33 (to dc power switch) (Figure 3–49).
14. Unplug J34 (remote on/off jumper to auxiliary power supply) (Figure 3–49).
15. Unplug J35 (+12 VDC power to the airflow sensor) (Figure 3–49).

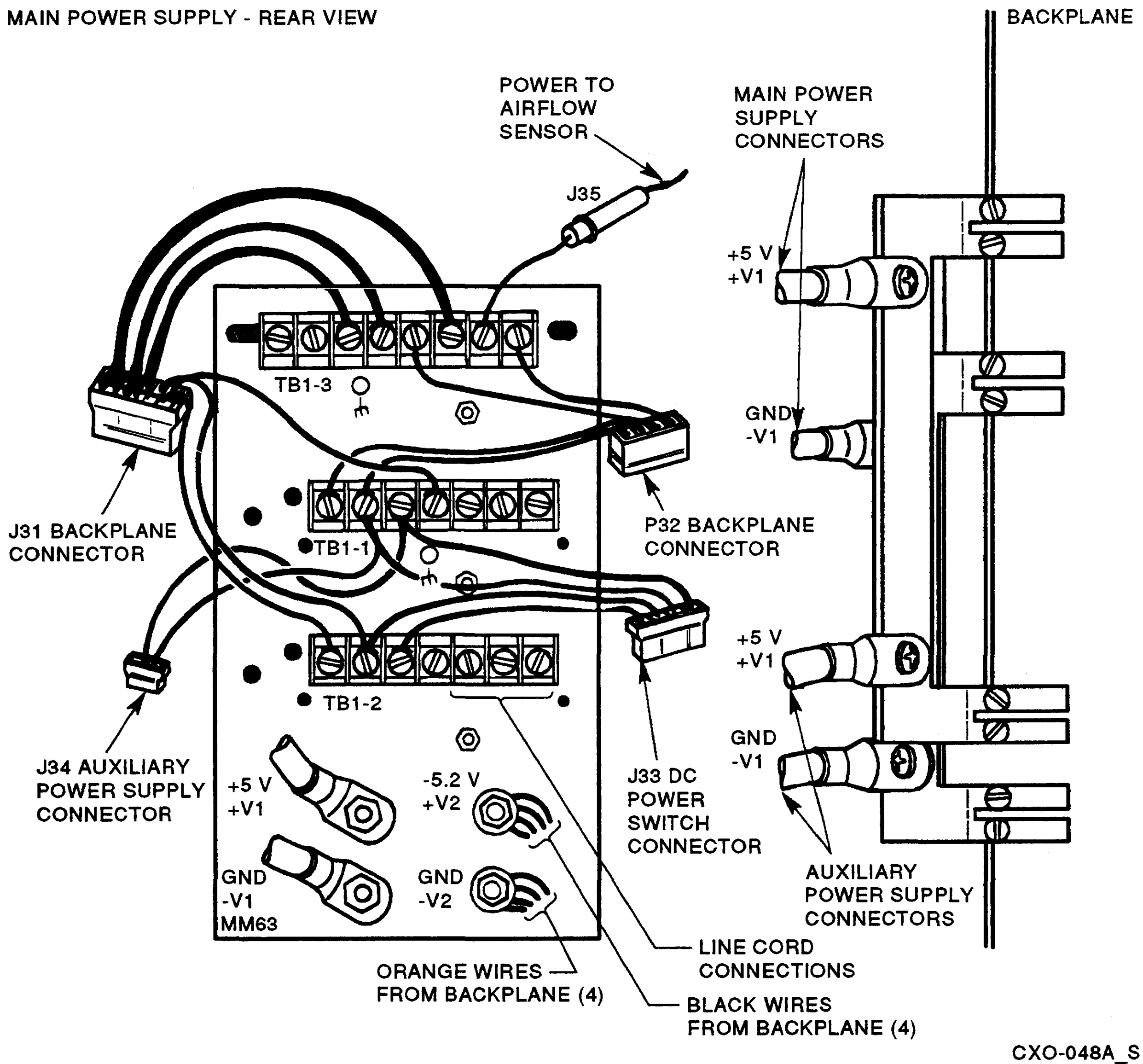


Figure 3–49 HSC50 Main Power Supply Cables and Voltage Test Points

16. Turn the four captive screws on the front of the power supply counterclockwise (Figure 3–50).

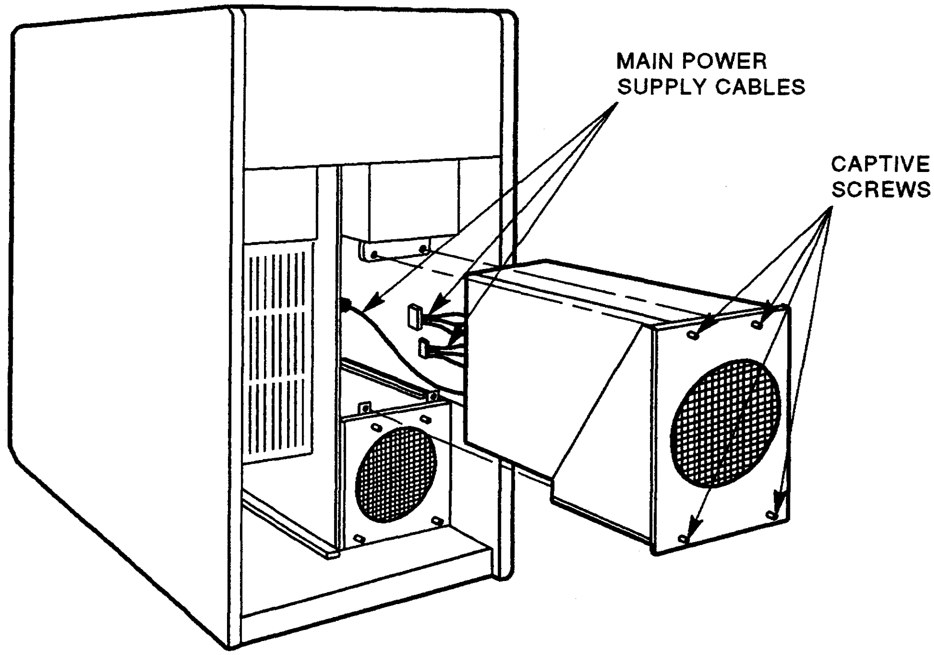


Figure 3–50 Removing and Replacing the HSC50 Main Power Supply

17. Pull the power supply out about an inch. Check the back of the cabinet to ensure the cables 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.
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.

20. Reverse the removal procedure to replace the HSC50 main power supply.

3.6.13 Removing and Replacing the HSC Auxiliary Power Supply

An HSC requires an auxiliary power supply if the total module count in the card cage is more than eight. The auxiliary power supply is mounted directly beneath the main power supply.

Use the following procedure to remove and replace the HSC auxiliary power supply:

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

1. Open the back door using a 5/32-inch hex wrench.
2. Turn off ac power (CB1) on the power controller (Figure 3–51).

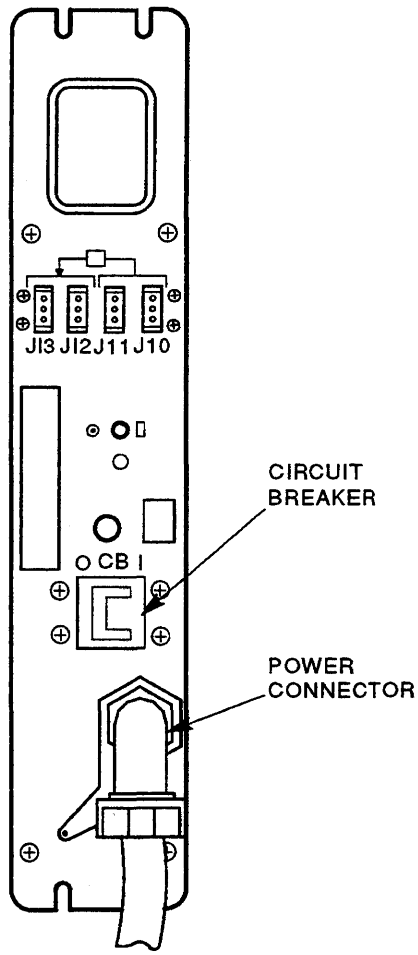


Figure 3–51 881 Power Controller Circuit Breaker

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 3.6.9).
6. Unplug the auxiliary power supply line cord at the power controller.

NOTE
While performing 7 through 10, refer to Figure 3–52.

7. Remove the nut from the +V1 stud (+5 volt) on the back of the power supply.
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).

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)

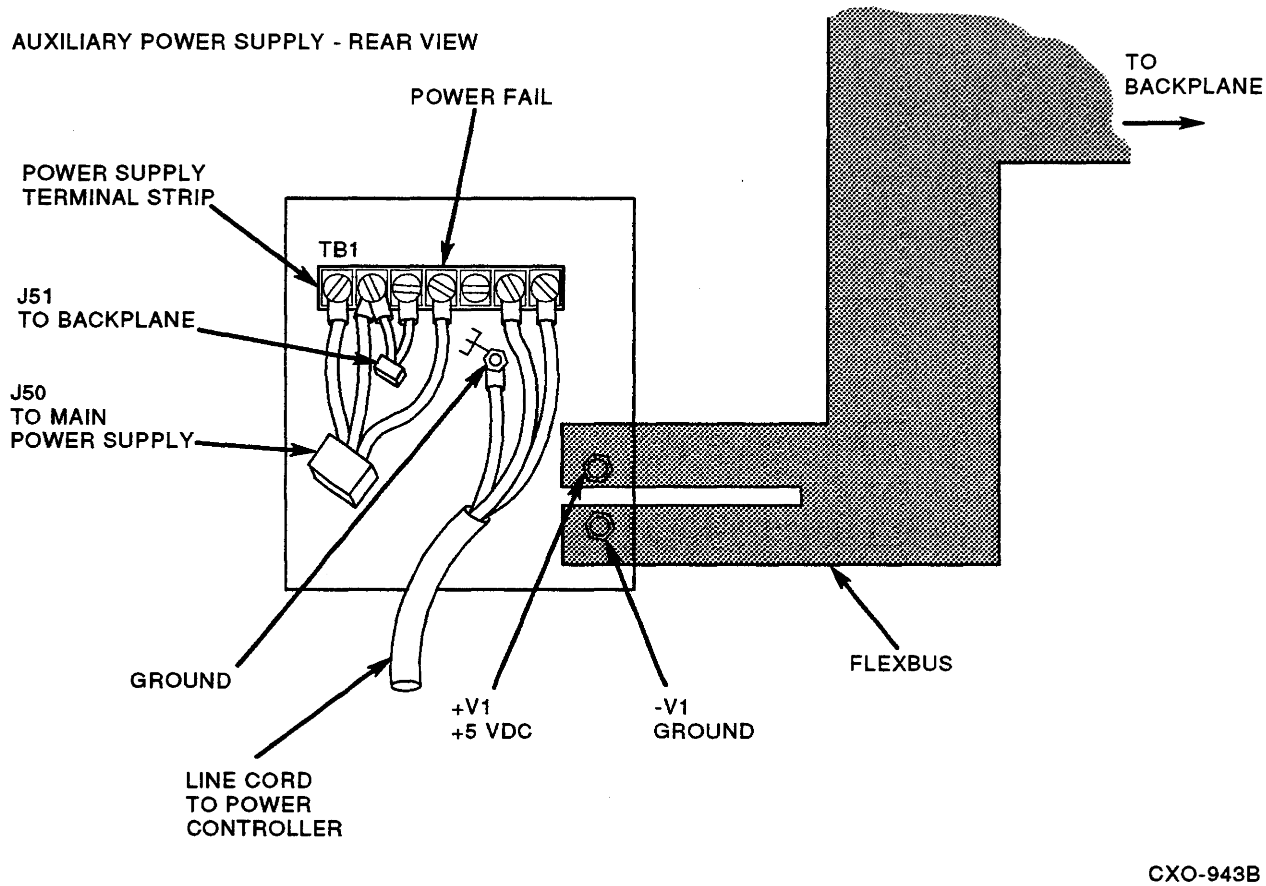


Figure 3–52 HSC Auxiliary Power Supply Cable and Test Points

11. Figure 3–52 shows the HSC auxiliary power supply test points.
12. Turn the four captive screws on the power supply counterclockwise (Figure 3–53).

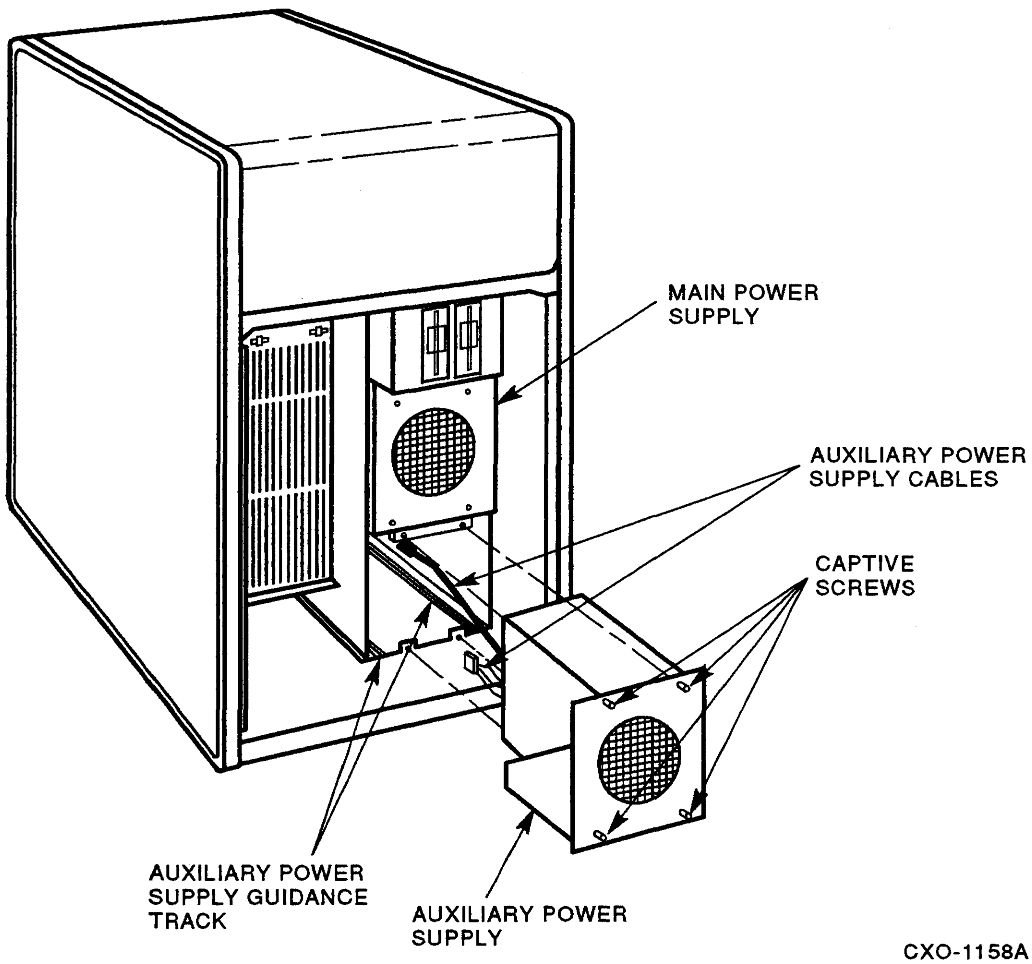


Figure 3–53 Removing and Replacing the HSC Auxiliary Power Supply

13. Pull the power supply out about an inch. Check the back of the cabinet to ensure the cables and flexbus connectors are clear.
14. Carefully slide the power supply out through the front of the HSC.
15. Remove the power cord from the failing unit and install it to the new power supply.

    NOTE
    Spare supplies are not shipped with a power cord.

16. Reverse the removal procedure to replace the HSC auxiliary power supply.

3.6.14 Removing and Replacing the HSC50 Auxiliary Power Supply

An HSC50 requires an auxiliary power supply if the total module count in the card cage is more than eight. The auxiliary power supply is mounted directly beneath the main power supply.

Use the following procedure to remove and replace the HSC50 auxiliary power supply:

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

1. Open the back door using a 5/32-inch hex wrench.
2. Turn off the ac circuit breaker (CB1) on the HSC50 power controller (Figure 3–54).

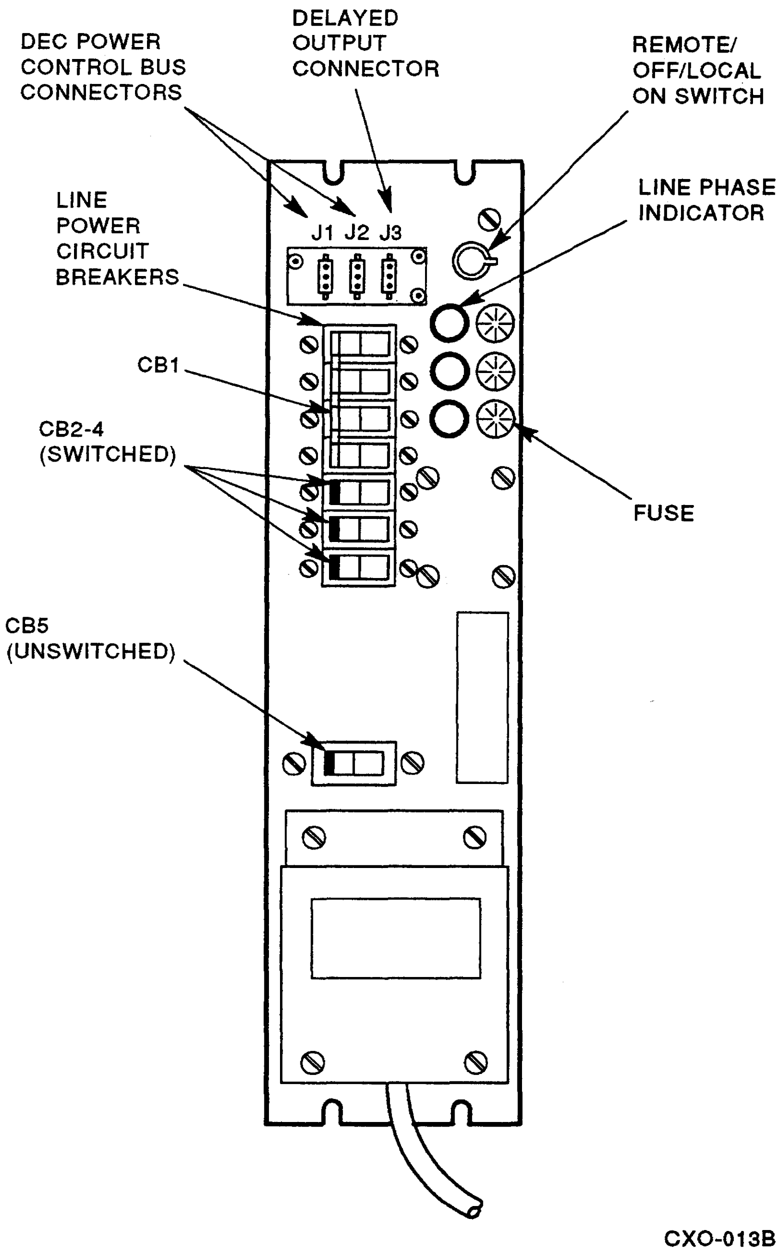


Figure 3–54 HSC50 Power Controller Circuit Breaker

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 3.6.10).
6. Unplug the auxiliary power supply line cord at the power controller.

NOTE
While performing 7 through 10, refer to Figure 3–55.

7. Remove the nut from the +V1 stud (+5 volt) on the back of the power supply (Figure 3–55).
8. Remove the nut from the –V1 stud (ground) on the back of the power supply (Figure 3–55).
9. Disconnect J50 (sense line to voltage comparator). (Figure 3–55).
10. Disconnect J51 (dc on/off jumper) (Figure 3–55). Refer to Figure 3–55 for the HSC50 auxiliary power supply test points.

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)

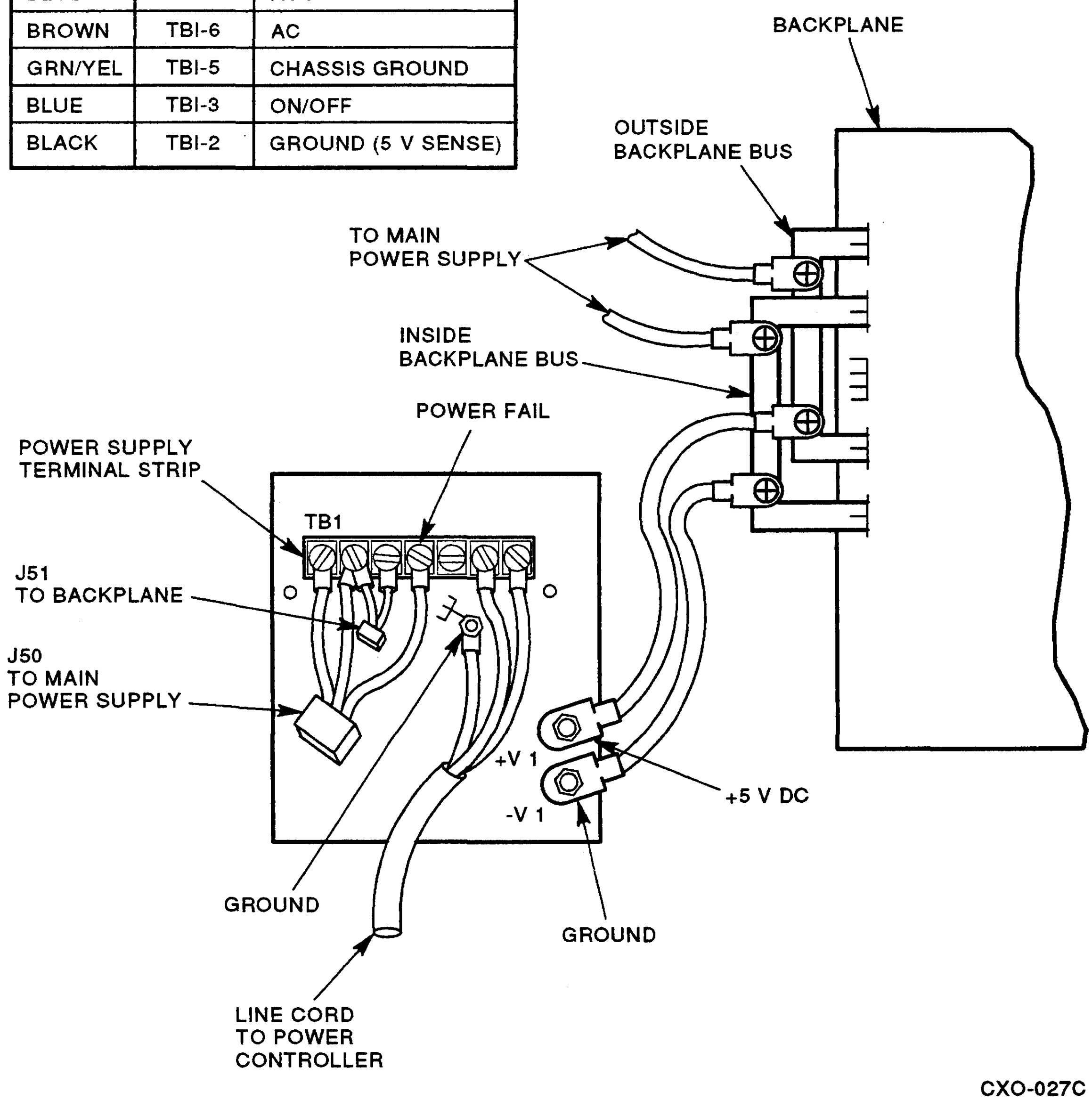


Figure 3–55 HSC50 Auxiliary Power Supply Cable and Voltage Test Points

11. Turn the four captive screws on the power supply counterclockwise (Figure 3–56).

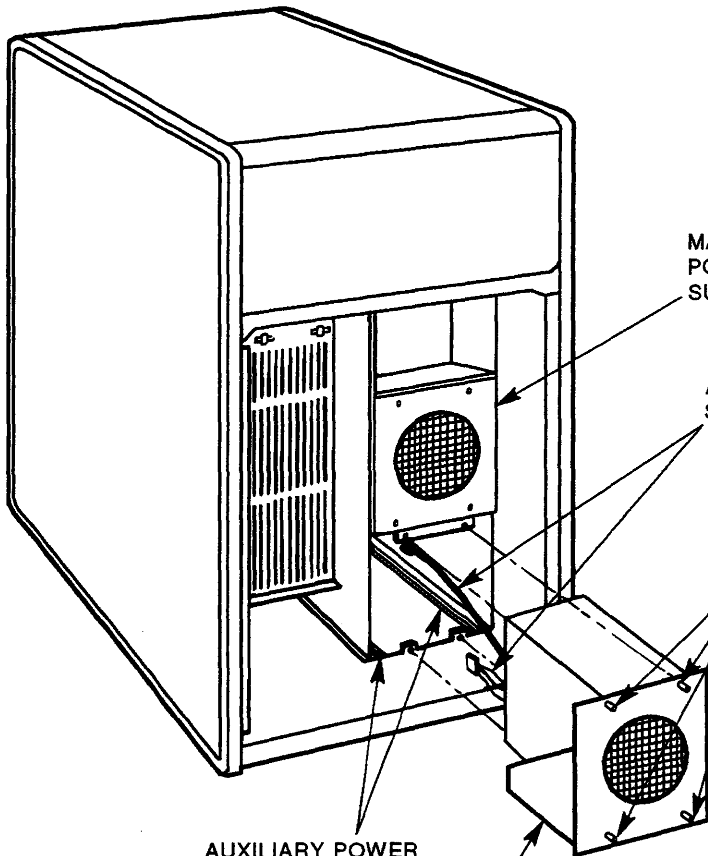


Figure 3–56 Removing and Replacing the HSC50 Auxiliary Power Supply

12. Pull the power supply out about an inch. Check the back of the cabinet to ensure the cables and connectors are clear.
13. Carefully slide the power supply out through the front of the HSC50.
14. Remove the power cord from the failing unit and install it to the new power supply.

    NOTE
    Spare supplies are not shipped with a power cord.

15. Reverse the removal procedure to replace the auxiliary power supply.

4 Initialization Procedures

4.1 Introduction

This chapter contains procedures for connecting the console terminal on the HSC and the auxiliary terminal on the HSC50, and initialization procedures for both HSC models.

A malfunction during initialization may be reported by a fault code displayed on the operator control panel (OCP). These fault codes are explained in Chapter 8.

4.2 Console/Auxiliary Terminal

The console or auxiliary terminal designated for the HSC can be a VT2xx, VT3xx, VT1xx, or an LA12 DECwriter. An LA75 or LA50 printer for hardcopy output is connected to the VT2xx, VT3xx, and can be connected to the VT1xx if the VT1xx has the printer port option installed. Detailed operating information is provided in the appropriate owner manuals accompanying the VTxxx and LAxx models.

NOTE
The VT3xx series terminal can be connected to an RS-232 compatible port only. Connection to another type of port will result in initialization failure and FCC violations.

4.2.1 Console Terminal Connection

Figure 4–1 shows the placement of the EIA terminal connectors on the HSC rear bulkhead. The console terminal connects to the J60 connector as shown.

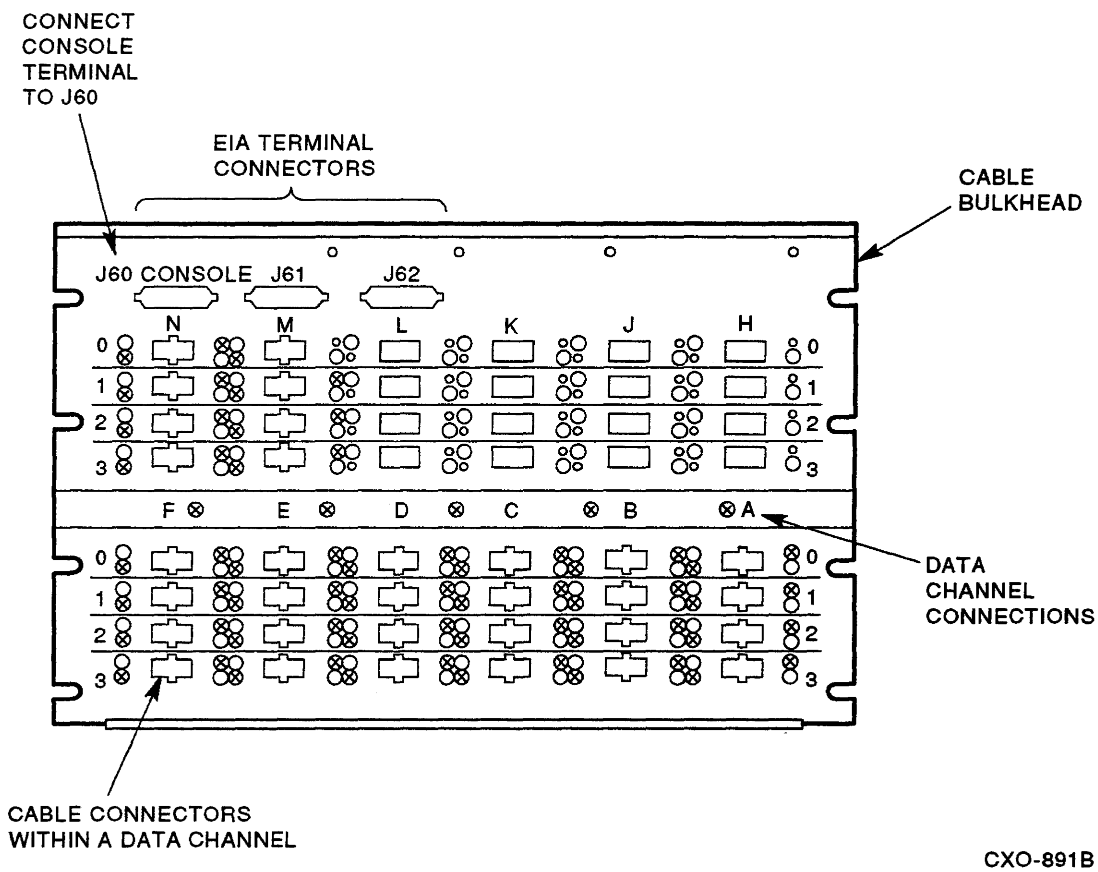


Figure 4–1 Console Terminal Connection

Preferably, power is turned off before the console terminal is installed. However, power can be left on while connecting the terminal. Use the following procedure for installing the console terminal with power on or off:

1. Put the Secure/Enable switch in the secure position.
2. Change terminal state (plug in, remove power, connect EIA line, and so forth).
3. 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 HSC may enter micro-on-line debugging tool (ODT) mode. This mode is indicated by an @ symbol on the screen. Typing a P (PROCEED) should exit this mode.

4.2.2 HSC50 Auxiliary and Maintenance Terminal Connections

Figure 4–2 shows the placement of the two ASCII ports on the HSC50 and HSC50 (modified). The auxiliary terminal can be connected to either the rear or the front ASCII port. Two terminals *cannot* be connected at the same time.

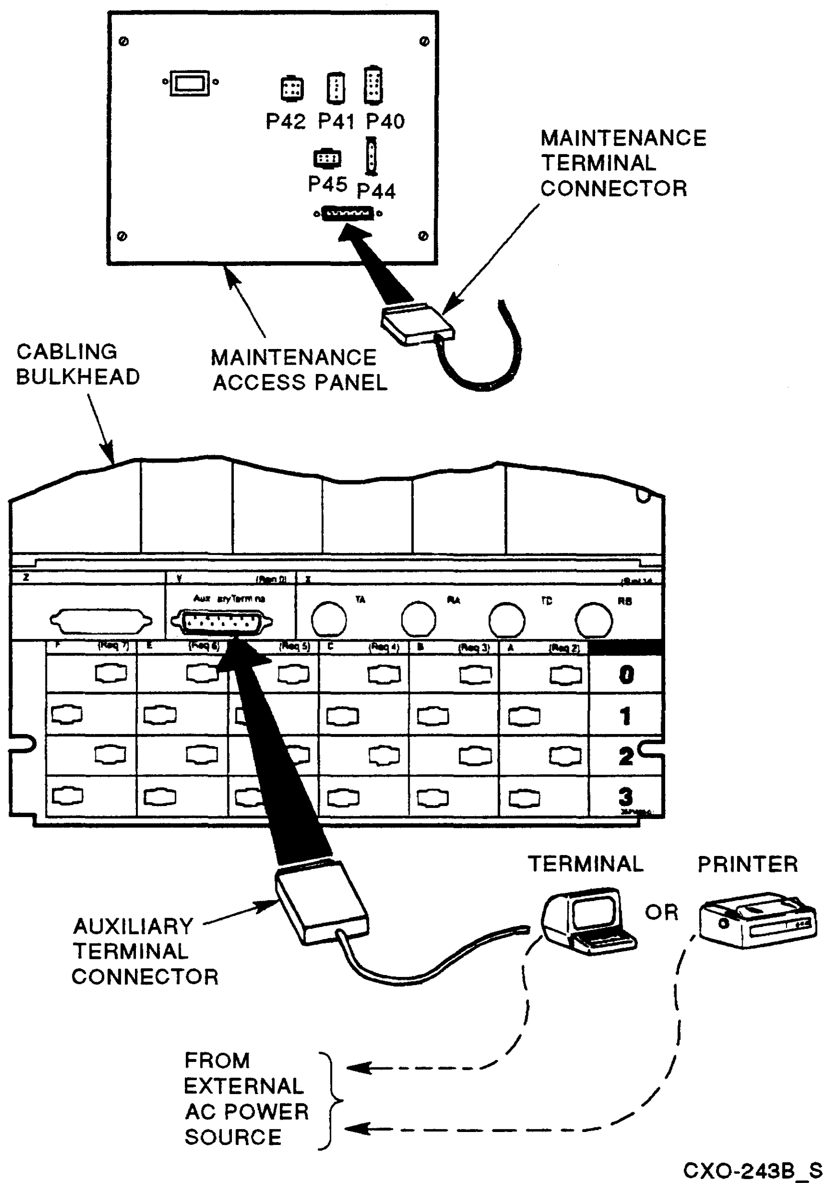


Figure 4–2 Auxiliary or Maintenance Terminal Connection

Preferably, power is turned off before the console terminal is installed. However, power can be left on while connecting the terminal. Use the following procedure for installing the console terminal with power on or off:

1. Put the Secure/Enable switch in the secure position.
2. Change terminal state (plug in, remove power, connect EIA line, and so forth).
3. Type three space characters on the terminal keyboard.

4. If it is necessary to put the Secure/Enable switch in the enable position, do so at this point.

NOTE
If this procedure is not followed, the HSC50 may enter micro-on-line debugging tool (ODT) mode. This mode is indicated by an @ symbol on the screen. Typing a P (proceed) should exit this mode.

4.2.3 LA12 Parameters

Detailed information on LA12 terminal installation and operation is found in the *DECwriter Correspondent Technical Manual* (EK-CPL12-TM).

When an LA12 is used as an auxiliary terminal, the following parameters must be established:

1. Communications:
   - Auto - Ansbk = no
   - Buffer = 1024
   - Comm Port = EIA
   - Disk - HDX = none
   - Echo - Local = no
   - Fault = none
   - G - HDX Start Mode = Rcv
   - H - Hi Speed (bps) = 9600
   - L - Lo Speed (bps) = 300
   - M - Line Prot = FDX - Data Leads
   - O - Rcv Error Ovride = no
   - Parity = 7/M
   - Q - SRTS Polarity = lo
   - Restraint = Xon/Xoff
   - S - Speed Select = hi
   - Turn Char = none
   - U - Power Up = line
   - V - Frequency = bell 103
2. Keyboard:
   - Auto - Linefeed = no
   - Break = no
   - C - Keyclick = no
   - Keypad = normal
   - Language = USA
   - Repeat = yes
3. Printer:
   - A - G0 Char Set = USA

- B - G1 Char Set = USA
- C - G2 Char Set = USA
- D - G3 Char Set = USA
- End-of-line = wrap
- Form Length = 264
- G - Print Cntrl Chars
- Horiz Pitch (CPI) = 10
- Newline Char = none
- Print Force = hi
- Vertical Pitch (LPI) = 6

4.3 HSC Initialization

This section describes the initialization procedures for the HSC using the system diskette. This diskette also contains the software necessary to execute the device integrity tests and the utilities. To boot and run the off-line diagnostics from a separate off-line diskette, refer to Chapter 7.

System initialization is started by powering on the unit or (if the unit is already on) by pressing 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.

NOTE
In order to run the HSC device integrity tests, 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.

Logic in the following areas is tested with the Init P.ioj diagnostic:

- Control processor—The rest of the instruction set not tested by the ROM bootstrap, interrupts, memory management, and the control memory lock-cycle circuitry are included. Detected failures result in an error code display on the OCP (Figure 4–3).
- Memory—Program memory is tested from the I/O control processor. However, the control and data memories are tested by the highest-numbered available requestor controlled by the I/O control processor. Again, detected failures result in an OCP error code display.
- Host interface and data channels—Module status is collected and placed in a table for the HSC operating software initialization process. As each module is enabled, it automatically executes internal microdiagnostics. These internal diagnostics test the following:
  - ROM (sequencer, checksum, parity, and so forth)
  - Special logic unique to that particular module

Upon completion of diagnostics for each module, a status code is passed to the I/O control processor. Status codes for the various modules are discussed in Chapter 5.

If the module diagnostics complete successfully, the status code represents the module type and the green LED is turned on. If the diagnostics fail, the status code indicates the failing microtest. In addition, detected failures cause a red LED to light on that module. K.ci, K.sdi, K.sti, and K.si failures are also displayed on the console terminal after the boot is completed.

DESCRIPTION	HEX	OCT	BINARY	OCP INDICATORS				
				INIT	FAULT	ONLINE		
K.PLI ERROR ***	01	01	00001	OFF	OFF	OFF	OFF	ON
K.SDI/K.SI INCORRECT VERSION OF MICROCODE ***	02	02	00010	OFF	OFF	OFF	ON	OFF
K.STI/K.SI INCORRECT VERSION OF MICROCODE ***	03	03	00011	OFF	OFF	OFF	ON	ON
P.IOJ CACHE FAILURE *	08	10	01000	OFF	ON	OFF	OFF	OFF
K.CI FAILURE *	09	11	01001	OFF	ON	OFF	OFF	ON
DATA CHANNEL MODULE ERROR *	0A	12	01010	OFF	ON	OFF	ON	OFF
P.IOJ/C MODULE FAILURE	11	21	10001	ON	OFF	OFF	OFF	ON
M.STD2 MODULE FAILURE *****	12	22	10010	ON	OFF	OFF	ON	OFF
BOOT DEVICE FAILURE **	13	23	10011	ON	OFF	OFF	ON	ON
PORT LINK NODE ADDRESS SWITCHES OUT OF RANGE	15	25	10101	ON	OFF	ON	OFF	ON
MISSING FILES REQUIRED ****	16	26	10110	ON	OFF	ON	ON	OFF
NO WORKING K.CI, K.SDI, K.STI, OR K.SI IN SUBSYSTEM	18	30	11000	ON	ON	OFF	OFF	OFF
INITIALIZATION FAILURE	19	31	11001	ON	ON	OFF	OFF	ON
SOFTWARE INCONSISTENCY	1A	32	11010	ON	ON	OFF	ON	OFF
ILLEGAL CONFIGURATION	1B	33	11011	ON	ON	OFF	ON	ON

* THESE ARE THE SO-CALLED SOFT OR NONFATAL ERRORS.
** POSSIBLE MEMORY MODULE/CONTROLLER ON HSC70.
*** INCORRECT VERSION OF MICROCODE.
**** THIS FAULT CODE WILL ALSO BE DISPLAYED IF THE L0105 MODULE IS NOT AT THE MINIMUM REV LEVEL.
*****SWAP MEMORY MODULE FIRST. IF PROBLEM PERSISTS, TRY THE P.IO MODULE.

CXO-905D

Figure 4–3 Operator Control Panel Fault Code Displays

NOTE

Lighting of the red LED on the L0100 or L0118 LINK module *does not* indicate a failure of the module.

For a detailed description of the boot process, refer to the HSC Boot Flowchart in Chapter 8.

4.3.1 Init P.io Test (INIPIO)

The INIPIO test completes the P.ioj module and the HSC memory testing previously started by the ROM bootstrap tests. All P.ioj logic not tested by the bootstrap is tested by INIPIO. In addition, the HSC Program, Control, and Data memories are tested.

This test runs in a standalone environment (no other HSC processes are running). If a failure is detected, the failing module is flagged by illumination of the red LED on the module. If the test runs without finding any errors, the HSC 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 off-line P.io test that provides standard HSC error messages.

4.3.2 INIPIO Test System Requirements

In order to run this test, the following hardware is required:

- P.ioj (processor) module with HSC boot ROM
- K.ci
- At least one M.std2 (memory) module
- RX33 controller with at least one working drive

In addition, an HSC system diskette (RX33 media) is required.

4.3.3 INIPIO Test Prerequisites

The INIPIO test is loaded by the HSC 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 INIPIO 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.

4.3.4 INIPIO Test Operation

Follow these steps to start the INIPIO test:

1. Insert the HSC system diskette in the RX33 unit 0 drive (left-hand drive).
2. Power on the HSC or press and release the Init button on the HSC OCP with the Secure/Enable switch enabled. The Init lamp lights and the following occurs:
   - The RX33 drive-in-use LED lights within 10 seconds, indicating the bootstrap is loading the INIPIO test to the Program memory.
   - The I/O State light is on after diskette motion stops and the INIPIO test begins testing.
   - The INIPIO test displays the following message on the HSC console when it begins:

     ```
     INIPIO-I BOOTING
     ```

   - HSC operational software is being loaded when the State light flashes rapidly.
   - HSC operational software indicates it has loaded properly when the State light blinks slowly.
   - HSC displays its name and version indicating it is ready to perform host I/O.

Once initiated, the INIPIO test is terminated only by halting and rebooting the HSC. If the test fails to load using the preceding startup procedure, perform the next four steps:

1. Check the OCP fault light. If the fault light is on, press the fault light once and check the fault code (Figure 4–3).
2. Boot the diskette from the RX33 unit 1 drive (right-hand drive).
3. Boot using another diskette. If that diskette boots, the original diskette is probably damaged or worn.
4. Boot using the HSC Off-line Diagnostic diskette. This diskette contains the off-line P.io test, which provides extensive error reporting features. A console terminal must be connected to run the off-line tests.

The progress of the INIPIO 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 at a steady rate. If the test detects an error, the Fault lamp on the HSC OCP is lit.

4.4 HSC50 Initialization

In order to run the HSC50 device integrity tests and utilities, the HSC50 operating software must be initialized with both the system and utilities cassettes loaded in the TU58 drives. Before inserting the system tape into the TU58 drive, check the black RECORD tab. This tab must be in the record position (as indicated by an arrow on the tab) to ensure proper system operation. The utilities tape need not be write-enabled.

NOTE
In order to run the HSC50 device integrity tests, the system tape must reside in the TU58 drive. Customarily, this tape resides in TU58 drive 0. Drive 1 and drive 0 are identical, and tape placement is arbitrary.

However, the utilities tape does not contain a bootable image, and if drive 0 contains the utilities tape, the system will try to boot from drive 1.

The HSC50 can be initiated by either powering on the unit if it is powered down or, if power is already applied, by pressing and releasing the Init switch with the Secure/Enable switch in the enable position. This causes the P.ioc bootstrap ROM tests to run and then load the Init P.ioc test.

4.4.1 HSC50 Off-line Diagnostics Tape

The off-line diagnostics tape can be booted in either TU58 drive and need not be write-enabled. The off-line tape can be booted by either powering on the unit or pressing and releasing the Init switch with the Secure/Enable switch in the enable position. This causes the P.ioc bootstrap ROM tests to run and then load the off-line P.ioc test.

4.4.2 Init P.ioc Diagnostic

The Init P.ioc test is loaded by the P.ioc ROM bootstrap test each time the HSC50 system tape is booted. This diagnostic completes the testing of the P.ioc module and the HSC50 memories. At the successful completion of these tests, the HSC50 operating software is loaded and started.

Logic in the following areas is tested with the Init P.ioc diagnostic:

- Control processor—The rest of the instruction set not tested by the ROM bootstrap, interrupts, memory management, and the control memory lock-cycle circuitry are included. Detected failures result in an error code display on the OCP (Figure 4–3).

- Memory—Program memory is tested from the I/O Control Processor. However, the control and data memories are tested by the highest-numbered available requestor controlled by the I/O control Processor. Again, detected failures result in an OCP error code display.
- Host interface and data channels—Module status is collected and placed in a table for the HSC50 operating software initialization process. As each module is enabled, it automatically executes internal microdiagnostics. These internal diagnostics test the following:
  - ROM (sequencer, checksum, parity, and so forth)
  - Special logic unique to that particular module

Upon completion of diagnostics for each module, a status code is passed to the I/O control processor. Status codes for the various modules are discussed in Chapter 5.

If the module diagnostics complete successfully, the status code represents the module type and the green LED is turned on. If the diagnostics fail, the status code indicates the failing microtest. In addition, detected failures cause a red LED to light on that module. K.ci, K.sdi, K.sti, and K.si failures are also displayed on the auxiliary terminal after the boot is completed.

NOTE
Lighting of the red LED on the L0100 or L0118 LINK module *does not* indicate a failure of the module.

For a detailed description of the boot process, refer to the HSC50 Boot Flowchart in Chapter 8.

4.5 Fault Code Interpretation

All failures occurring during the Init P.io test are reported on the OCP 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 HSC 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 pressing the Fault switch a second time. To restart the boot procedure, press the Init switch. To identify the probable failing module, see Figure 4–3. For detailed descriptions of OCP fault codes, see Chapter 8.

5
Device Integrity Tests

5.1 Introduction

Device integrity tests executing in the HSC do not interfere with normal operation other than with the device being tested. The device integrity tests can be found on the HSC system media disk or HSC50 utilities media tape.

The tests described in this chapter are:

- ILRX33—RX33 integrity tests
- ILTU58—TU58 integrity tests
- ILMEMY—Memory integrity tests
- ILDISK—Disk drive integrity tests
- ILTAPE—Tape device integrity tests
- ILTCOM—Tape compatibility tests
- ILEXER—Multidrive exerciser

5.1.1 Device Integrity Tests Common Areas

Device integrity tests prompts and error messages so that they conform to standard formats. All prompts issued by these integrity tests use a generic syntax.

- Prompts requiring user action or input are 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.
- Square brackets enclose the prompt default or, if empty, indicate no default exists for that prompt.

5.1.2 Generic Error Message Format

All device integrity tests follow a generic error message format, as follows:

```
   XXXXXX>x>tt:tt T#aaa E#bbb   U-ccc
<TEXT STRING DESCRIBING ERROR>

   FRU1-dddddd FRU2-dddddd
   MA-eeeeee
   EXP-yyyyyy
   ACT-zzzzzz

   Where:

    XXXXXX> is the appropriate device integrity test prompt.
    x> is the letter indicating the type of integrity test
 that was initiated:
     D> is the demand integrity test.
     A> is the automatic integrity test.
     P> is the periodic integrity test
    tt:tt is the current time.
    aaa is the decimal number denoting test that failed.
    bbb is the decimal number denoting error detected.
    ccc is the unit number of drive being tested.
    FRU1 is the most likely field replaceable unit (FRU).
    FRU2 is the next most likely FRU.
    dddddd is the name of field replaceable unit.
    MA is the media address.
    eeeeee is the octal number denoting offset within block.
    yyyyyy is the octal number denoting data expected.
    zzzzzz is the octal number denoting data actually found.
```

The first line of the error message contains general information about 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.

NOTE
If a P.ioj/c or M.std/2 module fails during the periodic ILMEMY tests, the FACILITY section of the crash code displays PRMEMY, which indicates the failure occurred during the periodic tests.

If a K.sdi, K.sti, or K.si module fails during the periodic K tests, the FACILITY section of the crash code displays PRKSDI, PRKSTI, or PRKSI, which indicates the failure occurred during the periodic tests.

5.2 ILRX33 — RX33 Device Integrity Tests

The ILRX33 exerciser runs a test of either of the RX33 drives attached to the HSC. ILRX33 runs concurrently with other HSC processes and uses the services of the HSC control program and the Diagnostic Execution Monitor (DEMON).

ILRX33 performs several writes and reads to verify the RX33 internal data paths and read/write electronics.

A scratch diskette is not required. ILRX33 does not destroy any data on the system software.

The exerciser tests only the RX33 and the data path between the P.ioj and the RX33. All other system hardware is assumed to be working properly.

ILRX33 verifies only a particular RX33 drive and controller combination is working or failing. Therefore, the test should not be used as a subsystem troubleshooting aid. This test does not support flags. If the test indicates a drive or controller is not operating correctly, replace the drive and/or controller. The controller is located on the memory module.

5.2.1 System Requirements

Hardware and software requirements include:

- P.ioj (processor) module with boot ROMs
- M.std2 memory/disk controller module
- RX33 controller with at least one working drive
- HSC system media
- Console terminal

5.2.2 Operating Instructions

Press CTRL/Y to get the HSC> prompt. Next, type either RUN ILRX33 or RUN DXn:ILRX33 to initiate the tests.

NOTE
The term DXn: refers to the RX33 disk drives (DX0: or DX1:).

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

5.2.3 Test Termination

ILRX33 can be terminated by pressing CTRL/Y. The test automatically terminates after reporting an error with one exception: if the error displayed is Retries Required, the test continues.

5.2.4 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 HSC control program.

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

5.2.5 Progress Reports

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

```
ILRX33>D>tt:tt Execution Complete

Where:
  tt:tt is the current time.
```

5.2.6 Test Summary

The ILRX33 test summary is contained in the following paragraph.

Test 001, Read/Write Test—Verifies that 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.

5.2.7 Error Message Example

All error messages produced by ILRX33 conform to the HSC device integrity test error message format (Section 5.1.2). 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 Error Messages

The following paragraphs list specific information about each of the errors produced by the ILRX33. 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**—Reported if ILRX33 is aborted by pressing 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 through 2399 decimal). Because the ILRX33 reads and writes disk block 001, it may indicate a software problem.
- **Error 006, Unknown Status STATUS=xxx**—Indicates ILRX33 received a status code it did not recognize. The octal value xxx represents the status byte received. RX33 reads and writes are performed for ILRX33 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 through a Software Performance Report (SPR). See Appendix B for detailed information on SPR submission.

- **Error 007, Data Compare Error**—Indicates data subsequently read back.

```
MA -aaaaaa
EXP-bbbbbb
ACT-cccccc

where:

 aaaaaa represents the address of the failing word
   within the block (512 bytes) that was read.
 bbbbbb represents the data written to the word.
 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.

5.3 ILTAPE–TU58 Device Integrity Test

The ILTU58 tests either of the TU58 drives attached to the HSC50. This test runs concurrently with other HSC processes and uses the services of the HSC control program and the Diagnostic Execution Monitor (DEMON). The test can be initiated with only the system tape installed.

Because the HSC50 operating system tests the TU58 every time it is used, ILTU58 performs only minimal testing. Several read and write operations are performed to test the internal data paths and the read/write circuitry of the TU58.

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

ILTU58 tests only the TU58 and the data path between the P.ioc and the TU58. All other system hardware is assumed to be working properly.

ILTU58 verifies that only a particular TU58 drive and controller combination is working or failing. Therefore, the test should not be used as a troubleshooting aid. This test does not support flags. If the test indicates that a drive or controller is not operating correctly, replace the drive and/or controller.

5.3.1 System Requirements

Hardware and software requirements include:

P.ioc module with boot ROMs
M.std memory/disk controller module
TU58 controller with at least one working drive
HSC system media

5.3.2 Operating Instructions

Press CTRL/Y to get the HSC> prompt. Then type RUN ILTU58 or RUN DDn::ILTU58 to initiate the tests.

5.3.3 Test Termination

ILTU58 can be terminated by pressing CTRL/Y.

5.3.4 Error Messages

The following error messages are issued:

- **Retries Required**—Indicates a Read or Write operation failed in the first attempt but succeeded on one of the retries. It may also mean that the tape media is degrading and should be replaced.
- **Operation Aborted**—Reported if the test is interrupted by CTRL/Y.
- **Write Protected**— Indicates the drive being tested has a write protected tape. Try again with Write enabled. If the tape is not write-protected, then the drive or controller is faulty.
- **No Cassette Mounted**—Indicates the drive does not have a cassette. Insert a cassette tape then repeat the test. Otherwise, the drive or the controller is faulty.
- **Hard I/O Error**—Indicates the program encountered a hard error while attempting to read or write the cassette tape.
- **Bad Record**—Indicates the program encountered a bad record while attempting to read or write the cassette tape.
- **Bad Opcode**—Indicates the program encountered an illegal opcode.
- **Bad Record**—Indicates the program encountered a bad record while attempting to read or write the cassette tape.
- **Seek Error**—Indicates a seek error status was set after a SEEK command.
- **Bad Unit #**—Indicates the diagnostic interface could not find the unit number specified. The drive may have been off line.
- **Failed Self-Test**—Indicates the HSC diagnostic interface failed.
- **End of Medium**—Indicates the end of the cassette tape.
- **Unknown Status**—Indicates the TU58 device integrity test received a status code it did not recognize. This status code is displayed to help determine the cause of the problem.
- **Data Compare Error**—Indicates that data subsequently read back did not match data written. The test writes a pattern to block 1, then reads block 1 and check that the pattern read is the same as the pattern written. If the patterns differ, this error message is displayed. This step is performed three times, each time with a different pattern. The patterns are:

  MA -aaaaaa—represents the address of the failing word.
  EXP-bbbbbb—represents the data written to the word.
  ACT-cccccc—represents the data read back from the word.

  Because this test only reads and writes block 1 of the cassette tape, all failures occur while trying to access physical block 1.

- **Illegal Device Name**—Indicates the user specified an illegal device name when prompted for the name of the drive to be tested. After displaying this error, the program again prompts for a device name. Enter one of the legal device names to continue the test.

5.4 ILMEMY — Memory Integrity Tests

The memory integrity test is designed to test HSC data buffers. This test can be initiated automatically or on demand. It is initiated automatically to test data buffers that produce a parity or nonexistent memory (NXM) error when in use by the HSC control program or any of the K modules.

Buffers that fail the memory test are removed from service by sending them to the disabled buffer queue. The disabled buffer accepts only 16 entries. When the buffer has accepted 16 entries, it acts as a first-in-first-out (FIFO) buffer.

NOTE
The contents of the disabled buffer queue are lost during a reboot of the HSC. As a result, all bad memory locations are lost.

Buffers sent twice to this test are also sent to the disabled buffer queue even if they did not fail the test. Buffers that pass the memory test and have not been tested previously are sent to the free buffer queue for further use by the HSC control program.

When the test is initiated on demand, any buffers in 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 HSC processes and uses the services of the HSC control program and the Diagnostic Execution Monitor (DEMON).

5.4.1 System Requirements

Hardware requirements include:

- P.ioj (processor) module with HSC boot ROMs, or P.ioc (processor) module with HSC50 boot ROMs.
- At least one M.std2 memory module (HSC) or the M.std memory module (HSC50).
- RX33 controller with one working drive (HSC) or TU58 controller with one working drive (HSC50).
- A console terminal for demand initiation only.

This program only tests Data Buffers located in the HSC Data memory. All other system hardware is assumed to be working.

Software requirements include:

- HSC control program (system diskette or tape)
- Diagnostic Execution Monitor (DEMON)

5.4.2 Operating Instructions

Press CTRL/Y to get the attention of the HSC keyboard monitor. The keyboard monitor responds with the prompt:

```
HSCxx>
```

Type RUN dev:ILMEMY to initiate the memory integrity test. This program has no user-supplied parameters or flags.

NOTE
ILMEMY tests only data memory buffers. Control/program memory errors typically cause a reboot of the HSC.

If the memory integrity test is not contained on the specified device (dev:), an error message is displayed.

5.4.3 Test Termination

ILMEMY can be terminated at any time by pressing CTRL/Y.

5.4.4 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.
```

5.4.5 Test Summaries

Test 001 receives a queue of buffers for testing. If the ILMEMY is initiated automatically, the queue consists of buffers from the suspect buffer queue.

When the HSC control program detects a parity or nonexistent memory (NXM) error in a Data Buffer, the buffer is sent to the suspect buffer queue. While in this queue, the buffer is not used for data transfers. The HSC periodic 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 in-line 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.

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 memory integrity 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 memory integrity test, it is returned to the free buffer queue for further use by the HSC control program. In this last case, the address of the buffer is saved in case the buffer again fails and is sent to the memory integrity test a second time.

When all buffers on the test queue are tested, the memory integrity test terminates.

