

UNISYS

OS/3

**Interactive
Services
Commands
and Facilities**

**Programming
and Operations
Quick-Reference
Guide**

Relative to Release
Level 11.0

August 1987

Priced Item

Printed in U S America
UP-9973

UNISYS

OS/3

**Interactive
Services
Commands
and Facilities**

**Programming
and Operations
Quick-Reference
Guide**

Copyright© 1987 Unisys Corporation
All Rights Reserved

Unisys is a trademark of Unisys Corporation.

Previous Title: OS/3 Interactive Services

Commands and Facilities Summary

Relative to Release
Level 11.0

August 1987

Priced Item

Printed in U S America
UP-9973

NO WARRANTIES OF ANY NATURE ARE EXTENDED BY THE DOCUMENT. Any product and related material disclosed herein are only furnished pursuant and subject to the terms and conditions of a duly executed Program Product License or Agreement to purchase or lease equipment. The only warranties made by Unisys, if any, with respect to the products described in this document are set forth in such License or Agreement. Unisys cannot accept any financial or other responsibility that may be the result of your use of the information in this document or software material, including direct, indirect, special or consequential damages.

You should be very careful to ensure that the use of this information and/or software material complies with the laws, rules, and regulations of the jurisdictions with respect to which it is used.

The information contained herein is subject to change without notice. Revisions may be issued to advise of such changes and/or additions.

FASTRAND, ✦SPERRY, SPERRY✦UNIVAC, SPERRY, SPERRY UNIVAC, UNISCOPE, UNISERVO, UNIS, UNIVAC, and ✦ are registered trademarks of Unisys Corporation. ESCORT, PAGERWRITER, PIXIE, PC/IT, PC/HT, PC/microIT, SPERRYLINK, and USERNET are additional trademarks of Unisys Corporation. MAPPER is a registered trademark and service mark of Unisys Corporation. CUSTOMCARE is a service mark of Unisys Corporation.

PAGE STATUS SUMMARY

ISSUE: UP-9973 Update B

RELEASE

LEVEL: 11.0 Forward

Part/Section	Page Number	Update Level
Cover		B
Title Page/Disclaimer		B*
PSS	1, 2	B
Preface	1	Orig.
Contents	1 thru 3	Orig.
Text	Tab Breaker 1	Orig. Orig.
	Tab Breaker 2 thru 5	Orig. Orig.
	Tab Breaker 6	Orig. A
	7	Orig.
	Tab Breaker 8	Orig. Orig.
	9 thru 12	B
	13 thru 19	Orig.
	Tab Breaker 20 thru 23	Orig. Orig.
	Tab Breaker 24	Orig. Orig.
	25 thru 29	A
	29a, 29b	B*
	30	B
	31, 32	A

*New pages

Part/Section	Page Number	Update Level
Text (cont)	Tab Breaker	Orig.
	33, 34	Orig.
	35	B
	36 thru 68	Orig.
	69	B
	70, 71	Orig.
	Tab Breaker	Orig.
	72 thru 80	Orig.
	Tab Breaker	Orig.
	81 thru 108	Orig.
	Tab Breaker	Orig.
	109 thru 111	Orig.
	Tab Breaker	Orig.
	112 thru 118	Orig.
	Tab Breaker	Orig.
	119 thru 121	Orig.
	Tab Breaker	Orig.
	122 thru 126	Orig.
	Tab Breaker	Orig.
	127	B
	128 thru 130	Orig.
User Comment Form		

Preface

This summary helps you use the interactive commands and facilities available on the OS/3 operating system. Included are:

- All the commands you may enter from a workstation
- Editor commands
- Information on:
 - Entering information from a workstation
 - Command conventions used in this manual
 - Logon and ICAM terminal sign-on procedures
 - Initiating the various interactive facilities (such as job control dialog)
 - Function keys
 - Use of various types of terminals as workstations



Contents

PAGE STATUS SUMMARY

PREFACE

CONTENTS

INTRODUCTION

GENERAL INSTRUCTIONS

 ENTERING INFORMATION ON A WORKSTATION

 Using the Correct Mode

 Using more than One Line to Enter

 a Command, Response, or Message

COMMAND CONVENTIONS USED IN THIS MANUAL

1

2

2

2

3

4

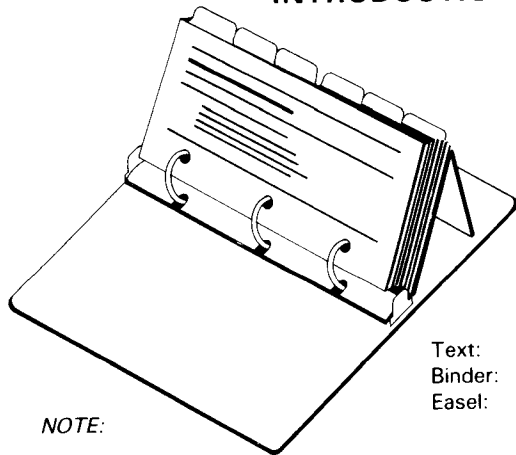
LOGON PROCEDURE	6
STANDARD TERMINAL DIALOG FOR ICAM TERMINALS	7
RUNNING JOBS	8
CHANGING JOB SCHEDULING	
CONTROLLING JOBS	20
CONTROLLING SPOOLING	24
UTILITY COMMANDS	33
INTERACTIVE FACILITIES	72
INTERACTIVE DATA UTILITIES	72
JOB CONTROL DIALOG	73
SCREEN FORMAT GENERATOR	73
USING SCREEN FORMATS IN YOUR PROGRAM	74
MENU GENERATOR	75

