CRAY-1 S
COMPUTER SYSTEM
I/O SUBSYSTEM STATION
OPERATOR'S WORKBOOK

T-0202

CHAPTER 1

I/O SUBSYSTEM HARDWARE OVERVIEW

INPUT/OUTPUT SUBSYSTEM

INCREASES CRAY-1 S CPU THROUGHPUT BY REDUCING ITS I/O AND FRONT-END RESPONSIBILITIES.

STREAMS DATA TO CENTRAL MEMORY OVER AN 810 MBIT/S CHANNEL.

PROVIDES ACCESS TO ADDITIONAL PERIPHERALS. (TAPES)

FUNCTIONS AS A MAINTENANCE CONTROL UNIT.

DRIVES UP TO 48 DD-29 DISK DRIVES FOR MASS STORAGE.

ALLOWS OPERATOR CONTROL OF COS.

COLLECTS AND CONCENTRATES DATA FROM FRONT-ENDS.

PROVIDES FOR JOB AND DATA ENTRY.

DISTRIBUTES CPU OUTPUT TO SLOWER PERIPHERAL EXPANDER DEVICES.

CONSISTS OF TWO TO FOUR I/O PROCESSORS WITH A SHARED BUFFER MEMORY.

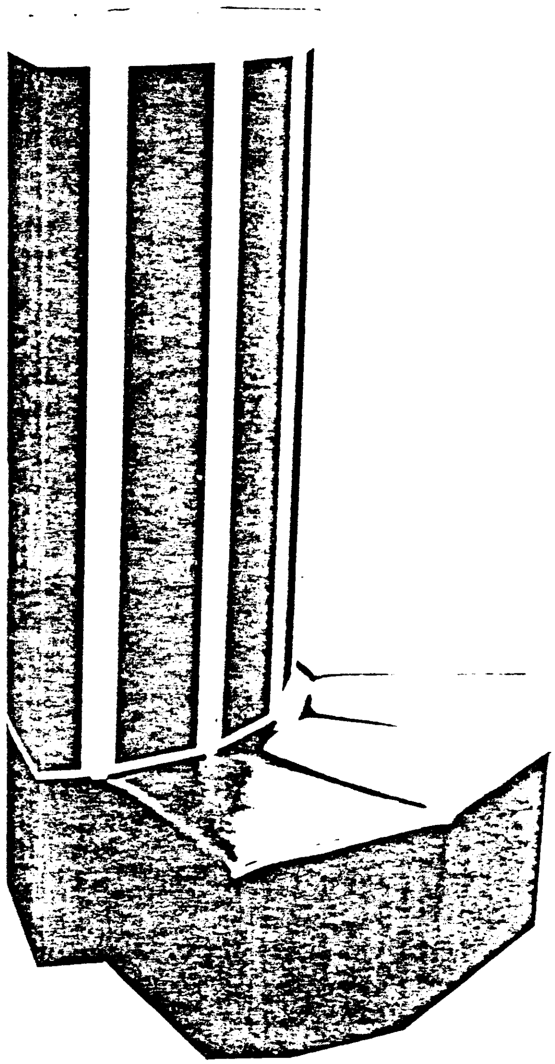

FIGURE 1-1. I/O SUBSYSTEM

INDIVIDUAL I/O PROCESSOR

LOCAL MEMORY:

65,536 WORDS
16 BITS/WORD
2.43 GIGABIT BAND WIDTH

COMPUTATION SECTION:

INSTRUCTION CONTROL NETWORK
2 FUNCTIONAL UNITS (ADDER AND SHIFTER)
LOGICAL 'AND' OPERATION
512 OPERAND REGISTERS
1 PROGRAMMER-VISIBLE ACCUMULATOR
SINGLE ADDRESS MODE
76 MEGAHZ CYCLE TIME

I/O SECTION:

6 DIRECT MEMORY ACCESS (DMA) PORTS
810 MEGABIT/S MAXIMUM TRANSFER RATE PER PORT
SEVERAL CHANNELS MAY MULTIPLEX INTO ONE PORT

MODEL NUMBERING CONVENTION

CRAY-1 S/1200 THROUGH /4400 CONTAIN AN I/O SUBSYSTEM

FIRST DIGIT INDICATES SIZE OF CENTRAL MEMORY IN MEGAWORDS

SECOND DIGIT INDICATES NUMBER OF I/O PROCESSORS

EXAMPLE:

CRAY-1 S/2300 HAS 2 MILLION WORDS
OF CENTRAL MEMORY AND 3 I/O PROCESSORS.

CRAY-1 S/x300

THREE PROCESSOR SYSTEM

1. MASTER I/O PROCESSOR (MIOP)

   CONTROLS FRONT END-INTERFACES.

   HAS THREE DISPLAY CONSOLES.

   HAS AN EXPANDER CHANNEL WHICH MULTIPLEXES A PRINTER AND A MAG TAPE UNIT.

   CONNECTS TO BUFFER MEMORY THROUGH A DMA CHANNEL.

   EXCHANGES CONTROL SIGNALS WITH CPU OVER A LOW-SPEED CRAY-1 S CHANNEL PAIR.

   COMMUNICATES WITH OTHER IOPs OVER ACCUMULATOR CHANNELS.

   MAINTAINS SYSTEM INFORMATION ERROR LOG.

   COORDINATES ACTIONS OF CPU AND OTHER IOPs.

2. BUFFER I/O PROCESSOR (BIOP)

   HANDLES DATA TRANSFERS BETWEEN CPU AND I/O SUBSYSTEM

   CONNECTS DIRECTLY TO CENTRAL MEMORY.

   CONNECTS TO BUFFER MEMORY THROUGH A DMA CHANNEL.

   COMMUNICATES WITH OTHER IOPs OVER ACCUMULATOR CHANNELS.

   DRIVES UP TO 16 DD-29 DISK DRIVES.

3. BLOCK MULTIPLEXER I/O PROCESSOR (XIOP)

CONNECTS TO BUFFER MEMORY THROUGH A DMA CHANNEL.

COMMUNICATES WITH OTHER IOPs OVER ACCUMULATOR CHANNELS.

HANDLES 1 TO 16 BLOCK MUX (IBM COMPATIBLE) CHANNELS.