5.4.6 Error Message Example

All error messages produced by the memory integrity test conform to the HSC integrity test error message format (Section 5.1.2). 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
```

When all buffers on the test queue are tested, the memory integrity test terminates.

5.4.7 Error Messages

The following list shows specific information about each of the errors displayed by the memory integrity 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 that the memory integrity test detected no error. Because this is the first time the buffer was sent to the memory integrity test, it is returned to the free buffer queue for further use by the HSC control program. The address of the buffer is stored by the memory integrity 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.
- **Error 004, NXM Trap (Buffer Retired)** – Indicates an unknown address or memory location is being referenced.
- **Error 005, Can't Allocate Timer, Test Aborted**—Indicates the program failed to allocate a timer for SLEEP.

5.5 ILDISK — DISK Drive Integrity Tests

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

ILDISK runs in parallel with disk I/O from a host CPU. However, the drive being diagnosed cannot be on line to any host. ILDISK can be initiated upon demand through the console terminal or automatically by the HSC control program when an unrecoverable disk drive failure occurs.

ILDISK is automatically invoked by default whenever a drive is declared inoperative, with one exception: if a drive is declared inoperative while in use by an integrity test or utility.

Automatic initiation of ILDISK can be inhibited by issuing the SETSHO command SET AUTOMATIC DIAGNOSTICS DISABLE. If the SET AUTOMATIC DIAGNOSTICS command is issued and DISABLE is specified, 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 HSC disk server.

1. Drive unknown (to the HSC disk server)—It is either unable to communicate with the HSC or was declared inoperative when it failed while communicating with the HSC. 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/K.si and the disk drive and command the drive to run its integrity tests. 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 HSC disk server, that is, 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.

To find the drives known to the Disk and Tape Servers, type the SETSHO command SHOW DISKS or SHOW TAPES.

5.5.1 System Requirements

The software requirements for this test reside on the system media and include:

- HSC executive (CRONIC)
- ILDISK.DIA program
- Diagnostic process (DEMON)
- K.sdi/K.si microcode (installed with the K.sdi/K.si module)

Hardware requirements include:

- Disk drive
- Disk data channel, connected by an SDI cable

The test assumes the I/O Control Processor module and the memory module are working.

Refer to the disk drive documentation to interpret errors that occur in the drive's integrity tests.

5.5.2 Operating Instructions

The following steps are used to initiate ILDISK.

NOTE
To prevent access from another HSC, deselect the alternate port switch on the drive to be tested. The alternate port switch is the drive port switch allowing alternate HSC access to the drive.

NOTE
The HSC system RX33 must be present at all times.

1. Press CTRL/Y.
2. The following prompt appears:

```
HSCxx>
```

3. Type RUN dev:ILDISK and enter a carriage return.
4. Wait until ILDISK is read from the system software load media into the HSC Program memory.
5. Enter parameters after ILDISK is started. Refer to Section 5.5.5.

5.5.3 Availability

If the software media containing ILDISK is not loaded when the RUN ILDISK command is entered, an error message is displayed. Insert the software media containing ILDISK and repeat Section 5.5.2.

5.5.4 Test Termination

ILDISK is terminated by pressing CTRL/Y or CTRL/C. Test termination may not take effect immediately because certain parts of the program cannot be interrupted. An example would be during SDI commands. If an SDI DRIVE DIAGNOSE command is in progress, interfering with the disk drive at this time can cause the program to wait 2 minutes before aborting.

5.5.5 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 through the HSC diagnostic interface. If the unit is acquired successfully, ILDISK next prompts for the drive integrity test to be executed.

If the acquire fails, one of the following conditions is encountered:

1. The specified drive is unavailable. This indicates the drive is connected to the HSC, but is currently on line to a host CPU or an HSC utility. On-line drives cannot be diagnosed. ILDISK 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 disk data channel port is broken and cannot communicate with the disk functional software.
   - The drive was previously communicating with the HSC but a serious error occurred, and the HSC has ceased communicating with the drive (marked the drive as inoperative).

   In either case, ILDISK prompts for a requestor number and port number. Refer to Section 5.5.6.

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

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

Answering N causes the drive to execute its entire integrity test set. Answering Y executes a single drive integrity test. If a single drive integrity test is selected, the test prompts:

```
DRIVE TEST NUMBER (H) [] ?
```

Enter a number (in hex) specifying the drive integrity test 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. Pressing RETURN without entering a number runs the test once.

5.5.6 Specifying Requestor and Port

Drives unknown to the HSC disk functional software are tested by specifying the requestor number and port number of the drive. The requestor number is any number 2 through 9 (HSC) or 2 through 7 (HSC50 [modified] or HSC50) specifying the disk data channel connected to the drive under test.

The port number is 0 through 3; it specifies 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 HSC bulkhead connector, and 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 pressing CTRL/Y. Type SHOW DISKS in response to the HSC 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 the 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 HSC 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 5.5.5. If the requestor and port do match a known drive, ILDISK reports error 08.

5.5.7 Progress Reports

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

5.5.8 Test Summaries

Test summaries for ILDISK follow.

- **Test 0, Parameter Fetching**—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**—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/K.si test 10).
- **Test 02, Check for Clocks and Drive Available**—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**—Issues a 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**—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/command data and the read/response data lines of the SDI cable.

- **Test 05, Run Drive Integrity Tests**—Directs the drive to run its internal integrity test. The drive is commanded to run a single integrity test or its entire set of integrity tests depending upon user response to the prompt:

  ```
  Run a Single Drive Diagnostic ?
  ```

  Before commanding the drive to run its integrity tests, the drive is brought on line to prevent the drive from giving spurious Available indications to its other SDI port. The drive integrity tests 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 integrity tests. This can require 2 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**—Sends a DISCONNECT command to the drive and then issues a GET LINE STATUS internal command to the K.sdi/K.si 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**—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**—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 On Line**—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 On-line state. Read/Write Ready is expected to be true; Available and Attention are expected to be false.

- **Test 10, Recalibrate and Seek**—Issues a RECALIBRATE command to the drive. This ensures the disk heads start from a known point on the media. Then 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**—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 On Line**—Attempts to bring the disk drive to the On-line state. Test 12 is executed only for drives known to the HSC 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 HSC diagnostic interface to bring the drive on line.

  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 on-line is performed through the HSC diagnostic interface, invoking the same software operations a host invokes to bring a drive on line. An on-line at this level constitutes more than just sending a SDI ONLINE command. The FCT and RCT of the drive also are 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 on-line operation fails. If test 12 fails, ILDISK skips the remaining tests and goes to test 17.

- **Test 13, Read Only I/O Operations Test**—Tests that all read/write 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 or integrity test use.) Test 13 attempts to read at least one sector on every track in the read only area of the drive's 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 when test 13 completes.
- **Test 14, I/O Operations Test (Read/Write 512 byte format)**—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 or integrity test 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 14 completes, ILDISK proceeds to test 17.
- **Test 17, Terminate ILDISK**—Is the ILDISK termination routine. The following steps are performed:
  1. If the drive is unknown to the HSC 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 integrity test use. This returns the drive to the Available state.
  4. The drive is reacquired for exclusive integrity test 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.

5.5.9 Error Message Example

All error messages produced by the disk drive integrity tests conform to the HSC integrity test error message format (Section 5.1.2). 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
```

5.5.10 Error Messages

Messages produced by ILDISK are described in the following list:

- **Error 01, DDUSUB Initialization Failure**—The HSC diagnostic interface did not initialize. Error 01 is not recoverable 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 5.5.5).
- **Error 03, Drive Unavailable**—The selected disk drive is not available for ILDISK.
- **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 through 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 5.5.6 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 (HSC) or 2 through 7 (HSC50), 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/K.si). Specify a requestor that contains a disk data channel.
- **Error 08, Specified Post 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 through requestor number and port number.
- **Error 09, Drive Can't Be Brought On Line**—A failure occurred when ILDISK attempted to bring the specified drive on line. One of the following conditions occurred:
  1. Unit Is Off Line—The specified unit went to the Off-line state and now cannot communicate with the HSC.
  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 HSC, both with the same unit number.
  4. Unknown Status from DDUSUB—The HSC diagnostic interface returned an unknown status code when ILDISK attempted to bring the drive on line. Refer to error 04 for related information on this error.
- **Error 10, K.sdi Does Not Support Microdiagnostics**—The K.sdi/K.si connected to the drive under test does not support microdiagnostics. This indicates the K.sdi/K.si microcode is not at the latest revision level. This is not a fatal error, but the K.sdi/K.si 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/K.si 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 integrity test 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 only.) 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 5.5.11.

- **Error 15, Read/Write Ready Not Set in On-line 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 On-line state, but the Read/Write Ready bit was not set in the status.

  This could be caused by a failing disk drive, bad R/W logic, or bad software media.

- **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 integrity test or utility is active on this HSC, submit an SPR. See Appendix B for detailed information on SPR submission.

- **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 HSC 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 dialog 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 HSC probably will 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 dialog 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 dialog control block (DCB) 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 On-line 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 HSC disk functional software to declare the drive inoperative, so 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 integrity test use.
  3. Unknown Status from DDUSUB—Refer to Error 04.

  The drive may be allocated to an alternate HSC. Check the drive port lamp to see if this caused the error.

- **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 line 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 HSC diagnostic interface. This indicates a software error and should be reported through a Software Performance Report (SPR). See Appendix B for detailed information on SPR submission.
  2. Could Not Acquire Control Structures—The HSC diagnostic interface could not acquire sufficient control structures to perform the operation.
  3. Could Not Acquire Buffer—The HSC diagnostic interface could not acquire a buffer needed for the operation.
  4. Unknown Status from DDUSUB—The HSC diagnostic interface returned an unknown status code. Refer to Error 04.

  NOTE
  Retry ILDISK during lower HSC activity for the second and third problems 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 HSC diagnostic interface notified ILDISK that no I/O operation was in progress. This error may have been induced by a hardware failure, but it actually indicates a software problem, and the error 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 Off Line**—The drive, previously acquired by the integrity test, is now unknown to the disk functional code. This indicates the drive spontaneously went off line 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 HSC diagnostic interface. However, the specified requestor does not have a control area. This indicates a software problem and should be reported through 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 HSC 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 integrity tests and is not in use by the HSC.

  Open the HSC 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 HSC does not contain a requestor in the specified position. If a requestor is in the module slot, check that the red LED on the lower front edge of the module is lit. If so, the requestor failed and was disabled by the HSC. If the red LED is not lit, a software problem exists and should be reported through 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 or integrity test 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 5.5.11.
- **Error 34, Drive Diagnostic Detected Error**—The SDI verification test directed the disk drive to run an internal integrity test. The drive indicated the integrity test 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 documentation for that drive.
- **Error 35, Drive Diagnostic Detected Fatal Error**—The SDI verification test directed the disk drive to run an internal integrity test. The drive indicated the integrity test 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 integrity test 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 issued a SEEK command to the drive which resulted in an unexpected ATTENTION condition. The drive status is displayed with the error report. Refer to the service manual for that drive.
- **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 also was asserted. Read/Write Ready should never be asserted when a drive is in the Available state.
- **Error 42, Available Set in On-line 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 On-line Drive**—The SDI verification routine issued an ONLINE command to the drive. The drive entered the On-line 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 K.sdi/K.si cable 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 HSC disk software tables. The drive may have changed state and disappeared while ILDISK was running. This error also can be caused by inconsistencies in HSC 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 HSC.

- **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 through the SDI interface. To perform the SDI tests, ILDISK must take exclusive control of the Drive State Area; otherwise, the HSC 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 originally found it, 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 HSC 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 (through 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 Off-line 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 read/write 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 read/write 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 cannot complete due to lack of required memory structures. To perform certain drive tests ILDISK needs to acquire timers, a dialog 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/K.si 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**—K.sdi/K.si detected an error during a Format operation. Each error bit set in the fragment request block (FRB) is translated into a text message that accompanies the error report.

- **Error 57, Wrong Structure on Completion Queue**—While formatting, ILDISK checks each structure returned by the K.sdi/K.si 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/K.si 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/K.si 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 operation 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/K.si.
- **Error 61, Format Operation Timed Out**—The K.sdi/K.si 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/K.si, or the K.sdi/K.si may be unable to sample the 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/K.si 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/K.si with no error indicated. However, the disk rotational access table (DRAT) for the Format operation was never returned, indicating a probable K.sdi/K.si 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. If this error is detected, ILDISK discontinues testing to prevent inadvertent interaction with the other disk drive.
- **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 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 read/write DBN area to contain at least one good block that can be read without error. An Error 67 indicates a good block was not found on the first track of the read/write DBN area, so the formatting tests are skipped.

5.5.11 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

5.6 ILTAPE — TAPE Device Integrity Tests

The following tests can be initiated through ILTAPE:

- Tape formatter-resident integrity tests
- Functional test of the tape transport
- Full test of the K.sti/K.si interface

  When a full test is selected, the K.sti/K.si microdiagnostics are executed, line state is verified, an ECHO test is performed, and a default set of formatter tests is executed. See the DRIVE UNIT NUMBER prompt in the Section 5.6.3 for information on initiating a full test. Detected failures result in fault isolation to the FRU level.

Three types of tape transport tests are listed below. See Section 5.6.6 for a summary of each.

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

Hardware requirements necessary to run ILTAPE include:

- HSC subsystem with K.sti/K.si
- STI-compatible tape formatter
- TA78, TA81, or other DSA tape drive (for transfer commands only)
- Console terminal
- RX33 disk drive or equivalent (HSC)
- TU58 tape device or equivalent (HSC50)

In addition, the I/O control processor, program memory, and control memory must be working.

Software requirements necessary to run ILTAPE include:

- CRONIC
- DEMON
- K.sti/K.si microcode (installed with the K.sti/K.si module)

5.6.1 Operating Instructions

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

1. Boot the HSC.

   Press the Init button on the HSC OCP. The following message appears:

   ```
   INIPIO-I Booting...
   ```

   The boot process takes about 1 minute, and then the following message appears:

   ```
   HSC Version xxxx Date Time  System n
   ```

2. Press CTRL/Y.

This causes the KMON prompt:

```
HSC>
```

3. Type R DXn:ILTAPE.

This invokes the tape device integrity test program ILTAPE. The DXn is the HSC device name. The n refers to the unit number of the specific HSC drive. For example, DX1: refers to RX33 Drive 1 (HSC) and DD1: refers to TU58 Drive 1 (HSC50). The following message appears:

```
ILTAPE>D>hh:mm Execution Starting
```

5.6.2 Test Termination

The test can be terminated by pressing CTRL/C. Certain errors that occur during execution will cause ILTAPE to terminate automatically.

5.6.3 User Dialog

The following paragraphs describe ILTAPE/user dialog during execution of ILTAPE. Note that the 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) []?
```

To run formatter tests or transport tests, enter Tnnn, where nnn is the MSCP unit number (such as T316).

For 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 2 through 9, with no default value.

```
ENTER PORT NUMBER (0-3) []?
```

Enter the port number. The port number must be 0, 1, 2, or 3 with no default value. After this prompt is answered, ILTAPE executes the K.sti/K.si interface test.

```
EXECUTE FORMATTER DIAGNOSTICS (YN) [Y]?
```

Enter RETURN to execute formatter tests. The default is Y. Entering N will not run formatter tests.

```
MEMORY REGION NUMBER (H) [0]?
```

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

```
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 dialog 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]?
```

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 HSC device. User sequences are described in Section 5.6.4. 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 5.6.4 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 1 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 bits per inch (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 concerns 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) ssss ips
   (2) ssss ips
   (3) ssss ips
   (4) ssss ips
```

The supported speed in inches per second is shown as ssss. The maximum number of supported speeds is 4. Thus, n cannot be greater than 4. 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
```

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

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

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

The next prompt asks for the data pattern.

```
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 3. If the response is 0, the following prompts appear:

```
HOW MANY DATA ENTRIES (D) []?
```

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

```
DATA ENTRY (H) []?
```

Enter the data pattern word (in hex), 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. The maximum record size that can be specified is 12288.

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.

5.6.4 User Sequences

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

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

To initiate the user sequence dialog, type 0 in response to the prompt:

```
FUNCTIONAL TEST SEQUENCE NUMBER (D) [1]?
```

The following paragraphs describe the ILTAPE user dialog 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 INPUT STEP prompt 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 HSC load device 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 <BOLD>(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 HSC device as sequence number 3 and can be recalled at a later execution of ILTAPE.

5.6.5 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 test is indicated by the following message on the terminal:

```
    TEST nnnn DONE
```

The formatter test number is represented by nnnn.

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

5.6.6 Test Summaries

The following sections summarize the tests contained in ILTAPE.

5.6.6.1 Interface Test Summary

This portion of ILTAPE tests the standard tape interface (STI) of a specific tape data channel and port. It also performs low-level testing of the formatter by interfacing to the K.sti/K.si drive service area (port) and executing various Level 2 STI commands. The testing is limited to dialog operations; no data transfer is done. The operations performed are DIAGNOSE, READ MEMORY, GET DRIVE STATUS, and READ LINE STATUS.

K.sti/K.si microdiagnostics are executed to verify the tape data channel. A default set of formatter tests (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 channels, the STI set, or the formatter. The STI set includes a small portion of the K.sti/K.si module and the entire STI (all connectors, 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/K.si (least probable)

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

5.6.6.2 Formatter Test Summary

Formatter tests are executed out of a formatter memory region selected by the user. Refer to the tape drive service manual for a description of the formatter tests. Failures detected identify the formatter as the FRU.

5.6.6.3 User Sequence Test Summary

User sequences are used to exercise the tape transport. The particular sequence is totally user-defined. Refer to Section 5.6.4.

5.6.6.4 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 the tape at all supported densities. Using a user-selected record size, the canned sequence 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.

5.6.6.5 Streaming Sequence Test Summary

The streaming sequence is a fixed sequence that 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 pressing 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 test programs.

5.6.7 Error Message Example

ILTAPE conforms to the test generic error message format (Section 5.1.2). An example of an ILTAPE error message follows:

```
ILTAPE>D>09:31 T # 011 E # 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. Table 5–1 defines the ILTAPE test levels.

Table 5–1 ILTAPE Test Levels

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

The optional text is dependent upon the type of error.

5.6.8 Error Messages

The following list describes ILTAPE error messages.

- **Error 01, Initialization Failure**—Tape path software interface cannot be established due to insufficient resources (buffers, queues, timers, and so forth).
- **Error 02, Selected Unit Not a Tape**—Selected drive is not known to the HSC as a tape.
- **Error 03, Invalid Requestor/Port Number**—Selected requestor number or port number is out of range or requestor selected is not known to the system.
- **Error 04, Requestor Not A K.sti**—Selected requestor is not known to the system as a tape data channel.
- **Error 05, 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 06, Requested Device Unknown**—Device requested is not known to the tape subsystem.
- **Error 07, Requested Device Is Busy**—Selected device is on line to another controller or host.
- **Error 08, Unknown Status from Tape Diagnostic Interface**—An unknown status was returned from the test software interface TDUSUB.
- **Error 09, 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 HSC device. Check to see if the HSC load device is write-protected. The prompt calls for a user sequence number. To break the loop of reprompts, press 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 test running in the formatter detects an error. Any error text from the formatter is displayed.
- **Error 14, Formatter Diagnostic Detected Fatal Error**—A test running in the formatter detects a fatal error. Any error text from the formatter is displayed.
- **Error 15, Load Device Read Error**—While attempting to read a user sequence from the load device, a read error was encountered. Ensure a sequence has been stored on the load device as identified by the user sequence number. The program reprompts for a user sequence number. To break the loop of reprompts, press 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 through 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, Error 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 On-line Formatter**—After successful completion of a formatter ONLINE command to the formatter, the Available signal is set.
- **Error 29, Load Device Error—File Not Found**—During the user sequence dialog, ILTAPE was unable to locate the sequence file associated with the specified user sequence number. Ensure load device media is properly installed. The program reprompts for a user sequence number. To break the loop of reprompts, press 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 through 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 On Line to Another Controller**—The device identified by the selected unit number is either unknown to the system or it is on line to another controller. Verify the selected unit number is correct and run ILTAPE again.

5.7 ILTCOM — Tape Compatibility Test

ILTCOM tests the compatibility of tapes that may have been written on different systems and different drives with STI compatible drives connected to an HSC through 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 file groups (called bunches) of specific data pattern records.

Each bunch contains a header record and several data records of different sizes and 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 5–2 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
4	Processor serial number	6 bytes	123456
5	Date	6 bytes	093083
6	Comment [1]	18 bytes	Comment

[1]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, and so forth, through the 33rd and 66th records containing 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 data patterns.

The data patterns used are shown in Table 5–3.

Table 5–3 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	

5.7.1 System Requirements

The hardware requirements necessary to run ILTCOM include:

- An HSC subsystem with K.sti/K.si
- STI-compatible tape formatter
- Tape 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.

ILTCOM software requirements include:

- CRONIC
- DEMON
- K.sti/K.si microcode
- TFUNCT
- TDUSUB

5.7.2 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. 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 appears:

   ```
   INIPIO-I Booting...
   ```

   The boot process can take several minutes, and then the following message appears:

   ```
   HSC Version xxxx Date Time System n
   ```

2. Press CTRL/Y.

   This causes the KMON prompt to appear:

   ```
   HSC>
   ```

3. Type R DXn:ILTCOM. The variable n equals the number of the RX33 drive containing the HSC system diskette. When running ILTCOM on an HSC50, use DDn: to access the TU58 tape drive.

   This invokes the compatibility test program ILTCOM. The following message appears:

   ```
   ILTCOM>D>hh:mm Execution Starting
   ```

The subsequent program dialog is described in the next section.

5.7.3 Test Termination

ILTCOM is terminated normally by selecting the exit function (EXIT) or by pressing CTRL/Y or CTRL/C. Certain errors that occur during execution cause ILTCOM to terminate automatically.

5.7.4 Parameter Entry

ILTCOM allows the writing, reading, listing, or modifying of compatibility tapes. The following describes the user dialog 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) four characters of the density desired (1600 for 1600 bpi).

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

Enter the function by typing the characters that uniquely identify the desired function (for instance, REA for read).

The subsequent dialog 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 to proceed with the initial write. The default is no, in which case program control is continued at the function selection prompt. If the response is yes, the following prompt occurs.

  ```
  WRITE ENTIRE TAPE (YN) [N]?
  ```

  Type Y if the entire tape is to be written. Writing of bunches begins at the current tape position and continues to physical EOT. Type the default N 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 has 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 HSC SERIAL NUMBER 005993 ON 09-18-84
  ```

  The number of bunches to be read is user selectable. All reads are from beginning of tape (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.

  ILTCOM does not allow the user to erase all bunches. At least one bunch must remain. For example, with five bunches on the tape, only four bunches can be erased.

  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 0 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.

5.7.5 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 due to data compare errors or unexpected formats are found, they are reported. ILTCOM makes no attempt to isolate faults during execution; it merely reports incompatibilities and other errors as they occur.

5.7.6 Error Message Example

ILTCOM conforms to the test generic error message format (Section 5.1.2). An example of an ILTCOM error message follows:

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

5.7.7 Error Messages

The following are the ILTCOM error messages.

- **Error 01, Initialization Failure**—Tape path cannot be established due to insufficient resources.
- **Error 02, Selected Unit Not a Tape**—User selected a drive not known to the system as a tape.
- **Error 03, Command Failure**—A command failed during execution of ILTCOM. The command in error may be one of several types (MSCP level, STI Level 2, and so forth). 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 05, Specified Unit Not Available**—The selected unit is on line to another controller.
- **Error 06, Specified Unit Cannot Be Brought On Line**—The selected unit is off line or not available.
- **Error 07, Specified Unit Unknown**—The selected unit is unknown to the HSC configuration.
- **Error 08, Unknown Status from TDUSUB**—An unknown error condition returned from the software interface TDUSUB.

- **Error 09, Error Releasing Drive**—After completion of execution or after an error condition, the tape drive could not successfully be 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. ACTual and EXPected values are displayed in the optional text of the error message.

5.8 ILEXER — Multidrive Exerciser

ILEXER 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 issues 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.

ILEXER runs with other processes on the HSC subsystem. It is loaded from the RX33 or TU58 and uses the services of the Diagnostic Execution Monitor (DEMON) and the HSC control software.

5.8.1 System Requirements

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

1. HSC subsystem:
   a. Console terminal
   b. P.io
   c. K.sdi, K.sti, or K.si
   d. Program, Control, and Data memories
   e. RX33 (HSC) system diskette or TU58 (HSC50) system tape 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/K.si microcode

      and/or

d. K.sti/K.si 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 interfaces are not available (K.sdi, K.sti, or K.si).

5.8.2 Operating Instructions

Perform the following steps to initiate ILEXER:

1. Press CTRL/Y.
2. The HSC responds with:

```
HSCxx>
```

3. Type RUN DX0:ILEXER.

DX0: refers to RX33 Drive 1. On an HSC50, DD1: refers to the TU58 Drive 1.

The system loads the program from the specified local HSC load media (any appropriate media with the image ILEXER in an RT 11 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 test 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
```

5.8.3 Test Termination

Upon completion of the exercise on each selected drive, reporting of any errors found, and displaying 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 pressing 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.

5.8.4 Parameter Entry

The parameters in ILEXER follow the format:

```
PROMPT DESCRIPTION (DATATYPE) [DEFAULT]?
```

Where:

- 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 —Number (in hex)

- 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) [] ?
```

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 that corresponds to the number printed on the drive's unit plug. 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 through 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 on line to a host CPU or HSC utility. On-line 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/K.si port is broken and cannot communicate with the disk functional software.
   - The drive was 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.

When a disk drive is specified, one set of prompts is presented. When a tape drive is selected, an entirely different set of prompts is presented. Pressing CTRL/Z at any time during parameter input selects the default values for the remaining parameters.

After a drive is selected and ILEXER has both acquired the drive and brought it on line, or if a nondefaultable parameter is encountered, the following prompt appears:

```
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.

5.8.5 Disk Drive Prompts

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

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

Answering Y to this and the next prompt directs ILEXER to perform testing in the user data area. It is the operator's responsibility to see that 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 or integrity tests (DBN area). When testing is confined to the DBN area, the following five prompts are not displayed.

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

Answering N causes the DBN area to be exercised. Answering Y allows the exercise to take place in the user data area of the disk.

```
DO YOU WANT BBR (Y/N) [Y]?
```

Answer N if the drive is suspected as bad. If you are positive the drive is good, answer Y to enable BBR.

```
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 begins 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 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 is 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:

```
MINIMUM DISK TRANSFER LENGTH IN SECTORS (1 TO 400) [10]?
MAXIMUM DISK TRANSFER LENGTH IN SECTORS (1 TO 400) [10]?
```

These prompts request the range of size in sectors of each data transfer issued to the disk drives. The default disk transfer length is 10 sectors.

Once the preceding parameters are entered, ILEXER continues with the prompts listed as global user prompts (Section 5.8.7).

5.8.6 Tape Drive Prompts

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

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

Answering N results in a reprompt for the drive unit number. Answering Y 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 5.8.8.

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

The response to this prompt is 1, 2, or 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 documentation 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 documentation for available speeds.

```
RECORD LENGTH IN BYTES (1 to 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. Answering Y causes the following prompt:

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

Answering Y selects data compares to be performed on every tape Read operation. Answering N causes data compares to be performed on 15 percent of the tape reads.

```
ANOTHER DRIVE (Y/N) []?
```

If answered with Y, the prompts beginning with the prompt for DRIVE UNIT NUMBER are repeated. If answered with N, the global prompts in Section 5.8.7 are presented. This prompt has no default, allowing the operator to default all other prompts and to be able to set up another drive for this pass of ILEXER.

5.8.7 Global 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 minutes. 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]?
```

The number of hard errors allowed for the drives being exercised can be specified. The limit can be set from 0 to 20. 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 attempted 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

The prompt next calls for:

```
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 5.8.12. This report format is intended for use by small hand-held terminals.

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

Answering Y enables soft error reports. By default, the operator does not see any soft error reports specific to the number of retires required on a tape I/O operation.

Answering N 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]?
```

ILEXER expects a 4 character hex value as the answer to this prompt.

5.8.8 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.

The following are data patterns for disks:

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

The following are data patterns for tapes:

Pattern 16 Alternating one and zero bits 125252 125252	Pattern 17 All ones	Pattern 18 Alternating bytes of all ones	Pattern 19 Alternating bytes of all ones and all zeros
Pattern 20 Alternating two bytes ones and two bytes zeros	Pattern 21 Alternating four bytes ones and four bytes zeros	Pattern 22 Alternating three bytes of ones and one byte zeros	Pattern 23 Alternating bytes of ones and zeros with high byte in pattern number also zero*

*Pattern 23 is used with odd record sizes.

5.8.9 Setting/Clearing Flags

The Enable Soft Error Report display prompt in Section 5.8.7 allows the operator to inhibit the display of soft error reports. No other error reports can be inhibited.

5.8.10 Progress Reports

ILEXER has three basic forms of progress reports: the Data Transfer error report, the performance summary, and the Communication error report.

1. The Data Transfer error report is printed each time an error is encountered in one of the drives being tested.
2. The Performance summary is printed when ILEXER completes a pass on each drive being exercised or when the operator terminates the pass through a CTRL/Y. This Performance summary also is printed on a periodic basis during the execution of ILEXER.
3. 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.

5.8.11 Data Transfer Error Report

The ILEXER Data Transfer error report is printed on the terminal each time a data transfer error is found during execution 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 ILEXER is a data compare error. In this case, ILEXER has performed a check and found an error during the compare, resulting in an ILEXER error report.

5.8.12 Performance Summary

The Performance summary is printed on the terminal at the end of a manually terminated testing session, or after the specified number of minutes for the periodic Performance summary. This report provides statistical data which is 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 correctible transfers. For tape drives being exercised by ILEXER, an additional report breaks down the software correctible errors into eight different categories.

The frequency of report display is altered in the following fashion:

1. Press CTRL/G during the execution of ILEXER.
2. The following prompt is displayed:

```
Interval time for performance summary in seconds (D) [30]?
```

The format of the Performance summary follows:

```
                     PERFORMANCE SUMMARY (DEFAULT)

UNIT  R  SERIAL      POSI   KBYTE      KBYTE     HARD  SOFT  SOFTWARE
NO    --  NUMBER     TION   READ       WRITTEN   ERROR ERROR CORRECTED

Dddd   HHHHHHHHHHHH ddddd ddddddddd ddddddddd ddddd ddddd  ddddd
Tddd   HHHHHHHHHHHH ddddd ddddddddd ddddddddd ddddd ddddd  ddddd
```

A Performance summary is displayed for each disk drive and tape drive active on the HSC. The following list explains the performance summary:

- **Unit Number**—The unit number of the drive. D is for disk, T is for tape. The number is reported in decimal.
- **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.
- **Serial Number**—The serial number (in hex) for each drive.
- **Position**—The number of seeks.
- **Kbyte Read**—The number of Kbytes read by ILEXER on each drive.
- **Kbyte Written**—The number of Kbytes written by ILEXER.
- **Hard Error**—The number of hard errors reported by ILEXER for a particular drive.
- **Soft Error**—The number of soft tape errors reported by the exerciser if enabled by the operator.
- **Software Corrected**—The number of correctible ECC errors encountered by ILEXER. Only ECC errors above the specific drive ECC error threshold are reported through normal functional code error reporting mechanisms. ECC errors below this threshold are not reported through an error log report, but are included in this count maintained by ILEXER.

If any tape drives are 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
```

An explanation of the summary columns follows:

- **Media Error**—The number of bad spots detected on the recording media.
- **Double TRKERR**—The number of double-track errors encountered during a read or write forward.
- **Double TRKREV**—The number of double-track errors encountered during a read reverse or write reverse.
- **Single TRKERR**—The number of single-track errors detected during a read or write in the forward direction.

- **Single TRKREV**—The number of single-track errors encountered during a read reverse or write reverse.
- **Other Err A-C**—Reserved for future use.

```
                    PERFORMANCE SUMMARY (NARROW)

ILEXER>D>PER SUM
D[T]ddd
SN HHHHHHHHHHHH
P ddddd
R dddddddddd
W dddddddddd
HE ddddd
SE ddddd
SC ddddd
```

This report is repeated for each drive tested.

If 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
```

5.8.13 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 test specifically designed to isolate the failure (ILDISK or ILTAPE).

5.8.14 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**—Multidrive Exerciser is the main program within ILEXER. It is responsible for calling all other portions of ILEXER. It obtains the buffers and control structures for the exerciser. It verifies that 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 operator. It brings all selected drives on line 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 that 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:

  a. If the drive is read only and data compare is not requested, a Read operation is queued to the drive.

  b. If read only and data compare (occasional) are requested, a Read operation is queued along with a random choice of compare/not-compare.

  c. If read only and data compare (always) are requested by the operator, a READ-COMPARE command is queued to the drive.

  d. If write only is requested, and data compare is not, then a write request is queued up to the disk drive.

  e. If write only and data compare (occasional) are requested, a Write operation is queued along with a random choice of compare/not-compare.

  f. If write only and data compare (always) are requested, a WRITE-COMPARE command is queued to the drive.

  g. If only data compare (occasional) is requested, then a random selection of read/write and compare/not-compare is done.

  h. 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.

- **Test number 9, RANSEL**—RANSEL is the part of the tape exerciser that 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:

  a. No reads when no records exist before or after the current position.

  b. No writes when records exist after the current position.

  c. No position of record when no records exist before or after the current position.

  d. Reverse commands are permitted on the drive when 16 reverse commands previously have 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.

- **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.
- **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.
- **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.

5.8.15 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 test error message format (Section 5.1.2).

5.8.15.1 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
```

The error message is an ASCII string describing the type of error discovered.

5.8.15.2 Data Compare Error Format

A data compare error occurs when an error is detected during the exercise of a particular drive.

The two formats for the data compare error are data word compare error and pattern word error.

A data word 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 is the current system time.
 T is the test number in the exerciser.
 E corresponds to the error number.
 U is the unit number for which the error is being reported.
 MA is the media address (block number) where the error occurred.
 EXP is the EXPected data.
 ACT is the data (or code) actually received.
 MSCP STATUS CODE is 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.
```

5.8.15.3 Pattern Word Error Format

The format for the pattern word error is slightly different from the data word 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.

5.8.15.4 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 is the time stamp for the start of ILEXER.
  T is the test number in the exerciser.
  E corresponds to the error number.
  U is the unit number for which the error is being reported.
  Error Description is an ASCII string describing the
    error encountered.
  Optional Data lines -- A maximum of eight optional lines per report.
```

5.8.16 Error Messages

The following section lists the informational and error messages and explains the cause of the error. A typical error message is:

```
ILEXER>D>09.32 T#006 E#204 U-T00100
ILEXER>D>Comm Error: TBUSUB call failed
```

5.8.16.1 Informational Messages

The informational messages are not fatal to the exerciser and are intended only to:

- Alert the user to incorrect input to parameters
- Indicate missing interfaces
- Provide user information

The following list describes informational messages.

- **Number Must Be Between 0 and 15**—Reported when the user entered an erroneous value for the data pattern on a disk.
- **Pattern Number Must Be Within Specified Bounds**—Reported when the operator tries to specify a disk pattern number for a tape.
- **You May Enter at Most 16 Words in a Data Pattern**—Reported if the operator specifies more than 16 words for a user-defined pattern, and the operator is reprompted for the value.
- **Starting LBN Is Either Larger than Ending LBN or Larger than Total LBN on Disk**—Reprompts for the correct values. The operator selected a starting block number for the test which is greater than the ending block number selected, or it is greater than the largest block number for the disk.
- **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.
- **Disk Interface Not Available**—Indicates the disk functionality is not available to exercise disk drives. This means the K.sdi/K.si is not available or not operable.
- **Tape Interface Not Available**—Indicates the tape functionality is not available to exercise tape drives. This means the K.sti/K.si is not available or not operable.
- **Please Wait—Clearing Outstanding I/O**—Printed when the operator presses CTRL/Y to stop ILEXER. All outstanding I/O commands are aborted at this time.

5.8.16.2 Generic Errors

The following list indicates the error number, text, and cause of errors displayed by ILEXER.

- **Error 01, No Disk or Tape Functionality...Exerciser Terminated**—Neither the K.sdi, K.sti, nor K.si interfaces are available to run the exercise. This terminates ILEXER.
- **Error 02, Could Not Get Control Block for Timer—Stopping ILEXER**—ILEXER could not obtain a transmission queue for a timer. This should occur only on a heavily loaded system and is fatal to ILEXER.
- **Error 03, Could Not Get Timer—Stopping ILEXER**—The exerciser could not obtain a timer. Two timers are required for ILEXER. This should only occur on a loaded system and is fatal to ILEXER.
- **Error 04, 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.

- **Error 05, 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.
- **Error 06, 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 on line.
  2. The cables to the K.sdi, K.sti, or K.si are loose.
- **Error 07, 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.
- **Error 08, Drive Is Unavailable—Choose Another Drive**—This may be the result of:
  1. The drive port button is disabled for that port.
  2. The drive is on line to another controller.
  3. The drive is not able to communicate with the controller on the port selected.
- **Error 09, Drive Cannot Be Brought On Line**—ILEXER was unable to bring the selected drive on line. 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 on line.
- **Error 12, 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.
- **Error 13, 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.
- **Error 14, 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.
- **Error 15, Record Length Larger than 12K or 0** – The record length requested for the transfer to tape was either greater than 12K or 0.
- **Error 16, This Unit Already Acquired**—A duplicate unit number was specified for a drive and the drive has already been acquired.
- **Error 18, Invalid Time Entered...Must Be from 1 to 3599**—The user entered an erroneous value to the performance summary time interval prompt.
- **Error 20, Could Not Get Buffers for Transfers**—The buffers required for a tape transfer could not be acquired.
- **Error 21, Tape Rewind Commands Were Lost - Cannot Continue**—The drive was unloaded during ILEXER execution.

5.8.16.3 Disk Errors

The following list includes the error number, text, and cause of ILEXER disk errors.

- **Error 102, Drive Spindle Not Up to Speed—Spin Up Drive and Restart**—The disk drive is not spun up.
- **Error 103, Drive No Longer Exercised**—A disk drive reached the hard error limit or the drive went off line to the HSC during the exercise.
- **Error 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.
- **Error 105, No DACB Available—Notify Customer Service, Submit SPR**—This is reported if no DACBs can be acquired. If this happens, contact Customer Service as soon as possible and submit an SPR.
- **Error 106, Some Disk I/O Failed to Complete**—An I/O transfer did not complete during an allotted time period.
- **Error 107, Command Failed—Invalid Header Code**—ILEXER did not pass a valid header code to the diagnostic interface for the HSC.
- **Error 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.
- **Error 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.
- **Error 111, Write Requested on Write-Protected Drive**—The operator requested an initial Write operation on a drive that 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.
- **Error 112, Data Compare Error**—Bad data was detected during a Read operation.
- **Error 113, Pattern Number Error**—The first two bytes of each sector, which contain the pattern number, did not match.
- **Error 114, EDC Error**—Error Detection Code error: invalid data was detected during a Read operation.
- **Error 116, 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.
- **Error 117, Disk Unit Numbers Must Be Between 0 and 4094 decimal**—The operator specified a disk unit number out of the allowed range of values.
- **Error 118, Hard Failure on Disk**—A hard error occurred on the disk drive being exercised.

  NOTE
  The following disk errors identify the function attempted by ILEXER that caused an error to occur. Error logs do not indicate the operation attempted.

- **Error 119, Hard Failure on Compare Operation**—A hard failure occurred during a compare of data on the disk drive.
- **Error 120, Hard Failure on Write Operation**—A hard fault occurred during a Write operation on the specified disk drive.
- **Error 121, Hard failure on Read Operation**—A hard failure occurred during a Read operation on the disk drive being exercised.

- **Error 123, Hard Failure on INITIAL WRITE Operation**—A hard failure occurred during the first write to the disk drive.
- **Error 124, Drive No Longer On Line**—A drive that 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.

5.8.16.4 Tape Errors

The following list includes the error number, text, and cause of ILEXER tape errors.

- **Error 201, Couldn't Get Formatter Characteristics**—A communication problem with the drive is indicated. It could be caused by the unit not being on line.
- **Error 202, Couldn't Get Unit Characteristics**—The drive is not communicating with ILEXER. The unit could be off line.
- **Error 203, Some Tape I/O Failed To Complete**—The drive or formatter stopped functioning properly during a data transfer.
- **Error 204, Comm Error: TDUSUB Call Failed**—ILEXER cannot communicate with the drive through interface structures. They have been removed. Either the drive went available from on line, the drive is off line, or there is a fault.
- **Error 205, Read Data Error**—A Read operation failed during a data transfer, and none was transferred.
- **Error 206, Tape Mark Error...rewinding to restart**—ILEXER does not write tape marks. If this error occurs, it indicates a drive failure.
- **Error 207, Tape Position Lost...rewinding to restart**—An error occurred during a data transfer or a retry of one.
- **Error 209, Data Pattern Word Error Defect**—The first two bytes of a record containing the data pattern did not match.
- **Error 210, Data Read EDC Error**—Error Detection Code error; incorrect data was detected.
- **Error 211, Could Not Set Unit Char...removing from test**—The drive is off line and not communicating.
- **Error 213, Truncated Record Data Error...rewinding to restart**—More data was received than expected, indicating a drive problem.
- **Error 214, Drive Error...Hard Error**—A hard failure occurred with the drive being exercised.
- **Error 215, Unexpected Error Condition...removing drive from test**—This is caused by MSCP error conditions, which are not allowed (invalid commands, unused codes, write-protected drive write, and so forth).
- **Error 216, Unexpected BOT Encountered...will try to restart**—The drive is experiencing a positioning problem.
- **Error 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.
- **Error 218, Unrecoverable Read Error...rewinding to restart**—A hard error occurred during a Read operation and a data transfer did not take place.
- **Error 219, Controller Error...Hard Error..rewinding to restart**—A communications problem exists between the controller and the formatter.
- **Error 220, Formatter Error...Hard Error**—A communications problem exists between the formatter and the controller and/or drive.

- **Error 221, Retry Required on Tape Drive**—A failed Read/Write operation required a retry before succeeding.
- **Error 222, Hard Error Limit Exceeded...removing drive from test**—The drive exceeded the threshold of hard errors determined by a global user parameter (Section 5.8.7). The drive is then removed from the exercise.
- **Error 224, Drive Went Off Line...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.
- **Error 225, Drive Went Available...removing from test**—The drive became available to ILEXER and was not at the beginning of the exercise.
- **Error 226, Short Transfer Error...rewinding to restart**—Less data was received than transferred.
- **Error 227, Tape Position Discrepancy**—The tape position was lost, indicating a hard failure.

6
Off-line Diagnostics

6.1 Introduction

This chapter describes the off-line diagnostics, how to run them, errors that can occur, and summaries of the tests in each diagnostic. Included in the off-line diagnostics are:

- ODL—Off-line diagnostics loader
- OFLCXT—Off-line cache test (HSC only)
- OBIT—Off-line bus interaction test
- OKTS—Off-line K test selector
- OKPM—Off-line K/P memory test
- OMEM—Off-line memory test
- OFLRXE—Off-line RX33 exerciser (HSC only)
- ORFT—Off-line refresh test
- OOCP—Off-line operator control panel (OCP) test

The off-line diagnostics contain specific common characteristics, which are discussed in the following three sections. These characteristics are listed below:

- Identical software requirements
- Common load procedure
- Identical bootstrap initialization procedures
- Generic error message format

6.1.1 Software Requirements

All off-line diagnostics require boot media containing a bootable image of the diagnostics software programs. For an HSC, an RX33 off-line diagnostic diskette is required. For an HSC50, a TU58 off-line diagnostic tape cassette is required.

6.1.2 Off-line Diagnostics Load Procedure

For the HSC, the off-line diagnostics diskette boots from either RX33 drive and should not be write-enabled. This diskette contains the necessary software to run all the HSC off-line diagnostics. Booting is done either by powering on or by pressing 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 off-line P.ioj test.

The off-line diagnostics TU58 cassette boots from either TU58 drive and should not be write-enabled. This TU58 contains the necessary software to run all the HSC50 off-line diagnostics. Booting is accomplished either by powering on or by pressing and releasing the Init switch with the Secure/Enable switch in the enable position. This causes the P.ioc ROM bootstrap tests to run followed by the off-line P.ioc test.

NOTE
For off-line diagnostics, the HSC 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 RX33 drive containing a diskette with a bootable image for the HSC
- TU58 controller and at least one TU58 drive containing a cassette with a bootable image for the HSC50

Before control is turned over to the HSC bootstrap ROMs, internal microcode tests execute in the J11 (HSC) or F11 (HSC50) chip sets. Refer to Table 2–1 for definitions of the J11/F11 module (P.ioj/c) LEDs.

6.2 ROM Bootstrap

The P.ioj/c ROM bootstrap verifies the basic integrity of the P.ioj/c module, part of the Program memory, and the boot device. The goal of the bootstrap tests is to test enough of the HSC to allow further test loading from the boot device.

The bootstrap test is the first step in the HSC initialization process. It is run for every bootstrap or reload of the HSC operating system (CRONIC). The bootstrap is initiated automatically each time the HSC is powered on and also is initiated by CRONIC when a software reboot is required.

The bootstrap is a PDP-11 program written to execute in a DCJ11/DCF11 CPU in a standalone environment. This means no other software processes coexist with the bootstrap.

Bootstrap failures are reported through the fault lamp mechanism, which specifies the module most likely causing the problem. See Figure 4–3 for the fault code definitions. An error table is maintained in Program memory addresses 00000400 through 00000412. These addresses contain the reasons for each load device boot failure.

6.2.1 Initialization Instructions

The following procedure lists the operating instructions for the P.ioj/c ROM bootstrap. Refer to Section 6.2.4 if this procedure fails.

1. HSC—Insert the off-line diagnostics diskette with a bootable image into the RX33 unit 0 drive (left-hand drive).
2. HSC50—Insert the off-line diagnostics tape with a bootable image into either of the TU58 drives.
3. Turn power ON.

4. Set the Secure/Enable switch to the enable position, then press the Init switch. The bootstrap initiates automatically.

At this point, the P.ioj/c module executes internal microdiagnostics and then begins to execute from the boot ROM. The Init lamp lights on the HSC operator control panel (OCP) when the bootstrap PDP-11 tests are done. The load device drive-in-use LED lights 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.

6.2.2 Failures

Most bootstrap failures result in lighting the fault lamp on the HSC OCP. When this happens, press the Fault switch momentarily, and read the failure code displayed in the OCP lamps. Section 6.2.5 indicates the HSC modules most likely causing the bootstrap failure. Momentarily pressing the Init switch on the OCP reinitiates the bootstrap.

The microdiagnostic LEDs on the J11/F11 module indicate if a hard fault exists causing the J11/F11 to hang before control is passed the boot ROM. Section 6.2.5 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/c, the exact instruction that failed is determined by pressing the terminal break key and noting the address displayed on the terminal. With a bootstrap listing, this address indicates the instruction that failed.

NOTE
The bootstrap does not accept user-modifiable flags.

6.2.3 Progress Reports

The bootstrap does not issue progress reports in the usual sense; however, certain indications of bootstrap progress are evident. These indications are given in the following list:

- **Lamps clear**—Clears all of the HSC OCP lamps. If the lamps fail to clear immediately after the bootstrap is initiated, a failure of the P.ioj/c is probable. (Circuitry on the P.ioj/c 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/c PDP-11 processor.
- **RX33 drive-in-use**—Lights as the bootstrap tries to load the Init P.io test (or off-line 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/c test (or off-line P.ioj/c 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.

6.2.4 Error Information

Specific error codes for the P.ioj/c bootstrap (Codes 21, 22, and 23) are described in detail in Chapter 4.

Because the bootstrap operates in a standalone environment, it does not use the terminal as an error-reporting mechanism. Instead, the HSC OCP 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 OCP. When the Fault switch is pressed, the bootstrap displays a failure code in the OCP lamps. The failure code blinks on and off at one-half second intervals.

6.2.5 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 pressed. The program is halted by pressing the break key on the console terminal. If 17772340 is typed, ODT responds by displaying the contents of address 17772340, the test number. Use the test number to refer to the appropriate test in Section 6.2.6.
- **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 pressed, the HSC did not initiate its boot sequence. If the Secure/Enable switch is in the correct position, the J11/F11 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/F11 module. At powerup, set all the LEDs and then turn them off as the J11/F11 proceeds through its microdiagnostic sequence.

6.2.6 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:

  a. Single operand instructions (both word and byte mode):

     ADC, CLR, COM, INC, DEC, NEG, TST, ROR, ROL, ASR, ASL, SWAB, NOP

  b. Double operand instructions (both word and byte modes):

     MOV, CMP, BIT, BIC, BIS, ADD, SUB

  c. Branch instructions:

     BR, BNE, BEQ, BPL, BMI, BCC (BHIS), BCS (BLO), BGE, BLT, BGT, BLE, BHI, BLOS, BVC, BVS

  d. Jump and miscellaneous instructions:

     JMP, JSR, RTS, SOB, MTPS, MFPS, CCC, CLN, CLV, CLZ, SEN, SEV, SEZ

  e. Addressing modes:

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 6.2.2.

- **Test 1, Program Memory (Swap Bank)**—The memory (Swap Bank) HSC 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/c 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/c module must be replaced.
- **Test 2, Program Memory (Vector Area)**—In order for the HSC 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 through the fault lamp mechanism (Section 6.2.2). A failure in this test indicates the M.std2/M.std module must be replaced.
- **Test 3, Program Memory (8-Kword Partition)**—After verifying the first part of Program memory is working, the bootstrap tries to find an 8-Kword piece of Program memory between address 00004000 and address 00160000. This partition is used to load the Init P.ioj/c test from the load device. If insufficient memory is available, a Program memory error is reported through the fault lamp mechanism.

  A failure in this test indicates the M.std2/M.std 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 0. 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/c test (or Off-line P.ioj/c 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/c test or the Off-line P.ioj/c 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. Table 6–1 shows the RX33 error table addresses and meanings.

Table 6–1 RX33 Error Table

Address	Meaning
00000400	Contains controller error code (code 1 or code 2)

Table 6–1 (Cont.) RX33 Error Table

Address	Meaning
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 the following sequence to examine the RX33 error code table.

1. Press the break key on the console terminal.

   The terminal displays the address of the current instruction of the bootstrap, then prompts for input with an @ character.

2. Type the appropriate address, nnn.

   The terminal displays the (octal) contents of that address.

3. Press LINEFEED to examine the Table 6–2 controller error and related failure information.

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, and so forth). 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/c or M.std2/M.std failures. When the loaded image is started, the State lamp is lit, and the Init lamp is turned off.

6.2.7 Generic Error Message Format

All of the diagnostics use a common method of reporting errors and a common error message format. All errors are reported on the console terminal as they occur. In all off-line 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 also is 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.
    En is an error number with a range of 1 through 77 (octal).
    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.

6.3 ODL—Off-line Diagnostics Loader

The off-line diagnostics loader provides a software environment for the HSC off-line diagnostics. The loader supports a command language that loads and executes an off-line diagnostic from the load device into Program memory. The loader command language also permits the display and modification of any address contents in the HSC Program, Data, or Control memories.

The software environment provided for off-line diagnostics includes a load device driver and a terminal driver. A standard software interface between the diagnostics and the load device 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.

6.3.1 Loader System Requirements

Hardware required to run the off-line diagnostics loader includes:

- I/O Control Processor module with HSC Boot ROM
- At least one memory module
- RX33 controller with at least one working drive

  or
- TU58 with at least one working drive
- Terminal connected to I/O Control Processor console interface

6.3.2 Loader Prerequisites

In the process of loading the off-line diagnostics loader, several diagnostics are run. The ROM bootstrap tests the basic PDP-11 instruction set, tests a partition in Program memory, and tests the load device used for the boot. Then the bootstrap loads the off-line P.ioj/c test, which completes the PDP-11 tests and the remainder of the I/O Control Processor module tests.

After these tests, the off-line diagnostics loader is loaded from the load device 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.

6.3.3 Loader Operating Instructions

Follow these steps to start the off-line loader:

1. Insert the HSC off-line diagnostics media into the load device.
2. Power on the HSC, or press and release the Init button on the HSC OCP.
3. The load device drive-in-use LED lights within a few seconds, indicating the bootstrap is loading the off-line diagnostic loader to Program memory.
4. In less than 30 seconds, the off-line diagnostics loader indicates it has loaded properly by displaying the following:

```
HSC OFL Diagnostic Loader, Version Vnnn
Radix=Octal,Data Length=Word,Reloc=00000000
ODL>
```

5. The off-line loader is now ready to accept commands. Section 6.3.4 contains information on the loader command language.

6.3.4 Loader Commands

The following sections describe the commands recognized by the off-line loader.

6.3.4.1 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 load device and displays the contents of this file on the HSC console terminal. Section 6.3.5.2 contains a listing of the loader help file.

6.3.4.2 SIZE Command

The off-line system sizer is invoked by the SIZE command. The sizer determines the sizes of the HSC Program, Control, and Data memories, and the type of requestor in each HSC requestor position. The term 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.

6.3.4.3 TEST Command

The off-line diagnostics loader TEST command is used to invoke the various off-line diagnostics available on the HSC. 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 off-line memory test.

- **Off-line Cache Test**—Verifies the full functionality of the onboard cache. The off-line cache test is invoked by the TEST CACHE command.
- **Bus Interaction Test**—Invoked by the TEST BUS command. The bus interaction test generates contention on the HSC 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**—Invoked by the TEST K command. The K test selector allows specific requestor microdiagnostics to run.
- **K/P Memory Test**—Invoked by the TEST MEMORY BY K command. The K/P memory test uses one of the HSC requestors to test either Data or Control memory. This test runs faster than the off-line 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.
- **Off-line Memory Test**—Invoked by the TEST MEMORY command. This test uses the I/O Control Processor to test Program, Control, or Data memories.
- **Off-line RX33 Exerciser**—A combined hardware diagnostic and exerciser for the M.std2/RX33 subsystem of the HSC. Invoke the off-line RX33 exerciser by the TEST RX command.
- **Memory Refresh Test**—Invoked by the TEST REFRESH command. The memory refresh test allows the refresh feature of the memories to be tested.
- **OCP Test**—Invoked by the TEST OCP command. The OCP test checks the HSC lights and switches. The test requires manual intervention by an operator.

6.3.4.4 LOAD Command

The LOAD command loads a program into HSC Program memory without starting it. The command format is LOAD <filename>, where <filename> is the name of any file on the HSC OFFLINE diskette. The loader finds the specified file and loads it into Program memory. This command is for patching a program image in before starting execution. After the patch is made, the program can be initiated through the START command.

6.3.4.5 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, or it may be used to reinitiate the last loaded off-line diagnostic. This saves the time required to reload the program from the load device. For example, you have previously typed SIZE to initiate the off-line system sizer program and after the sizer completes, you wish to run it again. Typing START restarts the sizer without reloading the program from the load device, saving many seconds of load time.

6.3.4.6 EXAMINE and DEPOSIT Commands

The EXAMINE and DEPOSIT commands are used to display or modify the contents of any location in the HSC 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 also can 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 6.3.4.11).

EXAMINE Command:

The EXAMINE command is used to display the contents of any location in the HSC 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 also are permitted (Section 6.3.4.7).

In the following 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.

```
ODL> E 14017776
  (D) 14017776 125252
```

DEPOSIT Command:

The DEPOSIT command is used to modify the contents of any location in the HSC 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 also are permitted (Section 6.3.4.7).

In the next 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).

```
ODL> D 14017776 123456
```

6.3.4.7 EXAMINE and DEPOSIT 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 the contents of address 16012344 were just examined and the value 1234 is to be deposited into the same address, type DEPOSIT * 1234 instead of typing DEPOSIT 16012344 1234.
- **Plus sign (+)**—This sign also is 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 plus sign (+) 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 6.3.4.11).

  If the current default data length is a byte, the loader adds one to the last address. If the default is 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 want to examine a table of words beginning at address 14125234, examine the first location by typing EXAMINE 14125234. The next location can now be examined by typing EXAMINE + instead of typing EXAMINE 14125236.

- **Minus sign (–)**—This sign also is 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 minus sign (–) 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 6.3.4.11).

  If the current default data length is a byte, the loader subtracts one from the last address. If the default is 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 a table of words that ends at address 14012346 is to be examined, the operator 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 typing EXAMINE 14012344.

- **At (@)**—The @ symbol also is used as a symbolic address. This symbol means the loader uses the data from the last EXAMINE or DEPOSIT command as an address. This feature is useful when following linked lists. For example, first examine location 123434 which contains a pointer to a linked list. Now type EXAMINE @ to examine the location pointed to by the first location.

6.3.4.8 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 EXAMINE or DEPOSIT. To stop a repeated command, type CTRL/C.

In the following example of repeating a DEPOSIT command, 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:

```
REPEAT DEPOSIT 14017776 125252
```

or

```
RE D 14017776 125252
```

In the repeating an EXAMINE command example, the contents of address 14017776 can be continuously examined. The format of the EXAMINE command does not change. The EXAMINE command is just preceded by the word REPEAT.