USING MENUS IN YOUR PROGRAM	76
INTERACTIVE DUMP/RESTORE HARDWARE UTILITY	78
BASIC	79
ESCORT	80
GENERAL EDITOR COMMANDS	81
RPG II EDITOR	
COBOL EDITOR	109
DDP COMMANDS	112
INTERACTIVE TERMINALS	119
FUNCTION KEY SUMMARY	122
COMMAND INDEX	127



To use this summary most effectively, you should order the binder (UP-8573) and the easel (UP-8573.1) from your Sperry representative. To assemble, open the binder and open the rings. Hook the folded vinyl easel onto the rings on the right-hand side. The shorter flap of the easel (the side without punched holes) should lie against the back cover of the open binder. Place the pages of the book on the rings on the left-hand side. The first page of the book should rest against the inside front cover of the binder. Close the rings. Lay the book flat on your desk or workstation so that the long side of the book is facing you. Unfold the easel so that it forms a triangle with the back cover of the book. Now raise the pages of the book against the easel. You can now flip down through the book, guided by the tabs.

INTRODUCTION



Text: UP-9973
Binder: UP-8573
Easel: UP-8573.1

NOTE:

To receive both the binder and easel, you must indicate both UP numbers to your Sperry representative.

GENERAL INSTRUCTIONS

ENTERING INFORMATION ON A WORKSTATION

Using the Correct Mode

To enter commands and respond to messages at the System 80 workstation, you must be in SYSTEM mode. To get into SYSTEM mode, press the FUNCTION key, hold it down, and press the SYS MODE key.

To enter data and output to user-created and system programs (including facilities like EDT) you must be in WORKSTATION mode. To get into WORKSTATION mode, press the FUNCTION key, hold it down, and press the WS MODE key.

For instructions on how to use a terminal other than the System 80 workstation, refer to the discussion entitled INTERACTIVE TERMINALS in this manual.

Using More than One Line to Enter a Command, Response, or Message

If you need more than one line (80 characters) of a workstation screen to enter a command, response, or message, place a dash (-) as the last character on the line. The dash is a continuation character. It prompts the system to ask you, in a message, for further input. Use as many lines of the screen as you need; just place a dash at the end of every line except the last one.

Example:

- ```
1. COPY MODULE=MYMODULE,FILENAME=MYFILE,VSN=REL071 TO MODULE=NEWMODUL,FILENAME=-
2. 0G? IS93 ENTER CONTINUATION?
3. 0G NEWFILE,VSN=NEWVOL,SIZE=10,CONTIG=NO
```

1. After keying in almost a full line, add a dash as the last character on the line. Press the XMIT key.
2. The system asks you for a continuation.
3. To answer, key in the system message number (0G), enter a space, and enter the rest of the command. Press the XMIT key.

## COMMAND CONVENTIONS USED IN THIS MANUAL

- If part of a command or parameter is underscored, it means that only the underscored portion has to be keyed in. When a command or parameter is not underscored, you must enter the entire command or parameter. In the following example, ER of the ERASE command is underscored. You must enter at least ER for the command to work, but you can enter ERA, ERAS, or ERASE.

### ERASE

Where ER is the minimum keyin required to successfully use the ERASE command.

- Parameters printed in lowercase letters designate undefined variables:

`,VSN=volume serial number`

Where volume serial number is the name of the volume you're working with.

- Optional parameters are enclosed in brackets:

`[,WRPASS=write password]`

- Alternate choices for a parameter are enclosed in braces:

,DEVICE= { nnn  
          { RDR }

- Default values are values automatically generated by the system when you do not specify a value for a parameter. Default values are shown shaded in each command format:

,COPIES= { n  
          { 1 }

- Spaces are indicated by a delta ( $\Delta$ ) symbol:

COPY $\Delta$ MODULE



## LOGON PROCEDURE

You must begin every workstation session with the LOGON command. To log on, press XMIT and fill in the logon menu displayed to you. You may also log on by entering SYSTEM mode and entering the LOGON command as follows:

```
LOGON_userid[,acct][,password][,exec-pro] [,BULLETIN={NO }] [,LOG={NO }]
```

```
[,NEWPASS=new-password]
```

|          |                             |
|----------|-----------------------------|
| userid   | 1-6 alphanumeric characters |
| acct     | 1-4 alphanumeric characters |
| password | 1-8 alphanumeric characters |
| exec-pro | 1-8 alphanumeric characters |

To log *off* the workstation, enter the LOGOFF command:

### LOGOFF

There are no parameters entered with this command.

# STANDARD TERMINAL DIALOG FOR ICAM TERMINALS

If you are using a remote terminal as a workstation, you must connect the terminal to ICAM (Integrated Communications Access Method) before you log on. You connect the terminal to ICAM with the following command:

**\$\$\$ON△xxxxyyyy**

**xxxx** Specifies the logical name of your terminal.

**yyyy** Specifies the logical name of the program you are signing on to use.

After entering the \$\$\$ON command, you will receive a message informing whether or not your sign-on attempt was successful.

After finishing your workstation session and logging off, you terminate your communications link with ICAM by issuing the following command:

**\$\$\$OFF** There are no parameters entered with this command.

## RUNNING JOBS CHANGING JOB SCHEDULING

The following commands are used to run jobs from the workstation and change the scheduling for execution of those jobs.