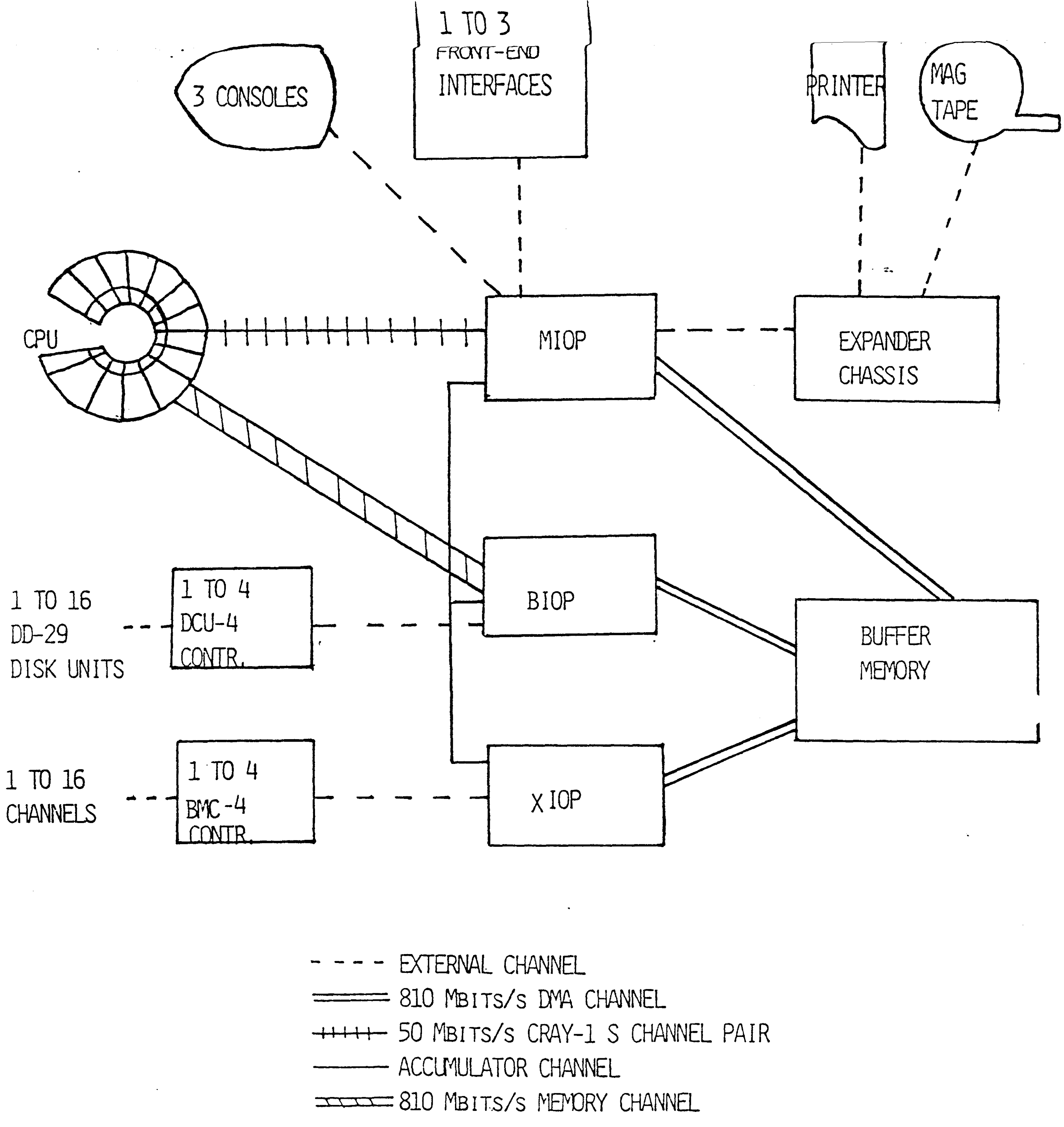

FIGURE 1-2 BLOCK DIAGRAM OF S/1300, S/2300 AND S/4300 SYSTEMS WITH BLOCK MULTIPLEXER CHANNELS.

CHAPTER 2

OPERATING SYSTEM OVERVIEW

FUNCTIONS

PERFORMS I/O BETWEEN CPU AND PERIPHERALS

MANAGES FRONT-END COMMUNICATIONS

PERFORMS STATION FUNCTIONS

DEADSTARTS CPU

CHARACTERISTICS

MULTI-TASKING

UP TO 32 TASKS ACTIVE AT A TIME

NONPRE-EMPTIVE SCHEDULING

SIMPLE 16 LEVEL TASK PRIORITY SCHEME

INTERRUPT DRIVEN

EXTENSIVE USE OF OVERLAYS

SYSTEM COMPONENTS

KERNEL

- NUCLEUS OF OPERATING SYSTEM.
- LOCAL MEMORY RESIDENT.
- EXECUTES IN EACH I/O PROCESSOR WITH MINOR MODIFICATIONS.
- RESPONSIBLE FOR:
  - ACTIVITY MANAGEMENT
  - INTER-ACTIVITY COMMUNICATION
  - RESOURCE MANAGEMENT
  - INTERRUPT HANDLING
  - INTER-PROCESSOR COMMUNICATION

OVERLAYS

- RESIDE IN BUFFER MEMORY.
- READ INTO LOCAL MEMORY WHEN NEEDED.
- MAKE UP THE BULK OF THE SYSTEM.
- NOT ALL USED BY ANY ONE PROCESSOR.

- DISK SUBSYSTEM
  - RESIDES MOSTLY IN BUFFER MEMORY AS OVERLAYS.
  - EXECUTES IN BIOP OR DIOP.

- STATION SUBSYSTEM
  - RESIDES IN BUFFER MEMORY AS OVERLAYS.
  - EXECUTES MOSTLY IN MIOP.

- CONCENTRATOR SUBSYSTEM
  - RESIDES IN BUFFER MEMORY AS OVERLAYS.
  - EXECUTES MOSTLY IN MIOP.

- INTERACTIVE CONCENTRATOR SUBSYSTEM
  - RESIDES IN BUFFER MEMORY AS OVERLAYS.
  - EXECUTES MOSTLY IN MIOP.

- TAPE SUBSYSTEM
  - RESIDES IN BUFFER MEMORY AS OVERLAYS.
  - EXECUTES IN XIOP.