```
REPEAT EXAMINE 14017776
```

or

```
RE E 14017776
```

In the examples shown, the contents of location 14017776 are displayed continuously on the terminal. This slows down the repetition of the command and wastes paper on hardcopy 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 6.3.4.10.

6.3.4.9 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 0 when the loader is initiated, so it normally has no effect on the addresses typed in an EXAMINE or DEPOSIT command.

Use the following example to examine many locations in 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 an EXAMINE command with an address of 0 is issued, 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 many locations in 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. You can stop the display with a CTRL/C.

6.3.4.10 EXAMINE and DEPOSIT Qualifiers (Switches)

The following list describes the EXAMINE and DEPOSIT qualifiers.

- **/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 also displays 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 is stored in the location specified and the next number of locations following.

- **/BYTE/WORD/LONG/QUAD**—These qualifier switches are used to control the data length of examined or deposited data. Normally, the loader uses the default data length (Section 6.3.4.11) 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 a byte quantity at address 16001234 is to be examined. Typing EXAMINE 16001234/BYTE would display the proper byte without affecting the default data length.

- **/OCTAL/DECIMAL/HEX**—These qualifier switches 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 6.3.4.11), but the /OCTAL/DECIMAL/HEX qualifiers are used to override the default radix for a single EXAMINE command. For example, assume the default radix is octal. Typing EXAMINE 14001234/HEX 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)—This qualifier switch inhibits the display of examined data 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.

6.3.4.11 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 off-line loader first starts, it sets the default radix to octal. Type SET DEFAULT HEX 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 SET DEFAULT LONG 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.

6.3.4.12 Executing INDIRECT Command Files

The loader is capable of executing indirect command files stored on the load device. These command files consist of valid off-line 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 Off-line Loader Help file (Section 6.3.5.2) is an indirect command file that contains only comments.

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 off-line diagnostics diskette. Any editor that does not insert line numbers in the output files can be used to create command files.

6.3.5 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 is the address of the trap or interrupt vector.
xxx is the virtual PC of the loader at the time of trap.
yyy are the contents of PSW at the time of trap.
zzz is the address of the 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 issued only 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 6.3.5.1 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. The contents of the VPC includes the address of the instruction following the instruction executing when the trap occurred. Notify Customer Service to analyze such failures.

NXM traps can be caused by EXAMINE or DEPOSIT commands if an address not contained in a particular HSC is specified. For example, if an HSC contains only Data memory from addresses 14000000 through 14177776, and an EXAMINE or DEPOSIT is tried for 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 HSC memories are powered on, the parity bits are in random states. Thus, if a user examines a location not written since power-on, 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 power-on, a memory error is indicated.

NOTE
The I/O Control Processor and Ks have bits allowing them to write bad parity for testing the parity circuit. These bits are for diagnostics engineering purposes only.

6.3.5.1 Trap and Interrupt Vectors

Table 6–3 is a list of trap and interrupt vectors for various devices and error conditions recognized by the I/O Control Processor PDP-11 processor.

Table 6–3 Trap and Interrupt Vectors

Vector	Device or Error Condition
000004	Nonexistent 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

6.3.5.2 Help File

The help file display is started by entering HELP at the ODL prompt. This file is unique to each version of software, as shown by Vnnn in Example 6–1. This display is the complete help facility for the off-line diagnostics. Exit the help file display by typing a CTRL/C.

```
!
! HSC OFL Diagnostic Loader Help File - Vnnn
! 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
! 'RUN <filename>'  ; Do implicit LOAD and START of <filename>
! 'SEt Default <option>,<option>;set default radix or data length
!    <options> = Byte,Word,Long,Quad,Hex,Octal,Decimal
! 'SEt Relocation:#'  ;set relocation register to #
!    ;   relocation register is 22-bit positive #
!    ;   added to address of all 'Ex' and 'De'
!    ;   commands.
! 'SHow'   ;display defaults and Loader version
! 'SIze'   ;Size memories and display K status
! 'Start'   ;start program in diagnostic partition
! 'Test Bus'   ;load and start the OFL Bus Interaction test
! 'Test K'   ;load and start the OFL K Test Selector
! 'Test MEmory'   ;load and start the OFL Memory Test
! 'Test MEmory By K'  ;load and start the OFL K/P Memory Test
! 'Test OCP'   ;load and start the OFL OCP Test
! 'Test Refresh'  ;load and start the OFL Refresh Test
! 'Test Cache'   ;load and start the OFL cache test (L0111-xx)
! 'Test Rx'   ;load and start the OFL Rx33 test (L0111-xx)
! 'WCS'    ;load and start the OFL control store loader
!
!Qualifiers (switches) for 'Ex' and 'De':
! '/Next:#'   ;repeat Ex or De on next '#' addresses
! '/Byte,/Word,/Long,/Quad' ;use specified data length instead of default
! '/Octal,/Decimal,/Hex' ;use specified radix for Ex display
! '/INH'   ;inhibit display of examined data
! <end of help file>
```

Example 6–1 Example HELP file display

6.4 OFLCXT—Off-line Cache Test

The off-line cache test (OFLCXT) is a diagnostic that runs under the off-line loader in a standalone environment. It provides in-depth testing of the cache logic on the J11 P.ioj and 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.

6.4.1 System Requirements

OFLCXT is loaded into memory through the off-line 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 off-line loader.

6.4.2 Operating Instructions

This section contains operating instructions specific to OFLCXT. If the HSC is not booted and running the off-line loader, necessary instructions are found in Section 6.1.2 and in Section 6.2. To run the off-line loader, enter the TEST CACHE command at the ODL> prompt.

This command loads OFLCXT from the media and transfers control to the diagnostic. When it starts, OFLCXT displays the following:

```
HSC OFFLINE Cache Test      Vxxx
```

```
Where: Vxxx is a three-digit version/edit number.
```

6.4.3 Test Termination

OFLCXT can be terminated by typing CTRL/Y.

6.4.4 Parameter Entry

Following are the three user-modifiable parameters for OFLCXT. In each case, the default, invoked by a "," (comma), is shown in brackets. If no default is possible, the brackets are empty.

- **Select Data Reliability test**—This 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 off-line cache test prints:

  ```
  Run extended cache ram test (Y/N)   [N] ?
  ```

  Selection of this optional test increases test time per pass to about 4 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. OFLCXT prints:

  ```
  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) [D] ?
  ```

  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] ?
  ```

To repeat the last test specified using the parameters, answer this prompt with Y or RETURN. To cause the test to prompt for new parameters, answer the prompt with N.

Use the DELete key to delete mistyped parameters before terminating the entry with RETURN. If an error in a parameter was terminated with RETURN, type CTRL/C to return to the initial prompt and re-enter all parameters.

6.4.5 Progress Reports

OFLCXT provides summary information at the end of each pass. The end of pass message is similar to the following:

```
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.

For any OFLCXT prompts, use the DELete key to delete mistyped parameters before terminating the entry with RETURN. If an error in a parameter was terminated with RETURN, type CTRL/C to return to the initial prompt and re-enter all parameters.

6.4.6 Test Summaries

Following are summaries of the OFLCXT 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 (17777746), the hit/miss register (17777752), and the memory system error register (17777744). To perform further diagnosis, these registers must respond.
- **Test 2, Cache control register bits**—Tests the read/write bits of the cache control register (17777746) for stuck-at faults. In addition, bits (8,11:15), which are write-only, are checked for read data of 0. 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 is 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 goes directly to main memory and bypasses the cache. The cache is enabled, and a read access to the test location results in a hit condition in the hit/miss register. Then the test location offset by 8-Kwords is accessed. This results 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 1's and 0's 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 byte-write to the test location not allocated followed by a byte-read of the test location. The read results 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 HSC.

  PDR bypass is tested by remapping all PDRs to point to control memory. Control memory is written by 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) is 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 records a hit and then invalidates the location. If the location is written or read again, it results 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 that is 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 is a hit, and the second is 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), the tag parity error bit is set (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 trap 114, or abort through trap 114 on parity errors. An interrupt through trap 114 sets the correct error bit(s) in the memory system error register. Also, the instruction detecting the parity error completes.

  On an abort through trap 114, the correct error bit(s) is set, but the instruction does 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 results in a cache miss when it accesses after the DMA write.
- **Test 13, Check blockage of parity error on NXM abort**—Generates simultaneous NXM and parity errors. The NXM trap occurs, 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 are cache hits, and only these locations 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 4 minutes are required to complete one pass of this test.

6.4.7 Error Information

OFLCXT displays the errors detected during execution on the console terminal. Error messages follow the diagnostics generic error message format preceded by an OFLCXT> prompt.

A typical OFLCXT error message format follows:

```
OFLCXT>hh:mm T aaa E bbb U-000
 < Text describing error >
 MA-xxxxxxxx
 EXP-yyyyyy
 ACT-zzzzzz

where:

  hh is the elapsed hours since last bootstrap.
  mm is the elapsed minutes.
  aaa is the decimal number denoting test.
  bbb is the decimal number denoting the error detected.
  MA-xxxxxxxx is the address of location causing the error.
  EXP-yyyyyy is the data pattern that was expected.
  ACT-zzzzzz is the data pattern that was actually found.
```

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.

6.4.8 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 also is 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 that 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. 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 contents 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 17777757776. 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 load device subsystem did not respond correctly to the DMA test operation. There may be faults in the load device controller or the interrupt service logic. This message is information only.
- **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.

6.4.9 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 is 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, also may be caused by the tag compare logic, which is separate from the gate arrays and the data path.

6.5 OBIT—Off-line Bus Interaction Test

The off-line bus interaction test (OBIT) creates Control and Data bus contention among the requestors in the HSC 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, I/O Control Processor interaction can be selected with the Program, Control, and Data memories, with the OCP, and/or with 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 nine 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 OBIT fails, it must first be determined if the failure was caused by an interaction problem. This is determined by running the off-line 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 OBIT. Also specify the same starting and ending addresses displayed with the error report from the bus interaction test. If the requestor also fails the off-line K/P memory test, the original problem was not an interaction problem. The problem is localized in the same manner as any ordinary memory failure.

6.5.1 System Requirements

Hardware required to run OBIT is shown in the following list:

- I/O Control Processor module with HSC boot ROMs
- Memory module
- Working Control and Data memories
- Load device 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)

6.5.2 Off-line Bus Interaction Test Prerequisites

Booting procedures and testing through successful loading of the off-line diagnostics loader program are described in Section 6.1.2 and in Section 6.2.

Due to the sequence of tests that precede the memory test, OBIT assumes the I/O Control Processor module and the load device are tested and working. OBIT also assumes the Control and Data memories were previously tested with the off-line memory test or the off-line K/P memory test, and are working.

6.5.3 Operating Instructions

At the loader prompt ODL>, the operator types the TEST BUS command and OBIT 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.

6.5.4 Test Termination

OBIT can be terminated by typing CTRL/C. OBIT may continue running for a few seconds after it is terminated.

6.5.5 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 OBIT prompts, use the DELete key to delete mistyped parameters. If an error in a parameter already terminated with is noted, type a CTRL/C to return to the off-line loader. Then type START to restart the test from the beginning.

The test prompts for selection of the requestors used for the test, as follows:

```
Use requestor #001, K.ci (Y/N) [Y] ?
```

Answer with a Y if the K.ci should be used. Answer with N 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 the requestors that should be used have been indicated:

```
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 RETURN (or Y) if I/O Control Processor interaction with memory is wanted. Answer with N if I/O Control Processor interaction with memory is not wanted. If the prompt is answered with an N, the following three prompts are skipped. If the prompt is answered with a 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 RETURN if I/O Control Processor is to interact with the specified memory while the requestors are generating contention on the Control and Data buses. Answer with N if the I/O Control Processor is not 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 I/O Control Processor interaction with the OCP is wanted, answer with RETURN. If OCP interaction is not wanted, answer with N. The test then prompts for load device interaction.

```
Interact with load device (Y/N) [Y] ?
```

If I/O Control Processor interaction with the load device is wanted, answer with RETURN. If such interaction is not wanted, answer with N. 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 is repeated. Entering 0 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 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 Y or RETURN. To cause the test to prompt for new parameters, answer the prompt with N.

Use the DELete key to delete mistyped parameters before terminating the entry with RETURN. If an error in a parameter was terminated with RETURN, type CTRL/C to return to the initial prompt and re-enter all parameters.

6.5.6 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 is the decimal count of the number of passes completed.
  xxxxxx is the decimal count of the number of errors detected
         on the current pass.
  yyyyyy is the decimal count of the total number of errors detected
         since the test was initiated.
```

6.5.7 Test Summary

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 0, rewrite the same location with a 1 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 1 bit and reread to check that the write worked properly.
4. Repeat step 3 until the test pattern consists of a word containing all 1's (pattern 177777).
5. Repeat steps 1 through 4, but this time start at the highest memory address and work down to the lowest each time. However, instead of adding an additional 1, add an additional 0. This changes each memory location from all 1's back to all 0's.

6. End of test. All memory is cleared to 000000.

6.5.8 Error Information

Error messages produced by this test conform to the HSC generic diagnostic error message format. Off-line bus interaction test error messages are preceded by an OBIT> prompt.

Following is a typical OBIT 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:

  hr is the hours since Off-line Loader was last booted.
  mm is the minutes since Off-line Loader was last booted.
  aaa is the decimal number denoting test.
  bbb is the decimal number denoting the error detected.
  MA-xxxxxxxx is the address of location causing the error.
  EXP-yyyyyy is the data pattern that was expected.
  ACT-zzzzzz  is the data pattern that was actually found.
  < K-error-Summary-Info >
  Memory Test Configuration:
```

Refer to Section 6.5.9 for information on Requestor Error Summary, and to Section 6.5.10 for information on Memory Test Configuration.

6.5.9 Requestor Error Summary

When the requestor reports a memory test failure to the I/O Control Processor, the following information is supplied:

a. Address of the failing memory location

b. Data EXPected and data ACTually found

c. Error summary information

The error summary information is supplied as a 3-bit field, including:

a. A bit indicating a parity error occurred while reading the location

b. A bit indicating an NXM error occurred while accessing the location

c. 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/c test), a Control bus error is probably caused by a problem with the requestor's drivers that assert the CCYCLE lines.

6.5.10 Memory Test Configuration

The memory test configuration lists each requestor being used for OBIT 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, or K.si) 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

6.5.11 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

Where:

  MA is the 22-bit address of the failing location.
  EXP is the data pattern EXPected by the requestor.
  ACT is the data pattern found by the requestor.
  Memory Test Configuration are the other requestors that
  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**—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, and one of these requestors may be responsible for the time-out.

When the I/O Control Processor enables the requestor to perform the memory test, the requestor begins its initialization sequence, including execution of 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 the K.ci) after 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 1 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 is the address causing the parity trap.
  - VPC is 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 Nonexistent 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 the 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 is to 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 installed in error.

- **Error 005, Memory Test Error (P.ioj/c 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.

- **Error 006, TU58 Synchronization Failure** – This is an HSC50-only error. The TU58 drive was unable to properly establish synchronization.

- **Error 007, General TU58 Error—** This is an HSC50-only error. Text is provided with the error message to indicate details about the TU58 failure.
- **Error 008, TU58 Checksum Error—** This is an HSC50-only error. The data checksum read from the TU58 did not match the one generated during the read operation.
- **Error 009, TU58 End Packet Error—** This is an HSC50-only error.
- **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 HSC when load device interaction is enabled.

- **Error 011, RX33 Drive Not Ready**—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—** The RX33 detected 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**—A Recalibrate (seek to track 0) operation is performed before each block of Read operations. The RX33 did not show correct status after the RECAL command.
- **Error 014, RX33 Seek Timeout—** The RX33 did not respond by interrupting during a seek.
- **Error 015, RX33 Seek Error**—Sets the seek error bit (bit 4 of the CSR). At the end of a Seek operation, the RX33 was 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 a 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 off-line interaction bus test performs reads in blocks of four.

6.6 OKTS –Off-line K Test Selector

The off-line K test selector (OKTS) allows a K to perform an internal microdiagnostic self-test on command. OKTS executes from the P.ioj/c and uses the HSC K Control Area for instruction. Select the K for testing and the test number of the microdiagnostic test for execution.

6.6.1 System Requirements

The following hardware is required to run OKTS:

- P.ioj/c module with HSC Boot ROMs
- M.std2/M.std memory module
- A working section of Control memory for use as a K Control Area
- One working load device drive
- Terminal connected to the P.ioj/c console interface

Due to the sequence of tests that precede this test, assume the P.ioj/c, Program memory, and load device are working.

6.6.2 Operating Instructions

If the HSC is not booted and loaded, refer to Section 6.1.2 and Section 6.2. If the loader prompt ODL> is displayed, follow these steps to start the K test selector:

1. Type TEST K. The load device drive-in-use LED lights as the test is loaded.
2. Test K indicates it has been loaded properly by displaying the following:

```
HSC OFL K Test Selector
```

3. The test next prompts for parameters.

6.6.3 Test Termination

OKTS can be terminated by typing CTRL/C.

6.6.4 Parameter Entry

This section gives detailed information on how to enter the test parameters for the OKTS. Items in square brackets are the default value for each particular prompt. If no default is possible, the brackets are empty.

OKTS first prompts:

```
Requestor # of K (1 through 9) [] ?
```

Answer this question with a single digit (1 through 9) that specifies the requestor number of the K to be used. Terminate the response by typing 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 through 20) (O) [] ?
```

Legal test numbers are octal numbers between 1 and 20, 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 entry with RETURN.

Refer to the following 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 — ALU test
   Test 2 — Data bus test
   Test 3 — Control bus test
   Test 4 — PROM parity test
   Test 5 — Memory test (unavailable through K test selector)
   Test 6 — RAM test
   Test 7 — PLI 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 — ALU test
   Test 2 — Data bus test

Test 3 — Control bus test
Test 4 — PROM parity test
Test 5 — Memory test (not available through K test selector)
Test 6 — RAM test
Test 7 — SERDES/RSGEN test
Test 10 — Partial SDI interface test
Test 11 — Control memory access test
Test 12 — Lock 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 — ALU test
   Test 2 — Data bus test
   Test 3 — Control bus test
   Test 4 — PROM parity test
   Test 5 — Memory test (not available through K test selector)
   Test 6 — RAM test
   Test 7 — SERDES test
   Test 10 — Partial STI interface test
   Test 11 — Control memory access test
   Test 12 — Lock test

4. K.si microdiagnostics—The following list shows the test number and name of each of the K.si microdiagnostics:

   Test 0 — 2911 test
   Test 1 — ALU test
   Test 2 — ROM parity/traps test
   Test 3 — Scratchpad test
   Test 4 — Data bus test
   Test 5 — MOVI for system test
   Test 6 — Control bus test
   Test 7 — RTS gate array test
   Test 10 — SIECL test
   Test 11 — Frame test (U ⇒ L)
   Test 12 — Frame test (L ⇒ U)
   Test 13 — Sector test (U ⇒ L)
   Test 14 — Sector test (L ⇒ U)
   Test 15 — WCS load/verify test
   Test 16 — WCS MOVI test
   Test 17 — WCS EDC check
   Test 20 — INIT packet search test

The test then prompts:

```
# of passes to perform (D) [1] ?
```

Enter a decimal number between 1 and 2147483647 to specify the number of times the memory test should be repeated. (Entering 0, or just RETURN, results in the performance of one pass.)

The P.ioj/c 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/c displays an end-of-pass message. If the K fails to complete within 4.2 seconds, the P.ioj/c 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] ?
```

To repeat the last test specified using the parameters, answer this prompt with Y or RETURN. To cause the test to prompt for new parameters, answer the prompt with N.

Use the DELete key to delete mistyped parameters before terminating the entry with RETURN. If an error in a parameter was terminated with RETURN, type CTRL/C to return to the initial prompt and re-enter all parameters.

6.6.5 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/c times out.

The end-of-pass message is displayed as follows:

```
End of Pass nnnnnn, xxxxxx Errors, yyyyyy Total Errors

where:

  nnnnnn is the number of passes.
  xxxxxx is the number of errors detected during the current pass.
  yyyyyy is the number of errors detected for all passes.
```

6.6.6 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 off-line 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 is 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.

6.6.7 Test Summaries

The following is a list of OKTS test summaries.

- **Test 000, Moving Inversions Test**—The moving inversions (MOVI) memory test is used by the P.ioj/c 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/c. 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 0, rewrite the same location with a single 1 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 1 bit. Reread it to verify the Write operation worked properly.
4. Repeat step 3 until the test pattern consists of a word containing all 1's (pattern 177777).
5. Repeat step 3, but this time substitute a single extra 0 each time, instead of a 1.
6. Continue step 5 until the test pattern consists of a word of all 0's (pattern 000000).
7. Repeat steps 1 through 6, but this time start at the highest memory address and work down to the lowest each time. This will work each memory location from all 0's to all 1's, and back to all 0's.
8. End of test. All memory is cleared to 000000.

- **Test 001 through test 020, K Microdiagnostics**—Refer to the following four 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 — ALU test
   Test 2 — Data Bus test
   Test 3 — Control Bus test
   Test 4 — PROM Parity test
   Test 5 — Memory test (unavailable through K test selector)
   Test 6 — RAM test
   Test 7 — PLI 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 — ALU test
   Test 2 — Data Bus test
   Test 3 — Control Bus test
   Test 4 — PROM Parity test
   Test 5 — Memory test (not available through K test selector)
   Test 6 — RAM test
   Test 7 — SERDES/RSGEN test
   Test 10 — Partial SDI Interface test
   Test 11 — Control Memory Access test
   Test 12 — Lock 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 — ALU test
   Test 2 — Data Bus test
   Test 3 — Control Bus test
   Test 4 — PROM Parity test
   Test 5 — Memory test (not available through K test selector)
   Test 6 — RAM test
   Test 7 — SERDES test
   Test 10 — Partial STI Interface test
   Test 11 — Control Memory Access test
   Test 12 — Lock test

4. K.si microdiagnostics—The following list shows the test number and name of each of the K.si microdiagnostics:

   Test 0 — 2911 test
   Test 1 — ALU test
   Test 2 — ROM Parity/Traps test
   Test 3 — Scratchpad test
   Test 4 — Data Bus test
   Test 5 — MOVI for System test
   Test 6 — Control bus test
   Test 7 — RTS Gate Array test
   Test 10 — SIECL test
   Test 11 — Frame test (U ⇒ L)
   Test 12 — Frame test (L ⇒ U)
   Test 13 — Sector test (U ⇒ L)
   Test 14 — Sector test (L ⇒ U)
   Test 15 — WCS Load/Verify test
   Test 16 — WCS MOVI test
   Test 17 — WCS EDC check
   Test 20 — INIT Packet Search test

6.6.8 Error Information

Error messages produced by this test conform to the HSC generic diagnostic error message format. Off-line K selector test error messages are preceded by an OKTS> prompt.

A typical OKTS error message format follows:

```
OKTS>hh:mm T aaa E bbb U-000
 < Text describing error >
 MA-xxxxxxxx
 EXP-yyyyyy
 ACT-zzzzzz
 < K-Error-Summary-Info >
```

```
where:

  hh is the elapsed hours since last bootstrap.
  mm is the elapsed minutes.
  aaa is the decimal number denoting test.
  bbb is the decimal number denoting the error detected.
  MA-xxxxxxxx is the address of location causing the error.
  EXP-yyyyyy is the data pattern that was expected.
  ACT-zzzzzz is the data pattern that was actually found.
```

6.6.9 Error Messages

Errors detected by OKTS fall into one of three classes:

1. Control memory errors occurring when the P.ioj/c is testing the portion of Control memory used to communicate with the K. (The P.ioj/c does not test Data memory.) Error numbers 000 through 007 are all Control memory errors detected by the P.ioj/c. 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/c (NXM and Parity). Errors 010 and 011 are unexpected trap errors detected by the P.ioj/c. Error 010 signifies a parity trap occurred, and error 011 indicates a nonexistent 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/c is testing the K Control Area at a memory location that did not contain the expected pattern, where:
  - MA is the address of the failing location.
  - EXP is the data pattern EXPected.
  - ACT is 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 1.
  2. The additional 1 bit occurs immediately to the left of the left-most 1 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 1's 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 left-most 1 in the EXPected data:

        ```
        EXP=000777, ACT=002777
        ```

- **Error 001**—Occurs in the moving inversions test when the P.ioj/c 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 is the address of the failing location.
  - EXP is the data pattern EXPected.
  - ACT is 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 probably is 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/c is testing the K Control Area. A memory location did not contain the expected pattern, where:
  - MA is the address of the failing location.
  - EXP is the data pattern EXPected.
  - ACT is 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 0 than the EXPected data.
  2. The additional 0 occurs in the same bit position as the left-most bit 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 1's than the EXPected data:

```
EXP=037777, ACT=077777
```

  3. The bit in error is not in the same bit position as the left-most bit in the EXPected data:

```
EXP=001777, ACT=001377
```

- **Error 003**—Occurs in the moving inversions test when the P.ioj/c 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 is the address of the failing location.
  - EXP is the data pattern EXPected.

- ACT is 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 probably is defective.

If the error occurs in many locations and the bits in error are randomly spaced throughout the word, the memory or bus timing probably is 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/c is testing the K Control Area. A memory location did not contain the expected pattern, where:
  - MA is the address of the failing location.
  - EXP is the data pattern EXPected.
  - ACT is 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 1.
  2. The additional 1 bit occurs immediately to the left of the left-most bit in the EXPected data. For example:

```
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 1's 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 left-most bit in the EXPected data:

     ```
     EXP=000777, ACT=002777
     ```

- **Error 005**—Occurs in the moving inversions test when the P.ioj/c 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 is the address of the failing location.
  - EXP is the data pattern EXPected.
  - ACT is 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 probably is 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/c is testing the K Control Area. A memory location did not contain the expected pattern, where:
  - MA is the address of the failing location.
  - EXP is the data pattern EXPected.
  - ACT is 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 0 than the EXPected data.
  2. The additional 0 occurring in the same bit position as the left-most bit 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 1's than the EXPected data:

```
EXP=037777, ACT=077777
```

3. The bit in error is not in the same bit position as the left-most bit in the EXPected data:

```
EXP=001777, ACT=001377
```

- **Error 007**—Occurs in the moving inversions test when the P.ioj/c 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 is the address of the failing location.
  - EXP is the data pattern EXPected.
  - ACT is 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 probably is 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/c 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/c 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 is the address causing the parity trap.
  - VPC is 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/c detected a nonexistent 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 is the address causing the NXM trap.
  - VPC is 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/c 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 is to 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 off-line loader command TEST MEMORY to test Control memory.
- **Error 013—Cache parity trap, VPC = xxxxxx**—This can happen during any test. The J11/F11 trapped through the parity vector. The error was caused by the cache.

  During the run of the diagnostic, the J11/F11 took a trap through the parity error vector. This is a cache error and the virtual PC at the time of the trap is printed.

6.7 OKPM—Off-line K/P Memory Test

The off-line K/P memory test (OKPM) tests the HSC control and data memories from a K.sdi, K.sti, K.si, or K.ci. OKPM 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.

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 detecting transient errors caused by bus and memory timing problems. Errors are reported at the console terminal as they occur.

6.7.1 System Requirements

Hardware required by OKPM includes:

- I/O Control Processor module with HSC boot ROMs
- Memory module
- Load device and controller with at least one working drive
- Terminal connected to I/O Control Processor console interface
- At least one working K.sdi, K.sti, K.si, or K.ci
- Working Control memory for a K Control Area

6.7.2 Operating Instructions

If the HSC is not booted and loaded, refer to Section 6.1.2 and Section 6.2. If these preceding steps are complete, the ODL> prompt is present. Follow these next steps to start OKPM.

1. Type TEST MEMORY BY K in response to the loader prompt ODL>. The load device LED lights as the memory test is loaded.

2. OKPM indicates it has been loaded properly by displaying the following:

```
HSC OFL K/P Memory Test
```

6.7.3 Test Termination

OKPM can be terminated by typing CTRL/C. The test may continue running for a few seconds after it is terminated.

6.7.4 Parameter Entry

This section describes the various parameters for OKPM.

NOTE
For any of the OKPM prompts, use the DELete key to delete mistyped parameters before terminating the entry with RETURN. If an error in a parameter entry was terminated with RETURN, type CTRL/C to return to the initial prompt and re-enter all parameters.

OKPM 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 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 0 to test Control memory or type 1 to test Data memory. Type RETURN to terminate the response. (Typing just RETURN selects the Control memory test.) The memory test next prompts for the first address to test.

```
First (min=XXXXXXXX) [min] ?
```

Enter the first address to be tested. Addresses are 8 octal digits in length. The [min] address displayed is the lowest address that may be entered for the memory chosen. After typing the address, terminate the response with RETURN. (Typing just RETURN causes the first address to default to the [min] address.)

NOTE
Because requestors test Control memory in 4-byte units, the lowest 2 bits of the starting address are ignored (treated as binary 0's). 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 6 bits of the starting address are ignored (treated as binary 0's). 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 the system being worked on does not have a fully populated memory, the last address that may be tested is less than the max address displayed. If a last address that exceeds the amount of memory in this system is chosen, the memory test displays a Nonexistent Memory (NXM) error when the test reaches the first address beyond the end of the memory. (Use the off-line 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 1's). 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 1's). 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 2147483647 to specify the number of times the memory test is to be repeated. (If 0 or just RETURN is entered, 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] ?
```

To repeat the last test specified using the parameters, answer this prompt with Y or RETURN. To cause the test to prompt for new parameters, answer the prompt with N.

Use the DELete key to delete mistyped parameters before terminating the entry with RETURN. If an error in a parameter was terminated with RETURN, type CTRL/C to return to the initial prompt and re-enter all parameters.

6.7.5 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
where:
  nnnnnn is the number of passes.
  xxxxxx is the number of errors detected during the current pass.
  yyyyyy is the number of errors detected for all passes.
```

6.7.6 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/c detected) is reported, type 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 RETURN. The contents of the I/O Control Processor switch control and status register (SWCSR) are displayed as (I) 17770042 nnnnnn.

4. Type De * nnnn4n. The nnnn4n represents the previous contents of the register, including a 1 in bit 5. I/O Control Processor parity traps are now disabled.
5. Return to the memory test by typing START.
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 through 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. The nnnn0n represents original contents of the I/O Control Processor SWCSR, before parity traps were disabled (refer to step 5).

6.7.7 Test Summaries

The following is a summary of individual K/P memory tests.

- **Test 000, moving inversions test from P.ioj/c**—This 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 0, rewrite the same location with a single 1 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 1 bit. Reread it to verify the Write operation worked properly.
  4. Repeat step 3 until the test pattern consists of a word containing all 1's (pattern 177777).
  5. Repeat step 3, but this time substitute a single extra 0 each time instead of a 1.
  6. Continue step 5 until the test pattern consists of a word of all 0's (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 0's to all 1's and back to all 0's.
  8. End of test. All memory is cleared to 000000.

- **Test 001, moving inversions test from K**—This 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 through the K Control Area. The I/O Control Processor is responsible for displaying the error report.

6.7.8 Error Information

Errors produced by OKPM 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.)

A typical OKPM error message follows:

```
OKPM>hh:mm T aaa E bbb U-000
 < Text describing error >
 MA-xxxxxxxx
 EXP-yyyyyy
 ACT-zzzzzz
 < K-Error-Summary-Info >

where:

  hh is the elapsed hours since last bootstrap.
  mm is the elapsed minutes.
  aaa is the decimal number denoting test.
  bbb is the decimal number denoting the error detected.
  MA-xxxxxxxx is the address of location causing the error.
  EXP-yyyyyy is the data pattern that was expected.
  ACT-zzzzzz is the data pattern that was actually found.
```

6.7.9 Requestor Error Summary

When the requestor reports a memory test failure to the I/O Control Processor, the following information is supplied:

- Address of the failing memory location
- Data EXPected and data ACTually found
- Error summary information

The error summary information is supplied as a 3-bit field, including the following:

- A bit indicating a parity error occurred while reading the location.
- A bit indicating an NXM error occurred while accessing the location.
- 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/c test), a Control bus error is most likely caused by a problem with the requestor's drivers that assert the CCYCLE lines.

6.7.10 Error Messages

Error messages produced by OKPM 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 begins either 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 is 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.

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 Nonexistent Memory (NXM) 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 when the I/O Control Processor is testing the K Control Area. A memory location did not contain the expected pattern, where:
  - MA is the address of the failing location.
  - EXP is the data pattern EXPected.
  - ACT is 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 1.
  2. The additional 1 bit occurs immediately to the left of the left-most bit 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 1's 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 left-most bit in the EXPected data:

     ```
     EXP=000777, ACT=002777
     ```

- **Error 001**—Occurs in the moving inversions test 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 is the address of the failing location.
  - EXP is the data pattern EXPected.
  - ACT is 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 probably is 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 probably is 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 when the I/O Control Processor is testing the K Control Area. A memory location did not contain the expected pattern, where:
  - MA is the address of the failing location.
  - EXP is the data pattern EXPected.
  - ACT is 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 0 than the EXPected data.
  2. The additional 0 occurs in the same bit position as the left-most bit 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 1's than the EXPected data:

```
EXP=037777, ACT=077777
```

3. The bit in error is not in the same bit position as the left-most bit in the EXPected data:

```
EXP=0017777, ACT=00377
```

- **Error 003**—Occurs in the moving inversions test 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 is the address of the failing location.
  - EXP is the data pattern EXPected.
  - ACT is 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 probably is 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 probably is 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 when the I/O Control Processor is testing the K Control Area. A memory location did not contain the expected pattern, where:
  - MA is the address of the failing location.
  - EXP is the data pattern EXPected.
  - ACT is 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 1.
  2. The additional 1 bit occurs immediately to the left of the left-most bit 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 1's 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 left-most bit in the EXPected data:

   ```
   EXP=000777, ACT=002777
   ```

- **Error 005**—Occurs in the moving inversions test 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 is the address of the failing location.
  - EXP is the data pattern EXPected.
  - ACT is 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 probably is 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 when the I/O Control Processor is testing the K Control Area. A memory location did not contain the expected pattern, where:

  - MA is the address of the failing location.
  - EXP is the data pattern EXPected.
  - ACT is 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 0 than the EXPected data.

2. The additional 0 occurs in the same bit position as the left-most bit 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 1's than the EXPected data:

```
EXP=037777, ACT=077777
```

3. The bit in error is not in the same bit position as the left-most bit in the EXPected data:

```
EXP=001777, ACT=001377
```

- **Error 007**—Occurs in the moving inversions test 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 is the address of the failing location.
  - EXP is the data pattern EXPected.
  - ACT is 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 probably is 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 probably is 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 is the 22-bit address of the failing location.
  - EXP is the data pattern EXPected by the K.
  - ACT is the 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 Nonexistent 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 6.7.9 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 is the address causing the parity trap.
  - VPC is 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 as described in Section 6.7.6. 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 Nonexistent Memory (NXM) trap. A 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, where:
  - MA is the address causing the NXM trap.
  - VPC is 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 the 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 is to 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 off-line 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.

6.8 OMEM—Off-line Memory Test

The off-line memory test (OMEM) exercises the HSC memories. Control, Data, or Program memory may be selected for testing. Three memory testing algorithms are used: the quick verify algorithm, the moving inversions algorithm, and the walking 1's 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.

6.8.1 System Requirements

Following are the OMEM hardware requirements:

- I/O Control Processor module with HSC boot ROMs
- Memory module
- Load device with at least one working drive
- Terminal connected to I/O Control Processor console interface

6.8.2 Operating Instructions

If the HSC is not booted and loaded, refer to Section 6.1.2 and Section 6.2. If the HSC is booted and loaded, the terminal displays an ODL> prompt. At this point, follow these steps to start OMEM:

1. Type SIZE in response to the loader prompt ODL>.

   The load device drive-in-use LED lights as the off-line 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 load device drive-in-use LED lights as OMEM is loaded. OMEM test indicates it has been loaded properly by displaying the following:

   ```
   HSC OFL Memory Test
   ```

6.8.3 Test Termination

OMEM can be terminated by typing CTRL/C.

6.8.4 Parameter Entry

This section describes the OMEM parameter entry.

NOTE
For any of the OMEM prompts, use the DELete key to delete mistyped parameters before the typing RETURN. If an error in a parameter already terminated with RETURN is noted, type CTRL/C to return to the initial prompt and re-enter all parameters.

The following are the parameters that can be modified:

```
Control(0), Data(1), or Program(2) Memory [0] ?
```

Type 0 to test control memory.

Type 1 to test data memory.

Type 2 to test program memory.

The memory test next prompts for the first address to test.

```
First (min=XXXXXXXX) [min] ?
```

Enter the first address to be tested. Addresses are 8 octal digits long. The default is the lowest address that may be entered for the memory chosen

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 in the memory chosen. 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 an address exceeds the memory in the system, the memory test displays a nonexistent memory (NXM) error when the test reaches the first address beyond the end of the 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 0 results in one pass.)

After the first memory test is complete, the following prompt is issued:

```
Reuse parameters (Y/N) [Y] ?
```

To repeat the last test specified using the parameters, answer this prompt with Y or RETURN. To cause the test to prompt for new parameters, answer the prompt with N.

Use the DELete key to delete mistyped parameters before terminating the entry with RETURN. If an error in a parameter was terminated with RETURN, type CTRL/C to return to the initial prompt and re-enter all parameters.

6.8.5 Progress Reports

A complete pass through OMEM consists of one pass through the quick verify test, one pass through the moving inversions test, and one pass through the walking 1's test. After each complete pass, an end-of-pass message is displayed as follows:

```
End of Pass nnnnnn, xxxxxx Errors, yyyyyy Total Errors

where:

  nnnnnn is a decimal total of the complete passes made.
  xxxxxx is the number of errors detected on the current pass.
  yyyyyy is number of errors detected during the passes completed so far.
```

NOTE
A complete pass through the memory test for program memory may take about 8 hours. Unless exhaustive memory testing is required, allow this test to run only until the quick verify pass complete message is displayed. This takes no more than 10 minutes.

6.8.6 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/c detected) is reported, type 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. The contents of the I/O Control Processor switch control and status register (SWCSR) are displayed as (I) 17770042 nnnnnn.
4. Type De * nnnn4n. The nnnn4n represents the previous contents of the register, including a 1 in bit 5. I/O Control Processor parity traps are now disabled.
5. Return to the memory test by typing START.
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 through 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. The nnnn0n represents original contents of the I/O Control Processor SWCSR, before parity traps were disabled (refer to step 5).

6.8.7 Test Summaries

The following list describes the three algorithms used by OMEM.

- **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 000000 to each location of the memory

FOR i = First to Last address

IF < location i does not contain 0 >
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 000000 in each location of the memory.

2. Reads all locations in order from lowest to highest. After reading a location and checking for a 0, rewrites the same location with a single 1 in the least significant bit. Then rereads the location and verifies the Write worked correctly.
3. Again reads all locations in order from lowest to highest. Checks that each location contains the data previously written. Rewrites the data found with a single additional 1 bit. Rereads it to verify the Write operation worked properly.
4. Repeats step 3 until the test pattern consists of a word containing all 1's (pattern 177777).
5. Repeats step 3 but this time substitutes a single extra 0 each time instead of a 1.
6. Continues step 5 until the test pattern consists of a word of all 0's (pattern 000000).
7. Repeats steps 1 through 6 but this time starts at the highest memory address each time and works down to the lowest. This writes each memory location from all 0's to all 1's and back to all 0's.
8. Clears all memory to 000000.

- **Test 002, walking 1's test**—An algorithm that stresses semiconductor memories and is effective in locating timing problems on the memory module or on the bus.

  The walking 1's algorithm performs the following:

  1. Writes all memory to 0's (pattern = 000000).
  2. Checks all memory for 0's. Declares error 008 if not 0.
  3. Sets TESTADDRESS equal to the first address to test.
  4. Writes 177777 to contents of TESTADDRESS.
  5. Checks that all other locations are equal to 000000. Declares an error 009 if not equal to 000000.
  6. Checks that TESTADDRESS contains 177777. Declares an error 010 if not equal to 177777.
  7. Writes 000000 to contents of TESTADDRESS.
  8. IF TESTADDRESS is the last address to be tested, testing is complete. If TESTADDRESS is not the last address to be tested, 2 will be added to TESTADDRESS and the process will go back to step 4. This will continue until TESTADDRESS is the last address to be tested.

6.8.8 Error Information

OMEM displays the errors detected during execution on the console terminal. All error messages follow the diagnostics generic error message format preceded by an OMEM> prompt.

A typical OMEM error message format follows:

```
OMEM>hh:mm T aaa E bbb U-000
 < Text describing error >
 MA-xxxxxxxx
 EXP-yyyyyy
 ACT-zzzzzz

where:

  hh is the elapsed hours since last bootstrap.
  mm is the elapsed minutes.
  aaa is the decimal number denoting test.
  bbb is the decimal number denoting the error detected.
  MA-xxxxxxxx is the address of location causing the error.
  EXP-yyyyyy is the data pattern that was expected.
  ACT-zzzzzz is the data pattern that was actually found.
```

Parity trap and NXM trap errors do not include expected and actual data.

6.8.9 Error Messages

Error messages produced by OMEM 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; errors 008 through 010 are declared in the walking 1's algorithm.

Errors 011 and 012 are unexpected trap errors. Error 011 signifies a parity trap occurred and error 012 indicates a nonexistent 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 (Section 6.8.7). A memory location did not contain the expected pattern, where:
  - MA is the address of the failing location.
  - EXP is the data pattern EXPected.
  - ACT is 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. 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 1.
  2. The additional 1 bit occurs immediately to the left of the left-most bit 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 1's 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 left-most bit in the EXPected data:

   ```
   EXP=000777, ACT=002777
   ```

- **Error 001**—Occurs in the moving inversions test (Section 6.8.7) 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, where:
  - MA is the address of the failing location.
  - EXP is the data pattern EXPected.
  - ACT is 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 probably is 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 probably is defective.

  If the error occurs in many locations and the bits in error are randomly spaced throughout the word, the memory or bus timing probably is the problem.

  If the error occurs in more than one location but the addresses of the failing locations are similar and 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 6.8.7). A memory location did not contain the expected pattern, where:
  - MA is the address of the failing location.
  - EXP is the data pattern EXPected.
  - ACT is 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 0 than the EXPected data.
  2. The additional 0 occurs in the same bit position as the left-most bit 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 1's than the EXPected data:

   ```
   EXP=037777, ACT=077777
   ```

3. The bit in error is not in the same bit position as the left-most bit in the EXPected data:

   ```
   EXP=001777, ACT=001377
   ```

- **Error 003**—Occurs in the moving inversions test (Section 6.8.7). A location was written with a pattern. Immediately after the write, the location was read and found to contain an incorrect pattern, where:
  - MA is the address of the failing location.
  - EXP is the data pattern EXPected.
  - ACT is the 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 probably is 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 (Section 6.8.7) when the I/O control processor is testing the K Control Area. A memory location did not contain the expected pattern, where:
  - MA is the address of the failing location.
  - EXP is the data pattern EXPected.
  - ACT is 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. 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 1.
2. The additional 1 bit occurring immediately to the left of the left-most bit 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 1's 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 left-most bit in the EXPected data:

   ```
   EXP=000777, ACT=002777
   ```

- **Error 005**—Occurs in the moving inversions test (Section 6.8.7) 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 and it contained an incorrect pattern, where:

  - MA is the address of the failing location.
  - EXP is the data pattern EXPected.
  - ACT is 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 probably is 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 probably is 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 6.8.7) when the I/O control processor is testing the K Control Area. A memory location did not contain the expected pattern, where:

  - MA is the address of the failing location.

- EXP is the data pattern EXPected.
- ACT is 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 0 than the EXPected data.
2. The additional 0 occurs in the same bit position as the left-most bit 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 1's than the EXPected data:

   ```
   EXP=037777, ACT=077777
   ```

3. The bit in error is not in the same bit position as the left-most bit in the EXPected data:

   ```
   EXP=001777, ACT=001377
   ```

- **Error 007**—Occurs in the moving inversions test (Section 6.8.7) 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 and found to contain an incorrect pattern, where:
  - MA is the address of the failing location.
  - EXP is the data pattern EXPected.
  - ACT is 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 probably is 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 probably is 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 1's test (Section 6.8.7). All locations in the memory under test were written with the pattern 000000. Then all locations were read to check that they contained 000000. When the location specified in the error report was read, it did not contain 000000, where:

  - MA is the address of the failing location.
  - EXP is the data pattern EXPected (000000).
  - ACT is the data pattern ACTually found.

  Because all locations were cleared to 000000 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 probably is 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 1's test (Section 6.8.7). One location in the memory under test was written with the pattern 177777 and all the other locations should contain the pattern 000000. While reading to check that all other locations are clear, a location was found containing something other than 000000, where:

  - MA is the address of the failing location.
  - EXP is the data pattern EXPected (000000).
  - ACT is the 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 177777. During this part of the test, one location in the memory was written to 177777. 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 177777. 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* 177777. 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 probably is 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 1's test (Section 6.8.7). At this step of the test, one location in the memory under test was set to the pattern 177777 and all other locations were cleared to 000000. After checking that all other locations contain 000000, the location that should contain 177777 was read. It contained some other pattern, where:

  - MA is the address of the failing location.

- EXP is the data pattern EXPected (177777).
- ACT is the data pattern ACTually found.

Because only Read operations were performed after writing the 177777, a dual-addressing problem is highly improbable.

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 probably is 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 is the address of the location causing the parity trap.
  - VPC is 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. To further localize the problem, disable parity errors and rerun the test. (Refer to Section 6.7.6.)

  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 nonexistent memory (NXM) 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 is the address being tested at the time the NXM trap occurred.
  - VPC is 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.

  This error frequently occurs when trying to test beyond system memory addresses.

  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 to which the module is to 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 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 is 0. Error 013 indicates a location contain something besides a 0 before the first pattern was written.

If the ACTual data in the error report is 031463(8) or 146314(8), a dual-addressing problem probably is the cause of the error. (When an address lower in memory was written with a test pattern, the failing location also was 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 1 state. The first pattern written is 146314(8). The second pattern written is the 1's 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 off-line 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.

6.9 OFLRXE—RX33 Off-line Exerciser

OFLRXE is a combined hardware diagnostic and exerciser for the HSC 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 standalone diagnostic running under the off-line 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.

6.9.1 System Requirements

To run OFLRXE, the HSC must be booted from the off-line diagnostic diskette. When the system is booted from this media, the ODL> prompt is displayed. Hardware and software requirements are:

- P.ioj module
- M.std2 memory/controller module
- At least one RX33 drive
- One scratch diskette for each drive to be tested (maximum of two)
- Testing of the J11 chip set and cache is assumed if it is turned on
- Two tested 4-Kword partitions of memory

6.9.2 Operating Instructions

If the HSC is not booted and loaded, refer to Section 6.1.2 and Section 6.2. If the HSC is already booted and displaying the off-line loader prompt ODL>, proceed as follows:

At the ODL> prompt, invoke OFLRXE by typing TEST RX. This loads the OFLRXE from the media, and transfers control to the diagnostic. At the start, the diagnostic prints out the following string:

```
HSC Off-line RX33 Exerciser Vxxx
  Where: Vxxx is a 3-digit version/edit number.
```

NOTE
If unable to boot from drive 0, move the diskette to drive 1, try again, or use a backup copy of the off-line diagnostics diskette.

6.9.3 Test Termination

OFLRXE can be terminated by typing CTRL/C or CTRL/Y. OFLRXE also terminates on expiration of the allotted time or on fatal errors.

6.9.4 Parameter Entry

Following are the OFLRXE modifiable parameters. Drive selection is prompted for by the program in the following manner:

```
Test drive n (Y/N) [Y] ?
  Where: n is the drive number (0 or 1)
```

The default is Y. The prompt repeats for each available diskette on the HSC. The test prompts if the initial Write operation is to be performed:

```
Perform initial write on this drive (Y/N) [Y] ?
```

The default is Y. 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 the previous prompts have been answered, the program directs placement of a scratch diskette in the selected drive:

```
Insert a scratch diskette in the drive, type a carriage return to continue.
```

At this point, insert the 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. 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 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 selected 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 prompts you by printing:

```
Reuse parameters (Y/N) [Y] ?
```

To repeat the last test specified using the parameters, answer this prompt with Y or RETURN. To cause the test to prompt for new parameters, answer the prompt with N.

Use the DELete key to delete mistyped parameters before terminating the entry with RETURN. If an error in a parameter was terminated with RETURN, type CTRL/C to return to the initial prompt and re-enter all parameters.

6.9.5 Progress Reports

OFLRXE does not run in a conventional sense. There are no pass-completed messages. Instead, informational messages are printed indicating what the exerciser is doing. The program has a user-requested status available. If you type CTRL/T, the program responds:

```
Number of sectors transferred = xxxxxx, yyyyyy errors.

where:

  xxxxxx is a 16-digit number of sectors successfully transferred.
  yyyyyy is a 6-digit cumulative number of errors detected.
```

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 you. When the requested time has expired, the program prints the following string:

```
    Exerciser completed.
```

The program then returns to the parameter entry routine.

6.9.6 Test Summaries

The following is a summary of OFLRXE tests.

- **Test 1, RX33 controller registers**—Performs stuck-at testing on the RX33 controller registers at 17777400, 17777402, 17777404, and 17777406. A simple walking 1's test is performed on each register, except for the CSR register at 177400, which only has the high byte tested.
- **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.
- **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.
- **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 also is lumped into this test. Correct handling of NXM errors and correct reporting by the error bit in the CSR is checked.
- **Test 5, verify track counters and registers**—Uses the step function of the diskette controller chip to verify that 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, and so forth). The verify option is set on each step command so the diskette controller reads headers on each track to verify position.

- **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;
   CHECK_STATUS;
   If outcnt <> rxtrk then error 11
   outcnt =outcnt-1;
   seek incnt;
   CHECK_STATUS;
     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.

- **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 is compared to the data that was written. This test takes about 10 minutes per drive.

- **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 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 is to 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.

6.9.7 Data Patterns

Four unique data patterns were selected to give maximum delta of frequency with the modified frequency modulation (MFM) encoding used on the RX33. These patterns are shown in Example 6–2.

```
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
-----------------+--------------------------------------
```

Example 6–2 Off-line RX33 Exerciser Data Patterns

6.9.8 Error Information

A generic message format for all off-line diagnostic errors is found in Section 6.2.7. The following section contains information on specific errors associated with OFLRXE.

A typical OFLRXE error message follows:

```
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 a 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.

6.9.9 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 shows 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, the priority at the time of the interrupt can be determined.

- **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 probably is fundamental; the controller cannot read the diskette at all. If the errors occur in a random fashion, the problem probably is 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 probably is 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 probably is 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 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 problem point. 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 there is not a write-protected diskette in the drive). This error terminates the diagnostic, as a write-protected diskette cannot be written on.
- **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 probably is 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; 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, Parity Error Was Not Forced in Main Memory**—Applicable to test 4. Indicates a write to Program memory with bad parity set (bit 11 of the CSR) 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.

6.10 ORFT—Off-line Refresh Test

The off-line memory refresh test (ORFT) finds memory problems related to refresh. Patterns are written to memory and then checked after waiting 1 minute. Three separate patterns are used to test each memory bit (including parity bits) in both the 1 and 0 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.

ORFT can find problems in the memories not detected by the normal memory tests. ORFT is not intended to be run on memories that fail the normal memory tests.

6.10.1 System Requirements

The following hardware is required to run ORFT:

- I/O Control Processor module with HSC boot ROMs
- Memory module that passes the off-line memory test and/or the off-line K/P memory test
- HSC load device with at least one working drive
- Terminal connected to I/O Control Processor console interface

ORFT assumes the HSC memories pass both the off-line memory test and the off-line K/P memory test. In addition, ORFT assumes the memories are working except for the refresh circuitry.

6.10.2 Operating Instructions

If the HSC is not booted and loaded, refer to Section 6.1.2, and Section 6.2. If the HSC is already booted and displaying the off-line loader prompt ODL>, proceed as follows:

1. Type TEST REFRESH in response to the prompt ODL>.
2. ORFT indicates it is loaded properly by displaying the following:

   ```
   HSC OFL Memory Refresh Test
   ```

3. The refresh test now prompts for parameters.

6.10.3 Test Termination

ORFT can be terminated by typing CTRL/C.

6.10.4 Parameter Entry

This section describes the prompts for the ORFT parameters.

NOTE
For any of the ORFT prompts, use the DELete key to delete mistyped parameters before typing RETURN. If an error in a parameter already terminated with RETURN is noted, type CTRL/C to return to the initial prompt and re-enter all parameters.

ORFT first prompts with:

```
# 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 is to 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 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] ?
```

To repeat the last test specified using the parameters, answer this prompt with Y or RETURN. To cause the test to prompt for new parameters, answer the prompt with N.

Use the DELete key to delete mistyped parameters before terminating the entry with RETURN. If an error in a parameter was terminated with RETURN, type CTRL/C to return to the initial prompt and re-enter all parameters.

6.10.5 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

where:

  nnnnnn is a decimal total of the complete passes made.
  xxxxxx is the number of errors detected on the current pass.
  yyyyyy is number of errors detected during the passes completed so far.
```

6.10.6 Test Summaries

The following are the ORFT test summaries.

- **Test 01, pattern 177777**—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 off-line 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 the pattern. All of Program memory not occupied by the refresh test and the off-line 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 off-line loader is also filled with the pattern. After filling the memories, the program delays for 1 minute, then each memory location is read and checked for the pattern. Any errors detected are reported on the terminal.

6.10.7 Error Information

All error messages produced by ORFT conform to the HSC diagnostic error message format (refer to Section 6.2.7). Following is a typical ORFT error message.

```
ORFT>hh:mm T aaa E bbb U-000
 < Text describing error >
 MA-xxxxxxxx
 EXP-yyyyyy
 ACT-zzzzzz

where:

  hh is the elapsed hours since last bootstrap.
  mm is the elapsed minutes.
  aaa is the decimal number denoting test.
  bbb is the decimal number denoting the error detected.
  MA-xxxxxxxx is the address of location causing the error.
  EXP-yyyyyy is the data pattern that was expected.
  ACT-zzzzzz is the data pattern that was actually found.
```

6.10.8 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 off-line 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 1 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 is the address causing the parity trap.
  - VPC is 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. 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 NXM 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 the 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 is to 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.

6.11 OOCP—Off-line Operator Control Panel (OCP) Test

OOCP checks the operation of the HSC lamps and switches. Testing includes the five OCP lamps and switches, the State LED, Secure/Enable switch, and the enable LED.

This section includes troubleshooting procedures for localizing faults detected by this test.

6.11.1 System Requirements

The following hardware is required to run OOCP:

- I/O Control Processor module with HSC boot ROMs
- Memory module
- HSC load device with at least 1 working drive
- Terminal connected to I/O Control Processor console interface
- OCP

Due to the sequence of tests that precede this test, it is safe to assume the I/O Control Processor module, Program memory, and HSC load device are tested and working.

6.11.2 Operating Instructions

If the HSC is not booted and loaded, refer to Section 6.1.2, and Section 6.2. If the HSC is already booted and displaying the off-line loader prompt ODL>, proceed as follows:

Type TEST OCP in response to the ODL> prompt. The HSC load device in motion LED is ON.

The test indicates it is loaded properly by displaying the following message:

```
HSC OFL OCP Test
```

6.11.3 Test Termination

The test may be aborted at any time by typing CTRL/C.

6.11.4 Parameter Entry

OOCP first checks the position of the Secure/Enable switch through 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, OOCP 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 6.11.8. The program waits until the Secure/Enable switch is changed to the enable position and issues the following message:

```
(Enable LED is lit, State LED is 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, and 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 6.11.8. 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 1's (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 6.11.8.

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 1 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 6.11.8. 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 6.11.8. 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 has no effect. (Do not press any other switch or an error message results.) If the HSC 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 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 does not cause the J11/F11 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 6.11.8. After pressing the break key, type 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 off-line loader.

To repeat the last test specified using the parameters, answer this prompt with Y or RETURN. To cause the test to prompt for new parameters, answer the prompt with N.

Use the DELete key to delete mistyped parameters before terminating the entry with RETURN. If an error in a parameter was terminated with RETURN, type CTRL/C to return to the initial prompt and re-enter all parameters.

6.11.5 Test Summaries

The following sections summarize test 000 through test 009.

- **Test 000, observe enable and State LEDs**—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 HSC card cage). If neither State LED is blinking, the problem probably is caused by the bit in the I/O Control Processor CSR register that controls the State LED (refer to Section 6.11.8.4). If one of the State LEDs is blinking but the other is not, the nonblinking LED probably is wired wrong or is faulty.

- **Test 001, lamp test through Fault switch**—Performs an automatic lamp test. When the Fault switch is pressed, all lamps 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 1). 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 6.11.8.1 to localize the problem.

- **Test 003, Fault switch**—Directs pressing the lit Fault 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 properly wired.
  - Fault switch bit in the I/O Control Processor CSR cannot be set.

  Refer to the troubleshooting procedures in Section 6.11.8.

  If pressing the Fault switch results in an error message, refer to Section 6.11.7.

- **Test 004, Online switch**—Directs pressing the lit Online 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.

  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 6.11.8.

  If pressing the Online switch results in an error message, refer to Section 6.11.7.

- **Test 005, first unmarked switch**—Directs pressing the lit first unmarked 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.

  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 properly wired.
  - First unmarked switch bit in the I/O Control Processor CSR cannot be set.

  Refer to the troubleshooting procedures in Section 6.11.8.

  If pressing the first unmarked switch results in an error message, refer to Section 6.11.7.

- **Test 006, second unmarked switch**—Directs pressing the lit second unmarked 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.

  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 cannot be set.

  Refer to the troubleshooting procedures in Section 6.11.8.

If pressing the second unmarked switch results in an error message, refer to Section 6.11.7.

- **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 0, indicating the switch is in the secure position. Then the program tells the operator to observe that 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. The test prompts for the Init switch to be pressed 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 HSC 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 6.11.7 for further information.

- **Test 009, break key in secure mode**—Checks if 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/F11 CPU to halt and enter ODT.) The prompt is to press the break key and to observe if the HSC 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/F11 CPU.

6.11.6 Error Information

All error messages produced by OOCP conform to the HSC diagnostic error message format (refer to Section 6.2.7). A typical OOCP error message follows:

```
OOCP>hh:mm T aaa E bbb U-000
 < Text describing error >
 MA-xxxxxxxx
 EXP-yyyyyy
 ACT-zzzzzz
```

```
where:

  hh is the elapsed hours since last bootstrap.
  mm is the elapsed minutes.
  aaa is the decimal number denoting test.
  bbb is the decimal number denoting the error detected.
  MA-xxxxxxxx is the address of location causing the error.
  EXP-yyyyyy is the data pattern that was expected.
  ACT-zzzzzz is the data pattern that was actually found.
```

6.11.7 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 other than the switch bit being tested set in the I/O Control Processor switch/display register. 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 media address (MA) 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 and ACTual data each consist of only one bit, the failure was caused by either 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 6.11.8.

