Educational Services

digital™

EY-9222E-SB-0001

VAXcluster Maintenance
Student Workbook
Volume 2

The information in this document is subject to change without notice and should not be construed as a commitment by Digital Equipment Corporation. Digital Equipment Corporation assumes no responsibility for any errors that may appear in this document.

The software described in this document is furnished under a license and may be used or copied only in accordance with the terms of such license.

No responsibility is assumed for the use or reliability of software on equipment that is not supplied by Digital Equipment Corporation or its affiliated companies.

The following are trademarks of Digital Equipment Corporation:

CI	J-11	RX
DEC	LA12/50/70	UNIBUS
DECnet	MASSBUS	VAX
DECsystem-10	PDP	VAXcluster
DECSYSTEM-20	PDT	VAXsim
HSC	RSTS	VMS
IAS	RSX	digital

CONTENTS

Volume 2

INTRODUCTION TO THE HSC

Introduction to the HSC

Lesson Introduction

This module gives an overview of the HSC Controller. It outlines the physical attributes of both the HSC50 and HSC70, and defines the various controls and indicators.

Lesson Objectives

1. Locate and identify the major components of the HSC and describe the function of each.

2. Describe each of the bulkhead cabling connections.

3. Identify each control and indicator and explain its function.

Lesson Outline

I. Overview
II. Controls and Indicators
III. Physical Introduction

HSC50/70 Overview

The HSC50 and HSC70 are intelligent mass storage controllers for a cluster environment. The HSC50 is an earlier version, but the two models are very similar. Both are members of the product family designated as Digital Storage Architecture (DSA), which includes storage units (such as RA disks) and controllers (such as the UDA50). The following is a list of the HSC features:

- Host Connection
  - The HSC connects to the rest of the cluster only through the CI Bus (two redundant pairs of coax cable).
  - The HSC receives commands from the host and returns end messages that follow the rules of Mass Storage Control Protocol (MSCP). Data blocks exchanged between the host and HSC also follow the same rules.
  - Messages on the CI Bus that relate to the cluster connection are governed by rules of the System Communication Architecture (SCA) and System Communication Services (SCS).
  - Because it is a cluster ~~member~~, the HSC is a controller that can handle multiple hosts.
- Storage Unit Connection
  - The HSC handles both disks and tapes that are connected in radial fashion using the Standard Disk Interconnect (SDI) and Standard Tape Interconnect (STI) cables.
  - Commands, responses, and data transmission between the HSC and the storage device is done using the protocol of SDI/STI.
  - The HSC handles SDI/STI ports through "data channel modules," each module driving four ports.
  - A data channel module is for a disk or a tape, but not both.
  - The HSC manages I/O operations by decoding an MSCP command and initiating the seeks and head switches necessary to select the proper track. It also monitors rotational position, checks headers, reads/writes data, and checks for errors.

HSC50/70 Overview (Cont.)

- The HSC can reorder the sequence of commands from other hosts in order to optimize throughput.
- Long transfers are fragmented for faster throughput.
- The HSC provides Controller Initiated Bad Block Replacement (CIBBR).
- Disk drives that are supported include RA60, RA80, RA81, and RA82. Tape drives supported include TA78 and TA81.

- HSC System Software
  - When the HSC is powered on, it performs an extensive self-test and then loads the operating system software called CRONIC (Colorado Rudimentary Operating Nucleus for Intelligent Controllers).
  - The self-test and boot process are initiated automatically on power-up, but can also be started by the operator, CRONIC, or host software.
  - The CRONIC operating system has no knowledge of file management, file protection, or file structure, all of which are the responsibility of VMS on the host nodes.
  - The system software is divided into the following categories: operating system, utilities, inline diagnostics, and offline diagnostics.

- HSC Diagnostics
  - There are no host-resident (VAX) diagnostics to test the HSC or the storage units attached to it.
  - Inline diagnostics are run in addition to the self-tests that are run automatically during initialization. They are usually started by the operator and can be run while the HSC is online to the cluster. Inline diagnostics test the units attached to the HSC or the HSC itself.
  - Offline diagnostics test only the HSC internals. They are loaded from the HSC load device (TU58 or RX floppy) by the operator at the local terminal. They are run in place of the operating system (CRONIC cannot be up). While running offline diagnostics, the HSC is effectively off the CI bus and therefore not functioning as an MSCP server.

HSC50/70 Overview (Cont.)

- Utilities are run with CRONIC up and the HSC online to the cluster. They provide services such as formatting, disk verification, bad block replacement, and storage unit exercising. ILTAPE ILMSCP ILDISK

- Error Indicators

  - The HSC ensures data integrity with the following types of checks:

    a. CRC on all CI Packets.
    b. EDC on all data blocks. FOR PARALLEL DATA PATH
    c. ECC on all data blocks. FOR SERIAL DATA PATH (SDI CABLES)
    d. Byte parity on all internal memory.

  - The HSC provides an Operator Control Panel (OCP) fault code when it detects a fatal error.

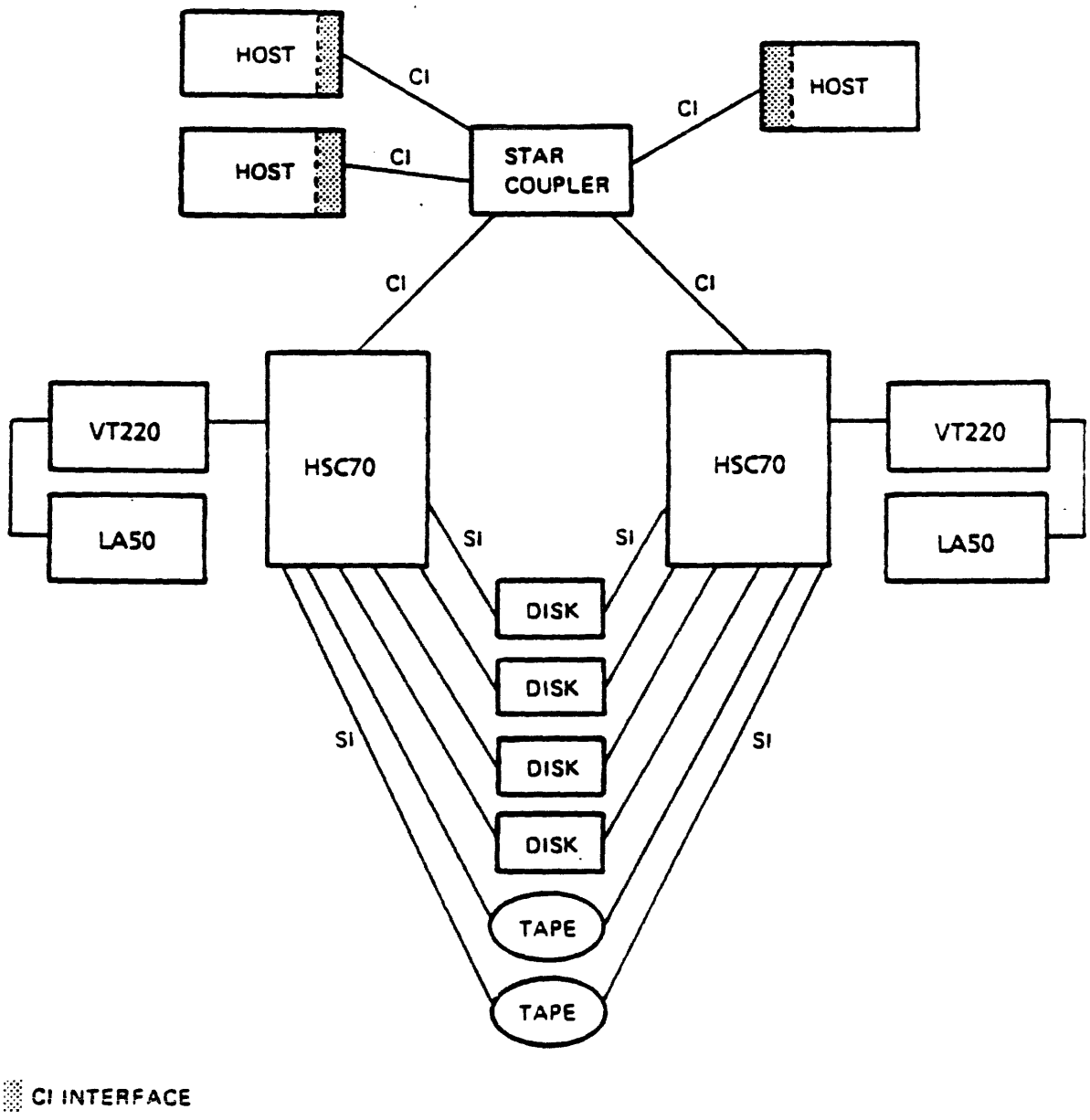
  - Red/green (go/nogo) LEDs on each of the modules help to locate the failing FRU.

  - If the HSC detects an error that cannot be corrected, it reboots itself. System status at the time of the crash is recorded in the System Configuration Table (SCT) located on the load medium.

SCS

SO DON'T WRITE PROTECT CONSOLE MEDIA

Show exception - Last reason for reboot



CX-886B

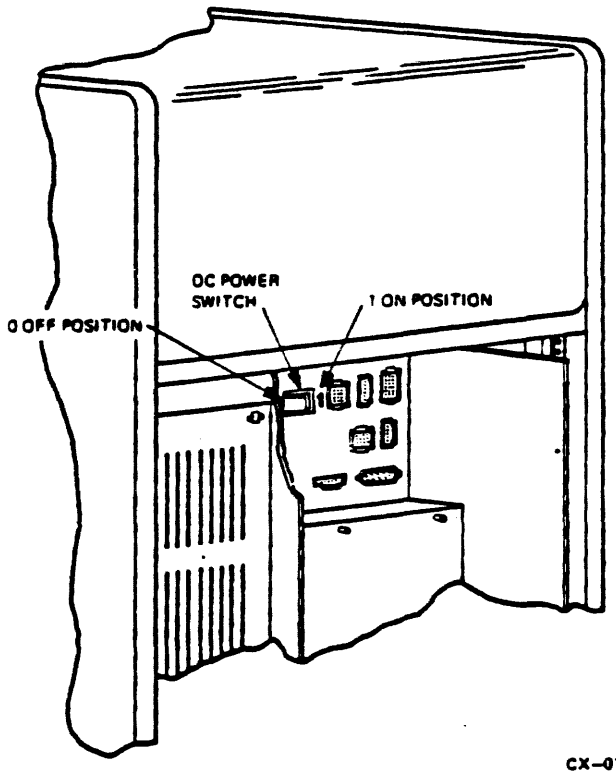
Redundant Cluster Configuration

HSC50/70 Differences

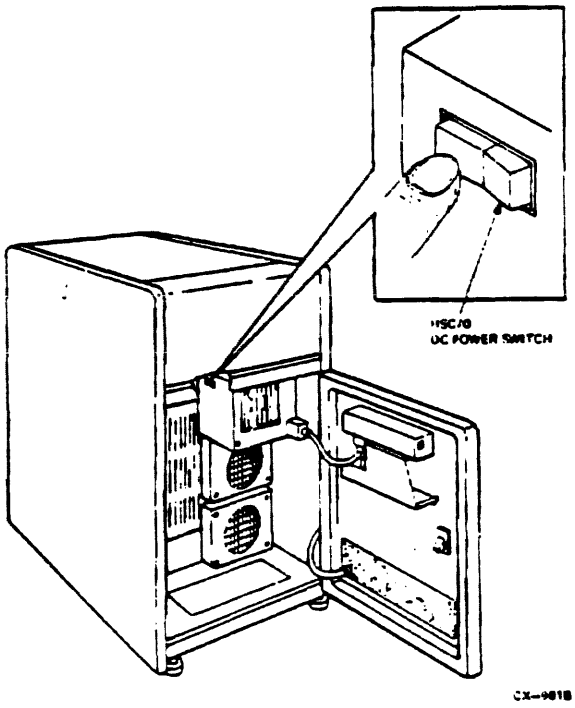
	HSC50	HSC70	HSC40
Console terminal	LA12 or VT220 and LA50/70	VT220 and LA50/70	SAME AS 70
Load device	TU58	RX33	SAME - RX33
Auxiliary power supply	Optional	Standard	NOT AVAILABLE
I/O control processor	F-11 based	J-11 based	
Program Memory	256Kb	1Mb	
Control Memory	128Kb	512KB	
Data memory	128Kb	512KB	
Data Channel modules installed	Up to 6	Up to 8	3 MAX
Disk drive/tape formatters connected	Up to 24	Up to 32	12 MAX

HSC50/70 Operator Control and Indicators

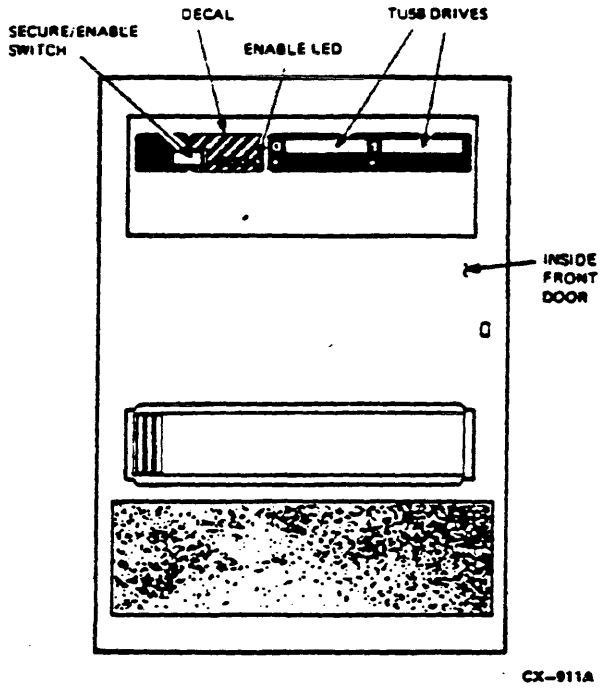
- Outside the HSC -- the Operator Control Panel (OCP) is the only means by which the user can interface with the HSC.
- Inside the HSC
  - The DC power switch removes power from the HSC backplane to enable board module removal. Cooling fans and main power supplies continue to operate.
  - The Secure/Enable switch disables/enables the front panel operation. When it is in the "secure" position, the HSC software ignores any switch changes on the OCP. However, fault light and fault codes are displayed normally. Also, the console terminal can be used with the SHOW commands of the SETSHO utility.



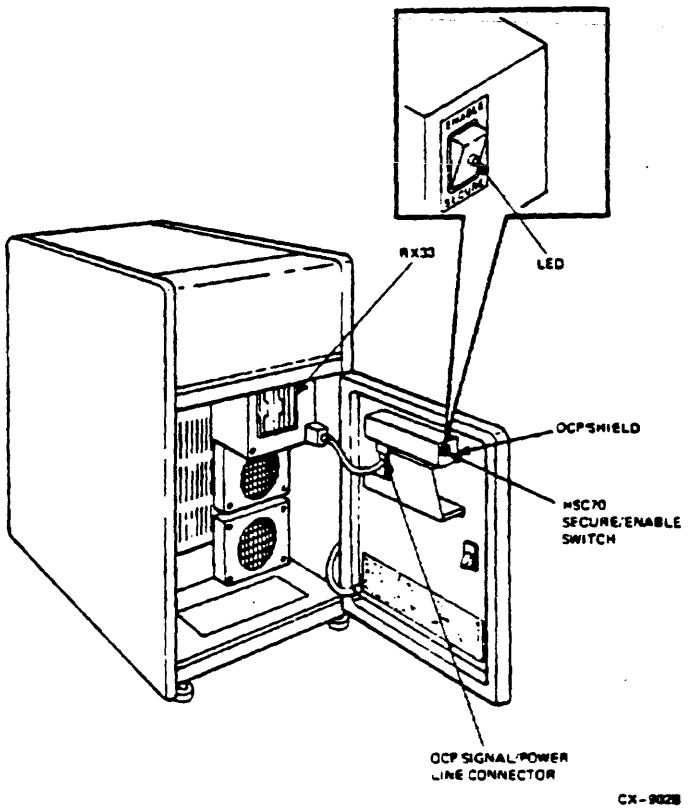
HSC50 DC Power Switch



HSC70 DC Power Switch

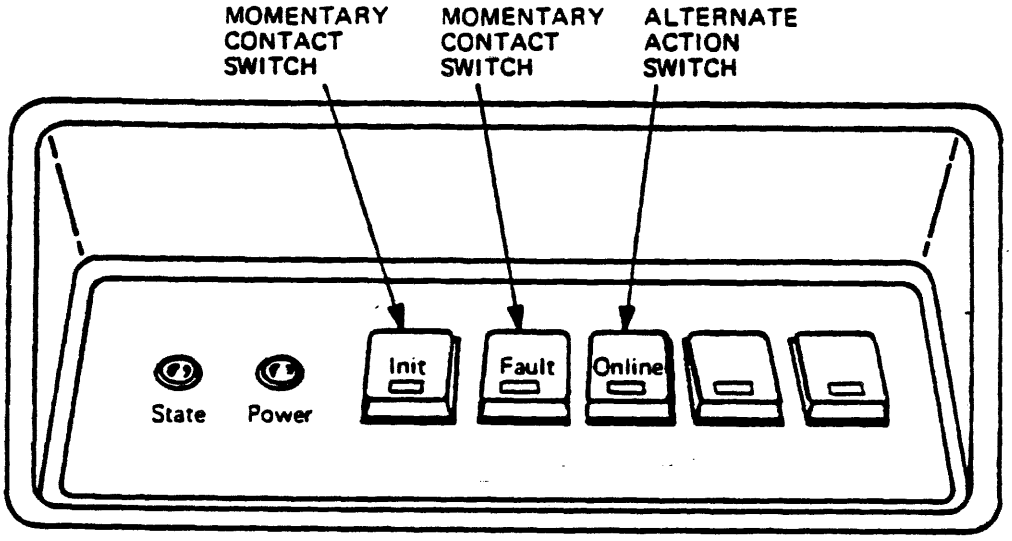


HSC50 Secure/Enable Switch



HSC70 Secure/Enable Switch

Operator Control Panel



CX–0088

- State and INIT indicators -- these indicators describe the state of the HSC. Under run-time conditions, the INIT indicator is off while the State indicator is pulsing. During HSC initialization (booting) these indicators change to reflect the current initialization phase of the subsystem.

- Power indicator -- the Power indicator is driven from a dc comparator circuit on the I/O control processor module. This circuit constantly monitors the +5, +12, and -5.2 voltages. If it detects a drop of approximately one-third in any of these three voltages, the indicator goes off.

ONLY SHOWS WHEN POWER IS PRESENT

  NOTE: **The power indicator does not mean these voltages are within specification of + 5 percent.**

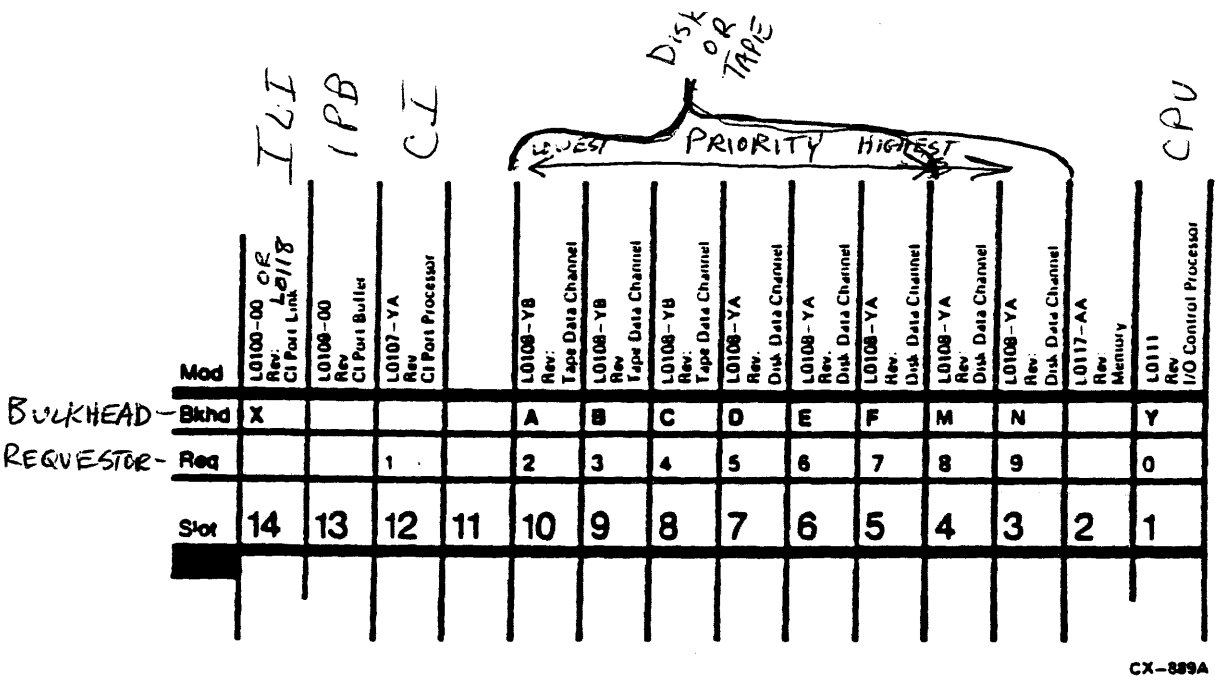
- Online indicator -- when the Online indicator is on, a virtual circuit is established between the HSC and a host CPU. When this indicator is off, no virtual circuits have been established with any host.

- Blank indicators -- the two Blank indicators form a part of the fault display to produce a 5-bit fault code.

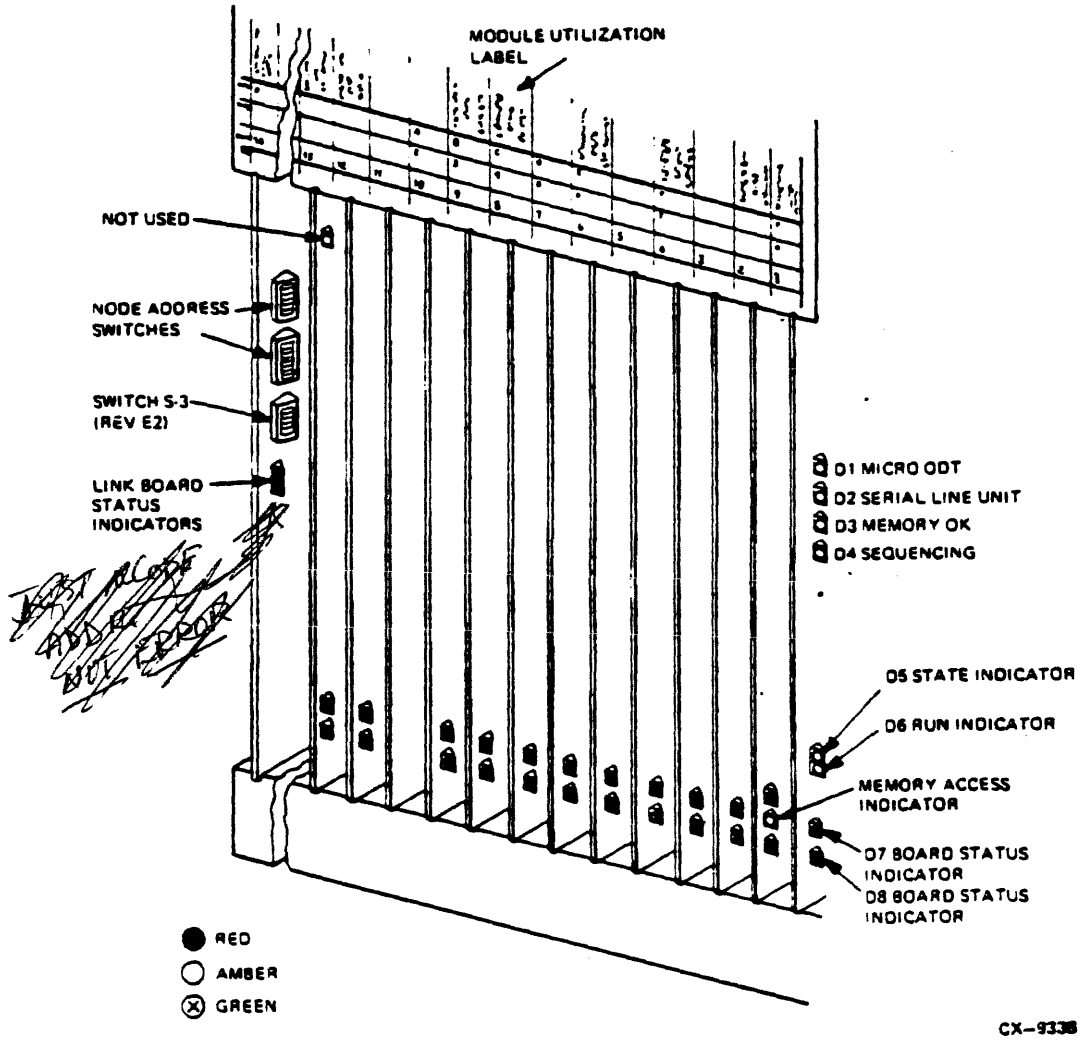
Operator Control Panel (Cont.)

- INIT switch -- this is a momentary contact switch that causes the HSC to initialize and reboot as if powering on.
  - Holding in the INIT switch engages the loopback mode for terminal testing.
  - The Secure/Enable switch must be in the enable position for the INIT switch to be operational.
- Fault switch -- the Fault indicator comes on when the HSC logic detects a fault, indicating a problem internal to the HSC.
  - Pushing and releasing this switch will yield a blinking 5-bit error code on the OCP indicators. These error codes will be covered in detail later.
  - Up to eight error codes can be displayed.
  - Used to create template System Configuration Table (SCT) BY HOLDING FAULT IN A WHILE IT REBOOTS (INIT TOO)
  - LAMP TEST
- Online switch -- when the Online switch is pushed-in, the HSC logic is in the available state. This allows a host to establish a virtual circuit with the HSC. When this switch is released, no new virtual circuits can be made; however, current virtual circuits are not broken. WORKS OK BUT WHILE OUT TO EXISTING NO

IF A SYS REBOOTS IT WILL NOT SEE IT WHEN IT COMES UP.



HSC70 Module Utilization Label Example

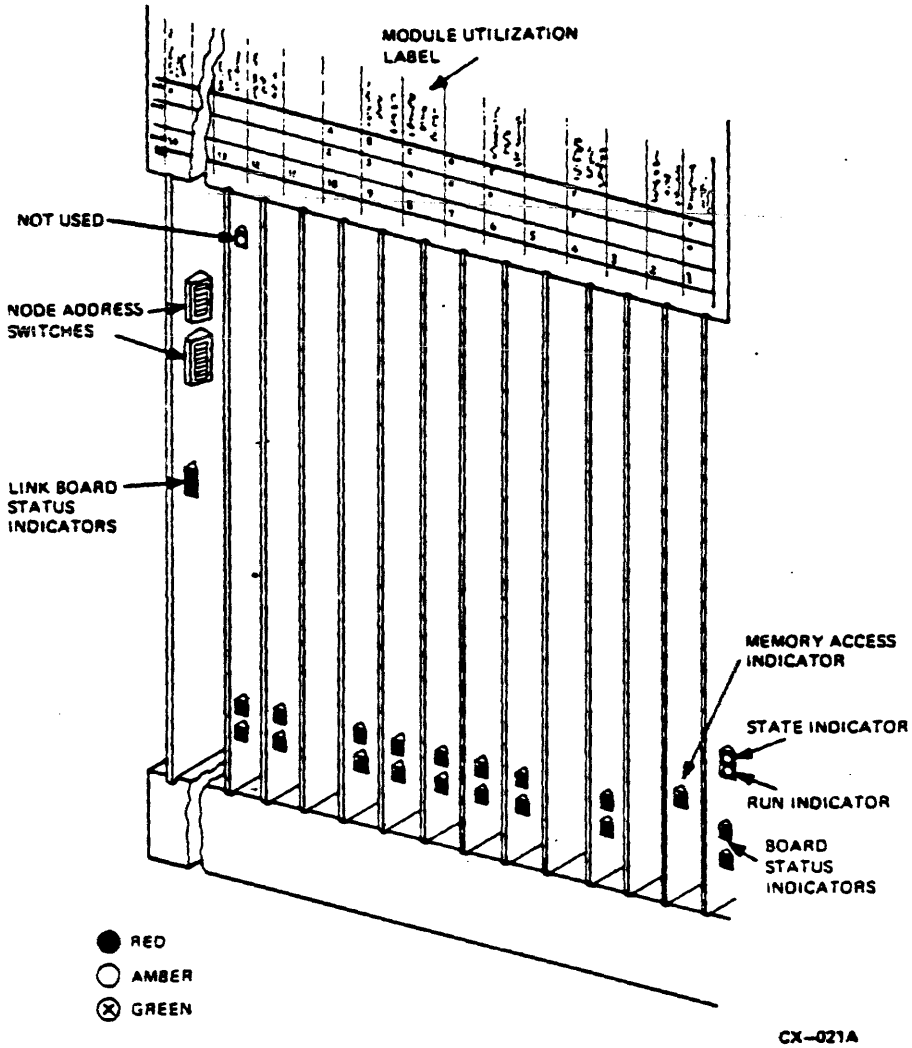


HSC70 Module Layout

Mod	L0100-00 Rev E CI Port Link	L0109-00 Rev CI Port Buffer	L0107-YA Rev CI Port Processor		L0108-YB Rev Tape Data Channel	L0108-YB Rev Tape Data Channel	L0119 Rev Data Channel		L0108-YA Rev Disk Data Channel		L0108-YA Rev Disk Data Channel		L0106-AA Rev Memory	L0105-00 Rev I/O Control Processor
Bkhd	X				A	B	C	D	E		F			Y
Req			1		2	3	4	5	6		7			0
Slot	14	13	12	11	10	9	8	7	6	5	4	3	2	1

CX-2838

HSC50 Module Utilization Label Example



HSC50 Module Layout

Module Layout Summary

Host Interface

- Called the K.ci.
- Interfaces the CI Bus with the HSC.
- Consists of the following three modules:
  - Port Processor module
  - Port Buffer module
  - Port Link module

Port Processor Module

- Called the K.pli.
- Moves command/message packets to/from the HSC control memory.
- Moves data packet to/from HSC data memory.
- Contains 2900-series microprocessor with onboard ROM.

Port Buffer Module

- Called the PILA module.
- Acts as high speed memory buffer for traffic to/from the CI Bus.

Port Link Module

- Called the LINK module.
- Implements packet transmission onto CI Bus and packet reception off CI Bus.
- Body of packet handed to PILA board.

Disk Data Channel Module

- Called the K.sdi.
- Monitors and controls disks attached to the HSC.

K.sti – NEW MODULE
CAN BE .sti OR .sdi (1 OR OTHER ONLY)
PROGRAMMABLE - DEPENDS
ON WHAT uCODE GETS LOADED.
COMPATABLE WITH ALL HSC's.

Module Layout Summary (Cont.)

- Each module can control up to 4 RA drives.
- Contains a 2900-series microprocessor with onboard ROM.

Tape Data Channel Module

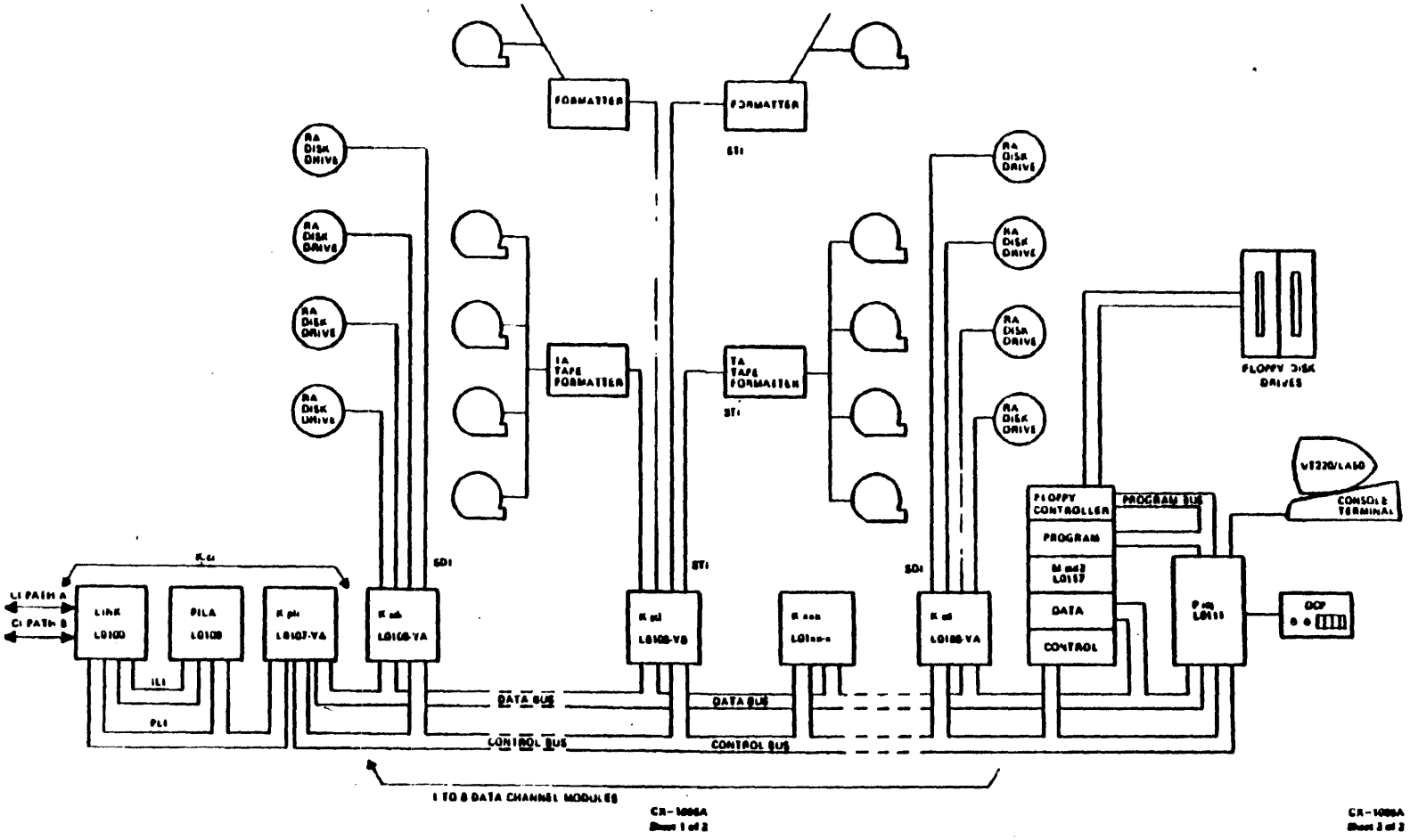
- Called the K.sti.
- Monitors/controls data transmission between the HSC and any attached tape drives.
- Contains a 2900-series microprocessor with onboard ROM.