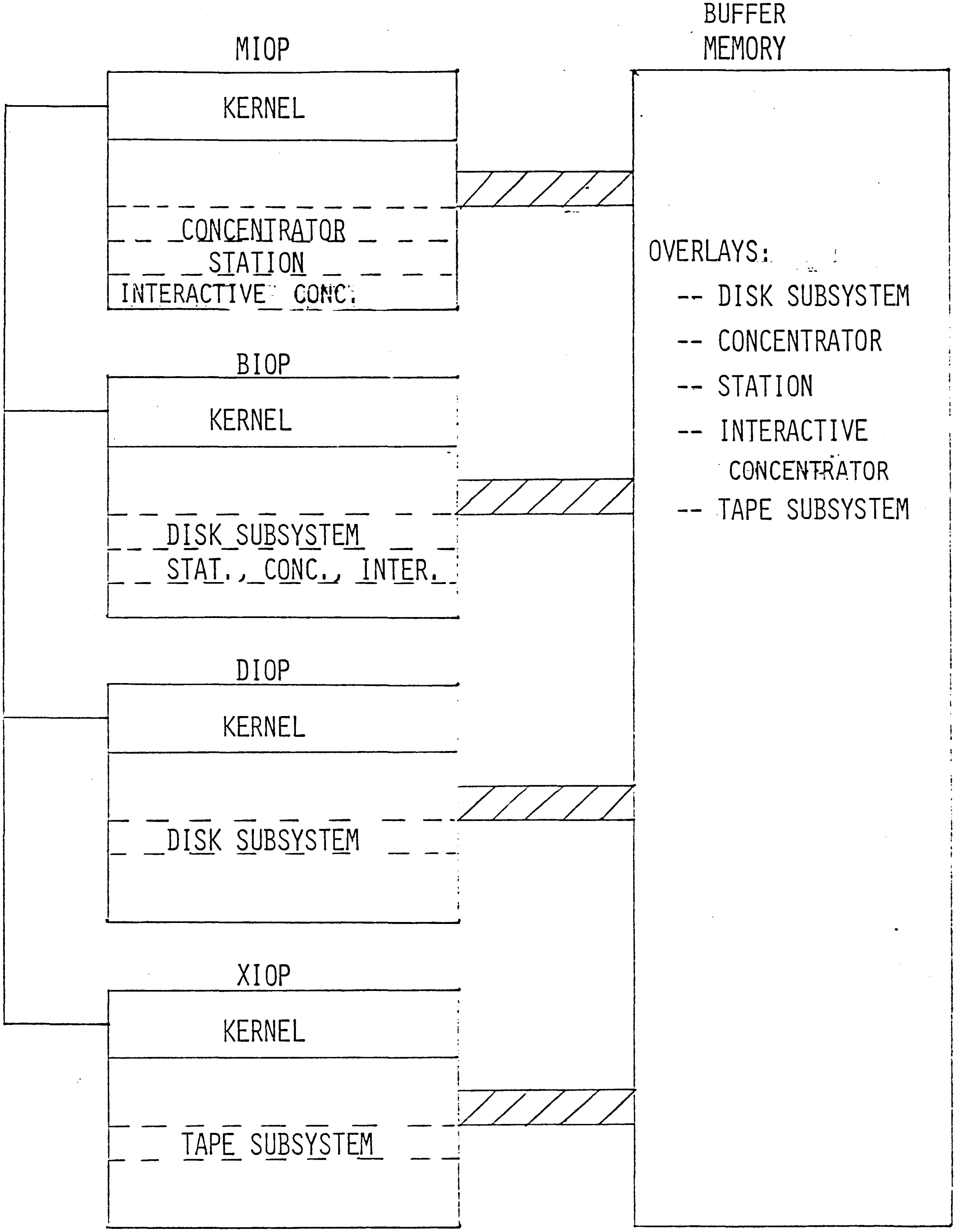

FIGURE 2-1 IOS SOFTWARE CONFIGURATION

KERNEL CONSOLE CALLABLE OVERLAYS

CERTAIN OVERLAY ACTIVITIES MAY BE CREATED BY KEYING IN OVERLAY NAME AT KERNEL CONSOLE.

KERNEL CREATES ACTIVITY FOR THE OVERLAY AND PUTS IT ON IOP CENTRAL PROCESSOR QUEUE.

ACTIVITY THEN PROCEEDS AS ANY OTHER ACTIVITY.

OPERATOR MAY USE THIS FACILITY TO:

DEADSTART CPU
BRING UP THE STATION
START A CONCENTRATOR
ENTER THE INTERACTIVE CONCENTRATOR
RUN TEST ROUTINES

IOS STATION

FUNCTIONS

PROVIDES A MEANS FOR OPERATOR-CPU COMMUNICATION.

CONTROLS OPERATOR CONSOLES

MAY BE USED AS A BATCH JOB ENTRY STATION

JOBS OR DATASETS STAGED FROM TAPE

MAY ACCEPT CPU OUTPUT AND DISTRIBUTE IT TO MAG TAPE OR PRINTER

ALLOWS ON-LINE DEBUGGING OF CPU

CHARACTERISTICS

RESIDES IN BUFFER MEMORY AS OVERLAYS.

EXECUTES MOSTLY IN MIOP WITH SOME HIGH SPEED TRANSFERS THROUGH BIOP.

MAY HAVE MORE THAN ONE STATION ACTIVE AT A TIME.

- EACH STATION MUST HAVE A DEDICATED CONSOLE.
- THEY MUST SHARE THE EXPANDER PERIPHERALS.
- TWO OR MORE CONSOLES MAY BE SUPPORTED BY ONE STATION.

COMMUNICATES WITH CPU IN STANDARD CRAY MESSAGE FORMAT.

- APPEARS TO BE JUST ANOTHER FRONT-END STATION TO CPU.

OPERATOR COMMANDS VIRTUALLY IDENTICAL TO THOSE OF THE PRESENT DATA GENERAL ECLIPSE STATION.

OVERVIEW OF STATION MESSAGE FLOW

1. OPERATOR INITIATES A STATION COMMAND.

2. MIOP BUILDS A MESSAGE.

3. MIOP WRITES THIS MESSAGE TO BUFFER MEMORY.

4. BIOP SENDS THE MESSAGE TO CENTRAL MEMORY.

5. MIOP TELLS CPU MESSAGE IS IN CENTRAL MEMORY.

6. CPU PROCESSES MESSAGE AND BUILDS A RESPONSE.

7. BIOP READS RESPONSE MESSAGE AND PUTS IT IN BUFFER MEMORY.

8. MIOP READS RESPONSE MESSAGE IN FROM BUFFER MEMORY.

9. IF APPROPRIATE, CONSOLE IS UPDATED WITH NEW INFORMATION.

CONCENTRATOR

FUNCTIONS

ALLOWS APPARENT DIRECT COMMUNICATION BETWEEN THE CPU AND A FRONT-END.

> LOOKS LIKE A CRAY-1 S CHANNEL PAIR TO FRONT-END.
>
> > THUS NO CHANGES NECESSARY TO EXISTING FRONT-END STATIONS.

MAY REDUCE THE NUMBER OF INTERRUPTS TO THE CPU PER FRONT-END MESSAGE.

WILL REDUCE CENTRAL MEMORY MESSAGE SEGMENT BUFFERS WHEN BACK DOOR TO DISK IS AVAILABLE.

CHARACTERISTICS

RESIDES IN BUFFER MEMORY AS OVERLAYS.

EXECUTES MOSTLY IN MIOP WITH HIGH SPEED TRANSFERS TO CPU THROUGH BIOP.

ONE ACTIVE CONCENTRATOR FOR EACH FRONT-END CHANNEL PAIR.

> MAY HAVE SEVERAL LOGICAL ID'S LOGGED ON TO ONE CONCENTRATOR.
>
> > EACH ID MAY HAVE A DIFFERENT SEGMENT SIZE.

CONTROLLED VIA CONC AND ENDCONC KERNEL CONSOLE COMMANDS.

CHAPTER 3

KERNEL CONSOLE COMMANDS

THE FOLLOWING OPERATOR COMMANDS PERFORM NON-STATION FUNCTIONS.

THESE ARE ENTERED AT THE KERNEL CONSOLE OF AN IOP.

CRAY

STATION

CONC

ENDCONC

START

CONFIG

TRACE

HPLOAD

CRAY

INITIALIZES THE INTERFACE BETWEEN THE IOS AND THE CPU.

THIS IS NECESSARY BEFORE MESSAGES CAN BE EXCHANGED.

PERFORMED IMPLICITLY BY CONC AND START COMMANDS.

STATION [,num]

INITIALIZES THE IOS STATION AT CONSOLE num. num=0,1,2,3

IF num NOT SPECIFIED, A DEFAULT CONSOLE IS USED.

CONC [,ch]

INITIALIZES A CONCENTRATOR FOR THE FRONT-END CHANNEL PAIR IDENTIFIED BY ch.ch=0,1,2.

IF ch NOT SPECIFIED, 0 IS USED.

ENDCONC [,ch]

TERMINATES CONCENTRATOR ch.

IF ch NOT SPECIFIED, 0 IS USED.