- **Error 001, bit set when Init is 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 media address (MA) field of the error report gives the address of the I/O Control Processor switch display register. The EXPected data is always 0 (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 6.11.8.

6.11.8 Troubleshooting Registers and Displays through ODT

The following paragraphs and layouts are included to assist you with troubleshooting.

6.11.8.1 Switch Check through ODT

To check the operation of an HSC 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/F11 CPU halts and displays 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. Type a slash (/) to re-examine the switch display register.
5. To restart the off-line loader (or the diagnostic that was interrupted), type RETURN, then type P.

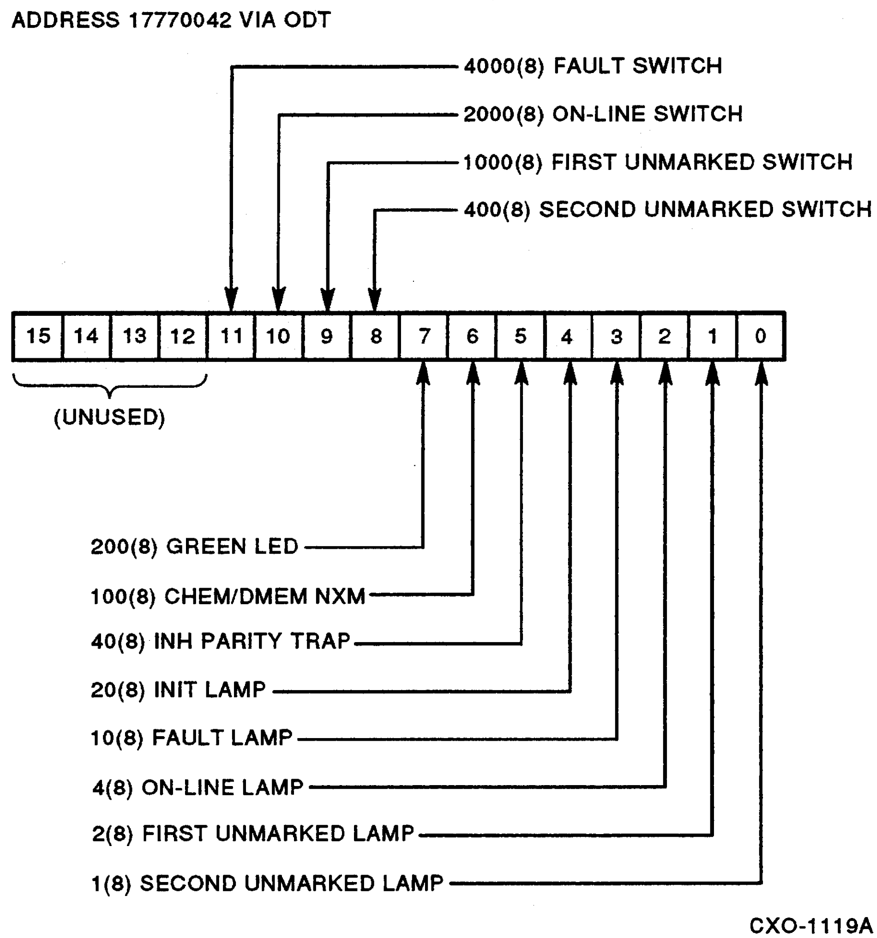


Figure 6–1 P.ioj Switch Display Register Layout

Using this method, the switch bits of the switch/display register can be monitored when various switches are in the ON or OFF position.

6.11.8.2 Lamp Bit Check

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/F11 CPU halts and displays an @ symbol.

2. Type 17770042 RETURN.

   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 lights.

4. To light a lamp, type the octal value that corresponds to the proper lamp, then type RETURN. The lamp lights.

5. Type a slash (/) to re-examine the contents of the switch/display register.

6. Type RETURN to restart the off-line loader (or the diagnostic that was interrupted), then type a P.

Using this method, various lamps can be manually enabled or disabled.

6.11.8.3 Secure/Enable Switch Check

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 HSC is stuck in the secure mode, this method cannot be used because break is disabled.)
2. The I/O Control Processor J11/F11 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. Figure 6–2 identifies the various bits of this register.

   When the Secure/Enable switch is in the enable position, the contents of the register is 1xxxxx. When in the secure position, the contents is 0xxxxx.
5. Type RETURN and then a slash (/) to re-examine the register.
6. Type RETURN, then type P to restart the off-line loader (or the diagnostic that was interrupted).

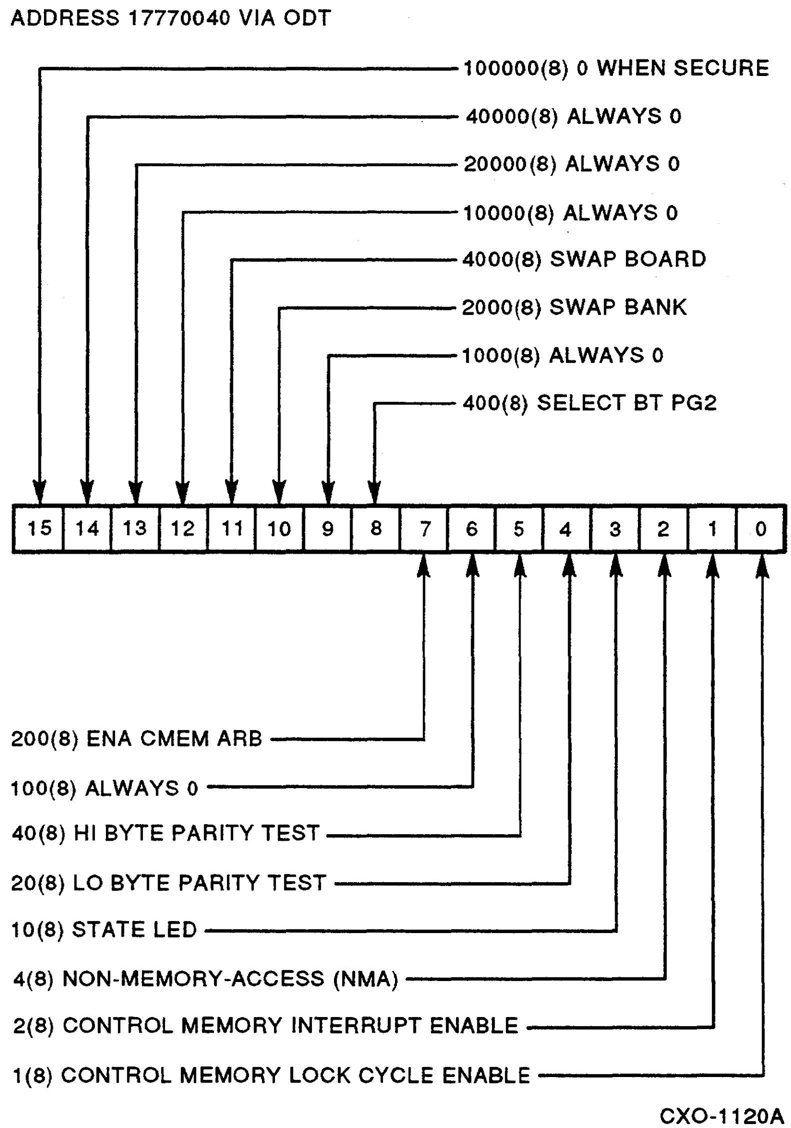


Figure 6–2 P.ioj Control and Status Register Layout

6.11.8.4 State LED Check

There are two State LEDs in the HSC. One is on the OCP, far left. The other State LED is on the I/O Control Processor module (rightmost module in the HSC 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/F11 CPU halts and displays 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 RETURN. To extinguish the State LED, put a 0 in the same bit position and press RETURN.

   CAUTION
   Bit 7 of the I/O Control Processor CSR must be set to allow the HSC Ks to access Control memory. The setting of other bits in the CSR can result in side effects. Be careful not to set any bits except the State LED bit and leave bit 7 set when done.

5. Type a slash (/) to re-examine the contents of the I/O Control Processor CSR.
6. To restart the off-line loader (or the diagnostic that was interrupted), type RETURN, then type P.

7 Utilities

7.1 Introduction

This chapter contains the information required to run the following off-line utilities:

- DKUTIL — Off-line Disk Utility
- VERIFY — Off-line Disk Verify Utility
- FORMAT — Off-line Disk Formatter Utility
- PATCH — Off-line Load Media Modification Utility

The HSC must be in the command mode before running the off-line utilities. Type CTRL/Y to get the command prompt.

Topics covered in this chapter include initiating the utility, using commands, and interpreting error messages. These HSC utilities are interactive and therefore are prompt-oriented. Note that prompt information displayed in square brackets is the default.

Refer to the *HSC User Guide* for information on other HSC utilities that are not documented in this manual.

7.2 DKUTIL — Off-line Disk Utility

DKUTIL is a general utility for displaying disk structures and disk data. Unlike some other utilities, DKUTIL is a command language interpreter. It is intended for use in debugging utilities, diagnostics, error recovery, and bad block replacement. DKUTIL has become a general utility for displaying disk structure and data.

Initially, the program goes into command mode. The user must then issue a GET command to obtain the unit to which other commands are to be applied. DKUTIL then returns to the command mode, prompting for a command, executing it, and prompting for another command. DKUTIL is terminated by CTRL/C, CTRL/Y, CTRL/Z, or the EXIT command.

7.2.1 Starting DKUTIL

DKUTIL is started with the standard CRONIC command syntax RUN DKUTIL.UTL. It immediately enters the command mode. A drive must be acquired and brought on line before any other commands can be executed. Type GET Dnnn to acquire the drive and bring it on line.

```
DKUTIL> GET Dnnn
```

The format for entering the drive is a D followed by the unit number. If the drive parameter is omitted, DKUTIL defaults to D000 (unit 0).

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 on line 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. The program returns to the command mode and prompts for a command.

```
DKUTIL>
```

Comment lines can be entered by prefixing them with an exclamation point (!). A null line is ignored. Entering CTRL/Z terminates the program. Commands are executed immediately and take only the time necessary to print the results. Entering CTRL/Y or CTRL/C at any time aborts the program and releases the drive.

7.2.2 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 be 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 descriptions explained in this chapter, only the command (or part of the command) in **bold print** must be specified.

Some command options take optional parameters which, if omitted, default.

7.2.3 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 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 through the DEFAULT command. The initial default modifiers for DUMP are /DATA, /EDC, and /IFERROR.

7.2.4 Command Descriptions

This section contains the DKUTIL command descriptions. Command options are shown by separate lines in the syntax specification. Parameters are indicated lowercase in the syntax by braces ({}). Options indicated by brackets ([ ]) can be omitted.

7.2.4.1 Command Summary

A summary of all DKUTIL commands follows:

- DEFAULT—Change default modifiers for DUMP command.

  DEFAULT

- 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}]
  DUMP LBN [{block}]
  DUMP RBN [{block}]
  DUMP RCT [BLOCK {number}] [COPY {copy}]
  DUMP XBN [{block}]

- EXIT—Terminate execution of the program.

  EXIT

- GET—Acquire or change the current drive.

  GET [{drive}]

- POP—Restore save buffer to current buffer.

  POP

- PUSH—Save current buffer in save buffer.

  PUSH

- REVECTOR—Force bad block replacement for the given LBN(s).

  REVECTOR {block} [block]

- SET—Change various program parameters.

  SET [SIZE {size}]

7.2.4.2 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.
  - /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 is also specified.
  - /HEADERS (NOHEADERS) (defaults OFF)—Displays the header fields in the buffer.
  - /ALL (NONE)—The 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 non-primary) LBNs. If /IFERROR is in effect, this modifier applies only to dumping a revectored LBN.
  - /NZ (NONZ) (defaults OFF)—Prevents the data from being displayed if it is all 0's. Instead, a single line indicating the data is 0 is printed. It has no effect if the /DATA modifier is not specified or if it is defaulted OFF.
  - /BBR (NOBBR) (defaults OFF)—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 are:

  DEFAULT/NONE
  DEF/RAW/NODATA
  DE/A/OR/NZ

7.2.4.3 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 the letter O.
- 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, disk characteristics, 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, number 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 resume 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 numbers in decimal and octal, cylinder, group, track, position, and PBN in decimal and octal. For RBNs, the RCT block numbers and offset also are displayed. For LBNs, the primary RBN number and its RCT block number and offset also are 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. Current drives display only 16 bytes of error log data. Succeeding drives display the error log header and all selected error log entries.
- — **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 non-primary revector is distinguished by the character sequence "*->".
- — Examples are:

```
DISPLAY/FULL ALL
DI/F A
DI C D
DIS CHAR LBN 1000
DI/NOI RCT
```

7.2.4.4 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 the letter O.
  - — 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 the letter D. The value must be in the range 1 through nonpad area of the 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.
  - — **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 0.

- **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 of DUMP command parameters are:

```
DUMP RCT BLOCK 3 COPY 4
DU/NZ RCT C 2
DU LBN 1000
D F B 2
D X
D/DATA
```

- 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 7.2.4.2.
  - /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 is also specified.
  - /HEADERS (**NOHEADERS**) (defaults OFF)—Displays the header fields in the buffer.
  - /ALL (**NONE**)—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 applies only to dumping a revectored LBN.
  - /NZ (**NONZ**)—Prevents data from being displayed when it is all 0's. Instead, a single line prints indicating the data is 0's. /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.

7.2.4.5 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 are:

  EXIT
  E

7.2.4.6 GET Command

The GET command is outlined as follows:

- Purpose: To obtain a drive or 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—Allows the reading of FCT block 0 to determine the mode and the reading and writing of RCT block 0 to verify the RCT is valid. If this modifier is specified, the IMF MSCP modifier is not used in the on-line mode and these actions take place. By default, a new drive is brought on line with the IMF (MD.IMF) MSCP modifier.
  - /WP—Brings the drive on line 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 on line with the MSCP SET WRITE PROTECT modifier. The drive is *not* software or volume write-protected.
  - NOONLINE—The drive is acquired but not brought on line with the MSCP on-line command. Only the display characteristics, display errors, and the set size commands can be executed on a drive in this state.
- Usage: The current drive is released. The new drive is acquired and then brought on line with the requested modifiers and unit flags. If the drive is nonexistent, in use, or inoperative, the user is put back in command mode. 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 are:

  GET D133
  G/WP D64
  G

7.2.4.7 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 are:

  POP
  P

7.2.4.8 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 previously in the current buffer is saved in the save buffer. The data in the save buffer is lost. Examples are:

  PUSH
  PU

7.2.4.9 REVECTOR Command

The REVECTOR command is outlined as follows:

- Purpose: To force bad block replacement for one or more given LBNs.
- Syntax: **REVECTOR** {block} [{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 the letter O.
- Modifiers: None.
- Usage: The specified LBNs are 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 are:

  REVECTOR 1000
  R 100
  R 200 210

7.2.4.10 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 bytes.
- Modifier: None.
- Usage: SET 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 are:

  SET SIZE 576
  S S 512

7.2.5 Sample Session

The following is a sample session using DKUTIL. User input is indicated in **bold print**. Enter CTRL/Y to get the HSC> command prompt.

```
^Y
HSC> RUN DKUTIL

DKUTIL> GET 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
```

7.2.6 Error and Information Messages

DKUTIL error messages conform to the HSC utility error message format.

7.2.6.1 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, and so forth)
status = MSCP status (an octal number)
text = The actual text in error
XBN = DBN, LBN, and so forth

xCT = FCT or RCT

7.2.6.2 Error Message Severity Levels

Each DKUTIL error message contains the utility name at the start of the message followed by a letter indicating the severity level of the message. These are defined as:

E = Error
F = Fatal
I = Information

7.2.6.3 Fatal Error Messages

The following is a list of the DKUTIL fatal error messages:

- **DKUTIL-F Insufficient resources to RUN!**—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 I/O request was rejected!**—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.

7.2.6.4 Error Messages

The following is a list of the DKUTIL error messages.

- **DKUTIL-E Drive went AVAILABLE**—Prints if the unit selected goes available while DKUTIL is running. DKUTIL then goes into command mode and the user must issue a GET or EXIT command at the DKUTIL> prompt.
- **DKUTIL-E Drive went OFFLINE!**—Prints if the selected unit goes off line while DKUTIL is running. DKUTIL then goes into command mode and the user must issue a GET or EXIT command at the DKUTIL> prompt.
- **DKUTIL-E Illegal response to start-up question**—Prints if an invalid response to a start-up question or to a prompt for the GET command is entered. The program reprompts with the same question.
- **DKUTIL-E Nonexistent unit number**—Prints if the unit number entered does not correspond to any known unit. DKUTIL then goes into command mode and the user must issue a GET or EXIT command at the DKUTIL> prompt.
- **DKUTIL-E Unit is not available**—Prints if the unit requested is unavailable. The unit may be in use by a host or another diagnostic or it may be inoperative. DKUTIL then goes into command mode and the user must issue a GET or EXIT command at the DKUTIL> prompt.
- **DKUTIL-E Cannot bring unit ONLINE**—Prints if the requested unit is available, but the ONLINE command failed. The unit is released, and DKUTIL then goes into command mode and the user must issue a GET or EXIT command at the DKUTIL> prompt.
- **DKUTIL-E Invalid decimal number**—Prints if an invalid decimal number is entered in a command line.
- **DKUTIL-E Invalid octal number**—Prints if the user entered an invalid octal number in a command line.
- **DKUTIL-E Missing parameter**—Prints if a command line is entered with a required parameter missing.
- **DKUTIL-E There is no buffer to dump**—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.
- **DKUTIL-E Missing modifier (only / was specified)**—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**—Prints when an SDI command is rejected by the drive. A DISPLAY ERRORS command for an RA60 drive always generates this message.
- **DKUTIL-E n is an invalid par number; maximum is n**—Prints if an out-of-range number is entered for a BLOCK or COPY value for the DUMP command.
- **DKUTIL-E xxx is an invalid xxx**—Generic error message that 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**—Prints 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**—Prints when FCT or RCT blocks cannot be read correctly with error recovery, 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 xCT Block n are bad**—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, or when the DISPLAY FCT or DISPLAY RCT command is being used.
- **DKUTIL-E Invalid sector size; only 512 and 576 are legal**—Prints if the sector size entered for the SET SIZE command is other than 512 or 576 bytes.
- **DKUTIL-E Revector for LBN n failed, MSCP Status: (status)**—Prints if a revector (using the REVECTOR command) fails. If the status indicated that the drive went OFFLINE or AVAILABLE, DKUTIL goes into command mode.
- **DKUTIL-E Error log corrupted, cannot display header**—Prints when the DISPLAY ERRORS command reads a header that does not begin with the standard FFFB code.
- **DKUTIL-E Error log corrupted, cannot display entries**—Prints when the DISPLAY ERRORS command is unable to read a valid entry from region FFFB.
- **DKUTIL-E Unable to read error log**—Prints when the DISPLAY ERRORS command is unable to execute the read memory command.
- **DKUTIL-E Error log not implemented in drive**—Prints when the DISPLAY ERRORS command is executed on an RA60.
- **DKUTIL-E Drive must be acquired to execute this command**—Prints if the requested command requires that a drive must first be acquired before the command can be executed. A drive can be acquired and not brought on line by using the /NOONLINE modifier with the GET command.
- **DKUTIL-E Drive must be on line to execute this command**—Prints if the requested command requires that a drive first be acquired and brought on line before the command can be executed.

7.2.6.5 Information Messages

DKUTIL has the following information message:

- **DKUTIL-I CTRL/Y or CTRL/C Abort!**—Termination message that prints if DKUTIL is aborted by typing CTRL/Y or CTRL/C.

7.3 VERIFY — Off-line Disk Verifier Utility

VERIFY is a utility that checks the integrity of the disk architectural structure. This utility checks 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 which should be in the RCT but is not is found, 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 requested.

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 prints saying the drive is OK. Otherwise, the message indicates the number of inconsistencies.

7.3.1 Running VERIFY

VERIFY is started with the command:

```
RUN VERIFY
```

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]?
```

Enter Y or press RETURN if the disk has not been accessed by a host or diagnostic since it was formatted.

This question is asked because certain errors are classed as inconsistencies only when the disk has not undergone bad block replacement after formatting. The next prompt determines whether errors not considered inconsistencies should be reported.

```
VERIFY-Q Print informational (non-warning) messages (Y/N) [N]?
```

Enter N or press RETURN if you want VERIFY to only report inconsistencies—but not information messages.

VERIFY reports the total number of transient errors in its final summary. You can also request that VERIFY display individual blocks with transient errors. If you answered Y to the above prompt, VERIFY next prompts:

```
VERIFY-Q Report transient errors by block (Y/N) [N]?
```

If you enter Y, VERIFY displays a message for each block that contains a transient error. If you enter N or press RETURN, you will not get this report.

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.

A CTRL/Z can be entered 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, notification and reprompt for another unit number are received. If the unit can be accessed, it is acquired and brought on line. VERIFY runs to completion, unless aborted by CTRL/Y or CTRL/C.

7.3.2 Sample Session

The following is a sample session using VERIFY. User input is in **bold print**.

```
CTRL/Y
HSC50> RUN VERIFY

VERIFY-Q Enter unit number to verify (U) [D0]? D133
VERIFY-Q Was this unit just FORMATted (Y/N) [Y]? RETURN
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 information message indicating that. If information messages had been suppressed by responding with N to VERIFY-Q Print information (nonwarning) messages, this information would show only in the summary of the RCT dump.

In the example, VERIFY also printed information messages for the three blocks it found with solid one-symbol correctable ECC errors. If information 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 that although no messages were printed for them, the two revectors in the RCT were verified (as indicated in the summary statistics). Note the odd date for the First 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 VERIFY error messages would also print.

7.3.3 Error 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 Variable Output Error 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, and so forth

7.3.3.2 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:

F = Fatal
I = Information
t = Type: either W or I, depending on the error
W = Warning

7.3.3.3 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 off line 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. This message could be an indication of a problem in the VERIFY utility, but it is more likely that DDUSUB could not obtain resources to complete the I/O request.
- **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.
- **VERIFY-F Too many bad blocks**—VERIFY detected more bad blocks than it can fit into its internal buffers. This may not mean that the disk is bad, but you should not run VERIFY on this disk.

7.3.3.4 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**—When VERIFY searches the RCT for an LBN or RBN corresponding to a known bad PBN, it displays the PBN in brackets and counts it. This message is displayed if the count is greater than zero.
- **VERIFY-W Cannot ONLINE unit**—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**—VERIFY failed to access an LBN space or RBN space track. This error may be caused by a hardware problem.
- **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.
- **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 non-primary revector but was recorded in the RCT as such. When VERIFY reads an LBN with a header indicating it is a non-primary revector, it looks it up in the collected RCT information and flags the fact if it was not found to be so.
- **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 not found to be so.
- **VERIFY-W LBN n. revectors to RBN n. which is bad**—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 is printed.

- **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, or 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 DKUTIL command WRITE/BAD is used).
- **VERIFY-W XBN n. I/O error in access (MSCP Code: o)**—Indicates that an inconsistency was found in the drive or data channel module. VERIFY provides its own error processing for records read where the K detects errors. This message is displayed under two conditions:

  a. If requests do not return with a SUCCESS code, indicating a problem in the drive or disk data channel.

  b. If the return from the I/O operation is not successful after VERIFY reads the record in error one more time with error recovery enabled.
- **VERIFY-W RBN *block* is good but not used for a revector**—VERIFY found a valid RBN (with valid EDC) in the verification pass, but it is not recorded in the RCT as being used. This record should not exist just after FORMAT has been run.

  If you answered YES to the **Was Unit Formatted** prompt, this message is displayed with a severity of W (warning). If the disk has undergone bad block replacement, this message is displayed with a severity of I (informational).—If the drive was just formatted, reformat. If the drive was not recently formatted, no action is necessary. If the message is displayed again, submit an SPR.
- **VERIFY-W RBN *block_no* marked bad in the RCT was not bad**—VERIFY flagged a bad RBN in the RCT that is not bad. This condition should not exist after FORMAT has been run.

  If you answered YES to the **Was Unit Formatted** prompt, this message is displayed with a severity of W (warning). If the disk has undergone bad block replacement, this message is displayed with a severity of I (informational).
- **VERIFY-W LBN *block* has corrupted data (forced error)**—A bad block replacement in which the data could not be recovered produced a revectored LBN with forced error set, indicating the data is probably bad. No such LBNs should exist after FORMAT has been run.

  If you answered YES to the **Was Unit Formatted** prompt, this message is displayed with a severity of W (warning). If the disk has undergone bad block replacement, this message is displayed with a severity of I (informational).
- **VERIFY-W LBN *n* marked primary in RCT, not revectored to its primary**—The specified LBN number that was marked primary in the RCT was not revectored to its primary RBN.
- **VERIFY-W *x*BN *block* has an uncorrectable ECC error**—VERIFY detected an LBN or an RBN with an uncorrectable ECC error that was not marked bad in the RCT. An LBN with an uncorrectable ECC error should be revectored by FORMAT or bad block replacement; an RBN with an uncorrectable ECC error should be marked bad in the RCT. Both of these errors are classed as inconsistencies.

  NOTE
  If VERIFY detects an RBN with an uncorrectable ECC error and it IS marked bad in the RCT, this message is displayed as an informational message.

7.3.3.5 Information Messages

Following are descriptions of the information messages printed by VERIFY. Note that this type of message may or may not need information messages enabled in order to print.

- **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**—A termination message which prints at the end of VERIFY if no inconsistencies were discovered.
- **VERIFY-I There were n inconsistencies found for this drive**—A termination message which prints at the end of VERIFY if inconsistencies were discovered.
- **VERIFY-I Copy n of xCT Block n (XBN n.) is bad**—Prints if information messages are enabled for RCT or FCT blocks that cannot be read correctly with error recovery.
- **VERIFY-I DBN area should probably be reformatted**—Prints whether or not information messages are enabled. If more than nine DBNs were detected with EDC errors (not forced errors), this message prints after the DBN scan.
- **VERIFY-I INITIAL WRITE should be specified for ILEXER**—Prints whether or not information messages are enabled. If more than nine DBNs were detected with EDC errors, this message prints after the DBN scan.
- **VERIFY-I LBN n., a primary has a bad header (is non-primary)**—Prints if information messages are enabled. An LBN is recorded in the RCT as a primary revector, but has a garbled header. Such a condition is abnormal but not an error.
- **VERIFY-I XBN n. has a transient (n out of 6) *error type* error**—Prints if an LBN or RBN has been read six times with a least one error-free read when information 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 information 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: *error type***—Prints for LBNs or RBNs with errors on all six accesses when information 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 is printed along with the following:

  ```
  "NOTE: Table is null or empty (NO BAD PBNs).
  ```

- **VERIFY-I RBN *block* is good but not used for a revector**—VERIFY found a valid RBN (with valid EDC) in the verification pass, but it is not recorded in the RCT as being used. This record should not exist just after FORMAT has been run.
- **VERIFY-I RBN *block_no* marked bad in the RCT was not bad**—VERIFY flagged a bad RBN in the RCT that is not bad. This condition should not exist after FORMAT has been run.
- **VERIFY-I LBN *block* has corrupted data (forced error)**—A bad block replacement in which the data could not be recovered produced a revectored LBN with forced error set, indicating the data is probably bad. This condition should not exist after FORMAT has been run.
- **VERIFY-I *x*BN *block* has an uncorrectable ECC error** — VERIFY detected an RBN with an uncorrectable ECC error that was marked bad in the RCT.

  NOTE
  If VERIFY detects an RBN with an uncorrectable ECC error and it is NOT marked bad in the RCT, this message is displayed with a severity of W (Warning).

- **VERIFY–I *n* Blocks with hard EDC errors**—The number of blocks with hard EDC errors.
- **VERIFY–I *n* LBNs with corrupted data**—The number of revectored LBNs with forced error set. This indicates the data is probably bad.
- **VERIFY–I *n* unused RBNs with good EDC**—The number of RBNs with valid EDCs that are recorded in the RCT as being used.
- **VERIFY–I *n* good RBNs marked bad in the RCT**—The number of RBNs marked bad in the RCT that are actually good.
- **VERIFY–I *n* blocks with solid (non–ECC) errors**—Indicates the number of blocks with solid (not ECC or EDC) errors.
- **VERIFY–I *n* total ECC symbols corrected**—Indicates the total number of ECC symbols corrected during the VERIFY operation.
- **VERIFY–I *n* blocks with *n* symbol ECC errors**—Indicates the number of blocks in which a specified number of ECC symbols were corrected.
- **VERIFY–I *n* blocks with uncorrectable ECC errors**—The number of blocks in which uncorrectable ECC errors were detected.
- **VERIFY–I *n* blocks with transient errors**—The number of blocks in which transient errors were detected.
- **VERIFY–I *n* revectors verified** – Indicates the number of revectors that were correctly verified. Correct verification requires that all blocks that are primary or non-primary revectors are recorded as such in the RCT.
- **VERIFY–I *n* bad RBNs verified** – The number of bad RBNs that were verified as bad in the RCT.
- **VERIFY–I *n* total blocks with any error**—The total number of blocks that contained an error(s) of any kind.
- **VERIFY–I *n* bad DBNs**—The number of bad DBNs VERIFY encountered.
- **VERIFY–I *n* blocks with positioner errors**—The number of blocks in which positioner errors were detected.
- **VERIFY–I *n* blocks with header compare errors**—The number of blocks in which header compare errors were detected.
- **VERIFY–I *n* blocks with EDC errors**—The number of blocks in which EDC errors were detected.
- **VERIFY–I *n* blocks with non–header, non–EDC errors**—The number of blocks with in which non–header, non–EDC errors were detected.
- **VERIFY–I Exiting**—Has completed and is exiting, or you pressed CTRL/C or CTRL/Y.

7.4 FORMAT — Off-line Disk Formatter Utility

FORMAT is the utility used to format disks. It formats with either a 512- or 576-byte sector size. It can be used to format only the read-only DBN space or to format both the LBN area and the read-only DBN space.

CAUTION
The FORMAT utility destroys user data if used by persons not familiar with DSA.

The DBN area is always formatted. If the user requests it, the LBN area also is formatted. When the LBN area is formatted, there are two modes of operation: the reformat and the best guess modes.

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 corrupt, 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 Running FORMAT

FORMAT is started with 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 be formatted or whether only the DBN (diagnostic) area should be formatted. If this prompt is answered with a Y, user data is destroyed.

```
FORMAT-Q Format user data area (Y/N) [N]?
```

If replied with an N or a carriage return only (to obtain the default), the program starts executing and formatting only the DBN area. If a Y is entered, 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 only the carriage return is pressed, the sector size used is 512 bytes. Otherwise, either 512 or 576 should be entered.

```
FORMAT-Q Continue if bad block information is inaccessible (Y/N) [N]?
```

If an N is entered, 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, reformat mode is used if the FCT is valid or a modified best guess mode is used if the FCT is null or corrupt.

If the response to the preceding 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 when 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. The first prompt option is:

```
FORMAT-Q Revector blocks with 1 symbol ECC errors (Y/N) [N]?
```

Normally, blocks discovered during the check pass of formatting with one-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 two-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 that 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 Sample Session

The following is a sample session using FORMAT. User input is in **bold print**.

```
CTRL/Y
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 inaccessible (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!   *
   *                                                         *
   ***********************************************************
```

CAUTION

The message in the BOLD indicates VERIFY must be run to complete verification. This is an essential step and should not be skipped.

The preceding example is the output for an actual session for an RA80 disk with one bad PBN in the FCT. Notice the message that indicates it was retired because it was in the FCT and also the RCT area. Note the information message that is printed every 10 cylinders. This confirms that progress is actually being made and to show at what rate. Also, note the two LBNs that were retired because they had two-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.

7.4.3 Error 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 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 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 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 format error (drive FORMAT command Failed)!**—Prints if a FORMAT command fails for five retries when formatting the DBN 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 remain 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 on line.
- **FORMAT-F LBN format error (drive FORMAT command failed)!**—Prints if a FORMAT command fails for five retries when formatting the LBN area.
- **FORMAT-F 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 that 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 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 Information Messages

Following are the information 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 non-primary or with a header 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 preceding four messages, if the user requested the special option to print bad block position.

- **FORMAT-I CTRL/Y or CTRL/C abort!**—An information 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**—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 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 Success Messages

Following are the FORMAT success messages.

- **FORMAT-S Format begun**—Prints when FORMAT actually begins formatting the disk.
- **FORMAT-S Format completed**—Prints after the format process is done, and all verification tests are complete.

7.5 PATCH — Off-line Load Media Modification Utility

The PATCH utility is designed to modify files on the HSC load media. You can also use it to examine locations within a file without modifying the file.

After you make changes to the load medium, the system version number is incremented on both the system and utility software. Because the version number is included in the checksum, make sure you apply patches in the exact order supplied. All previously shipped patches must be made before you make a new patch. Make sure you supply a checksum to verify the patch has been applied properly.

7.5.1 PATCH Commands

Table 7–1 lists the PATCH commands to enter in response to the PATCH prompts. All commands, except control characters (such as CTRL/C or CTRL/Y), must be terminated with a RETURN. In all instances, the current location refers to the file location of the address (Base + Offset) and the contents displayed under the *Old* column.

Table 7–1 PATCH Commands

Command	Function
RETURN	Closes the current location without modifying it and opens and displays the next consecutive location.

Table 7–1 (Cont.) PATCH Commands

Command	Function
n	Sets the current location to a value of n, closes it, and opens and displays the next location.
^	Closes the current location without modifying it and opens and displays the previous location.
n^	Sets the current location to n, closes it, and opens and displays the previous location.
;B	Backs up to the previous prompt. If requesting new value, backs up to Offset. If at Offset, backs up to Base. If at Base, prompts for the checksum.
;C	Restores the value originally in the open location. Used when an incorrect location is modified.
;E	Exits to the checksum prompt. If entered at the checksum question, causes PATCH to exit without making any modifications to the file.
;Q!n ;Q!n^	Performs a logical inclusive OR of the contents of the open location with n, and opens and displays the contents of the next (or previous) location.
;Q&n ;Q&n^	Performs a logical AND of the contents of the open location with n, and opens and displays the contents of the next (or previous) location.
;V	Displays all changes made to the file during the current session.
CTRL/C CTRL/Y	Exits PATCH without modifying the file.

7.5.2 Running PATCH

The following section describes the steps in the PATCH operation. Each PATCH prompt is described, as well as the input you must supply.

Before running PATCH, take all disks and tapes off line that are attached the HSC to be patched. Take the HSC off line, or fail over to an alternate HSC in the cluster.

To initiate PATCH, enter the following command at the HSC> prompt:

```
HSC> RUN PATCH
```

PATCH prompts for the device and name of the file to be patched.

```
HSC>PATCH-Q RX33 Unit and filename (dev:file.ext) [SY:]?dev:file.ext
```

Enter the file and device name. To use the default system drive, specify only the filename. Refer to the following table for special considerations.

If	Then
The file being modified with PATCH is on the device containing the system or utility software	Install system and utility software of the same revision level.
The system version numbers do not match	PATCH displays an error message, releases all acquired resources, and terminates.
The file or device name is invalid	PATCH displays an error message and prompts for another file and device name.

If	Then
The filename extension is .SAV	The load device unit is acquired for exclusive access and all changes by other programs are locked out.
The filename extension is not .SAV	The SYSCOM.INI file on the system medium and the 2NDTAP.VER on the utility medium are accessed to compare the system version numbers.

PATCH locates the file and checks the version numbers if necessary.

PATCH prompts for the base address of the patch.

```
Base    (O)  [000034]? RETURN
```

If the base address is odd, PATCH displays all values as bytes, rather an word values, unless no file was specified.

PATCH prompts for the offset address of the patch.

```
Offset (O)  [000000]? (100)
```

The default offset address is zero. If the offset address is odd, PATCH displays all values as bytes, rather an word values, unless no file was specified.

PATCH prompts for the new contents of the file location.

```
Base     Offset    Old      New?
aaaaaa   bbbbbb    cccccc   ?
```

Enter the new contents of the file location in the **NEW** field. Each field in the prompt is described in the following table:

Field	Meaning
aaaaaa	Base address as supplied
bbbbbb	Current offset
cccccc	Current contents of file location (sum of base and offset)
?	New contents of the file location, as supplied in the patch documentation

Enter RETURN after typing in the new location contents. PATCH automatically increments the offset to the next address, displays the current contents of that location, and prompts for a new location contents. Use ;B to enter new address/offset values as directed by the patch documentation, and continue installing the patch data until completed.

Enter ;V to list the patch locations for verification. Check the changes carefully, and when you are satisfied that all data has been entered correctly, exit the data entry prompt by typing ;E. If you discover a mistake, enter CTRL/C to abort the program without the patch, and start over.

After entering ;E, PATCH next prompts for the checksum.

```
Enter checksum?
```

Enter the checksum given in the patch documentation. If you enter the wrong value, patch gives you an error message and reprompts for the checksum. PATCH does not make any changes to the file until you have entered the checksum.

PATCH then proceeds with the patch operation and displays a message indicating whether the patch was successful and what patches were made.

After installing the patch, reboot the HSC and verify that the system version number is incremented on both the system and utility software. COPY the patched media to any backup media as required and return the system to service.

7.5.3 Sample Session

Example 7–1 is a sample PATCH session. The checksum shown is an arbitrary value. *This example is for display only.*

```
HSC>RUN PATCH
PATCH-Q Unit and filename (dev:file.ext) [SY:]? PATCH.UTL
Base (O)    [000034]?12000 ❶
Offset (O)  [000000]? RET ❷
   Base     Offset    Old      New?
  012000    000000    004767   ?137 ❸
  012000    000002    010524   ?15000 ❹
  012000    000004    012700   ?;B ❺
Offset (O)  [000004]?;B ❻
Base (O)    [012000]?15000 ❼
Offset (O)  [000004]?0 ❽
   Base     Offset    Old      New?
  015000    000000    000000   ?12701 ❾
  015000    000002    000000   ?177560
  015000    000004    000000   ?137
  015000    000006    000000   ?1004
  015000    000010    000000   ?;E ❿
Enter checksum?12345 ⓫
PATCH-S Wait...
PATCH-S 6 changes made
```

Example 7–1 Example Patch of a File

❶ Base is *12000*

❷ Offset is *0* (default)

❸ Replace *4767* with *137*

❹ Replace *10524* with *15000*

❺ Back up to new Offset

❻ Back up to new Base

❼ New Base of *15000*

❽ New Offset of *0*

❾ Insert new values...

❿ Exit command

⓫ Enter checksum of *12345*

7.5.4 Error and Information Messages

This section describes the messages that may be displayed during a PATCH operation. The most common causes of the errors and the correct actions are included.

7.5.4.1 Fatal Error Messages

The following fatal error messages are issued by PATCH:

- **PATCH–F Cannot access PATCH data file** – PATCH cannot access the file you specified to be patched.
- **PATCH–F Cannot Access Version On Off-line Diagnostic medium**—PATCH could not access the file OFLLDR.SAV on the off-line diagnostic utility medium.
- **PATCH–F Cannot Access Version On System medium** – PATCH could not access the file SYSCOM.INI on the system medium. If this message displays after the *Wait ...* message is displayed, a serious bug or problem exists with the load medium. If the message displays just after the filename has been entered, the system medium is probably not mounted.
- **PATCH–F Cannot Access Version On Utility medium**—PATCH could not access the file 2NDTAP.VER on the utility medium. If this message displays after the *Wait ...* message is displayed, a serious bug or problem exists with the load device. If this message displays just after the filename has been entered, the utility medium is probably not mounted.
- **PATCH–F File Not Found**—The user specified a nonexistent file. PATCH exits cleanly.
- **PATCH–F Insufficient Resources To Run**—The program could not acquire the resources to run. Sufficient common pool memory is not available to allocate the necessary structures.
- **PATCH–F Read Failure: *block–number***—When PATCH attempted to read the specified block of the file, a media error occurred that cannot be recovered. PATCH exits cleanly.
- **PATCH–F You cannot PATCH this file**—The filename you specified cannot be patched.
- **PATCH–F Unit(s) write-protected: update was not done**—The disk unit on which the load medium to be patched resides is write-protected.
- **PATCH–F Version On System Medium Does Not Match Utility Medium**— The version numbers on the system and utility media do not match, indicating media from two different revision levels are being used.
- **PATCH–F Write Failure: *block–number***—When PATCH attempted to update the file with the requested changes, a media error that cannot be recovered occurred in the specified file block. This is the most serious of errors because both file and medium integrity are questionable. The usual recovery is to restore the medium with the backup copy made before starting the patch.
- **PATCH–F Write failure during write check, status: *block number***—PATCH verifies that a file can be written before it actually writes to the file. You will get this message if the the file cannot be written for some reason other than the file being write-protected or a recoverable software error.

7.5.4.2 PATCH Error Messages

The following error messages are issued by PATCH:

- **PATCH–E Incorrect Checksum**—You entered a checksum different from the checksum PATCH calculated. PATCH repeats the checksum question.
- **PATCH–E Invalid Command**—You entered information out of context.
- **PATCH–E Invalid Device Name Or Switch**—You entered an invalid device name (not the load medium) or the syntax of the filename is incorrect.

7.5.4.3 Warning Messages

- **PATCH–W Buffer Space Exhausted**—You entered a patch when the internal buffers were full.
- **PATCH–W Nonfile Structured Mode Assumed** – You entered a null filename. This message is a reminder that all patches will now be applied to the load medium as a whole instead of to a specific file.

7.5.4.4 Informational Messages
(O))

- **PATCH–I Checksum = *octal–checksum* (O)**—You entered the **;X** command, requesting that a checksum prompt not be displayed.
- **PATCH–I CTRL/Y Or CTRL/C Abort**—You typed a CTRL/Y or CTRL/C. No changes are made to the file unless the abort occurs during the update, after the *Wait...* message has been displayed.
- **PATCH–I No patches recorded**—In a normal session, PATCH informs you of how many patches have been made. You will get this message if PATCH records no patches in the file PATCH.DAT, the file used to keep track of the patches made.
- **PATCH–I Patches made:**—PATCH completes and displays the patches made following the colon (:).

7.5.4.5 Success Messages

- **PATCH–S *patch–count* Changes Made**—You exited PATCH. The value of *patch–count* is the number of changes made to the file. Note that if you modify and then restore a location, it is not considered a change. The file is only rewritten when *patch–count* is not a zero.
- **PATCH–S Wait...**—PATCH has started writing changes to the file.

8
Troubleshooting Techniques

8.1 Introduction

This chapter describes the types of errors occurring during HSC boot and operation. The major divisions are initialization errors and system-type errors. Initialization errors occur while the HSC is trying to boot. System-type errors occur while the HSC 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 HSC 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 displayed by 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 on the console are listed in alphabetical order in Section 8.5 and are listed in this manual's index. Section 8.3 divides initialization errors into three types:

- OCP fault codes
- Module LEDs
- Boot diagnostic messages

HSC console error message descriptions for system-type errors are described in this chapter and are organized into the following sections:

- MSCP/TMSCP errors, Section 8.4.1
  - Controller errors, Section 8.4.2.5
  - MSCP SDI errors, Section 8.4.2.6
  - Disk transfer errors, Section 8.4.2.7
- BBR errors, Section 8.4.3
- 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
- Out-of-band errors, Section 8.4.5

8.3 Initialization Error Indications

Initialization errors are indicated by:

- OCP fault code displays
- Module LEDs
- 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 HSC 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.

DESCRIPTION	HEX	OCT	BINARY	OCP INDICATORS				
				INIT	FAULT	ONLINE		
K.PLI ERROR ***	01	01	00001	OFF	OFF	OFF	OFF	ON
K.SDI/K.SI INCORRECT VERSION OF MICROCODE ***	02	02	00010	OFF	OFF	OFF	ON	OFF
K.STI/K.SI INCORRECT VERSION OF MICROCODE ***	03	03	00011	OFF	OFF	OFF	ON	ON
P.IOJ CACHE FAILURE *	08	10	01000	OFF	ON	OFF	OFF	OFF
K.CI FAILURE *	09	11	01001	OFF	ON	OFF	OFF	ON
DATA CHANNEL MODULE ERROR *	0A	12	01010	OFF	ON	OFF	ON	OFF
P.IOJ/C MODULE FAILURE	11	21	10001	ON	OFF	OFF	OFF	ON
M.STD2 MODULE FAILURE *****	12	22	10010	ON	OFF	OFF	ON	OFF
BOOT DEVICE FAILURE **	13	23	10011	ON	OFF	OFF	ON	ON
PORT LINK NODE ADDRESS SWITCHES OUT OF RANGE	15	25	10101	ON	OFF	ON	OFF	ON
MISSING FILES REQUIRED ****	16	26	10110	ON	OFF	ON	ON	OFF
NO WORKING K.CI, K.SDI, K.STI, OR K.SI IN SUBSYSTEM	18	30	11000	ON	ON	OFF	OFF	OFF
INITIALIZATION FAILURE	19	31	11001	ON	ON	OFF	OFF	ON
SOFTWARE INCONSISTENCY	1A	32	11010	ON	ON	OFF	ON	OFF
ILLEGAL CONFIGURATION	1B	33	11011	ON	ON	OFF	ON	ON

* THESE ARE THE SO-CALLED SOFT OR NONFATAL ERRORS.
** POSSIBLE MEMORY MODULE/CONTROLLER ON HSC70.
*** INCORRECT VERSION OF MICROCODE.
**** THIS FAULT CODE WILL ALSO BE DISPLAYED IF THE L0105 MODULE IS NOT AT THE MINIMUM REV LEVEL.
*****SWAP MEMORY MODULE FIRST. IF PROBLEM PERSISTS, TRY THE P.IO MODULE.

CXO-905D

Figure 8–1 Operator Control Panel Fault Codes

8.3.1.1 Fault Code Interpretation

All failures occurring during the Init P.io test are reported on the OCP 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 HSC 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 pressing the Fault switch a second time. To restart the boot procedure, press the Init switch. To identify the probable failing module, see Figure 8–1.

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

- **Fault Code 1, K.pli error**

CODE VALUE			OCP INDICATORS				
HEX	OCT	BINARY	INIT	FAULT	ONLINE		
01	01	00001	OFF	OFF	OFF	OFF	ON

CXO-2666A

Figure 8–2 OCP Fault Code 1

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 K.pli.

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 off-line 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.

- **Fault Code 2, K.sdi/K.si incorrect version of microcode**

CODE VALUE			OCP INDICATORS				
HEX	OCT	BINARY	INIT	FAULT	ONLINE		
02	02	00010	OFF	OFF	OFF	ON	OFF

CXO-2667A

Figure 8–3 OCP Fault Code 2

All K.sdi/K.si modules are initialized during the Disk Server functional code initialization. If a K.sdi/K.si passes initialization, the Disk Server initialization code checks the K.sdi/K.si 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 K.sdi/k.si.

- **Fault Code 3, K.sti/K.si incorrect version of microcode**

CODE VALUE			OCP INDICATORS				
HEX	OCT	BINARY	INIT	FAULT	ONLINE		
03	03	00011	OFF	OFF	OFF	ON	ON

CXO-2668A

Figure 8–4 OCP Fault Code 3

Indicates tape data channel microcode is incompatible.

- **Fault Codes 10, 11, and 12, Soft errors**

  These are the 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 HSC 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**

CODE VALUE			OCP INDICATORS				
HEX	OCT	BINARY	INIT	FAULT	ONLINE		
08	10	01000	OFF	ON	OFF	OFF	OFF

CXO-2669A

Figure 8–5 OCP Fault Code 10

Results in disabling the cache and displaying this soft fault code for any failure detected in the J-11 instruction cache during HSC subsystem initialization while the HSC continues operation. Replace the P.ioj module (L0111/L0111-YA) and reboot.

— **Fault Code 11, K.ci failure**

CODE VALUE			OCP INDICATORS				
HEX	OCT	BINARY	INIT	FAULT	ONLINE		
09	11	01001	OFF	ON	OFF	OFF	ON

CXO-2670A

Figure 8–6 OCP Fault Code 11

One or more modules of the K.ci set 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 LINK module (L0100/L0118).

— **Fault Code 12, Data channel module failure**

CODE VALUE			OCP INDICATORS				
HEX	OCT	BINARY	INIT	FAULT	ONLINE		
0A	12	01010	OFF	ON	OFF	ON	OFF

CXO-2671A

Figure 8–7 OCP Fault Code 12

Reports 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 for a K.sdi (L0108-YA) or 203 for a K.sti (L0108-YB). The K.si (L0119-YA) module will answer with 002 if it is initiated as a disk data channel, or with 203 if it is initiated as a tape data channel. The data channel with the red LED on is the failing module.

- **Fault Code 21, P.ioj/c module failure**

CODE VALUE			OCP INDICATORS				
HEX	OCT	BINARY	INIT	FAULT	ONLINE		
11	21	10001	ON	OFF	OFF	OFF	ON

CXO-2672A

Figure 8–8 OCP Fault Code 21

Indicates the P.ioj/c module is the most probable cause of the failure detected by the Init P.io test. If possible, run the off-line P.io test for a more definitive report on the error. Otherwise, replace the P.ioj/c module and run the Init P.io test again. If the test still fails, run the off-line P.io test to help further isolate the failure.

- **Fault Code 22, M.std2 module failure**

CODE VALUE			OCP INDICATORS				
HEX	OCT	BINARY	INIT	FAULT	ONLINE		
12	22	10010	ON	OFF	OFF	ON	OFF

CXO-2673A

Figure 8–9 OCP Fault Code 22

Indicates the M.std2 module (L0117) 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/c 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 17772340, 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 off-line 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.

- **Fault Code 23, Boot device failure**

CODE VALUE			OCP INDICATORS				
HEX	OCT	BINARY	INIT	FAULT	ONLINE		
13	23	10011	ON	OFF	OFF	ON	ON

CXO-2674A

Figure 8–10 OCP Fault Code 23

Indicates a problem with an HSC boot device, the system media, the boot device controller, or the read/write logic on the memory module. This fault can be any of the following, in order of probability:

— A failure in the P.ioc.

— A failure in the read/write logic of the M.std2 module. Replace M.std2 (L0117).

— A faulty boot device controller/drive interface cable. Replace the cable.

— Diskettes or tapes not installed in the drives.

— Doors left open on the RX33 drives

— Tape improperly inserted in the TU58 drive.

— No bootable image in the system device media.

Ensure a known good HSC bootable media is properly loaded in the system boot device. If checking the obvious (doors, diskettes, or tapes) does not remedy the situation, refer to Chapter 6 for more information before beginning repair. Running the off-line P.io and off-line RX33 or TU58 tests (if possible) is strongly recommended before modules are replaced. These tests may help further isolate or define the problem.

- **Fault Code 25, port link node address switches out of range**

CODE VALUE			OCP INDICATORS				
HEX	OCT	BINARY	INIT	FAULT	ONLINE		
15	25	10101	ON	OFF	ON	OFF	ON

CXO-2675A

Figure 8–11 OCP Fault Code 25

Indicates the LINK (L0100/L0118) module node address switches are set to a value outside the currently suggested range of 32 decimal (HSC software V3.90).

- **Fault Code 26, missing files required**

CODE VALUE			OCP INDICATORS				
HEX	OCT	BINARY	INIT	FAULT	ONLINE		
16	26	10110	ON	OFF	ON	ON	OFF

CXO-2676A

Figure 8–12 OCP Fault Code 26

Indicates the system diskette does not contain one of the files necessary for operation of the HSC control program. This failure should occur only if one of the required files is inadvertently deleted from the HSC system media.

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

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

— SYSCOM.INI is not present on the load device.

— EXEC.INI is not present on the load device.

— A version mismatch exists between either EXEC, SUBLIB, or SYSCOM and OLBVSN (Object Library Version Number).

If the State light was blinking before the fault, it can mean:

— Any of the the normally loaded programs (SINI, CERF, DEMON, etc.) are not present on the load device.

— A version mismatch exists on any one of the normally-loaded programs.

Replace the system media with a backup copy.

- **Fault Code 30, No working K.ci, K.sdi, K.sti, or K.si in subsystem**

CODE VALUE			OCP INDICATORS				
HEX	OCT	BINARY	INIT	FAULT	ONLINE		
18	30	11000	ON	ON	OFF	OFF	OFF

CXO-2677A

Figure 8–13 OCP Fault Code 30

Indicates the HSC does not contain any working K.ci, K.sti, K.sdi, or K.si modules. Either none is installed in the HSC, or all of those 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 HSC Off-line Diagnostic media into the appropriate system drive and reboot the HSC. When the off-line loader prompts with ODL>, type SIZE. 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/L0111-YA) or P.ioc (L0105) and check subsystem power for proper operation.

- **OCP error code of 31, Initialization failure**

CODE VALUE			OCP INDICATORS				
HEX	OCT	BINARY	INIT	FAULT	ONLINE		
19	31	11001	ON	ON	OFF	OFF	ON

CXO-2678A

Figure 8–14 OCP Fault Code 31

Indicates a crash occurred while the HSC was attempting to load and initialize its control program.

Use micro-ODT to diagnose these initialization crashes, as follows:

1. Press the break key on the local console terminal.
2. Type 17 777 656/.

   This is the address of the UPAR7 register. The reasons 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 8–1 shows the addresses of the UPAR registers.

Table 8–1 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

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

Table 8–2 Control Program Bits

16 Bit Message	Meaning	FRUs
X XXX XXX 1XX XXX XXX	NXM	P.ioj (L0111/L0111-YA) M.std2 (L0117) Software
X XXX XX1 0XX XXX XXX	Illegal inst.	P.ioj (L0111/L0111-YA) M.std2 (L0117) Software

Table 8–2 (Cont.) Control Program Bits

16 Bit Message	Meaning	FRUs
X XXX XX1 1XX XXX XXX	Parity trap	M.std2 (L0117) P.ioj (L0111/L0111-YA)
X XXX X10 0XX XXX XXX	Level 7 interrupt	K.xx (L0108/L0119-YA) K.pli (L0107)
X XXX X10 1XX XXX XXX	MMU trap	P.ioj (L0111-L0111-YA) Software
X XXX X11 0XX XXX XXX	Software crash	Software
X XXX X11 1XX XXX XXX	K.ci host reset	M.std2 (L0117)
X XXX 100 0XX XXX XXX	User requested reboot	N/A

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).

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**: Replace the P.ioj/P.ioc module.

— **Parity trap**: Use the same method for parity traps as for NXM traps to determine the failing address.

— **Level 7 interrupt**: Determine which K UPAR4. Refer to Table 8–1 for the address of each UPAR register. Each byte of each register contains module status for each requester (K) in the HSC. Refer to Appendix C to determine a failing status code. Refer to Table 8–3 for the designation of requestors to UPAR registers for a level 7 interrupt.

Table 8–3 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

— **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 an MMU trap occurs, status of the MMU is found in these registers.

— **Software crash**: Try using another copy of the boot media. If the problem is not corrected, replace the P.ioj/P.ioc module. If the problem still persists, replace the memory module.

- **K.ci host reset**: Press 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 pressed, the contents of control window 0 are displayed. Enter a 0 into this location followed by a carriage return. Then type 17 760 002/. This is the second location in Control memory. The number displayed as the contents of 17 600 002 is the number of the host that issued the HOST RESET command.

- **OCP error code 32, Software inconsistency**

CODE VALUE			OCP INDICATORS				
HEX	OCT	BINARY	INIT	FAULT	ONLINE		
1A	32	11010	ON	ON	OFF	ON	OFF

CXO-2679A

Figure 8–15 OCP Fault Code 32

Indicates an inconsistency in the software. Reboot the HSC. If this failure persists, use a backup copy of the system media. If the failure still persists, use the Off-line diagnostics to help isolate any hardware failures in the subsystem. Also, try using an earlier version of the HSC operating software.

- **OCP error code 33, Illegal configuration**

CODE VALUE			OCP INDICATORS				
HEX	OCT	BINARY	INIT	FAULT	ONLINE		
1B	33	11011	ON	ON	OFF	ON	ON

CXO-2906A

Figure 8–16 OCP Fault Code 33

Indicates that you have installed an illegal configuration of modules in the HSC40 backplane. Check the configuration of modules in the backplane and install the modules according to the following rules:

1. Install any combination of K.si, K.sdi, or K.sti data channel modules in requestors 2 through 4 (backplane slots 8 through 10).
2. Leave requestors 5 through 9 (backplane slots 3 through 7) unoccupied.

8.3.2 Module LEDs

HSC modules contain LEDs used as State indicators for each module. Descriptions of these LEDs follow in the next sections. Also, refer to Chapter 2 for the locations of the module LEDs.

8.3.2.1 P.ioj/c LEDs

Table 8–4 shows the P.ioj/c (L0111, L0111-YA, or L0105) LEDs and their functions.

Table 8–4 P.ioj/c LEDs

LED	Color	Meaning
D1	Yellow	Micro-ODT—Used during J-11 power-up microdiagnostics. ON when J-11 is executing micro-ODT.
D2	Yellow	Terminal Port OK—Used during J-11 power-up microdiagnostics. Serial Line Unit (SLU) output of UART.
D3	Yellow	Memory OK—Used during J-11 power-up microdiagnostics. Turned OFF as J-11 successfully accesses Program memory.
D4	Yellow	Sequencing indicator—Used during J-11 power-up microdiagnostics. Turned OFF as J-11 verifies proper functioning of its sequencers for control store.
D5	Yellow	State indicator—Mirrors the OCP State indicator (under software control).
D6	Yellow	Run indicator—Pulses at the on-board microprocessor run rate. Blinks once for every PDP-11 instruction fetched (J-11 run LED).
D7	Red	Board status—Indicates an inoperable module except during initialization when it comes on during module testing.
D8	Green	Board status—Indicates the module has passed all applicable 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 8–4. First, LED numbers D8 and D7 are used to indicate whether the P.ioj/c 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 8–5 shows the M.std2 (L0117) and M.std (L0106) module LEDs and their functions. These LEDs are controlled by a bit in the system boot device FDC MAR02 register. The green LED is set to ON by the P.ioj/c boot/ROM self-test diagnostics after the system boot device has passed its self-tests, and Program memory has found 8-Kwords to load INIPIO/OFLPIO.