---

### FILE Command

$$\underline{\text{FILE}} \left\{ \begin{array}{l} ([\text{did}], \text{label}) \\ (\text{RDR}, \text{label}) \end{array} \right\} \Delta \left[ \begin{array}{l} \left( \begin{array}{l} \text{:alt-filename} \\ \text{:alt-filename, } \left\{ \begin{array}{l} \text{RES} \\ \text{RUN} \\ \text{vsn} \end{array} \right\} \end{array} \right) \\ \left( \begin{array}{l} \text{:alt-filename, } \left[ \begin{array}{l} \text{RES} \\ \text{RUN} \\ \text{vsn} \end{array} \right], \text{write-pass} \end{array} \right) \end{array} \right]$$

The FILE command files jobs and job control procedures (jprocs) into the permanent job control stream library file (\$Y\$JCS) or into an alternate SAT library file.

---

## RUN Command

$$\text{RUN} \left\{ \begin{array}{l} [(did), label] \\ (RDR, label) \end{array} \right\} \Delta [jobname] [(new-name)] \left[ \begin{array}{l} (:alt-filename \\ :alt-filename, \left\{ \begin{array}{l} RES \\ RUN \\ vsn \end{array} \right\} \\ :alt-filename, \left[ \begin{array}{l} RES \\ RUN \\ vsn \end{array} \right], read-pass \end{array} \right] \right]$$
  
$$\left[ \begin{array}{l} , \left\{ \begin{array}{l} \underline{P}RE \\ \underline{H}IGH \\ \underline{N}OR \\ \underline{L}OW \end{array} \right\} \\ [time] \left[ \begin{array}{l} + \left\{ \begin{array}{l} d1 \\ \vdots \\ d9 \end{array} \right\} \end{array} \right] [key-1=val-1, \dots, key-n=val-n]$$

The RUN command enables you to execute user jobs from the workstation. The command causes the job control stream associated with your job to be read, expanded, and scheduled for execution. The RUN command is used when an input device, either a data set label diskette or the input spool file (RDR), is required.

---

## RV Command

RV△ jobname [(new-name)]  $\left[ \begin{array}{l} \text{:alt-filename} \\ \text{:alt-filename, } \left\{ \begin{array}{l} \text{RES} \\ \text{RUN} \\ \text{vsn} \end{array} \right\} \\ \text{:alt-filename, } \left[ \begin{array}{l} \text{RES} \\ \text{RUN} \\ \text{vsn} \end{array} \right], \text{read-pass} \end{array} \right] \left[ \begin{array}{l} \text{, } \left\{ \begin{array}{l} \text{PRE} \\ \text{HIGH} \\ \text{NOR} \\ \text{LOW} \end{array} \right\} \text{ [time]} \\ \text{+ } \left\{ \begin{array}{l} \text{d1} \\ \text{:} \\ \text{d9} \end{array} \right\} \end{array} \right]$

[,key-1=val-1,...,key-n=val-n]

The RV command enables you to execute user jobs from the workstation. The command causes the job control stream associated with your job to be read, expanded, and scheduled for execution. The RV command is used when there is no input device needed.

---

## SI Command

SI {([did],label)}Δ [jobname] [(new-name)]  
{(RDR,label)}

$\left[ \begin{array}{l} \text{:alt-filename} \\ \text{:alt-filename, } \left\{ \begin{array}{l} \text{RES} \\ \text{RUN} \end{array} \right\} \\ \text{:alt-filename, } \left\{ \begin{array}{l} \text{RES} \\ \text{RUN} \\ \text{vsn} \end{array} \right\}, \text{read-pass;} \end{array} \right]$   $\left[ \begin{array}{l} \text{PRE} \\ \text{HIGH} \\ \text{NOR} \\ \text{LOW} \end{array} \right] [\text{time}] \left[ \begin{array}{l} + \\ \text{d1} \\ \vdots \\ \text{d9} \end{array} \right]$

The SI command enables you to run jobs previously saved in the expanded state. The SI command is used when embedded data in the job control stream is to be replaced.

## SC Command

SC△ jobname [(new-name)] [ { :alt-filename  
: (alt-filename, { RES }  
                          { RUN }  
                          vsn) }  
: (alt-filename, [ { RES } ], read-pass) } ] [ { PRE  
                  HIGH  
                  NOR  
                  LOW } ] [time] [ + { d1  
                                      :  
                                      d9 } ]

The SC command enables you to run jobs that you have previously saved in their expanded state. The SC command is used when the job control stream does not require an input device to replace embedded data.

---

## EXECUTE Command

**EXECUTE**△program-name

The EXECUTE command enables you to run programs in an interactive job environment.

To use EXECUTE, you must first run the super-set job control stream that incorporates the program you want to use.

---

## ULD Command

ULD△,filename,vsn,SIZE=n△

|   |         |   |   |   |         |   |
|---|---------|---|---|---|---------|---|
| { | PRINT   | } | △ | { | SCRATCH | } |
| { | NOPRINT | } |   | { | SAVE    | } |

The upline dump command enables a UTS 400 user to get a dump of the terminal's memory. A user can choose to print or save the dump file.

---



---

## DLOAD Command

DLOADΔ {program-name}  
          {/OFFLINE }

The DLOAD command allows a UTS 400 terminal user or a UTS 40/40D workstation user to downline load a program to terminal memory. Any program you load using the DLOAD command must reside in the \$\$LOD library.