START $\left\{\begin{array}{l}\text{MT}\phi\text{:n,}\left[\text{, SV/sysdsn}\right]\\ \text{sysdsn}\end{array}\right\}$ $\left\{\begin{array}{l}\text{MT}\phi\text{:n}_2\left[\text{, SV/pardsn}\right]\\ \text{pardsn}\end{array}\right\}$

DEADSTARTS THE CPU ACCORDING TO SPECIFIED COS BINARY FILE (sysdsn) AND PARAMETER FILE (pardsn).

IF DEADSTART IS FROM TAPE, THE FILES MAY BE SAVED IN THE COS AND PAR DIRECTORIES ON DISK.

REFER TO CHAPTER 5 FOR A DETAILED DISCUSSION OF DEADSTART.

CONFIG

DISPLAYS RESOURCES AND CHANNEL CONFIGURATION INFORMATION AT THE KERNEL CONSOLE.

MAY BE DONE AT ANY IOP KERNEL CONSOLE.

TRACE command

ALLOWS OPERATOR TO SELECTIVELY ENABLE/DISABLE SELECTED EVENTS, AND TO PRINT OUT LOCAL AND BUFFER MEMORY TRACE BUFFERS.

MAY BE DONE AT ANY IOP KERNEL CONSOLE.

REFER TO CHAPTER 6 FOR A DETAILED DISCUSSION OF THE COMMANDS.

HPLOAD fn

LOADS HEADER PAGE DATA FROM TAPE FILE NUMBER fn.

CHAPTER 4

IOS STATION COMMANDS

STATION COMMANDS ARE ENTERED AT A DEDICATED STATION CONSOLE.

TERMINATE EACH COMMAND WITH THE RETURN KEY.

THE STATION PROCESSES EACH COMMAND AS IT IS ENTERED AND INDICATES COMPLETION BY DISPLAYING THE > PROMPT CHARACTER.

NORMALLY, STATION COMMANDS ARE DISPLAYED ON THE LOWER THREE LINES OF THE DISPLAY.

ALL BUT THREE OF THE REMAINING LINES ARE USED TO DISPLAY REQUESTED INFORMATION.

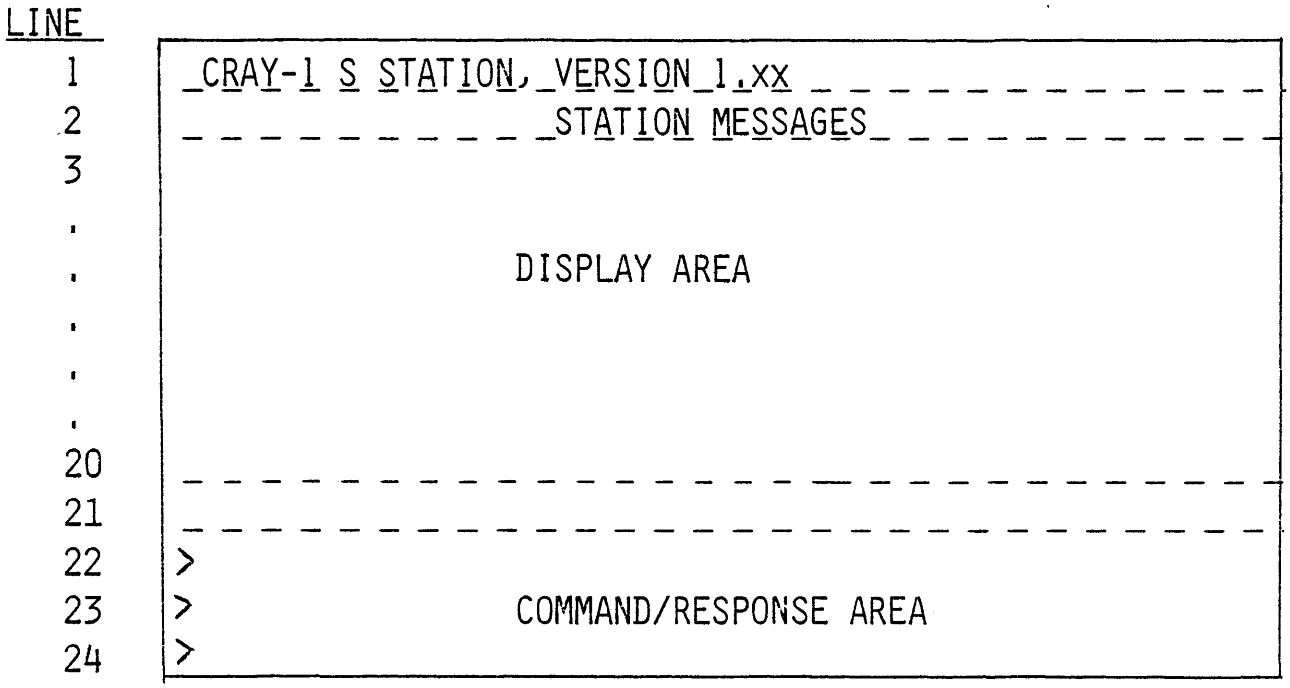

FIGURE 4-1. STATION DISPLAY FORMAT

KEY	FUNCTION
RETURN	INITIATES THE COMMAND.
RUBOUT	BACKSPACES AND ERASES ONE CHARACTER ON THE LINE.
CONTROL-U	DELETES THE ENTIRE LINE.
+ OR >	ROLLS DISPLAY FORWARD ONE FRAME IF ENTERED AS FIRST CHARACTER OF A LINE.
- OR <	ROLLS DISPLAY BACKWARD ONE FRAME IF ENTERED AS FIRST CHARACTER OF A LINE.
ESC	DISCARDS UNPROCESSED KEYBOARD INPUT AND REFRESHES THE ENTIRE SCREEN.
CONTROL-A	RECALLS PREVIOUS COMMAND, EDITED BY ANY SUBSEQUENT KEYBOARD INPUT, AND POSITIONS THE CURSOR AT THE END OF THE COMMAND.
CONTROL-X	EQUIVALENT TO CONTROL-A AND A RETURN.

TABLE 4-1. IOS STATION SPECIAL KEYS

STATION COMMANDS CAN BE CLASSIFIED AS THREE BASIC TYPES:

BATCH ENTRY COMMANDS
OPERATOR STATION COMMANDS
DISPLAY AND STATUS RESPONSE COMMANDS

Command	Function
Station software activation	
LOGON	Establishes communications between a I/O Subsystem station and the CRAY-1 CPU
LOGOFF	Terminates communications between a I/O Subsystem station and the CRAY-1 CPU
Staging control	
STAGE	Halts or resumes staging of datasets between CRAY-1 CPU and the I/O Subsystem Station.
SUBMIT	Queues file for staging to CRAY-1 mass storage where dataset will be entered into job input queue
SAVE	Queues a file to CRAY-1 mass storage where it will be made a permanent dataset
Miscellaneous control	
SNAP	Copies display screen image to a line printer
END	Terminates console operation
DELAY	Suspends processing of a command for a specified time interval
CONSOLE	Adds another MIOP console to the Station software
POLL	Sets the rate at which control messages are exchanged with the CRAY-1 CPU
SET	Modifies the default value associated with an operator station's ID and TID parameters