Memory Module

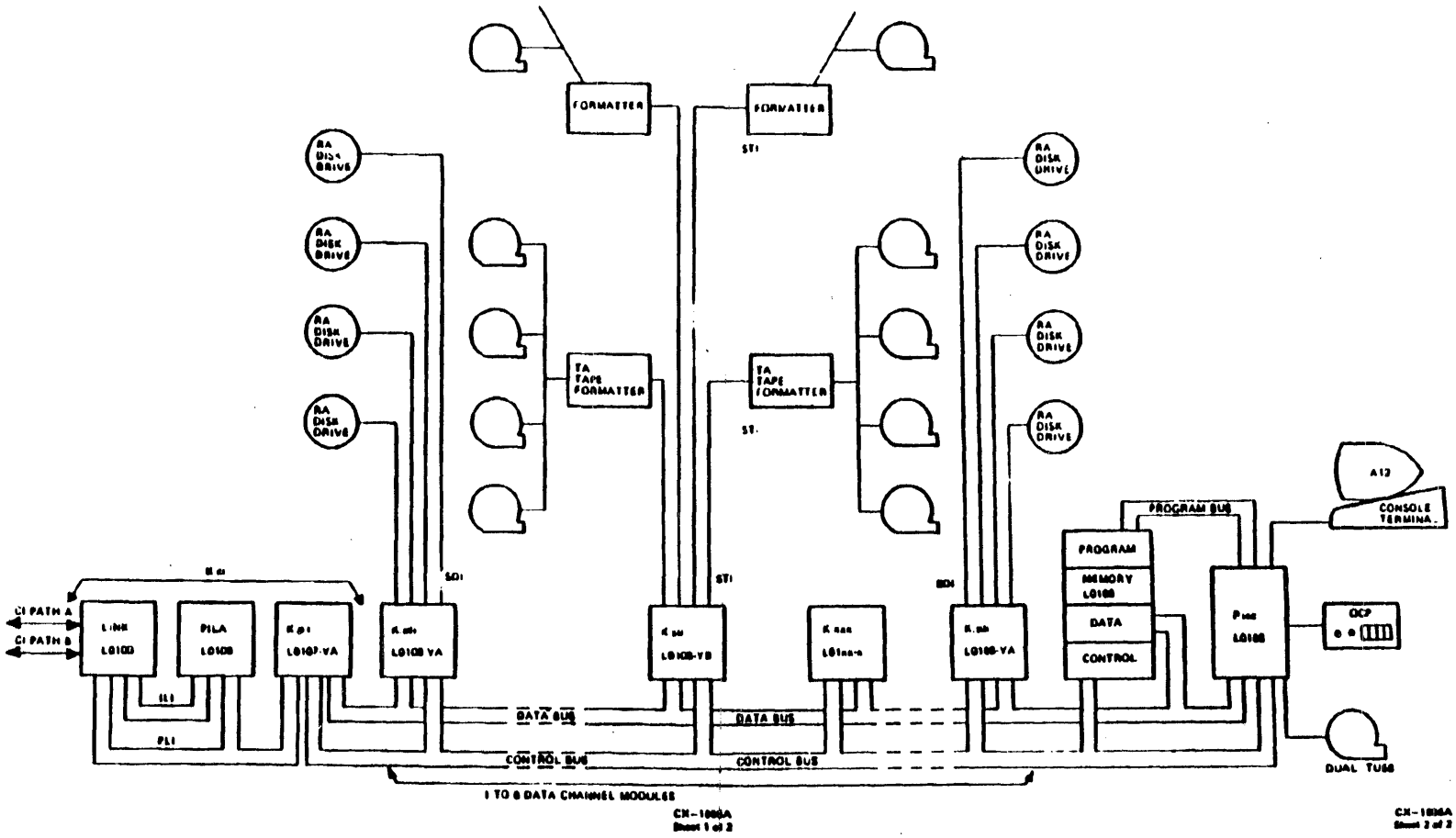
- Called the M.std on the HSC50.
- Called the M.std2 – 2MEG ? on the HSC70.
- Contains three separate and independent memory sections (data, program, and control).
- Considered to be HSC's main memory.

I/O Control Processor Module

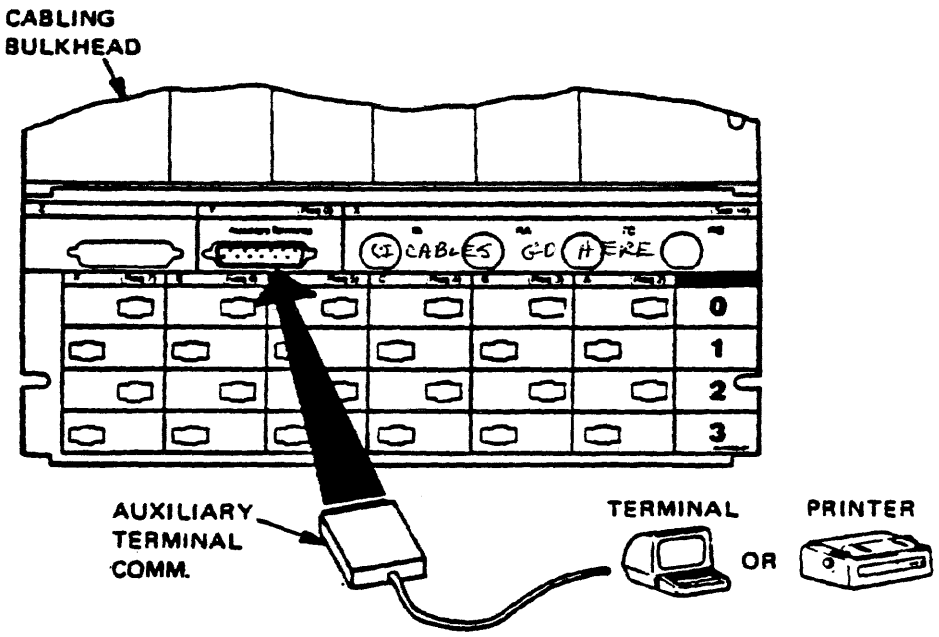
- Called the P.ioc on the HSC50.
- Called the P.ioj on the HSC70.
- Contains F-11 chip set (PDP-11/34) on the HSC50.
- Contains J-11 chip set (PDP-11/70) on the HSC70.
- Considered the main CPU of the HSC.



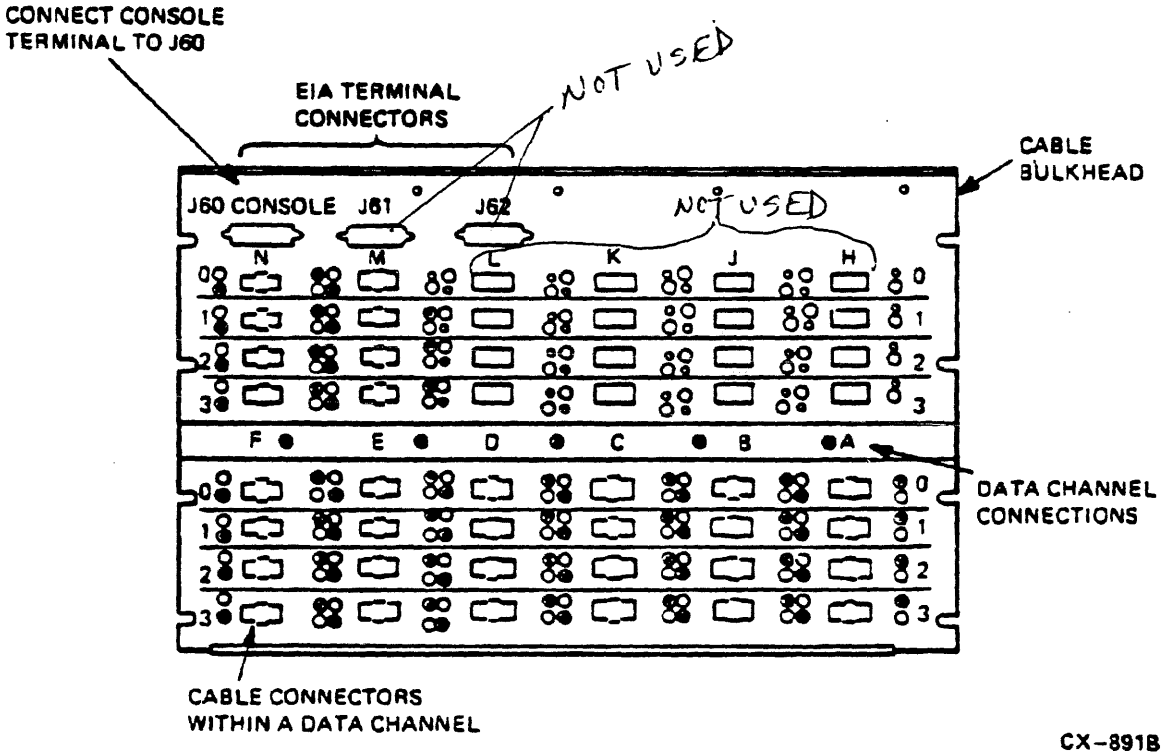
HSC70 Functional Block Diagram



HSC50 Functional Block Diagram



HSC50 Console Terminal Connection



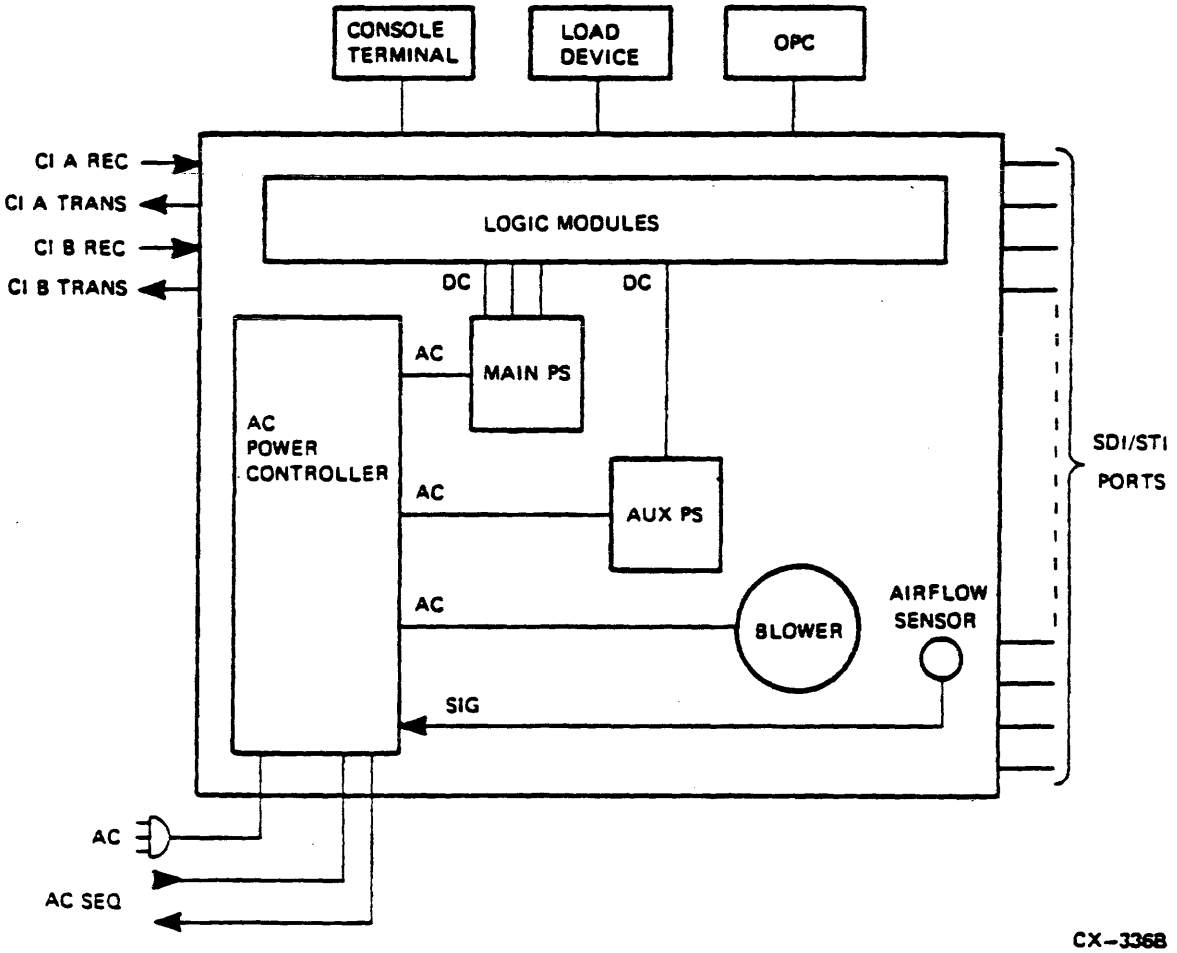
HSC70 Console Terminal Connection

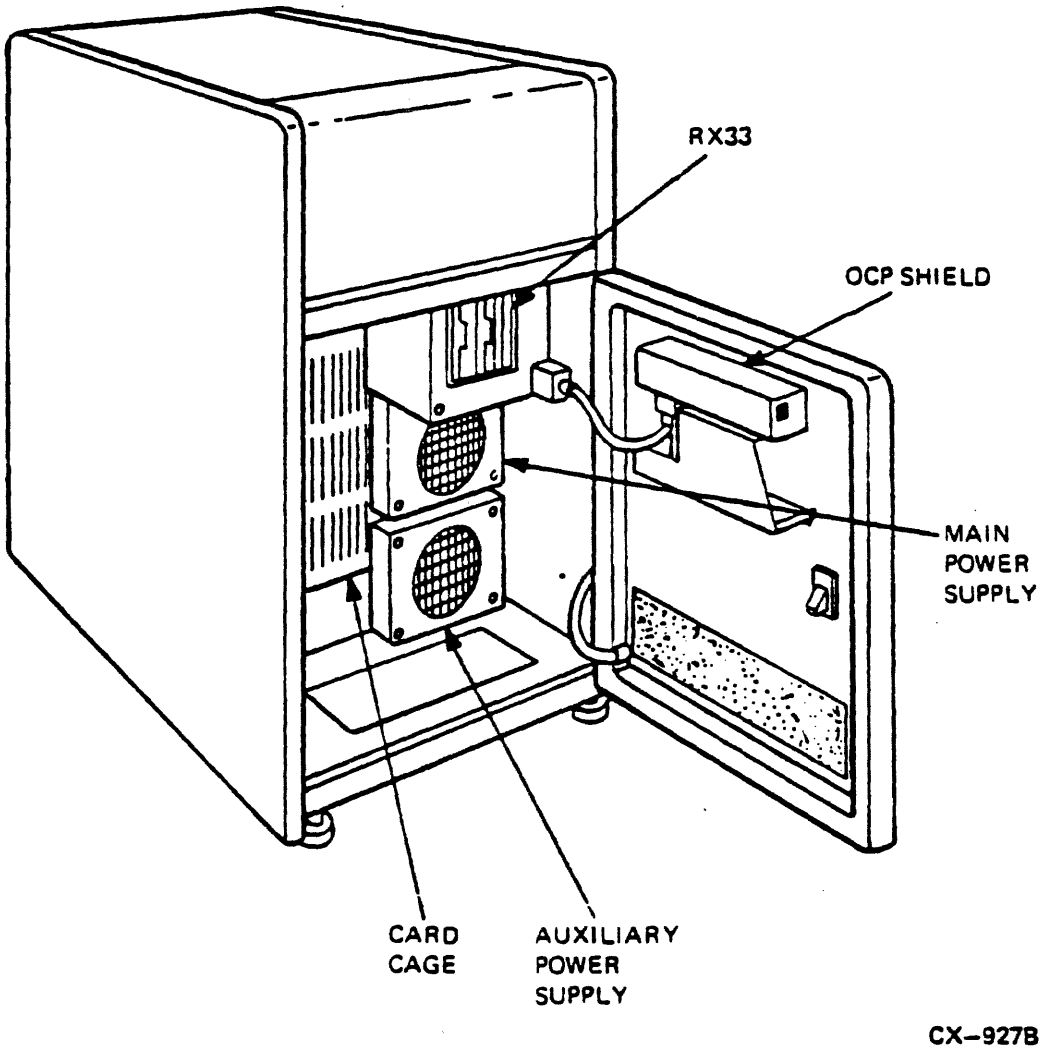
Power Distribution in the HSC

Just +5V

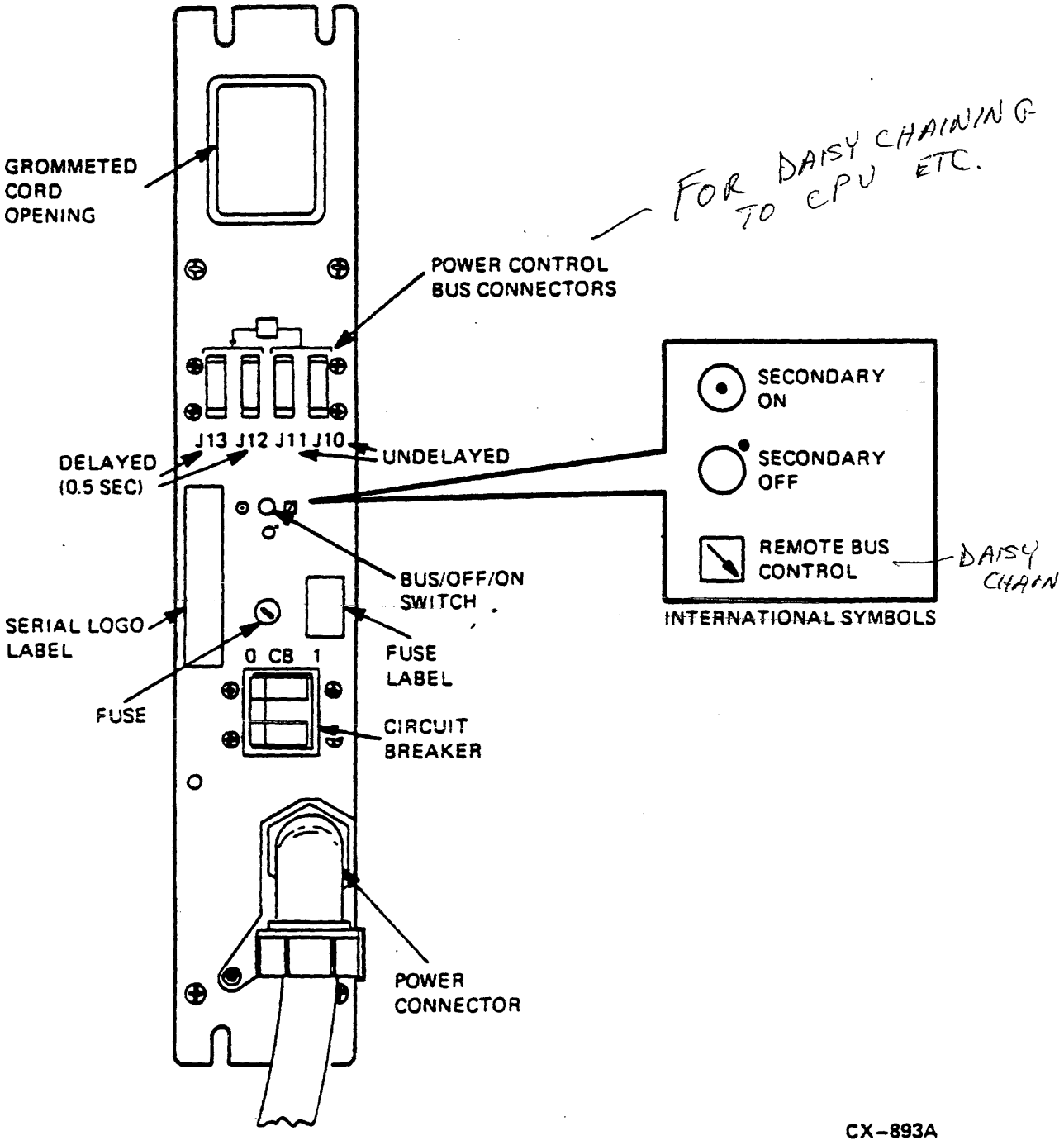
- HSC contains two power supplies: main supply and auxiliary supply.
- HSC50 always has a main supply and will have an auxiliary supply if more than eight modules are installed.
- HSC70 always has both supplies.
- No adjustments in any of the supplies.
- HSC provides following the voltages:

  + 12V
  - 5.2V
  + 5V

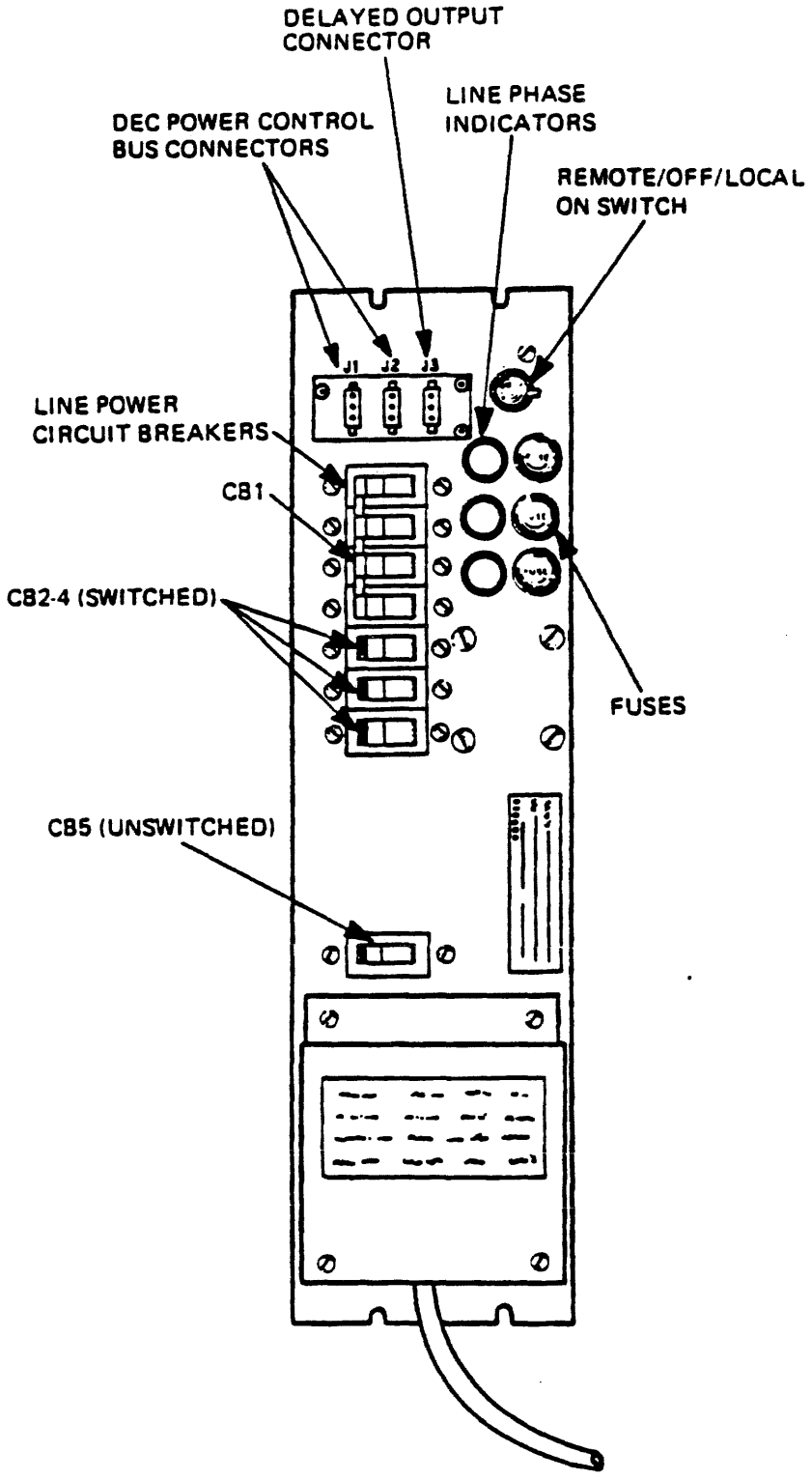




Location of Power Supplies in HSC

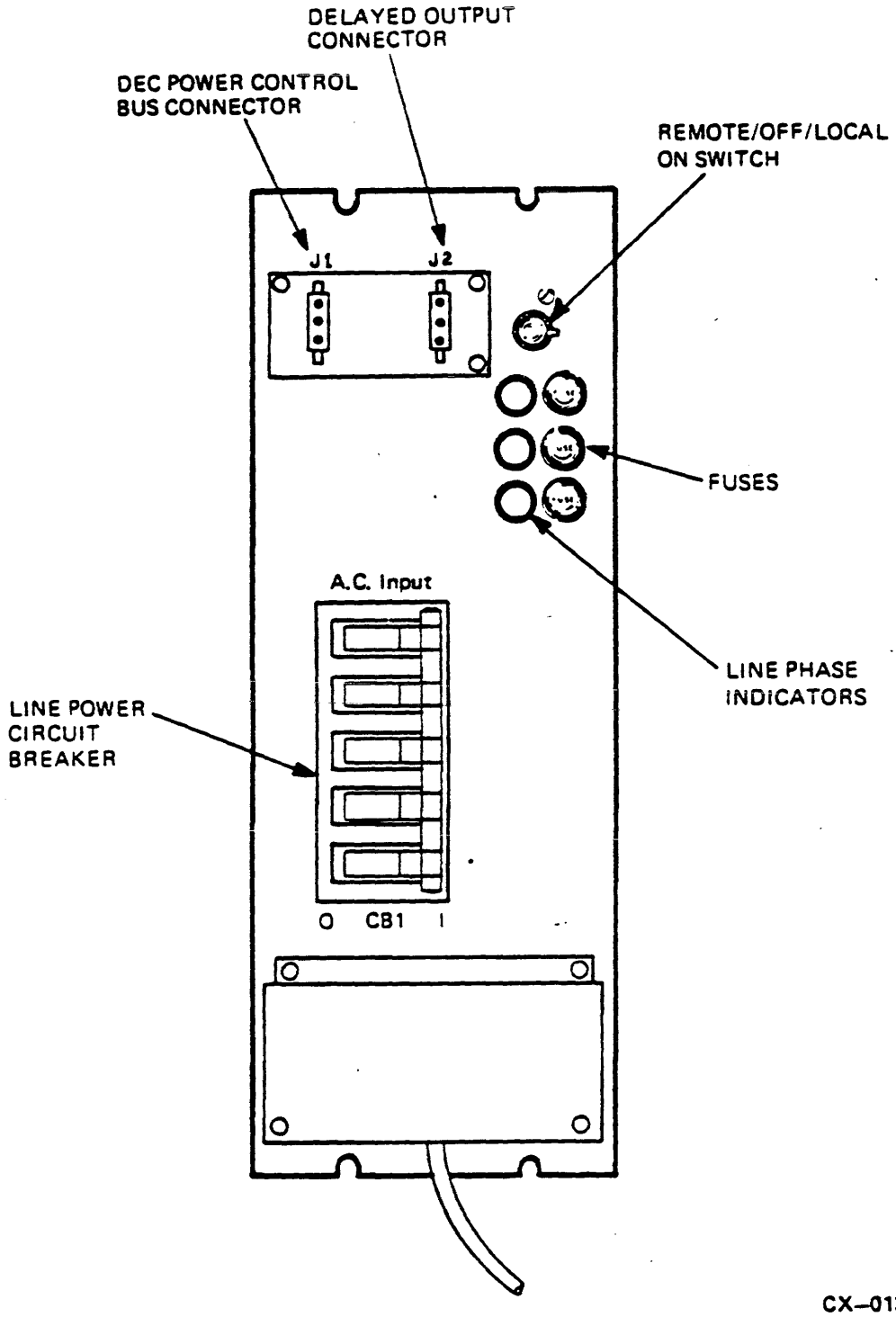


HSC70 AC Power Controller -- Front View



CX-013B

HSC50 AC Power Controller (120/208 Vac) -- Front View



HSC50 AC Power Controller (380-415 Vac) -- Front View

HSC MODULE DESCRIPTION

HSC Module Description

Lesson Introduction

This lesson discusses the modules used in the HSC50 and HSC70. The HSC contains three basic parts: the I/O processor, the Host interface, and the Data Channels.

The HSC50 and HSC70 use different I/O processors. The I/O processor for the HSC70 is called P.ioj . The I/O Processor for the HSC50 is called P.ioc. The rest of the modules are interchangeable between the two HSCs.

Lesson Objectives

1. Name each of the modules in the HSC and describe its function.

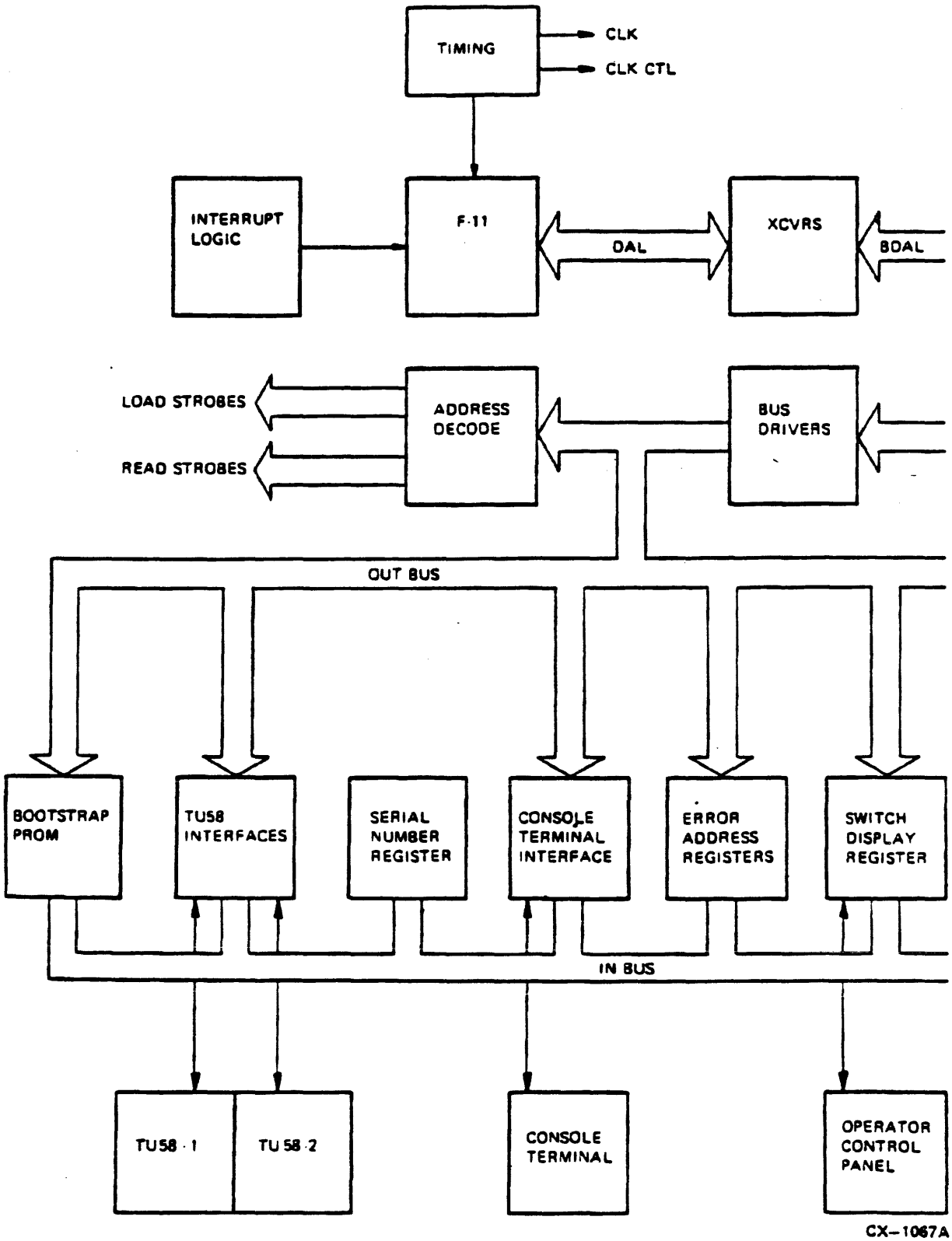
2. Describe the buses present in the HSC and how each is used.