Table 8–5 M.std2 and M.std LEDs

LED	Color	Meaning
D2	Red	Module not OK
D2	Green	Module OK
D2	Yellow	Memory active

NOTE
The entire LED package on the M.std2 or M.std is called D2. All three LEDs are contained in the D2 package.

8.3.2.4 Data Channel LEDs

Table 8–6 shows the K.sdi/K.sti (L0108-YA/YB) and K.si (L0119-YA) data channel module LEDs and their functions with the system software.

Table 8–6 K.sdi/K.sti and K.si LEDs

LED	Color	Meaning
	Red	Module failure—Indicates a module microdiagnostic failed to successfully complete, or this module is still under initialization by the subsystem.
	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.
LED pack (K.si only)	Amber (eight LEDs)	D1—OFF for PROM load, ON for RAM load. D2 through D8—Upper register #2 contents. The LEDs reflect the implemented bits of the upper error register #2. When a microinstruction parity error is detected, the module clocks are inhibited, stopping the module. The bit content of the upper error register #2 is displayed on the LEDs.

8.3.2.5 Host Interface LED

Table 8–7 shows the three modules in the K.ci set, their LEDs, and the functions of the LEDs with the system software.

Table 8–7 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.
PILA	D3	Yellow	Not found on all module revisions. ON when K.pli is asserting Init. When Init is true, both the red and the green PILA LEDs are forced OFF.

Table 8–7 (Cont.) K.ci (LINK, PILA, K.pli) LEDs

Module	LED	Color	Meaning
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 HSC to complete its initialization and not report the fact on the local console terminal. This is an indication of a failure in the serial communication path between the UART chip on the P.ioj/c and the local console terminal.

As a method of testing this serial path, the HSC 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 that the series of characters is 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 to P.ioc/j baud-rate mismatch (default is 9600), a P.ioj/c module failure, or a problem within the terminal-cabling subsystem. Ensure the terminal setup parameters are correct. Refer to the *HSC Installation Manual* for the proper terminal configuration, the *VTxxx Owner's Manual* for problem-solving techniques related to the VTxxx, and the *DECwriter Correspondent Technical Manual* for problem-solving techniques related to the LA12.

8.3.4 Requestor Status for Nonfailing Requestors

When a requestor successfully completes all internal microdiagnostics, the requestor status (Status=) contains 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/K.si with the disk channel microcode loaded).
- Code 004 represents a K.si with no microcode loaded.
- Code 203 represents a properly functioning tape data channel module (K.sti/K.si with the tape channel microcode loaded).
- 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, K.sti, and K.si detected failures.

8.3.5 HSC Boot Flow and Troubleshooting Chart

The HSC boot flow and troubleshooting chart calls out useful visual milestones that aid in troubleshooting problems which can occur during initialization.

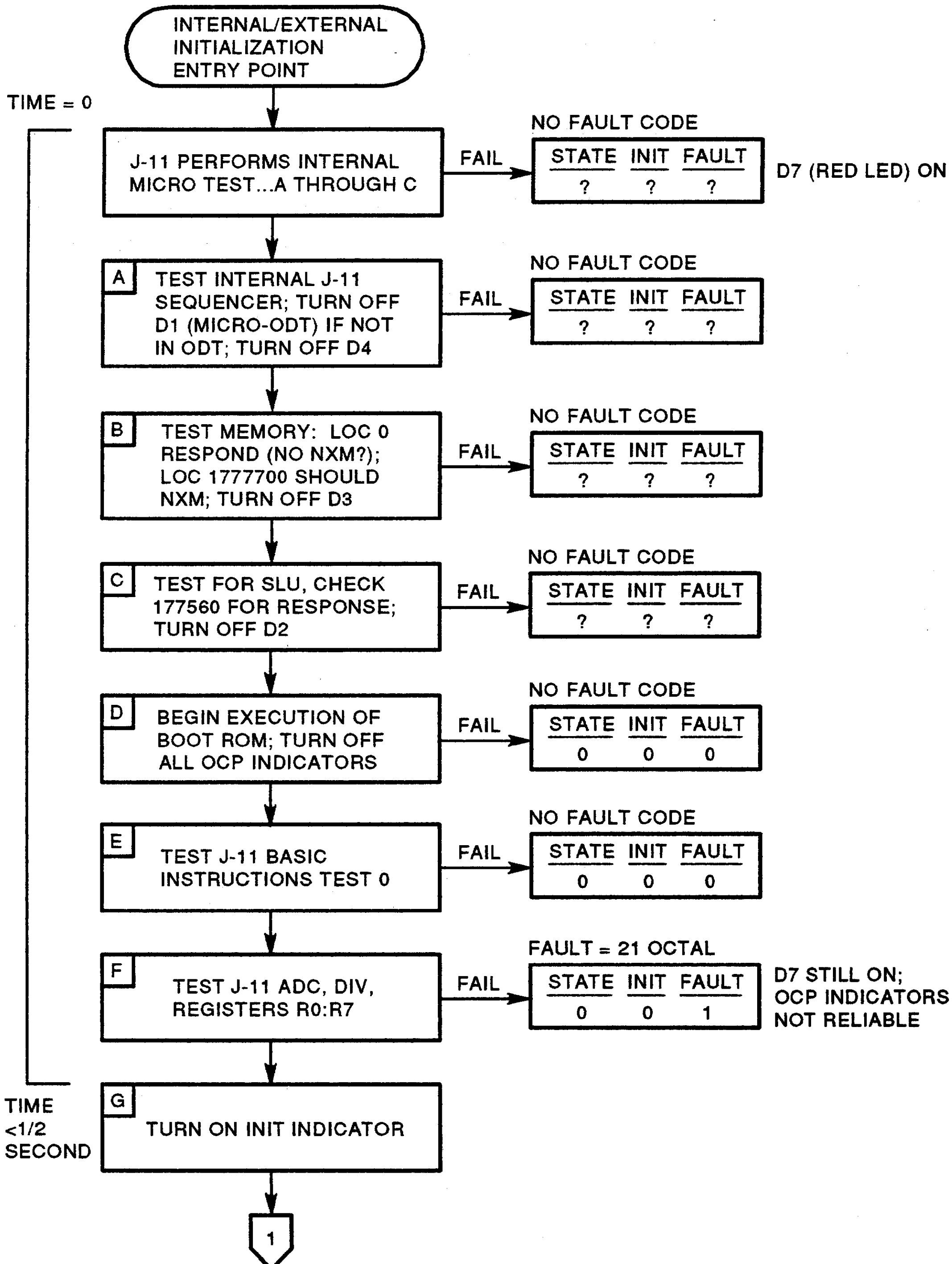
The flowchart has three main divisions:

1. Information on activity common to both the system and off-line 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 off-line diskette is contained in boxes OA through OG

The flowchart begins when one of the following occurs:

- Init button is pushed.
- Powerup has started.
- Other software caused reboot.

Figure 8–17 maps the entire HSC boot sequence.



NOTES:
1. LEDs D1-D4 AND D7 ARE ON THE P.IOJ MODULE.
2. ? MEANS OCP LEDs ARE INDETERMINATE AND HAVE NO MEANING AT THIS TIME.

CXO-945C
Sheet 1 of 5

Figure 8–17 (Cont.) HSC Boot Flow and Troubleshooting Chart

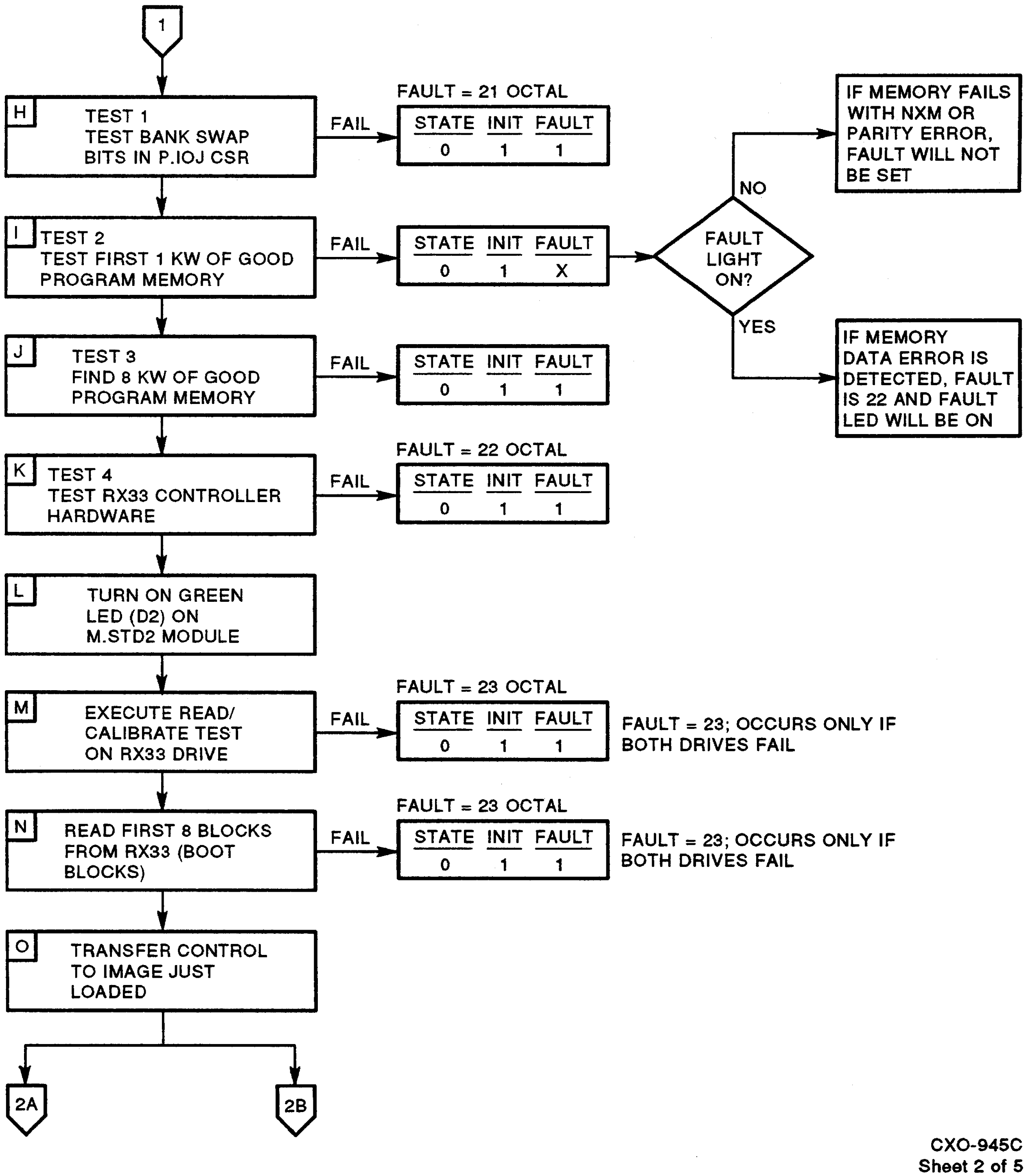


Figure 8–17 (Cont.) HSC Boot Flow and Troubleshooting Chart

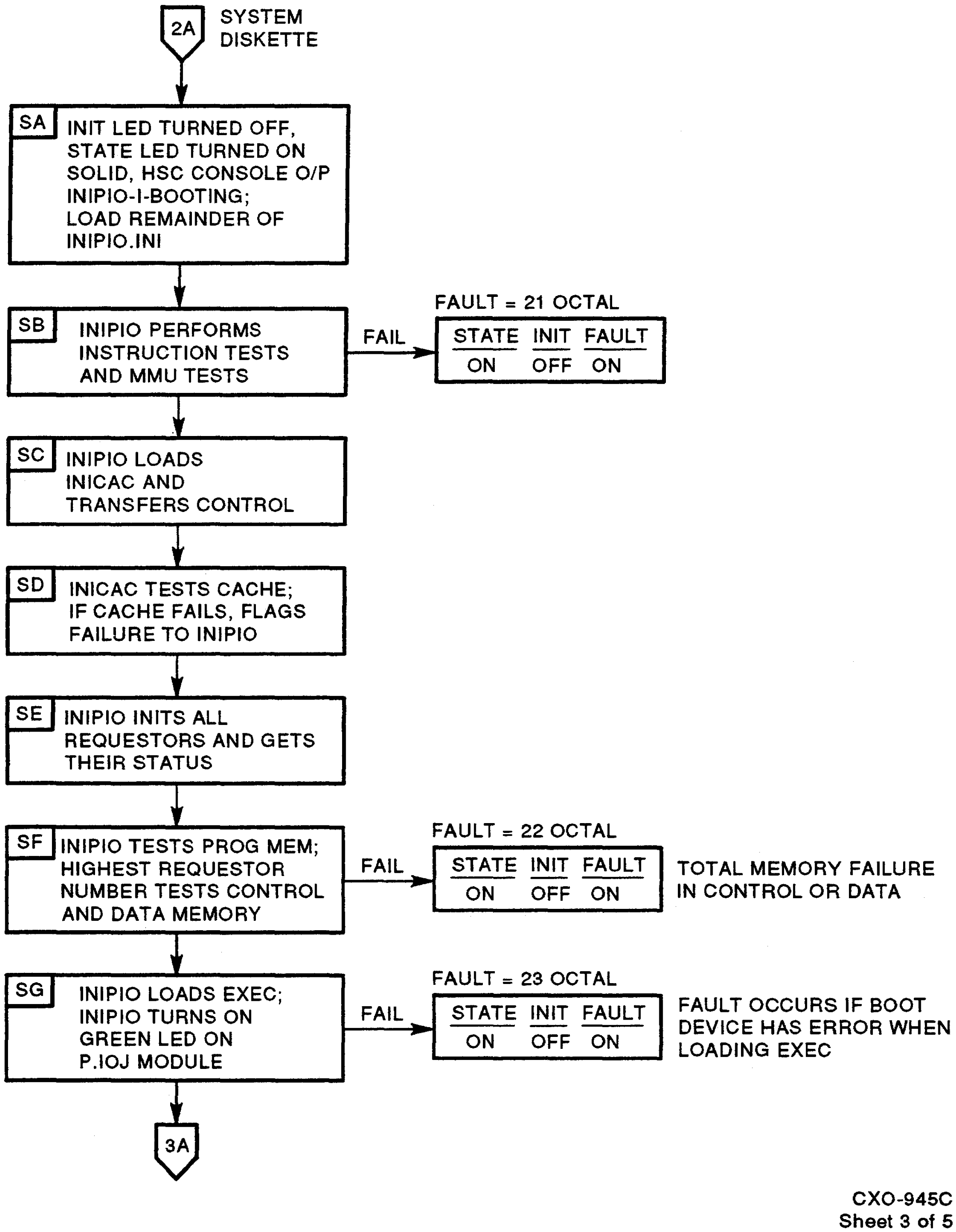


Figure 8–17 (Cont.) HSC Boot Flow and Troubleshooting Chart

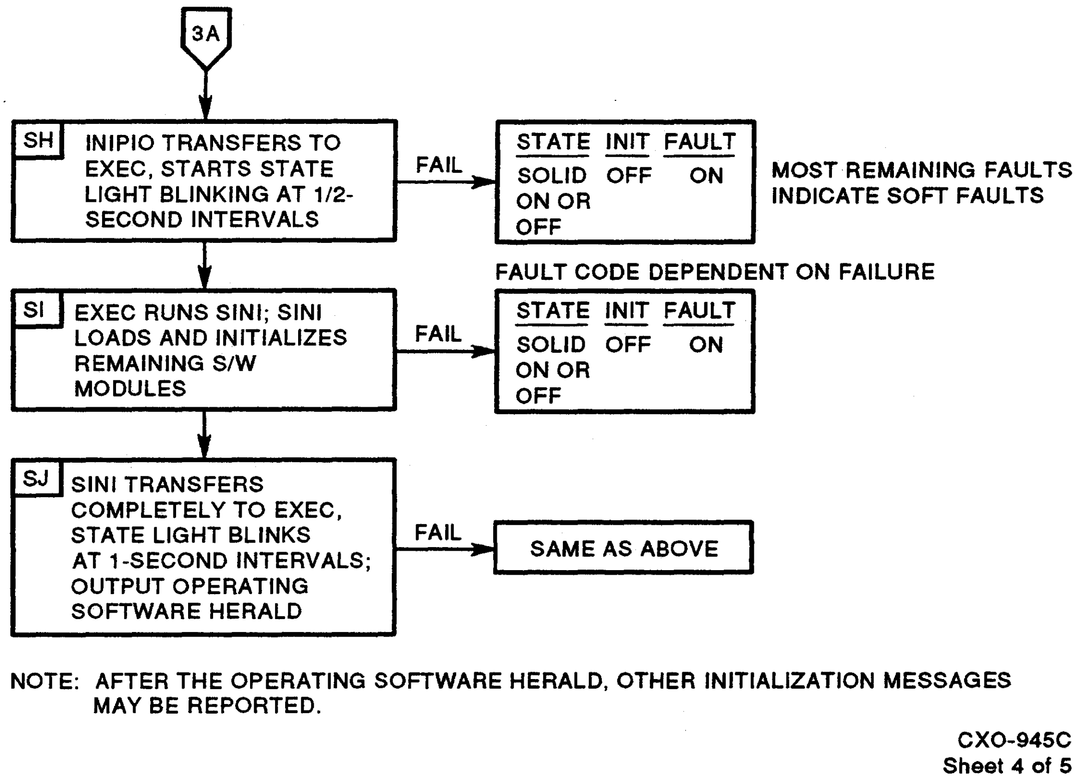
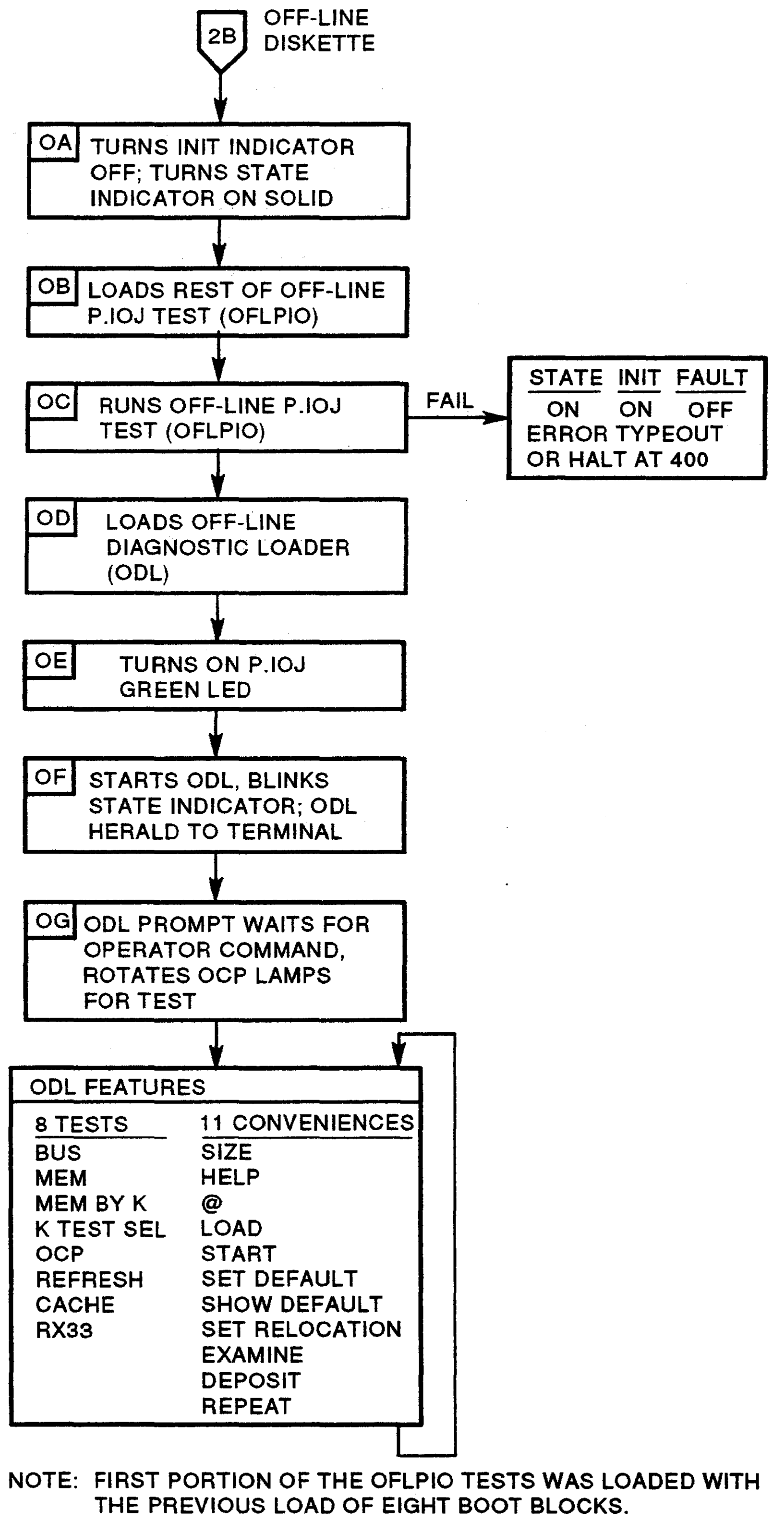


Figure 8–17 (Cont.) HSC Boot Flow and Troubleshooting Chart



CXO-945C
Sheet 5 of 5

Figure 8–17 HSC Boot Flow and Troubleshooting Chart

8.3.6 HSC50 Flow and Troubleshooting Chart

The HSC50 boot flow and troubleshooting chart calls out useful visual milestones that aid in troubleshooting the problems which can occur during initialization.

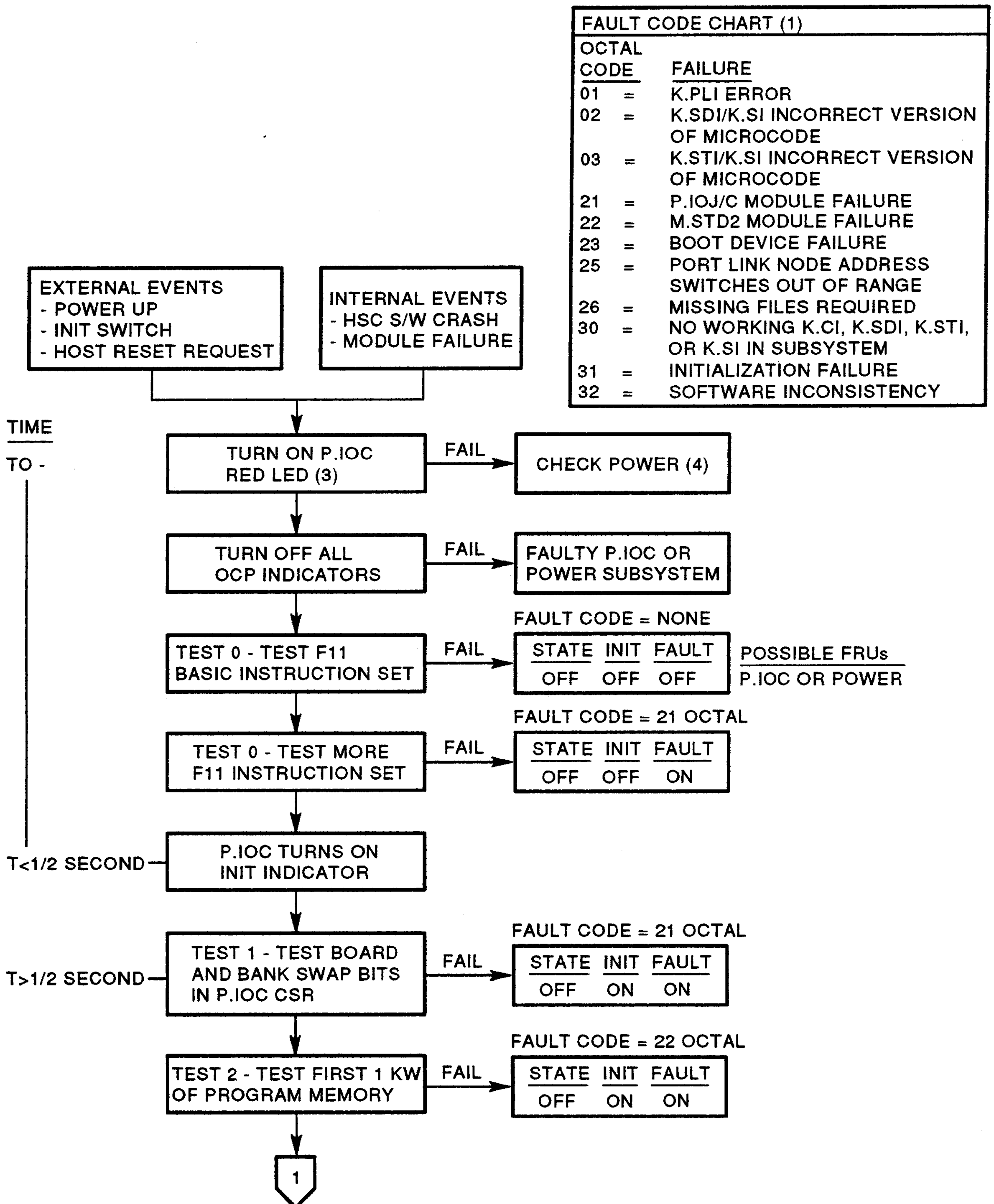
The flowchart has three main divisions:

1. Information on activity common to both the system and off-line media.
2. Information on activity specific to the system media.
3. Information on activity specific to the off-line media.

The flowchart begins when one of the following occurs:

- Init button is pushed.
- Powerup has started.
- Other software caused reboot.

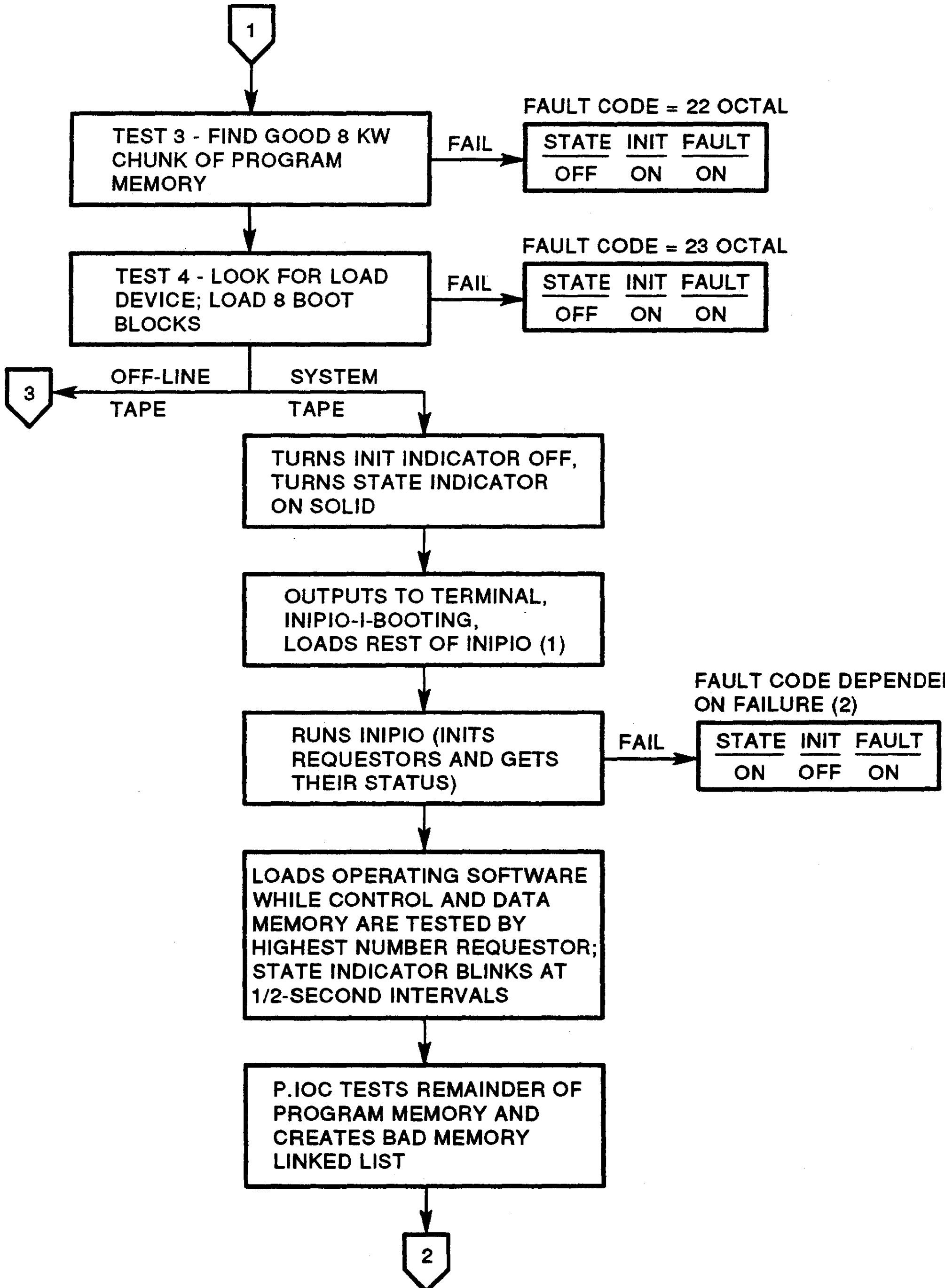
Figure 8–18 maps the entire HSC50 boot sequence.



NOTES:
1. Ks (REQUESTERS) = K.SDI, K.STI, K.PLI.
2. ALL MODULE RED INDICATORS ON EXCEPT MEMORY (NO RED LED ON IT).
3. REFER TO POWER SUPPLY TROUBLESHOOTING FLOWCHART.

CXO-052C
Sheet 1 of 4

Figure 8–18 (Cont.) HSC50 Boot Flow and Troubleshooting Chart

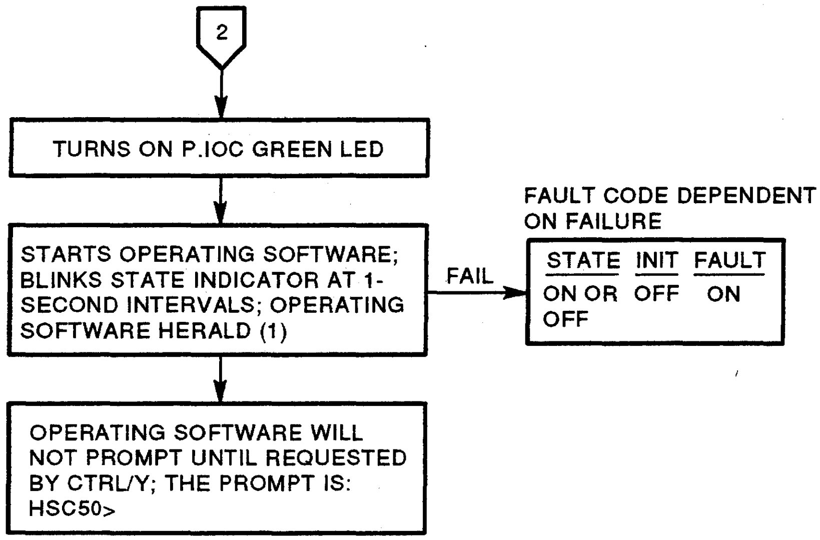


NOTES:

1. FIRST PORTION OF INIT P.IOC TESTS (INIPIO) WAS LOADED WITH PREVIOUS LOAD OF EIGHT BOOT BLOCKS.
2. FOR DETAILED INFORMATION ON INIPIO TESTS AND ERROR REPORTS, REFER TO HSC50 IN-LINE DIAGNOSTICS USER DOCUMENTATION.

CXO-052C
Sheet 2 of 4

Figure 8–18 (Cont.) HSC50 Boot Flow and Troubleshooting Chart



NOTE: AFTER THE OPERATING SOFTWARE HERALD, ONE OF SEVERAL NONFATAL INITIALIZATION FAILURE MESSAGES MAY BE PRINTED:

- REQUESTOR n FAILED TO INIT STATUS = XXX

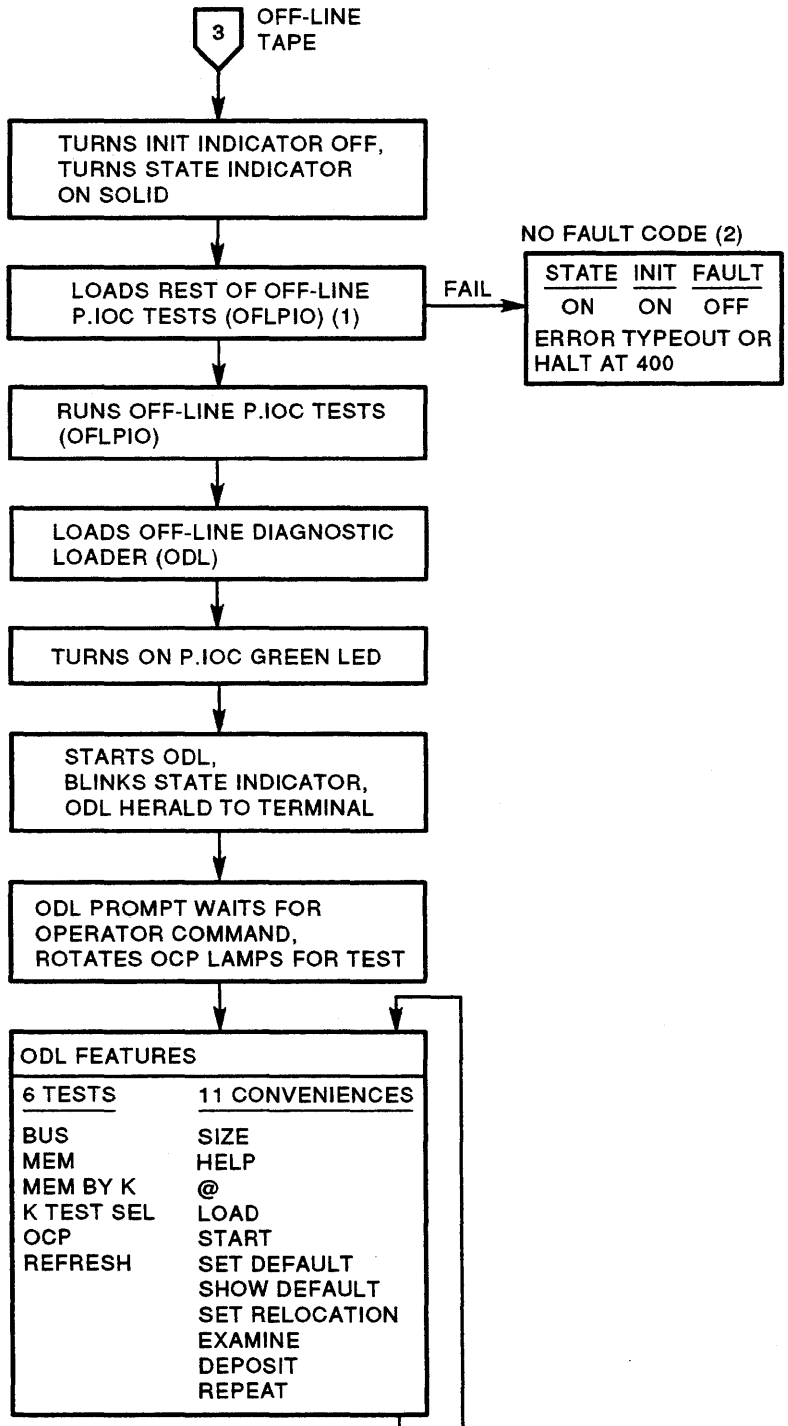
  THIS MESSAGE INDICATES SPECIFIED REQUESTOR FAILED ITS INTERNAL INITIALIZATION SELF-TEST OR COULD NOT BE LOADED WITH MICROCODE (K.SI).

- SWAP BANK BIT SET

  THIS MESSAGE INDICATES THAT A GOOD CONTIGUOUS SECTION OF 8 KW PROGRAM MEMORY COULD NOT BE FOUND WITHOUT USING THE SWAP BANK BIT IN THE P,.IOC CSR. THE MEMORY MODULE IS SUSPECT.

CXO-052C
Sheet 3 of 4

Figure 8–18 (Cont.) HSC50 Boot Flow and Troubleshooting Chart



NOTES:

1. FIRST PORTION OF THE OFLPIO TESTS WAS LOADED WITH PREVIOUS LOAD OF EIGHT BOOT BLOCKS.
2. REFER TO FAULT CODE CHART. FOR DETAILED INFORMATION ON INITPIOC TESTS, REFER TO THE HSC50 IN-LINE DIAGNOSTICS USER DOCUMENTATION.

CXO-052C
Sheet 4 of 4

Figure 8–18 HSC50 Boot Flow and Troubleshooting Chart

8.3.7 Boot Diagnostic Indications

The HSC can pass boot diagnostics with a failing requestor. Although the HSC 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 HSC 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. Refer to Appendix C for a listing of STATUS = nnn codes.

See Section 8.3.1 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

This section explains the different error types in each category, and shows examples of console error formats for each type in a category.

8.4.1 Mass Storage Control Protocol Errors

The Mass Storage Control Protocol (MSCP) or Tape Mass Storage Control Protocol (TMSCP) errors printed out at the console terminal and reported to a host can be one of the following types:

1. Controller errors
2. SDI errors
3. Disk transfer errors
4. STI communication errors
5. STI formatter errors
6. 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. See Section 8.5 for listings and explanations of controller errors.

8.4.2.1 Error Format

Example 8–1 shows the generic error format for all MSCP/TMSCP errors. Optional lines may be used with some errors to display additional information.

```
ERROR-X Text of message at (date) (time)
        Command Ref #   xxxxxxxx
        Err Seq #       x.
        Format Type     xx
        Error Flags     xx
        Event           xxxx
        (Optional line)
        (Optional line)
        (Optional line)
ERROR-X End of error.
```

Example 8–1 MSCP/TMSCP Error Message Format

8.4.2.2 Error Message Fields

Table 8–8 describes the various fields found in an MSCP/TMSCP error message. These are common fields to all error messages of this type.

Table 8–8 MSCP/TMSCP Error Message Field Descriptions

Field	Description
ERROR-X	The X is a code indicating the severity level of an error. The codes are: E for non-fatal, 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 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 number (in hexadecimal) is the byte that describes the detailed format of the error log message. Table 8–9 defines the format type codes. Format Type xx basically defines the type of error packet.
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 8–10.
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 C.

8.4.2.3 Format Type Codes

Table 8–9 defines the format type code numbers. The format type code numbers are in hexadecimal.

Table 8–9 MSCP/TMSCP Error Message Format Type Code Numbers

Number	Definition
00	Controller errors
01	Host memory access errors with memory address
02	Disk transfer errors
03	SDI errors
04	Small disk errors
05	Tape transfer errors
06	STI errors
07	STI drive error log
08	STI formatter error log
09	Bad block replacement

8.4.2.4 Error Flags

Table 8–10 defines the MSCP/TMSCP error flags.

Table 8–10 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	This is MSCP-specific. If set, the identified logical block number (LBN) needs replacement.
4	10	This is 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 was sent to the receiving class driver.

8.4.2.5 Controller Errors

Example 8–2 is a example of a typical MSCP/TMSCP controller error.

```
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
        Format Type      00
        Event            012A
        Buffer Addr      143611
        Source Req.      0.
        Detecting Req.   3.
ERROR-I End of error.
```

Example 8–2 Controller Error Message Example

The direction of data transfer may be deduced from the types of requestors identified in the Source Requestor and Detecting Requestor fields of the error message. These fields correspond to the requestor slot in the HSC backplane. In this example, the source requestor number shows it to be a P.ioj/c, which filled the buffer. The detecting requestor number is 3, which is reading the buffer.

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 8–8. The controller/compare specific fields are shown in Table 8–11.

Table 8–11 MSCP/TMSCP Controller Error Message Field Descriptions

Field	Description
Buffer Addr	This number (in octal) is the starting address of the HSC Data Buffer where the error occurred.
Source Req.	This is the number (in decimal) of the requestor that originally filled the buffer with data.
Detecting Req.	This is the number (in decimal) of the requestor that detected the error.

8.4.2.6 MSCP SDI Errors

The SDI-type errors total 15. Example 8–3 shows a typical SDI error message. Table 8–12 describes the fields specific to SDI errors. Table 8–13, Table 8–14, Table 8–15, and Table 8–16 further define the fields in Table 8–12. For the remaining fields, refer to Table 8–8. For listings and explanations of SDI type errors, see Section 8.5.

```
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
        Format Type      03
        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.
```

Example 8–3 MSCP SDI Error Example

Table 8–12 describes the SDI error example fields.

Table 8–12 MSCP SDI Error Field Descriptions

Field	Description
RA81 unit #	This is the number of the unit the error log message relates to, or is 4095 if the unit number is unknown. In this example, the RA81 indicates the drive is an RA81 and is unit 124.
Request	This number (in hexadecimal) is a byte describing the various requests from the drive for controller action. Figure 8–19 shows the bits of this byte field, and Table 8–13 describes the bits. In this example, the 1B indicates: • RUN/STOP switch in • Port switch in • Logable information in extended area • Spindle ready

Table 8–12 (Cont.) MSCP SDI Error Field Descriptions

Field	Description
Mode	This number (in hexadecimal) is a byte describing the mode of the unit. These modes can be altered by the controller. Figure 8–20 shows the bits of this byte field, and Table 8–14 describes the bits. In this example, the 00 indicates: • No subunits are write-protected. • The disk is in 512-byte sector format.
Error	This number (in hexadecimal) is a byte describing the the current drive error conditions that prevent normal drive operations. Figure 8–21 shows the bits of this byte field, and Table 8–15 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 8–22 shows the bits of this byte field, and Table 8–16 describes the bits. In this example, the 00 indicates: • No subunits with attention available message suppressed in the controller. • 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.)
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.

Table 8–12 (Cont.) MSCP SDI Error Field Descriptions

Field	Description
	In this example, the extended status is: • 88—Controller command functional code last executed by the drive. (In this case, a GET SUBUNIT CHARACTERISTICS command.) • 00—Interface error status bits which are all reset. • 03—Low-order cylinder address bits of the last Seek operation. • 00—High-order cylinder address bits of the last Seek operation. • 07—The present group address. • 4B—Error code (index pulse error) displayed by the drive LEDs during the execution of a drive-resident diagnostic. • 1A—Error code (servo fine positioning error) displayed on the OCP 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.)

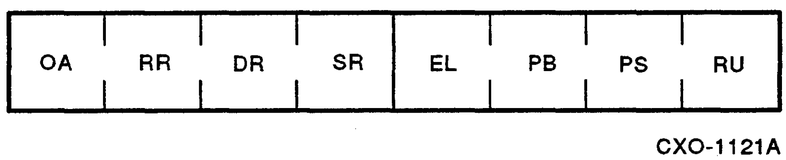


Figure 8–19 Request Byte Field

Table 8–13 Request Byte Field Descriptions

Bit	Description
OA	A logical 1 in this position indicates the drive is unavailable to the controller. A logical 0 indicates the drive is available to the controller.
RR	A logical 1 in this position indicates the drive requires an internal readjustment. Some drives do not use this bit.
DR	A logical 1 in this position indicates a request is outstanding to load a diagnostic in the drive microprocessor memory. A logical 0 indicates no diagnostic is being requested of the host system.
SR	A logical 1 in this position indicates the drive spindle is up to speed. A logical 0 indicates the drive spindle is not up to speed.
EL	A logical 1 in this position indicates usable information in the extended status area. A logical 0 indicates no information is available in the extended status area.
PB	A logical 1 in this bit position indicates the drive is connected to the controller through Port B. A logical 0 indicates the drive is connected through Port A.

Table 8–13 (Cont.) Request Byte Field Descriptions

Bit	Description
PS	A logical 1 in this bit position indicates the drive port select switch for this controller is pushed in (selected). A logical 0 indicates the switch is out.
RU	A logical 1 in this position indicates the RUN/STOP switch is pushed in (RUN). A logical 0 indicates the switch is out (STOP).

W4	W3	W2	W1	DD	FO	DB	S7

CXO-1122A

Figure 8–20 Mode Byte Field

Table 8–14 Mode Byte Field Descriptions

Bit	Description
W4-W1	A logical 1 in any of these four bit positions represents the write-protect status for the subunit. (For example, a 0001 indicates subunit 0 within the selected drive is write-protected.)
DD	A logical 1 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 0 indicates the drive is enabled for communication with a controller.
FO	A logical 1 in this position indicates the drive can be formatted.
DB	A logical 1 in this position indicates the diagnostic cylinders on the drive can be accessed.
S7	A logical 1 in this position indicates the 576-byte sector format is selected. A logical 0 indicates that the 512-byte sector format is selected.

DE	RE	PE	DF	WE			

CXO-1123A

Figure 8–21 Error Byte Field

Table 8–15 Error Byte Field Descriptions

Bit	Description
DE	A logical 1 in this position indicates a drive error has occurred and the drive Fault lamp may be on.
RE	A logical 1 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.

Table 8–15 (Cont.) Error Byte Field Descriptions

Bit	Description
PE	A logical 1 in this position indicates improper command codes or parameters were issued to the drive.
DF	A logical 1 in this position indicates a failure in the initialization routine of the drive.
WE	A logical 1 in this position indicates a write-lock error has occurred.

S4	S3	S2	S1	C1	C2	C3	C4

CXO-1124A

Figure 8–22 Controller Byte Field

Table 8–16 Controller Byte Field Descriptions

Bit	Description
S4-S1	This is a 4-bit representation of the subunits with attention available messages suppressed in the controller. The right-most bit (S1) represents subunit 1. The left-most bit (S4) represents subunit 4. 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.

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.

8.4.2.7 Disk Transfer Errors

Disk transfer errors are either data or media format type errors. Example 8–4 shows a disk transfer error example, and Table 8–17 describes the various fields of the example. See Section 8.5 for listings and explanations of the disk transfer errors.

```
ERROR-E SEVEN Symbol ECC Error at 27-Mar-1985 12:15:15.00
        Command Ref #       50400015
        RA81 unit #         120.
        Err Seq #           9.
        Format Type         02
        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.
```

Example 8–4 Disk Transfer Error Example

Table 8–17 describes the fields in a disk transfer error message not described in Table 8–8. 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 8–17 Disk Transfer Error Field Descriptions

Field	Description
RA81 unit #	This is the number of the unit the error log message relates to, or is 4095 if the unit number is unknown. In this example, the RA81 indicates the drive is an RA81 and is unit 120.
Recovery level	This number indicates the drive 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 drive 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.
Original error flags	This number (octal) indicates the original errors associated with this error. Table 8–18 describes the bits associated with this field. In this example, the 100020 indicates: • ECC error • EDC error
Recovery flags	This number (octal) indicates the recovery flags the software processes should take to recover from this error. Table 8–19 describes the bits associated with this field. In this example, the 000003 indicates: • An LBN should be replaced. • The current error should be logged on the console and to the host if a connection is present.
Lvl A retry count	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 count	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 address	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 Requestor	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 Requestor	This number is the requestor that detected that error. In this example, the 5 indicates requestor number 5 detected the ECC error.

Table 8–18 shows definitions of the original error flags and Table 8–19 defines the recovery flags.

Table 8–18 Original Error Flags Field Descriptions

Bit	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 (high header mismatch)
10	002000	Data sync timeout
09	001000	Drive clock timeout
08	000400	SDI State line pulse or parity 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 = 1's complement of correct EDC)
00	000001	Drive inoperative

Table 8–19 Recovery Flags Field Definitions

Bit	Mask (Octal)	Definition
07	000200	Indicates a revector was done for this LBN.
06	000100	Indicates a positioner error was detected on this block.
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 RCT 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	000001	Indicates the logical block should be replaced.

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:

- Data sync timeout
- ECC symbol error above the threshold
- Header compare error
- Header sync timeout
- Loss of R/W Ready at end of read from disk (SERDES read)
- 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. See Section 8.5 for listings and explanations of BBR errors.

```
ERROR-W Bad Block Replacement (Success) at 18-Dec-1985 18:05:37.1
        Command Ref #    B8590012
        RA60 Unit #      251
        Err Seq #        2
        Format Type      09
        Error Flags      80
        Event            0014
        Replace Flags    8000
        LBN              205
        Old RBN          0
        New RBN          5
        Cause Event      00E8
ERROR-I End of error
```

Example 8–5 Bad Block Replacement Error Example

Table 8–20 defines BBR error fields not previously described in Table 8–8.

Table 8–20 Bad Block Replacement Error Field Definitions

The replace flags field bits are defined in Table 8–21.

Field	Description
Replace Flags	This number (in hexadecimal) indicates bit flags used to report in detail the outcome of the bad block replacement attempt. 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.

Table 8–20 (Cont.) Bad Block Replacement Error Field Definitions

Field	Description
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 for a listing of generic error log fields. In this example, the 00E8 indicates an uncorrectable ECC error caused the bad block replacement.

Table 8–21 Replace Flags Field Bit Descriptions

Bit Number	Bit Mask (Hex)	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 is recommended.
13	2000	Nonprimary revector—This bit is set if the replacement process was accomplished and required putting the bad block data into a replacement block that is not the bad block's primary RBN.
12	1000	Reformat error—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 is reformatted. NOTE: The HSC does not use the REPLACE command as it initiates its own BBR. This message is printed for the HSC equivalent of the REPLACE command such as FORMAT SECTOR.
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 other LBN.

8.4.4 TMSCP 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. (Refer to Section 8.4.1)

See Section 8.5 for listings and explanations of tape errors.

8.4.4.1 STI Communication or Command Errors

Example 8–6 is a sample console printout of an STI communication or command error. Table 8–22 explains the fields not previously defined in Table 8–8.

```
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
```

Example 8–6 STI Communication or Command Error Example

The following table explains the error fields:

Table 8–22 STI Communication or Command Error Printout Field Descriptions

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, and 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 example, and Table 8–23 explains the fields not previously defined in Table 8–8.

```
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.
```

Example 8–7 STI Formatter Error Log Example

The following table explains the error log fields:

Table 8–23 STI Formatter Error Log Field Descriptions

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.
STI Formatter Error Log	See Table 8–24.

Table 8–24 STI Formatter E Log

Byte No.	Byte Data	Description
1	40	Formatter error
2	00	Not set for this example
3	00	Not set for this example
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 example. Table 8–26 describes GEDS Text field, and Table 8–27 describes the Drive Error Log field.

```
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
```

Example 8–8 STI Drive Error Log Example

The following table explains the error log fields:

Table 8–25 STI Drive Error Log Field Descriptions

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 8–26.
Drive Error Log	See Table 8–27.

Refer also to Section 8.4.4.4 for field definitions and bit decoding.

Table 8–26 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	Gap count = 1
6	00	
7	00	
8	00	

The information shown in Table 8–27 is product specific to the TA78. See the *TA78 Service Manual* for details.

Table 8–27 STI Drive Error Log (TA78 Drive Product Specific)

Byte No.	Byte Data	Description
1, 2	00	No soft error
3	00	
4	00	
5	50	Set byte count
6, 7	3B, 04	Operational error Error ID = 59 CRC error ACRC error Pointer mismatch Uncorrectable or two-track error set in ECCSTA register Unknown fault number
8	0D	RMC write fail bits
9	46	Statistics select clock stopped STATUS VALID

Table 8–27 (Cont.) STI Drive Error Log (TA78 Drive Product Specific)

Byte No.	Byte Data	Description
10	FF	Non-BOT command status is OK
11	07	Last cmd sent to M8953 through RCMD = normal NON-BOT read
12	FF	Read channel AMTIE status (CH 7:0)
13	00	
14	00	Read channel illegal status (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)
22	22	Two-track ECC performed on data AMTIE during data of record
23	04	Channel 0 tie bus 2 Amplitude track in error AMTIE
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	STI bus error byte
32	08	CRC to WMC DR bus
33, 34	00	Tape unit selected = 0
35	D9	R/W Data, intermediate DRD bus
36, 37	FF	Byte count = 65535
38, 39	FF	PAD counter = 65535
40, 41	FF	Unknown error code
42	47	DR MBD parity error
43	E6	PE write parity error POWER OK
44	E0	Tape unit ready and on line

Table 8–27 (Cont.) STI Drive Error Log (TA78 Drive Product Specific)

Byte No.	Byte Data	Description
45	00	
46	16	125 ips tape drive
47, 48	25, 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 HSC terminal:

```
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
```

Example 8–9 Tape Drive Related Error Message

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 HSC. 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 left-most byte is referenced as the first byte and the right-most byte as the eighth byte.