TABLE 4-2. BATCH ENTRY COMMAND SUMMARY

Command	Function
Job control	
LIMIT	Sets maximum number of jobs that CRAY-1 CPU can process at one time
DROP	Ends processing of job at CRAY-1 CPU immediately but does not delete output datasets
KILL	Depending upon the job's status, either deletes job's input dataset from input queue or deletes output dataset
RERUN	Ends processing of job at CRAY-1 CPU immediately and attempts to rerun the job
SUSPEND	Suspends processing of one or all jobs
RESUME	Reschedules processing of jobs suspended by SUSPEND or SHUTDOWN commands
SHUTDOWN	Idles down job activity in preparation for a system interruption
RECOVER	Lifts suspension from jobs suspended by a SHUTDOWN or system interruption
ENTER	Assigns a new time limit or priority to a job (or its output dataset); changes ID for the station at which the job originated or to which its output dataset is to be sent, or changes its class assignment if it is in the input queue
CLASS	Turns job class or classes on or off
MESSAGE	Enters message into a job logfile, the system logfile, or both
SWITCH	Sets or clears a job sense switch
Link Control	
ROUTE	Changes the station ID for all jobs and output datasets having a specific ID to a new ID
OPERATOR	Changes the ID of the System Operator Station
STREAM	Changes the input, output, and active stream counts for the specified operator station

TABLE 4-3. OPERATOR STATION COMMAND SUMMARY

Command	Function
Channel control	
CHANNEL	Turns CPU I/O channel on or off
Mass storage control	
DEVICE	Sets or clears read-only mode for CRAY-1 mass storage device

TABLE 4-3. OPERATOR STATION COMMAND SUMMARY (CONTINUED)

Command	Function
CLEAR	Clears display area of screen
COMMENT	Inserts comment in command stream
SCROLL	Causes entire display area to be used as command/response area
REFRESH	Sets display refresh rate
JOB	Displays status of a specific job
STATUS	Displays status of all jobs in job input queue, all datasets in output staging queue, and all executing jobs
DATASET	Responds with status of specific dataset
LINK	Displays status of station link
STATION	Displays status of station
STORAGE	Displays status of mass storage devices
ERROR	Displays information from the Error Log Table when errors are sensed

TABLE 4-4. DISPLAY AND STATUS RESPONSE COMMAND SUMMARY

CHAPTER 5

DEADSTART

OVERVIEW

IOS IS INITIALLY DEADSTARTED FROM TAPE.

SUBSEQUENT RESTARTS MAY BE FROM DISK.

THE CPU IS INITIALLY DEADSTARTED BY THE IOS FROM TAPE.

SUBSEQUENT DEADSTARTS AND RESTARTS MAY BE FROM DISK.

I/O SUBSYSTEM DEADSTART

TAPE DEADSTART:

THE CONVENTIONAL IOS TAPE LAYOUT IS:

FILE 0 - TAPELOAD
FILE 1 - DMP
FILE 2 - KERNEL
FILE 3 - OVERLAYS

PROCEDURE:

1. MOUNT THE IOS DEADSTART TAPE ON THE IOS TAPE UNIT

2. PUSH MASTER CLEAR AND DEADSTART BUTTONS AT THE POWER UNIT.

3. TYPE "2" IN RESPONSE TO THE TAPELOAD "FROM MTO:" MESSAGE AT THE MIOP KERNEL CONSOLE.

4. IF THE KERNEL WAS ASSEMBLED WITH THE ON-LINE DEBUGGER, TYPE "X" WHEN THE ! PROMPT CHARACTER APPEARS.

5. WHEN DEADSTART IS COMPLETE, A SYSTEM MESSAGE WILL BE POSTED AT EACH KERNEL CONSOLE.

6. DEADSTART THE CPU, IF APPROPRIATE.

DISK DEADSTART:

UNDER CERTAIN CONDITIONS, THE IOS MAY BE RESTARTED FROM A FILE IN THE IOS DIRECTORY ON DISK.

PREREQUISITE:

A FILE, ios, HAS PREVIOUSLY BEEN SAVED WITH THE COPY FILE UTILITY.

PROCEDURE:

1. TYPE CNTRL-D AT THE MIOP KERNEL CONSOLE. IF "SYSDUMP?" APPEARS, GO TO 5.

2. IF NO RESPONSE, MAKE SURE THERE IS NO TAPE LOADED ON THE TAPE DRIVE AND PUSH MASTER CLEAR AND DEADSTART AT THE POWER UNIT.

3. IF 2 RESULTS IN ENTERING THE DEBUGGER, TYPE CNTRL-D TO EXIT.

4. TYPE CNTRL-D AGAIN. IF "SYSDUMP?" DOES NOT APPEAR, A TAPE DEADSTART MUST BE PERFORMED.

5. TYPE "Y" OR "N" IN RESPONSE TO "SYSDUMP".

6. WHEN DUMP COMPLETE (OR IMMEDIATELY),"RESTART?" WILL BE POSTED. TYPE "Y".

7. ENTER ios IN RESPONSE TO "ENTER RESTART FILE NAME:" MESSAGE

8. IF AN ERROR OCCURS, IT MAY BE NECESSARY TO DEADSTART FROM TAPE.

MESSAGE	MEANING
DISK ERROR	AN UNRECOVERABLE DISK ERROR OCCURRED.
LABEL NOT FOUND	MASTER DEVICE LABEL COULD NOT BE FOUND.
DIRECTORY NOT FOUND	THE IOS DIRECTORY COULD NOT BE FOUND.
FILE NOT FOUND	THE NAMED FILE COULD NOT BE FOUND IN IOS DIRECTORY.
MOS ERROR	AN UNRECOVERABLE ERROR OCCURRED WHILE READING BUFFER MEMORY.
RETRY?	DISPLAYED AFTER ERROR MESSAGES. ENTER "Y" IF ANOTHER TRY AT RESTART IS DESIRED. A NEW PROMPT FOR FILE NAME WILL ALSO BE DISPLAYED.

TABLE 5-1. IOS DISK RESTART ERROR MESSAGES

CPU DEADSTART

CPU DEADSTART REQUIRES A COS BINARY FILE AND A PARAMETER FILE.

EITHER OF THESE CAN RESIDE ON TAPE OR DISK.

THE PARAMETER FILE MAY ALSO BE INPUT FROM
THE CONSOLE; OR AN EXISTING ONE MAY BE
EDITED THROUGH THE CONSOLE.

THE FORMAT OF THE START COMMAND, INPUT AT THE MIOP KERNEL CONSOLE IS:

START cosfile parfile[,ED]

WHERE cosfile IS:
MTO:n_1 [,SV/sysdsn]
n_1 IS TAPE FILE NUMBER.
sysdsn IS DESIRED NAME OF SAVED FILE.

sysdsn - NAME OF FILE IN COS DIRECTORY
ON DISK.

parfile IS:
MTO:n_2 [, SV/pardsn]
n_2 IS TAPE FILE NUMBER.
pardsn IS DESIRED NAME OF SAVED FILE

pardsn - NAME OF FILE IN PAR DIRECTORY
ON DISK.

TTI - PARAMETER FILE IS INPUT FROM CONSOLE

ED INDICATES PARAMETER FILE IS TO BE EDITED FIRST.