---

---

## UNLOAD Command

UNLOAD

The UNLOAD command frees devices that were allocated to a downline-loaded program.

---

---

## BEGIN Command

Rescheduling of All Jobs or All Jobs on a Particular Job Queue:

BEGIN $\Delta$ JBQ [ , ( A )  
                  { H  
                  { N  
                  { P  
                  { L ) ]

Rescheduling of an Individual Job:

BEGIN $\Delta$ jobname

The BEGIN command reinstates scheduling of jobs deferred by the HOLD command. You may reschedule individual jobs initiated under your user-id, all of your jobs present in a particular queue, or all of your jobs (all those initiated under your user-id).

---

---

## HOLD Command

Holding All Jobs or All Jobs on a Particular Job Queue:

HOLDΔJBQ [ , ( A ) ]  
                  { H  
                  { N  
                  { P  
                  { L

Holding an Individual Job:

HOLDΔjobname

The HOLD command permits you to defer scheduling of jobs initiated under your user-id. You may hold individual jobs initiated under your user-id, all your jobs present in a particular queue, or all of your jobs (all those initiated under your user-id).

---

---

## CHANGE Command

CHANGEΔjobname,  $\left\{ \begin{array}{l} H \\ N \\ P \\ L \end{array} \right\}$


The CHANGE command permits you to change the scheduling queue in which your job is residing.

---

---

## DELETE Command

Deleting All Jobs from One or All Job Queues:

DELETE△JBQ,  [, LOG]

Deleting A Specific Job:

DELETE△jobname [, LOG]

The DELETE command permits you to remove jobs from scheduling queues. Jobs deleted will not be executed. You may remove single jobs (initiated under your user-id), all your jobs on a particular queue, or all your jobs on all queues.

---

---

## DISPLAY JBQ Command

DISPLAYΔJBQ, {  
A  
H  
N  
P  
L

The DISPLAY JBQ command permits you to display, on the workstation screen, the contents (names of jobs) of each scheduling queue.

---

## CONTROLLING JOBS

The following commands permit your workstation to act as a minisystem console to control those jobs initiated or running under your user-id. You may *only* control jobs initiated or running under your user-id.

---

### CANCEL Command

CANCEL $\Delta$ jobname [ { D } ]  
                          [ { N } ]

The CANCEL command enables you to immediately halt processing of a job; the currently executing job step is not completed. You may specify whether or not you want a dump taken when you enter the CANCEL command.

---

---

## CONNECT Command

CONNECT△job[,filename]

The CONNECT command is used to connect the WORKSTATION mode of your workstation to a job running on the system. You connect to a job at any time while it is running.

---

---

## DISPLAY JS Command

DISPLAY△JS[,jobname]

The DISPLAY JS command enables you to obtain information about jobs initiated under your user-id. You receive information about the status of your job, whether it is on a scheduling queue, or executing, and if not executing, why not.

---



---

## FREE Command

FREE

The FREE command permits you to manually disconnect a workstation from a job. When this command is issued, the WORKSTATION mode of your workstation is disconnected from the program to which it was connected.

---

---

## PAUSE Command

PAUSE△jobname

The PAUSE command permits you to suspend processing of a job. You may enter the PAUSE command at any time, and job processing is immediately suspended. Use the GO command to restart job processing.

---

---

## GO Command

GO△jobname[,nn]

The GO command reactivates jobs suspended by the PAUSE command or by certain statements within the job control stream associated with the suspended job.

---

---

## STOP Command

STOP△jobname

The STOP command enables you to terminate a job at the completion of the currently executing job step.

---

## CONTROLLING SPOOLING

The following commands allow you to control the processes of input and output spooling for your job. For your convenience, the spool file directories are reproduced here:

| Directory | File Function                                                                |
|-----------|------------------------------------------------------------------------------|
| ALL       | Makes all directories accessible to the command in which it is specified     |
| LOG       | Indicates that the file referenced is a job log file                         |
| PUNCH     | Indicates that the file is to be output to either a card punch or a diskette |
| PRINT     | Indicates that the file is to be output to a printer                         |
| RDR       | Indicates that the file was originally from a card reader or diskette        |

## DISPLAY Command

Obtaining Information about Active Spool Files:

DISPLAY $\Delta$ ACT [ , { ALL } ] [ , ACT=acctno  
                  { PRINT }  
                  { PUNCH } ]

[ , CART=cartridge-name ] [ , DEV= { 770 }  
                                      { 776 }  
                                      { 789 } ]

[ , FILE=filename ] [ , FORM=formname ] [ , JOB=jobname ] [ , STEP=stepno ]

The DISPLAY ACT command permits you to obtain information about spool files being created.

## DISPLAY Command

Obtaining Information about Completed Spool Files:

```
DISPLAY/SPL [{ ALL
 LOG
 PRINT
 PUNCH
 RDR }] [, ACCT=acctno]
```

```
[, CART=cartridge-name] [, DEV={ 770
 776
 789 }]
```

```
[, FILE=filename] [, FORM=formname] [, JOB=jobname] [, STEP=stepno]
```

The DISPLAY SPL command permits you to obtain information about completed spool files.

---

## HOLD SPL Command

```
HOLD SPL, { ALL
 PRINT
 PUNCH
 LOG
 RDR } [,ACCT=acctno][,BNUMB=binary jobno]

 [,CART=cartridge-name] [,DEV={ 770
 776
 789 }]

 [,FILE=filename][,FORM=formname][,JOB=jobname][,STEP=stepno]
```

The HOLD SPL command permits you to place files in a hold state, unavailable for processing.