Bytes in the GEDS Text field are described in the following list:

- **First byte = Speed**—Currently set speed of the drive; drive speed is defined as 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**—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**—Contain the drive unit number (in 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 HSC 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.
```

Example 8–10 Additional Tape Drive-Related Error Message

The HSC receives the GSS Text field form of this error message from the tape formatter when the HSC issues the GET SUMMARY STATUS (GSS) and TOPOLOGY commands. The field is also the unsuccessful response for all Level 2 commands. Figure 8–23 is a breakdown of this response.

AF	A3	A2	A1	A0	OA	PS	DR	SUMMARY MODE BYTE 1
FE	TE	PE	DF	CE				SUMMARY ERROR BYTE
		AC	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

CXO-2116A

Figure 8–23 GSS Text Field Bits Summary Breakdown

8.4.4.6 GSS Text Field Bit Interpretation

An interpretation of the GSS text field bits follows:

- **AC: Cache attention**
- **AF: Formatter attention asserted**
- **A3: Drive 3 attention asserted**

- **A2: Drive 2 attention asserted**
- **A1: Drive 1 attention asserted**
- **A0: Drive 0 attention asserted**
- **ASM: Automatic speed management**
- **AV: Drive available to formatter**
- **BM: Block mode**
- **BOT: Beginning of tape**
- **CB: Cache busy**
- **CC: Cache capable**
- **CDL: Cache data list**
- **CE: Cache error**
- **CF: Cache full**
- **CMT: Cache empty**
- **C*n*: Controller flags (C0 - C8)**—The following combinations are implemented. All other combinations are reserved.

  C0: Normal operation
  C1: Formatter off line—Formatter is off line to hosts due to being under diagnostic control.

- **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.
- **DIR: Direction**—When clear, indicates the tape will be positioned in the forward direction.
- **DR: 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.
- **ER: Erase**—When set, indicates that a Rewind operation will erase the tape from the current position forward to EOT before rewinding the tape.
- **EX: Exception condition**—Asserted whenever the formatter encounters TM, BOT, or EOT during a data Transfer operation.
- **FD: Retry Bit, failure/direction** – 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.

- **LP: Lengthy operation in progress**—Asserted when a Rewind operation (including the optional data security erase portion of a rewind) is in progress.
- **LS: Long/short success time**—When a formatter rejects a command with cache busy and cache full, it also appropriately sets the LS bit. If the formatter thinks the rejected command can be accepted if immediately issued, LS is clear. If not, LS is set. LS is clear if CB is cleared, and therefore if CB is clear, LS must be clear.
- **MR: Maintenance mode request**—Asserted when the drive is put into maintenance mode. On the TA78, this is accomplished through a thumbwheel switch on the operator panel.
- **NR: No read-ahead**—Set if read-ahead caching is disabled on this unit.
- **OA: Formatter on line or available (for the TOPOLOGY command)**
- **OL: Drive on line to formatter**
- **PB: Active port button**—PB = 0 if the formatter is connected to the controller through port A; PB = 1 if the 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.
- **PR: Position for retry**
- **PS: Port switch**—Asserted when the port switch is enabled.
- **PT: Position for termination**—Positions the tape to where it would have been had there been no error and exits the error recovery state.
- **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
- **RR: Read reverse is supported**
- **RT: Request transfer**—Refer to the explanation for RP.
- **RWC: Rewrite capable**—This bit must not be set if CC is not set.
- **RWE: Rewrite error recovery**—Can only be set if CC is set.
- **SG: Space gaps**—Indicates a location where the tape operation will position the tape. This is determined by the number of gaps specified in the gap field.
- **SM: Speed mask**—SM = 0, supports up to four fixed speeds. SM = 1, supports totally variable speeds.
- **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).
- **SR: Space records**—Positions the tape according to the number of records in the count field.
- **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**
- **U0: Low order drive number bit**—The drive to which a command applies.
- **U1: High order drive number bit**—The drive to which a command applies.
- **UN: Unload**—Unloads a tape after rewind.
- **WB: Write back**—Set if write back caching is enabled on this unit. CC also must be set.
- **WL: Write locked**
- **WP: Write protect**—Set when the controller desires to illuminate the write protect light on the selected unit.
- **ZT: Zero threshold**—Instructs the formatter to change all error thresholds from their default values to zero.

NOTE
Always verify proper dc voltage levels if the indicated possible FRUs do not rectify failure.

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 each of these errors.

The HSC 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 classified into the five categories listed below:

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

See Section 8.5 for listings and explanations of the above categories of out-of-band errors.

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, K.sdi, K.si, and microcode-detected errors.

NOTE
When replacing indicated FRUs, always verify correct dc voltage levels before and after replacing a module.

8.4.5.1 RX33 Errors

Detected errors from the RX33 load device are classified in 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:** Software version mismatch (driver versus operating code).
- **200:** Command aborted through 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 software-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.

In the example, the failing diskette 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 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 8–24.

PAR ERR	NXM ERR	NTR ENA	DMA DIS	TST HI PAR	TST LO PAR	MOTR ENA	DRV SEL	S7	S6	S5	S4	S3	S2	S1	S0

STATUS REGISTER BITS (S7–S0)

CXO-1125B

Figure 8–24 RX33 Floppy Controller CSR Breakdown

Table 8–28 shows the status of the lower CSR bits S7 through S0.

Table 8–28 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 break down. Figure 8–25 shows the breakdown of the last line of the example RX33 error message.

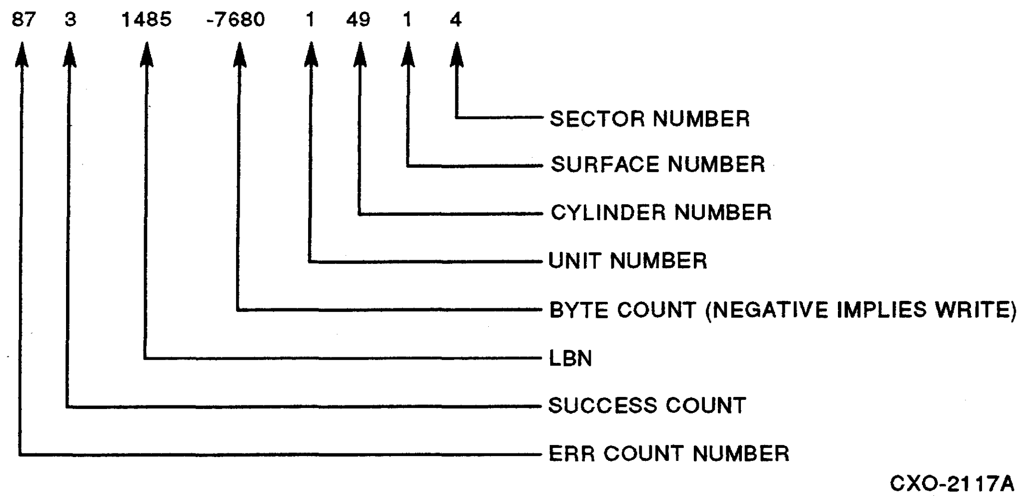


Figure 8–25 RX33 Error Message Last Line Breakdown

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.

8.4.5.2 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, action, and probable FRUs for the disk functional out-of-band errors are listed in Section 8.5.

8.4.5.3 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. See Section 8.5 for listings and explanations of out-of-band tape functional errors.

8.4.5.4 Miscellaneous Errors

Miscellaneous errors are identified by the SINI-E identifier printed on the local console terminal. Many of these messages are one-line or two-line messages, but some have several lines of informational text that result from subsystem exceptions. Subsystem exceptions detect inconsistencies in the operating software. Listings and explanations of SINI errors are located in Section 8.5.

The SINI error messages are a result of the operating software performing a consistency check which failed. When consistency checks fail, the HSC 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 HSC 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 HSC detects an unrecoverable problem, a soft Init or crash occurs. A system dump is performed under the heading SUBSYSTEM EXCEPTION. The HSC then reboots.
2. When the HSC reboots, a message indicating it 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 through 4, Trap through 10, Trap through 114, and Trap through 134) are the same as are found in the 11/70 CPU.

8.4.6.1 NXM (Trap through 4)

If the error registers in the NXM printout equal 170024 000077, the error is not a Nonexistent Memory (NXM) 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 NXM trap produces a subsystem exception printout similar to the example in Section 8.4.6.6.

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 through 10)

The subsystem exception message for this trap indicates the vector number is 10 and identifies the trap as ILOP (an illegal Opcode). Refer to the (PC-6) to (PC): field in the Level 7K interrupt example of Section 8.4.6.6.

With a Trap through 10, 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 be executing in the right place, indicating the problem could be either hardware or software.

8.4.6.3 Parity Error (Trap through 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, 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 through 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, K.si, or K.ci microcode. Requestor hardware-detected errors are the 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/c. The P.ioj/c 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/c. The P.ioj/c 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.

8.4.6.6 Level 7 K Interrupt Example

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 to Appendix D to determine if the failure was a Control bus error.

At this time the HSC reboots. A message is displayed on the local console terminal stating the HSC 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 HSC 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 6 is the failing requestor. The INIPIO-I Booting statement indicates the HSC is attempting to reboot.

When the HSC 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.

8.4.6.7 MMU (Trap through 250)

Following is an sample printout of a detected Memory Management Unit (MMU) failure.

```
**SUBSYSTEM EXCEPTION**    V# Y10B         HSC 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 MMU trap, look first at the register contents of MMSR0 (memory management status register 0). Refer to Figure 8–26 for a breakdown of the bits in MMSR0.

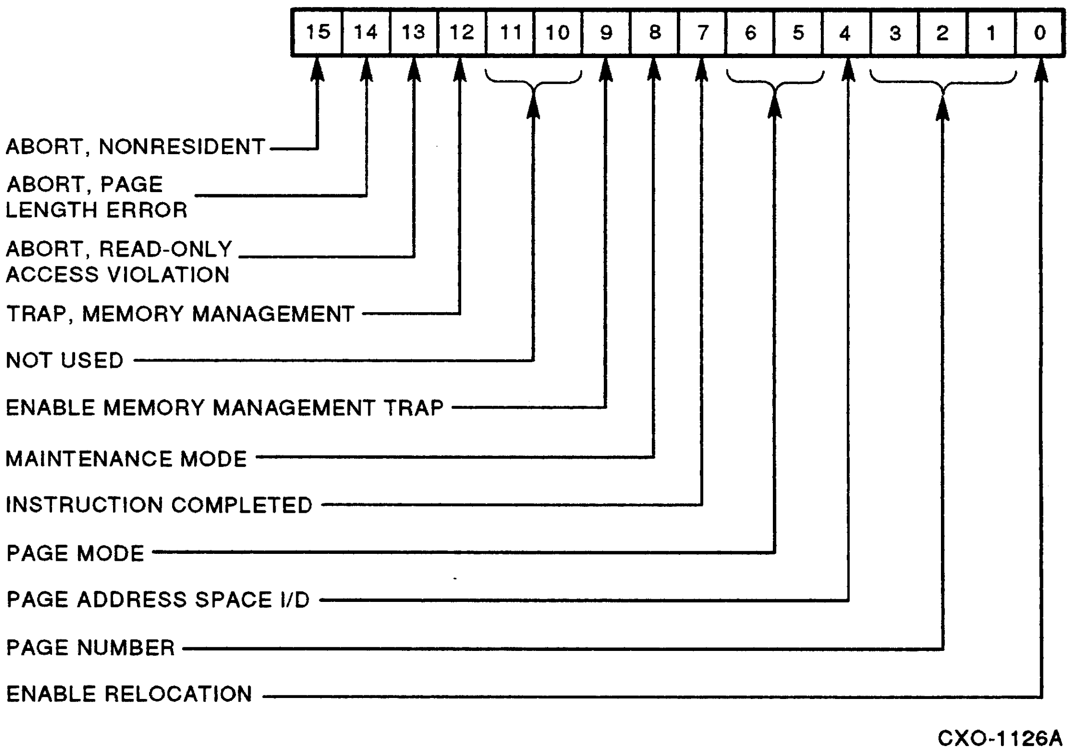


Figure 8–26 MMSR0 Bit Breakdown

Look at the printout lines for MMSR (0-2). Compare the bits set in MMSR0 to the bit breakdown in Figure 8–26. 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 in which the HSC reported this error. A 14xxxx in the PSW means User mode, a 00xxxx 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 two starts with a 013., indicating a short page.

Two possible problems cause this error:

1. Memory Management Unit on the P.ioj/c
2. Software

Software inconsistency (Trap through 20) is reported similar to an MMU trap. A subsystem exception is dumped on the local console terminal with the trap vector reported being a Trap through 20 (AT). An example printout and explanation are found in Appendix B.

The subsystem exception is followed by the HSC reboot. Upon successful reboot, the following message is displayed.

```
HSC 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
```

8.5 Alphabetical Listing of Software Error Messages

Each message description includes the following:

a. **Actual error message**—Displayed in English at the HSC console terminal.
b. **Error type**—The subsystem where the error occurred or was detected.
c. **Error message severity level**—Included in the error message.
d. **Message description**—A review of what the error is about.
e. **Action**—Remedies or troubleshooting paths that the Customer Service representative can take to correct the problem.
f. **Possible FRUs**—Suggested components that are likely suspects causing the malfunction.

Aborting Error Recovery Due to Excessive RECALs

Disk Unit xxx.
Requestor xx
Port xx

Error Type: Disk functional out-of-band

Severity: Error

Description: For each group is a transfer, a count of the number of RECALS issued to the drive is kept. If the count exceeds a hard-coded value, this message is printed. Recovery from an error is not possible because of excessive RECALS and the drive is declared inoperative.

Action: Refer to the drive service manual and any other type errors being logged to determine reasons for persistent positioning failures.

Possible FRUs: Drive modules. (Refer to the drive service manual.)

Aborting Error Recovery Due to Excessive Timeouts

Disk Unit xxx.

Error Type: Disk functional out-of-band

Severity: Informational

Description: The HSC detects several timeouts on the disk drive. A timeout occurs because the drive did not complete its expected work in the expected time. All error recovery attempts will be aborted and the drive will be declared inoperative.

Action: May need to replace the following FRUs. Further testing may be necessary.

Possible FRUs:

1. Drive modules. (Refer to the drive service manual.)
2. K.sdi/K.si module

Acknowledge Not Asserted at Start of Transfer

Error Type: Tape error

Severity: Error

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.

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/K.si module
3. STI cable set

ATN Message Sent to Node xx, for Unit xx.

Error Type: Disk functional out-of-band

Severity: Informational

Description: An attention condition was found on the indicated drive unit and an attention message has been sent to the host to notify it of this condition.

Action: None

Possible FRUs: None

Attention Condition serviced for ONLINE disk unit xxx.

Error Type: Disk functional out-of-band

Severity: Informational

Description: An attention condition indicating a state change in the drive needs servicing. A GET STATUS exchange is invoked to the drive. Note: This may not indicate a failure condition.

Action: Refer to the console printed Get Status response.

Possible FRUs: Drive modules. (Refer to the drive service manual.)

Bad Block Replacement (Block OK)

Error Type: BBR error

Severity: Warning

Description: Block tested OK—not replaced.

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 8–20.

Bad Block Replacement (Drive Inoperative)

Error Type: BBR error

Severity: Warning

Description: Replacement failure or 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.

Action: Drive should be tested further. Move the drive to another K.sdi/K.si (or to just another K.sdi/K.si port) if available. If the problem persists, failure is probably in the drive.

Possible FRUs: Drive modules. (Refer to the drive service manual.)

Bad Block Replacement (RCT Inconsistent)

Error Type: BBR error

Severity: Warning

Description: Replacement failure–the RCT table is not usable.

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 the drive may be necessary.

Possible FRUs: Drive modules. (Refer to the drive service manual.)

Bad Block Replacement (Recursion Failure)

Error Type: BBR error

Severity: Warning

Description: Replacement failure—recursive failure. Two successive RBNs were bad.

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 8–20.

Bad Block Replacement (REPLACE Failed)

Error Type: BBR error

Severity: Warning

Description: Replacement failure—REPLACE command or its analogue failed. The status returned from the replacement process indicates the command was not successful.

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 the drive may be necessary.

Possible FRUs: Drive modules. (Refer to the drive service manual.)

Bad Block Replacement (Success)

Error Type: BBR error

Severity: Warning

Description: The bad block was successfully replaced.

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 8–20.

Bad Dispatch State in CB...

Error Type: CI-detected out-of-band

Severity: Warning

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/M.std, or P.ioj/c modules.

Action: Replace the following FRUs.

Possible FRUs:

1. K.pli module
2. M.std2/M.std module
3. P.ioj/c module

Booted from Drive 1. Drive 0 Error (text)

Error Type: SINI error

Severity: Informational

Description: The system was booted from system device drive 1. Normal boot is from drive 0.

Action: None

Possible FRUs: Drive 0

Buffer EDC Error

Error Type: Tape error

Severity: Error

Description: The K.sti/K.si detected an EDC error on the Data Buffer it read from memory on a Write operation.

Action: Test the data path from K.sti/K.si to HSC Data memory and the K.ci.

Possible FRUs:

1. Formatter
2. M.std2/M.std module
3. K.sti/K.si module
4. K.ci module

Cache Disabled Due to Failure

Error Type: SINI error

Severity: Error

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.

Action: Load the Off-line 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

Clock dropout from ONLINE disk unit xx.

Error Type: Disk functional out-of-band

Severity: Error

Description: The on-line disk has lost its real-time state clock.

Action: Check the path between the K.sdi/K.si 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 the drive service manual.)
2. SDI cable
3. K.sdi/K.si module

Compare Error

Error Type: Controller error

Severity: Error

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.

Action: Isolate the FRU by moving the disk or tape drive 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. K.sdi/K.sti/K.si module
3. M.std2/M.std module
4. K.ci module set
5. Host CI/memory

Controller Detected Position Lost

Error Type: Tape error

Severity: Error

Description: Information contained in the response from the formatter to the HSC POSITION command did not match the expected tape drive position.

Action: Check the formatter. If the error persists, run the In-line Tape (ILTAPE) diagnostic to help isolate to the FRU.

Possible FRUs: Formatter

Controller Transfer Retry Limit Exceeded

Error Type: Tape error

Severity: Error

Description: The controller failed to perform the command within the limit of allowable retries.

Action: Check the formatter and the drive.

Possible FRUs:

1. Drive modules. (Refer to the drive service manual.)
2. Formatter

Controller Detected Transmission or Timeout Error

Error Type: SDI

Severity: Error

Description: The controller detected an invalid framing code or a checksum error in a level 2 response from the SDI drive.

Action: Determine if this error is occurring on more than one drive, which may indicate a K.sdi/K.si problem. However, if it is occurring only on 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/K.si module
4. SDI transition bulkheads

Could Not Complete On-line Sequence

Error Type: Tape error

Severity: Error

Description: Could not complete on-line sequence due to a condition in the drive.

Action: Check the formatter and the drive.

Possible FRUs:

1. Drive modules. (Refer to the drive service manual.)
2. Formatter

Could Not Get Extended Drive Status

Error Type: Tape error

Severity: Error

Description: Issued the GET EXTENDED DRIVE STATUS command and the drive did not respond with the extended drive status.

Action: Check the formatter.

Possible FRUs: Formatter

Could Not Get Formatter Summary Status During Transfer Error Recovery

Error Type: Tape error

Severity: Error

Description: Issued the command and the formatter did not respond with the formatter summary.

Action: Check the formatter.

Possible FRUs: Formatter

Could Not Get Formatter Summary Status While Trying to Restore Tape Position

Error Type: Tape error

Severity: Error

Description: Issued the command and the formatter did not respond with the formatter summary status.

Action: Check the formatter.

Possible FRUs: Formatter

Could Not Position for Formatter Retry

Error Type: Tape error

Severity: Error

Description: The HSC issued a command for data recovery with position required, and the drive could not complete the command.

Action: Check the media, drive, and formatter.

Possible FRUs:

1. Drive modules. (Refer to the drive service manual.)
2. Media
3. Formatter

Could Not Set Byte Count

Error Type: Tape error

Severity: Error

Description: Issued command to set byte count and could not complete command.

Action: Check the formatter.

Possible FRUs: Formatter

Could Not Set Unit Characteristics

Error Type: Tape error

Severity: Error

Description: Issued command to set unit characteristics and could not complete command.

Action: Check the formatter.

Possible FRUs: Formatter

Data Bus Overrun

Error Type: Controller error

Severity: Error

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 than 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.

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 Error Flagged in Backup Record

Disk Unit xx LBN xx
Tape Unit xx

Error Type: Tape functional out-of-band

Severity: Warning

Description: During a backup, a data error was encountered. During the BBR, the record was written with a forced error bit set.

Action: Check BBR history on source drive.

Possible FRUs:

1. Disk unit
2. Media

Data Memory Error (NXM or Parity)

Error Type: Controller error

Severity: Error

Description: The HSC detected an error in internal Data memory. The error was either a parity error, detected through 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).

Action: Determine if this error is repetitive; if so, the problem is probably the M.std2/M.std 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/M.std module or a possible Data bus problem.

Data Ready Timeout

Error Type: Tape error

Severity: Error

Description: The controller did not detect Data Ready from the formatter within the timeout interval after sending it a Level 1 command.

Action: Check the STI path.

Possible FRUs:

1. STI cable set
2. K.sti/K.si module
3. Formatter

Data Sync Not Found

Error Type: Disk transfer error

Severity: Error

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/K.si has already read a valid header and is awaiting the data SYNC character.

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/K.si module.

Possible FRUs:

1. Drive modules. (Refer to the drive service manual.)
2. K.sdi/K.si module
3. SDI interface

Date/Time Set By Node nn

Error Type: CI-detected out-of-band error

Severity: Informational

Description: The HSC received either a START or STACK (start acknowledge) message over the CI, and the date and time was not set.

Action: None. This is a normal message as part of establishing a VC between a host and an HSC.

Possible FRUs: None

Deferred ATTN. Message for Node xx, Unit xx.

Error Type: Disk functional out-of-band error

Severity: Informational

Description: An attention message is delayed in process.

Action: None

Possible FRUs: None

Disk unit xx. (Requestor xx., Port xx.) being INITialized.

DCB addr: xxxxxx

Error Type: Disk functional out-of-band

Severity: Informational

Description: A disk is being initialized.

Action: None

Possible FRUs: None

Disk unit xx. ready to transfer.

Retrieval failure or subsystem deadlock probable.

Error Type: Disk functional out-of-band

Severity: Informational

Description: A disk transfer did not complete within the allowable timeout period. The HSC software cannot detect any problems to account for the failure. Possible problems include:

1. No available buffers
2. Drive problems
3. K.sdi/K.si problems

Action: Check data transfer path. This error may indicate too many utilities or in-line diagnostics running simultaneously. The problem might also be an HSC software problem.

Possible FRUs: K.sdi/K.si module

Disk unit xxx. (Requestor xx., Port xx.) declared inoperative.

Intervention required.

Error Type: Disk functional out-of-band

Severity: Error

Description: The Disk Path process has concluded that the drive is no longer usable. Any pending I/O is cleaned up and the drive state is set to either UNDEFINED or OFFLINE. The HSC ignores the disk until it detects some intervention.

Action: Examine previous error reports to help resolve failure. Toggle port switch on drive.

Possible FRUs: Drive modules. (Refer to the drive service manual.)

DRAT/SEEK timeout, disk unit xxx.

Error Type: Disk functional out-of-band

Severity: Informational

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 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/K.si 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. A SEEK has timed out.

4. Another element in the subsystem that should have supplied resources to the disk transfer operation in a reasonable time did not.

Action: Check the drive.

Possible FRUs: Drive modules. (Refer to the drive service manual.)

DRIVE CLEAR attempt on disk unit xx. (Requestor xx., Port xx.).

DCB addr: xxxxxx Error count ****.**

Error Type: Disk functional out-of-band

Severity: Informational

Description: The drive detected some previous error and the HSC is now attempting to clear that error.

Action: Examine the host error log to determine what error the drive is trying to clear.

Possible FRUs: Drive

Drive Clock Dropout

Error Type: SDI error

Severity: Error

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, usually by means of a timeout.

Action: Determine if this error is occurring on more than one drive, which may indicate a K.sdi/K.si 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/K.si module
4. SDI transition bulkheads

Drive Detected Error

Error Type: SDI error

Severity: Error

Description: The controller received a GET STATUS command or unsuccessful response with the EL bit set, or the controller received a response with the DR flag set and does not support automatic diagnosis for that SDI drive type.

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 the drive service manual.)

2. SDI cables
3. SDI bulkheads

Drive Inoperative

Error Type: SDI error

Severity: Error

Description: The HSC has marked the drive inoperative due to an unrecoverable error in the previous level 2 exchange, the drive's C1 flag is set, or the drive has a duplicate unit identifier. Once the HSC reports the drive as inoperative, the drive state clocks must transition to return the drive to an operational state.

Action: Refer to the drive service manual. Run ILDISK to help isolate failure between HSC and drive.

Possible FRUs:

1. Drive modules. (Refer to the drive service manual.)
2. K.sdi/K.si module
3. SDI cables

Drive Requested Error Log (EL Bit Set)

Error Type: SDI error

Severity: Error

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.

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.)

Duplicate Disk Unit xx

Error Type: Disk functional out-of-band

Severity: Informational

Description: Disk unit numbers are duplicated within the system.

Action: Locate the duplicate disks and change the plug number on one.

Possible FRUs: Drive modules. (Refer to the drive service manual.)

EDC Error

Error Type: Controller error

Severity: Error

Description: The sector was read with correct or correctable ECC and invalid EDC. A fault probably exists in the 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).

Action: Determine if other errors indicate a problem with the data path circuitry on the indicated requestor modules.

Possible FRUs:

1. K.sdi/K.si module
2. M.std2/M.std module, if an address parity error on Data memory occurs, as this is checked by the EDC field.

ERASE Command Failed

Error Type: Tape error

Severity: Error

Description: Issued ERASE command and command failed.

Action: Check the formatter.

Possible FRUs: Formatter

ERASE GAP Command Failed

Error Type: Tape error

Severity: Error

Description: Issued ERASE GAP command and command failed.

Action: Check the formatter.

Possible FRUs: Formatter

Forced Error

Error Type: Disk transfer error

Severity: Error

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.

Action: Restore the media from a previous backup. A VMS (HSC) backup and restore of the current media will clear the forced error condition but will leave the sector corrupt.

Possible FRUs: None

Formatter Detected Position Lost

Error Type: Tape error

Severity: Error

Description: The formatter lost track of tape position.

Action: Check the media, drive, and formatter.

Possible FRUs:

1. Drive modules. (Refer to the drive service manual.)
2. Formatter
3. Media

Formatter and HSC Disagree On Tape Position

Error Type: Tape error

Severity: Error

Description: The formatter and the HSC disagree on position of the tape.

Action: Check the formatter.

Possible FRUs:

1. Drive modules. (Refer to the drive service manual.)
2. Formatter
3. K.sti/K.si module

Formatter Requested Error Log

Error Type: Tape error

Severity: Error

Description: The formatter detected an error and set the EL bit to request an error log be taken.

Action: Check the formatter.

Possible FRUs: Formatter

Formatter Retry Sequence Exhausted

Error Type: Tape error

Severity: Error

Description: The formatter failed to complete a command within the retry limit.

Action: Check the media, drive, and formatter.

Possible FRUs:

1. Drive modules. (Refer to the drive service manual.)
2. Formatter
3. Media

FRB Error: K.ci, 1st LBN xx., xx. buffers, FE$SUM xx

Error Type: Disk functional out-of-band

Severity: Informational

Description: An error was detected by the K.ci while processing a Fragment Request Block (FRB) and the FRB has been sent to the disk error process. Example: EDC error.

Action: If excessive, reformat drive.

Possible FRUs: Drive modules. (Refer to the drive service manual.)

FRB Error: K.sdi, Unit xx., 1st LBN xxx., xx. buffers, FE$SUM xx

Error Type: Disk functional out-of-band

Severity: Informational

Description: An error was detected by the K.sdi/K.si while processing a Fragment Request Block (FRB) and the FRB has been sent to the disk error process. Example: Suspected Positioner error.

Action: If excessive, reformat drive.

Possible FRUs: Drive modules. (Refer to the drive service manual.)

Hard transfer error loading (file) xx

Error Type: SINI out-of-band

Severity: Error

Description: The P.ioj/c detected a hard error while loading a file from the system media 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.

Action: Load the file from the other system load device; load the back-up media.

Possible FRUs:

1. System media
2. System load device

Hard transfer error writing SCT xx

Error Type: SINI out-of-band error

Severity: Error

Description: The HSC detected an error while attempting to write the SCT on the console load media. The xx designates the octal byte that is the error status value returned from the device driver.

Action: Make sure the drive is not write-protected; try the back-up media; try the other system load device.

Possible FRUs:

1. System media
2. System load device

Header Error

Error Type: SDI error

Severity: Error

Description: The subsystem reads an inconsistent or invalid header for the requested sector.

The header is inconsistent if three out of four copies of the high order header word do not match.

The header is considered invalid if all of the following are true:

- The header is consistent (three out of four copies of the high order header word 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 to read the 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 mis-sync, header sync timeout, and an unreadable header.

Action: Determine if this error is repetitive on this unit indicating a deteriorating media.

Possible FRUs:

1. Drive modules. (Refer to the drive service manual.)
2. K.sdi/K.si module

HML$ER set—HM$ERR = nn

Error Type: CI-detected out-of-band

Severity: Warning

Description: A Host Memory Block (HMB) operation resulted in an error. A breakdown of HMB error word (HM$ERR) bits follow.

- 000002 HME$BM—Insufficient BMBs to receive message.
- 000004 HME$NC—Sequenced message received over a connection with 0 in credit field.
- 000010 HME$NC—Sequenced message received over a connection with credit field > 1. Excess has been added to CB$EM.
- 000020 HME$OV—Oversize message received (>1096. bytes).
- 000040 HME$DN—Data memory NXM during BMB operation.
- 000100 HME$DP—Data memory parity error in BMB operation.
- 000200 HME$DO—Data memory overrun during BMB operation.
- 000400 HME$FP—Reception buffer parity error in packet header. Message not receivable.
- 001000 HME$PL—Reception buffer parity error in body of message.
- 002000 HME$CN—Transmission not attempted because connection not valid.
- 004000 HME$VC—Transmission not attempted because VC closed or connection invalid.
- 010000 HME$TE—Transmission attempted but failed (no ACK).
- 020000 HME$TP—Transmission failed due to transmission buffer parity error.
- 040000 HME$HC—Packet inconsistent with K.ci context received from host.
- 100000 HME$IC—Illegal control function Opcode.

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/M.std 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/M.std module

Host Clear from CI Node

Error Type: SINI out-of-band

Severity: Error

Description: The host cannot function with the HSC for some reason, such as a nonresponse within a certain amount of time or too many errors on the CI.

Action: Check the HSC console messages and the error logs of the systems connected to the HSC.

Possible FRUs:

1. CI cable
2. HSC/CI interface
3. Host/CI interface
4. HSC software
5. System software

Host interface (K.ci) failed INIT diags, status = xxx

Error Type: SINI out-of-band

Severity: Error

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.

Action: Determine which is the failing module by comparing the failing status value to the values in Appendix D. This comparison points 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

Error Type: SINI out-of-band

Severity: Error

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.

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 Off-line diagnostics. This error generates a soft fault and is examined by pressing the fault button on the OCP.

Possible FRUs: See list below and 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)

Host Requested Retry Suppression On A Formatter Detected Error

Error Type: Tape error

Severity: Error

Description: The formatter detected an error and the host issued a command to suppress the retry of the command that failed.

Action: Check the formatter.

Possible FRUs: Formatter

Host Requested Retry Suppression On A K.sti/K.si Detected Error

Error Type: Tape error

Severity: Error

Description: An error was detected in the K.sti/K.si and the host issued a command to suppress the retry of the command that failed.

Action: Check the K.sti/K.si.

Possible FRUs: K.sti/K.si module

Illegal bit change in status from disk unit xxx.

EL bit forced on so status logged.

Error Type: Disk functional out-of-band

Severity: Error

Description: An unsupported bit was received in status returned from the disk unit.

Action: Check the drive and the version of software in HSC.

Possible FRUs:

1. Drive modules. (Refer to the drive service manual.)
2. Version of software

Insufficient Control Memory for K.sti/K.si in Requestor xx

Error Type: Tape functional out-of-band

Severity: Error

Description: Not enough Control memory left in the pool to allocate a control block. A certain amount of Control memory is needed to set up control blocks. Enough memory was not found to set up control blocks for turning on the K.sti/K.si functional code.

Action: Use the HSC SETSHO utility to show available HSC memory (Control, Data, and Program). If less than 87.5 percent of available Control memory is usable, replace M.std2/M.std module. Run Off-line TEST MEM by K diagnostic and test Control memory.

Possible FRUs:

1. M.std2/M.std module
2. P.ioj/c module
3. Software

Insufficient Private Memory remaining for TMSCP Server

Error Type: Tape functional out-of-band

Severity: Error

Description: In the SCT, a parameter determines the maximum number of supported tape formatters. During initialization, all the working K.sti/K.si 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.

Action: Use HSC SETSHO utility to show available HSC Program memory. If less than 87.5 percent of available Program memory is usable, replace M.std2. Run Off-line Test Mem or Test Refresh to test Program memory. Use the SETSHO SET MAX FORMATTER command to reduce the maximum number of formatters supported.

Possible FRUs:

1. M.std2/M.std module
2. P.ioj/c module
3. Software

Internal Consistency Error

Error Type: Controller error

Severity: Error

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 probably caused by the requestor microcode or hardware.

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/M.std memory module

K.ci exception detected, code = nnn

Error Type: CI-detected out-of-band

Severity: Warning

Description: The code is composed of 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.

000001 KHF$PD—Path(s) disabled by K.ci due to a transmit error or VC breakage due to other K.ci-detected errors.
000002 KHF$EQ—Item(s) placed on error queue (KH$EQ).
000004 KHF$BL—Data memory error during BMB list operation.
000010 KHF$UP—Unreceivable packet. K.ci stopped (causes a crash).
000100 KHF$NH—Sequenced message received while reserved-to-receive queue was empty.

040000 KHF$PD—Set by diagnostics to disable interrupts.

Action: Compare the code from the printout to the previous list, and determine whether the error code points to an HSC module or to the host.

Possible FRUs:

1. Status 1: K.pli module
2. Status 4: M.std2/M.std module
3. Status 10: PILA module, host K.ci set
4. Status 100: Host K.ci set

K.ci loopback microcode loaded

Error Type: CI-detected out-of-band

Severity: Error

Description: The CIMGR detected K.ci loopback microcode was loaded during initialization. When this message occurs, a problem with the K.pli (L0107) module probably exists.

Action: Replace the following FRU.

Possible FRUs: K.pli module

K.sdi/K.si in slot xx. failed its init DIT, status = xxx

Error Type: Disk functional out-of-band

Severity: Error

Description: A requestor fails during boot. The displayed K.sdi/K.si has failed with the displayed status. This message is displayed only at the end of the boot procedure.

Action: Record the status for module repair purposes.

Possible FRUs: The K.sdi/K.si module displayed.

K.sti/K.si in Requestor xx has microcode incompatible with this TMSCP Server

Error Type: Tape functional out-of-band

Severity: Error

Description: The data structure version within the microcode version residing on the K.sti/K.si module is a lower version than the TMSCP Server can support.

Action: Use the SET REQUESTOR command to ensure the version of microcode on the K.sti/K.si module is up to current revision. If not, replace the microcode or replace the K.sti/K.si module with a K.sti/K.si module of the current revision.

Possible FRUs: K.sti/K.si module

Last soft init resulted from unknown cause

Error Type: SINI out-of-band

Severity: Error

Description: Software has a list of known reasons for reboot (Trap through 134, Trap through 250, CRASH$, SETSHO, etc.). If no reason for reboot is apparent, the software may have failed to detect where the error came from.

Action: Check the HSC console error messages and the system error logs on all the systems connected to the HSC. This error indicates a probable software problem.

Possible FRUs: Dependent upon the information obtained from the error logs.

LBN xx. repaired for shadow member unit xx.

Error Type: Disk functional out-of-band

Severity: Informational

Description: A shadow Repair operation was done in which good data was written to bad members of a shadow set.

Action: An uncorrectable error occurred on an LBN on the subject drive and was successfully rewritten. If the problem persists, check for other errors that would give information on what the uncorrectable error was.

Possible FRUs: Drive modules or media. (Refer to the drive service manual.)

LBN Restored with Forced Error in RESTOR operation

Disk Unit xx, LBN xx.
Tape Unit xx.

Error Type: Disk functional out-of-band

Severity: Warning

Description: An error was detected in the LBN data during backup. A forced error bit was set in the LBN.

Action: If excessive, reformat drive.

Possible FRUs: Drive modules. (Refer to the drive service manual.)

Less than 87.5 percent of (Control, Data, Program) memory is available

Error Type: SINI out-of-band

Severity: Error

Description: These three messages are a result of the P.ioj/c polling the memories on initialization and finding an insufficient amount of working memory in either one. Any combination of the three messages may appear.

Action: The error printout determines which memory is failing.

Possible FRUs: M.std2/M.std module

Level 7 K interrupt (Trap through 134)

Error Type: SINI out-of-band, subsystem exception

Severity: Error

Description: A level 7 K interrupt occurs when any requestor detects a fatal error condition while executing functional code. The requestor, upon detecting the error, generates a level 7 K interrupt to the P.ioj/c. The P.ioj/c 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.

Action: In some cases, the error printout shows a failing requestor when the real problem is in the M.std2/M.std module. This error can also be caused by software problems.

Wait for two or more failures of this type to determine if the real problem is the M.std2/M.std module. If the M.std2/M.std 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/M.std module

Lost Read/Write Ready

Error Type: SDI error

Severity: Error

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.

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 the drive service manual.)
2. K.sdi/K.si module
3. SDI cables
4. SDI transition bulkheads

Lost Receiver Ready

Error Type: SDI error

Severity: Error

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's timeout expiring for a Transfer operation (Level 1 real-time command).

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 the drive service manual.)
2. K.sdi/K.si module
3. SDI cables
4. SDI transition bulkheads

Lower Processor Error

Error Type: Tape error

Severity: Error

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.

Action: Check the K.sti/K.si and the tape formatter.

Possible FRUs: K.sti/K.si module or tape formatter

Lower Processor timeout

Error Type: Tape error

Severity: Error

Description: The Upper Processor in the K.sti/K.si detected the Lower Processor had stopped and restarted it.

Action: Check the K.sti/K.si and tape formatter.

Possible FRUs: K.sti/K.si module or tape formatter

MMU (Trap through 250)

Error Type: SINI out-of-band, subsystem exception

Severity: Error

Description: A failure was detected in the Memory Management Unit (MMU) on the P.ioj/c. The active process is displayed as well as the bit assignments for the memory management status registers.

Action: Examine the MMSR registers to determine the failure in the MMU.

Possible FRUs: P.ioj/c module or software error

nnn Symbol ECC Error

Error Type: Disk transfer error

Severity: Error

Description: If a drive has more symbols in error than a drive-defined threshold, the HSC will print one of the following error messages, even though the error might have been corrected.

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
Uncorrectable ECC Error

The following description covers all of the ECC error types that are printed.

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, depending on the drive type, 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 through an end packet.

NOTE
An uncorrectable ECC error is reported when a transfer with the Suppress Error Correction modifier encounters an ECC error of any severity.

Action: Determine if the ECC errors are just normal events 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 the drive service manual.)
2. K.sdi/K.si module

No control block available to satisfy HMB request.

Error Type: CI-detected out-of-band

Severity: Warning

Description: The CIMGR tried to allocate an Host Memory Block (HMB) 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.

Action: Type in the SHOW MEMORY command for HSC50 software version V300 and later and HSC 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 12.5% has been disabled from use, replace the memory module.

For HSC50 software before V300, run the off-line 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/M.std module
2. Software

No tape drive structures available for Requestor xx Port xx Unit xx

Increase structures through SET MAX_TAPE command

Error Type: Tape functional out-of-band

Severity: Error

Description: An additional tape drive has been added to an existing tape formatter, but the tape structures set up in initialization have been exceeded.

Action: Use the SETSHO utility to increase to the number of tape structures with the SET MAX_TAPE command.

Possible FRUs: None

No tape formatter structures available for Requestor xx Port xx

Increase structures through SET MAX_FORMATTER command

Error Type: Tape functional out-of-band

Severity: Error

Description: An additional tape formatter has been added to the HSC, but since tape formatter structures are set up during initialization, not enough structure space is available for this additional tape formatter.

Action: Use the SETSHO utility to set the structure level higher to compensate for the additional tape formatter with the SET MAX_FORMATTER command.

Possible FRUs: None

No usable K.sti/K.si boards were found by the TMSCP Server

Error Type: Tape functional out-of-band

Severity: Error

Description: The TMSCP Server polled the HSC and found no working K.sti/K.si modules. This message does not appear frequently because the TMSCP Server software is not usually loaded if there are no K.sti/K.si modules.

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/K.si(s) module displaying the failing status.

Node nn cables have gone from crossed to uncrossed

Error Type: CI-detected out-of-band

Severity: Error

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 the error message Node nn Cables have gone from uncrossed to crossed.

Action: Note, if both the "uncrossed to crossed" and "crossed to uncrossed" messages are occurring, it is probably an indication of failing hardware, not a cable problem. See the Action in the next 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 cables have gone from uncrossed to crossed

Error Type: CI-detected out-of-band

Severity: Warning

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 was used by the sending node. If these two bits do not indicate the same path the HSC received the message on, this error occurs.

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: LINK (L0100/L0118), PILA (L0109), and K.pli (L0107)
3. Host CI module set
4. Duplicate node address settings

Node nn path has gone from bad to good

Error Type: CI-detected out-of-band

Severity: Warning

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 the 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 ID_REQ (ID Request) packets to all nodes and receives successful ID_RSP ID Response messages.

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

Node nn path (A or B) has gone from good to bad

Error Type: CI-detected out-of-band

Severity: Warning

Description: K.ci microcode detects a hard (unrecoverable) transmission error on a previously good path. Examples of hard transmission errors are:

Transmit Buffer Parity Error
Unrecoverable NACK
Unrecoverable NO_RSP
Transmitter Attention Timeout

Determining the reason for failure using the error message is not possible.

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 probably in the Star Coupler's host side. If the problem involves multiple hosts, it is probably 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. HSC/CI interface
3. Host/CI interface

NXM (Trap through 4)

Error Type: SINI out-of-band, subsystem exception

Severity: Error

Description:

a. A memory location did not respond within the specified timeout period.

b. A stack overflow occurred.

c. An odd address access was attempted. For example, a word access instead of a byte.

d. A halt was executed in User mode.

Action: Determine which memory is failing by examining the low and high error address registers for module repair.

Possible FRUs:

1. M.std2/M.std module
2. P.ioj/c module

Parameter change, process yyy

PC xxx
PSW xxx
Reason xxx

Error Type: SINI out-of-band, subsystem exception

Severity: Informational

Description: A parameter has been changed through the SET/SHO utility.

Action: None

Possible FRUs: None

Parity Error (Trap through 114)

Error Type: SINI out-of-band, subsystem exception

Severity: Error

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.

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/c module
2. M.std2/M.std module

P.ioj/c running with memory bank or board swap enabled

Error Type: SINI out-of-band

Severity: Error

Description: Upon initialization, an error was detected in the low address space of private memory. The P.ioj/c asserted the Swap Bank signal, and the second bank of private memory was enabled. The P.ioj/c and memory combination can still function under limited capabilities.

Action: Replace the M.std2/M.std module. The HSC still functions with limited capabilities.

Possible FRUs: Replace memory module

PLI Receive Buffer Parity Error

Error Type: Controller error

Severity: Error

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/L0118) and checked by the K.pli module (L0107). The PILA module stores the data without checking or generating parity.

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 Off-line 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 the sequence of possible FRUs.

Possible FRUs:

1. PILA module
2. K.pli module
3. LINK module

PLI Transmit Buffer Parity Error

Error Type: Controller error

Severity: Error

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.

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 Off-line Test K diagnostic to test K.ci. Any error report should more clearly indicate specific K.ci module failure. For very intermittent failures follow the sequence of possible FRUs.

Possible FRUs:

1. PILA module

2. LINK module
3. K.pli module

Position or Unintelligible Header Error

Error Type: SDI error

Severity: Error

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 transfer will be retried several times and the error is considered recoverable if the error flags bit indicates success or a subsequent replacement succeeds.

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 the drive service manual.)
2. K.sdi/K.si module

Positioner error on disk unit xxx. DRAT addr: xxx

Desired hdr (lo,hi):xxx, xxx
Actual hdr (lo,hi):xxx, xxx

Error Type: Disk functional out-of-band

Severity: Informational

Description: The drive positioned the heads in the wrong place or the HSC software is processing transfers out-of-order.

Action: Check drive modules and the K.sdi/K.si module.

Possible FRUs:

1. Drive modules . (Refer to the drive service manual.)
2. K.sdi/K.si module

Premature LP flag in RTNDAT sequence from host node xx

Error Type: Disk functional out-of-band

Severity: Warning

Description: A violation of packet protocol; the last packet flag was set before all data was received from a host.

Action: If the problem is transient, monitor error for repetitive node numbers as this may indicate a host CI problem. If the problem is persistent across all cluster nodes, test the K.ci.

Possible FRUs:

1. K.ci modules
2. CI cables

Pulse or Parity Error

Error Type: SDI error

Severity: Error

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.

Action: If the error is reported on more than one drive, a K.sdi/K.si 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 the drive service manual.)
2. SDI cable
3. SDI transition bulkhead
4. K.sdi/K.si module

RCT Corrupted Error

Error Type: Disk transfer error

Severity: Error

Description: The RCT search algorithm encountered an invalid RCT entry. The subcode may be returned under the following conditions:

During replacement of a block
During nonprimary revectoring of a block
When bringing a unit on line

Action: Determine if this error is repetitive for this unit possibly indicating a defective media or drive read path failure. Run the HSC utility VERIFY on the drive.

Possible FRUs: Drive modules. (Refer to the drive service manual.)

Receiver ready not asserted at start of transfer

Error Type: Tape error

Severity: Error

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.

Action: Check the formatter, cable, and K.sti/K.si.

Possible FRUs:

1. Formatter
2. Cable
3. K.sti/K.si module

Record EDC error

Error Type: Tape error

Severity: Error

Description: On a read from tape operation, the EDC calculated by the K.sti/K.si did not match the EDC generated by the tape formatter.

Action: Check the formatter, cable, and K.sti/K.si.

Possible FRUs:

1. Formatter
2. Cable
3. K.sti/K.si module

Requestor xx failed INIT diags, status = xxx

Error Type: SINI out-of-band

Severity: Error

Description: The data channel in the displayed requestor has failed initialization diagnostics with the displayed status.

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, K.sti, or K.si) module exhibiting the failing status.

Requestor xx has failed initialization diagnostics with status = xx

Error Type: Tape functional out-of-band

Severity: Error

Description: The requestor in slot xx has failed initialization diagnostics with the displayed status. The message indicates the failed K.sti/K.si module.

Action: Refer to Appendix C to determine what the displayed status indicates the failure to be.

Possible FRUs: The K.sti/K.si module in the indicated slot.

Reserved Instruction (Trap through 10)

From process yyyy
PC xxx
PSW xxx

Error Type: SINI out-of-band, subsystem exception

Severity: Error

Description: The P.ioj/c detected an Opcode, resulting in the execution of an invalid instruction. The process indicated is the process that executed the nonexistent instruction.

Action: Determine what process was active for module repair.

Possible FRUs:

1. P.ioj/c module
2. M.std2/M.std module
3. Software

Resource lost to K.ci—xxx xxx HMBs

Error Type: CI-detected out-of-band

Severity: Error

Description: A Control memory Host Message Block (HMB) data structure was lost. HMBs were expected in the sequence message ready to receive queue (.KHSRR), but none were found.

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. Main power supply

Retry limit exceeded while attempting to restore tape position

Error Type: Tape error

Severity: Error

Description: A command was issued to restore the tape position, and the command failed in the limit of retries.

Action: Check the formatter.

Possible FRUs: Formatter

Reverse retry currently not supported

NOTE
As of V3.50 and above, *Reverse Retry* is supported.

Error Type: Tape error

Severity: Error

Description: Reverse Retry requests from the formatter were not supported before Version 3.50 of HSC software.

Action: Update software

Possible FRUs: None

Rewind failure

Error Type: Tape error

Severity: Error

Description: A command for a rewind was issued, and the command failed (the controller received an unsuccessful response from the formatter).

Action: Check the drive and/or formatter.

Possible FRUs:

1. Drive modules. (Refer to the drive service manual.)
2. Formatter

SCT read or verification error. Using template SCT.

Error Type: SINI out-of-band

Severity: Error

Description: An error was detected by the P.ioj/c as it attempted to read the System Configuration Table (SCT) or as it attempted to verify the SCT. This error message will occur when new, previously uninitialized system diskette is 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. If the system has been previously booted from the same media, a system load device failure is indicated.

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 HSC to validate these values and ensure system continuity.

Possible FRUs: System diskette

SDI exchange retry on disk unit xxx. (Requestor xx. Port xx.)

DCB addr xx Error count xx.

Error Type: Disk functional out-of-band

Severity: Informational

Description: Retry the SDI command on the drive.

Action: None

Possible FRUs: None

SDI Clock Persisted after INIT

Error Type: SDI error

Severity: Error

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.

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 the drive service manual.)
2. SDI cable

SI Clock Resumption Failed after INIT

Error Type: SDI error

Severity: Error

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.

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 the drive service manual.)
2. SDI cable

SI Command Timeout

Error Type: SDI error

Severity: Error

Description: The controller timeout expired for either a level 2 exchange or the assertion of Read/Write Ready after an INITIATE SEEK command. 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.

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 the drive service manual.)
2. SDI cable

Ensure the drive and all HSC modules are at the latest revision levels.

SI Receiver Ready Collision

Error Type: SDI error

Severity: Error

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 reasserting Drive Receiver Ready, yet the controller has never received the response.

The HSC K.sdi/K.si 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.

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 the drive service manual.)
2. SDI cable
3. K.sdi/K.si module

SI Response Length or Opcode Error

Error Type: SDI error

Severity: Error

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 SDI 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/K.si 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.

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 the drive service manual.)
2. K.sdi/K.si module

SI Response Overflow

Error Type: SDI error

Severity: Error

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/K.si.

Action: Determine if the drive is failing in other ways, indicating a drive problem. If not, the K.sdi/K.si may be the more likely cause.

Possible FRUs:

1. Drive modules. (Refer to the drive service manual.)
2. K.sdi/K.si module

SERDES Overrun

Error Type: Controller error

Severity: Error

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.

Action: Determine if other errors have occurred that may indicate a K.sdi/K.si problem. Move the offending drive to another requestor. If the problem persists, test the drive further.

Possible FRUs: K.sdi/K.si module

Software inconsistency (Trap through 20)

Error Type: SINI out-of-band, subsystem exception

Severity: Error

Description: During operation, the operating software performs numerous consistency checks. When one of these consistency checks fails, the HSC crashes and reboots. The active process is displayed, as well as the stack dump.

Action: Submit a Software Problem Report (SPR). (Refer to Appendix B).

Possible FRUs: None

Tape drive requested error log

Error Type: Tape error

Severity: Warning

Description: The drive detected an error condition and set the EL bit for an error log to be taken.

Action: Check the drive.

Possible FRUs: Drive modules. (Refer to the drive service manual.)

Tape formatter connected to Requestor xx Port xx has been declared inoperative. Intervention required.

Error Type: Tape functional out-of-band

Severity: Error

Description: The K.sti/K.si has sent a nondata transfer command over the STI cable to the displayed tape formatter three times and has received back the same error three times. The HSC then ignores the tape formatter until it detects some intervention such as a change in the state clock.

Action: Replace the possible FRUs. Deasserting the tape drive's port switches, recycling power, unplugging the STI cable, or any action causing the state clock to come and go is considered an intervention. The HSC 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/K.si module

Tape unit number xx connected to Requestor xx Port xx ceased to exist while on line

Error Type: Tape functional out-of-band

Severity: Error

Description: This message is similar to the previous error message except in the case where the HSC was using the tape drive to do data transfers when the tape drive went off line.

Action: Check to see if a breaker has blown. The tape drive may be in testing mode also, causing the tape drive to go off line.

Possible FRUs:

1. Tape drive
2. Tape formatter
3. STI cable

Tape unit number xx connected to Requestor xx Port xx dropped state clock while on line

Error Type: Tape functional out-of-band

Severity: Error

Description: The formatter supplies the state clock over the STI cable. The state bits are encoded on this state clock waveform such as AVAILABLE and ATTENTION. As long as the K.sti/K.si is receiving a state clock, the STI cable must still be plugged in, and the formatter must be operating correctly. Dropping state clock is equivalent to disconnecting the STI cable from the HSC.

Action: First isolate the problem to the HSC, STI cable, or tape drive. Next, try replacing or swapping the K.sti/K.si module exhibiting the failure. If the problem is not solved, try a known good tape drive.

Possible FRUs:

1. STI cable
2. Tape drive
3. K.sti/K.si module

Tape unit number xx connected to Requestor xx Port xx is not asserting available when it should be

Error Type: Tape functional out-of-band

Severity: Error

Description: The formatter is not on line and is not asserting its Available signal to the HSC. The HSC does not detect the Available signal and displays this message on the local console terminal.

Action: First isolate the problem to either the HSC, the STI cable, or the tape drive. Next, try replacing or swapping the K.sti/K.si module exhibiting the failure. If the problem is not solved, try a known good tape drive.

Possible FRUs:

1. STI cable
2. Tape drive
3. K.sti/K.si module

Tape unit number xx connected to Requestor xx Port xx went available without request

Error Type: Tape functional out-of-band

Severity: Error

Description: When the formatter is on line, Available is not normally asserted to the HSC. When the formatter is on line and doing I/O and an Available is asserted, the HSC detects this as an error. A formatter does not need to send Available unless the K.sti/K.si requests it.

Action: First isolate the error to the formatter or to the active K.sti/K.si.

Possible FRUs:

1. K.sti/K.si module

2. Formatter
3. STI cable

Tape unit number xx connected to Requestor xx Port xx went off line without request

Error Type: Tape functional out-of-band

Severity: Error

Description: The formatter lost contact with one of the tape drives. The HSC detected this loss of a tape drive and printed this message.

Action: Check to see if a breaker has blown. The tape drive may be in diagnostic mode also, causing the tape drive go off line.

Possible FRUs:

1. Tape drive
2. Tape formatter
3. STI cable

TMSCP fatal initialization error—TMSCP functionality not available

Error Type: Tape functional out-of-band

Severity: Error

Description: Something went wrong during initialization with the tape functional code (TFUNCT). 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.

Action: Take action depending on the previously displayed 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.

Error Type: Tape functional out-of-band

Severity: Error

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/K.si modules and the number of tape formatters on the HSC.

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/M.std module
2. P.ioj/c module
3. Software

TOPOLOGY command failed

Error Type: Tape error

Severity: Error

Description: A TOPOLOGY command was issued and the command failed.

Action: Check the formatter.

Possible FRUs: Formatter

TTRASH fatal initialization error

Error Type: Tape functional out-of-band

Severity: Error

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).

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/M.std module
2. Software

Unable to position before LEOT

Error Type: Tape error

Severity: Error

Description: The command to position the tape was unable to complete before LEOT was detected.

Action: Check the drive.

Possible FRUs: Drive modules. (Refer to the drive service manual.)

Unclearable Drive Error

Error Type: Tape error

Severity: Error

Description: Issued a clear bit three times and the bit does not clear.

Action: Check the formatter and drive. Further analysis of tape drive error log may be necessary.

Possible FRUs:

1. Drive modules. (Refer to the drive service manual.)
2. Formatter
3. STI cable set
4. K.sti/K.si module

Unclearable Formatter Error

Error Type: Tape error

Severity: Error

Description: Issued a clear bit three times and the error does not clear.

Action: Check the formatter

Possible FRUs:

1. Formatter
2. STI cable set
3. K.sti/K.si module

Unexpected AVAILABLE signal from ONLINE disk unit xx.

Error Type: Disk functional out-of-band

Severity: Informational

Description: The disk is asserting AVAILABLE while the drive state is ONLINE. This is not an expected condition.

Action: Determine why the disk drive is asserting the Available signal.

Possible FRUs: Drive modules. (Refer to the drive service manual.)

Unit xx. declared inoperative because no progress made on Command Reference xxxxx.

Error Type: Disk functional out-of-band

Severity: Error

Description: The HSC Disk Path has made no progress on the host command represented by the given reference number in an extended time period. This scenario can occur if the drive is degraded to a point where the Disk Path spends too much time in error recovery and can make no progress on the host command.

Action: The HSC was unable to complete error recovery on the drive and took it off line. Check the drive with diagnostics to determine the nature of the problem.

Possible FRUs: Drive modules. (Refer to the drive service manual.)

Unknown K.tape error

Error Type: Tape error

Severity: Error

Description: The ER bit was set but was undefined.

Action: Check the formatter.

Possible FRUs: Formatter

Unrecoverable error on disk unit xx. Drive appears inoperative. Intervention required.

Error Type: Disk functional out-of-band

Severity: Error

Description: An error log message from the drive caused this message, or the drive may be off line. The Disk Path has concluded that the drive in unusable.

Action: Check the error log and drive.

Possible FRUs: Drive modules. (Refer to the drive service manual.)

Unsuccessful SEEK initiation, disk unit xxx. DCB addr: xxx

Error Type: Disk functional out-of-band

Severity: Informational

Description: The dialog control block sent the SEEK exchange and the DCB was sent to its error queue by the K.sdi/K.si. The SEEK may have been rejected, lost, or completed with an error.

Action: Check drive.

Possible FRUs: Drive modules. (Refer to the drive service manual.)

VC closed due to timeout of RTNDAT/CNF from host node xx

Error Type: Disk functional out-of-band

Severity: Informational

Description: The host issued a request over the CI, and the response timed out.

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

VC closed with node nn due to disconnect timeout

Error Type: CI-detected out-of-band

Severity: Warning

Description: A second disconnect call for the same connection block has been received by the CI Manager.

Action: Verify other cluster nodes have not failed or have CI port problems. If the problem persists, run Off-line 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

Error Type: CI-detected out-of-band

Severity: Warning

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.**

Action: Set error and outband to info. See the descriptions and action for the following error messages:

Node nn path (A or B) has gone from bad to good
Node nn path (A or B) has gone from good to bad

Possible FRUs:

1. K.ci hardware interface in HSC
2. CI cables
3. Host CI hardware

VC closed with node nn due to START received

Error Type: CI-detected out-of-band

Severity: Warning

Description: A start message is received over the CI to an already open Virtual Circuit (VC).

Action: Check for two HSCs with the same ID (not node address) on the cluster. This happens when a new HSC is installed on the cluster and is given an existing ID.

Possible FRUs: CI cables

VC closed with node nn due to unexpected disconnect

Error Type: CI-detected out-of-band

Severity: Warning

Description: The HSC receives a DISCONNECT_REQ packet, and the following conditions exist inside the HSC.

- A connection is not open.
- The HSC is not in the DISCONNECT_SENT state. (The DISCONNECT_SENT state indicates the HSC also sent a DISCONNECT_REQ packet.)

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 Off-line Test K diagnostic to test K.ci. If no failure, verify no duplicate node addresses exist in this cluster with the LINK module (L0100\L0118) node address switches.

Possible FRUs: K.pli module

VC open with node nn

Error Type: CI-detected out-of-band

Severity: Informational

Description: A Virtual Circuit (VC) has been established with the given node. The Online lamp on the HSC Operator Control Panel lights the first time a VC is established to an HSC.

Action: None is required; this message is for informational purposes only.

Possible FRUs: None

WARNING K.sti microcode too low for large transfers.

Error Type: Tape functional out-of-band

Severity: Warning

Description: The amount of I/O the K.sti can accommodate is restricted. The code still attempts to do transfers, but a warning has been issued.

Action: Update the microcode version level to 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.

Word rate clock timeout

Error Type: Tape error

Severity: Error

Description: The K.sti/K.si detected the loss of clocks from a drive during a transfer.

Action: Check the formatter and the cable.