START EXAMPLES

```
START MTO: 0    MTO: 3
```

- COS BINARY ON TAPE FILE 0; PARAMETER FILE ON TAPE FILE 3.

```
START MTO:0, SV/COS1     MTO:2, SV/PAR1
```

- STARTUP FROM TAPE FILES 0 AND 2; SAVE COS BINARY FILE IN COS DIRECTORY AS COS1; SAVE PARAMETER FILE IN PAR DIRECTORY AS PAR1.

```
START COS1  PAR1, ED
```

- STARTUP FROM DISK FILE COS1 WITH PARAMETER FILE PAR1 BEING EDITED FIRST.

```
START MTO:2  TTI
```

- STARTUP FROM TAPE FILE 2 WITH PARAMETER FILE ENTERED AT CONSOLE.

DEADSTART DISK FILES

THREE DISK DIRECTORIES ARE SET ASIDE BY COS AT INSTALL TIME FOR DEADSTART FILES.

THE DIRECTORIES ARE:

COS — USED TO STORE COS BINARY FILES. THE FILES ARE CREATED, NAMED AND SAVED USING THE SV OPTION ON THE START COMMAND OR THE COPY UTILITY.

PAR — USED TO STORE PARAMETER TEXT FILES. THESE ARE CREATED USING THE SV OPTION ON THE START COMMAND; THE COPY UTILITY; AND THE PARAMETER FILE EDITOR.

IOS — USED TO STORE IOS BINARY FILES. THESE ARE CREATED USING THE COPY UTILITY.

THE NAMES OF FILES RESIDING IN THESE DIRECTORIES MUST BE 15 OR LESS ASCII CHARACTERS.

THEY CANNOT BEGIN WITH MT OR TT.

FILE UTILITIES

THERE ARE SEVEN UTILITIES AVAILABLE FOR MANIPULATING FILES IN THE COS, PAR AND IOS DIRECTORIES.

1. EDIT fn

   INVOKES THE PARAMETER FILE EDITOR.

   fn MAY BE THE NAME OF A FILE ALREADY IN THE PAR DIRECTORY; OR TTI, IF A NEW FILE IS TO BE CREATED.

2. COPY fn_1 fn_2

   COPY FILE fn_1 TO FILE fn_2. THE COPY IS EITHER FROM TAPE TO DISK OR DISK TO TAPE. IF COPY IS FROM TAPE TO DISK, fn_2 CANNOT ALREADY BE IN USE IN THE SPECIFIED DIRECTORY.

   DISK FILES ARE DENOTED AS 'DIR'/fn, WHERE 'DIR' IS COS, PAR OR IOS.

   WHEN COPYING TO THE IOS DIRECTORY, THE OVERLAY FILE MUST IMMEDIATELY FOLLOW THE KERNEL FILE. WHEN COPYING THE OTHER WAY, ALLOW TWO CONSECUTIVE TAPE FILES.

3. FSTAT 'DIR' $\left[\text{/}fn_1\text{,...}\right]$

   DISPLAY FILE STATUS (CREATED, WORD LENGTH) OF ONE OR MORE FILES WITHIN THE SPECIFIED DIRECTORY.

   IF NO FILE NAMES SPECIFIED, THEN STATUS OF ALL FILES IN THE DIRECTORY WILL BE DISPLAYED.

4. PURGE 'DIR'/fn_1,...

DELETE THE SPECIFIED FILES FROM THE SPECIFIED DIRECTORY.

5. CLEAR 'DIR'

DELETE ALL FILES FROM THE NAMED DIRECTORY.

6. FDUMP MTO:Y 'DIR' [/fn_1,...]

EXECUTE A FORMATTED DUMP OF THE SPECIFIED FILES TO TAPE FILE Y.

IF NO FILE NAMES SPECIFIED, ALL FILES IN THE DIRECTORY WILL BE DUMPED.

7. FLOAD MTO:Y [fn_1, fn_2,...]

LOAD A PREVIOUSLY 'FDUMPED' TAPE FILE INTO THE ORIGINAL DIRECTORY.

IF NO FILE NAMES SPECIFIED, ALL FILES ON THE TAPE WILL BE LOADED.

IF A FILE ALREADY EXISTS IN THE DIRECTORY, THE FILE ON TAPE WILL NOT BE LOADED.

FDUMP AND FLOAD ARE USEFUL WHEN A DIRECTORY GETS FRAGMENTED.

PARAMETER FILE EDITOR

PROVIDES FOR CREATION AND MODIFICATION OF PARAMETER TEXT FILES REQUIRED FOR CPU DEADSTART.

THE EDITOR IS RUN THROUGH THE MIOP KERNEL CONSOLE.

EACH OF THE FOLLOWING WILL INVOKE THE EDITOR:

1. ED OPTION ON THE START COMMAND.

2. SPECIFYING TTI FOR parfile ON THE START COMMAND.

3. EDIT fn.

THE EDITOR OPERATES IN TWO MODES:

1. COMMAND INPUT MODE.

   THIS MODE IS RECOGNIZED BY A '>' IN COLUMN 1.

2. TEXT INPUT MODE.

   INDICATED BY A LINE NUMBER IN COLUMN 1

   INPUT IS ACCEPTED ON A LINE-BY-LINE BASIS.

   TERMINATE LINES BY CARRIAGE RETURNS OR LINE FEEDS.

   THE ESC KEY RETURNS CONTROL TO COMMAND INPUT MODE.

EDITOR COMMANDS

THERE ARE SEVEN COMMANDS AVAILABLE FOR EDITING PARAMETER TEXT FILES.

1. INSERT ln

   INSERT TEXT FOLLOWING THE SPECIFIED LINE NUMBER.

2. APPEND

   APPEND TEXT TO THE FILE.

   IF FILE IS EMPTY, TEXT WILL BE ACCEPTED STARTING AT LINE 1.

3. DELETE ln_1 ln_2

   DELETE LINES ln_1 TO ln_2 INCLUSIVE.

4. REPLACE ln_1 ln_2

   REPLACE LINES ln_1 TC ln_2 , INCLUSIVE, WITH TEXT TO BE INPUT.

5. TYPE ln_1 ln_2

   TYPE LINES ln_1 TO ln_2, INCLUSIVE, TO THE CONSOLE.

6. PRINT ln_1 ln_2

PRINT LINES ln_1 TO ln_2, INCLUSIVE, ON THE PRINTER.

7. BYE

TERMINATE THE EDITOR.

THE FOLLOWING MESSAGE IS DISPLAYED:

"SAVE?"

NO - EDITED VERSION IS DISCARDED. IF EDITOR WAS CALLED FROM START, EDITED VERSION WILL BE SENT TO CPU BUT NOT MADE PERMANENT.

YES- "ENTER FILE NAME:" MESSAGE IS DISPLAYED. EDITED VERSION OF THE FILE WILL BE SAVED IN THE PAR DIRECTORY UNDER THE SPECIFIED NAME.