Lesson Outline

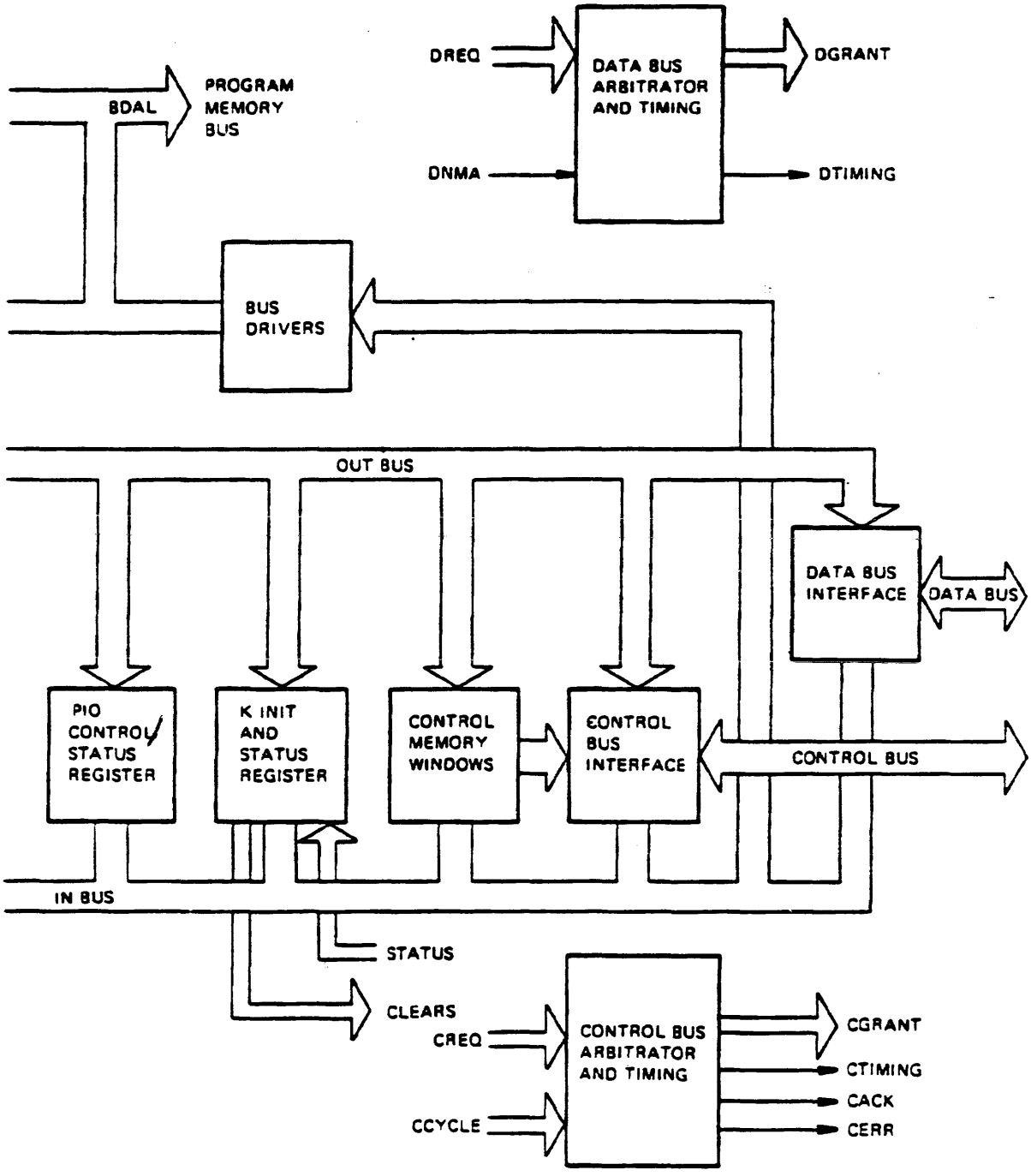
I. P.ioc
II. P.ioj
III. M.std
IV. M.std2
V. Data Channel
VI. K.ci
VII. L0118

Input/Output Control Processor (HSC50) P.ioc L0105

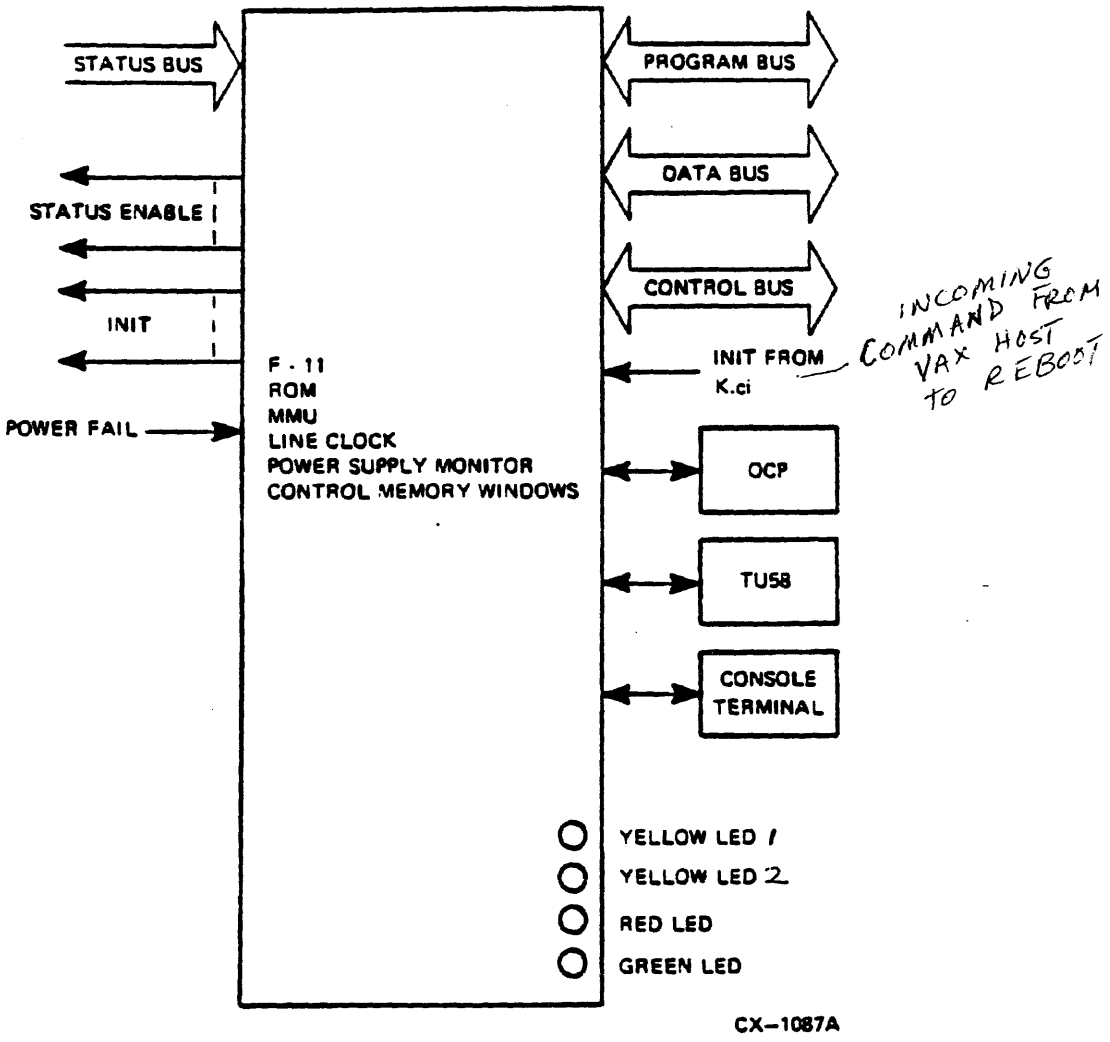
- The P.ioc performs high-level control of the HSC50 and is similar to a CPU in a computer system.
- An F-11 chip set performs the standard PDP-11/34 instruction set.
- P.ioc contains the following:
  - Memory management unit (MMU) for 16-to-22 bit translation.
  - ROM for booting and self-tests.
  - Fixed frequency line clock.
  - Control and data bus arbitration logic.
  - Parity generation and checking for the three memories.
- The P.ioc directly interfaces to the following devices:
  - Operator control panel.
  - Secure/Enable switch.
  - TU58 controller.
  - Local console terminal (LA12, VT220).
  - Status bus for each K.
- The Program bus belongs exclusively to the P.ioc; it is dedicated to system program execution.



HSC50 Input/Output Control Processor Module Block Diagram (Sheet 1 of 2)



HSC50 Input/Output Control Processor Module Block Diagram (Sheet 2 of 2)



Input/Output Control Processor Module (HSC50)

LED Indicators on P.ioc

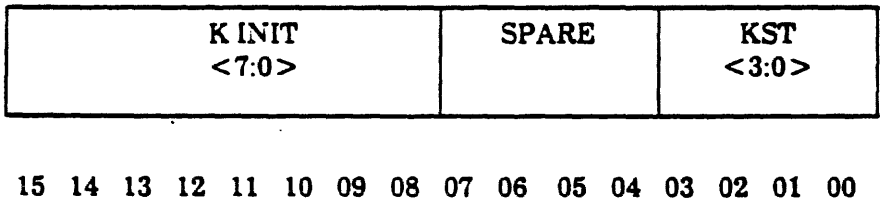
Yellow 1	Blinks under normal operation and is tied directly to the State indicator on OCP.
Yellow 2	Run indicator that blinks each instruction fetch of the F-11. NORMALLY DIM
Red	Indicates module has not passed self-test.
Green	Indicates module has passed self-test.

Detailed Block of P.ioc (HSC50)

- The F-11 processor consists of three chips:
  - The Data Chip (DC302) performs ALU functions and handles data/address transfers onto DAL bus.
  - The Control Chip (DC303) performs micro-program sequencing for PDP-11 instruction decode and contains Control Store ROM.
  - The MMU Chip (DC304) contains registers for 22-bit memory addressing.
- A 16-bit virtual address enters on DAL, is relocated by MMU to 22-bit physical address, and returned to DAL for transmission to Program Memory bus.
- Address Decode provides read and load strobes for various I/O registers on P.ioc.
- Bootstrap PROM is 1Kword in length:
  - Starting address (cold start) = 17773000
  - Starting address (warm start) = 17773010
- TU58 Interface provides parallel-to-serial and serial-to-parallel connection to both load devices.
  - DDO occupies 17777520--17777526
  - DD1 occupies 17777530--17777536
  - No parity check done on either interface
  - Baud rate selectable on TU58 controller
- Console Interface is a DL-11D containing four registers:
  - Receiver Status Register (RCSR) -- 17777560
  - Receiver Buffer Register (RBUF) - -17777562
  - Transmit Status Register (XCSR) -- 17777564
  - Transmit Buffer Register (XBUF) - -17777566

Detailed Block of P.ioc (Cont.)

- The Serial Number Register is loaded from a set of jumper switches during SCT INIT; located at 17770056.
- The Error Address Registers latch a 22-bit address from the out bus on parity or NXM trap.
- The Switch/Display Register is located at 1777042
- The PIO Control/Status Register is located at 17770040
- The K INIT and Status Register:
  - K INIT = 17770044
    K Status = 17770046
  - Zero in K INIT Register (15:08) hangs the appropriate K, then a one frees K to begin microdiagnostics.



CX0-1140A

  - Bits 3:0 of K INIT decoded to select backplane slot (K) for status return.
- Control Memory Windows are needed to access control blocks scattered throughout Control Memory.
  - MMU is normally not used to access Control Memory.
  - Changes 16-bit address to 17-bit address.

Detailed Block of P.ioc (Cont.)

- The memory map of HSC50 is as follows:

22-BIT ADDRESS ALLOCATION

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

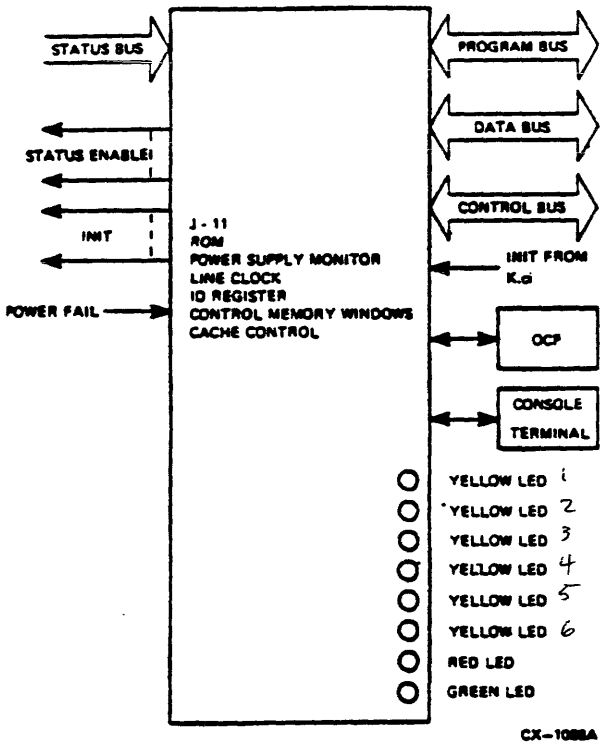
CX-338B

= undefined or unused areas of memory

Input/Output Control Processor (HSC70) P.ioj L0111

- The P.ioj performs high-level control of the HSC70 and is similar to a CPU in a computer system.
- A J-11 chip set performs the standard PDP-11/70 instruction set. The module also contains the following:
  - 8 Kbyte instruction cache.
  - ROM for booting and self-tests.
  - Fixed-frequency line clock.
  - Control and data bus arbitration logic.
  - Parity generation and checking for the three memories.
- The P.ioc directly interfaces to the following devices:
  - Operator control panel.
  - Secure/Enable switch.
  - Local console terminal (LA12, VT220).
  - Status bus for each K.
- The Program bus belongs exclusively to the P.ioj; it is dedicated to system program execution.

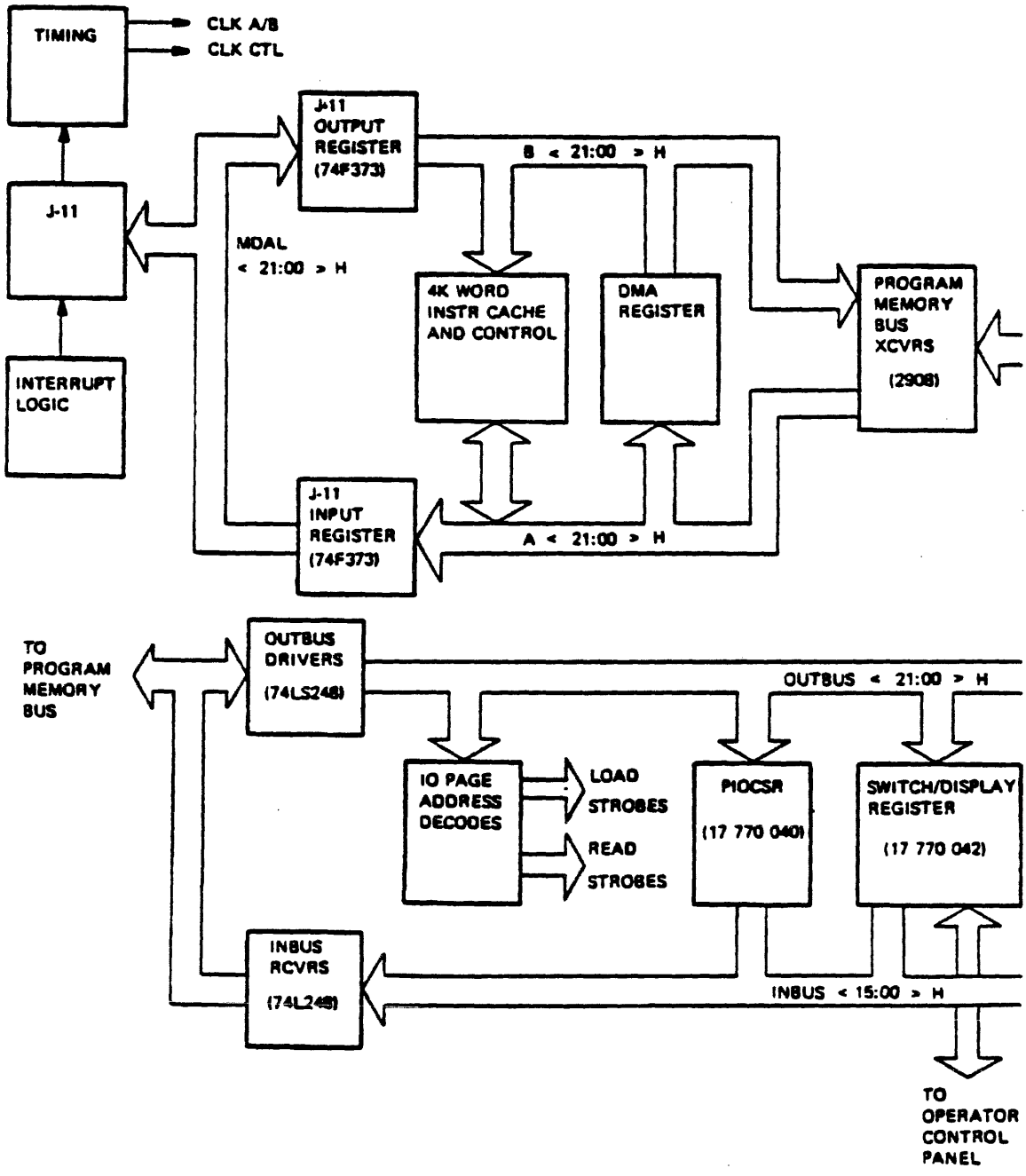
NO LONGER TALKS TO CONSOLE MEDIA DIRECTLY (AS IN 50) MEDIA TIED TO MEM.



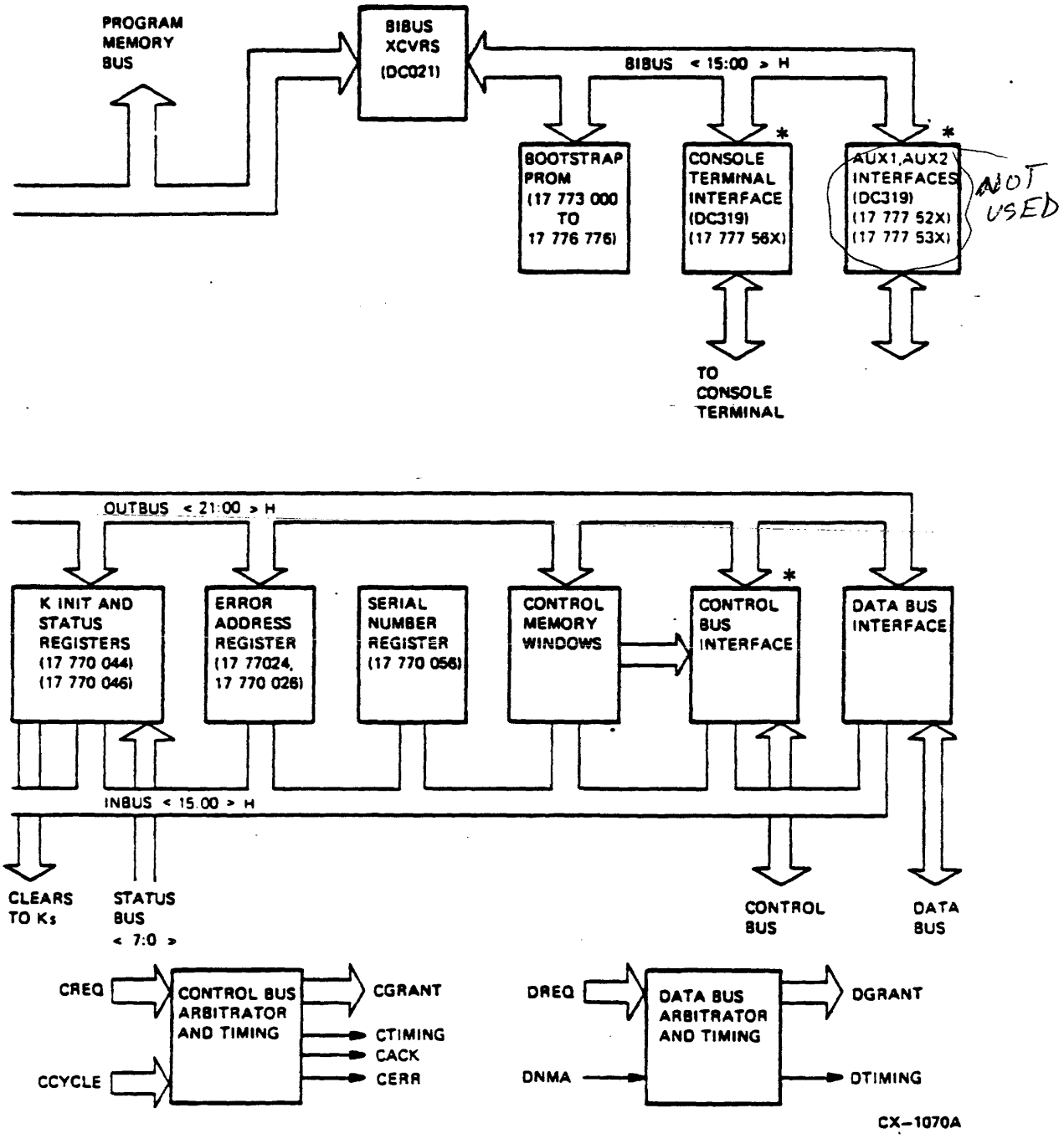
Input/Output Control Processor Module (HSC70)

LED Indicators on P.ioj

1 Yellow (top)	1ST OFF	On at power up and then turned off after internal J-11 sequencer test is done (micro-ODT). QUICK FLASH
2 Yellow	4TH OFF	On at power up and then turned off after check for presence of console terminal.
3 Yellow	3RD OFF	On at power up and then turned off after check for Program Memory space.
4 Yellow	2ND OFF	On at power up and then turned off; J-11 internal sequencer test.
5 Yellow		State indicator that blinks under normal operation and is tied directly to the State indicator on OCP.
6 Yellow		Run indicator that blinks each instruction fetch of J-11.
~~Red~~ RED		Indicates module has not passed self-test or module testing is in progress.
Green		Indicates module has passed self-test and operating software is running.



HSC70 Input/Output Control Processor Module Block Diagram (Sheet 1 of 2)



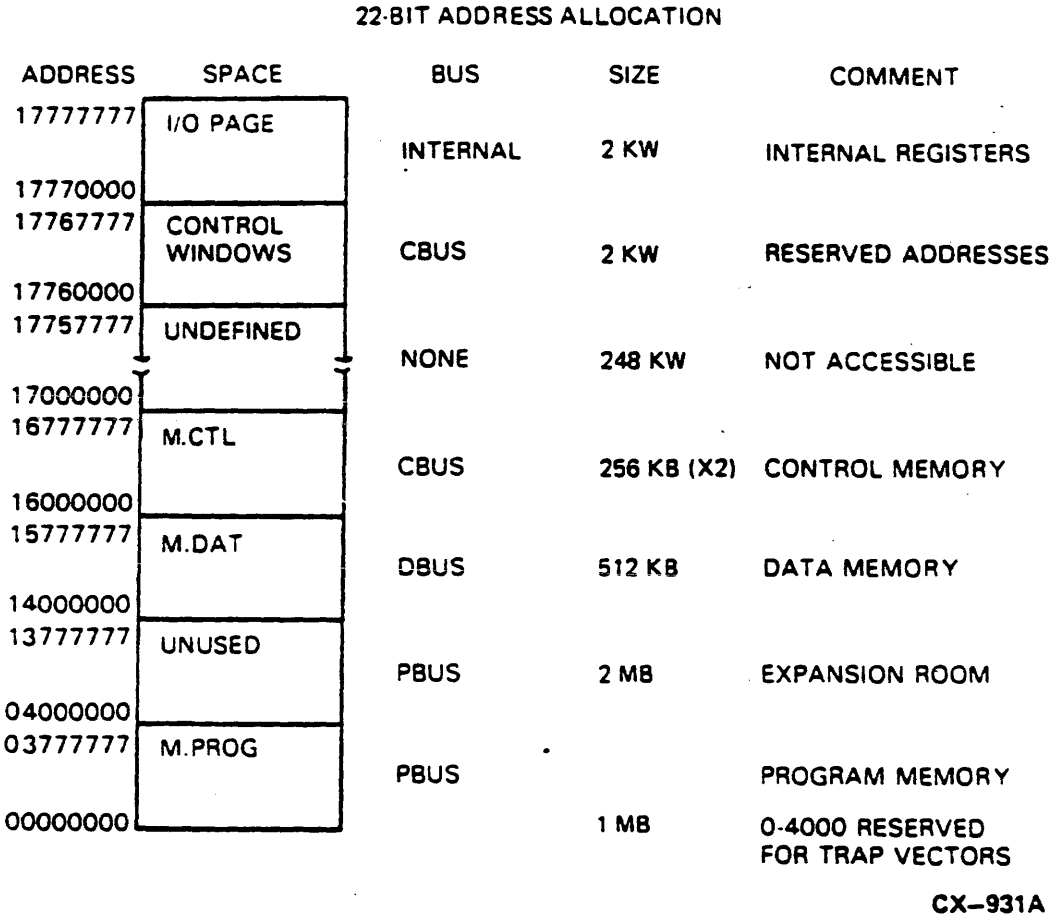
HSC70 Input/Output Control Processor Module Block Diagram (Sheet 2 of 2)

Detailed Block of P.ioj (HSC70)

- J-11 contains the same MMU as F-11 for 22-bit addressing of Program Memory. * MEM. MNGMNT. DONE ON CPU BOARD!
- The 4K Word Instr Cache is used only by Program Memory for faster execution:
  - On each memory access, instruction word plus next word are buffered in cache
  - On next instruction fetch (usually next consecutive word,) the instruction is found in cache (hit).
- The DMA Register related to cache operation:
  - Loaded with address when write is about to be done to Program Memory (loading from [redacted]RX33).
  - DMA address compared with that in cache; if identical then cache location is invalidated.
  - Now cache locations are identical to those in Program Memory.
- Bootstrap PROM is 2 Kwords in length but only 1 Kword is used at power-up.
  - If an error occurs in the first 1 Kword (page) then P.ioj can use the second page to boot.
  - Starting address = 17773000
- The Console Terminal Interface standard EIA port defaults to 9600 baud; uses identical registers and addresses as P.ioc of HSC50.
- The I/O Page Address Decodes generate read and load strobes to various registers in the I/O page of HSC70 memory.
- The PIO CSR is the at same address as P.ioc on HSC50.
- The Switch/Display Register is the same as on HSC50.
- K INIT and Status Register is the same as on HSC50 except ten requester codes (instead of eight) can be returned.

Detailed Block of P.ioj (Cont.)

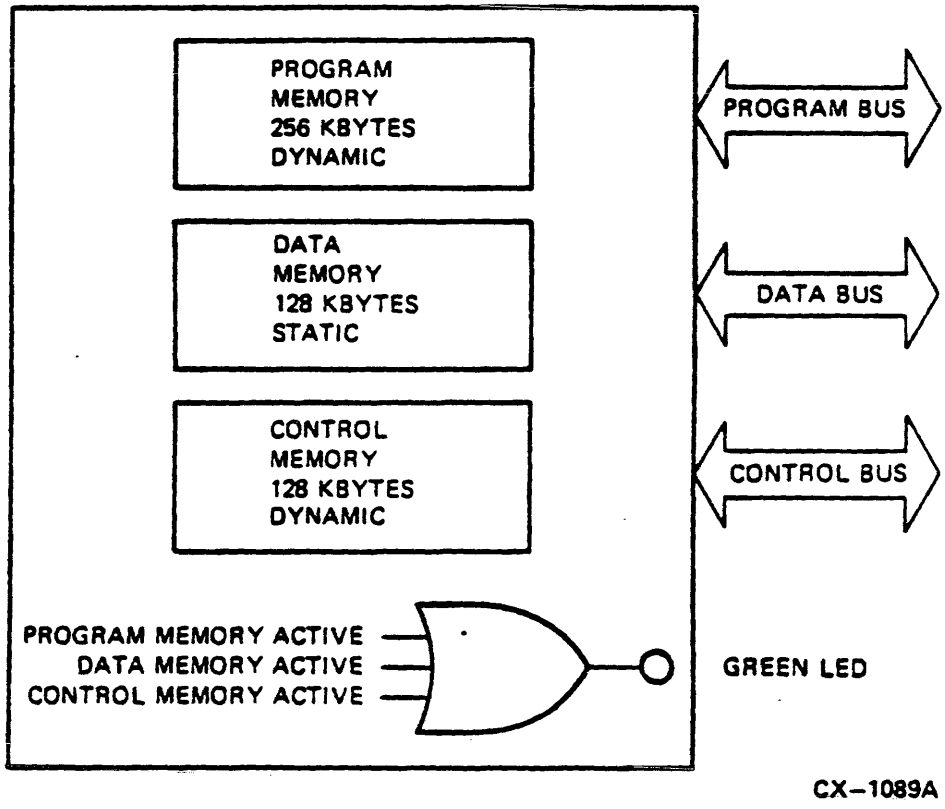
- The Error Address Register is the same as on HSC50.
- The Serial Number Register is the same as on HSC50.
- The Control Memory Windows are the same as on HSC50.
- A Memory Map of HSC70 is as follows:



1/2 MEG

Memory Module (HSC50) M.std L0106

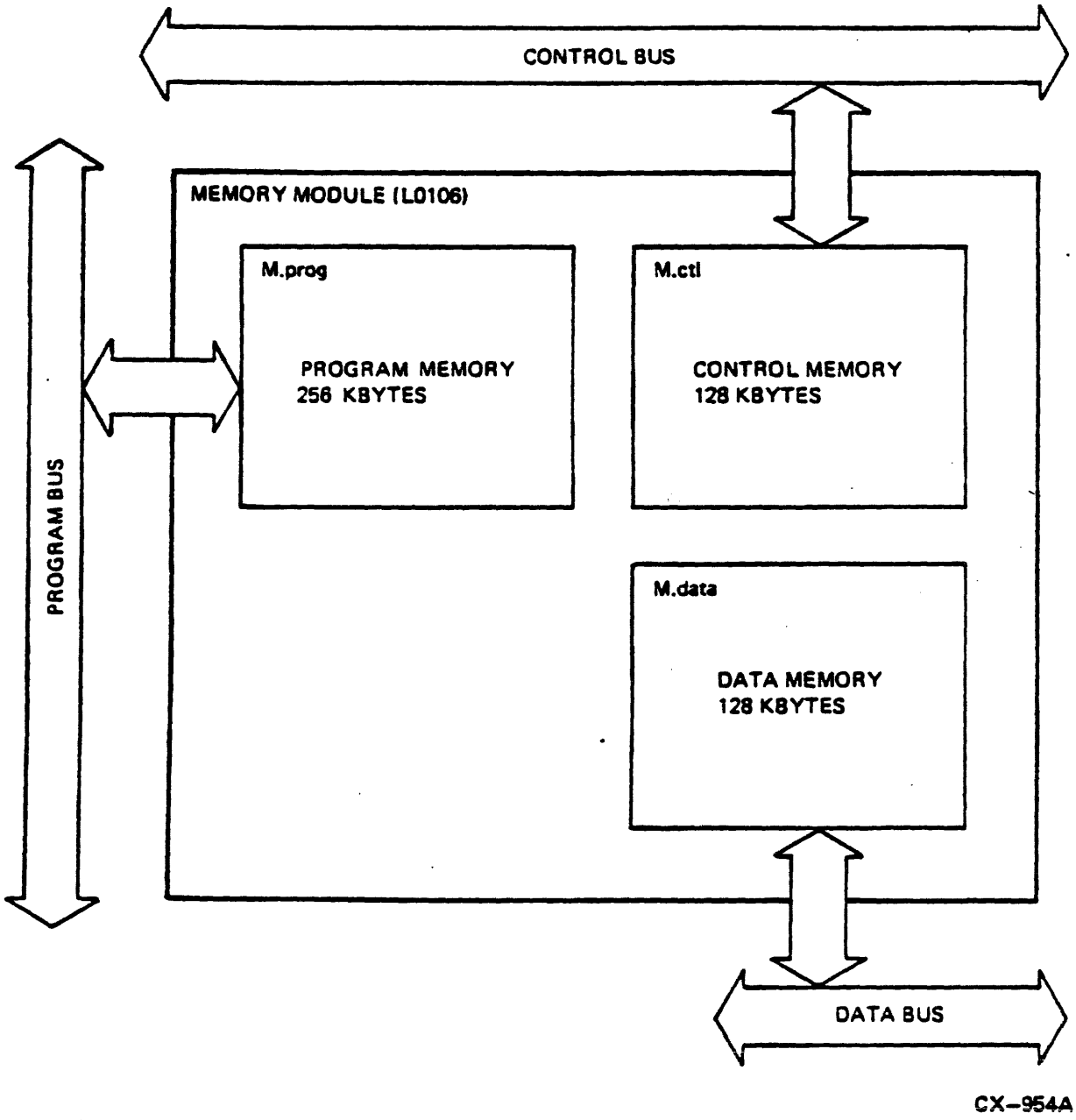
- The M.std board is divided into three independent sections:
  - Program Memory
    a. Contains system software executed by the P.ioc board.
    b. Loaded from the TU58.
    c. Contains 256Kb of dynamic RAM.
  - Control Memory
    a. Contains control structures set up by the system software and shared by the other K boards.
    b. Contains 128Kb of dynamic RAM.
  - Data Memory
    a. Provides sector size (512 byte) buffers for all data transfers through the HSC.
    b. Contains 128Kb static RAM. — NO REFRESH!