---

## BEGIN SPL Command

BEGIN SPL, ( ALL ) [,ACCT=acctno][,BNUMB=binary jobno]

LOG  
PRINT  
PUNCH  
RDR

[,CART=cartridge-name] [,DEV={770  
776  
789}]

[,FILE=filename][,FORM=formname][,JOB=jobname][,STEP=stepno] [,OUT={did  
NO}]

The BEGIN SPL command releases spool files held by a HOLD SPL command. Entering this command also loads the output writer to print the file as soon as a printer becomes available.

---

## DELETE SPL Command

DELETE SPL,  $\left. \begin{array}{l} \text{ALL} \\ \text{LOG} \\ \text{PRINT} \\ \text{PUNCH} \\ \text{RDR} \end{array} \right\}$  [, ACT=acctno][, BNUMB=binary jobno]

[, CART=cartridge-name]  $\left[ \text{DEV} = \left\{ \begin{array}{l} 770 \\ 776 \\ 789 \end{array} \right\} \right]$

[, FILE=filename][, FORM=formname][, JOB=jobname][, STEP=stepno]

The DELETE SPL command enables you to delete files from spool queues. You may delete completed files only – files being created may not be deleted.

---



---

## CHANGE SPL Command

CHANGEΔSPL, { ALL  
LOG  
PRINT }

[,modifier-1 ... modifier-n]

[,COPIES=nnn]

## CHANGE SPL Command (cont)

```
{ [,DVC={ 770
 776
 PPC
 ANY
 CLASS1
 CLASS2
 CLASS3
 }]
 [,DVC= AUX, ID={ *
 {user-id} }]
 [,ID={ *
 {user-id} }]
}
```

The CHANGE SPL command permits you to change the device type and/or the number of copies of a spool file.

---

## BRKPT Command

$\text{BRKPT} \Delta \left\{ \begin{array}{l} \text{P} \\ \text{I} \end{array} \right\}, \left\{ \begin{array}{l} \text{PR} \\ \text{PU} \end{array} \right\}, [ , \text{ACCT} = \text{acctno} ] [ , \text{CART} = \text{cartridge-name} ]$

$\left[ \begin{array}{l} \text{DEV} = \left( \begin{array}{l} 770 \\ 776 \\ 789 \end{array} \right) \end{array} \right] [ , \text{FILE} = \text{filename} ] [ , \text{FORM} = \text{formname} ], \text{JOB} = \text{jobname} [ , \text{HOLD} ]$

The BRKPT (breakpoint) command closes one or more spool files and makes them available to an output writer for printing or punching. The remainder of the file created after the BRKPT command is issued is placed on another file.

---

---

## PR/PU Command

{PR} Δ[function-code][,ACCT=acctno][,BNUMB=binary jobno]  
{PU}  
[,CART=cartridge-name][,FILE=filename]  
[,FORM=formname][,JOB=jobname]

The PR/PU command enables you to manually load an output writer to print (PR) or punch (PU) spooled files associated with your job.

---

---

## RP Command

```
RPΔ[function-code][,A_CCT=acctno][,B_NUMB=binary jobno]
 [,CART=cartridge-name][,F_FILE=filename]
 [,F_FORM=formname][,J_JOB=jobname]
```

The RP command enables you to manually load an output writer to print to an auxiliary printer. *To use auxiliary printers, you must have generated your system to use auxiliary printers and directed your print files to an auxiliary printer prior to the time your system prints your print files.* RP alone will not direct printing to an auxiliary printer.

---

The following commands permit you to perform utility functions, such as copying files, allocating files, scratching files, and obtaining status information on various aspects of your OS/3 interactive processing system:

---

### ALLOCATE Command

```

ALLOCATE {ST} , FILENAME = { filename } [, RDPASS = password [] , WRPASS = password]
 {MI} { 'filename' }
 { "filename" }

, VSN = volume [, CONTIG = { YES }] [, INC = { n }] , SIZE = n
 { NO }

```

(continued)

## ALLOCATE Command (cont)

The ALLOCATE command permits you to allocate space for files interactively, without the need for job control statements. You may allocate SAT and MIRAM files using this command.

### *NOTE:*

*When allocating space on diskettes:*

- *If the diskette is recorded in data-set-label (DSL) mode, give the SIZE= and INC= parameters in blocks.*
  - *If the diskette is recorded in format-label (FL) mode, give the SIZE= and INC= parameters in cylinders.*
-

---

## ASK Command

ASK $\Delta$ [user-id], 'text'

The ASK command enables you to ask questions of other workstation users or the system operator. The command displays your question to the other user, accepts the reply, and returns the reply to you.

---

## BRKPT LOG Command

BRKPT $\Delta$ LOG [ ,OUT= { TAPE  
                  DISK  
                  DISKETTE } ] [ ,HOLD ]

The BRKPT LOG command enables you to close a workstation log file and make it available to the output writer for printing before you log off the workstation.

---



---

## COMMENT Command

COMMENT△MODULE=modulename [ ,TIPE= $\left\{ \begin{array}{l} \text{module-type} \\ \mathbf{s} \end{array} \right\} \right]$  ,FILENAME= $\left\{ \begin{array}{l} \text{filename} \\ \text{'filename'} \\ \text{"filename"} \end{array} \right\}$

[ ,RDPASS=password][ ,WRPASS=password] ,VSN=volume△text

The COMMENT command enables you to put a comment in a library module header or replace an existing comment on a library module header. Comments may be up to 30 characters long.