MESSAGE	MEANING
COMMAND SYNTAX ERROR	THE COMMAND ENTERED WAS NOT IN LEGAL FORMAT
EXPANDER DEVICE ERROR	AN ERROR WAS ENCOUNTERED ON THE EXPANDER DEVICE BEING USED.
MOS NOT AVAILABLE	BUFFER MEMORY SPACE NOT AVAILABLE.
LOCAL MEMORY NOT AVAILABLE	LOCAL MEMORY NOT AVAILABLE.
DISK ERROR	AN UNRECOVERABLE DISK ERROR OCCURRED.
FILE NOT FOUND:'NAME'	THE SPECIFIED FILE COULD NOT BE FOUND IN THE CURRENT DIRECTORY.
LABEL NOT FOUND	THE LABEL ON THE MASTER DEVICE COULD NOT BE FOUND.
FILE DIRECTORY FULL	NO MORE ROOM IN THE CURRENT DIRECTORY FOR NEW FILES.
FILE BUFFERS DEPLETED	NOT ENOUGH DISK SPACE REMAINS IN THE CURRENT DIRECTORY TO LOAD THE FILE.
FILE DELETED: 'NAME'	FILE NAMED WAS DELETED FROM THE CURRENT DIRECTORY.
FILE CREATED:'NAME'	FILE NAME WAS CREATED IN THE CURRENT DIRECTOR
FILE ALREADY EXISTS: 'NAME'	THE NAMED FILE ALREADY EXISTS. IT MUST BE PURGED BEFORE IT CAN BE RE-CREATED.
FILE BEING UPDATED:'NAME'	NAMED FILE IS BEING WRITTEN OVER.
FILE DUMPED:'NAME'	NAMED FILE HAS BEEN DUMPED TO THE TAPE FILE.
FILE LOADED:'NAME'	NAMED FILE HAS BEEN LOADED FROM THE DUMP TAPE AND CREATED IN THE CURRENT DIRECTORY.

TABLE 5-2. START COMMAND AND FILE UTILITY MESSAGES

PARAMETER FILES

A PARAMETER FILE IS A TEXT FILE USED BY COS AT STARTUP TIME TO DETERMINE THE STARTUP MODE, DEVICE FLAWS AND STATUS, AND OTHER PERTINENT INFORMATION.

A PARAMETER FILE CONSISTS OF A SET OF DIRECTIVES, EACH OF WHICH CONSISTS OF An *, A KEYWORD, AND ANY NECESSARY ARGUMENTS.

> ARGUMENTS AND THE KEYWORD ARE SEPARATED BY COMMAS.
>
> EXCEPT WHEN ENTERING ASCII DATA INTO MEMORY, A BLANK INDICATES THE START OF THE COMMENT FIELD.

THERE ARE DIRECTIVES FOR:

- SELECTING A STARTUP MODE
- ENTERING MEMORY
- SETTING STARTUP BREAKPOINTS
- SETTING CENTRAL MEMORY SIZE
- CONTROL DEVICE ALLOCATION AND AVAILABILITY
- SPECIFYING DISK FLAWS
- CONTROLLING STARTUP DUMP PROCESSING
- RECOVERING ROLLED JOBS AFTER A RESTART
- CONTROLLING PERMANENT DATASET AND SYSTEM DIRECTORY RECOVERY
- INVOKING A NEW JOB CLASS STRUCTURE
- STARTING A SYSTEM FROM A CPU LOCAL DISK

PARAMETER FILE DIRECTIVES

ONLY THOSE DIRECTIVES MOST LIKELY TO BE USED BY AN OPERATOR WILL BE DISCUSSED HERE.

STARTUP MODE DIRECTIVES:

THERE ARE THREE MUTUALLY EXCLUSIVE STARTUP MODE DIRECTIVES.

THE LAST ONE IN THE FILE IS USED.

*INSTALL

STARTS COS AS FOR THE FIRST TIME.

ALL MASS STORAGE IS ASSUMED TO BE VACANT.

THE DATASET CATALOG (DSC) AND DEVICE LABELS ARE INTIALIZED.

SHOULD BE USED ONLY AFTER A MAJOR SYSTEM CRASH.

*DEADSTART

STARTS COS AS IF AFTER A NORMAL SYSTEM TERMINATION.

*RESTART

STARTS COS AS IF AFTER A NORMAL SYSTEM INTERRUPTION.

ALLOWS RECOVERY OF JOBS RUNNING WHEN SYSTEM WAS INTERRUPTED.

* END DIRECTIVE

SIGNALS THE END OF THE PARAMETER FILE.

IT IS A REQUIRED DIRECTIVE.

* MEMSIZ, size DIRECTIVE

OVERIDES THE I@MEM INSTALLATION PARAMETER.

size IS IN OCTAL WORDS AND MUST NOT EXCEED I@MEM.

* RRJ, n DIRECTIVE

THIS DIRECTIVE ALLOWS RECOVERY OF JOBS AFTER A RESTART.

IF n=1, RECOVERY IS ATTEMPTED. IF n=0, NO RECOVERY IS ATTEMPTED.

* LOCK, n DIRECTIVE

TELLS COS HOW TO HANDLE JOBS WITH A NON-ZERO JTEPC FIELD, BEING RECOVERED IF NEW SYSTEM IS DIFFERENT FROM PREVIOUS ONE.

IF n=1, RECOVER JOB WHEN THE JOB'S ORIGINAL SYSTEM IS DEADSTARTED. (IT COULD BE ON THIS DEADSTART ATTEMPT.)

* J CLASS, pdn [,id [,ed]]

ALLOWS OPERATOR TO INVOKE A NEW JOB CLASS STRUCTURE AFTER A DEADSTART OR RESTART.

pdn - NAME OF PERMANENT DATASET CONTAINING THE JOB CLASS STRUCTURE DEFINITION.

id - USER ID OF THE PERMANENT DATASET

ed - EDITION NUMBER

DEVICE DIRECTIVES:

DEVICE ALLOCATION CONTROL

* ON, ldv_1, ldv_2, ..., ldv_n

DECLARES A DEVICE ALLOCATABLE

ldv_i - A LOGICAL DEVICE

* OFF, ldv_1, ldv_2,...,ldv_n

DECLARES A DEVICE NON-ALLOCATABLE

ldv_i - A LOGICAL DEVICE

IF BOTH *ON AND *OFF REFERENCE THE SAME DEVICE, THE LAST DIRECTIVE IS USED.

THE MASTER DEVICE MAY NOT BE REFERENCED BY *OFF.

DEVICE AVAILABILITY CONTROL

*DOWN, ldv_1, ldv_2, ...,ldv_n

DECLARES A DEVICE UNAVAILABLE.

THE MASTER DEVICE MAY NOT BE REFERENCED BY THIS DIRECTIVE.

ldv_i - A LOGICAL DEVICE

*RELEASE, ldv_1, ldv_2,...,ldv_n

REMOVES THE DATASET ENTRIES FOR THIS DEVICE FROM THE DATASET CATALOG.

THIS MAKES THEM INACCESSIBLE TO THIS SYSTEM.

THE MASTER DEVICE MAY NOT BE REFERENCED BY THIS DIRECTIVE.

ldv_i - A LOGICAL DEVICE.

*UP, ldv_1, ldv_2,...,ldv_n

DECLARES A PREVIOUSLY UNAVAILABLE DEVICE AVAILABLE.

THE DEVICE LABEL IS INITIALIZED.

A *UP AFTER A *DOWN TAKES PRECEDENCE BUT A *DOWN AFTER A *UP IS ILLEGAL.