Memory Module (HSC50)

LED Indicators on M.std

Green	Memory access taking place.



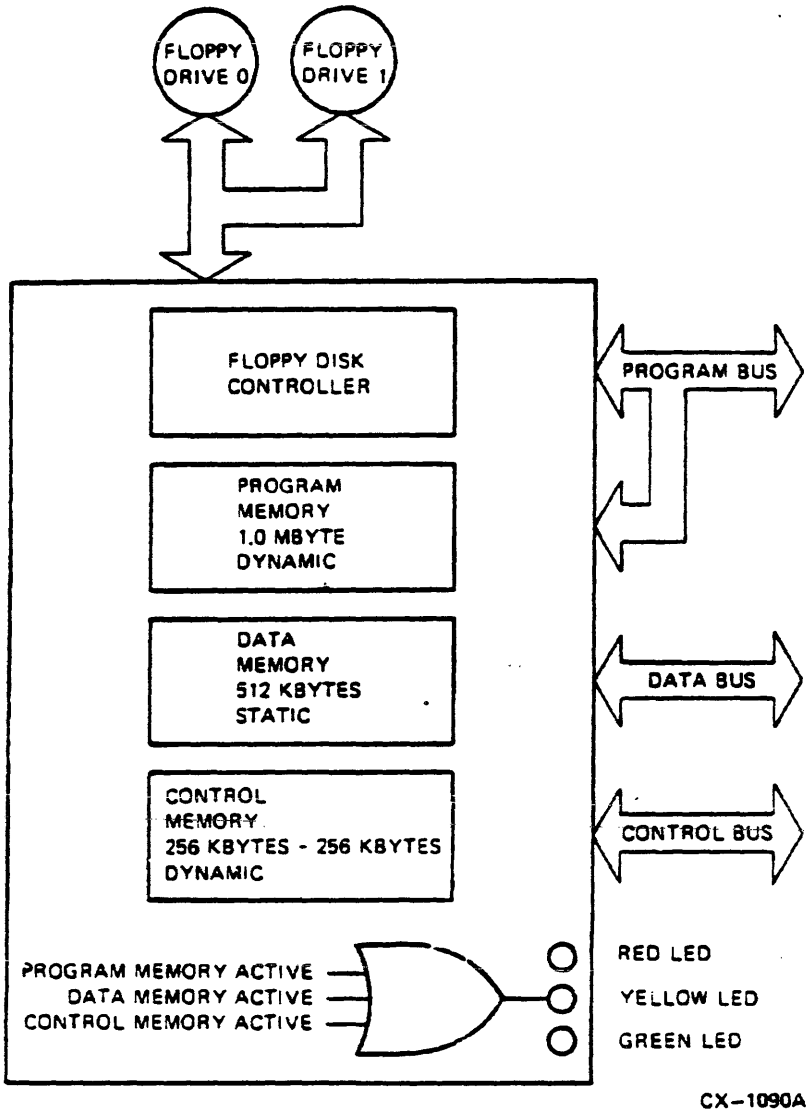
HSC50 Memory Module Block Diagram

Detailed Block of M.std (HSC50)

- M.std has no self-test:
  - Program Memory is tested by P.ioc.
  - Data and Control Memory are tested by one of the K boards.
- No parity checking is done on the memory board (it is done on P.ioc).
- Refresh is done on Control and Program Memory every 15 usec.
- Program Memory is divided into two banks of 128Kb each; on powerup either bank can be selected.
- Data Memory is organized into four banks of 18 RAMS (each RAM = 16K × 1):
  - Sixteen data bits.
  - Two parity bits.
- Control Memory is organized as one bank of 18 RAMS (each RAM = 64K × 1):
  - Sixteen data bits.
  - Two parity bits.

Memory Module (HSC70) M.std2 L0117

- The M.std2 module is divided into three independent sections:
  - Program Memory
    a. Contains system software executed by P.ioj.
    b. Loaded from RX33 floppy drive.
    c. Contains 1Mb of dynamic RAM.
  - Control Memory
    a. Contains control structures set up by system software and shared by the other K boards.
    b. Contains 512Kb of dynamic RAM
  - Data Memory
    a. Provides sector size (512 byte) buffers for all data transfers through HSC.
    b. 512Kb of static RAM. — NO REFRESH



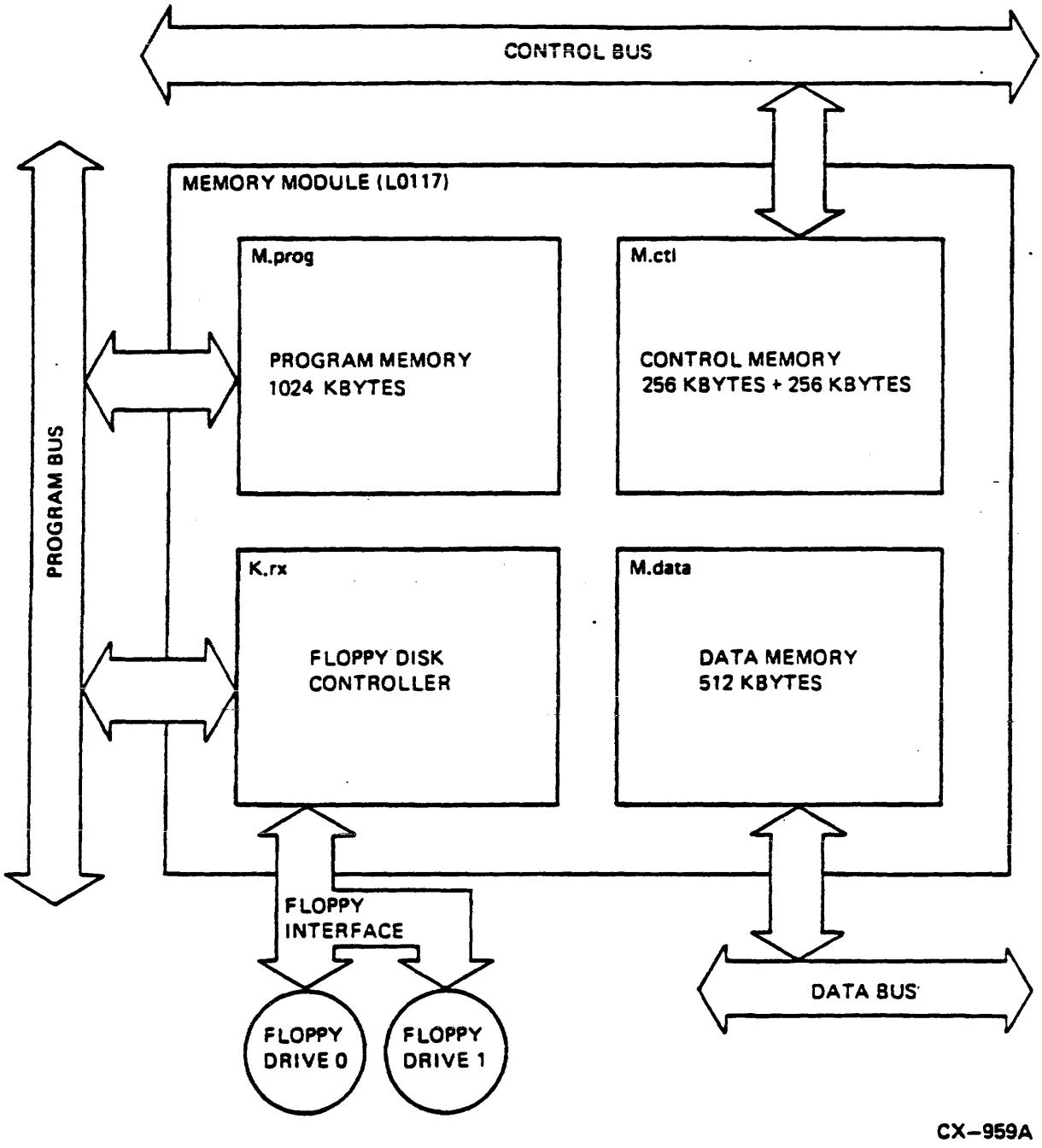
Memory Module (HSC70)

LED Indicators on M.std2

Red (top)	On and then off during initialization by P.ioj or during testing performed by a K.
Yellow	Memory cycle taking place. ~ ACTIVITY
Green	Testing has successfully finished and software is running.

Details of M.std2 (HSC70)

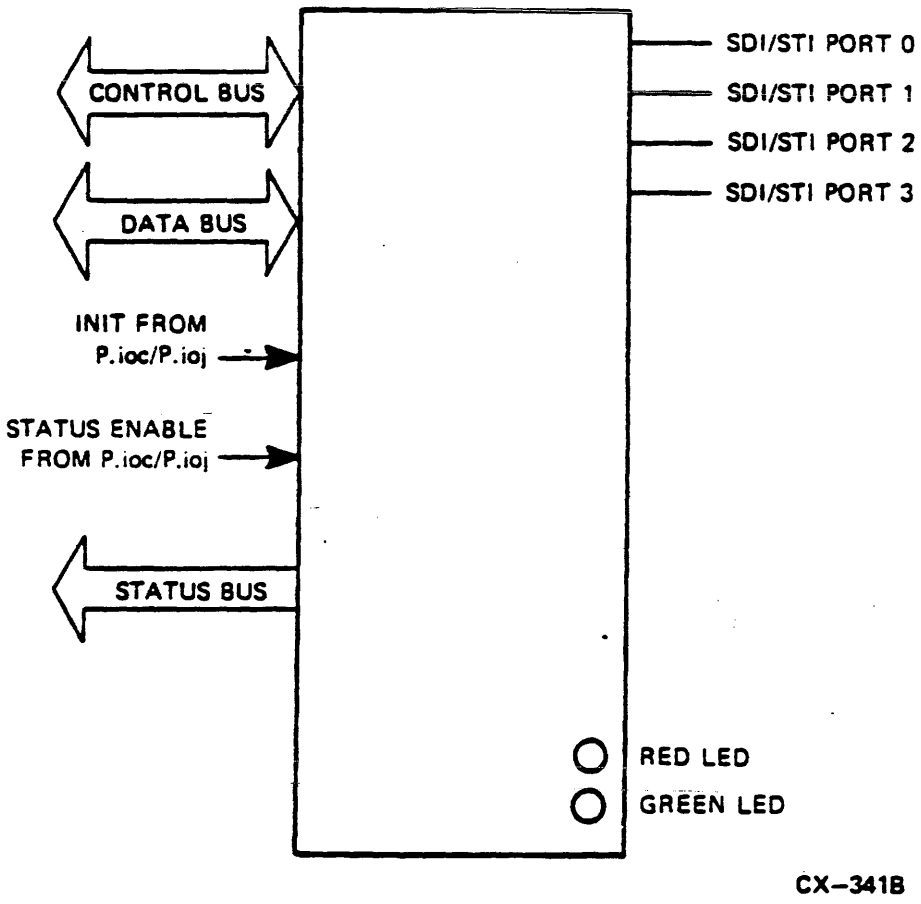
- M.std2 has no self-test:
  - Program Memory is tested by P.ioj.
  - Data and Control Memory is tested by one of the K boards.
- No parity checking is done on the memory board (it is done on P.ioj).
- Program Memory is divided into two banks of 512 Kb each; on power-up only one of the banks is selected.
- Floppy Disk Controller (FDC) logic:
  - DMAs done between the diskette and Program Memory.
  - Minimum data transfer length is one sector (512 bytes).
  - Maximum data transfer length is 15 sectors (1 track).
  - Byte parity checked for all data transfers.



HSC70 Memory Module Block Diagram

Disk Data Channel K.sdi L0108-YA

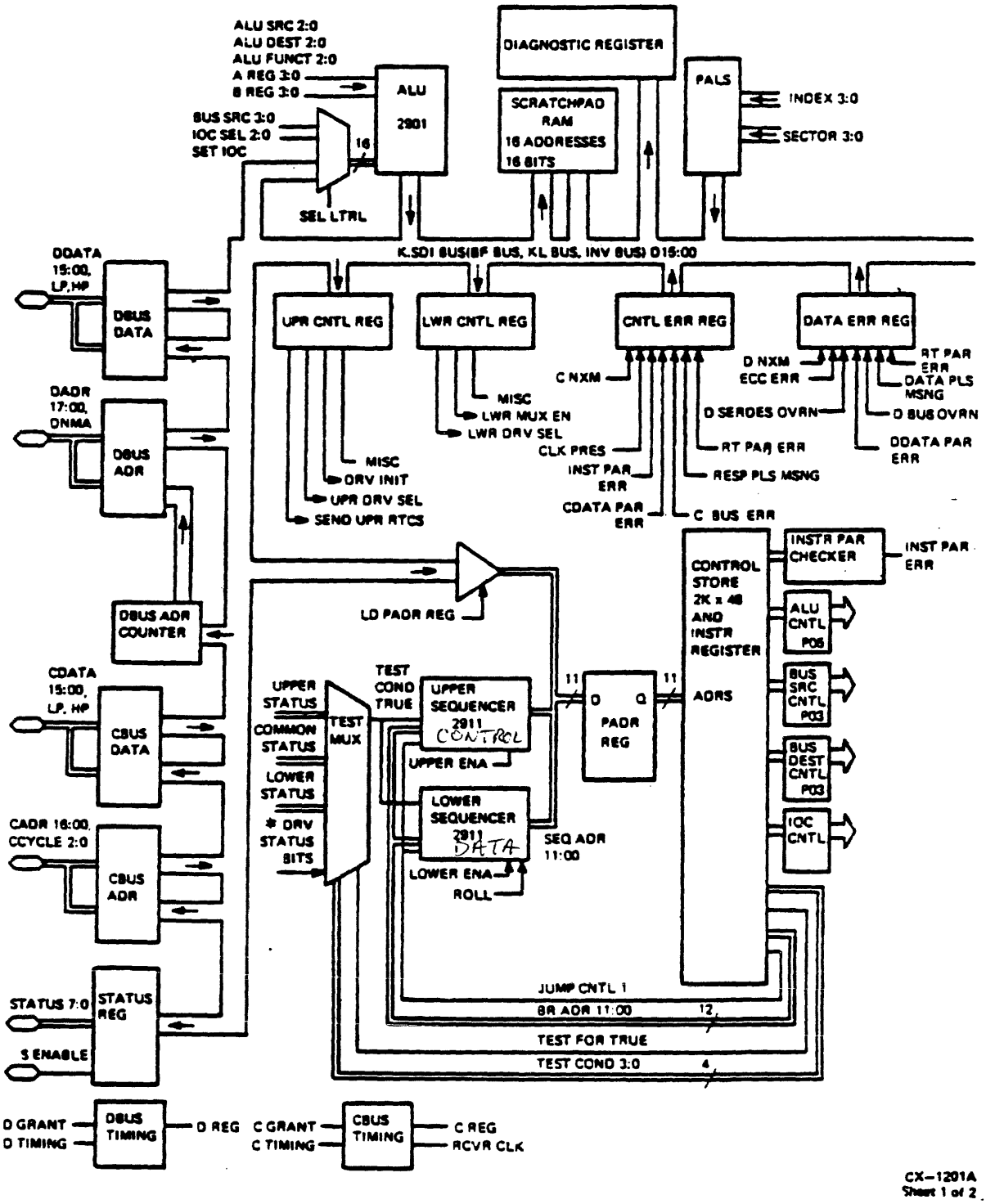
- The K.sdi module is the interface between the HSC system software and the disks attached to the HSC.
- The K.sdi module is identical to the Tape Data Channel module except for different ROM chips and one jumper.
- The K.sdi module has the following characteristics:
  - Supports up to four RA disk drives.
  - Monitors rotational position drive.
  - Transfers sector size data (512 bytes) between an RA drive and Data Memory.
  - Transfers control/status information between the RA drive and Control Memory.
  - Checks EDC on all transfers.
  - Checks/generates ECC error code.
- The K.sti module has the following characteristics:
  - Supports four formatters (sixteen TA78s or four TA81s).
  - Transfers data between Data Memory and the formatter.
  - Generates an EDC during a write operation to the formatter.



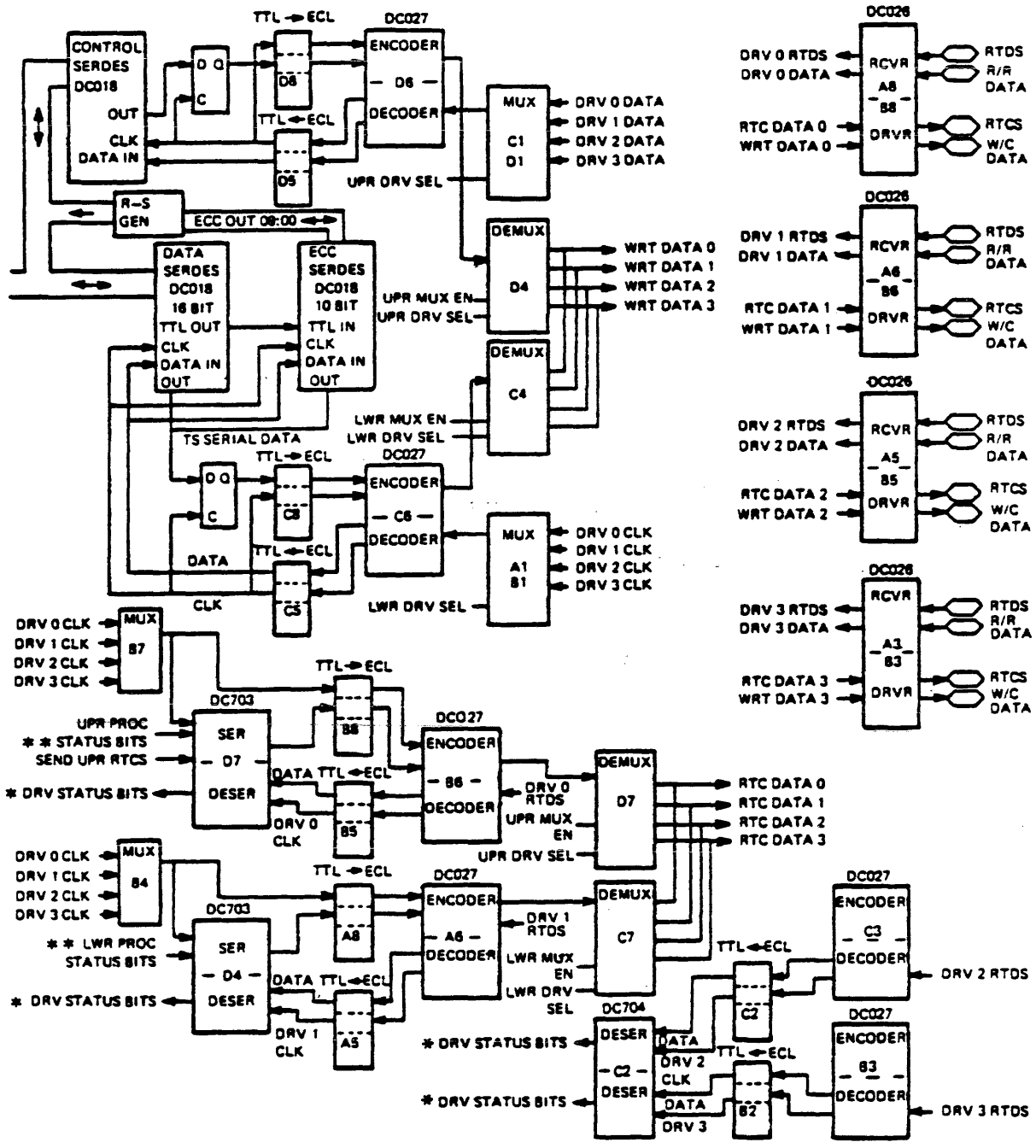
Data Channel Module

LED Indicators on K.sdi

Red (top)	Indicates module has not passed self-test.
Green	Indicates module has passed self-test.



Data Channel Module Block Diagram (Sheet 1 of 2)



* DRV STATUS BITS: AVAIL, R/W RDY, ATTN RDY TO REC, INDEX, SECTOR

* * RTCS BITS: DRV INIT, RCVR RDY, RD GATE, WRT GATE

CX-1201A
Sheet 2 of 2

Data Channel Module Block Diagram (Sheet 2 of 2)

Detailed Block of K.sdi

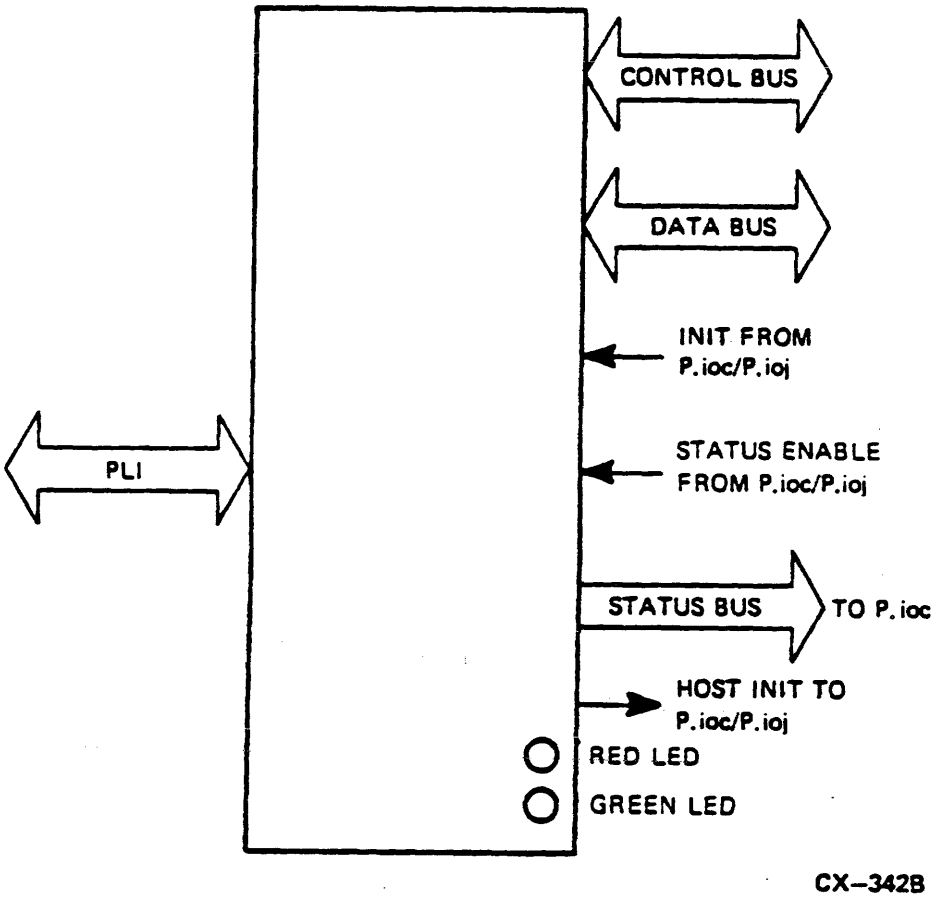
- Uses the 2900-family of bit slice devices.
- Dual microsequencers (2911s) alternately access different areas of common ROM:
  - Each sequencer has a micro-cycle time of 300 ns.
  - Each sequencer consists of three 2911s connected to form a 12-bit wide address bus.
  - The sequencers are defined as Upper and Lower:
    a. The Upper sequencer examines Control Memory for work, acquires/ retires buffers in Data Memory, sends level 2 SDI commands to drive, and has overall control of the Data Channel module.
    b. The Lower sequencer handles the actual data transfer between the drive and Data Memory.
- The Control Store microinstruction is 48-bits wide and each section is decoded as follows:
  - ALU CNTL determines the input, output and operation of ALU.
  - BUS SRC CNTL determines the source of data onto the K.sdi bus.
  - BUS DES CNTL determines the destination of data on the K.sdi bus.
  - IOC CNTL enables the various I/O components on the K.sdi.
- The Upper and Lower sequencers share a common ALU consisting of four 2901s connected to form a 16-bit data path.
- The Upper and Lower sequencers can communicate with each other through a register internal to the ALU and share scratchpad RAM.
- The sequencers, the Control Store, and the Instruction Register form a "pipeline" that allows one processor to fetch the next instruction while the other processor is presently executing an instruction
- The UPR and LWR CNTL REG are used by the upper/lower processors for various control purposes.

Detailed Block of K.sdi (Cont.)

- The DIAG REG is used by microcode to generate signals to test parity check circuits (used only in diagnostic mode).
- The CNTL ERR REG contains control bus error information or error information from an SDI/STI connection (pulse error, drive not present).
- The DATA ERR REG contains data bus-related errors or SDI/STI data transmission errors (pulse/parity on RTDS, ECC).
- Sector/Index PALS polled by upper sequencer to monitor rotational position of selected drive; jumper for K.sti disables these.
- K.sdi can perform seven basic functions through any of its four ports:
  - Send Level 2 SDI commands to drive.
  - Receive Level 2 SDI response from drive.
  - Send Level 1 SDI commands to drive.
  - Send "write" data to drive.
  - Receive "read" data from drive.
  - Send Real-Time Controller State (RTCS) to drive.
  - Receive Real-Time Drive State (RTDS) from drive.
- ECC SERDES takes data being written to the disk and:
  - Converts it to 10-bit parallel input for Reed Solomon Generator (RSGEN).
  - Synchronizes serial data flow out of the DATA SERDES.
- The Clock used to move data to/from disk is actually derived from the Rd/Response line of the target drive.

Port Processor Module K.pli L0107

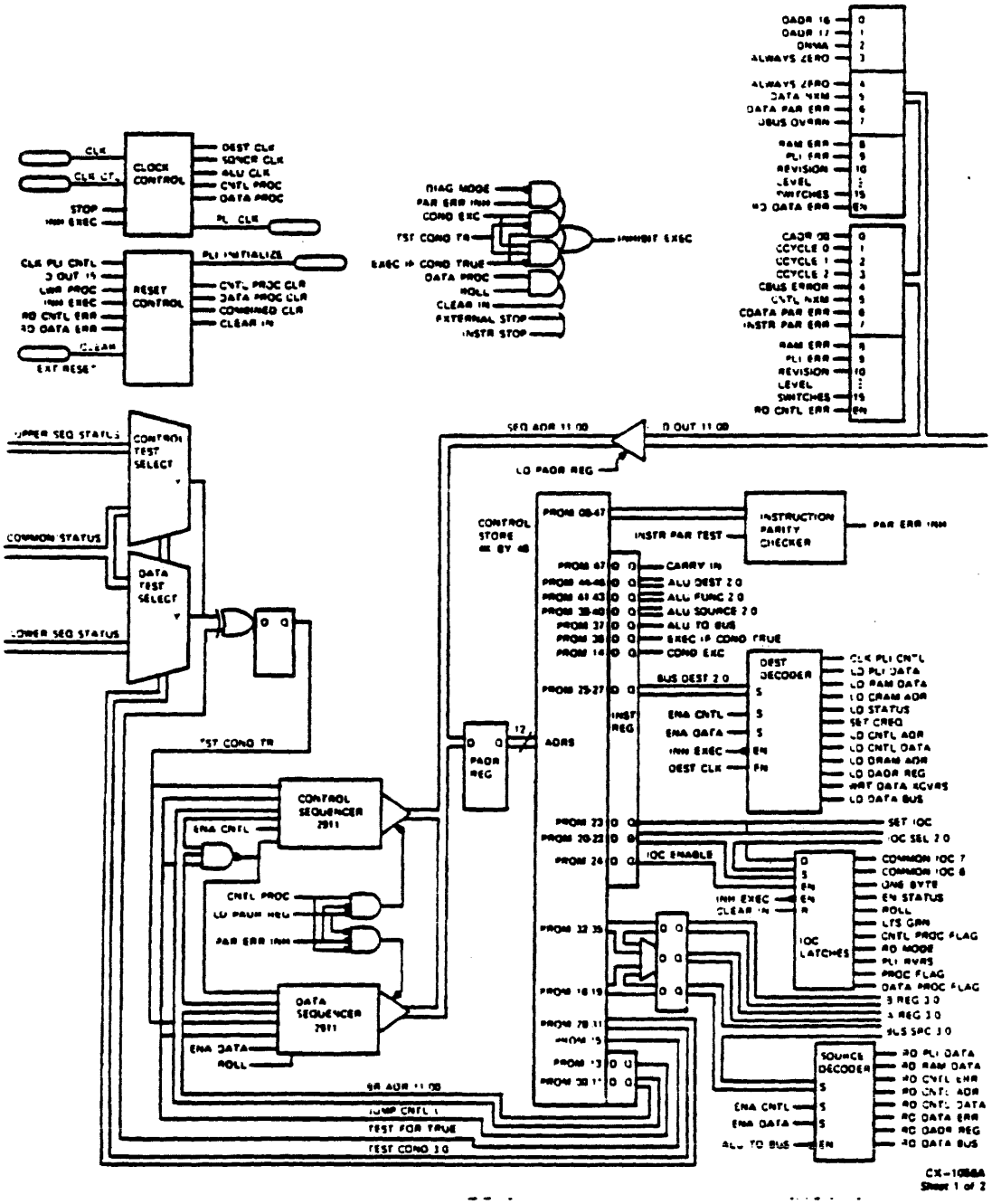
- The K.pli is the "intelligence" of the port processor (K.ci) in the HSC.
- K.pli has the following functions:
  - Passes MSCP messages to/from control memory via the Control bus.
  - Passes data blocks to/from data memory via the Data bus.
  - Generates a 16-bit EDC code for each block of write data.
  - Checks and strips-off EDC code for each block of read data.
  - Performs testing of Port buffer (PILI) and Link (ILI) modules.
  - Initializes the P.ioc/P.ioj (causing an HSC reboot) upon reception of a reset packet from host.
- The K.pli module can be initialized by the P.ioc/P.ioj.
- Upon receipt of INIT from the HSC CPU, the K.pli starts its self-test and the results of the test are put on the Status bus.
- The K.pli interfaces with other module in the HSC on the Control and Data buses; however, it communicates with the other elements of the K.ci via the PLI bus.