---

---

## COPY Command

Copying SAT or MIRAM Library Modules:

```
COPY△MODULE=modulename [,TYPE={module-type}],FILENAME={ filename
 { 'filename'
 { "filename"

[,RDPASS=password],VSN=volume△TO△MODULE=modulename [,TYPE={module-type}
 { S

,FILENAME={ filename } [,RCSZ=n[],WRPASS=password],VSN=volume
 { 'filename'
 { "filename"

[,CONTIG={ YES }] [,INC={ n }] [,SIZE=n] [,SAT={ YES }] △[NUMBER] [,HEX] [,WAIT]
 { NO }] [{ 1 }] [{ NO }]
```

The COPY command enables you to copy files and, in the process of copying, to alter the format of a file.

---

## COPY Command

Copying MIRAM Data Files:

COPYΔFILENAME= { filename } [, RDPASS=password], VSN=volume [, KEYNO= { n } ]  
{ 'filename' }  
{ "filename" }

[, DEVICE= { did } ] ΔTOΔFILENAME= { filename } [, WRPASS=password]  
{ **DISK** }  
{ 'filename' }  
{ **DISKETTE** }  
{ "filename" }

, VSN=volume [, CONTIG= { **YES** } ] [, INC= { n } ] [, KEYNO= { n } ] [, KEYi= { n:m }  
{ (n:m, { **DUP** }, { **CHG** } ) } ]  
{ (n:m, { **NDUP** }, { **NCHG** } ) } ]

(continued)

## COPY Command (cont)

Copying MIRAM Data Files (cont):

```
[,SIZE=n] [,INIT={YES
 NO}] [,RCB={YES
 NO}] [,RCFM={FIX
 VAR}] [,RCSZ=n] [,EXTEND={YES
 NO}]
[,BESZ=n] [,SCSZ={n
 256}] [,DEVICE={did
 DISK
 DISKETTE}]△[NUMBER][,HEX][,WAIT]
```

The COPY command enables you to copy files and, in the process of copying, to alter the format of a file.

---

## COPY Command

Copying Spool Files:

```
COPY△[JOB=jobname] [,HOLD={L
 {N
 {Y}}] [,FILENAME={filename
 {'filename'
 "filename"}] [,ACCT=acct] ,QUEUE={LOG
 {PRINT
 {PUNCH
 {RDR}} } [,ALL={YES
 {NO}}]

[,SKIP={n
 {0}}] △TO△[JOB=jobname] [,HOLD={N
 {Y}}] [,FILENAME={filename
 {'filename'
 "filename"}] ,QUEUE={PRINT
 {PUNCH
 {RDR}} }
```

△[NUMBER][ ,HEX][ ,WAIT]

The COPY command enables you to copy files and, in the process of copying, to alter the format of a file.

# COPY Command

Copying Tape Files:

COPY $\Delta$  [ FILENAME= { filename  
'filename'  
"filename" } ] [, RDPASS=password], VSN=volume, DEVICE= { did }  
                                                                                                          { TAPE }

[ , BKNO= { YES } ]  $\Delta$ TO $\Delta$   
                                                          { NO }

FILENAME= { filename } [, WRPASS=password], VSN=volume, DEVICE= { did }  
                                                          { 'filename' }                                                                                  { TAPE }  
                                                          { "filename" }

[ , INIT= { YES } ] [ , EXTEND= { YES } ]  
                                                          { NO }                                                          { NO }

(continued)

## COPY Command (cont):

Copying Tape Files (cont):

[,BFSZ=n] [,BKNO={YES  
NO}] [,RCFM={~~FIXUNB~~  
FIXBLK  
VARUNB  
VARBLK  
UNDEF}] [,RCSZ=n]△[NUMBER][,HEX][,WAIT]

The COPY command enables you to copy files and, in the process of copying, to alter the format of a file.

---

## COPY Command

Copying Unit Record Files:

COPY $\Delta$ DEVICE= { did  
                  DISKETTE } , FILENAME= { filename  
                                          'filename'  
                                          "filename" } , VSN=volume $\Delta$ TO $\Delta$ DEVICE= { did  
                                                                                          DISKETTE  
                                                                                          PRINT  
                                                                                          PUNCH }

[ , RCFM= { ~~FIX~~  
                  VAR } ] [ , RCSZ=n ] , FILENAME= { filename  
                                          'filename'  
                                          "filename" } , VSN=volume $\Delta$ [ NUMBER ] [ , HEX ] [ , WAIT ]

The COPY command enables you to copy files and, in the process of copying, to alter the format of a file.



---

## DEFKEY Command

```
DEFKEY△ {F#nn} , { 'command string' }
 {MW} { "command string" }
```

The DEFKEY command lets you assign any interactive command to a function key or the MESSAGE WAITING key.

---

## DEFKEY (delete) Command

```
DEFKEY△ {F#nn}
 {MW}
```

To free a function key or the MESSAGE WAITING key from a command assignment, you use the delete form of the DEFKEY command. The delete DEFKEY command is exactly like the DEFKEY command, minus the command string.

---

---

## DEFKEY DISPLAY Command

DEFKEY△DISPLAY

The DEFKEY DISPLAY command displays your function key and MESSAGE WAITING key assignments.