Possible FRUs:

1. Formatter
2. Cable

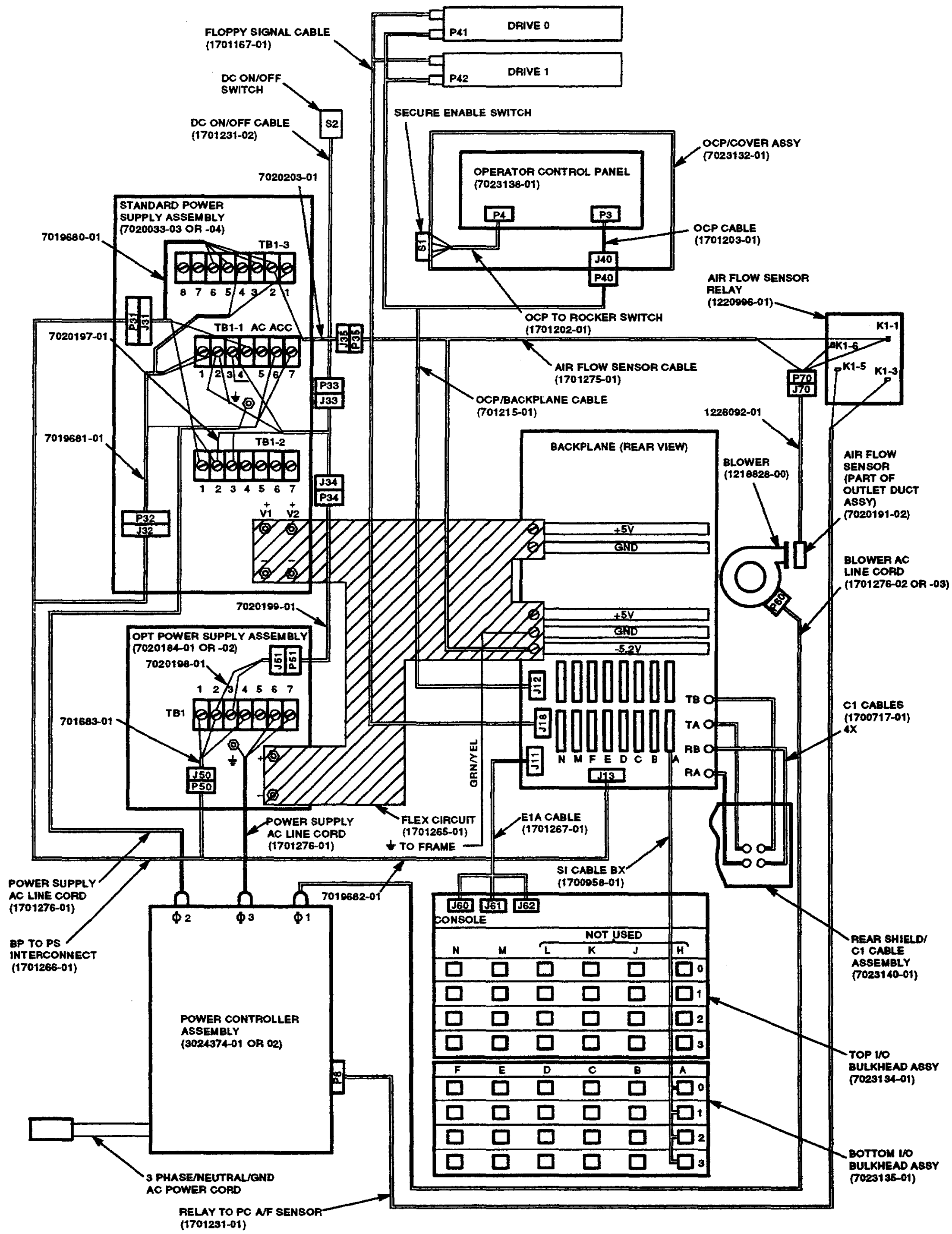
A
Internal Cabling Diagrams

A.1 Introduction

This appendix contains diagrams of the internal cabling for the HSC, HSC50 (modified), and HSC50.

A.2 HSC Internal Cabling

Figure A–1 is a diagram of the HSC internal cabling.



CXO-944A
Sheet 1 of 5

Figure A–1 (Cont.) HSC Internal Cabling

WIRE TABLE FOR 1701202-01 OCP TO ROCKER SWITCH				
COLOR	FROM	TO	SIGNAL	REMARKS
	P4-01		NO CONNECTION	KEYING PLUG
	P4-02		NO CONNECTION	
RED	P4-03	S1-3	+5 VOLT	
BLACK	P4-04	S1-6	GND (-5 VOLT)	
	P4-05		NO CONNECTION	SPARE
YELLOW	P4-06	S1-4	GND	
YELLOW	P4-07	S1-5	TERM ENABLE	
	P4-08		NO CONNECTION	SPARE
WHITE	P4-09	S1-1	INIT SWL	
WHITE	P4-10	S1-2	INIT L	

WIRE TABLE FOR 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 FOR 1701215-01 OCP/BACKPLANE				
COLOR	FROM	TO	OCP SIGNAL	REMARKS
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	
YEL/BLK/GRY	J12-11	P40-11	PANEL SWITCH 3 L	
YEL/GRN/ORG	J12-12	P40-12	PANEL SWITCH 2 L	
YEL/RED/WHT	J12-13	P40-13	BDCOK H (INIT L)	
BLACK	J12-14	P40-14	GND	
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	GND	
BLACK	J12-22	P41-03	GND	
BLACK	J12-23	P42-02	GND	
BLACK	J12-24	P42-03	GND	
VIOLET	J12-25	P41-01	+12 VOLTS	
VIOLET	J12-26	P42-01	+12 VOLTS	

CXO-944A
Sheet 2 of 5

Figure A–1 (Cont.) HSC Internal Cabling

WIRE TABLE FOR 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 FOR 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 (+5.2V)	
BLACK	S2-3	P33-1	S1-	

WIRE TABLE FOR 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		
BLACK	J13-5	P31-9	GND (+12V)		
BLACK	J13-6	P31-2	GND (+12V)		
BLACK	J13-7	P31-4	GND (+12V)	DOUBLE CRIMPED	
BLACK	J13-8		GND (+12V)		
ORANGE	J13-9	P31-6	-5.2V SENSE	TWISTED PAIR	
BLACK	J13-10	P31-8	GND (-5V SENSE)		
BROWN	J13-11	P31-10	POWER FAIL L		
BLACK	J13-14	J32-1	GND (+5V SENSE)	TWISTED PAIR	
RED	J13-13	J32-2	+5V SENSE		
BLACK	J13-16	J32-3	GND (+12V SENSE)	TWISTED PAIR	
VIOLET	J13-15	J32-4	+12V SENSE		
BLACK	J13-17	P50-2	GND (+5V SENSE)	TWISTED PAIR	OPTIONAL POWER SUPPLY
RED	J13-18	P50-1	+5V SENSE		
BROWN	J13-20	P50-3	POWER FAIL L		

WIRE TABLE FOR 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-	

CXO-944A
Sheet 3 of 5

Figure A–1 (Cont.) HSC Internal Cabling

WIRE TABLE FOR 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 FOR 1701275-01 A/F SENSOR CABLE				
COLOR	FROM	TO	SIGNAL	REMARKS
VIOLET	P70-1	K1-1	+5V	DOUBLE CRIMP
VIOLET	P35			
ORANGE	P70-2	K106	LOAD (-5V)	
ORANGE	P70-3	-5.2V BUSBAR 0 BACKPLANE	-5.2V	

WIRE TABLE FOR 1701276-01 STD POWER SUPPLY				
COLOR	FROM	TO	SIGNAL	REMARKS
GRN/YEL	GND STUD	ϕ 2 *	GND	*POWER CONTROLLER ϕ 2
BLUE	TB1-1-7	ϕ 2 *	ACC	
BROWN	TB1-1-6	ϕ 2 *	AC	

WIRE TABLE FOR 1701276-01 OPT POWER SUPPLY				
COLOR	FROM	TO	SIGNAL	REMARKS
GRN/YEL	GND STUD	ϕ 3 *	GND	*POWER CONTROLLER ϕ 3
BLUE	TB1-1-7	ϕ 3 *	ACC	
BROWN	TB1-1-6	ϕ 3 *	AC	

WIRE TABLE FOR 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	GND	
BLACK	P80-5	P80-4		JUMPER

WIRE TABLE FOR 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	GND	
BLACK	P80-7	P80-4		JUMPER
BLACK	P80-8	P80-5		JUMPER

WIRE TABLE FOR 7019680-01				
COLOR	FROM	TO	SIGNAL	REMARKS
VIOLET	J31-1	TB1-3-5	+12V	DOUBLE CRIMP
VIOLET	J31-3			
VIOLET	J31-5	TB1-3-6	+12V	DOUBLE CRIMP
VIOLET	J31-7			
BLACK	J31-9	TB1-3-3	GND (+12V)	DOUBLE CRIMP
BLACK	J31-2			
BLACK	J31-4	TB1-3-3	GND (+12V)	
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	

CXO-944A
Sheet 4 of 5

Figure A–1 (Cont.) HSC Internal Cabling

WIRE TABLE FOR 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 FOR 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	TB1-3	POWER FAIL	

WIRE TABLE FOR 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	SI-	DOUBLE CRIMP
BLACK	J34-1			

WIRE TABLE FOR 7020198-01				
COLOR	FROM	TO	SIGNAL	REMARKS
BLUE	J51-2	TB1-3	ON/OFF (+5V)	
BLACK	J51-1	TB1-2	S-	

WIRE TABLE FOR 7020199-01				
COLOR	FROM	TO	SIGNAL	REMARKS
BLUE	P34-1	J51-1	S-	
BLACK	P34-2	P51-2	ON/OFF (+5V)	

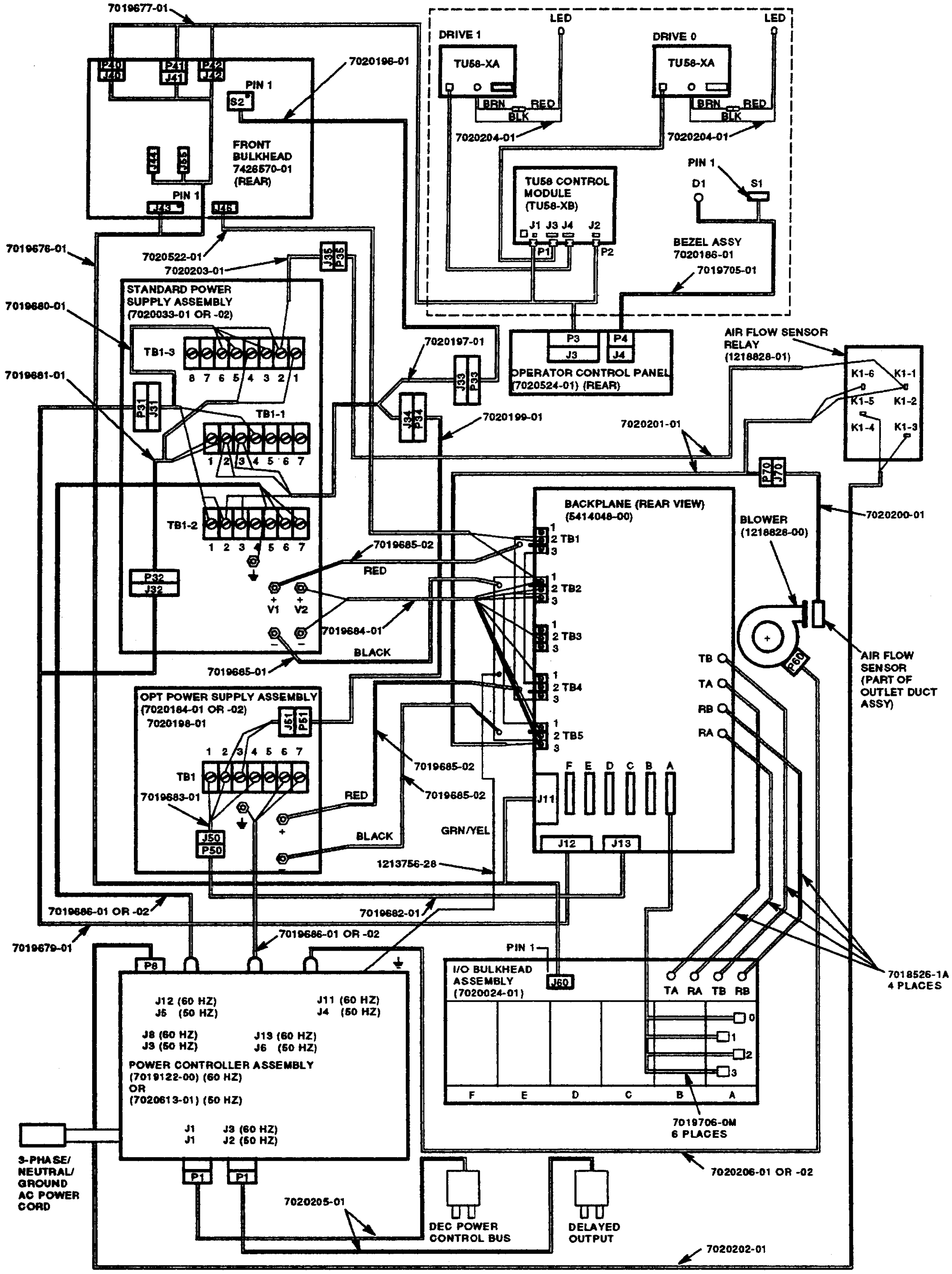
WIRE TABLE FOR 7020203-01				
COLOR	FROM	TO	SIGNAL	REMARKS
VIOLET	J35	TB1-3-2	+12V	

CXO-944A
Sheet 5 of 5

Figure A–1 HSC Internal Cabling

A.3 HSC50 Internal Cabling

Figure A–2 is a diagram of the HSC50 internal cabling.



CXO-051A
Sheet 1 of 6

Figure A–2 (Cont.) HSC50 Internal Cabling

WIRE TABLE FOR LINE CORD OF 7019122-00			
COLOR	FROM	TO	REMARKS
BLUE	W	LF1-N	
BLACK	Z	LF1-L3	
GRN/YEL	GND	LF1-GND	LINE SIDE
BLACK	Y	LF1-L2	
BROWN	X	LF1-L1	

WIRE TABLE FOR 7019676-01				
COLOR	FROM	TO	SIGNAL	REMARKS
VIOLET	J11-1	J41-01	+12V	
VIOLET	J11-2	J44-01	+12V	
BLACK	J11-3	J41-02	GROUND	
BLACK	J11-4	J41-03	GROUND	
BLACK	J11-5	J44-02	GROUND	
BLACK	J11-6	J44-03	GROUND	
RED	J11-7	J41-04	+5V	
RED	J11-8	J44-04	+5V	
WHITE	J11-12	J60-20	TERM PRESS L	
WHITE	J11-13	J60-01	XMT-	
WHITE	J11-14	J60-02	XMT+	
WHITE	J11-15	J60-03	RCV+	
	J60-20	J60-06	DATA SET READY COMMONING STRIP SEE NOTE *	
WHITE	J11-16	J60-07	ACV-	* PINS AT J43-06 AND J60-06 ARE WIRELESS PINS. THEY ARE TIED TO J43-20 AND J60-20 BY COMMONING STRIPS.
WHITE	J11-18	J43-20	TERM PRESS L	
WHITE	J11-19	J43-01	XMT-	
WHITE	J11-20	J43-02	XMT+	
WHITE	J11-21	J43-03	RCV+	
	J43-20	J43-06	DATA SET READY COMMONING STRIP SEE NOTE *	
WHITE	J11-22	J43-07	RCV-	
WHITE	J11-26	J42-01	TU0/1 PRESS L	
WHITE	J11-27	J42-02	TU0/1 XMT-	
WHITE	J11-28	J42-03	TU0/1 XMT+	
WHITE	J11-29	J42-04	TU0/1 REV+	
WHITE	J11-30	J42-05	TU0/1 RCV+	
WHITE	J11-34	J45-01	TU2/3 PRESS L	
WHITE	J11-35	J45-02	TU2/3 XMT-	
WHITE	J11-36	J45-03	TU2/3 XMT+	
WHITE	J11-37	J45-04	TU2/3 RCV+	
WHITE	J11-38	J45-05	TU2/3 RCV-	
YELLOW	J11-45	J40-01	STATE LAMP L	
YELLOW	J11-46	J40-02	POWER ON L	
YELLOW	J11-47	J40-03	LAMP 0 L	
YELLOW	J11-48	J40-04	TERM ENA L	
YELLOW	J11-49	J40-05	LAMP 2 L	
YELLOW	J11-50	J40-06	LAMP 1 L	
YELLOW	J11-51	J40-07	LAMP 4 L	
YELLOW	J11-52	J40-08	LAMP 3 L	
YELLOW	J11-53	J40-09	SWITCH 1 L	
YELLOW	J11-54	J40-10	SWITCH 0 L	
YELLOW	J11-55	J40-11	SWITCH 3 L	
YELLOW	J11-56	J40-12	SWITCH 2 L	
YELLOW	J11-57	J40-13	BDCOK H (INT L)	
BLACK	J11-58	J40-14	GROUND	
RED	J11-60	J40-15	+5V	

CXO-051A
Sheet 2 of 6

Figure A–2 (Cont.) HSC50 Internal Cabling

WIRE TABLE FOR 7019677-01				
COLOR	FROM	TO	SIGNAL	REMARKS
VIOLET	P41-1	P1-1	+12V	TU POWER
BLACK	P41-2	P1-3	GND (+12V)	TU POWER
BLACK	P41-3	P1-6	GND (+5V)	TU POWER
READ	P41-4	P1-5	+5V	TU POWER
WHITE	P42-1	P2-F	GND	TU SIGNAL
WHITE	P42-2	P2-D	RCV-	TU SIGNAL
WHITE	P42-3	P2-C	RCV+	TU SIGNAL
WHITE	P42-4	P2-J	XMT+	TU SIGNAL
WHITE	P42-5	P2-H	XMT-	TU SIGNAL
		P2-E		KEYING PLUG (TU SIG)
YELLOW	P40-1	P3-1	STATE LAMP L	OCP
YELLOW	P40-2	P3-2	POWER ON L	OCP
YELLOW	P40-3	P3-4	LAMP 0 L	OCP
YELLOW	P40-4	P3-3	TERM ENA L	OCP
YELLOW	P40-5	P3-6	LAMP 2 L	OCP
YELLOW	P40-6	P3-5	LAMP 1 L	OCP
YELLOW	P40-7	P3-8	LAMP 4 L	OCP
YELLOW	P40-8	P3-7	LAMP 3 L	OCP
YELLOW	P40-9	P3-10	SWITCH 1 L	OCP
YELLOW	P40-10	P3-9	SWITCH 0 L	OCP
YELLOW	P40-11	P3-12	SWITCH 3 L	OCP
YELLOW	P40-12	P3-11	SWITCH 2 L	OCP
YELLOW	P40-13	P3-15	BDCOKH (INIT L)	OCP
BLACK	P40-14	P3-14	GND	OCP
RED	P40-15	P3-16	+5V	OCP
		P3-20		KEYING PLUG (OCP)

WIRE TABLE FOR 7019679-01				
COLOR	FROM	TO	SIGNAL	REMARKS
VIOLET	J12-1	P31-1	+12V	
VIOLET	J12-2	P31-3	+12V	
VIOLET	J12-3	P31-5	+12V	
VIOLET	J12-4	P31-7	+12V	
BLACK	J12-5	P31-9	GND (+12V)	
BLACK	J12-7	P31-2	GND (+12V)	
BLACK	J12-9	P31-4	GND (+12V)	
ORANGE	J12-11	P31-6	-5V SENSE	TWISTED
BLACK	J12-12	P31-8	GND (-5V SENSE)	PAIR
BROWN	J12-13	P31-10	POWER FAIL	
	J12-16		NO CONNECTION	KEYING PLUG
BLACK	J12-17	J32-1	GND (+5V SENSE)	TWISTED
RED	J12-18	J32-2	+5V SENSE	PAIR
BLACK	J12-19	J32-3	GND (+12V SENSE)	TWISTED
VIOLET	J12-20	J32-4	+12V SENSE	PAIR

CXO-051A
Sheet 3 of 6

Figure A–2 (Cont.) HSC50 Internal Cabling

WIRE TABLE FOR 7019680-01				
COLOR	FROM	TO	SIGNAL	REMARKS
VIOLET	J31-1	TB1-3-5	+12V	DOUBLE CRIMP
VIOLET	J31-3			
VIOLET	J31-5	TB1-3-6	+12V	DOUBLE CRIMP
VIOLET	J31-7			
BLACK	J31-9	TB1-3-3	GND (+12V)	DOUBLE CRIMP
BLACK	J31-2			
BLACK	J31-4	TB1-3-3	GND (+12V)	
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 FOR 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 FOR 7019686-02 STD PWR SUPPLY (50 HZ)				
COLOR	FROM	TO	SIGNAL	REMARKS
GRN/YEL	GND STUD	J5*	GND	*POWER CONTROLLER J5
BLUE	TB1-2-7	J5*	ACC	
BROWN	TB1-2-6	J5*	AC	

WIRE TABLE FOR 7019686-01 AUX PWR SUPPLY (60 HZ)				
COLOR	FROM	TO	SIGNAL	REMARKS
GRN/YEL	GND STUD	J13*	GND	*POWER CONTROLLER J13
BLUE	TB1-7	J13*	ACC	
BROWN	TB1-6	J13*	AC	

WIRE TABLE FOR 7019686-02 AUX PWR SUPPLY (50 HZ)				
COLOR	FROM	TO	SIGNAL	REMARKS
GRN/YEL	GND STUD	J6*	GND	*POWER CONTROLLER J6
BLUE	TB1-7	J6*	ACC	
BROWN	TB1-6	J6*	AC	

WIRE TABLE FOR 7019705-01				
COLOR	FROM	TO	SIGNAL	REMARKS
	P4-01		NO CONNECTION	SPARE
	P4-02		NO CONNECTION	KEYING PLUG
RED	P4-03	D1-1	+5V	
BLACK	P4-04	D1-2	GND (+5V)	
	P4-05		NO CONNECTION	SPARE
YELLOW	P4-06	S1-4	GND	
YELLOW	P4-07	S1-5	TERM ENABLE	
	P4-08		NO CONNECTION	SPARE
WHITE	P4-09	S1-1	INIT SWL	
WHITE	P4-10	S1-2	INIT L	

CXO-051A
Sheet 4 of 6

Figure A–2 (Cont.) HSC50 Internal Cabling

WIRE TABLE FOR 7020196-01				
COLOR	FROM	TO	SIGNAL	REMARKS
YELLOW	S2-2	P33-4	ON/OFF (-5.3V)	
ORANGE	S2-1	P33-3	S2-	
BLUE	S2-4	P33-2	ON/OFF (+5V)	
BLACK	S2-5	P33-1	S1-	

WIRE TABLE FOR 7019682-01				
COLOR	FROM	TO	SIGNAL	REMARKS
	J13-2			KEYING PLUG
RED	J13-3	P50-1	+5V SENSE	TWISTED PAIR
BLACK	J13-4	P50-2	GND (+5V SENSE)	
BROWN	J13-5	P50-3	POWER FAIL	

WIRE TABLE FOR 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	TB1-3	POWER FAIL	

WIRE TABLE FOR 7019684-01				
COLOR	FROM	TO	SIGNAL	REMARKS
BLACK	+V2	TB1-1	GROUND (-5V)	
BLACK	+V2	TB1-1	GROUND (-5V)	
BLACK	+V2	TB3-1	GROUND (-5V)	
BLACK	+V2	TB3-3	GROUND (-5V)	
ORANGE	-V2	TB2-3	-5V	
ORANGE	-V2	TB3-2	-5V	
ORANGE	-V2	TB4-1	-5V	
ORANGE	-V2	TB5-3	-5V	

WIRE TABLE FOR 7019686-01 STD PWR SUPPLY (60 HZ)				
COLOR	FROM	TO	SIGNAL	REMARKS
GRN/YEL	GND STUD	J12*	GND	*POWER CONTROLLER J12
BLUE	TB1-2-7	J12*	ACC	
BROWN	TB1-2-6	J12*	AC	

WIRE TABLE FOR 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	SI-	DOUBLE CRIMP
BLACK	J34-1			

WIRE TABLE FOR 7020198-01				
COLOR	FROM	TO	SIGNAL	REMARKS
BLUE	J51-2	TB1-3	ON/OFF (+5V)	
BLACK	J51-1	TB1-2	S-	

WIRE TABLE FOR 7020199-01				
COLOR	FROM	TO	SIGNAL	REMARKS
BLUE	P34-1	J51-1	S-	
BLACK	P34-2	P51-2	ON/OFF (+5V)	

CXO-051A
Sheet 5 of 6

Figure A–2 (Cont.) HSC50 Internal Cabling

WIRE TABLE FOR 7020200-01				
COLOR	FROM	TO	SIGNAL	REMARKS
RED	A1-+	J70-1		
BLACK	A1-GND	J70-3		
WHITE	A1-LOAD	J70-2		

WIRE TABLE FOR 7020201-01				
COLOR	FROM	TO	SIGNAL	REMARKS
VIOLET	P70-1	K1-1	+12V	DOUBLE CRIMP
VIOLET	P35			
ORANGE	P70-2	K1-6	LOAD (-5V)	
ORANGE	P70-3	TB5-3	-5V	

WIRE TABLE FOR 7020202-01				
COLOR	FROM	TO	SIGNAL	REMARKS
WHITE	K1-3	P8-1	TRIP	
WHITE	K1-5	P8-2	RETURN	

WIRE TABLE FOR 7020203-01				
COLOR	FROM	TO	SIGNAL	REMARKS
VIOLET	J35	TB1-3-2	+12V	

WIRE TABLE FOR 7020206-01 (60 HZ)				
COLOR	FROM	TO	SIGNAL	REMARKS
BLUE	J11*	P80-1	AC	* POWER CONTROLLER J11
BROWN		P80-2	AC	
GRN/YEL		P80-3	GROUND	
BLACK	P80-7	P80-4		JUMPER
BLACK	P80-6	P80-5		JUMPER

WIRE TABLE FOR 7020206-02 (50 HZ)				
COLOR	FROM	TO	SIGNAL	REMARKS
BLUE	S2-2	P80-1	AC	* POWER CONTROLLER J4
BROWN		P80-2	AC	
GRN/YEL		P80-3	GROUND	
BLACK	P80-5	P80-4		JUMPER

WIRE TABLE FOR 7020522-01				
COLOR	FROM	TO	SIGNAL	REMARKS
RED	TB1-2	J46-2	+5V	FROM BACKPLANE
BLACK	TB2-1	J46-1	GROUND	FROM BACKPLANE

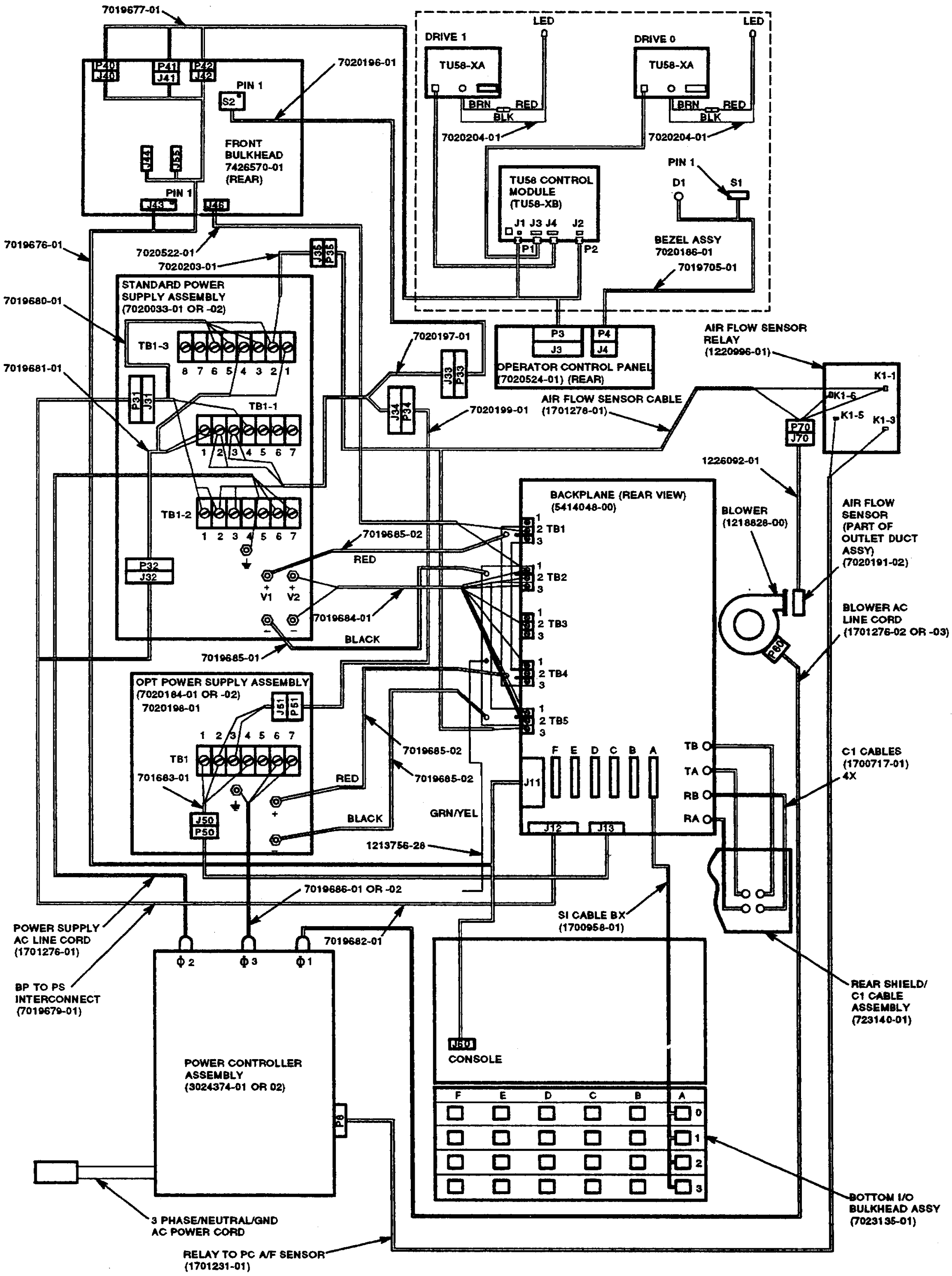
WIRE TABLE FOR LINE CORD OF 7020613-01				
COLOR	FROM	TO	SIGNAL	REMARKS
BROWN	X	LF1-PH3		LINE SIDE OF LF1
BLACK	Y	LF1-PH2		
BLACK (2)	Z	LF1-PH1		
BLUE	N	LF1-N		
GRN/YEL	GND	LF1-GND		

CXO-051A
Sheet 6 of 6

Figure A–2 HSC50 Internal Cabling

A.4 HSC50 (Modified) Internal Cabling

Figure A–3 is a diagram of the HSC50 (modified) internal cabling.



CXO-2076A
Sheet 1 of 6

Figure A–3 (Cont.) HSC50 (Modified) Internal Cabling

WIRE TABLE FOR 7019676-01				
COLOR	FROM	TO	SIGNAL	REMARKS
VIOLET	J11-1	J41-01	+12V	
VIOLET	J11-2	J44-01	+12V	
BLACK	J11-3	J41-02	GROUND	
BLACK	J11-4	J41-03	GROUND	
BLACK	J11-5	J44-02	GROUND	
BLACK	J11-6	J44-03	GROUND	
RED	J11-7	J41-04	+5V	
RED	J11-8	J44-04	+5V	
WHITE	J11-12	J60-20	TERM PRESS L	
WHITE	J11-13	J60-01	XMT-	
WHITE	J11-14	J60-02	XMT+	
WHITE	J11-15	J60-03	RCV+	
	J60-20	J60-06	DATA SET READY COMMONING STRIP SEE NOTE *	
WHITE	J11-16	J60-07	ACV-	* PINS AT J43-06 AND J60-06 ARE WIRELESS PINS. THEY ARE TIED TO J43-20 AND J60-20 BY COMMONING STRIPS.
WHITE	J11-18	J43-20	TERM PRESS L	
WHITE	J11-19	J43-01	XMT-	
WHITE	J11-20	J43-02	XMT+	
WHITE	J11-21	J43-03	RCV+	
	J43-20	J43-06	DATA SET READY COMMONING STRIP SEE NOTE *	
WHITE	J11-22	J43-07	RCV-	
WHITE	J11-26	J42-01	TU0/1 PRESS L	
WHITE	J11-27	J42-02	TU0/1 XMT-	
WHITE	J11-28	J42-03	TU0/1 XMT+	
WHITE	J11-29	J42-04	TU0/1 REV+	
WHITE	J11-30	J42-05	TU0/1 RCV+	
WHITE	J11-34	J45-01	TU2/3 PRESS L	
WHITE	J11-35	J45-02	TU2/3 XMT-	
WHITE	J11-36	J45-03	TU2/3 XMT+	
WHITE	J11-37	J45-04	TU2/3 RCV+	
WHITE	J11-38	J45-05	TU2/3 RCV-	
YELLOW	J11-45	J40-01	STATE LAMP L	
YELLOW	J11-46	J40-02	POWER ON L	
YELLOW	J11-47	J40-03	LAMP 0 L	
YELLOW	J11-48	J40-04	TERM ENA L	
YELLOW	J11-49	J40-05	LAMP 2 L	
YELLOW	J11-50	J40-06	LAMP 1 L	
YELLOW	J11-51	J40-07	LAMP 4 L	
YELLOW	J11-52	J40-08	LAMP 3 L	
YELLOW	J11-53	J40-09	SWITCH 1 L	
YELLOW	J11-54	J40-10	SWITCH 0 L	
YELLOW	J11-55	J40-11	SWITCH 3 L	
YELLOW	J11-56	J40-12	SWITCH 2 L	
YELLOW	J11-57	J40-13	BDCOK H (INT L)	
BLACK	J11-58	J40-14	GROUND	
RED	J11-60	J40-15	+5V	

CXO-2076A
Sheet 2 of 6

Figure A–3 (Cont.) HSC50 (Modified) Internal Cabling

WIRE TABLE FOR 7019677-01				
COLOR	FROM	TO	SIGNAL	REMARKS
VIOLET	P41-1	P1-1	+12V	TU POWER
BLACK	P41-2	P1-3	GND (+12V)	TU POWER
BLACK	P41-3	P1-6	GND (+5V)	TU POWER
READ	P41-4	P1-5	+5V	TU POWER
WHITE	P42-1	P2-F	GND	TU SIGNAL
WHITE	P42-2	P2-D	RCV-	TU SIGNAL
WHITE	P42-3	P2-C	RCV+	TU SIGNAL
WHITE	P42-4	P2-J	XMT+	TU SIGNAL
WHITE	P42-5	P2-H	XMT-	TU SIGNAL
		P2-E		KEYING PLUG (TU SIG)
YELLOW	P40-1	P3-1	STATE LAMP L	OCP
YELLOW	P40-2	P3-2	POWER ON L	OCP
YELLOW	P40-3	P3-4	LAMP 0 L	OCP
YELLOW	P40-4	P3-3	TERM ENA L	OCP
YELLOW	P40-5	P3-6	LAMP 2 L	OCP
YELLOW	P40-6	P3-5	LAMP 1 L	OCP
YELLOW	P40-7	P3-8	LAMP 4 L	OCP
YELLOW	P40-8	P3-7	LAMP 3 L	OCP
YELLOW	P40-9	P3-10	SWITCH 1 L	OCP
YELLOW	P40-10	P3-9	SWITCH 0 L	OCP
YELLOW	P40-11	P3-12	SWITCH 3 L	OCP
YELLOW	P40-12	P3-11	SWITCH 2 L	OCP
YELLOW	P40-13	P3-15	BDCOKH (INIT L)	OCP
BLACK	P40-14	P3-14	GND	OCP
RED	P40-15	P3-16	+5V	OCP
		P3-20		KEYING PLUG (OCP)

WIRE TABLE FOR 7019679-01				
COLOR	FROM	TO	SIGNAL	REMARKS
VIOLET	J12-1	P31-1	+12V	
VIOLET	J12-2	P31-3	+12V	
VIOLET	J12-3	P31-5	+12V	
VIOLET	J12-4	P31-7	+12V	
BLACK	J12-5	P31-9	GND (+12V)	
BLACK	J12-7	P31-2	GND (+12V)	
BLACK	J12-9	P31-4	GND (+12V)	
ORANGE	J12-11	P31-6	-5V SENSE	TWISTED PAIR
BLACK	J12-12	P31-8	GND (-5V SENSE)	
BROWN	J12-13	P31-10	POWER FAIL	
	J12-16		NO CONNECTION	KEYING PLUG
BLACK	J12-17	J32-1	GND (+5V SENSE)	TWISTED PAIR
RED	J12-18	J32-2	+5V SENSE	
BLACK	J12-19	J32-3	GND (+12V SENSE)	TWISTED PAIR
VIOLET	J12-20	J32-4	+12V SENSE	

CXO-2076A
Sheet 3 of 6

Figure A–3 (Cont.) HSC50 (Modified) Internal Cabling

WIRE TABLE FOR 7019680-01				
COLOR	FROM	TO	SIGNAL	REMARKS
VIOLET	J31-1	TB1-3-5	+12V	DOUBLE CRIMP
VIOLET	J31-3			
VIOLET	J31-5	TB1-3-6	+12V	DOUBLE CRIMP
VIOLET	J31-7			
BLACK	J31-9	TB1-3-3	GND (+12V)	DOUBLE CRIMP
BLACK	J31-2			
BLACK	J31-4	TB1-3-3	GND (+12V)	
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 FOR 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 FOR 7019705-01				
COLOR	FROM	TO	SIGNAL	REMARKS
	P4-01		NO CONNECTION	SPARE
	P4-02		NO CONNECTION	KEYING PLUG
RED	P4-03	D1-1	+5V	
BLACK	P4-04	D1-2	GND (+5V)	
	P4-05		NO CONNECTION	SPARE
YELLOW	P4-06	S1-4	GND	
YELLOW	P4-07	S1-5	TERM ENABLE	
	P4-08		NO CONNECTION	SPARE
WHITE	P4-09	S1-1	INIT SWL	
WHITE	P4-10	S1-2	INIT L	

WIRE TABLE FOR 7020196-01				
COLOR	FROM	TO	SIGNAL	REMARKS
YELLOW	S2-2	P33-4	ON/OFF (-5.3V)	
ORANGE	S2-1	P33-3	S2-	
BLUE	S2-4	P33-2	ON/OFF (+5V)	
BLACK	S2-5	P33-1	S1-	

WIRE TABLE FOR 7019682-01				
COLOR	FROM	TO	SIGNAL	REMARKS
	J13-2			KEYING PLUG
RED	J13-3	P50-1	+5V SENSE	TWISTED PAIR
BLACK	J13-4	P50-2	GND (+5V SENSE)	
BROWN	J13-5	P50-3	POWER FAIL	

WIRE TABLE FOR 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	TB1-3	POWER FAIL	

CXO-2076A
Sheet 4 of 6

Figure A–3 (Cont.) HSC50 (Modified) Internal Cabling

WIRE TABLE FOR 7019684-01				
COLOR	FROM	TO	SIGNAL	REMARKS
BLACK	+V2	TB2-1	GRN (-5V)	
BLACK	+V2	TB2-2	GND (-5V)	
BLACK	+V2	TB5-1	GND (-5V)	
BLACK	+V2	TB5-2	GND (-5V)	
ORANGE	-V2	TB2-3	-5V	
ORANGE	-V2	TB3-2	-5V	
ORANGE	-V2	TB4-1	-5V	
ORANGE	-V2	TB5-3	-5V	

WIRE TABLE FOR 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)	
BLUE	J34-2			
BLACK	J33-1	TB1-1-2	S1-	
BLACK	J34-1			

WIRE TABLE FOR 7020198-01				
COLOR	FROM	TO	SIGNAL	REMARKS
BLUE	J5-2	TB1-3	ON/OFF (+5V)	
BLACK	J5-1	TB1-2	S-	

WIRE TABLE FOR 7020199-01				
COLOR	FROM	TO	SIGNAL	REMARKS
BLACK	P34-1	P51-1	S-	
BLUE	P34-2	P51-2	ON/OFF (+5V)	

WIRE TABLE FOR 1228092-01 A/F SENSOR				
COLOR	FROM	TO	SIGNAL	REMARKS
RED	A1-+	J70-1		
BLACK	A1-GND	J70-2		
WHITE	A1-LOAD	J70-3		

WIRE TABLE FOR 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 FOR 1701275-01 A/F SENSOR CABLE				
COLOR	FROM	TO	SIGNAL	REMARKS
VIOLET	P70-1	K1-1	+5V	DOUBLE CRIMP
VIOLET	P35			
ORANGE	P70-2	K106	LOAD (-5V)	
ORANGE	P70-3	-5.2V BUSBAR 0 BACKPLANE	-5.2V	

WIRE TABLE FOR 1701276-01 STD POWER SUPPLY				
COLOR	FROM	TO	SIGNAL	REMARKS
GRN/YEL	GND STUD	ϕ 2 *	GND	*POWER CONTROLLER ϕ 2
BLUE	TB1-1-7	ϕ 2 *	ACC	
BROWN	TB1-1-6	ϕ 2 *	AC	

CXO-2076A
Sheet 5 of 6

Figure A–3 (Cont.) HSC50 (Modified) Internal Cabling

WIRE TABLE FOR 1701276-01 OPT POWER SUPPLY				
COLOR	FROM	TO	SIGNAL	REMARKS
GRN/YEL	GND STUD	ϕ 3 *	GND	*POWER CONTROLLER ϕ 3
BLUE	TB1-1-7	ϕ 3 *	ACC	
BROWN	TB1-1-6	ϕ 3 *	AC	

WIRE TABLE FOR 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	GND	
BLACK	P80-5	P80-4		JUMPER

WIRE TABLE FOR 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	GND	
BLACK	P80-7	P80-4		JUMPER
BLACK	P80-8	P80-5		JUMPER

CXO-2076A
Sheet 6 of 6

Figure A–3 HSC50 (Modified) Internal Cabling

B
Exception Codes and Messages

Certain software inconsistencies can cause an exception (crash) in the HSC. This appendix describes all HSC exception codes caused by software inconsistencies. It provides a description of the exception codes, the facility or program reporting, and the action you should take. For ease of reference, these codes are arranged in numerical order (octal radix).

To determine which exception code caused a particular crash, refer to the crash dump printed on the terminal. Note that the code number, but not the text, appears on hardcopy printouts.

B.1 Crash Dump Printout

In order to determine which exception code caused a particular crash, refer to the crash dump printed on the terminal. The following HSC crash dump example breaks down and describes the various fields. The HSCxx refers to the HSC model.

```
-*  SUBSYSTEM EXCEPTION  *-      V100           HSCxx  HSC001 ❶
         at 17-Nov-1858 00:13:34.20        up    0 00:13:34.20
User ❷  Pc: 015066 caused by (20      ) IOT ❸
PSW:  140001
DEMON ❹ active, PCB addr = 054214
R0-R5:
000005   000000   023004   147602   160020   154752
Kernel SP: 000774
Kernel Stack:
005045 ❺ 000004   053336   046004   001012   000000   046236   000000
047044   000000   047450   000000   052074   000000   055334   000000
User SP: 154734
User Stack:
002013 ❻ 104262   140310   102250   000034   035064   004305   000000
000000   000003   000001   000004   000000   002445   000000   000000

KPAR(0-7):
    .
    .
    .
Booting
INIPIO-I  Booting...
```

Example B–1 Crash Dump Example

❶ This line calls out a crash and indicates the HSCxx is at software version number V100. The last field is the assigned node name (set with SET NAME).

❷ Indicates the processor mode in which the crash occurred. This can be either *Kernel* or *User*.

❸ A 3-letter mnemonic indicating the type of crash. The example mnemonic IOT indicates that this is a software inconsistency. Any other combination of letters, such as NXM (Nonexistent Memory) would designate a crash outside the scope of this appendix. Hardware exceptions are defined in Appendix D.

❹ The initial name on this line identifies the process active at the time of the crash. It is valid only during user-mode crashes. Use this name as a crosscheck when looking up the crash description.

❺ If the mode notation is *Kernel*, check the first word of the *Kernel Stack* for the crash code.

❻ Because the mode notation in this example indicated *User*, check the *User Stack* for the crash code number. This code is always the first word of the stack (in this case, *002013*).

B.2 SINI-E Error Printout

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.

```
SINI-E Seq 1. at 17-Nov-1858 00:00:02.00
        Software inconsistency ❶
        Process DEMON  ❷
        PC  000002
        PSW 140001
        Stack dump:  002013  104262  140310
```

Example B–2 SINI-E Exception Code

❶ This line defines the cause of the crash.

❷ 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.

B.3 Submitting a Software Performance Report

Some of the exception messages listed in this appendix suggest submitting a Software Performance Report (SPR) with a copy of the crash dump. Before submitting the SPR, contact the Customer Support Center or the local field office to see if additional information will be needed. Submit an SPR only after eliminating other possibilities, such as hardware-related problems.

If the customer requires immediate use of the HSC, you can reboot the HSC using the Init switch on the OCP. But this reboot causes cause a loss of the information necessary for an SPR, so be sure to make a hardcopy crash dump message and other required information.

After two or three similar exception messages occur an SPR should be submitted. 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 to cause 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.

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 the data structure memory:

= indicates program memory
= = indicates Control memory
= == indicates data memory

Table B–1 Obtaining Data Structure Information

Data Structure Needed	Type At * 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.

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.

B.4 Exception Messages

In each of the exception messages, **Facility** indicates the process(es) running at the time the crash occurred. The first name listed is the major process. The second name is the module of the process that generated the exception. This module may be a subprocess of the main process or simply a different code module. Include the crash dump message and any other applicable hardcopy information with an Software Performance Report (SPR) submission.

001001 ($CKERSTK)
Execution of Kernel Stack

Facility: EXEC, EXEC

Explanation: The HSC executive executed stack space.

Action: Submit an SPR with a crash dump. You may reboot the HSC immediately.

001002 ($CPUM1)
Previous mode not user

Facility: EXEC, EXEC

Explanation: During a context switch of user processes, the previous mode (as indicated by the Program Status Word (PSW)) was not user mode.

Action: Submit an SPR with a crash dump. R5 points to PCB (Process Control Block).

001003 ($CEXPCB)
EXEC PCB was scheduled

Facility: EXEC, EXEC

Explanation: During process scheduling, the EXEC process control block (PCB) was scheduled. This dummy PCB is used only for loading the process and should never be scheduled.

Action: Submit an SPR with a crash 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 Page Descriptor Register (PDR) directed to program memory does not have "disable cache" set. A PDR directed to data memory does have “disable cache” set.

Action: Submit an SPR with a crash dump. R0 points to PDR.

001005 ($CPUM2)
Previous mode not user

Facility: EXEC, EXEC

Explanation: During a context switch of user processes, the previous mode (as indicated by the PSW) was not user mode.

Action: Submit an SPR with a crash dump.

001006 ($CCB4)
Spurious Interrupt from K at Control Bus Level 4

Facility: EXEC, EXEC

Explanation: One of the Ks interrupted the P.ioc at Level 4, but, upon queue examination, no elements were shown (an element should be on the Level 4 Interrupt queue).

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, yet, upon queue examination, no elements were shown (an element should be on the Level 5 Interrupt queue.)

Action: Submit an SPR. If this crash continues to occur, escalate the problem to Customer 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.

Action: Submit an SPR with a crash dump. R1 points to the 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.

Action: Submit an SPR with a crash dump. R1 points to the 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.

Action: The process specified as active is the offender. Submit an SPR with a crash 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.

Action: The process specified as active is the offender. Submit an SPR with a crash 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.

Action: The process specified as active is the offender. Submit an SPR with a crash dump.

001015 ($CRRTI)
RRTI$ on Semaphore with address of 0

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 0.

Action: The process specified as active is the offender. Submit an SPR with a crash dump.

001016 ($CRTI1)
RRTI$ on Semaphore with address of 0

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 0.

Action: The process specified as active is the offender. Submit an SPR with a crash dump.

001017 ($CRTI2)
RRTI$ on Semaphore with address of 0

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 0.

Action: The process specified as active is the offender. Submit an SPR with a crash 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.

Action: The process specified as active is the offender. Submit an SPR with a crash 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.

Action: The process specified as active is the offender. Submit an SPR with a crash 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.

Action: The process specified as active is the offender. Submit an SPR with a crash dump.

001023 ($CRMPP)
Receive/Dequeue Multiple from Queue with address of 0

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 0.

Action: The process specified as active is the offender. Submit an SPR with a crash dump.

001024 ($CRMCP)
Receive/Dequeue Multiple from Queue with address of 0

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 0.

Action: The process specified as active is the offender. Submit an SPR with a crash dump.

001025 ($CRMCV)
Receive/Dequeue Multiple from Queue with address of 0

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 0.

Action: The process specified as active is the offender. Submit an SPR with a crash 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 with a queue head address of 0.

Action: The process specified as active is the offender. Submit an SPR with a crash 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.

Action: The process specified as active is the offender. Submit an SPR with a crash 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.

Action: The process specified as active is the offender. Submit an SPR with a crash dump.

001031 ($CSCV)
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, ENQ$C TO$P, SEND$C TO$W, or ENQ$C TO$W System Services was called with a queue head address of 0.

Action: The process specified as active is the offender. Submit an SPR with a crash dump.

001032 ($CSHPP)
Send/Enqueue-to-Head to Queue with address of 0

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 0.

Action: The process specified as active is the offender. Submit an SPR with a crash dump.

001033 ($CSHCP)
Send/Enqueue-to-Head to Queue with address of 0

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 0.

Action: The process specified as active is the offender. Submit an SPR with a crash 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.

Action: The process specified as active is the offender. Submit an SPR with a crash 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.

Action: The process specified as active is the offender. Submit an SPR with a crash 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.

Action: The process specified as active is the offender. Submit an SPR with a crash 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.

Action: The process specified as active is the offender. Submit an SPR with a crash 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.

Action: The process specified as active is the offender. Submit an SPR with a crash 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 downcount 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.

Action: Submit an SPR with a crash dump.

001042 ($CTOVR)
Time-of-day overflowed

Facility: EXEC, EXEC

Explanation: During an update of the current time of day, the executive detected an overflow. This can happen if a node on the CI sets a false time to the HSC.

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 Customer Service support.

001043 ($CPWFL)
Power Failure

Facility: EXEC, EXEC

Explanation: The processor is still operating 5 seconds after a power failure indication. Therefore, CRONIC concludes that the power failure indication was false.

Action: Verify the ac voltages are correct. If so, and the problem persists, notify Customer Service support.

001201 ($CNOHIBER)
Process on Recoverable List not hibernating

Facility: EXEC, EXECLOAD

Explanation: Before loading a utility or diagnostic, the loader examined the Recoverable Memory List of cache programs to determine whether a program might be loaded from memory instead of from the load device. When a program was found on the Recoverable Memory List, its state was not Hibernate State. This software inconsistency should not be seen under normal circumstances.

Action: Submit an SPR with a crash dump, noting previous activity with the program requested.

Register R3 points to PCB (Process Control Block) for process to restart.

001202 ($CIMAGE)
Memory extent encroaches defined area

Facility: EXEC, EXECLOAD

Explanation: The process to be loaded specified additional memory or buffer space, as specified on the Loadable File Header (LFHEADER) directive. When the additional memory was allocated and mapped to the process, it had encroached upon the loaded area.

Action: Submit an SPR with a crash dump. Register R0 points to XFRB (extended function request block) for loading the image. Register R4 points to CH$ (Canonical File Header).

001203 ($CNOPROC)
No code parent process loaded

Facility: EXEC, EXECLOAD

Explanation: When a process was loaded, its PCB specified it should execute and share code associated with another process. When attempting to locate the code parent, the loader found that the parent was not loaded.

Action: Submit an SPR with a crash dump. Register R2 equals process number of code parent. Register R3 points to code child's PCB.

001204 ($CALLOCATE)
Insufficient Kernel Pool

Facility: EXEC, EXECLOAD

Explanation: When EXEC attempted to allocate either a PCB (PCB-Z.) or an address Descriptor (A.) structure from Kernel Pool for a new process, Kernel Pool was inadequate to support the additional structures.

Action: Submit an SPR with a crash dump.

001205 ($CLFAO)
FAO overrun

Facility: EXEC, EXECLOAD

Explanation: The FAO string returned during formatting of a module version mismatch message was too large for the buffer.

Action: Submit an SPR with a crash dump. If possible, send a copy of the RX33 diskette.

001401 ($CBUSY)
Performed receive when already busy with request

Explanation: The READ$/WRITE$ service, while in its exception routine, was already busy with one request while a RCV$P operation was performed.

Action: Submit an SPR with a crash dump.

001402 ($CNOLOADED)
Requested driver not loaded

Facility: EXEC, EXECRDWR

Explanation: A process within the HSC specified a READ$ or WRITE$ operation with a device control block (DDCB) 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.

Action: Submit an SPR with a crash 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 that 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 that was invalid, or it specified an invalid device type in the DD$TYPE field.

Action: Submit an SPR with a crash 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 that 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.

Action: Submit an SPR with a crash dump.

001502
Software Inconsistency—Non-RX33 command requested

Facility: EXEC, EXECRX33

Explanation: The RX33 driver received an XFRB (CRONIC transfer request), but the XFRB specified a DDCB for a non-RX33 device. R4 points to DDCB, R5 points to XFRB (extended function request block).

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).

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. R2 equals byte count, R5 points to XFRB (extended function request block).

Action: Submit an SPR with a crash dump.

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).

Action: Submit an SPR with a crash dump.

001506
Software Inconsistency—Invalid internal byte count

Facility: EXEC, EXECRX33

Explanation: The 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).

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.

Explanation: Check for RX33-related hardware failures. If the problem persists, submit an SPR with the crash dump.

001510
Software Inconsistency—Invalid Head Select

Facility: EXEC, EXECRX33

Explanation: The Software attempted to select a head other than 0 or 1. R0 equals head select.

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 is not set in MMR0.

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).

Action: Submit an SPR with a crash dump.

001513
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; that is, the crash does not happen when the interrupt is detected.

Action: If the problem persists, submit an SPR with the crash dump. Further testing of the HSC subsystem load device area may be necessary.

001514
Software Inconsistency—Invalid Internal Unit Number

Facility: EXEC, EXECRX33

Explanation: The unit number index value is not 0 or 2. This unit number index value is contained in R4.

Action: Submit an SPR with a crash dump.

001515
Software/Hardware Inconsistency – Nonexistent Memory

Facility: EXEC, EXECRX33

Explanation: The RX33 controller returned an NXM error.

User Action: Further testing of the HSC subsystem (load device area) may be necessary. If the problem persists, submit an SPR with the crash dump.

001601 ($CPAG1)
TYPE$ crosses page boundaries

Facility: EXEC, EXECTT

Explanation: A process requested a TYPE$ system service (or an ACPT$ service with a prompt) specifying a buffer that crosses a memory management page boundary. This is a restriction of the driver. R0 equals size of print string. R1 points to String Buffer. R4 points to DDCB (device control block). R5 points to XFRB (extended function request block).

Action: Submit an SPR with a crash dump, describing activity at the time of the exception.

001602 ($CPAG2)
ACPT$ crosses page boundaries

Facility: EXEC, EXECTT

Explanation: A process requested an ACPT$ System Service specifying a buffer that crosses a memory management page boundary. This is a restriction of the driver. R4 points to DDCB (device control block). R5 points to XFRB (extended function request block).

Action: Submit an SPR with a crash dump, describing activity at the time of the exception.

001603 ($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 the Terminal Service 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 EXEC searched the run queue specified in the Z.RUNQ field of the PCB, it could not find the PCB. This is a software inconsistency. R4 points to attached PCB.

Action: Submit an SPR with a crash dump.

001701 ($CPAGE)
READ$ or WRITE$ crossed page boundary

Facility: EXEC, EXECTU58

Explanation: A request to the TU58 driver specified a buffer that crossed a memory management page boundary. This is a restriction of the driver.

Action: Submit an SPR with a crash dump, describing activity at the time of the exception. The process listed as active may be the READ$/WRITE$ Service and not the process initiating 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.

Action: Submit an SPR with a crash dump. If a certain sequence of HSC operations induced this crash, include a description of that sequence. A software problem is the most likely cause of this crash.

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.

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.

Action: Submit an SPR with a 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.

Action: A failing P.ioc module is the most probable cause of this crash. 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/K.si or K.sti/K.si tests detected a failure. Failures in these tests are fatal, and the HSC must reboot after displaying a message describing the type of error and requestor number of the failed module.

Action: A failing K.sdi/K.si or K.sti/K.si module is the most probable cause of this crash. 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.

Action: 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.

Action: Submit an SPR with a 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.

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/K.si control area was not available. When ILDISK has exclusive ownership of a drive state area, the address of the K.sdi/K.si control area should always be available. A software problem is the most likely cause of this crash.

Action: Submit an SPR with a 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 2 minutes, the program found that one or more I/O requests had not completed. The HSC crashes and reboots because ILEXER cannot exit with a request outstanding.

Action: Submit an SPR with a 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.

A faulty disk drive is the most likely cause of this problem. Further testing of the 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 2 minutes, the program found that one or more I/O requests had not completed. The HSC crashed and rebooted because ILEXER cannot exit with a request outstanding.

Action: Submit an SPR with a 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.

A faulty tape drive or formatter is the most likely cause of this problem. Further testing of the suspect tape subsystem and associated requestor(s) may be necessary. This crash may also be caused by the K.sti/K.si clocks stopping due to a hardware error such as an Instruction Parity error.

002013
ILTAPE was supplied an illegal requestor number

Facility: DEMON, ILTAPE

Explanation: ILTAPE was automatically initiated to test a particular formatter. One of the parameters supplied to ILTAPE is the requestor number of the K.sti/K.si connected to the formatter. ILTAPE checked the specified requestor and found it was not a K.sti/K.si.

Action: Submit an SPR with a crash dump. Also include a summary of any tape error messages immediately preceding the crash. If a certain sequence of HSC operations caused 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: ILTAPE requested exclusive access to a tape formatter for testing. ILTAPE timed out because the request did not complete within 60 seconds.

Action: Submit an SPR with a crash dump. Also include a summary of any tape error messages immediately preceding the crash. If a certain sequence of HSC operations caused 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, but the command failed. In the process of preparing an error message, ILTAPE found that the command Opcode was an illegal or unknown value).

Action: Submit an SPR with a crash dump. Also include a summary of any tape error messages immediately preceding the crash. If a certain sequence of HSC operations caused 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 containing a zero. While trying to return a TACB to the table, ILTAPE discovered that all table entries were nonzero, implying that no TACBs were in use.

Action: Submit an SPR with a 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 state area, but the address of the K.sti/K.si control area was not available. When ILTAPE has exclusive ownership of a drive state area, the address of the K.sti/K.si control area should always be available. A software problem is the most likely cause of this crash.