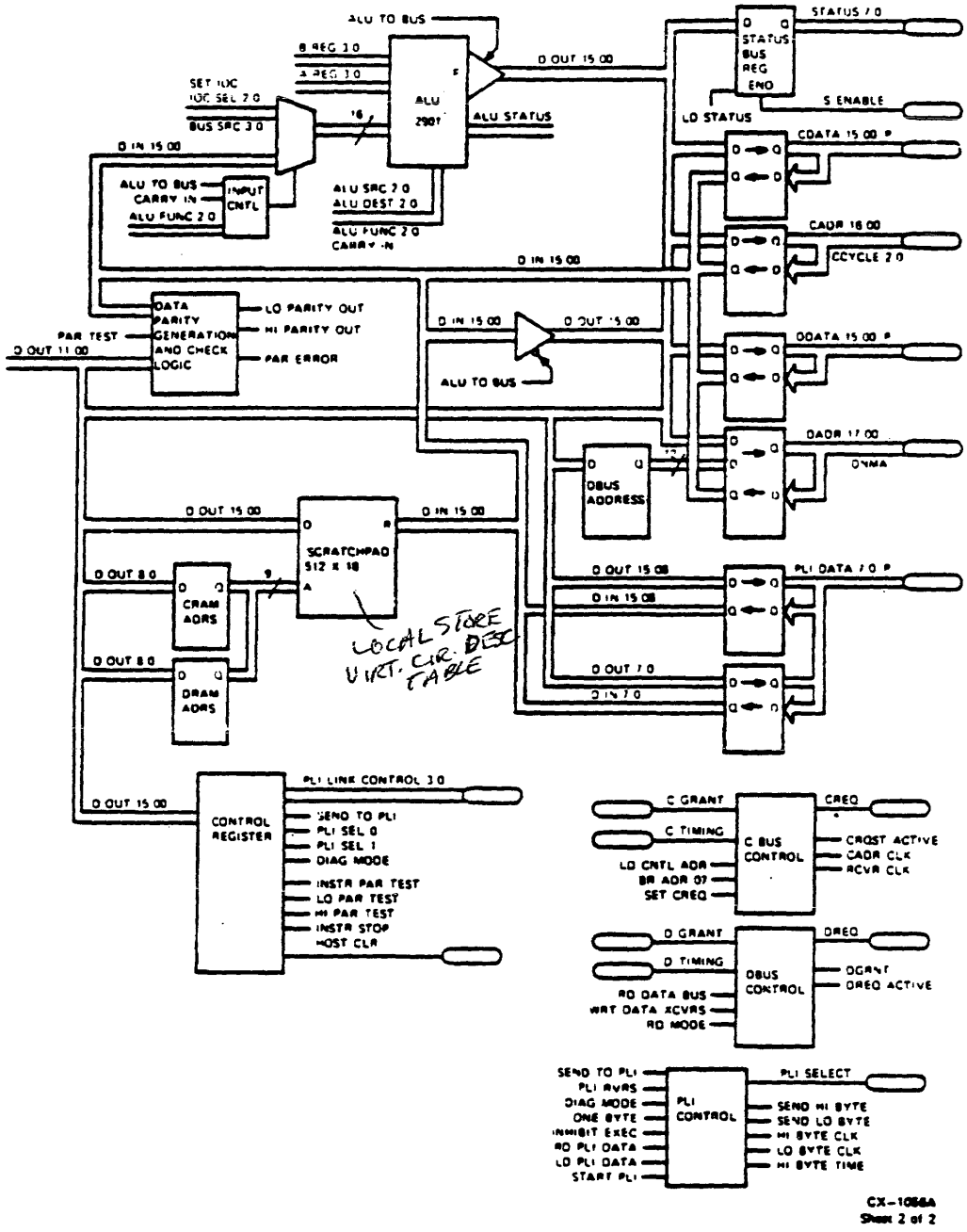
Port Processor Module (K.pli)

LED Indicators on K.pli

Red (top)	Indicates module has not passed self-test.
Yellow	On when P.ioc/P.ioj assert INIT (not present on later etch revs).
Green	Indicates module has passed self-test.



Port Processsor Module Block Diagram (Sheet 1 of 2)



Port Processsor Module Block Diagram (Sheet 2 of 2)

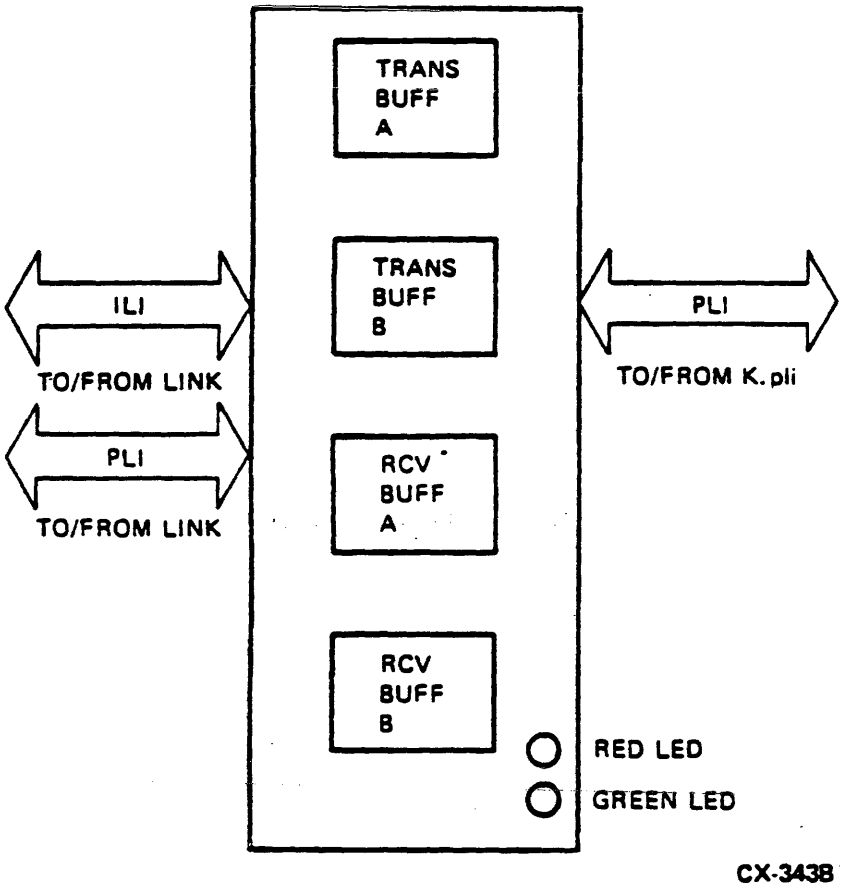
DOES NOT NEED TO HAVE MCODE LOADED (ON PROM)

Detailed Block of K.pli

- K.pli is the controller for the Port Buffer module (PILA) and the Link module (LINK).
- Uses 2900-series bit slice technology to form two sequencers that alternately and independently access a common control store and ALU:
  - Each sequencer consists of three 2911s connected to form a 12-bit address bus to the Control Store.
  - Sequencers are defined as Control and Data:
    a. The Control sequencers examine queues in Control Memory for work and acquire/retire buffers to accommodate traffic on the CI bus.
    b. The Data sequencers handle movement of data from the PILA buffers (CI Bus) to Data/Control Memory.
- Same "pipeline" approach as in the Data Channel module.
- Microinstruction is 48-bits wide and controls functions on the Port Processor, the Port Buffer, and the Link module.
- Scratchpad RAM (512 words) is shared by both processors and contains tables for CI virtual circuits and connection information (equivalent to VCDT of CI Interface).
- The Control Register contains logic for PLI interface control, Control Store parity test, data parity test, and P.ioj/P.ioc reset.
- The PLI interface logic performs byte-word conversion for data movement from PILA to HSC memory, and word-byte conversion for data movement from HSC memory to PILA.

Port Buffer Module PILA L0109

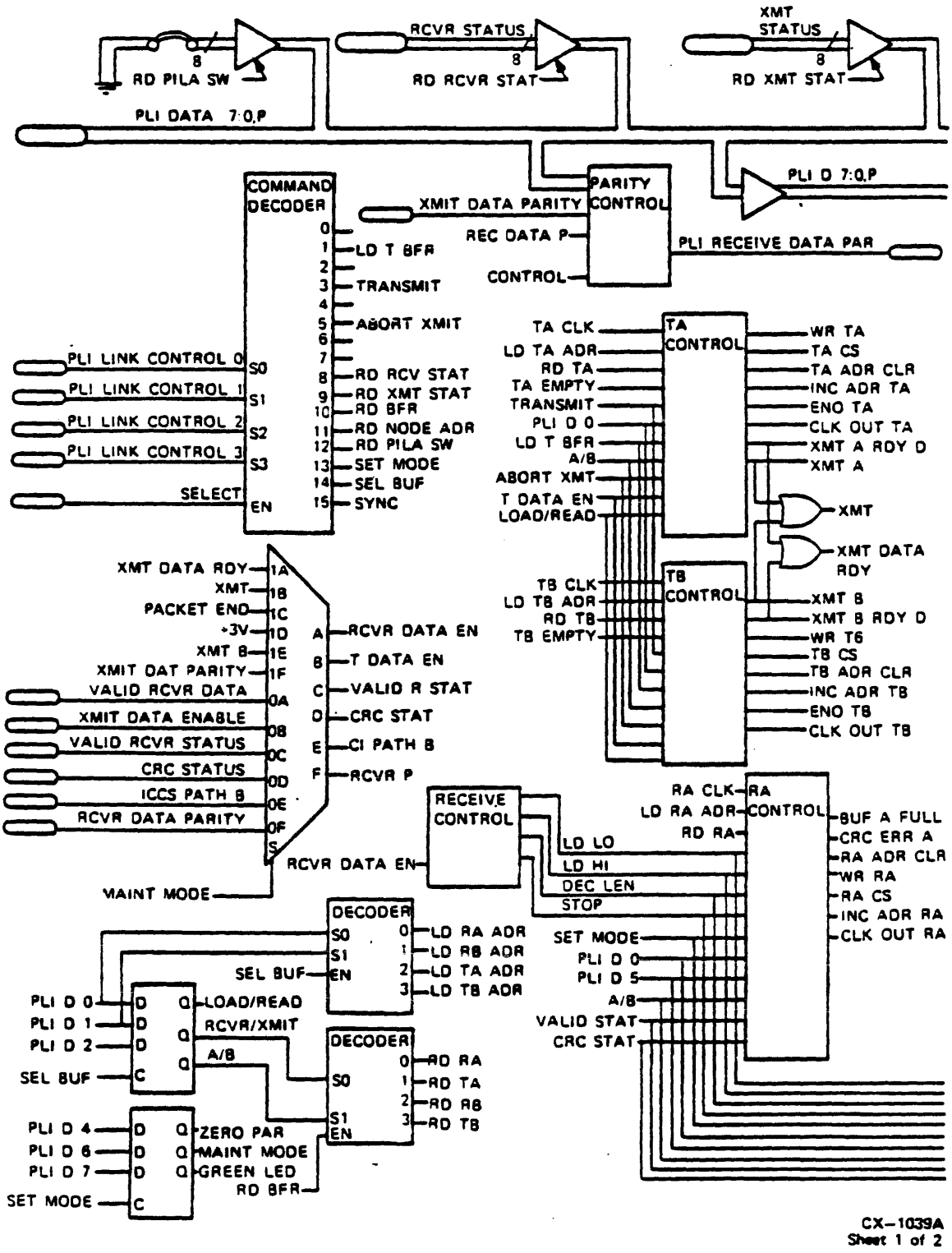
- The Port Buffer module (PILA) provides buffering for data and message packets to and from the CI Bus:
  - Two 1Kb receive buffers.
  - Two 1Kb transmit buffers.
- The PILA module communicates with the Port Link module over the ILI bus.
- Communicates with the Port Link module and the K.pli module over the PLI bus.
- Not connected to either the Control or Data bus in the HSC.
- Contains no self-test but is tested instead by the K.pli, which can loop data through the buffers in maintenance mode.



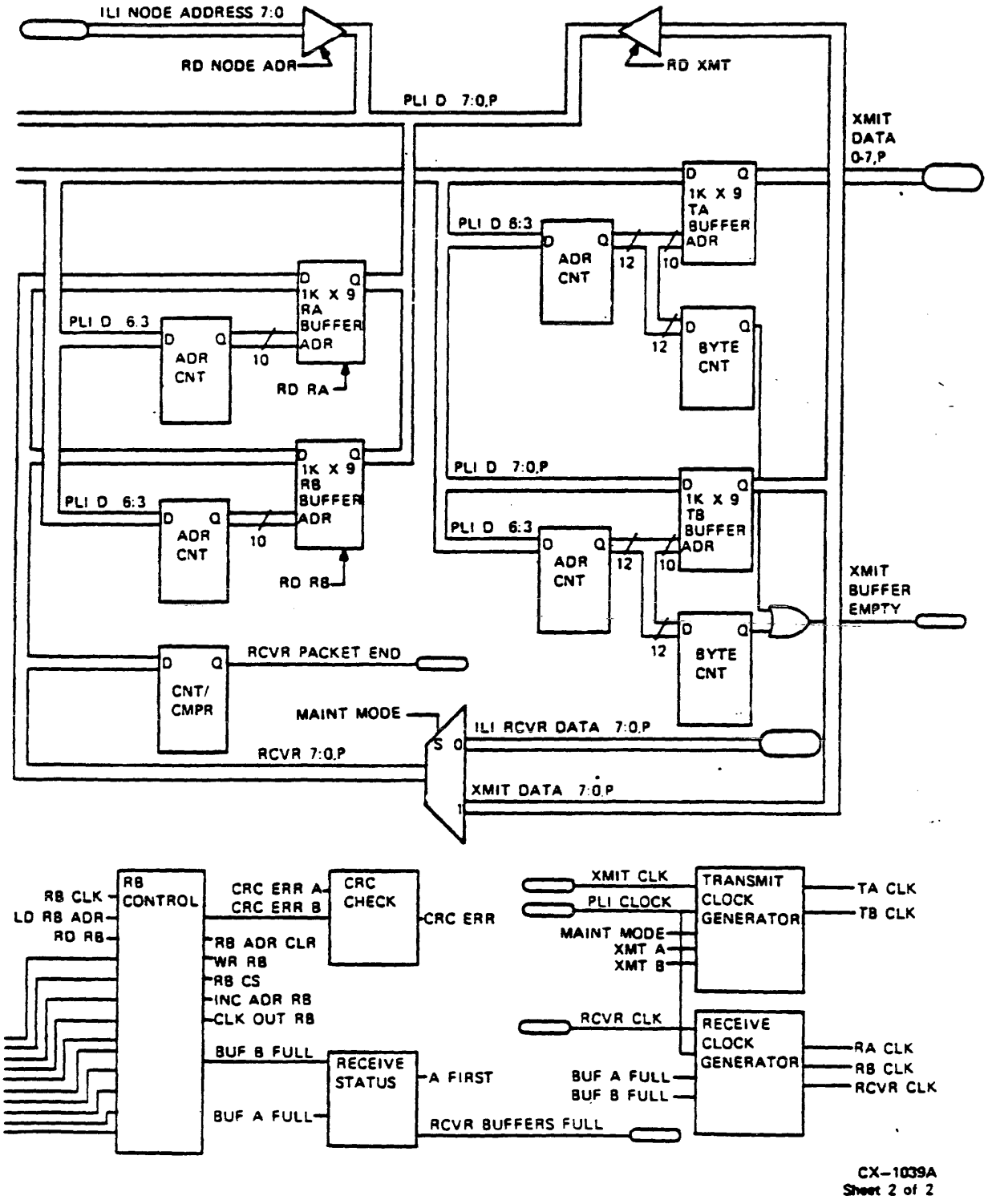
Port Buffer Module (PILA)

LED Indicators on PILA

Yellow (top)	Test point, not used, is always on (not on later etch revs).
Yellow	Indicates receipt of INIT from K.pli (not on later etch revs).
Red	Indicates module has not passed test performed by the K.pli.
Green	Indicates module has passed K.pli test and is operable.



Port Buffer Module Block Diagram (PILA) (Sheet 1 of 2)



Port Buffer Module Block Diagram (PILA) (Sheet 2 of 2)

Detailed Block of PILA

- PILA operation centers around the loading and unloading of the dual, independent 1Kb transmit buffers (TA and TB) and the dual independent 1Kb receive buffers (RA and RB).
- A typical receive operation (LINK to PILA) happens in this sequence:
  - LINK recognizes valid incoming data from the CI Bus upon receiving the CHAR SYNC byte of the packet.
  - LINK assembles the first byte (packet type/length, high) and places it on ILI RCVR DATA lines.
  - Output of mux is counter/comparator and input buffer RA.
  - LINK then asserts control signals that clear the address counter for RA, strobe the packet length byte into the counter/comparator and first location of RA, and increments the address counter to location one.
  - LINK assembles the next byte (packet length low) and strobes it into the counter/comparator and RA buffer, and increments the address counter.
  - As each byte received by LINK is passed to the RA buffer, the counter/comparator is decremented.
  - When the counter decrements to two, RCVR PACKET END is asserted back to LINK to begin the CRC check.
  - LINK asserts RCVR BUFFER A FULL to K.pli.
- A typical transmit operation (PILA to LINK) follows this sequence:
  - K.pli loads the TA buffer and byte count register (not shown) via the PLI bus.
  - K.pli commands LINK PALs to execute a transmit operation.
  - Address counter is cleared, byte counter is loaded, and outputs of the TA buffer to LINK are enabled.

Detailed Block of PILA (Cont.)

  - LINK strobes bytes onto XMIT DATA lines while incrementing address counter and decrementing byte counter.

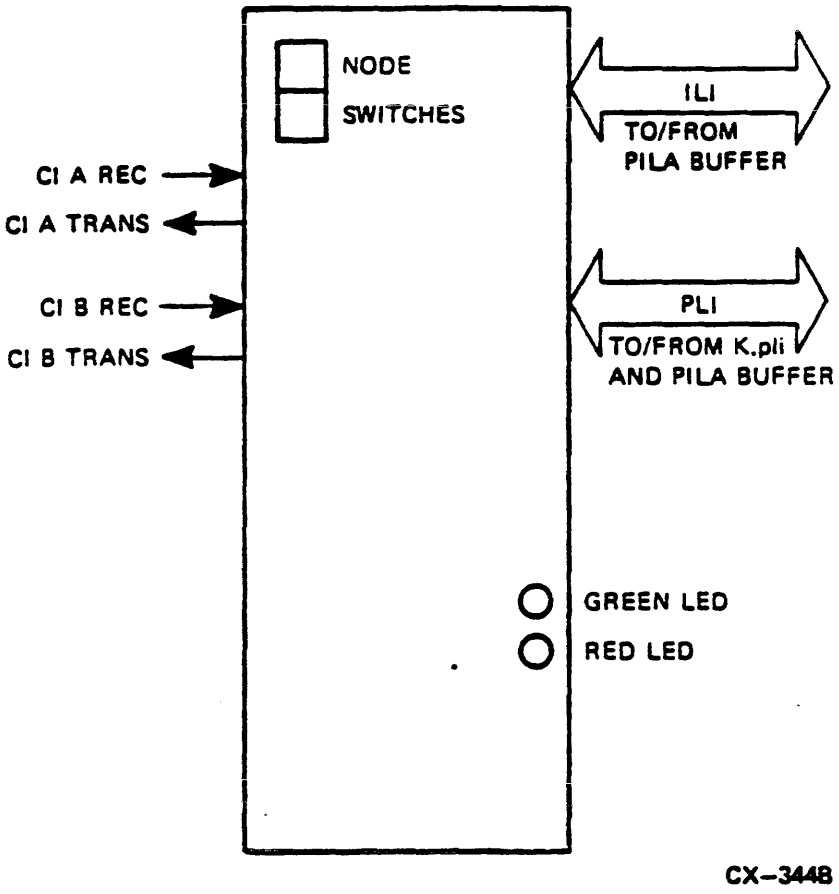
  - When the byte counter reaches zero, XMIT BUFFER EMPTY causes LINK to add CRC bytes.

- The COMMAND DECODER is the source of most commands from K.pli to LINK or PILA (note RD NODE ADDR output of decoder, this is how CRONIC can determine its node number).

- In maintenance mode, the transmit buffers can be loaded and then their output looped back to the receive buffers for checking by the K.pli.

Port Link Module ILI L0100

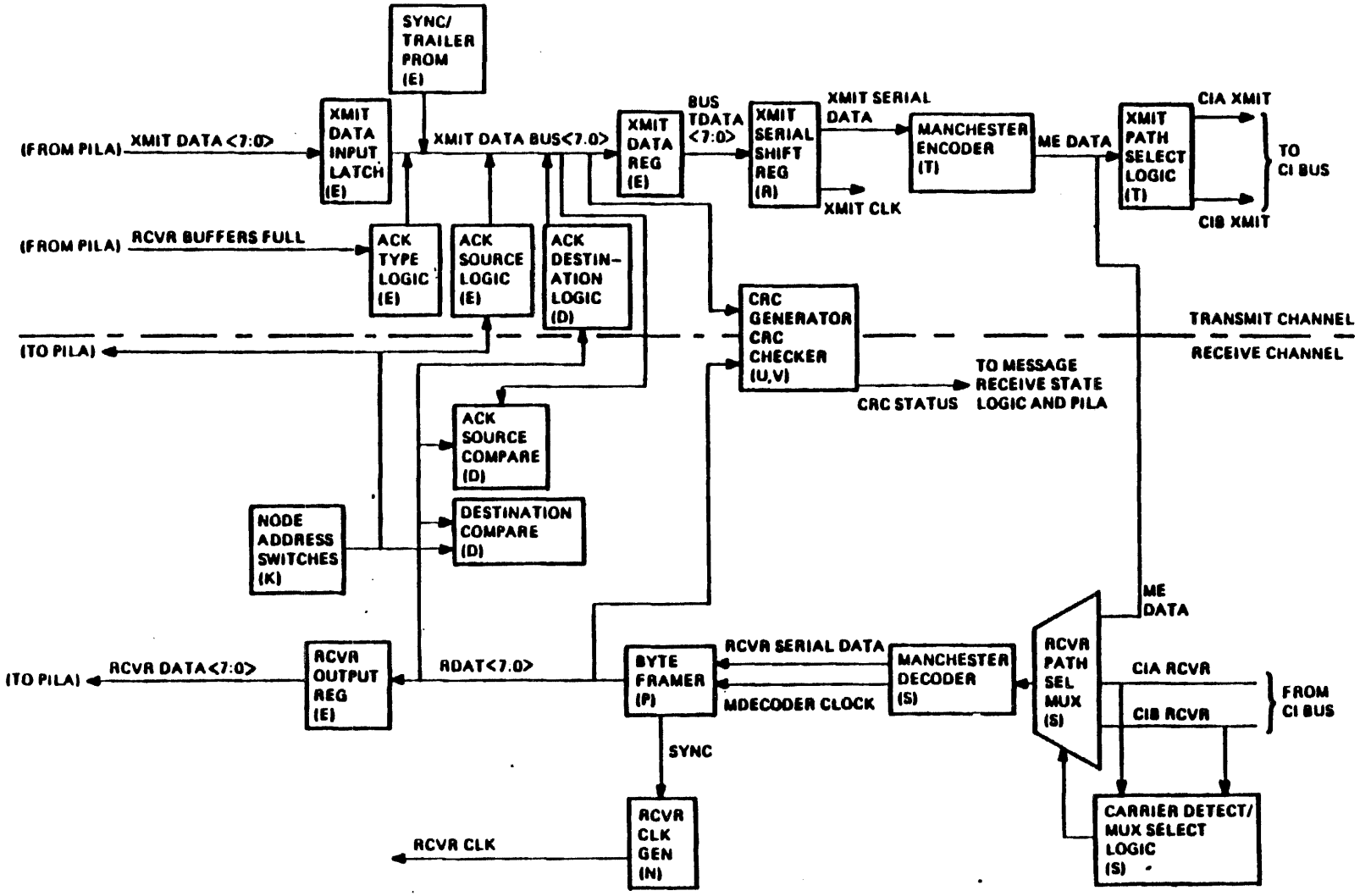
- The Port Link module is the HSC's direct interface to the CI Bus.
- It is the same module as in the CI750, CI780, or CIBCI port adapters.
- Contains the following:
  - Drivers/receivers for dual CI path operation.
  - CI Bus arbitration.
  - Node address switches.
  - Generates/checks CRC of CI Packet.
  - Interfaces to ILI and PILA buses.
- The newer LINK board (L0118) allows an upgrade of the cluster to greater than 16 nodes.



Port Link Module (LINK)

LED Indicators on ILI

Green (top)	Indicates external CI Bus activity.
Red	Internal loopback enabled, usually on when K.pli initializes LINK board.



NOTE: LETTER DESIGNATIONS IN PARENTHESES REFER TO ENGINEERING DRAWINGS CONTAINING CORRESPONDING LOGIC.

CX-951A

LINK Simplified Block Diagram

Port Link Module Block

- As a CI Port receiver, the LINK board:
  - Detects packet arrival.
  - Performs Manchester decoding.
  - Performs serial-to-parallel conversion.
  - Checks node address.
  - Directs packet contents to a receive buffer on the PILA module.
  - Checks CRC.
  - Generates response (ACK, NACK, NORESP).
- As a CI Port driver, the LINK board:
  - Arbitrates for the CI Bus.
  - Performs parallel-to-serial conversion.
  - Generates CRC code.
  - Listens for response (ACK or NACK).

HSC SYSTEM INITIALIZATION

HSC System Initialization

Lesson Introduction

This module discusses the initialization and boot sequence of the HSCs. It is broken into two major sections. We discuss first the HSC50 and then the HSC70.

The boot/initialization sequence is in three parts, 1) the ROM-based portion, 2) the offline media, and 3) the system media. The offline and system sections are executed by first loading them from the media into program memory. The ROM portion is run directly from ROMs on the individual requestors.

The ability to follow through a boot sequence step-by-step will greatly simplify the troubleshooting process.

Lesson Objectives

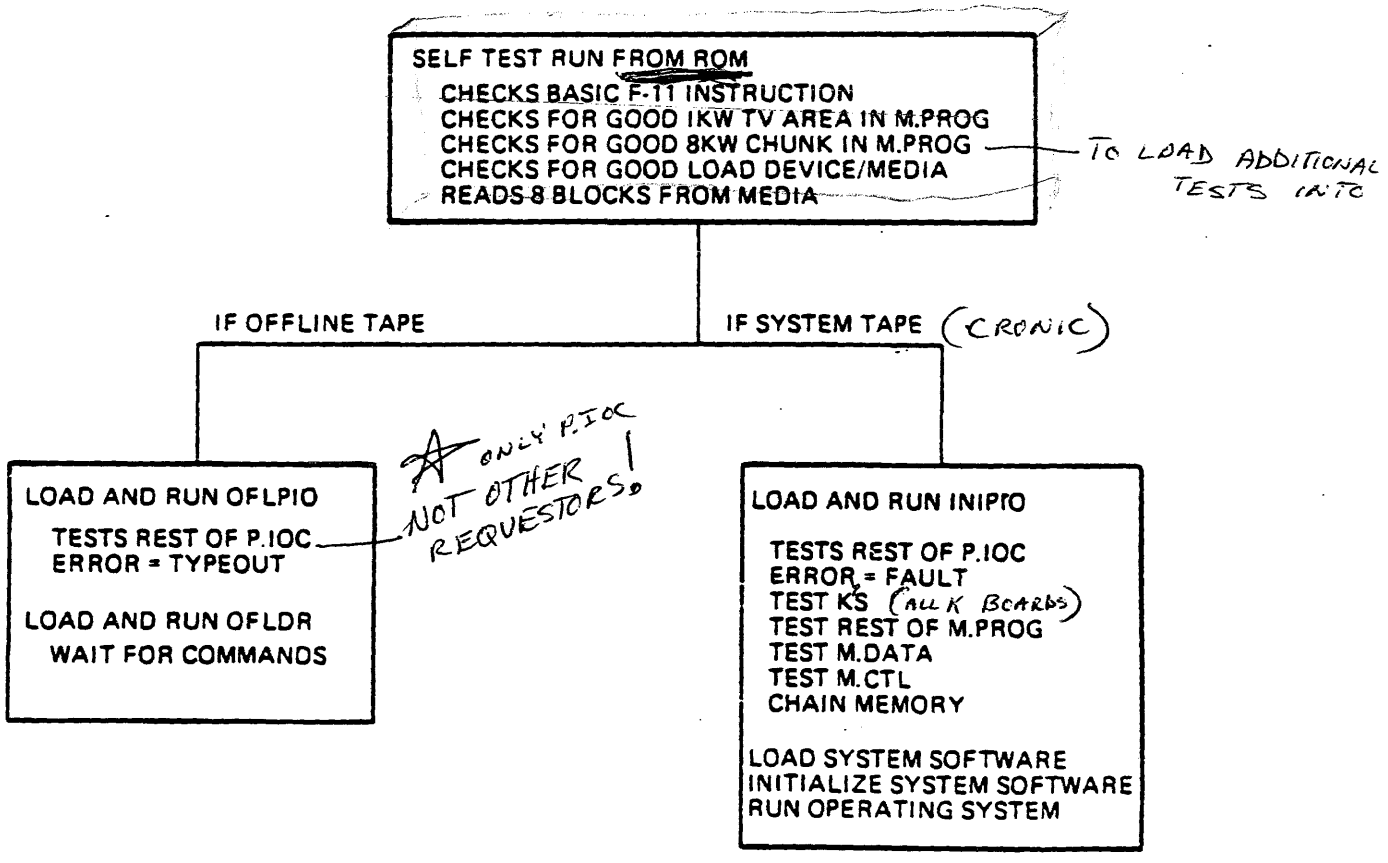
1. Describe what tests are run from ROM during initialization.
2. Define what is tested during a system boot.
3. Define what is tested during an offline boot.
4. Replace failing FRUs from QCP and module LED fault indications.

Lesson Outline

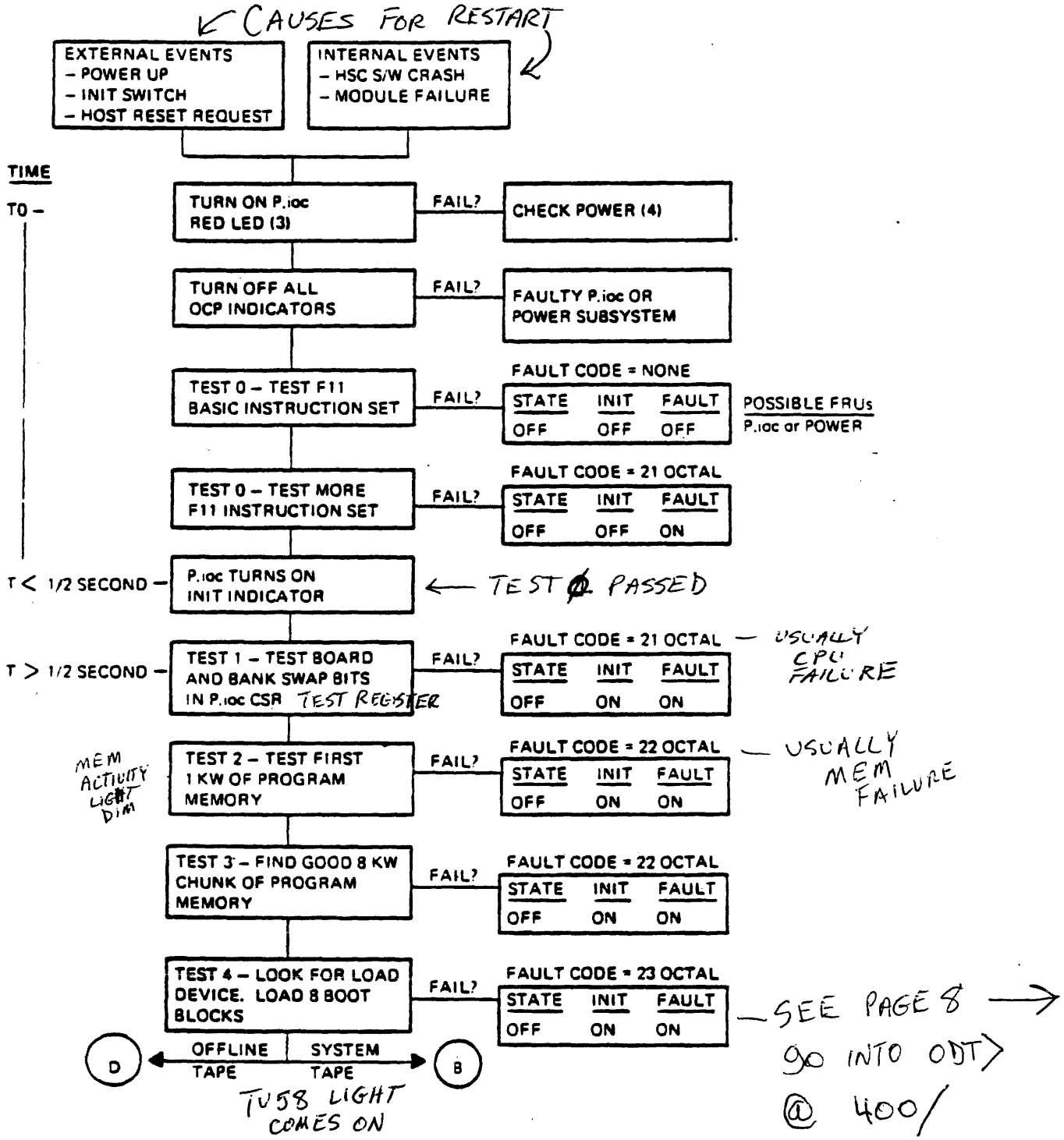
I. Introduction
II. HSC50 ROM-Based
III. HSC50 System Tape
IV. HSC50 Offline Tape
V. HSC70 ROM-Based
VI. HSC70 System Diskette
VII. HSC50 Offline Tape

Booting an HSC

- HSC boots itself due to the following external events:
  - Power is applied to the HSC.
  - INIT switch is pressed.
  - Host reset request is received on the CI Port.
  - Command from terminal.
- HSC also boots itself due to the following internal events:
  - HSC software crash.
  - Module failure.
- Whatever the cause of the re-boot, the events that compose an HSC initialization are outlined below:



CX-352A



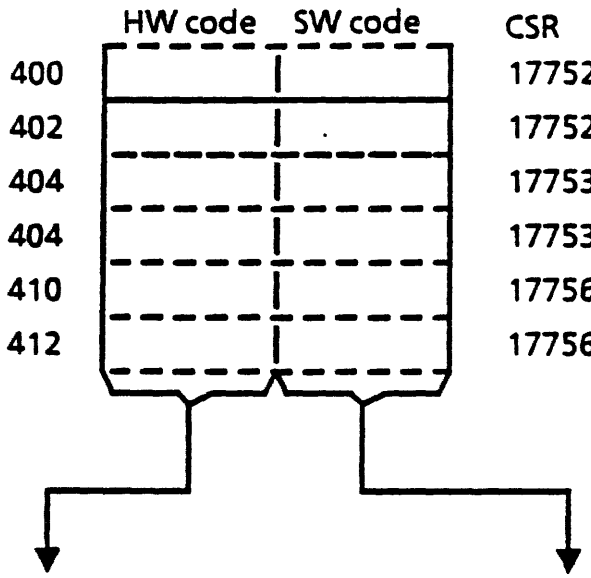
HSC50 Boot Flow (ROM Portion)

Detail of HSC50 Boot (ROM Portion)

- All of the tests shown on the HSC50 Boot Flow diagram are run from ROM on P.ioc.
- The starting address is 173000; test being run is found at address 172340.
- During these tests, all Ks are initialized (red lights on).
- Test 0
  - Tests address modes, single and double operand instructions, BRANCH, JMP, RTS, and JSR.
  - Does R1 through R6 and adder tests.
- Test 1
  - Sets and clears bank swap and board swap bits in P.ioc CSR.
- Test 2
  - Does moving inversions test on beginning 1KW of Program Memory (this is area reserved for trap vectors).
  - Attempts the bank swap if necessary.
  - Does not implement board swap feature.
- Test 3
  - Attempts to find contiguous 8KW chunk in Program Memory.
  - Moves the base of the 8KW chunk, if necessary, until the upper limit of 157777 (top of bank) reached.
  - Uses the 8KW partition to load either the offline P.ioc test or the Init P.ioc tests.