THE MASTER DEVICE MAY NOT BE REFERENCED BY THIS DIRECTIVE.

ldv_i - A LOGICAL DEVICE

DISK FLAW DIRECTIVES:

TWO FLAW DIRECTIVES, *FLAW AND *ENDFLAW, DELIMIT A SET OF FLAW CARDS WHICH DECLARE VARIOUS TRACKS AND CYLINDERS OF THE SPECIFIED DISK UNAVAILABLE.

THE FORMAT IS:

```
*FLAW, ldv
 flaw cards
*ENDFLAW
```

ldv - LOGICAL DEVICE TO BE FLAWED.

flaw cards - MAY BE ONE OR MORE OF THE FOLLOWING:

Cnnn,	FLAW ALL OF CYLINDER nnn
Cnnn-mmm	FLAW CYLINDERS nnn TO mmm
Cnnn,Tmm	FLAW TRACK mm OF CYLINDER nnn
Cnnn,Tmm-pp	FLAW TRACKS mm THROUGH pp OF CYLINDER nnn.

THESE DIRECTIVES ARE NORMALLY USED TO ADD FLAWS ENCOUNTERED DURING SYSTEM OPERATION WITHOUT REQUIRING RE-ASSEMBLY OF THE SYSTEM.

EXAMPLE OF A DEADSTART PARAMETER FILE

```
*DEADSTART          SPECIFY DEADSTART MODE.
*RRJ,0              DON'T RECOVER ROLLED JOBS.
*LOCK,1             DON'T RECOVER JOBS WITH JTEPC SET IF NEW SYSTEM
*UP,DD-29-21        DIFFERENT.
*OFF,DD-29-21
*UP,DD-19-30
*ON,DD-19-30
*UP,DD-19-31
*ON,DD-19-31        DECLARE DEVICES DD-19-21,30,31,40 AVAILABLE AND
*UP,DD-19-40        ALLOCATABLE AND WRITE DEVICE LABELS.
*ON,DD-19-40
*DOWN,DD-19-41
*OFF,DD-19-41       DECLARE DD-19-41 UNAVAILABLE AND UNALLOCATABLE.
*FLAW,DD-19-20
*ENDFLW
*FLAW,DD-29-21      FLAW DD-29-21
C0,T0                 FLAW CYLINDER 0, TRACK 0
C1-77                 FLAW CYLINDER 1 TO 77
C260,T05
C356,T10
C525,T01
C726,T01
C1130,T10
C1272,T04
C1424,T05
*ENDFLW
*FLAW,DD-19-30      FLAW DD-19-30
C051,T03
C053,T03
C404,T00
C406,T02
C410,T00
C411,T10
C414,T03
C414,T06
C422,T01
C422,T02
C434,T02
C522,T04
C543,T01
*ENDFLW
*FLAW,DD-19-31      FLAW DD-19-31
C065,T04
C066,T04
C067,T04
C110,T04
C521,T00
*ENDFLW

*FLAW,DD-19-40      FLAW DD-19-40
C000,T00
C057,T04
*ENDFLW
*FLAW,DD-19-41      FLAW DD -19-41
C110,T04
*ENDFLW
*END                END OF PARAMETER FILE
```

SYSTEM STARTUP/SHUTDOWN CONSIDERATIONS

WHEN INITIALLY STARTING COS, MAKE SURE ALL FRONT-ENDS ARE POWERED UP.

IF THE DISKS WILL BE RUN THROUGH THE IOS, THE DD-19 INTERLOCK SWITCH ON THE POWER UNIT MUST BE ON.

WHEN SHUTTING DOWN THE SYSTEM, USE THE STATION SHUTDOWN COMMAND TO SAVE ANY JOBS WHICH MIGHT BE EXECUTING.

WHEN RECOVERING A SYSTEM AFTER A NORMAL SHUTDOWN, USE THE STATION RECOVER COMMAND TO START EXECUTING INTERRUPTED JOBS WHERE THEY LEFT OFF.

SUMMARY OF DEADSTART PROCEDURE

1. MAKE SURE CPU AND FRONT-ENDS ARE PROPERLY POWERED UP.

2. DEADSTART THE I/O SUBSYSTEM EITHER FROM TAPE OR DISK.

3. USE THE START COMMAND TO DEADSTART THE CPU FROM TAPE OR DISK, MAKING CHANGES TO THE PARAMETER FILE WITH THE EDITOR, IF NECESSARY.

4. RECOVER ANY JOBS, IF NECESSARY.

5 IF SYSTEM WILL NOT START, CHECK ALL HARDWARE FOR FAILURE. (DISK FAULT)

6. TRY TO START SYSTEM AGAIN.

7. IF THIS DOES NOT WORK, TAKE A DUMP AND TRY AGAIN.

8. IF IT STILL FAILS, GET A SYSTEM ANALYST.

CHAPTER 6

DUMP UTILITIES

DMP

GIVES UNFORMATTED DUMP OF VARIOUS PARTS OF THE SYSTEM.

IS A STAND-ALONG PROGRAM DEADSTARTED INTO MIOP.

PRINTS OUT THE FOLLOWING REGISTERS AND MEMORIES:

- CENTRAL MEMORY
- BUFFER MEMORY
- IOP LOCAL MEMORIES
- IOP A,B,C, AND OPERAND REGISTERS AND EXIT STACK.
- LOCAL AND BUFFER MEMORY TRACE BUFFERS

SHOULD ONLY BE USED IF SYSDUMP CANNOT BE BROUGHT UP.

SEE APPENDIX B, IOS OPERATOR'S GUIDE SG-0051 FOR DETAILED OPERATION OF DMP.

SYSDUMP

DUMPS SELECTED RESOURCES TO AN AREA OF DISK PRE-SELECTED AT INSTALL TIME, OR SPECIFIED DURING A SYSDUMP.

THIS DUMP MAY THEN BE FORMATTED VIA FDUMP AND DISPOSED APPROPRIATELY.

DISK RESTART OF THE IOS MAY BE DONE WHEN THE DUMP IS COMPLETE.

THE FOLLOWING MEMORIES AND REGISTERS MAY BE DUMPED.

CENTRAL MEMORY

BUFFER MEMORY

IOP LOCAL MEMORIES

IOP OPERAND REGISTERS

IOP A, B, C, E REGISTERS AND EXIT STACK

IOP CHANNELS' BZ AND DN FLAGS

CPU B,T,V AND VM REGISTERS

SYSDUMP IS ENTERED BY TYPING CNTRL-D AT THE MIOP KERNEL CONSOLE.

SEE APPENDIX B, IOS OPERATOR'S GUIDE SG-0051 FOR DETAILED OPERATION OF SYSDUMP.

TRACE DUMP

AN ON-LINE DUMP OF THE HISTORY TRACE BUFFERS MAY BE TAKEN USING THE TRACE KERNEL COMMAND.

THE LOCAL TRACE BUFFER OF THE IOP AT WHICH THE COMMAND IS ENTERED MAY BE DUMPED TO THE PRINTER.

THE BUFFER MEMORY TRACE BUFFER OF EACH IOP MAY BE DUMPED TO THE PRINTER.

FORMAT:

$$\text{TRACE DUMP} \begin{Bmatrix} \text{LOCAL} \\ \text{MOS} \end{Bmatrix}$$

LOCAL - DUMP THIS IOP'S LOCAL MEMORY TRACE BUFFER.

MOS - DUMP ALL IOP BUFFER MEMORY TRACE BUFFERS.