---

---

## DISPLAY LOG Command

DISPLAY△LOG

The DISPLAY LOG command gives you a 1-line report on the status of your workstation log file. The display shows the number of workstation lines used since you logged on.

---

---

## EDT Command

EDT△[initial command]

The EDT command permits you to initialize the general editor.

---

---

## ENTER Command

To Run a Command Stream from a Library File:

ENTER△MODULE=modulename [ ,TYPE={module-type} ] ,FILENAME={filename  
{ 'filename' }  
{ "filename" }  
[ ,RDPASS=password ] ,VSN=volume

The ENTER command permits you to enter a series of workstation commands as a batch processing job. Sessions may be entered from a card reader, tape, diskette, spooled file, or library file. Output from the job is always directed to the printer. You may enter any workstation command except those directly affecting the workstation device, such as SCREEN.

---

---

## ENTER Command

To Run a Command Stream from a Spooled File:

$$\text{ENTER}\Delta \left[ \text{HOLD} = \begin{cases} \text{N} \\ \text{Y} \end{cases} \right], \text{FILENAME} = \begin{cases} \text{filename} \\ \text{'filename'} \\ \text{"filename"} \end{cases} \text{QUEUE} = \text{RDR}$$

The ENTER command permits you to enter a series of workstation commands as a batch processing job. Sessions may be entered from a card reader, tape, diskette, spooled file, or library file. Output from the job is always directed to the printer. You may enter any workstation command except those directly affecting the workstation device, such as SCREEN.

---

---

## ENTER Command

To Run a Command Stream from a DSL Diskette:

```
ENTER△ [FILENAME= { filename
 { 'filename'
 { "filename" } }] [,RDPASS=password],VSN=volume [,DEVICE= { did
 { DISKETTE }]
```

The ENTER command permits you to enter a series of workstation commands as a batch processing job. Sessions may be entered from a card reader, tape, diskette, spooled file, or library file. Output from the job is always directed to the printer. You may enter any workstation command except those directly affecting the workstation device, such as SCREEN.

---

---

## ENTER Command

To Run a Command Stream from a Tape:

```
ENTER△ [FILENAME = { filename
 { 'filename'
 "filename" } } [, RDPASS=password], VSN=volume [, DEVICE = { did
 { TAPE } }]
```

The ENTER command permits you to enter a series of workstation commands as a batch processing job. Sessions may be entered from a card reader, tape, diskette, spooled file, or library file. Output from the job is always directed to the printer. You may enter any workstation command except those directly affecting the workstation device, such as SCREEN.

---

---

## ENTER Command

To Run a Command Stream from a Card Reader:

ENTER△ DEVICE= {did}  
                          {RDR}

The ENTER command permits you to enter a series of workstation commands as a batch processing job. Sessions may be entered from a card reader, tape, diskette, spooled file, or library file. Output from the job is always directed to the printer. You may enter any workstation command except those directly affecting the workstation device, such as SCREEN.

---



---

## ERASE Command

Erasing a Library Module:

ERASE $\Delta$ MODULE=modulename [ , TYPE= $\left\{ \begin{array}{l} \text{module-type} \\ \mathbf{s} \end{array} \right\} \right]$  , FILENAME= $\left\{ \begin{array}{l} \text{filename} \\ \text{'filename'} \\ \text{"filename"} \end{array} \right\}$   
  
[ , WRPASS=password ] , VSN=volume

---

---

## ERASE Command

Erasing Library and MIRAM Data Files:

$$\text{ERASE}\Delta\text{FILENAME} = \left. \begin{array}{l} \text{filename} \\ \text{'filename'} \\ \text{"filename"} \end{array} \right\} [ , \text{WRPASS} = \text{password} ], \text{VSN} = \text{volume}$$

The ERASE command permits you to delete library and data files, as well as library modules. Before executing takes place, the ERASE command double-checks with you to make certain you want to erase a file. This guards against inadvertent erasures.

---

---

## FSTATUS Command

FSTATUSΔ[MODULE=modulename] [ ,TYPE= $\left\{ \begin{array}{l} \text{module-type} \\ \mathbf{S} \end{array} \right\} ]$ ,FILENAME= $\left\{ \begin{array}{l} \text{filename} \\ \text{'filename'} \\ \text{"filename"} \end{array} \right\}$

[ ,RDPASS=password ],VSN=volumeΔ[LONG]

The FSTATUS command permits you to interactively obtain information about files and their contents. You may reference only library file modules.

---

---

## HELP Command

`HELP△ { command  
          { message-no  
          { keyword-parameter } }`

The HELP command permits you to obtain information about how to use various workstation commands and their parameters and how to respond to error messages from the system. Entering HELP and a command message number or keyword parameter produces a display explaining the command, message, or parameter. You may not enter an abbreviated form of the command or parameter.

---

---

## MENU Command

`MENU△ [ [menu-name] [ , { filename, vsn  
          { SYSFMT [ , { RES } ] } ] ] ] ]`

The MENU command displays either a standard system menu or a user-created menu.

---

---

## PRINT Command

Printing an SAT or MIRAM Library Module:

```
PRINT△MODULE=modulename [,TYPE={module-type}],FILENAME={ filename
 { s
 'filename'
 "filename"
 }
]
[,RDPASS=password],VSN=volume [,COPIES={n}]△[DIRECT][,NUMBER][,HEX][,WAIT]
```

The PRINT command enables you to make a printed copy of a library file or module, a spool file, a MIRAM file, a tape file, or a unit record file. PRINT can run as a background job to allow you to use the workstation for other functions while the file is being printed.