Detail of HSC50 Boot (Cont.)

- Test 4
  - Checks for presence of TU58 controller.
  - Checks for presence of mounted media with RT-11 boot block.
  - Reads first 8 blocks into Program Memory starting at address 04000.
  - If error 23 occurs, enter uODT (press BREAK key) and examine M.prog locations 400 to 412 as follows:



	HW code	SW code	CSR	UNIT	SW desig
400			177520	0	DD0:
402			177520	1	DD1:
404			177530	0	DD2:
404			177530	1	DD3:
410			177560	0	
412			177560	1	

1774xx - Self-Test Failed
1770xx - EOT Encountered
1750xx - Hard Read Error
1740xx - Bad Unit #
1734xx - No Cassette Mounted
1724xx - Write Protected
1674xx - Data Check Error
1600xx - Seek Error
1574xx - Motor Stopped
1500xx - Bad Op Code
1444xx - Bad Block #

0 - no error
1 - NXM Trap Accessing CSR
2 - Sync Sequence Failed
3 - Self-Test Failed
4 - No Bootable Image
5 - Checksum Error In Packet
6 - Timeout On Transmit Ready
7 - Timeout On Receive Done
10 - Overrun or Framing Error
11 - Unknown Packet Type

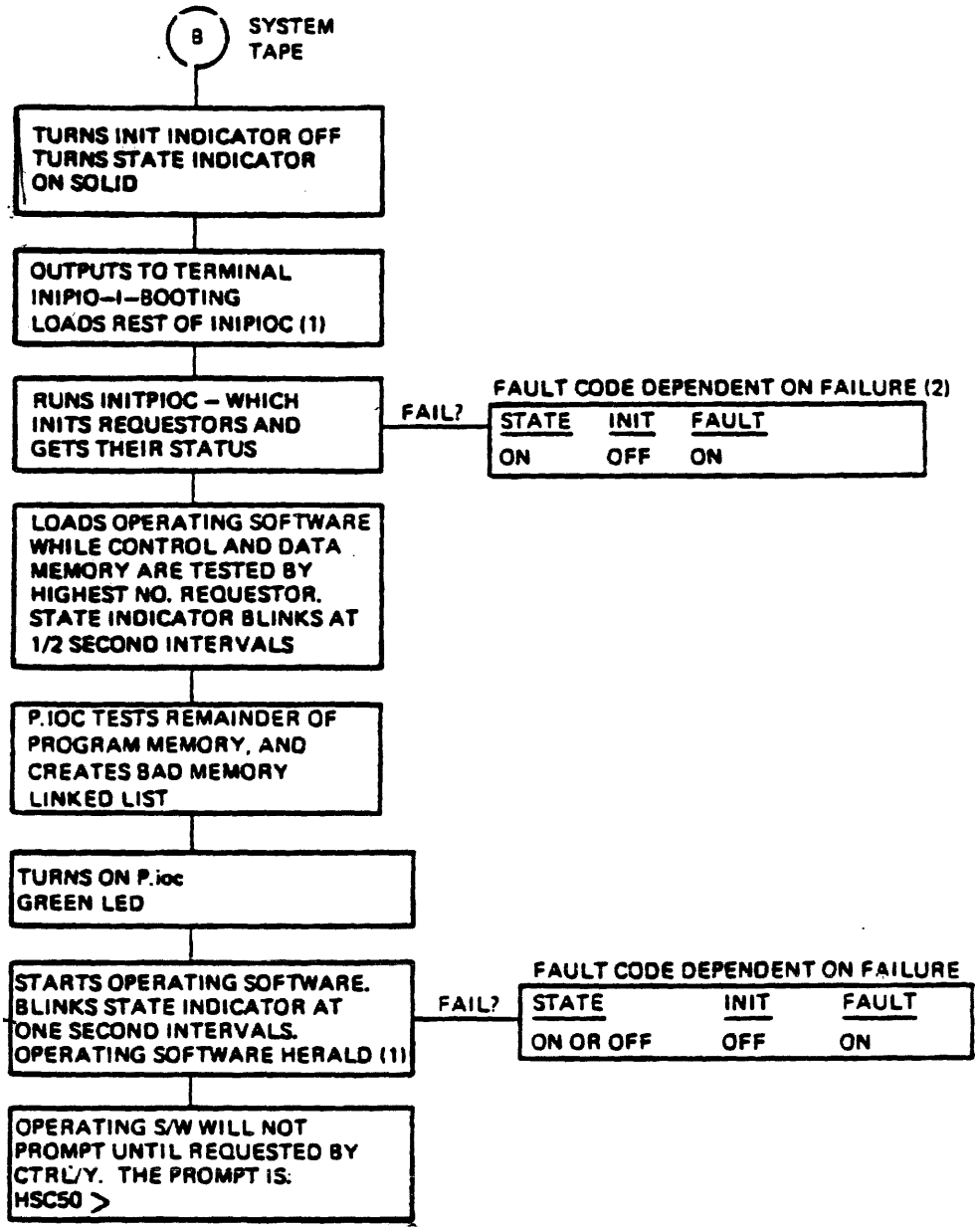
Detail of HSC50 Boot (Cont.)

- Sequence of LED action during the ROM portion of HSC50 boot is as follows:
  - All OCP lights turn off.
  - All modules turn on red LED.
  - P.ioc Run indicator (yellow) dim (instruction tests).
  - INIT light on OCP comes on.
  - TU58 yellow "Tape In Motion" LED on.
  - INIT light off and State indicator on solid.
- OCP fault code displays:

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

† INCORRECT VERSION OF MICROCODE.
* THESE ARE THE SO-CALLED SOFT OR NON-FATAL ERRORS.
**POSSIBLE MEMORY MODULE/CONTROLLER ON HSC70

CX-9058



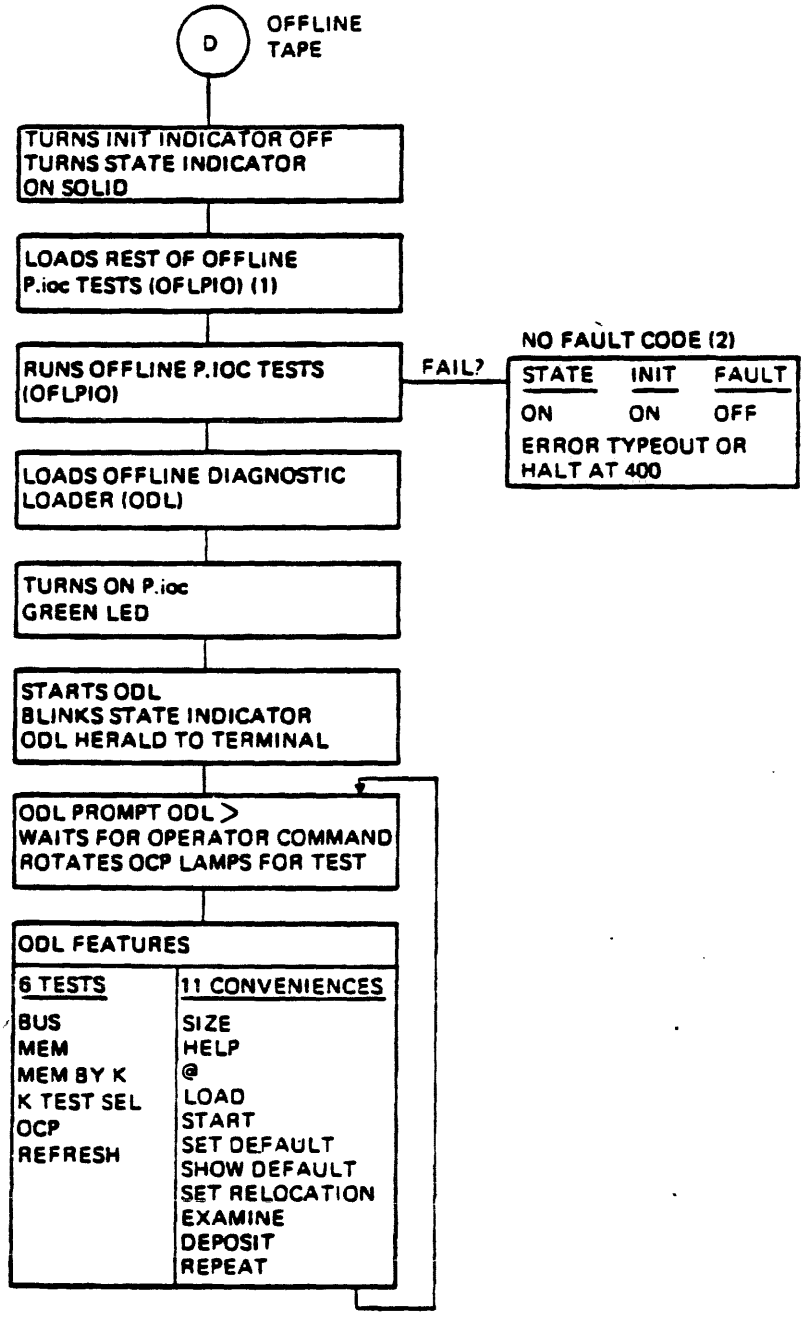
CX052A
Sheet 2 of 4
Sheet 3 of 4

HSC50 Boot Flow (System Tape)

Detail of HSC50 Boot (System Tape Portion)

- Position of Fault switch on OCP is saved.
- Program outputs "INIPIO-I- Booting".
- INIPIO.INI loaded from TU58 and run.
- INIPIO.INI tests the internals of the P.ioc (as OFLPIO did) and also does the following:
  - Rewinds TU58.
  - Releases the INIT line to the K.ci so that it will test itself and then be able to establish virtual circuits.
  - Releases the INIT lines for all other Ks so they can run their microdiagnostics.
  - Selects a K to test Control Memory.
  - Tests remaining Program Memory.
  - Builds suspect and disabled memory lists.
  - Examine saved position of Fault switch and proper copy of SCT used.
- Any errors during the above operations will display a fault code on the OCP.

IF YOU CHANGE BAD MEM OUT YOU MUST CLEAR OUT THESE TABLES ALSO USE SHO MEM TO SEE



CX-052B
Sheet 4 of 4

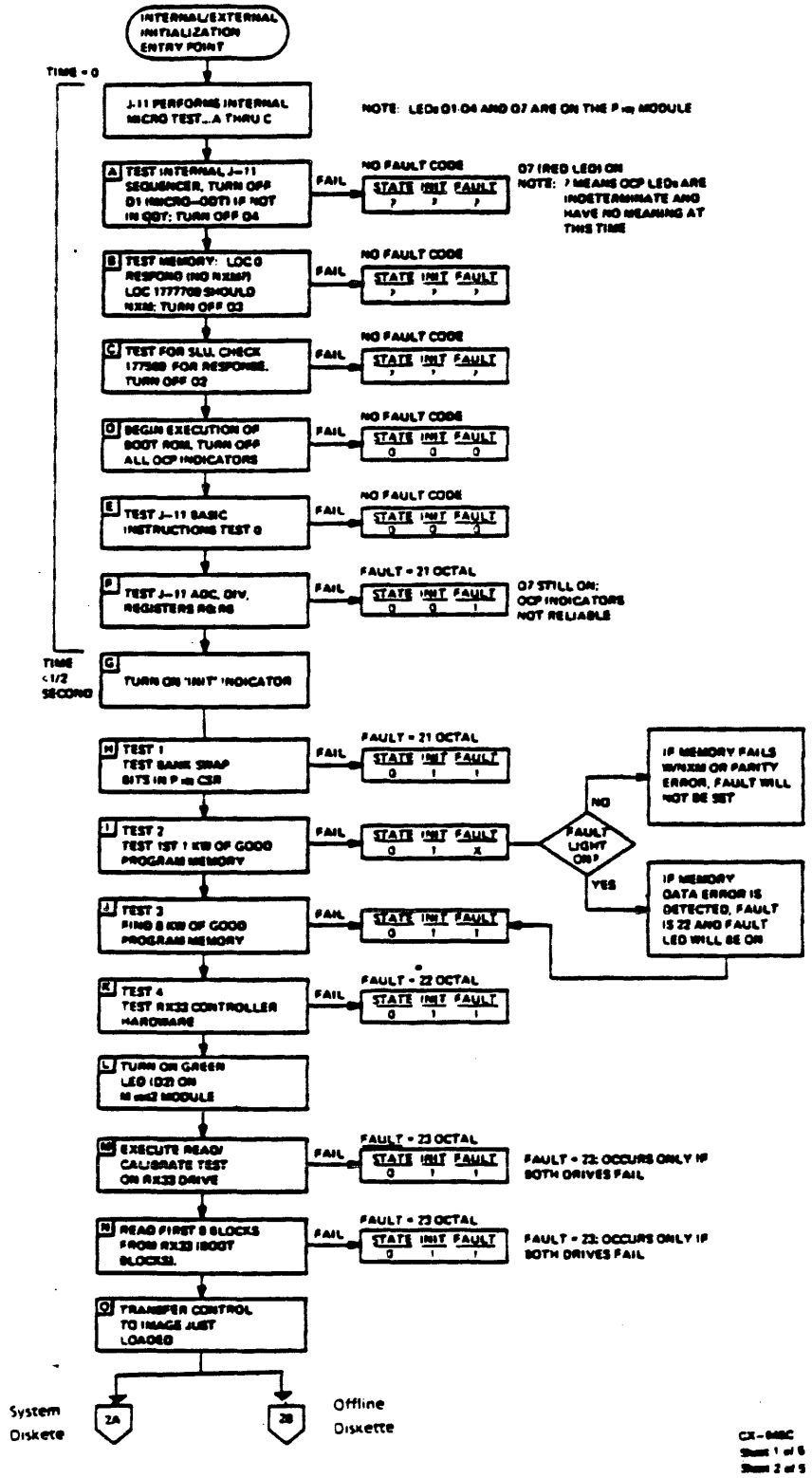
HSC50 Boot Flow (Offline Tape)

Detail of HSC50 Boot (Offline Tape Portion)

- Eight boot blocks previously loaded (in ROM portion of boot) now used to load Offline P.ioc (OFLPIO) tests.
- OFLPIO tests the following on the P.ioc module:
  - Remaining PDP-11 instructions DIV, MUL, XIR, SXT, and SBC.
  - Memory Management Unit (MMU).
  - I/O page registers.
  - Internal interrupts and traps.
  - P.ioc parity generation and checking.
  - Control windows.
  - C Bus interrupts, C Bus lock cycle, C Bus address and data lines (no Control Memory access).
  - D Bus address and data lines (no Data Memory access).
- OFLLDR.SAV (Offline Diagnostic Loader) is loaded from TU58 and run.
- Note that none of the Ks are allowed to complete initialization and no memory is tested (except 8KW of Program Memory).
- Format of the error message will be as follows:

```
OPIO>00:00 T#xxx E#xxx u-000
<text string describing error>
   MA - <22-bit address of failing location>
   EXP - <data expected>
   ACT - <data actually found>
```

- At the ODL> prompt, the offline loader rotates a light pattern through the OCP while waiting for operator input.



HCS70 Boot Flow (ROM Portion)

Detail of HSC70 Boot (ROM Portion)

- J-11 microtests A through C accomplish the following:
  - Exits from uODT.
  - Checks for existence of Program Memory.
  - Checks for console terminal.
- Test 0
  - Tests address modes, single and double operand instructions, BRANCH, JMP, RTS, and JSR.
  - Does R1 through R6 and adder tests.
- Test 1
  - Sets and clears bank swap and board swap bits in P.ioc CSR.
- Test 2
  - Does moving inversions test on beginning 1KW of Program Memory (this is area reserved for trap vectors).
  - Attempts bank swap if necessary.
  - Does not implement board swap feature.
- Test 3
  - Attempts to find contiguous 8KW chunk in Program Memory.
  - Moves base of 8KW chunk, if necessary, until upper limit of 160000 (top of bank) reached.
  - Uses this 8KW partition to load either the offline P.ioc test or the Init P.ioc tests.

Detail of HSC70 Boot (Cont.)

- Test 4 — GIVES ERROR CODE 22 IF FAILS
  - Tests the RX33 controller logic on Memory module.
  - Tests four floppy disk controller registers for stuck bits.
  - Checks DMA operation.
- Test 5
  - Checks for presence of RX-33 via drive ready bit in RX33 interface register on Memory module.
  - Sends recalibrate command to floppy.
  - Verifies recalibration took place.
  - Loads the first eight blocks from the diskette into 8KW parition found in Test 3.
  - If error 23 occurs, enter uODT (press BREAK key) and examine M.prog locations 400 to 410 as follows:

    400 Contains controller error code
    402 RX33 address being accessed, if applicable
    404 Expected result
    406 Actual result
    410 Drive error code, byte-encoded where:

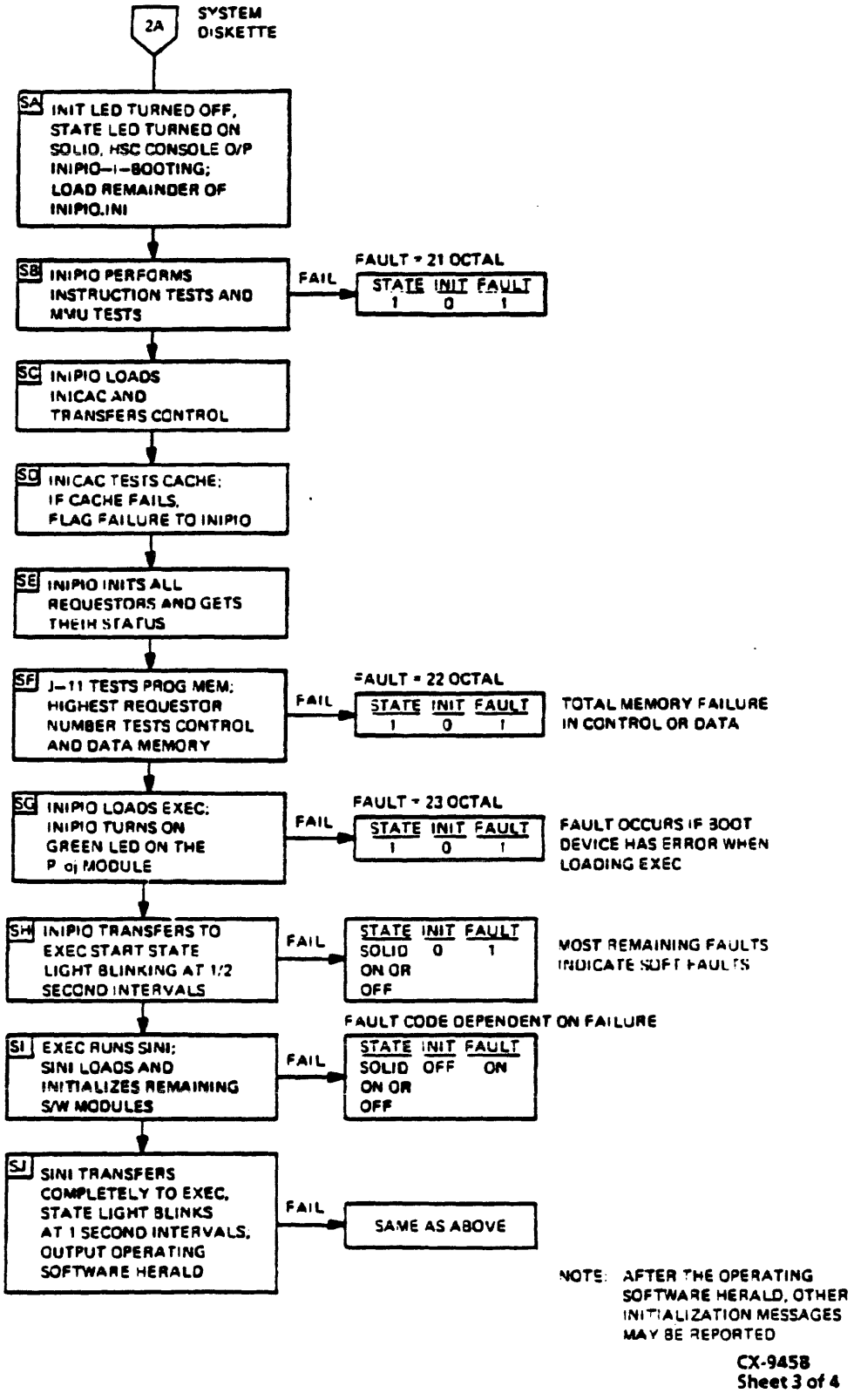
    drive1/drive0 = high-byte/low-byte

Detail of HSC70 Boot (Cont.)

- Error code will be in either 400 or 410 (not both):

  Code

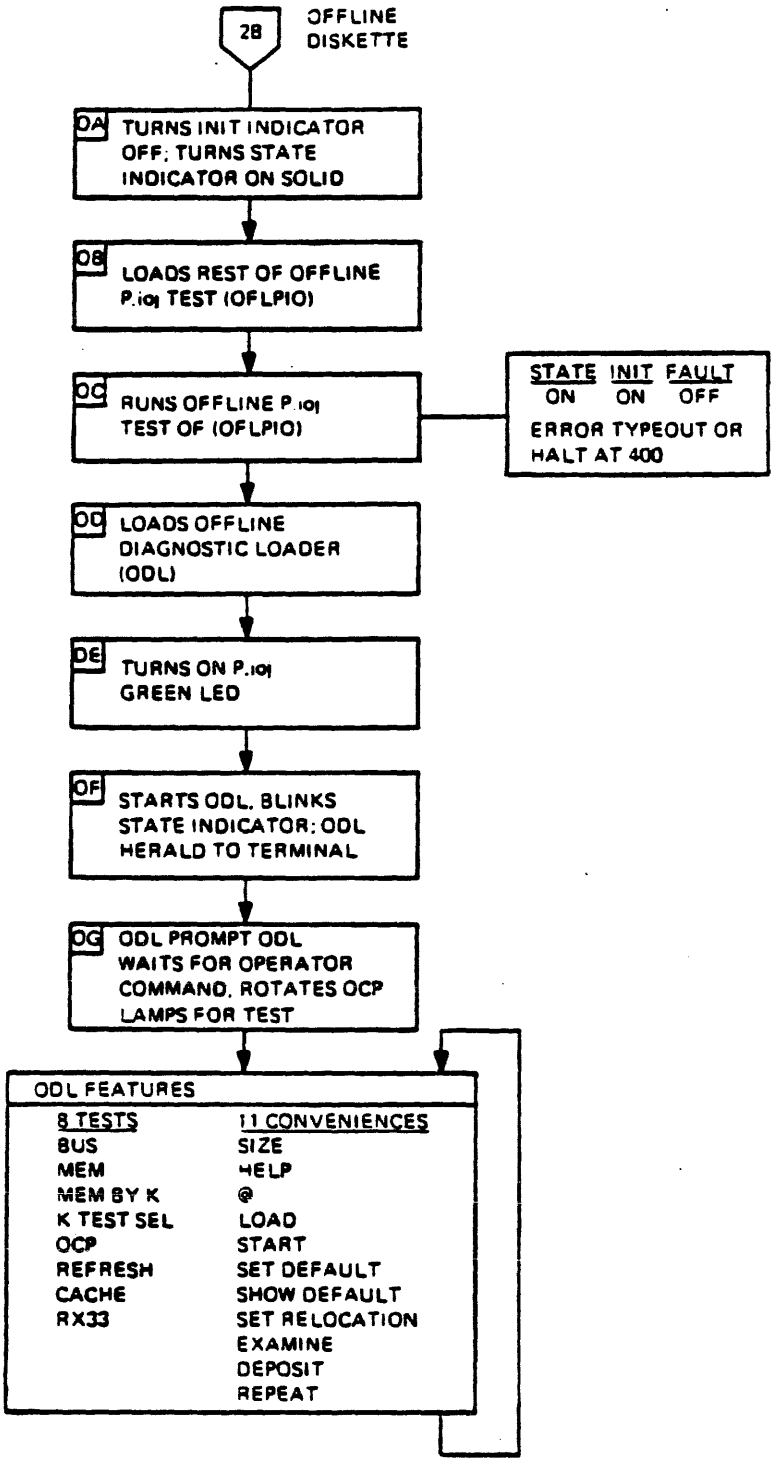
  1 NXM occurred while accessing RX33 registers
  2 Bit stuck in registers
  3 Force mode interrupt did not occur
  4 DMA test mode hardware error occured
  5 DMA address counters were wrong after transfer
  6 Incorrect data found after DMA test operation
  7 Data parity was bad after DMA test operation
  10 Drive not ready (no diskette or door open)
  11 Hard error occurred on recal/verify
  12 Track 0 bit was not set after recal
  13 SEEK command timeout occurred
  14 Seek error occurred
  15 READ SECTOR command timeout
  16 Hard error occurred on read
  17 Nonbootable image is first word



HSC70 Boot Flow (System Diskette)

Detail of HSC70 Boot (System Diskette Portion)

- Position of Fault switch on OCP is saved.
- Program outputs "INIPIO-I- Booting".
- INIPIO.INI loaded from RX33 and run.
- INIPIO.INI tests the internals of the P.ioj (as OFLPIO did) and also does the following:
  - Loads INICAC from the diskette that is run to test cache.
  - Releases the INIT line to the K.ci so that it will test itself and then be able to establish virtual circuits.
  - Releases the INIT lines for all other Ks so they can run their microdiagnostics.
  - Selects a K to test Control Memory.
  - Tests remaining Program Memory.
  - Builds suspect and disabled Memory lists.
  - Examines the saved position of Fault switch and proper copy of SCT used.
- Any errors during the above operations will display a fault code on the OCP.



CX-945C
Sheet 5 of 5

HSC70 Boot Flow (Offline Diskette)

Detail of HSC70 Boot (Offline Diskette Portion)

- Same as HSC50.
- Eight boot blocks previously loaded (in ROM portion of boot) now used to load Offline P.ioj (OFLPIO) tests.
- OFLPIO tests the following on the P.ioj module:
  - Remaining PDP-11 instructions DIV, MUL, XIR, SXT, and SBC.
  - Memory Management Unit (MMU).
  - I/O page registers.
  - Internal interrupts and traps.
  - P.ioj parity generation and checking.
  - Control windows.
  - C Bus interrupts, C Bus lock cycle, C Bus address and data lines (no Control Memory access).
  - D Bus address and data lines (no Data Memory access).
- OFLLDR.SAV (Offline Diagnostic Loader) is loaded from RX33 and run.
- Note that none of the Ks are allowed to complete initialization and no memory is tested (except 8KW of Program Memory).
- Format of the error message will be as follows:

```
OPIO>00:00 T#xxx E#xxx u-000
<text string describing error>
MA - <22-bit address of failing location>
EXP - <data expected>
ACT - <data actually found>
```

- At ODL> prompt, the offline loader rotates a light pattern through the OCP while waiting for operator input.

HSC OFFLINE DIAGNOSTICS

HSC Offline Diagnostics

Lesson Introduction

This module discusses the offline diagnostics. These diagnostics are run while the HSC is offline to the rest of the cluster. The offline diagnostic media contains a skeletal operating system called the Offline Loader. The HSC cannot be running its normal operating system while the Offline Loader is running, and therefore will not respond to the CI, SDI, or the STI buses.

Lesson Objectives

1. Describe the uses of the Offline Loader software.
2. List the commands which exist under the ODL> prompt and explain the use of each.
3. List the six offline diagnostics and describe the use and execution of each.
4. Describe the steps in booting the Offline Loader. CHANGE TAPE – HIT INIT

Lesson Outline

I. Overview
II. Commands
III. Diagnostics

Offline Diagnostics

- Test modules internal to the HSC.
- Can only be run by booting the HSC using the offline tape/diskette; this runs the Offline Diagnostic Loader (ODL) instead of CRONIC.
- The HSC must be offline (online switch out).
- Booting to the ODL> prompt takes 30 seconds on the HSC70, and approximately 2 minutes on the HSC50.
- While ODL waits for a request, a rotating pattern is continually passed through the indicators on the OCP.
- Available commands under ODL>:

  HELP
  EXAMINE
  DEPOSIT
  REPEAT
  SET/SHOW DEFAULT
  LOAD
  START
  SET RELOCATION
  SIZE — LIKE AUTOSIZER

- Available commands under ODL> that initiate testing:

  TEST CACHE
  TEST BUS
  TEST MEM
  TEST MEM BY K
  TEST REFRESH
  TEST OCP
  TEST RX (HSC70 only)
  TEST K

Convenience Commands Under ODL

HELP	Supplies an abbreviated list of all commands that the Offline Loader recognizes.
EXAMINE	Allows the user to examine the contents of any location in HSC memory (Program, Data, or Control) using a 22-bit address.
DEPOSIT	Allows the user to modify the contents of any location in HSC memory (Program, Data, or Control) using a 22-bit address.
REPEAT	Allows repeated EXAMINE or DEPOSIT commands.
SET/SHOW DEFAULT	Allows the user to set or show the number base (octal, decimal, or hex), or the data length used by ODL.
LOAD	Allows the user to load a program into HSC memory without starting it.
START	Starts the last test loaded into HSC memory.
SET RELOCATION	Loads a relocation register with a base address that is then added to address values used with EXAMINE and DEPOSIT commands. All subsequent EXAMINE and DEPOSIT commands use this base as a relocation value.
@	Allows execution of indirect command files located on the tape/diskette; no commands are available at this time.

Size Command

- Identifies available HSC memory.
- Determines type of requester in each slot.
- Initiates ROM-based self-tests on each K. The tests are:

K.sdi	K.sti	K.ci
0 - 2911	0 - 2911	0 - 2911
1 - 2901	1 - 2901	1 - 2901
2 - D bus	2 - D bus	2 - D bus
3 - C bus	3 - C bus	3 - C bus
4 - PROM parity	4 - PROM parity	4 - PROM parity
5 - HSC memory	5 - HSC memory	5 - HSC memory
6 - RAM	6 - RAM	6 - RAM
7 - SERDES/RSGEN	7 - SERDES	7 - PLI
10 - Partial SDI	10 - Partial STI	10 - PILA
		11 - Link

- Test #5 is not run on the SIZE command (it is run on the TEST MEM BY K command)
- Test #11 for K.ci will fail if no CI Path is present.

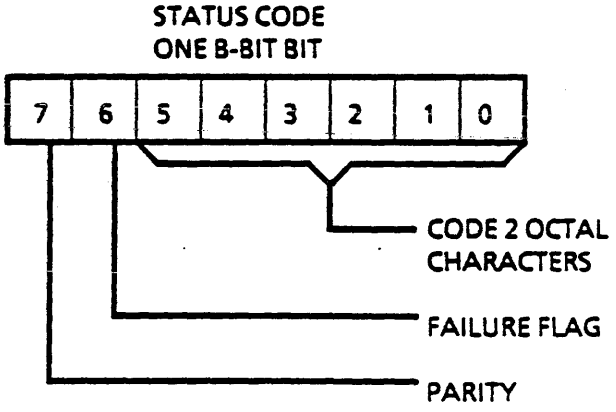
Size Command Example

```
ODL>SIZE

 HSC50 OFL  System  Sizer

Req.    Status Value    Meaning

001         111         Failed test 011
002         002         K.sdi
003         002         K.sdi
004         377         Empty requestor
005         377         Empty requestor
006         377         Empty requestor
007         377         Empty requestor

RESPONDING          ADDRESS            DECIMAL
MEMORY              RANGE              WORDS         KWORDS

Program       00000000 - 00777777      0131072       00128
Data          14000000 - 14377777      0065536       00064
Control       16000000 - 16377777      0065536       00064

     HSC50 OFL Diagnostic Loader.  Version V110
     Radix = Octal, Data Length=Word, Reloc=00000000

ODL>
```

Interpreting Requestor Status

Bit 7	Parity (ignore it).
Bit 6	Indicates how to interpret bits 0 through 5.
Bit 5:0	Bit 6 SET indicates that the K in the corresponding backplane slot has failed one of its initialization tests. Bits 0 to 5 contain the failing test number. Bit 6 CLEAR indicates that the K in the corresponding backplane slot has passsed its initialization tests. Bits 0 to 5 contain a code that identifies the type of K.