Action: Submit an SPR with a crash dump. If a certain sequence of HSC operations caused 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. R4 equals Format Type.

Action: Submit an SPR with a crash dump.

003002 ($CFAO1)
Output length too long

Facility: CERF

Explanation: When CERF processed an MSCP error message, the FAO output of the text string was too long for CERF's buffer. R1 equals number of bytes output.

Action: Submit an SPR with a crash dump.

003003 ($CFAO2)
Output length too long

Facility: CERF

Explanation: When CERF processed an out-of-band message, the FAO output of the text string was too long for CERF's buffer. R1 equals number of bytes output.

Action: Submit an SPR with a crash dump.

004002
BMB reserved but not found

Facility: DISK, many

Explanation: A Big Memory Buffer (BMB) was reserved through a system function but was not found when the table of BMBs was searched. This indicates memory corruption, mismanagement of the BMB pool, or lack of enough BMBs to handle the load on the disk.

Action: Submit an SPR with a crash dump. Specify which process was running and make note of the activity on the system at the time of the erase.

004004
Invalid action byte in Connect Block

Facility: DISK, SDI

Explanation: The subprocess within the disk path that processes requests from the CI Manager received a connect block with an invalid action byte. This indicates that an invalid structure was passed to the process, the structure was passed at the improper time, or memory was corrupted.

Action: Submit an SPR with a crash dump. Include the contents of user register 2 in the crash dump.

004005
Datagram received from a connection

Facility: DISK, MSCP

Explanation: The main MSCP disk command server process received a nonsequenced message from some connection. This may indicate memory corruption or improper message reception. It may also indicate that an improper structure was passed to the process, possibly by the host software.

Action: Submit an SPR with a 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 maximum, from some connection. This may indicate memory corruption or improper message reception. It may also indicate that an improper structure was passed to the process, possibly by the host software.

Action: Submit an SPR with a crash dump. Note all levels of host software running in the cluster.

004007
Invalid error signaled by K.ci

Facility: DISK, MSCP

Explanation: The main MSCP command server received an MSCP command packet, with invalid error bits set, from the K.ci. This may indicate memory corruption or improper message reception. It may also indicate that an improper structure was passed to the process, possibly by the host software.

Action: Submit an SPR with a 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.

Action: Submit an SPR with a crash dump. Note the current process running.

004011
Invalid module number

Facility: DISK, many

Explanation: The main disk process detected an invalid module number when it tried to switch to a different internal process represented by the module number. This indicates that memory is corrupted or that an invalid structure has queried to the main work queue.

Action: Submit an SPR with a crash dump. Note the current process running.

004013
State change to ONLINE requested through gatekeeper

Facility: DISK, SDI

Explanation: The state change processor within the sequential command gatekeeper received a DUCB extension requesting a state change to *on line*. This crash indicates an improper use of the state change mechanism.

Action: Submit an SPR with a crash dump.

004014
Inconsistent drive state detected

Facility: DISK, SDI

Explanation: The state change processor within the sequential command gatekeeper received a DUCB extension containing a different state than the current state in the DUCB. This crash indicates an improper use of the state change mechanism..

Action: Submit an SPR with a crash dump.

004015
Improper state change for shadow member

Facility: DISK, SDI

Explanation: The sequential gatekeeper mechanism completes action for shadow units before allowing a state change on any of the members of the shadow set. This crash indicates the mechanism failed to operate properly.

Action: Submit an SPR with a crash dump.

004016
Disk Unit Table (DU) inconsistency

Facility: DISK, many

Explanation: The disk server tried to add a unit to the DU when it was already there, or tried to remove a unit from the DU that was not present. This crash indicates improper sequencing of actions to add or remove a unit in the DU. This crash can also occur if the ordered list of DUCBs is destroyed.

Action: Submit an SPR with a crash dump.

004017
Invalid diagnostic HMB

Facility: DISK, MSCP

Explanation: The diagnostic interface within the disk path received a host message block (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.

Action: Submit an SPR with a crash dump. List any utilities or diagnostics running at the time of the crash.

004021
Diagnostic release of disk unit while on line

Facility: DISK, MSCP

Explanation: A diagnostic or utility attempted to release a disk unit while the disk unit was still on line.

Action: Submit an SPR with a crash dump. Specify the utilities or diagnostics running at the time of the crash.

004025
Error identification table overwritten

Facility: DISK, ERROR

Explanation: The disk error identification table was overwritten or a wild branch was taken. The most probable cause is a bad load.

Action: If this crash occurs immediately after a boot, try rebooting with a backup copy of the HSC software. Otherwise, submit an SPR with a crash dump.

004026
Invalid error bit value found during error recovery

Facility: DISK, ERROR

Explanation: The bit value describing a K.sdi/K.si error was not valid for a given stage of the error recovery. It is also possible, though unlikely, that a K.sdi/K.si is malfunctioning.

Action: If this error appears to recur from the same K.sdi/K.si, replace it.

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. It is also possible, though unlikely, that a disk is supplying invalid characteristics to the HSC.

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 disk unit, repair it.

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.

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/K.si, replace it. If no hardware problem exists, submit an SPR with a 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.

Action: This error could be caused by a malfunctioning K.sdi/K.si. Further testing of the suspected requestor may be necessary. If this error appears to recur from the same K.sdi/K.si, replace it. If no hardware problem exists, submit an SPR with a crash dump.

004032
Unable to get to FRB error state

Facility: DISK, ERROR

Explanation: Error recovery was unable to place a port in the FRB error state to perform an error recovery operation. This crash can occur in an extremely unlikely compound error situation.

Action: Reboot the HSC. If this error persists, submit an SPR with a 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.

Action: Submit an SPR with a crash dump.

004034
Level B retry in wrong state

Facility: DISK, ERROR

Explanation: A Level B retry operation was attempted without the drive port being in FRB error state.

Action: Submit an SPR with a crash dump.

004035
Level C retry in wrong state

Facility: DISK, ERROR

Explanation: A Level C retry operation was attempted without the drive port being in FRB error state.

Action: Submit an SPR with a 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/K.si is processing a DCB, but the DCB queue is empty.

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 suspect requestor may be necessary. If this error appears to recur from the same K.sdi/K.si, replace it. If no hardware problem exists, submit an SPR with a crash dump.

004037
Invalid error queue address in route

Facility: DISK, ERROR

Explanation: When the disk server attempted to route an FRB to an error queue, the error queue address in a route descriptor was invalid.

Action: Submit an SPR with a 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/K.si or K.ci.

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 suspect requestor may be necessary. If this error appears to recur from the same K.sdi/K.si, replace it. If no hardware problem exists, submit an SPR with a 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.

Action: Submit an SPR with a 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.

Action: Submit an SPR with a 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.

Action: Submit an SPR with a crash dump.

004044
Buffer not found for specified error

Facility: DISK, ERROR

Explanation: A call to the error recovery subroutine RCDHMX specified a buffer that was not in the list of buffers for the specified FRB.

Action: Submit an SPR with a crash dump.

004046
DRAT not found for FRB retirement

Facility: DISK, ERROR

Explanation: While attempting to retire an FRB by simulating route completion, the error recovery subroutine RETIRE could not locate the DRAT for downcounting.

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.

This crash is caused by either overwritten memory or a malfunctioning K.sdi/K.si. Further testing of requestors and HSC internal buses may be necessary. If this error appears to recur from the same K.sdi/K.si, replace it.

If no hardware problem exists, submit an SPR with a 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 that a DRAT queue for the drive was not empty.

Action: Submit an SPR with a 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.

Action: Submit an SPR with a crash dump.

004052
Known drive not found in the Disk Unit Table

Facility: DISK, MSCP

Explanation: When the disk server attempted to remove a known disk unit from the Disk Unit Table, the unit was not found in that table.

Action: Submit an SPR with a crash dump. Note any utilities or diagnostics running at the time of the crash.

004055
Attempt to enable drive interrupt already enabled

Facility: DISK, many

Explanation: The ARM subroutine was called to enable K.sdi/K.si interrupts to the disk server for drive state changes when interrupts were already enabled.

Action: Submit an SPR with a 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 K.sdi/K.si interrupts for drive state changes while a drive state change was being processed.

Action: Submit an SPR with a crash dump. Note the process running at the time of the crash.

004057
Invalid drive state change requested

Facility: DISK, many

Explanation: The SCHSQM subroutine was called to schedule a state change operation for a drive that has been declared inoperative but whose state is still recorded as available.

Action: Submit an SPR with a crash dump. Note the process running at the time of the crash.

004070
Nonzero status for SUCCESSful DCB

Facility: DISK, SDI

Explanation: Although a DCB (SDI command) completed with a status of SUCCESS, the error word indicated errors, or the SDI command opcode was invalid.

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/K.si, replace it. If no hardware problem exists, submit an SPR with a crash dump.

004072
DCB state is busy with empty DCB queue

Facility: DISK, many

Explanation: The drive state indicator in the K control area indicates a DCB is being processed by the K.sdi/K.si, but the DCB queue is empty.

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 the memory subsystem may be necessary. If this error appears to recur from the same K.sdi/K.si, replace it. If no hardware problem exists, submit an SPR with a crash dump.

004073
K.sdi is not responding

Facility: DISK, SDI

Explanation: A K.sdi/K.si failed to process an immediate DCB within a preset time.

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/K.si.

004100
No thread block for operation

Facility: DISK, SDI

Explanation: A thread block was not available to the SDI interface in order to block the current thread process

Action: Submit an SPR with a 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.

Action: Submit an SPR with a crash dump.

004106
DRAT allocation failure

Facility: DISK, many

Explanation: There was not enough free control memory to allocate a DRAT for a specific drive type.

Action: Submit an SPR with the crash dump.

004107
A command did not complete after the drive was declared inoperative

Facility: DISK, MSCX

Explanation: Since no processing was being done on an outstanding command, Get Command Status processing declared the drive inoperative. The outstanding command, however, the command still failed to complete in the timeout period.

Action: Submit an SPR with the crash dump. Note the drive type of the drive identified in the error message and any errors reported by the disk server prior to the crash.

004110
Get Command Status overflow

Facility: DISK, MSCP

Explanation: Get Command Status processing determined the calculated status will result in an overflow.

Action: Submit an SPR with the crash dump.

004111
A timer's link field values are inconsistent with its current operational state

Facility: DISK, many

Explanation: A timer was in a state that prevented adding or removing it from an active list.

Action: Submit an SPR with the crash dump.

004112
Inconsistent shadow member state detected

Facility: DISK, many

Explanation: A unit is incorrectly marked as a member of a shadow set, or the shadow unit links are inconsistent given the current state of the shadow unit.

Action: Submit an SPR with the crash dump.

004113
NO DRAT list is invalid

Facility: DISK, many

Explanation: The NO DRAT list was found to be invalid when declaring a drive inoperative.

Action: Submit an SPR with the crash dump.

004114
Connection closed after delay in ATTN process

Facility: DISK, AVLATT

Explanation: While the disk server was waiting to acquire resources to send an attention message to the host, the connection closed.

Action: Submit an SPR with the crash dump.

004115
DCB address inconsistency

Facility: DISK, SDI

Explanation: While processing an error on a seek DCB, the current seek DCB address was inconsistent with the DCB address stored in the DRAT.

Action: Submit an SPR with the crash dump.

004116
Bad error completion queue in DCB

Facility: DISK, MSCP

Explanation: An invalid error completion queue was found in the DCB when it was being setup for a seek operation. This indicates that after the previous seek operation, the DCB error completion queue was not properly restored before the DCB was retired.

Action: Submit an SPR with the crash dump.

004117
No DRAT was found on the K.sdi DRAT list when expected

Facility: DISK, many

Explanation: The DRAT list was empty when the disk server expected to find a DRAT queued to the K.sdi/K.si DRAT list. This most likely cause is a disk server design error.

Action: Submit an SPR with the crash dump.

004120
Too many DRATS in use during ESE transfer operations

Facility: DISK, MSCP

Explanation: The number of DRATs in use has exceeded the maximum value allowed. The possible causes include a design error in the disk transfer code or corruption of the count of DRATs in use.

Action: Submit an SPR with the crash dump.

004121
RBN access during an ESE transfer

Facility: DISK, MSCP

Explanation: The disk server is preparing to perform a transfer operation to an RBN, but the ESE has no RBNs.

Action: Submit an SPR with the crash dump.

004122
Invalid DRAT bit set

Facility: DISK, MSCP

Explanation: A DRAT on the K.sdi/K.si DRAT list did not have the "Set D bit on completion" flag set as expected. This indicates that the DRAT was probably not set up properly.

Action: Submit an SPR with the crash dump.

004123
DCB K.sdi list inconsistency

Facility: DISK, many

Explanation: More than one non-seek DCB was queued to the K.sdi/K.si during I/O rundown on an ESE. Only one non-seek DCB is expected to be active at a time.

Action: Submit an SPR with the crash dump.

004124
Buffer count inconsistency for ESE

Facility: DISK, MSCP

Explanation: During transfer processing, the DRAT buffer count indicated that the DRAT was full. However, the DRAT full flag was not set. The most likely cause is a transfer design error.

Action: Submit an SPR with the crash dump.

005001
ECC self-diagnostic string too big for FAO

Facility: ECC

Explanation: A self-diagnostic string generated for the ECC process was too big to print with the allocated FAO buffer. This crash can only occur if the self-diagnostic code is present and enabled. The self-diagnostic code is not enabled for distributed base levels.

Action: Submit an SPR with a crash dump.

005002
No ECC errors to correct

Facility: ECC

Explanation: An FRB without any errors was sent to the ECC process.

Action: Submit an SPR with a crash dump.

005003
Can't allocate XFRB to print self-diagnostic messages

Facility: ECC

Explanation: The ECC process failed to allocate an XFRB (extended function request block) or printing messages during self-diagnostic.

Action: Submit an SPR with a crash dump.

005004
ECC found more than a 10-bit symbol error

Facility: ECC

Explanation: The ECC process received a buffer containing more than a 10-bit symbol error. Error recovery processing should never pass on such a buffer.

Action: Submit an SPR with a crash dump.

006000
This class of crashes is for tape path software inconsistency errors

Facility: TAPE, TFxxxx

Explanation: A software inconsistency error occurred.

Action: Submit an SPR with a 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 fail 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.

Action: Verify that the K.sti/K.si tape data channel is working; if so, submit an SPR with a crash dump.

006002
Received an interrupt from an unknown tape data channel

Facility: TAPE, TFATNAVL

Explanation: The tape server 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.

Action: Submit an SPR with a 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 diagnostic interface. General Register 1 points to the windowed address of the connection block (CB) in question. General Register 2 contains the Opcode in question.

Action: Submit an SPR with a crash dump. Include the connection block (CB) structure.

006004
An illegal diagnostic Opcode was received

Facility: TAPE, TFDIAG

Explanation: A diagnostic 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.

Action: Submit an SPR with a crash dump. Specify the utilities or diagnostics active at the time of the crash. Include the HMB structure.

006005
Diagnostics trying to acquire assigned drive state area

Facility: TAPE, TFDIAG

Explanation: Diagnostics are trying to acquire the 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).

Action: Submit an SPR with a crash dump. Specify the diagnostics or utilities active at the time of the crash. Include the HMB, TFCB, and tape drive control block (TDCP) structures.

006006
Inconsistencies during drive state area acquisition

Facility: TAPE, TFDIAG

Explanation: The software context word KT$SFW is not equal to the tape formatter control block (TFCB) address and/or the 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).

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, be 0.

Action: Submit an SPR with the crash dump. Include details of the BACKUP operation. Include the host message block (HMB) (command packet) structure.

006010
No buffers supplied in BACKUP operation

Facility: TAPE, TFDIAG

Explanation: No disk data block buffers were supplied in the host message block (HMB) for the backup operation. General Register 3 points to the windowed Control Memory address of the HMB in question. General Register 0 should point to the buffer descriptor list for the backup operation.

Action: Submit an SPR with a crash dump. Include details of the BACKUP operation. Include the host message block (HMB) (command packet) structure.

006011
Could not allocate a XFRB

Facility: TAPE, TFLIB

Explanation: The tape server could not allocate an XFRB (extended function request block) through ALOCB, a CHRONIC system service.

Action: Submit an SPR with a crash dump.

006012
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).

Action: Submit an SPR with a 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 the command packet received. General Register 3 is the windowed Control Memory address of the command packet in question.

Action: Submit an SPR with a 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 is currently illegal. General Register 3 points to the windowed Control Memory address of the command packet in question. Offset HM$ERR is the field in question.

Action: Further testing of HSC hardware, particularly the K.ci, may be necessary. If no hardware problem exists, submit an SPR. Indicate the host software version and include the host message block (HMB) (command packet) structure.

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 the tape server searching the current chain of tape drive control blocks (TDCBs) 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) 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).

Action: Submit an SPR with a crash dump.

006016
Unable to allocate an HDB

Facility: TAPE, TFSEQUEN

Explanation: The tape server's attempt to add a host access, which requires allocation of a host disk block (HDB), failed for lack of resources.

Action: Submit an SPR with a crash dump.

006017
Tape formatter does not support allowed densities

Facility: TAPE, TFSEQUEN

Explanation: The tape formatter does not support a density that the HSC supports. General Register 4 points to the tape drive control block (TDCB) for the drive in question.

Action: Submit an SPR with a crash dump. Include the host software version and 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). General Register 4 points to the tape drive control block (TDCB) in question.

Action: Submit an SPR with a crash dump. Include the host message block (HMB) 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 the drive in question. General Register 2 points to the tape formatter control block (TFCB) for the formatter in question.

Action: Submit an SPR with a crash dump. Include the following structures: host message block (HMB), tape drive control block (TDCB), and tape formatter control block (TFCB).

006022
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.

Action: Submit an SPR with a crash dump. Include the host message block (HMB) structure.

006023
Requested transfer larger than 64 Kb

Facility: TAPE, TFSEQUEN

Explanation: The requested transfer size for a read reverse is larger than 64 Kb. 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.

Action: Submit an SPR with a 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 the formatter in question.

Action: Submit an SPR with a crash dump. Include the following structures: host message block (HMB), tape drive control block (TDCB), and 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 the drive in question. General Register 2 points to the tape formatter control block (TFCB) for the formatter in question.

Action: Submit an SPR with a crash dump. Include the following structures: host message block (HMB), tape drive control block (TDCB), and 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. General Register 3 points to the K control area in question.

Action: Submit an SPR with a crash dump. Include the K control area structure.

006031
No available stacks

Facility: TAPE, TFSERVER

Explanation: There are no available stacks for a process trying to suspend.

Action: Submit an SPR with a crash dump.

006033
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 server return routine.

Action: Submit an SPR with a crash dump.

006040
No stack available to suspend with

Facility: TAPE, TFSTI

Explanation: There is no stack available to suspend 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).

Action: Submit an SPR with the crash dump and include the following structures: TFCB, DCB, and K control area.

006041
DCB operation timed out

Facility: TAPE, TFSTI

Explanation: A dialogue control block (DCB) operation timed out.

Action: 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. If no hardware problem exists, submit an SPR.

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.

Action: Submit an SPR with a crash dump. Include the fragment request block (FRB) structure.

006044
Unexpected fragment request block (FRB) error received

Facility: TAPE, TFERR

Explanation: The tape server received an error from a software station rather than a hardware station. General Register 5 points to the fragment request block (FRB) in error.

Action: Submit an SPR with a crash dump. Include the FRB structure.

006045
Unknown fragment request block (FRB) error received

Facility: TAPE, TFERR

Explanation: An unidentifiable error is flagged in a fragment request block (FRB).

Action: Submit an SPR with a crash dump. Include the FRB structure.

006046
K.ci did not return a fragment request block (FRB)

Facility: TAPE, TFERR

Explanation: Transfer request blocks (TRBs) have associated fragment request blocks (FRBs) 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.

Action: Check the K.ci. If no hardware problem exists, submit an SPR with a crash dump.

006047
Invalid downcount occurred on a host message block (HMB) chain

Facility: TAPE, TFERR

Explanation: Whenever transfer request blocks (TRBs) were purged from the K.sti/K.si input queue, the associated host message block (HMB) were 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.

Action: Submit an SPR with a crash dump. Include the HMB.

006050
Sequence number corruption occurred

Facility: TAPE, TFERR

Explanation: Error recovery ensures against a deadlock on K.sti/K.si by preventing a transfer request block (TRB) from waiting for a diagnostic control block (DCB) that will never execute. This is a software inconsistency.

Action: Submit an SPR with a crash dump.

007000
This class of crashes includes CIMGR software consistency errors

Facility: CIMGR, many

Explanation: A software inconsistency error occurred.

Action: Submit an SPR with a 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.

Action: Submit an SPR with a crash dump. Include the HMB.

007002
Failed to acquire a control block from K.ci

Facility: CIMGR

Explanation: The POLLER process could not obtain a control block from the K.ci to resend a timed-out STACK datagram.

Action: Further testing of the HSC subsystem may be necessary, particularly the available control memory. If no hardware problem exists, submit an SPR with a crash dump.

007003
K.ci is hung

Facility: CIMGR

Explanation: During the polling interval (60 seconds), the CIMGR ensures K.ci is still running. This trap indicates it is not.

Action: Further testing of the HSC subsystem may be necessary, particularly the K.ci. If no hardware problem exists, submit an SPR with a crash dump.

007004
K.ci detected an unrecoverable error and stopped

Facility: CIMGR

Explanation: K.ci sent its control area to the CIMGR exception process. K.ci does this whenever it detects a nonrecoverable hardware error.

Action: Further testing of the HSC subsystem may be necessary, particularly the K.ci and data memory. If no hardware problem exists, submit an SPR with a crash dump.

007005
K.ci patch status check failed

Facility: CIMGR

Explanation: K.ci did not respond to a path status check within 8 seconds.

Action: If no hardware problem exists, submit an SPR with a crash dump.

Action: Further testing of the HSC subsystem may be necessary, particularly the K.ci. If no hardware problem exists, submit an SPR with a crash dump.

007006
System name is corrupted

Facility: CIMGR

Explanation: During initialization, the CIMGR discovered the system name in the SCT was corrupted.

Action: Release the Online button (out) on the HSC. Reboot the HSC by holding the Fault button in until the State light blinks. This will bypass using the SCT on the boot device. Run SETSHO to reset the system name and ID, then reboot the HSC again before pushing in the Online button on the front panel.

007007
HMB received with wrong number of BMBs

Facility: CIMGR

Explanation: CIMGR received a host message block (HMB) with the wrong number of big message blocks (BMBs), or CIMGR detected an inconsistent state. General Register 0 points to the HMB.

Action: Further testing of the HSC subsystem may be necessary, particularly the K.ci. If no hardware problem exists, submit an SPR with a crash dump.

007011
Connection incarnation inconsistent

Facility: CIMGR

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).

Action: Submit an SPR with a crash dump. Include the CB.

007012
Connection incarnation mismatch

Facility: CIMGR

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 a dangling reference to an old carnation of the connection.

General Register 2 points to the connection block (CB).

Action: Submit an SPR with a crash dump. Include the CB.

007013
Inconsistent connection state due to a VC closure

Facility: CIMGR

Explanation: CIMGR attempted an illegal state transition on a connection. The state transition was initiated by a virtual circuit closure. General Register 2 points to the connection block (CB).

Action: Submit an SPR with a crash dump. Include the CB.

007014
Unable to retrieve resource from K.ci during a disconnect

Facility: CIMGR

Explanation: During a disconnect, the CIMGR was unable to retrieve the resources from the K.ci associated with the credits on that connection.

Action: Submit an SPR with a crash dump.

007015
K.ci did not respond to notification of a VC closure

Facility: CIMGR

Explanation: The K.ci did not respond to notification of a virtual circuit closure with the 12-second time limit. This crash occurs if the response times out.

Action: Further testing of the HSC subsystem may be necessary, particularly the K.ci. If no hardware problem exists, submit an SPR with a crash dump.

007016
Illegal connector state

Facility: CIMGR

Explanation: CIMGR detected an illegal connector block (CB) state. General Register 2 points to the CB.

Action: Submit an SPR with a crash dump. Include the CB.

007017
Attempt to deallocate a connection block without an incarnation

Facility: CIMGR

Explanation: A connection block (CB) did not have a valid incarnation at the time it was deallocated.

Action: Submit an SPR with a crash dump. Include the CB.

007020
Failure to retrieve SCS resources from K.ci

Facility: CIMGR

Explanation: When CIMGR tried to allocate resources for use across a virtual circuit, the count of data memory resources was incorrect. The host message block (HMB) for serializing VC traffic must have two big message blocks (BMBs). General Register 0 points to the HMB.

Action: Submit an SPR with a crash dump. Include the HMB.

007021
The count of waiters for virtual circuit resources went negative

Facility: CIMGR

Explanation: While processing the list of waiters for virtual circuit transmission resources, CIMGR detected a nonempty list to indicate a negative number of waiters. General Register 1 points to the system block (SB).

Action: Submit an SPR with a crash dump. Include the SB.

007022
Invalid BMB address

Facility: CIMGR

Explanation: An HMB arrives at the resource collector with an invalid BMB address attached to it.

Action: Use the SETSHO **SHOW REQUESTORS** command to view the K.pli microcode revision level. If it is less than revision 45, contact your Digital Customer Service representative for the update. Submit an SPR with the crash dump and note the disk configuration.

007023
SCS buffer retrieval failure

Facility: CIMGR

Explanation: When changing the status of the virtual circuit, CIMGR tries to retrieve the SCS buffer from the K.ci.KHSRR queue. This buffer should be on the queue because it is not in use at the time of the crash. No elements were enqueued on the .KHSRR queue, therefore, CIMGR forced a crash.

Action: Submit an SPR with the crash dump.

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.

Action: Submit an SPR with an 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 BMB. Therefore, the HMB is invalid. The third word of the stack contains the value in HM$CN — the count of the number of BMBs.

Action: Submit an SPR with an exception dump or startup message indicating the contents of the stack. The second word of the stack contains the windowed address of the HMB.

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.

Action: Submit an SPR with an 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$NC. The second word of the stack contains the value in the HM$ERR field. (Extra credits received; HME$EC=10. No credits received; HME$NC=4.)

Action: Submit an SPR with an exception dump or startup message indicating the contents of the stack.

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).

Action: Submit an SPR with an exception dump or startup message indicating the contents of the stack. The second word of the stack contains the contents of the CB$ACT field.

012024
Bad Down Count

Facility: DUP

Explanation: DUP initiates a return of the endpacket to the host by down counting the reference counter in the related control block. The down-count action should return a one. If the downcount did not decrement the reference counter to 1, DUP crashes the HSC. The second word of the stack is the value of the counter following the downcount.

Action: Submit an SPR with an exception dump or startup message indicating the contents of the stack.

012036
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 zero.

Action: Submit an SPR with a crash dump.

042001
FAO message buffer overflow

Facility: DIRECT

Explanation: The program DIRECT was attempting to output the formatted directory end message, but the length of that message was longer than the allotted FAO output buffer.

Action: Submit an SPR with a crash dump.

043001
Wrong HMB received when trying to bring source on line

Facility: DKCOPY

Explanation: DKCOPY sent a host message block (HMB) to the disk server requesting the source unit be brought on line in a shadow set. When the completion queue of this HMB was checked, it pointed to a different (incorrect) HMB. This is crash $CDKCOPY+SRC_ONL_HMB.

Action: Submit an SPR with a crash dump. Top of stack equals crash code. Second word points to previous HMB.

043002
Bad downcount when trying to bring source on line

Facility: DKCOPY

Explanation: When an MSCP end message was to be sent over a connection to a host, a counter keeping track of the transaction (decrementing by 1) failed to operate properly. This occurred after DKCOPY asked the disk server to bring the source unit on line in a shadow set. This is crash $CDKCOPY+SRC_ONL_CNT.

Action: Submit an SPR with a crash 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: DKCOPY sent a host message block (HMB) to the disk server requesting it to send a GET COMMAND STATUS (GCS) command to the target unit. When the completion queue of this HMB was checked, it pointed to a different (incorrect) HMB. This is crash $CDKCOPY+TGT_GCS_HMB.

Action: Submit an SPR with a crash 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: When an MSCP end message was to be sent over a connection to a host, a counter keeping track of the transaction (decrementing by 1) failed to operate properly. This occurred after DKCOPY asked the disk server to send a GET COMMAND STATUS (GCS) command to the target unit. This is crash $CDKCOPY+TGT_GCS_CNT.

Action: Submit an SPR with a crash dump. Top of stack equals crash code. Second word points to counter.

043005
Bad downcount when trying to bring target unit on line

Facility: DKCOPY

Explanation: When an MSCP end message was to be sent over a connection to a host, a counter keeping track of the transaction (decrementing by 1) failed to operate properly. This occurred after DKCOPY asked the disk server to bring the target unit on line into the shadow set. This is crash $CDKCOPY+TGT_ONL_CNT.

Action: Submit an SPR with a crash 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: When an MSCP end message was to be sent over a connection to a host, a counter keeping track of the transaction (decrementing by 1) failed to operate properly. This occurred after DKCOPY asked the disk server to abort an ONLINE command to the target unit. This is crash $CDKCOPY+TGT_ABO_CNT.

Action: Submit an SPR with a crash 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: DKCOPY sent a host message block (HMB) 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. This is crash $CDKCOPY+SHA_AVL_HMB.

Action: Submit an SPR with a crash 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: When an MSCP end message was to be sent over a connection to a host, a counter keeping track of the transaction (decrementing by 1) failed to operate properly. This occurred after DKCOPY asked the disk server to send the available shadow unit. This is crash $CDKCOPY+SHA_AVL_CNT.

Action: Submit an SPR with a crash 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: The SETSHO main routine did not acquire an XFRB (extended function request block). A crash was initiated because the lack of an XFRB prevents communication between the HSC and the console. This is crash $CSETSHO+NOXFRB.

Action: Submit an SPR with a crash dump.

051002
Failed to properly send HMB to K.ci

Facility: SETSHO, SSMAIN

Explanation: SETSHO sent a host memory block (HMB) to the K.ci (the hardware that handles communication between the hosts and the HSC). A crash was initiated because SETSHO did not receive confirmation of the HMB from the K.ci within the required time. This is crash $CSETSHO+CIHMB.

Action: Submit an SPR with a crash dump.

051003
Too many characters intended for console printout

Facility: SETSHO, SSMAIN

Explanation: In this case, when SETSHO called Formatted ASCII Output (FAO), it generated more characters than the buffer size allocated would allow. The maximum buffer size is 510 characters. This is crash $SETSHO+PNTOVF. R1 points to string size.

Action: Submit an SPR with a crash dump.

051004
The SCT (System Control Table) crossed a page boundary

Facility: SETSHO, SSMAIN

Explanation: The SCT must remain on one page in memory. The crash typically indicates an incorrect amount of padding was placed at the end of the file SSDATA.MAC. This is crash $SETSHO+SCTXPG.

Action: Submit an SPR with a crash dump.

051101
Failed in sending HMB to disk server for SET Dn [NO]HOST

Facility: SETSHO

Explanation: SETSHO sent a host memory block (HMB) to the disk server to set a disk drive to HOST or NOHOST access. The crash was initiated because the confirmation of this command was not received within the required time. This is crash $CSETSHO+SETDSK.

Action: Submit an SPR with a crash dump.

051102
Failed in sending HMB to tape server for SET Tn [NO]HOST

Facility: SETSHO

Explanation: SETSHO sent a host memory block (HMB) to the tape server to set a tape drive to HOST or NOHOST access. The crash was initiated because the confirmation of this command was not received within the required time. This is crash $CSETSHO+SETTAP.

Action: Submit an SPR with a crash dump.

051201
Failed in sending HMB to disk server for SHOW Dn

Facility: SETSHO

Explanation: SETSHO sent a host memory block (HMB) to the disk server to show a specified disk drive. The crash was initiated because the confirmation of this command was not received within the required time. This is crash $CSETSHO+SHODSK.

Action: Submit an SPR with a crash dump.

051202
Failed in sending HMB to tape server for SHOW Tn

Facility: SETSHO

Explanation: SETSHO sent a host memory block (HMB) to the tape server to show a specified tape drive. The crash was initiated because the confirmation of this command was not received within the required time. This is crash $CSETSHO+SHOTAP.

Action: Submit an SPR with a crash dump.

051203
SCT crash context table contained too many characters

Facility: SETSHO

Explanation: The SCT crash context table contained too many characters. In this case, when SETSHO called FAO, it generated more characters than the buffer size would allow. The maximum buffer size is 510 characters. This is crash $SETSHO+CSHOVF. R1 points to string size.

Action: Submit an SPR with a crash dump.

052001 ($CDWMATH)
Doubleword math not consistent

Facility: SINI

Explanation: During calculation and allocation of control blocks (allocated in quantities of a doubleword), the count of words in control blocks was not a doubleword multiple. R0 points to memory descriptor (MD).

Action: Submit an SPR with a crash dump.

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.

Action: Submit an SPR with a crash dump.

052003 ($CMUL8)
Multiply operation set overflow

Facility: SINI

Explanation: During allocation of control blocks (set as 80 percent of available control memory), a multiply operation set the PSW Overflow bit.

Action: Submit an SPR with a crash dump.

061001
XCALL stack overflow

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 that should never change. This crash occurs when the routine DDUTIO is called. The low word of the stack contains a value different than the initialization value. The most probable cause of the crash is corruption by the process running.

Action: Submit an SPR with a 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 requesting an out-of-band error log be issued through 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.

Action: Submit an SPR with a crash dump.

062002
Common Pool memory returned twice

Facility: Many

Explanation: A process attempted to return a memory segment that was already in the common pool.

Action: Submit an SPR with the crash dump.

C
Generic Error Log Fields

C.1 Introduction

Some fields described on HSC console message printouts are generic, regardless of error type.

The following example is a typical printout of the error log fields. Table C–1 describes the error fields.

```
ERROR-S Bad Block Replacement (Success)
        Command Ref #    0A66000D
        RA81 unit #      77.
        Err Seq #        166.
        Format Type      09.
        Error Flags      80
        Event            002B
ERROR-I End of Error
```

Example C–1 Error Log Fields Example

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, date, and time.
Command Ref #	This number, in hexadecimal, is the MSCP command number that caused the error reported, or is 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 is 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. Refer to Table C–2 for a description of the error flags.
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.

C.2 Error flags

Table C–2 is a list of error flags that can be set. The first column is the bit number that is set. The second column is the bit mask hex number. The third column is the format description of the error flag.

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	This is MSCP-specific. If set, the identified logical block number (LBN) needs replacement.
4	10	This is 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.3 MSCP/TMSCP Status or Event Codes

Event codes are values reported to error logs and are equivalent to each status code.

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 Status or 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 that this is combined with the status code): offset * 256. + code offset * 256. is the command message and offset value in decimal for the field in error. + code is the symbol for the invalid command status code.
0002	Command Aborted	Command aborted.
0003	Unit Off Line	Unit unknown or on line to another controller.
0004	Unit Available	Unit available.
0007	Compare Error	Used only as an event code when the error occurs during a Read-Compare or Write-Compare operation.

Table C–3 (Cont.) MSCP/TMSCP Status or Event Codes

Event Code Hex	Class	Description
0008	Data Error	Disk—Sector was written with Force Error modifier. Tape—Long gap encountered.
0009	Host Buffer Access Error	Cause not available. The controller was unable to access a host buffer to perform a transfer and has no visibility into the cause of the error.
000A	Controller Error	Reserved for host-detected command timeout logging. This error is never reported by a controller.
000C	Shadow Set Status Has Changed	Disk—Shadow set status has changed. Tape—Formatter error.
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.
0020	Success	Disk—Spindown ignored; status only subcode. Tape—Unload ignored.
0023	Unit Off Line	Disk—No volume mounted or drive disabled via RUN/STOP switch. Unit is in known substate; status only subcode. Tape—No media mounted, disabled via switch setting, or on line to another controller.
002A	Controller Error	SERDES overrun or underrun error. Either the drive is too fast for the controller, or more typically, a controller hardware fault has prevented controller microcode from keeping up with data transfer to or from the drive.
002B	Disk Drive Error	Drive command timeout For SDI drives, the controller timeout expired for either a level 2 exchange or the assertion of Read/Write Ready after an Initiate Seek.
0034	Bad Block Replacement	Block verified good—not a bad block.
0035	Media Loader	Loader command timeout. The key length is too short for the specified key type.
0040	Success	Still connected; status only subcode.
0043	Unit Off Line	Unit is inoperative; status only subcode. For SDI drives, the controller has marked the drive inoperative due to an unrecoverable error in a previous level 2 exchange, the drive C1 flag is set or the drive has a duplicate unit identifier.
0044	Unit Available	Shadow set copy in progress; status only subcode.

Table C–3 (Cont.) MSCP/TMSCP Status or Event Codes

Event Code Hex	Class	Description
0048	Disk Data Error	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 or a valid header. For unrecoverable errors, this code implies the subsystem attempted nonprimary 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 mis-sync, header sync timeout, and an unreadable header.
0049	Host Buffer Access Error	Odd byte count.
004A	Controller Error	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 This can also be caused by any K module (including the K.ci) writing bad EDC into Data memory.
004B	Disk Drive Error	Controller-detected transmission error For SDI 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 analog failed.
0055	Media Loader	Controller-detected transmission error. The controller does not implement the specified key type.
0068	Disk Data	Data sync not found (data sync timeout).
0069	Host Buffer Access Error	Nonexistent Memory error.
006A	Controller Error	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 or hardware problem.
006B	Disk Drive Error	Positioner error (mis-seek). 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	Media Loader Error	Controller-detected protocol error.
0080	Success	Duplicate unit number; status only subcode.
0083	Unit Off Line	Duplicate unit number; status only subcode.
0084	Shadowing Unit Available	No members in shadow set. An on-line command was addressed to a virtual unit of an existing shadow set from which all members have been removed.
0085	Media Format (Shadowing) Error	Characteristics or protection mismatch for shadow member.

Table C–3 (Cont.) MSCP/TMSCP Status or Event Codes

Event Code Hex	Class	Description
0088	Disk Data Error	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 Access Error	Host memory parity error.
008A	Controller Error	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 Error	Lost Read/Write Ready during or between transfers. For SDI 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.
0094	Bad Block Replacement	Replacement failure—drive access failure. One or more transfers specified by the replacement algorithm failed.
00A5	Media Format Error	Disk—Not formatted with 512-byte sectors; status only subcode. The disk FCT indicates it is formatted with 576-byte sectors, although either the controller or the drive support only 512-byte sectors For tape—Block mode device not formatted for tape operations.
00A9	Host Buffer Access Error	Invalid page table entry.
00AA	Controller Error	LESI adapter card parity error on input (adapter to controller).
00AB	Disk Drive Error	Drive clock dropout. For SDI 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 Format Error	Disk not formatted or FCT corrupted; status only subcode. The disk FCT indicates the disk is not formatted in either 512- or 576-byte mode.

Table C–3 (Cont.) MSCP/TMSCP Status or Event Codes

Event Code Hex	Class	Description
00C9	Host Buffer Access Error	Invalid buffer name. The key in the buffer name does not match the key in the buffer descriptor, the V bit in the buffer descriptor is clear, or the index into the buffer descriptor table is too large.
00CA	Controller Error	LESI adapter card parity error on output (controller to adapter).
00CB	Disk Drive Error	Lost Receiver Ready for transfer For SDI 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 Error	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 algorithms, or a transfer with the Suppress Error Correction modifier encountered an ECC error of any severity. For tape—Unrecoverable read error.
00E9	Host Buffer Access Error	Buffer length violation. The number of bytes requested in the MSCP or TMSCP command exceeds the buffer length as specified in the buffer descriptor.
00EA	Controller Error	LESI adapter card *cable in place* not asserted.
00EB	Disk Drive Error	Drive-detected error. For SDI 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 on line; status only subcode.
0103	Unit Off Line	Unit disabled by field service or diagnostic; status only subcode. For SDI drives, the drive DD flag is set.
0105	Disk Media Format Error	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 on line.
0106	Write-Protected	Unit is data safety write-protected; status only subcode.
0108	Disk Data Error	One-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.
0109	Host Buffer Access Error	Access control violation. The access mode specified in the buffer descriptor is protected against the PROT field in the PTE.
010A	Controller Error	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.

Table C–3 (Cont.) MSCP/TMSCP Status or Event Codes

Event Code Hex	Class	Description
010B	Disk Drive Error	Controller-detected pulse or state parity error. For SDI 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 Format Error	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 on line.
0128	Disk Data Error	Two-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.
012A	Controller Error	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 HSC 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 Error	Drive-requested error log (EL bit set).
0145	Disk Media Format Error	No multicopy protection. All but one copy of a block in a multicopy structure are bad. The disk should be reformatted or replaced at the earliest convenient time.
0148	Disk Data Error	Three-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.
014A	Controller Error (Shadowing)	Insufficient resources. The controller is unable to honor a request to create a shadow set or to add an additional member to an existing shadow set. This is due to the lack of internal resources to support the new entity.
014B	Disk Drive Error	Controller-detected protocol error. For SDI 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, or was not a possible response in the context of the exchange.
0168	Disk Data Error	Four-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.
016A	Controller Error	PLI transmission buffer parity error.

Table C–3 (Cont.) MSCP/TMSCP Status or Event Codes

Event Code Hex	Class	Description
016B	Disk Drive Error	Drive failed initialization. For SDI 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 Error	Five-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.
018B	Disk Drive Error	Drive ignored initialization. For SDI 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 Error	Six-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.
01AB	Disk Drive Error	Receiver Ready collision. For SDI drives, the controller attempted to assert its Receiver Ready when the Receiver Ready of the drive was still asserted.
01C8	Disk Data Error	Seven-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.
01CB	Disk Drive Error	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/K.si.
01E8	Disk Data Error	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 on line.
0203	Unit Off Line	Exclusive use.
0208	Disk Data Error	Nine-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.
0220	Success	Still on line, unload ignored.
0228	Disk Data Error	Ten-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.
0248	Disk Data Error	Eleven-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.

Table C–3 (Cont.) MSCP/TMSCP Status or Event Codes

Event Code Hex	Class	Description
0268	Disk Data Error	Twelve-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.
0288	Disk Data Error	Thirteen-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.
02A8	Disk Data Error	Fourteen-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.
02C8	Disk Data Error	Fifteen-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.
0400	Success	Disk—Incomplete replacement; status only subcode. For tape—EOT encountered.
0404	Unit Available	Already in use; status only subcode.
044B	Tape Drive	Drive error. Controller retry limit exhausted.
0800	Drive error	Invalid RCT; status only subcode.
1000	Success	Read only volume format; status only subcode.
1006	Write-Protected	Unit is software write-protected; status only subcode.
2006	Write-Protected	Unit is hardware write-protected; status only subcode.
F3AA	Controller Error	Unknown K.sti/K.si error.
FCAA	Controller Error	Word rate clock timeout. The K.sti/K.si detected the loss of clocks from a drive during a transfer.
FCEA	Controller Error	Receiver Ready not asserted at start of transfer. 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.
FD2A	Controller Error	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 Error	Acknowledge not asserted at start of transfer. The HSC 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.

Table C–3 (Cont.) MSCP/TMSCP Status or Event Codes

Event Code Hex	Class	Description
FE2A	Controller Error	Record EDC error. On a read from tape operation the EDC calculated by the K.sti/K.si 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 Error	Lower Processor timeout. The Upper Processor in the K.sti/K.si detected the Lower Processor had stopped and restarted it.
FE8B	Tape Drive	Unable to position to before L_EOT.
FEAB	Tape Drive	Rewind failure.
FECB	Tape Drive	Could not complete on-line 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 Error	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 Error	Buffer EDC error. The K.sti/K.si 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/K.si-detected error.
FFAA	Controller Error	Data overflow due to pipeline error. No Data Buffers in HSC Data memory were available when the K.sti/K.si 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 HSC disagree on tape position.

Table C–3 (Cont.) MSCP/TMSCP Status or Event Codes

Event Code Hex	Class	Description
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.

D
Interpretation of Status Code Bytes

D.1 Introduction

This appendix lists all possible codes each K (e.g. K.ci or data channel) can generate after detecting a fatal error. Only K-detected errors are listed here.

When a K detects a fatal error, it puts a code in its status register and performs a level 7 Control bus interrupt to the P.io. 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.

The following 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 which forced the crash. Section D.2 explains this crash is detail. 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
```

Example D–1 (Cont.) K-detected Error Example

```
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

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
```

Example D–1 K-detected Error Example

D.2 K-Detected Error Example Examination

Notice the third line of Example D-1 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 that 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, K.sti, and K.si 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.

D.3 K-Detected Failure Code Analysis

The following sections aid field service in analyzing the K-detected failure codes through 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 the K.sdi status codes.
- Table D–3 describes the K.sti status codes.
- Table D–4 describes the K.si disk status codes.
- Table D–5 describes the K.si tape status codes.

The use of the status code tables requires information about 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 asterisk (*) 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/K.si
203 for a K.sti/K.si
377 means no requestor is in the slot

Any value other than a 001, 002, 203, or 377 means the K detected an error. Because the K.ci is always requestor 1, 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). When the HSC reboots, type **SHOW REQUESTOR** at the HSC> prompt to see whether the requestor detecting the error is a K.sdi, K.sti, or K.si. Find the number of the data channel that found the error in the displayed response. This display shows whether that requestor number is either a K.sdi, K.sti, or K.si.

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, K.sti, or K.si.

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.

NOTE
"(See NOTE.)" appears in several places in the following tables. In each table, this information appears at the end of that table.

Table D–1 K.ci Status Code 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 reason for 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 powerup/Init diagnostics.
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 which 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.)

Table D–1 (Cont.) K.ci Status Code Bytes

Status Code (Octal)	Description
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.
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 and 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.
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.

Table D–1 (Cont.) K.ci Status Code Bytes

Status Code (Octal)	Description
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 and 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.
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 and 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.

NOTE
The sequencers access Control memory several times 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 an asterisk (*) 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 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 K.ci hardware revision level

Table D–2 K.sdi Status Code 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 powerup/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.)
15	The Upper Processor tried to advance the buffer descriptor pointer when the old value of the pointer is zero (illegal).

Table D–2 (Cont.) K.sdi Status Code Bytes

Status Code (Octal)	Description
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, OR 4, 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.

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 may 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 (pulse error)
  - Bit 14 = Upper Processor RTCS clock present
  - Bit 15 = Parity error on RTDS Line

Table D–3 K.sti Status Code Bytes

Status Code (Octal)	Description
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.
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.)

Table D–3 (Cont.) K.sti Status Code Bytes

Status Code (Octal)	Description
	In the following status codes, bit 7 is the parity bit. Parity is always odd for microdiagnostic failures and is always even for functional code failures. Bit 6 is the error bit and is set for microdiagnostic and functional code failures.
000	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.
103	Control bus (CBUS) test failed upon powerup or reinitialization.
106	Scratchpad RAM test failed upon powerup or reinitialization.
110	Partial STI test failed upon powerup or reinitialization.
112	The K.sti encountered a Control bus/memory problem while searching for the K-Init packet in Control memory.
301	2901 ALU test failed upon powerup or reinitialization.
302	Data bus (DBUS) test failed upon powerup or reinitialization.
304	PROM test failed upon powerup or reinitialization.
307	SERDES test failed upon powerup or reinitialization.
313	After completing powerup/Init diagnostics, the K.sti could not find a properly formatted K-Init packet in Control memory.

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 may 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 (pulse error)
  - Bit 14 = Upper Processor RTCS clock present

Table D–5 (Cont.) K.si Tape Status Code Bytes

Status Code (Octal)	Description
03	Control Bus (CBUS) test failed at powerup or reinitialization.
04	The PROM/writable control store (WCS) parity test failed at powerup or reinitialization.
06	Scratchpad RAM test failed at powerup or reinitialization.
07	RTS Gate Array test failed at powerup or reinitialization.
10	Calibration of the SIECL failed during powerup or reinitialization.
11	WCS moving inversions test failed during off-line diagnostics.
12	The K.si encountered a Control Bus/memory problem while searching for the K-init packet in Control Memory.
13	Attempt to load module's WCS failed during powerup or off-line load attempt.
14	While trying to write the microcode version into the control area at KG$VRSN, the upper sequencer encountered a Control Bus error. R11 contains the contents of the upper error register.
15–22	Control Bus error. See note at the end of this table.
23	During transfer completion, the buffer descriptor link word in the FRB was zero. RAM7 contains the lower processor status.
24–33	Control Bus error. See note at the end of this table.
34	The lower processor has timed out on a transfer operation and the upper processor cannot restart it.
35–36	Control Bus error. See note at the end of this table.
37	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.
41–43	Control Bus error. See note at the end of this table.
44	While setting up a transfer, the next buffer descriptor in the FRB was zero because the buffer descriptor named a nonexistent buffer.
45	Control Bus Error.
70–73	If SIECL/SERDES path loop testing failed during powerup or reinitialization because of one of the following conditions: 70 — Frame loopback from upper to lower 71— Frame loopback from lower to upper 72— Sector loopback from upper to lower 73— Sector loopback from lower to upper
74	The upper processor 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.
75–76	Control Bus error. See note at the end of this table.

NOTE

When an error occurs, the upper processor transfers the contents of the upper processor error register at the time of the error to register R11 (octal).

The contents of R11 are given in the crash dump to help you narrow the error possibilities. The following list defines all the bits contained in R11 from the upper processor error register. Those bits that can indicate the possible cause of a crash are denoted with an asterisk (*).

Bit 0	Even/odd bit for 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

HSC70-AA/CA		REV	A1				B1	E5											
NUMBER	DESCRIPTION		REVISIONS																
L0100-00	ILI (CI LINK)	E2-ETCH	D1	→	→	→													
		C-ETCH	D1	→	→	→													
L0107-YA	K.PLI		C2	C3	→	→	C4												
L0108-YA (HSC5X-BA)	K.SDI	C-ETCH	C8	→	→	→													
		D-ETCH	C8	C9	C10	→													
		E-ETCH	C1	C2	C3	→													
		F-ETCH	C22				C22	C23	C24	C25									
L0108-YB (HSC5X-CA)	K.STI	D-ETCH	C10	→	→	→													
		E-ETCH	C3	C4	→	→													
		F-ETCH	C23				C23	C24	C25	C26									
L0109-00	PILA		E1	E2	→	→													
L0111-00	P.IOJ	C-ETCH	A1	→	→	→													
		D-ETCH	A2	→	→	→													
L0117-AA	M.STD2	A-ETCH	A2	→	→	→													
L0118	ILI (CI LINK)																		
L0119	K.SI	D-ETCH																	
5417764-01	BACKPLANE	C-ETCH	A1				C1	D1											

CXO-1271B
Sheet 1 of 4

Figure E–1 (Cont.) HSC Revision Matrix Chart

Figure E–1 (Cont.) HSC Revision Matrix Chart

HSC70-AA/CA	REV	A1		B1	E5													
NUMBER	DESCRIPTION	REVISIONS																
70-20033-03	STD PS ASSY - 120 VAC IN	C1	→															
70-20184-01	OPT PS ASSY - 120 VAC IN	B2	→															
30-24374-01	881A PWR CTRL 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-HSCMN-IN	INSTALLATION MANUAL	001	→															
QX926-H7	HSC70 SOFTWARE	V100		V300	V370	V370+												
BL-FH74X-DE	HSC70 OFF-LINE DIAGS	A	→															

**THIS BREAKDOWN IS FOR FIELD SERVICE INFORMATION ONLY.

CXO-1271B
Sheet 2 of 4

E.3 HSC50 (Modified) Revision Matrix Chart

Figure E–2 shows the revision status of all applicable HSC50 (modified) FRUs. An HSC50 (modified) must have all the FRUs at a particular revision level in order to be supported.

Figure E–2 (Cont.) HSC50 (Modified) Revision Matrix Chart

HSC50-AA		REV	E5																
NUMBER	DESCRIPTION		REVISIONS																
L0100-00	ILI (CI LINK)	E2-ETCH	D1																
		C-ETCH	D1																
L0105-00	P.IOC	D-ETCH	E1	E3	E5														
		E-ETCH	E2	E4	E6														
L0106-AA	M.STD		C1																
L0107-YA	K.PLI		C1	C2	C3	C4													
L0108-YA (HSC5X-B)	K.SDI	C-ETCH	C6	C7	C8														
		D-ETCH	C5	C6	C7	C8	C9	C10											
		E-ETCH	C1	C2	C3														
	K.SDI	F-ETCH	C23	C24	C25														
L0108-YB (HSC5X-C)	K.STI	D-ETCH	C10																
		E-ETCH	C3	C4															
	K.STI	F-ETCH	C24	C25	C26														
L0109-00	PILA		E1	E2															
54-14048-00	BACKPLANE	D-ETCH	A1	A2															
L0118	ILI (CI LINK)																		
L0119	K.SI	D-ETCH																	

CXO-2078A
Sheet 1 of 4

HSC50-AA	REV	E4								E5								
NUMBER	DESCRIPTION	REVISIONS																
70-20033-01	STD PS ASSY - 120 VAC IN	A1	B1	C1	C2	C3				A1	B1	C1	C2	C3				
70-20033-03	STD PS ASSY - 120 VAC IN	C1	C2							C1	C2							
HSC5X-EA	OPT PS KIT - 120 VAC IN	A1	A2							B2								
70-20184-01	OPT PS ASSY - 120 VAC IN	A1	B1	B2						A1	B1	B2						
70-19122-00	PWR CTRL ASSY - 120/208 V	A1	A2	A3	B1	B2	C1	C2										
ZD300-CG	HSC50 DK/TP SRVR FRMWR																	
70-20524-01	OCP ASSEMBLY	A2	A3							A2	A3							
54-15286-01	OCP	C								C								
70-20186-01	BEZEL ASSEMBLY (TU58)	A1								A								
TU58-XA	DRV MECH (70-15510-00)	F	K							F	K							
TU58-XB	S INTRFC (54-13489-00)	F3	F4	K	L	M				F3	F4	K	L	M				
EK-HSCMN-IN	INSTALLATION MANUAL	001	002							001	002							
AA-GMEAA-TK	USER GUIDE																	
OX926-HG	HSC50 SOFTWARE	V350								V350	V370	V370+						
BE-T493X-XX	HSC50 OFF-LINE DIAGS	E-DE								E-DE								
30-24374-01	CONTROLLER, PWR 120 V, 3-PHASE, 9-OUTLET									B1								

CXO-2078A
Sheet 2 of 4

Figure E–2 (Cont.) HSC50 (Modified) Revision Matrix Chart

HSC50-AB		REV	E6																
NUMBER	DESCRIPTION		REVISIONS																
L0100-00	ILI (CI LINK)	E2-ETCH	D1																
		C-ETCH	D1																
L0105-00	P.IOC	D-ETCH	E1	E3	E5														
		E-ETCH	E2	E4	E6														
L0106-AA	M.STD		C1																
L0107-YA	K.PLI		C1	C2	C3	C4													
L0108-YA (HSC5X-B)	K.SDI	C-ETCH	C6	C7	C8														
		D-ETCH	C5	C6	C7	C8	C9	C10											
		E-ETCH	C1	C2	C3														
	K.SDI	F-ETCH	C23	C24	C25														
L0108-YB (HSC5X-C)	K.STI	D-ETCH	C10																
		E-ETCH	C3	C4															
	K.STI	F-ETCH	C24	C25	C26														
L0109-00	PILA		E1	E2															
54-14048-00	BACKPLANE	D-ETCH	A1	A2															
L0118	ILI (CI LINK)																		
L0119	K.SI	D-ETCH																	

CXO-2078A
Sheet 3 of 4

Figure E–2 (Cont.) HSC50 (Modified) Revision Matrix Chart

HSC50-AB	REV	E4						E5						E6				
NUMBER	DESCRIPTION	REVISIONS																
70-20033-02	STD PS ASSY - 240 VAC IN	A1	B1	C1	C2	C3		A1	B1	C1	C2	C3		A1	B1	C1	C2	C3
70-20033-04	STD PS ASSY - 240 VAC IN	C1	C2					C1	C2					C1	C2			
HSC5X-EB	OPT PS KIT - 240 VAC IN	A1	A2					A1	A2					B2				
70-20184-02	OPT PS ASSY - 240 VAC IN	A1	B1	B2				A1	B1	B2				A1	B1	B2		
70-20613-01	PWR CTRL ASSY - 240/416 V	A1	A2	B1	C1			A1	A2	B1	C1							
ZD300-CG	HSC50 DK/TP SRVR FRMWR																	
70-20524-01	OCP ASSEMBLY	A2	A3					A2	A3					A2	A3			
54-15286-01	OCP	C						C						C				
70-20186-01	BEZEL ASSEMBLY (TU58)	A1						A1						A1				
TU58-XA	DRV MECH (70-15510-00)	F	K					F	K					F	K			
TU58-XB	S INTRFC (54-13489-00)	F3	F4	K	L	M		F3	F4	K	L	M		F3	F4	K	L	M
EK-HSCMN-IN	INSTALLATION MANUAL	001	002					001	002					001	002			
AA-GMEAA-TK	USER GUIDE																	
OX926-HG	HSC50 SOFTWARE	V300						V350						V350	V370	V370+		
BE-T493X-XX	HSC50 OFF-LINE DIAGS	E-DE						E-DE						E-DE				
30-24374-02	CONTROLLER, PWR 240 V, 3-PHASE, 9-OUTLET																	

CXO-2078A
Sheet 4 of 4

Figure E–2 HSC50 (Modified) Revision Matrix Chart

E.4 HSC50 Revision Matrix Chart

Figure E–3 shows the revision status of all applicable HSC50 FRUs. An HSC50 must have all the FRUs at a particular revision level in order to be supported.

D

F

N

O

P

Q

R

S

T