1 = K.ci
2 = K.sdi
3 = K.sti



ODL Test Commands

TEST CACHE	Tests cache logic and functionality of on-board cache on P.ioj (HSC70 only).
TEST BUS	Creates Control and Data bus contention among selected requesters; at least two requesters needed to run test; CTRL/C to return to offline loader.
TEST MEM	Tests memory from P.ioc/P.ioj; allows selection of memory area to test; only test for Program Memory; takes time (32k/hour).
TEST MEM BY K	Commands a K (K.sdi, K.sti, K.ci) to do memory testing; only Control or Data Memory tested; fastest way to test memory.
TEST REFRESH	Finds memory problems related to refresh; all three HSC memories tested (some static RAM failures resemble refresh problems).
TEST OCP	Checks operation of HSC lamps and switches.
TEST RX	Diagnostic and exerciser for memory board (M.std2) and RX33 drive (HSC70 only).
TEST K	Commands a K to perform internal microdiagnostic self-test (ROM-based); allows selection of specific K to test; only individual tests can be selected; test #5 actually TEST MEM BY K test; use SIZE command to do all tests.

HSC SYSTEM OPERATION

HSC System Operation

3.9-4 MOST CURRENT

Lesson Introduction

This module discusses the HSC Operating System, also known as the Colorado Rudimentary Operating Nucleus for Intelligent Controllers (CRONIC).

During troubleshooting, the engineer may encounter a system crash on the HSC. Therefore, system error messages and crashes are also covered in this module, as well as ODT and uODT.

Lesson Objectives

1. Describe what ODT and uODT are.
2. Describe how ODT and uODT are used.
3. List the available commands under the operating system and describe each one.
4. Analyze a system crash and replace the failing FRU.

Lesson Outline

I. The HSC Terminal
II. HSC Commands and Messages
III. The System Configuration Table
IV. uODT
V. HSC ODT
VI. HSC Processes
VII. Crashes

Operating the HSC From the Console

- Once CRONIC is up and running, the HSC functions as a mass storage controller requiring minor operator intervention.
- The HSC ignores console input until the operator types CTRL/Y (or CTRL/C), which produces the prompt:

  HSC>

- This prompt is called the Keyboard MONitor prompt (KMON).
- The BREAK key halts operation of HSC and returns an @ symbol indicating the HSC has been put into uODT state.

  a. From the @ symbol, the only means of returning to normal operation is to type "P" (proceed).

  b. The secure/enable switch will disable BREAK, but will also buffer it on the P.io board. HIT SPACE BAR SEVERAL TIMES

  c. The best way to disable BREAK is by using Set-up of console terminal.

- Holding the INIT switch in and typing on the console will echo the characters typed back to the console (loopback test). ALSO REBOOTS HSC!

HSC Console Commands

- RUN unit:name
  - Initiates an inline diagnostic or utility.
  - Unit is the load device on which the program resides.
  - Example:

    RUN DD1:DKUTIL
- SET <paramater> -- changes an attribute under the SETSHO utility.
- SHO <paramater> -- shows the present state of an attribute under the SETSHO utility.
- The DIR unit:
  - Outputs a directory of the load medium specified by unit.
  - If unit is not specified, the default is the device used for the last boot.
  - Legal units:

    DX0: or DX1: for HSC70
    DD0: or DD1: for HSC50
    LB: or SY: for device used for last boot
- PURGE
  - Removes all cached programs from Program Memory.
  - Not related to cache memory of HSC70.
- EXIT
  - Dumps user out of KMON prompt. SAME AS TIMEOUT AFTER NO RESP ON KYBD.
  - Performs the same function as CTRL/Z.

HSC Console Commands (Cont.)

- The following input control characters are acceptable on the HSC console:

CTRL/Y or CTRL/C	Abort operation, get the system's attention
CTRL/G	Input a change of test parameter
CTRL/O	Kill output line
CTRL/U	Kill input line
CTRL/S	XOFF
CTRL/Q	XON
CTRL/Z	Exit
DEL	Delete last input character

- Output messages from the console have a warning or severity code:

  Q = Inquiry

  I = Information

  F = Fatal error

  W = Warning

  E = Error

  S = Successful completition

System Configuration Table

- Contains information about the HSC configuration that must be preserved from boot to boot.
- Located on system tape/diskette (SCT.INI).
- Set to default values by holding Fault switch in during an INIT.
- Some of the parameters that are contained in the SCT:

  SYSTEM ID
  SYSTEM NAME
  FRONT PANEL (secure/enable switch position)
  BOOT DATE AND TIME
  CRONIC VERSION
  SECTOR SIZE
  AUTOMATIC DIAGNOSTICS

- The SETSHO utility is used to change or view contents of the SCT.

VMS V5.x

$>SET TIME
FROM ANY NODE
IN CLUSTER
SETS ALL
NODES TO
SAME TIME!

HSC ODT and uODT

- Occasionally, the console may display two prompts other than the usual KMON prompt:
  - An asterisk (*) indicates HSC ODT state.
  - An @ symbol indicates a uODT state.
- HSC ODT — NEVER USED!
  - A subset of RT-11 ODT.
  - Entered by typing CTRL/P (with ODT enabled under SETSHO), or after HSC crashes (with DUMP_BPT enabled under SETSHO).
  - Software-related -- used to examine and format data structures in the Control and Data Memory.
  - Exists as a file on the load device.
  - Can be exited by typing ";P". "SEMI COLON P"
- HSC uODT
  - Entered by pressing the BREAK key.
  - Hardware-related -- used to examine and deposit Control Memory.
  - Disables MMU so it only accepts 16-bit addresses.
  - Can be exited by typing the letter P.

HSC Processes

EXEC	Acts as the priority scheduler for all other processes attempting to use the HSC CPU.
DISK	The MSCP server for the HSC; directs and oversees all I/O for all attached disk drives.
TAPE	The TMSCP server for the HSC; performs all tape-related functions within the HSC.
ECC	Error Correction Code; performs error correction on data being moved from disk to host.
VTDPY	A utility that displays overall HSC system operations on a video terminal.
BACKUP	Used for disk-to-tape backups.
VERIFY	Checks disk format.
DEMON/PDEMON	Diagnostic Execution Monitor; controls error handling in the HSC.
DUP	Diagnostic and Utilities Protocol server; allows host to act as a console terminal.
POLLER	Used to continuously poll other CI nodes to detect transitions in the cluster configuration.
SCSDIR	Provides other CI nodes with information about which processes are available for outside use.
LOADER	Allocates space for system processes, utilities, or diagnostics in memory.
HOST	Opens virtual circuits, starts connection start-up protocol, and provides services to SYSAPS for connection maintenance.

HSC INLINE DIAGNOSTICS

HSC Inline Diagnostics

Lesson Introduction

This module discusses the inline diagnostics. These diagnostics are run in conjunction with the operating system. The HSC is still able to process CI, SDI, and STI communications while inline diagnostic testing is in progress.

Lesson Objectives

1. Run each of the inline diagnostics.
2. Explain the use of each inline diagnostic.
3. Interpret any error received while running an inline diagnostic.

Lesson Outline

I. Introduction
II. DUP and Demand Diagnostics
III. Periodic Diagnostics
IV. Automatic Diagnostics

Inline Diagnostics

- Different from offline diagnostics because they can be run without taking the HSC offline (CRONIC MUST be up and running).
- Does not interfere with the normal operation of HSC.
- Located on a utility tape (HSC50) or on the system diskette (HSC70).
- Device (tape or disk) being tested must be in an available state.
- Can be started four ways:
  - On demand (from console) HSC> RUN DDI: ILEXER
  - DUP DIAG. UTILITY PROTOCOL — RUN FROM VAX HOST
  - Automatic diagnostics IF HSC DETECTS AN ERROR DURING NORMAL OPERATION.
  - Periodic diagnostics

Inline Diagnostics Initiated on Demand or from DUP

- Inline diagnostics are initiated "on demand" by typing CTRL/Y at the HSC console and using the RUN <diag name> command.
- Diagnostic and Utilities Protocol (DUP) cannot be used to run inline diagnostics if the Secure/Enable switch on the HSC is in the Secure position.
- DUP can be started on any VAX node using the following commands:

```
$ MCR SYSGEN
SYSGEN> CONNECT FYA0/NOADAPTER
SYSGEN> EXIT
$SET HOST/HSC HSC000
HSC50>
```

TO LOG INTO HSC FROM VAX !

default name

- Once the HSC prompt appears, the following inline diagnostics can be started:
  - ILRX33 (HSC70 only)
    a. Verifies data can be written/read from diskette.
    b. Contents of block 1 read and saved; three different data patterns are written to and read from block 1; original contents are returned to block 1. USE SCRATCH JUST IN CASE
  - ILTU58 (HSC50 only)
    a. Initiates self-test of TU58.
    b. Reads and saves physical block 1, writes and reads several data patterns, then writes and reads back original contents.
  - ILMEMY -- tests data buffers that have been removed from service (previously placed on the Disabled Buffer Queue) and displays results on terminal.

FOR DUP TO WORK YOU MUST HAVE

→ DUP LOADED TO CK →

ON HSC> RUN SETSHO
SETSHO> SHO LOAD
LOOK FOR DUP
IF NO DUP
DO
SETSHO> SET LOAD DUP
EXIT
ENABLE REBOOT
REBOOT

Inline Diagnostics Initiated on Demand or from DUP (Cont.)

- ILDISK

  a. Isolates problem to disk drive, SDI cable, or K.sdi module.

  b. Selects drive to be tested by unit number, or requester and port.

  c. Checks SDI connection, commands drive to do internal drive diagnostics, and commands drive to do read and write tests to diagnostic cylinder area.

- ILTAPE

  a. Tests tape formatter.

  b. Executes functional test of tape transport

- ILTCOM – checks to see that tapes generated on one tape drive can be read on another tape drive (compatibility).

- ILEXER

  a. Tests multiple disks and tapes.

  b. Issues random READ, WRITE and COMPARE commands to selected drives.

Inline Diagnostics Initiated Periodically

- Run when the system is performing little or no I/O.
- Can be enabled/disabled using SETSHO utility.
- Run at specific time intervals that are defined under the SETSHO utility.
- The following diagnostics are available:
  - PRMEMY
    a. Tests control bus transceivers, data bus transceivers, and parity detection.
    b. Successful completion of test transparent to user.
    c. Failure causes HSC crash.
  - PRKSDI
    a. Runs a single microdiagnostic on any Disk Data Channel module (K.sdi) not busy at the time.
    b. Different microdiagnostic runs with each successive call to this routine.
    c. Successful completion of test transparent to user.
    d. Failure causes HSC crash.
  - PRKSTI -- same as PRKSDI, but for any Tape Data Channel module.

Inline Diagnostics Initiated Automatically

- Two inline diagnostics are run without any user intervention:
  - ILMEMY
    a. Initiated when parity errors occur in data buffers during normal system operation.
    b. Buffers that fail test are put on Disabled Buffer Queue.
  - ILDISK -- initiated automatically when an unrecoverable disk failure occurs.
- Can be enabled/disabled under the SETSHO utility.

HSC AS A DSA CONTROLLER

HSC as a DSA Controller

Lesson Introduction

This module discusses the Digital Storage Architecture (DSA) as it pertains to the HSC Storage Controller. It is important to understand DSA in order to use some of the utilities. Knowledge of DSA is also very helpful when troubleshooting problems relating to mass storage devices in a cluster environment.

Lesson Objectives

1. Describe the basic difference between Digital's older architecture and DSA.
2. Define and describe the different types of MSCP used in a cluster environment.
3. Define and describe SDI, STI, DSDF, and ANSI.
4. Discuss where each of the above is used in a cluster.
5. Describe how this relates to troubleshooting in a cluster.

Lesson Outline

I. DSA in General
II. Responsibilities
III. DSDF
IV. SDK

DSA in General

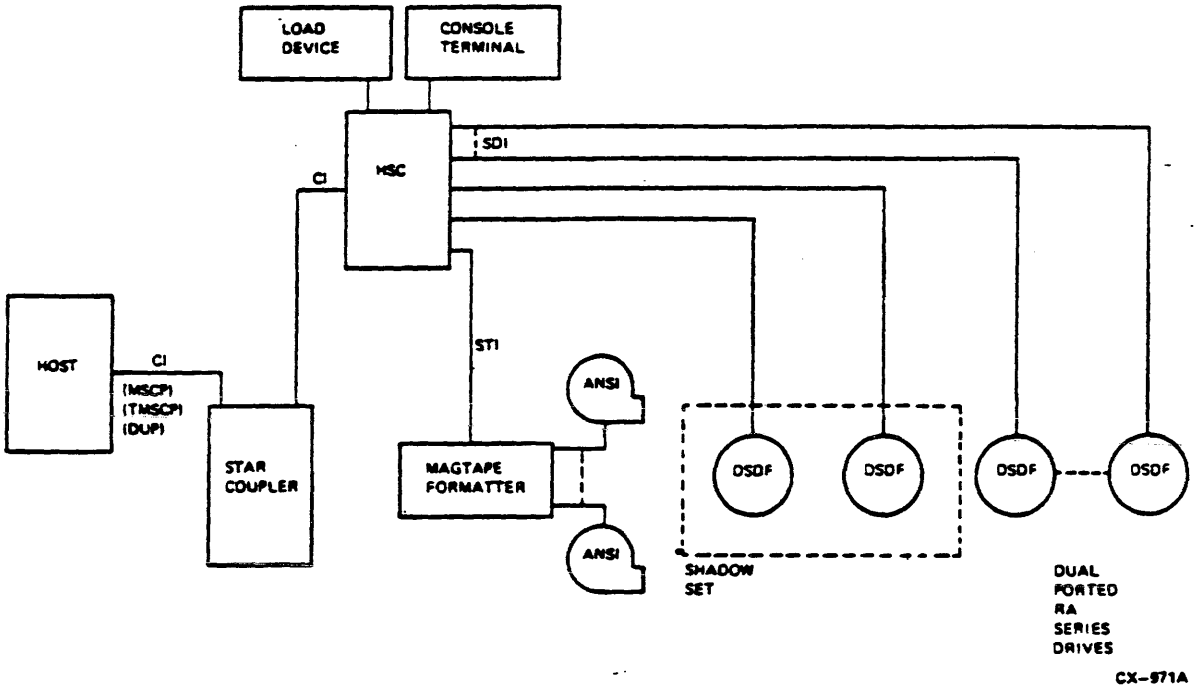
- In the Digital Storage Architecture (DSA) family, responsibilities have been redistributed.
- Management of I/O operations has been moved from the host to the controller.

ERR. DETECT
&
CORRECT
OPTIMIZATION
etc.

DSA in General (Cont.)

- MSCP (Mass Storage Control Protocol) defines a standard set of operations that DSA controllers perform.
- MSCP defines the responsibilities of:
  - The host.
  - The protocol used between the host and the controller.
  - The controller.
- TMSCP is the MSCP for tape controllers.
- SDI (Standard Disk Interconnect) is the protocol defining the relationship between a DSA controller and a disk unit, as well as the signals found on the cable connecting the two.
- STI (Standard Tape Interconnect) is the protocol defining the relationship between the controller and the tape unit, as well as the signals found on the cable connecting the two.
- DSDF (Digital Standard Disk Format) defines the layout and format of any disk attached to a DSA controller.
- ANSI defines the format of any magtape unit attached to a DSA controller.

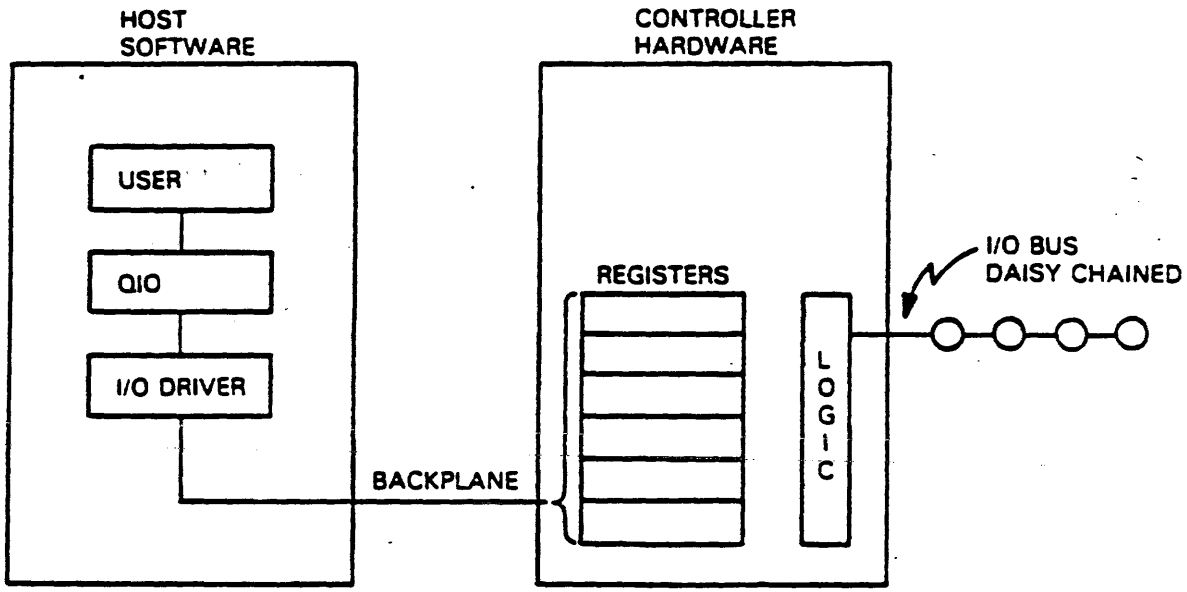
- DUP — SET HOST TO HSC FROM VAX


LOAD DEVICE
CONSOLE TERMINAL
HSC
SDI
CI
HOST
CI
(MSCP)
(TMSCP)
(DUP)
STAR COUPLER
STI
MAGTAPE FORMATTER
ANSI
ANSI
DSDF
DSDF
DSDF
DSDF
SHADOW SET
DUAL PORTED RA SERIES DRIVES
CX-971A

Non-DSA Devices

- I/O management is performed by the I/O driver host software.
- Controller hardware responds to bits in registers set by the I/O driver.
- One I/O operation involves multiple reads and writes to the controller registers (CPU time).
- Controller hardware provides a data path with little buffering.
- Controller has no decision-making power.
- Examples of non-DSA devices:

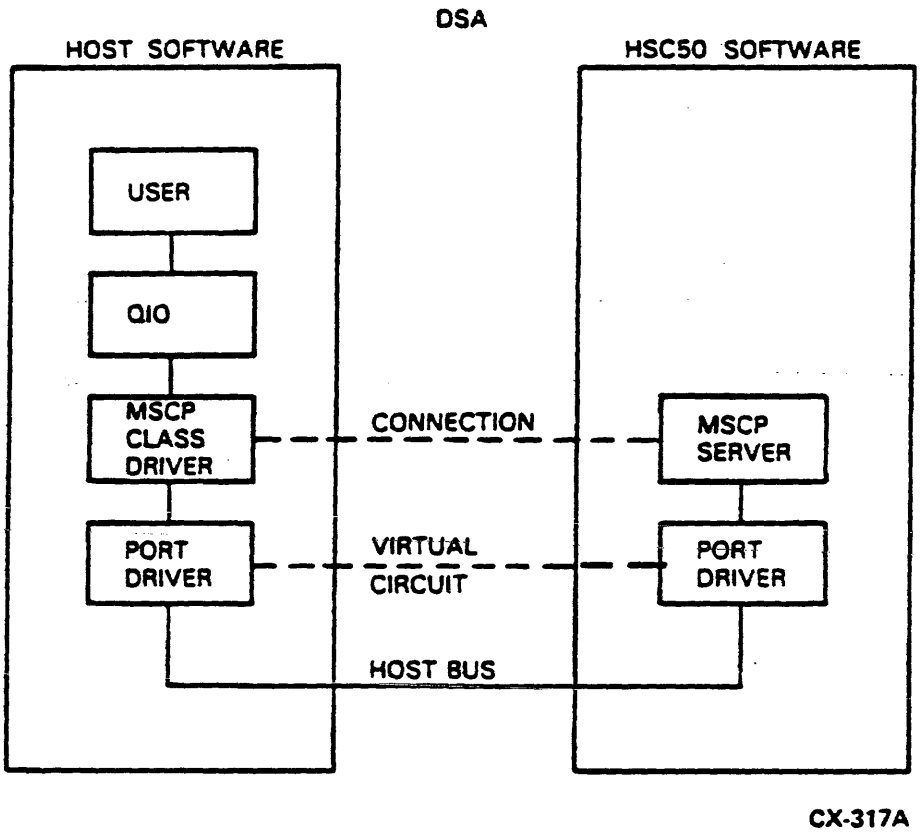
Disks	Controllers
RM05	Mass Bus Controller (MBA)
RL02	RX211
RP06	RK611
RK06	RL11
RK07	
RX02	


HOST
SOFTWARE
CONTROLLER
HARDWARE
USER
QIO
I/O DRIVER
BACKPLANE
REGISTERS
LOGIC
I/O BUS
DAISY CHAINED
CX-316A

DSA Devices

- I/O management resides primarily in the controller software entity called the MSCP Server.
- The host MSCP class driver sends packets of information to the MSCP Server over a logical connection.
- Actual delivery of the packets is bus-dependent (CI, UNIBUS, Q) and is handled by the port driver software.
- A typical data exchange between host memory and a mass storage media on any DSA subsystem involves the following:
  - The host class driver sends the MSCP packet to the controller requesting an exchange.
  - The host is now free to perform other tasks -- the requested transaction is considered outstanding.
  - The controller performs disk head positioning, buffering of data, and transfer of data to host memory.
  - The host receives the MSCP end message from the MSCP Server (through the port driver), indicating exchange has been completed.
- Controller essentially lightens word load of CPU.
- Sample DSA devices:

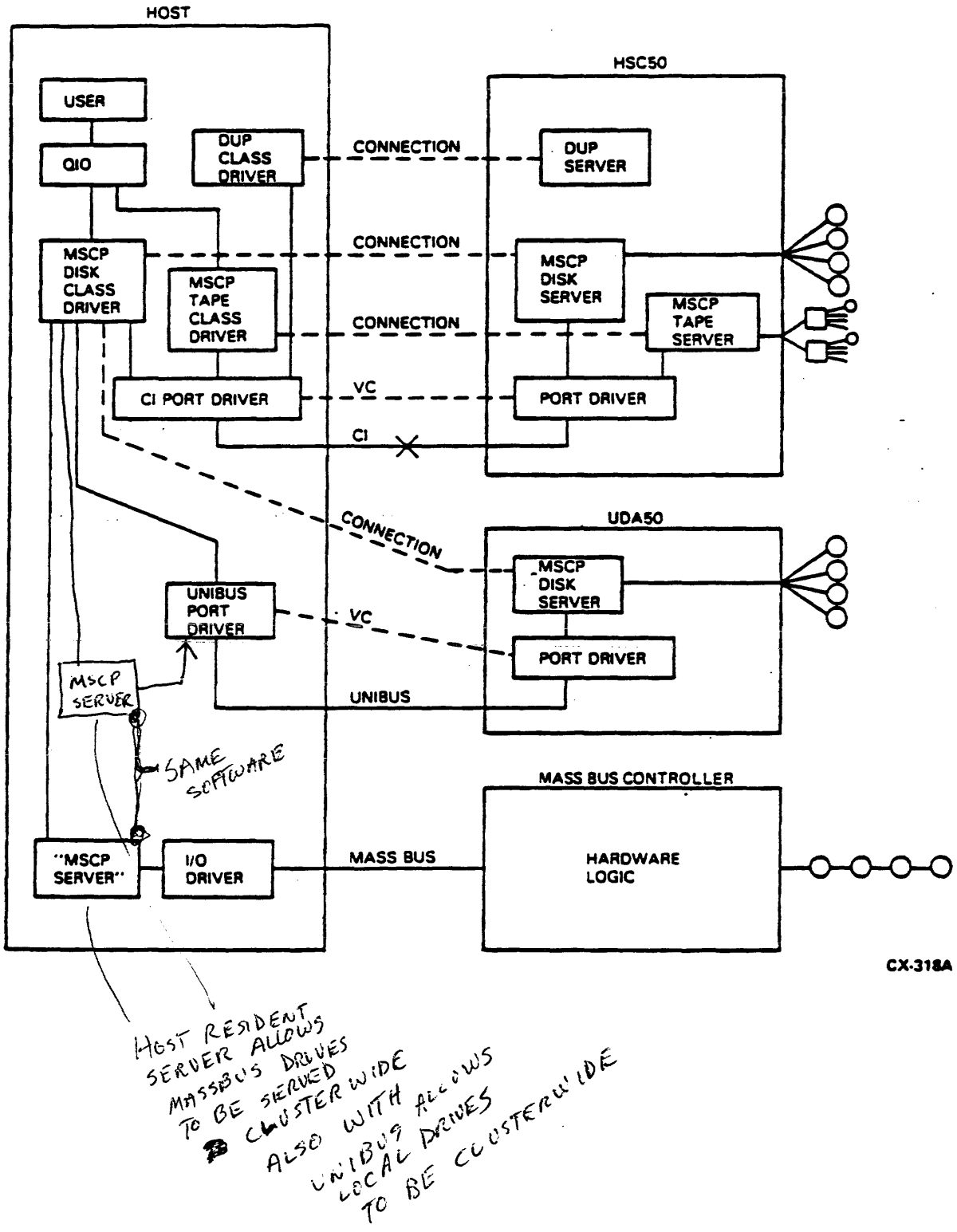
Disks	Controllers
RA60	UDA50 – UNIBUS
RA80	KDA50 – Q BUS
RA81	KDB50 – BI BUS
RA82	HSC – CI BUS
RA70	


DSA
HOST SOFTWARE
HSC50 SOFTWARE
USER
QIO
MSCP
CLASS
DRIVER
CONNECTION
MSCP
SERVER
PORT
DRIVER
VIRTUAL
CIRCUIT
PORT
DRIVER
HOST BUS

CX-317A

DSA Devices (Cont.)

- Each host CPU needs one Disk Class Driver for all DSA controllers and disks attached to it.
- Each host CPU needs one Tape Class Driver for all tape DSA controllers and tapes attached to it.
- MSCP commands received by the controller can be queued and rearranged for best throughput.
- Format of the MSCP message is the same for all controllers, regardless of the bus that carries the message.
- A non-DSA disk can be used as a DSA disk via the MSCP Server, which performs the necessary translation to DSA protocol; this feature is only available in a clustered environment.
- The DUP (Diagnostic and Utility Protocol) is a system application-level protocol used by a host to direct diagnostics on an HSC; it is implemented as a class driver on the host and as a corresponding server on the HSC.



HOST RESIDENT SERVER ALLOWS MASSBUS DRIVES TO BE SERVED CLUSTER WIDE
ALSO WITH ALLOWS UNIBUS LOCAL DRIVES TO BE CLUSTERWIDE

DSA Connections

DSA From the Host View

- Sees controller as a slave that can perform I/O operations specified by the MSCP.
- Expects to receive an MSCP end message for each MSCP command it generates. GUARANTEED MESSAGES
- Aborts the I/O operation if the end message is not received within time-out period.
- Accepts unexpected MSCP Error and MSCP Attention messages from controller.
- Views a mass storage unit as one of two classes:
  - Disk Class -- all RA drives
  - Tape Class -- all TA drives
- Views data on these storage devices as one contiguous string of logical blocks.
- Does not see media geometry such as physical cylinders, tracks, and heads.
- Does not see the revectoring process that occurs when a bad block is encountered during a read operation.
- Is totally responsible for file structure and file protection.

DSA From the Controller View

- The controller performs the following upon receiving an MSCP packet from the class driver:
  - Decodes the MSCP command from the host.
  - Queues the operations.
  - Performs seek and rotational optimization.
  - Transfers data to/from the mass storage unit.
  - Performs error checking, error handling, and error correction.
  - Revectors bad blocks.
  - Transfers data to/from host memory.
  - Sends MSCP End message back to the host upon completion of the operation.
  - Sends MSCP Error or Attention message back to host.
- Controller is not responsible for knowledge of file structure, data protection, or access restrictions.
- Interfaces with disk units only over the SDI bus and tape units over the STI bus.
- Continually monitors its connected units, noting their type, operating characteristics, and status.
- Responsible for translation of Logical Block Number (LBN) to a physically addressable sector on disk unit.

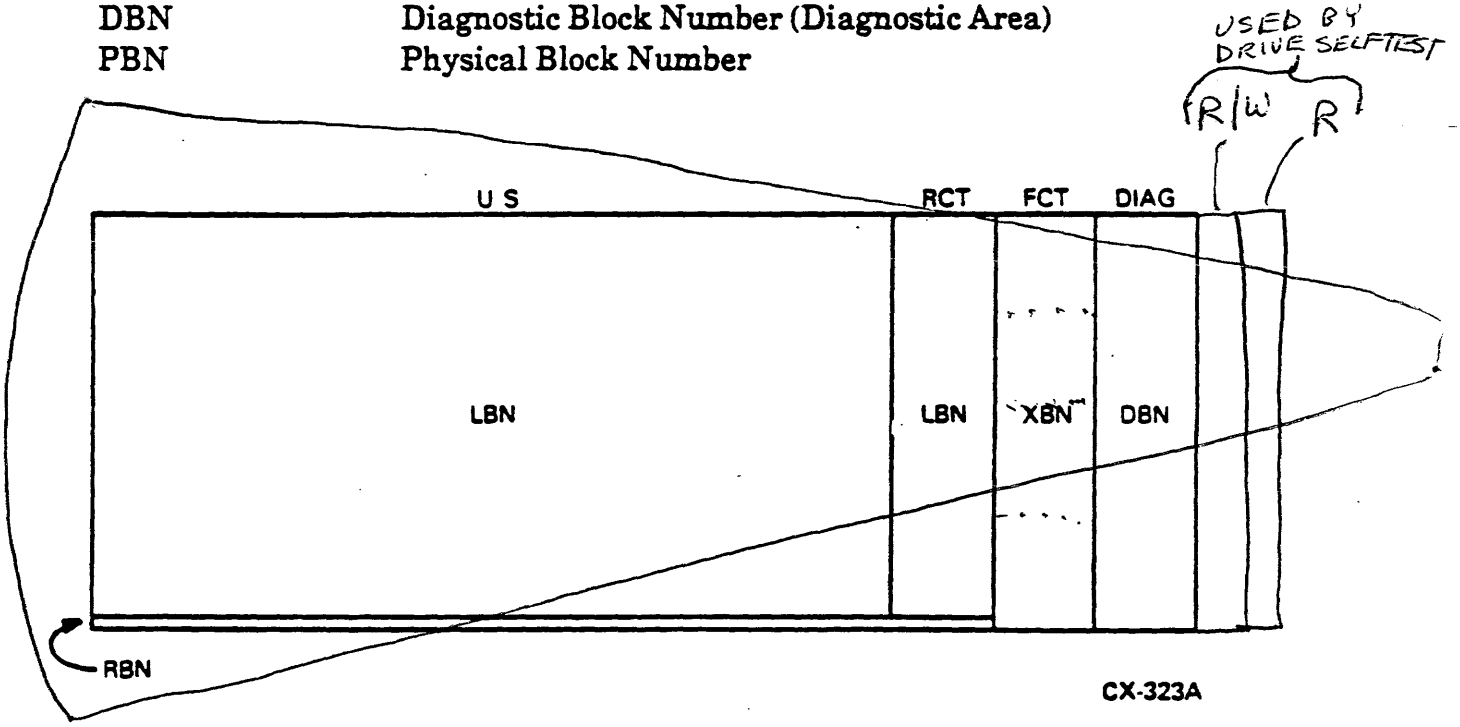
DSA From the Unit View

- All disk units that are designated RA use an SDI cable and SDI protocol to communicate with the controller.
- All tape units that are designated TAxx use an STI cable and STI protocol to communicate with the controller.
- All units have a unique unit number within a system (unit plug).
- All units support dual porting.
- All disk units use Digital Standard Disk Format (DSDF).
- All tape units use ANSI format.

DSDF Overview

- A disk is divided into five basic areas, according to LBN:
  - Host Application Area (user space)
  - Replacement Control Table (RCT) (REVECTOR)
  - Replacement Block Area
  - Format Control Table (FCT) — DO NOT GET REPLACED - HAS 4 COPIES
  - Controller Diagnostic R/W Area
- Each area is composed of blocks, the names of the blocks change according to what area the block falls within:

LBN	Logical Block Number (user space and RCT)
RBN	Replacement Block Number (Replacement Block Area)
XBN	Extended Block Number (FCT)
DBN	Diagnostic Block Number (Diagnostic Area)
PBN	Physical Block Number



DSDF Overview (Cont.)

- The following paragraphs are brief descriptions of the five sections that compose a Digital Standard Disk Format:

  - Host Application Area -- the largest area containing data blocks for use by normal host operating applications, as well as system operating software. This is the normal working area of the disk and where system and user files are stored.

  - RCT

    a. These tables record the location of all revectored logical blocks and the status of each replacement block on the unit.

    b. There are four copies of the table in the RCT space, each copy is organized in ascending order with an entry for each RBN of the unit. Each copy is stored on a different surface.

    c. The bad blocks listed in the RCT come from one of the following sources:

      - Previous format
      - Factory format information (found in the FCT)
      - Dynamic BBR (BAD BLOCK REPLACEMENT) done by VMS

  - Replacement Block Area

    a. Contains replacement blocks that are used to replace logical blocks in the host applications area.

    b. Is the last block on each track.

    c. If an RBN goes bad, another RBN may be used to replace it.

DSDF Overview (Cont.)

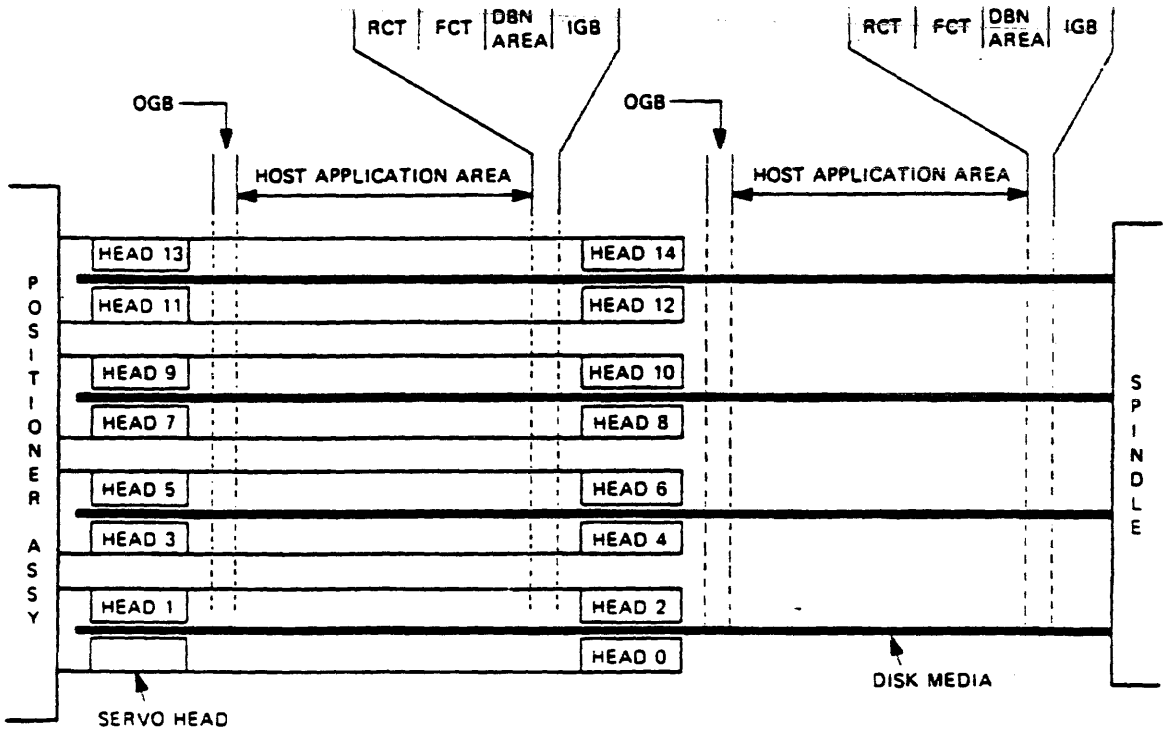
- FCT

  a. Contains the physical locations of the blocks found to be bad by the scanning process at the factory.

  b. Contains assorted information about the HDA such as media serial number, date of most recent format, number of times formatted, location of manufacturing-detected bad blocks, and indication as to whether the FCT is valid.

- Controller Diagnostic R/W Area -- contains blocks devoted to controller-resident diagnostic purposes; addressable by DBNs. DBNs are visible only to the controller and are used only when diagnostics are executed by the controller. DBNs are not replaced by RBNs when they become defective.

- The illustration below combines the physical and logical disk characteristics, showing how they relate to each other:

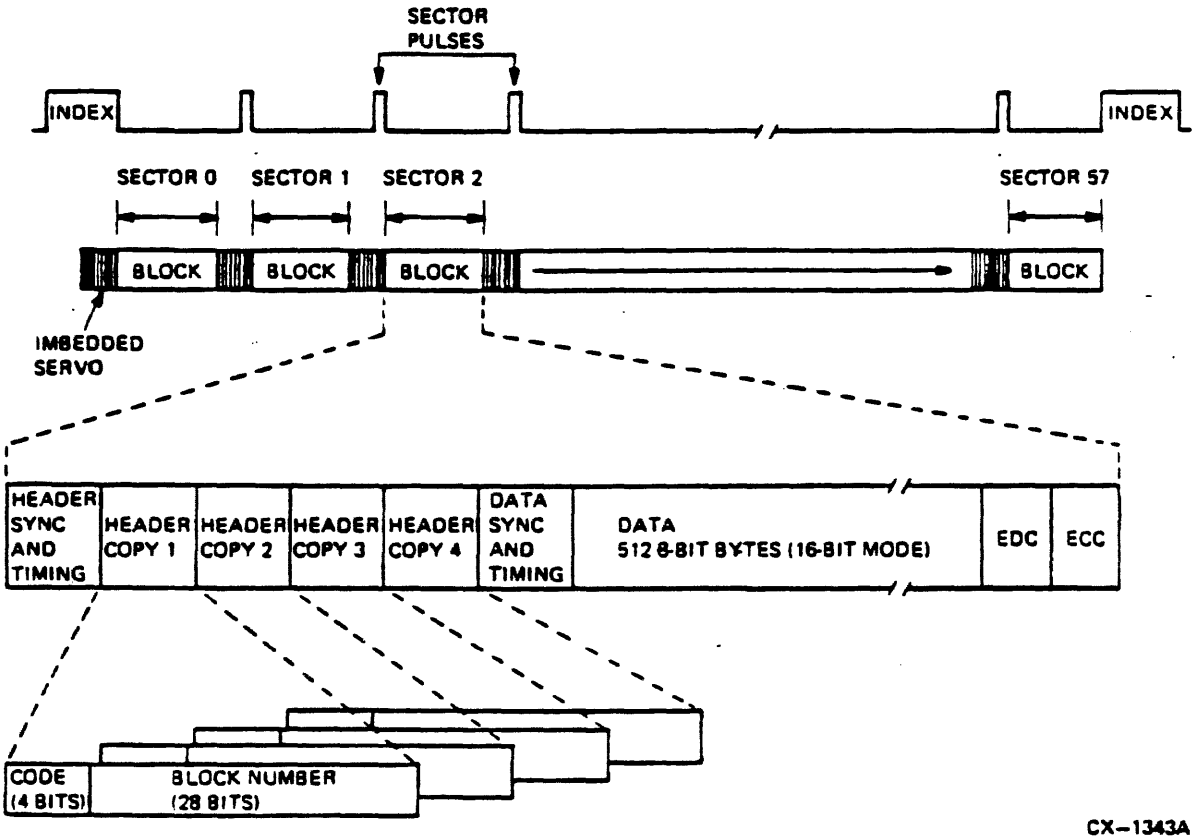


NOTE OGB = OUTER GUARD BAND
IGB = INNER GUARD BAND

CX-1049A

DSDF Overview (Cont.)

- The smallest addressable physical element on any disk is a Sector.
- From the host's viewpoint, a sector is a logical block.
- A sector contains 512 bytes of data.



DSDF Overview (Cont.)

- Each LBN sector contains four basic parts:

HEADER	There are four copies for each sector. Each copy contains 32 bits: a 28-bit LBN preceded by a 4-bit header code. The header code is used by the controller to identify the type of logical block that is being read and whether or not this block has been revectored. Two of the four header copies must match before a "header compare" succeeds.
DATA	The application-specific information recorded on the disk by the host.
EDC PARALLEL	The 16-bit Error Detection Code that is used to detect errors caused by problems internal to the controller. On the HSC, the EDC is computed when data is fetched from the host, then recomputed and checked when data is written to the disk.
ECC SERIAL	The 170-bit Error Correction Code. It is the last thing calculated and written to each sector, and the last thing to be recalculated at the end of each sector read. Thus, the new value is compared with that initially written on the disk and the type of error that results will enable the Read Solomon algorithm to find and correct up to 80 bits. The ECC is used to detect media-related problems (corrupted or deteriorating platter).

10 BITS OF ECC = 1 ECC SYMBOL.
CONTROLLER DRIVE COUNTS # OF ECC SYMBOLS TO DECIDE IF B.B.R. SHOULD BE DONE.

Bad Block Replacement

- Defects on the disk storage media cause sectors (blocks) to become unusable.
- Bad Block Replacement (BBR) is the replacement of a bad logical block (LBN) with a good replacement block (RBN).
- Once a block is replaced, attempted accesses to that block are "revectored" to the replacement block.
- Revectoring process executed by controller and is invisible to host.
- Important terms:

Bad Block Replacement	The process of moving data from a bad sector to a good sector and rewriting the header of bad sector to indicate that this has been done.
Revectoring	The process whereby access to a bad block is re-routed to the replacement block.

- Bad blocks in the RCT, FCT, and DBN areas are not replaced. The RCT and FCT contain four redundant copies of information to provide bad block protection.
- The 4-bit header code is the means by which the controller determines a logical block's availability.
- Two types of replacement:

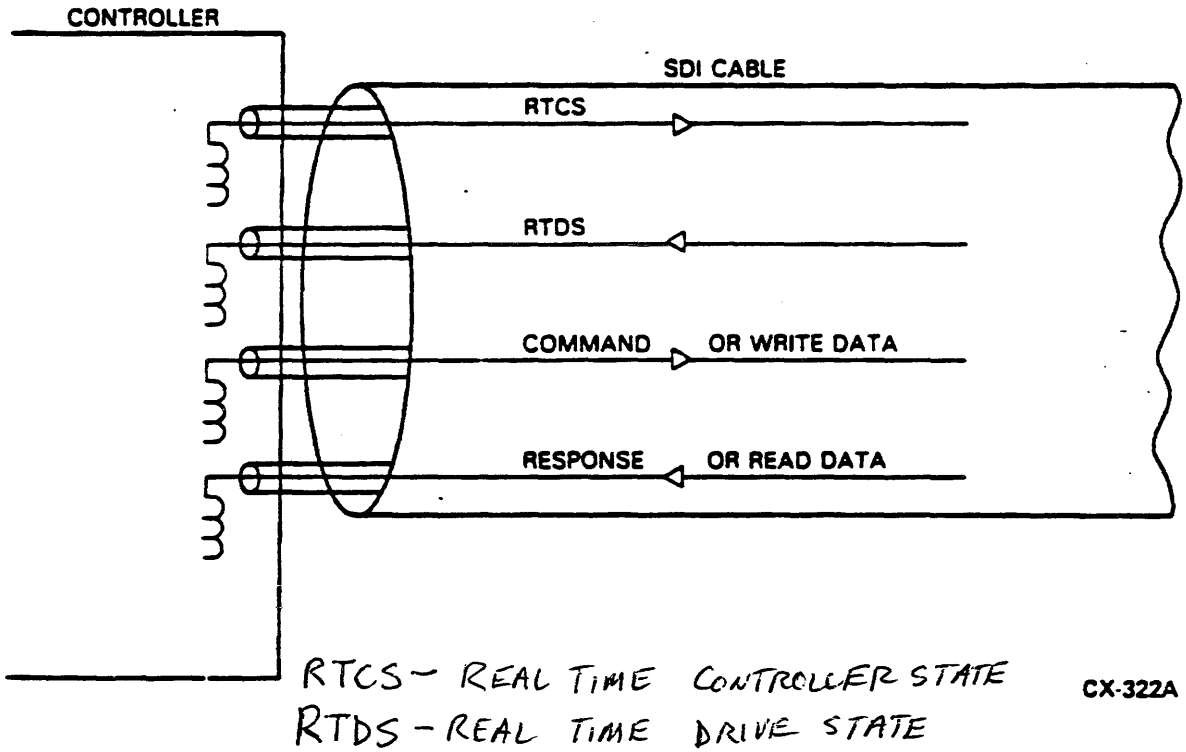
Primary Replacement	When the selected RBN resides on the same track as the block being replaced.
Nonprimary Replacement	When the selected RBN resides on a track other than the one containing the bad block being replaced.

BAD BLOCK REPLACEMENT

1. ECC ERRORS THAT EXCEED THRESHOLD
2. DATA MOVED TO RCT, ORIGINAL BLOCK R/W TESTED.
3. IF BAD — NEAREST RBN IS FOUND.
   A)- "PRIMARY" ~~AREA~~ RBN AT END OF TRACK
   B)- NEXT CLOSEST "SECONDARY" RBN
4. OLD LBN UPDATED — HEADER SET TO CURRENT STATUS
5. IF OLD DATA WAS UNCORRECTABLE — EDC INVERTED — "FORCED ERROR" FLAG SET

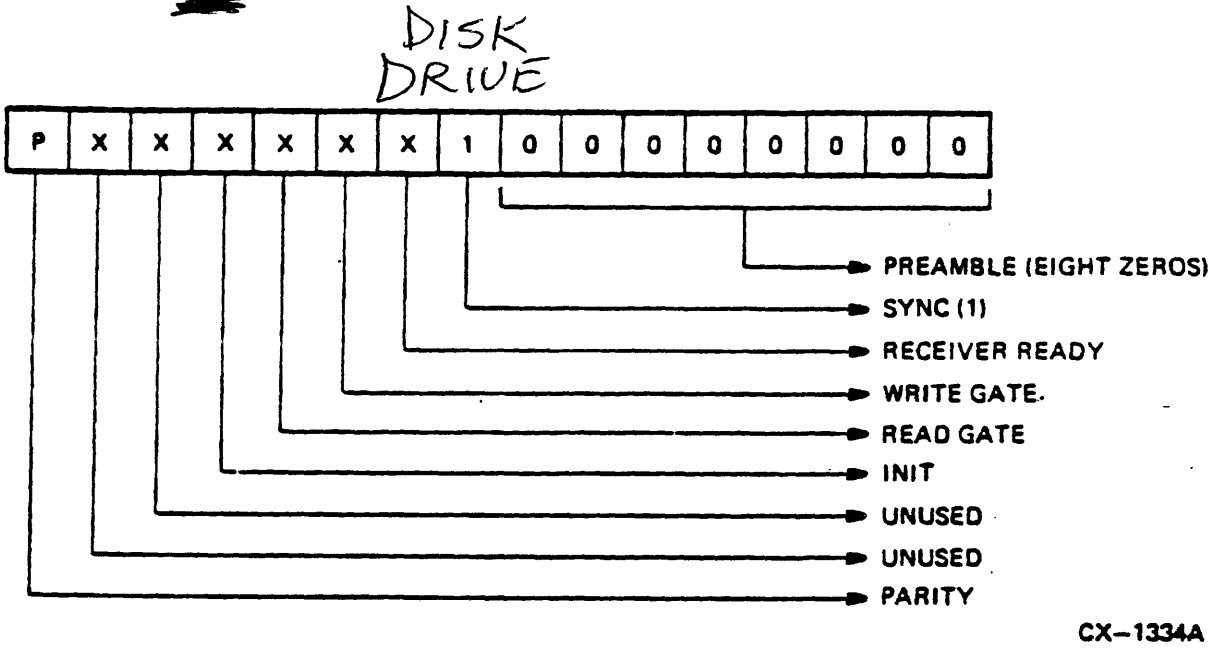
SDI Overview

- The SDI/STI interface cable consists of four lines: RTCs, RTDs, COMMAND or WRITE DATA, and RESPONSE or READ DATA.

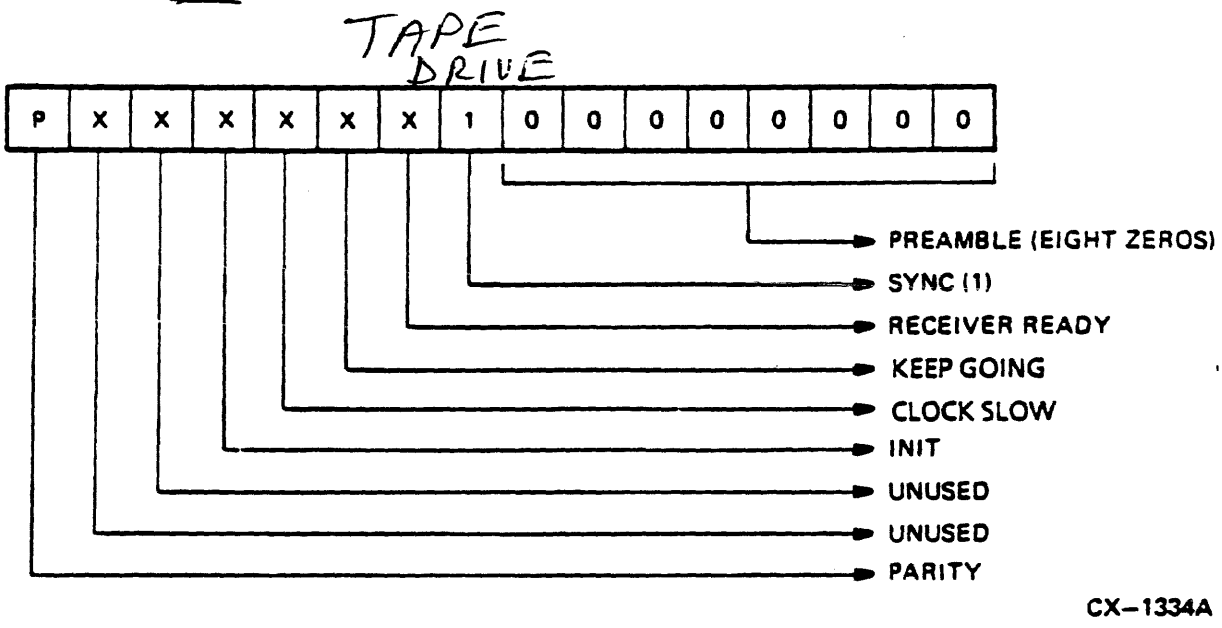


SDI Overview (Cont.)

- The Real-Time Controller State (RTCS) line passes information about the controller to the drive.
- For an SDI connection, the format of the RTCS burst is:

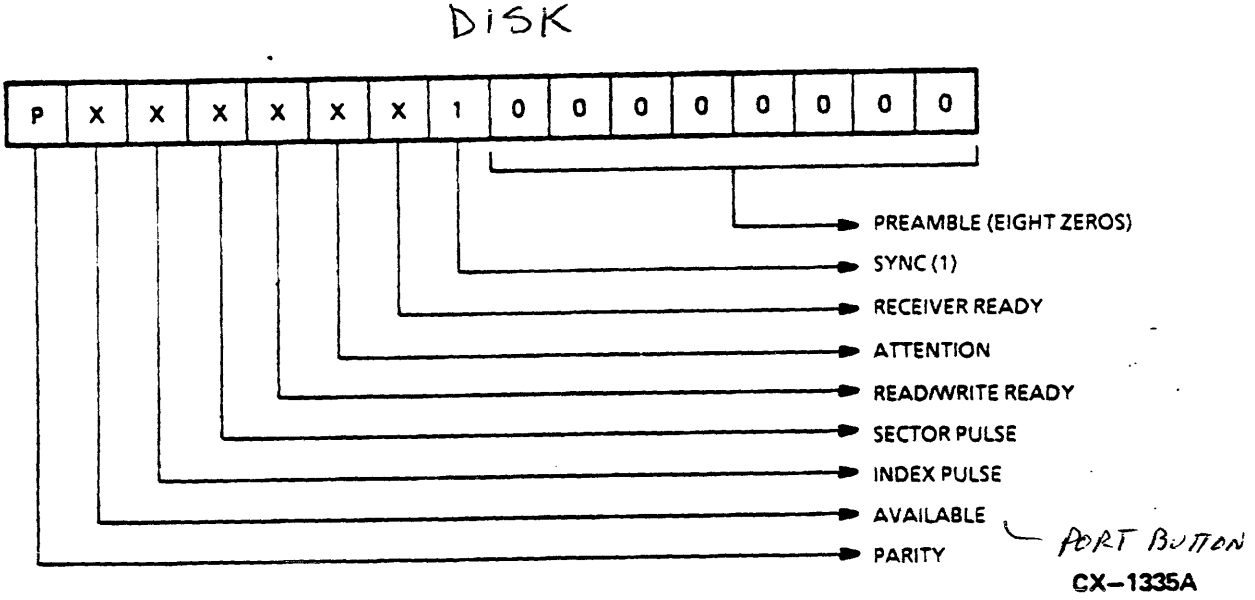


- For an STI connection, the format of the RTCS burst is:

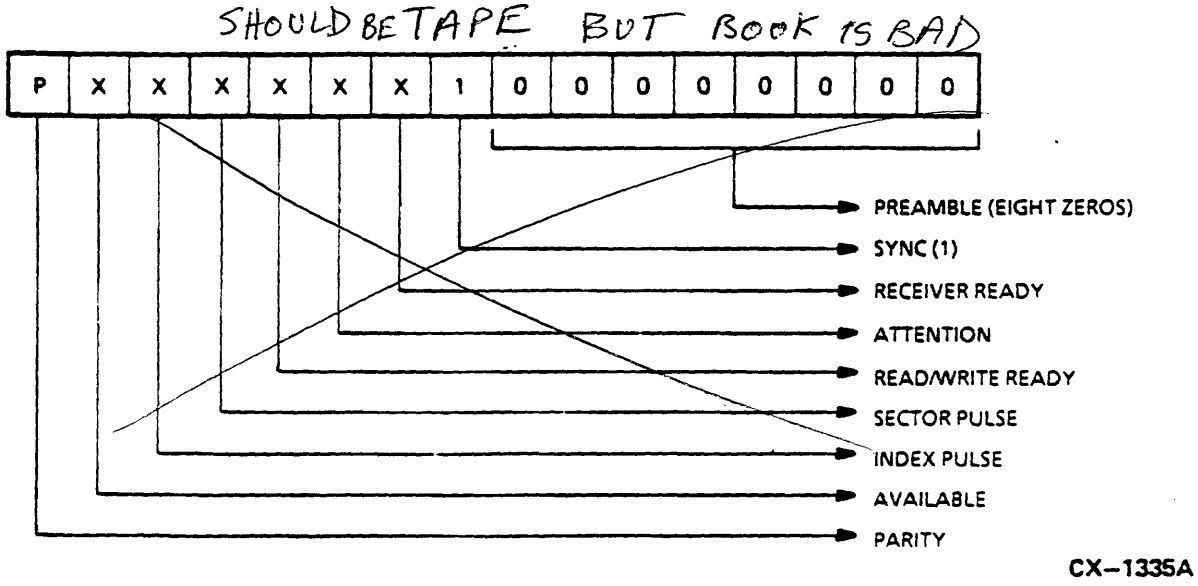


SDI Overview (Cont.)

- The Real-Time Drive State (RTDS) line passes information about the drive to controller.
- For an SDI connection, the format of the RTDS burst is:

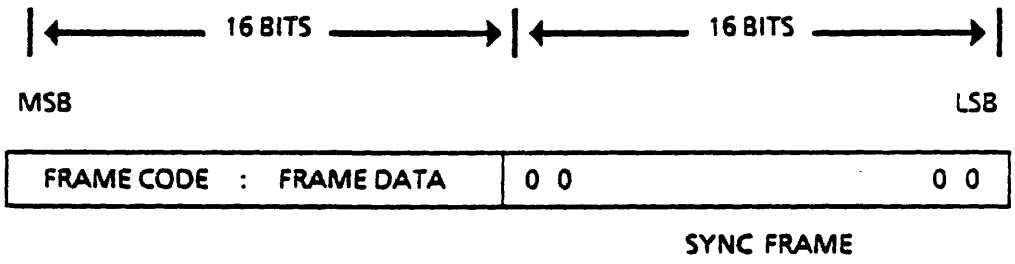


- For an STI connection, the RTDS line is really the Real-Time Formatter State, and can change depending on whether the formatter is in the available or online state; an example in the on-line state is:



SDI Overview (Cont.)

- On the Write/Command line, there can be one of two types of information:
  - Write data to disk
  - SDI command to disk
- SDI commands are either Level 1 or Level 2.
- Level 1 commands need no response from the drive on Read/Response line:



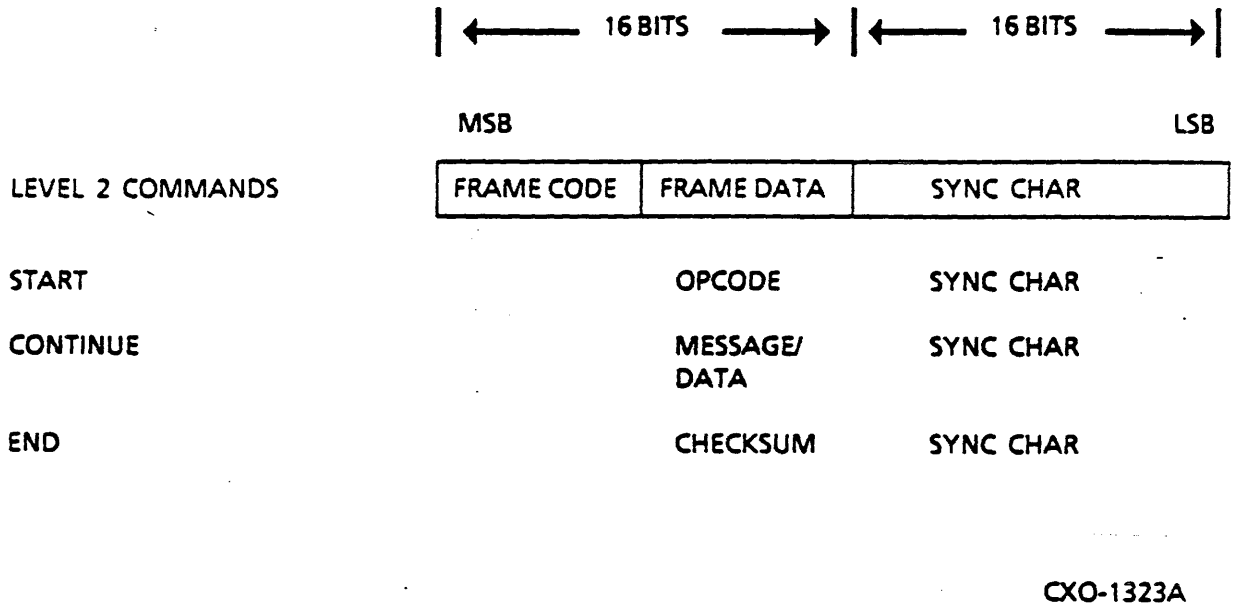
CXO-1321A

LEVEL 1 COMMANDS	FRAME CODE	FRAME DATA	SYNC CHAR
SELECT GROUP		GROUP #	SYNC CHAR
SELECT TRACK AND READ		TRACK #	SYNC CHAR
SELECT TRACK AND WRITE		TRACK #	SYNC CHAR
SELECT TRACK AND FORMAT ON INDEX		TRACK #	SYNC CHAR
FORMAT ON SECTOR OR INDEX		———	SYNC CHAR
DIAGNOSTIC ECHO		———	SYNC CHAR

CXO-1328A

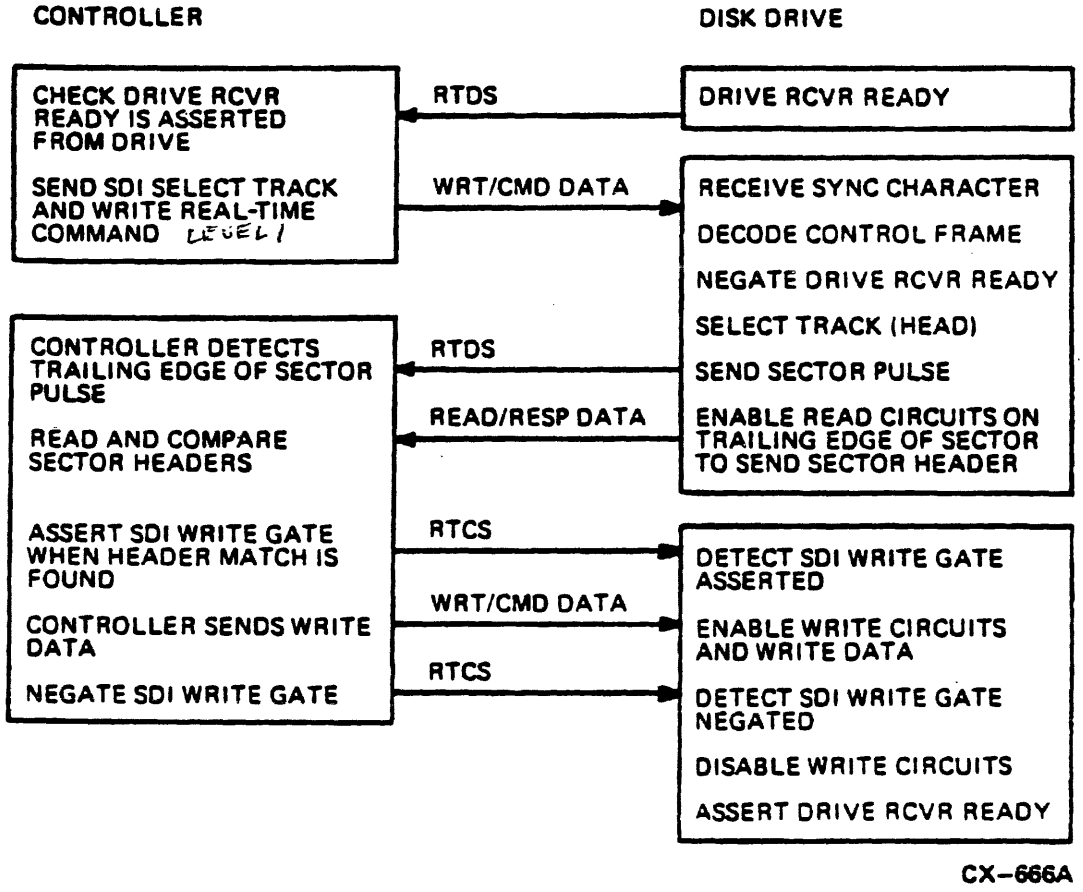
SDI Overview (Cont.)

- SDI Level 2 commands require multiple-frame transmissions from the controller.
- Level 2 commands also require a response from the drive on the Read/Response line.



SDI Overview (Cont.)

- All four lines of the SDI interface are used to implement a drive operation.
- A sample Level 1 command sequence (Select Track and Write) is given below:



HSC UTILITIES

HSC Utilities

Lesson Introduction

This module discusses the utilities that run under the HSC Operating System. Some are considered maintenance utilities while others are considered operational utilities.

The maintenance utilities are most often used to verify hardware. The operational utilities are operator-oriented tools for the integrity of data.

The HSC50 has these utilities split between two separate TU58 tapes; the HSC70 has all of these utilities on the same RX33 floppy.

Lesson Objectives

1. List the HSC utilities and describe the function of each.

2. Discuss how each utility is executed.

Lesson Outline

I. Introduction
II. Hardware Utilities
III. Software Utilities

HSC Utilities Overview

- Run with the CRONIC operating system up.
- Located on either the system tape or the utilities tape.

System tape	Utilities tape
SETSHO	VERIFY
COPY	FORMAT
	BACKUP
	RESTORE
	DKCOPY
	PATCH
	RXFMT
	DKUTIL
	VTDPY

- Can be divided into two general categories:
  - Hardware-related
  - Software-related

Hardware-Related Utilities

- VERIFY
  - Checks integrity of disk structure (DSDF).
  - Only reads disk; no bad block replacement or formatting done.
  - Use before and after formatting.
- FORMAT
  - Formats selected drive with 512-byte sector size.
  - Destroys user data.
  - Knowledge of DSDF (RCT, FCT, DBN, etc.) important to avoid destroying pack.
- DKUTIL
  - Displays disk structure and disk data.
  - Can be used as a VERIFY utility.
  - Can execute manual Bad Block Replacement (BBR).
- RXFMT -- formats and verifies the proper operation of the RX33 diskette.
- VTDPY
  - Displays the process activity within the HSC.
  - Only functions on a video terminal.

Software-Related Utilities

- SETSHO
  - Allows user to examine and alter HSC operating system parameters.
  - Accesses the System Control Table (SCT).
- BACKUP -- copies data on a disk drive to a tape drive as a backup copy. DOES RA81 IN 20 MINUTES. 4 TAPES. PHYSICAL BACKUPS ONLY
- RESTORE
  - Performs a restore of data that has been backed-up.
  - Does a read from magtape to disk only.
- PATCH -- used to modify system and utility files on a load device during a software patch.
- COPY
  - Performs backup of one load medium to another load medium.
  - Takes 5 minutes on an HSC70 and 20 minutes on an HSC50.
- DKCOPY -- copies the contents of one disk drive to a similar disk drive (RA81 to RA81, for example). FAST