---

---

## PRINT Command

Printing a MIRAM Data File:

```
PRINT△FILENAME= { filename } [,RDPASS=password],VSN=volume [,KEYNO={ n }]
 { 'filename' }
 { "filename" }

[,COPIES={ n }] [,DEVICE={ did }]△[DIRECT][,NUMBER][,HEX][,WAIT]
 { 1 } { DISKETTE }
 { DISK }
```

The PRINT command enables you to make a printed copy of a library file or module, a spool file, a MIRAM file, a tape file, or a unit record file. PRINT can run as a background job to allow you to use the workstation for other functions while the file is being printed.

---

---

## PRINT Command

Printing a Tape File:

```
PRINT△[FILENAME={filename }][,RDPASS=password],VSN=volume,DEVICE={did }
 { 'filename' }
 { "filename" }
[,BKNO={YES }]△[DIRECT][,NUMBER][,HEX][,WAIT]
 { NO }
```

The PRINT command enables you to make a printed copy of a library file or module, a spool file, a MIRAM file, a tape file, or a unit record file. PRINT can run as a background job to allow you to use the workstation for other functions while the file is being printed.

---

---

## PRINT Command

Printing a Unit Record File:

```
PRINT△DEVICE= { did } , FILENAME= { filename } , VSN=volume△[DIRECT][, NUMBER]
 { DISKETTE } { 'filename' }
 { RDR } { "filename" }
[, HEX][, WAIT]
```

The PRINT command enables you to make a printed copy of a library file or module, a spool file, a MIRAM file, a tape file, or a unit record file. PRINT can run as a background job to allow you to use the workstation for other functions while the file is being printed.

---



## PRINT Command

Printing a Spool File:

```
PRINT△[JOB=jobname] [,HOLD={ L }] [,FILENAME={ filename }] [,ACCT=acct]
 { N }
 { 'filename' }
 { "filename" }
 ,QUEUE={ LOG } [,ALL={ YES }] [,COPIES={ n }] [,SKIP={ n }]△[DIRECT][,NUMBER][,HEX]
 { PRINT }
 { PUNCH }
 { RDR }
 [,WAIT]
```

The PRINT command enables you to make a printed copy of a library file or module, a spool file, a MIRAM file, a tape file, or a unit record file. PRINT can run as a background job to allow you to use the workstation for other functions while the file is being printed.

---

## PUNCH Command

Punching a Library Module:

```
PUNCH△MODULE=modulename [,TYPE={module-type}] [,FILENAME={filename
 {s
 'filename'
 "filename"}]
[,RDPASS=password], VSN=volume [,COPIES={n}]△[DIRECT][,WAIT]
```

The PUNCH command allows you to make a punched-card copy of a library file or module, a spool file, a MIRAM file, a tape file, or a unit record file. PUNCH can be run as a background job to allow you to use the workstation for other functions while the file is being punched.

---

---

## PUNCH Command

Punching a MIRAM Data File:

```
PUNCH△FILENAME={filename } [,RDPASS=password],VSN=volume [,KEYNO={n }]
 {'filename'}
 "filename"

[,COPIES={n }] [,DEVICE={did }]△[DIRECT][,WAIT]
 {DISKETTE }
 {DISK }
```

The PUNCH command allows you to make a punched-card copy of a library file or module, a spool file, a MIRAM file, a tape file, or a unit record file. PUNCH can be run as a background job to allow you to use the workstation for other functions while the file is being punched.

---

## PUNCH Command

Punching a Spool File:

PUNCH $\Delta$ JOB=jobname [ ,HOLD= $\left\{ \begin{array}{c} L \\ N \\ Y \end{array} \right\}$  ] ,FILENAME= $\left\{ \begin{array}{c} \text{filename} \\ 'filename' \\ "filename" \end{array} \right\}$  [ ,ACCT=acct ]  
 ,QUEUE= $\left\{ \begin{array}{c} LOG \\ PRINT \\ PUNCH \\ RDR \end{array} \right\}$  [ ,ALL= $\left\{ \begin{array}{c} YES \\ NO \end{array} \right\}$  ] [ ,COPIES= $\left\{ \begin{array}{c} n \\ 1 \end{array} \right\}$  ] [ ,SKIP= $\left\{ \begin{array}{c} n \\ 0 \end{array} \right\}$  ]  $\Delta$ [DIRECT][ ,WAIT]

The PUNCH command allows you to make a punched-card copy of a library file or module, a spool file, a MIRAM file, a tape file, or a unit record file. PUNCH can be run as a background job to allow you to use the workstation for other functions while the file is being punched.

---

## PUNCH Command

Punching a Tape File:

```
PUNCH△[FILENAME={filename }][,RDPASS=password],VSN=volume ,DEVICE={did }
 { 'filename' }
 { "filename" }
[,BKNO={YES }]△[DIRECT][,WAIT]
 { NO }
```

The PUNCH command allows you to make a punched-card copy of a library file or module, a spool file, a MIRAM file, a tape file, or a unit record file. PUNCH can be run as a background job to allow you to use the workstation for other functions while the file is being punched.

---

---

## PUNCH Command

Punching a Unit Record File:

$$\text{PUNCH}\Delta\text{DEVICE} = \left\{ \begin{array}{l} \text{did} \\ \text{DISKETTE} \\ \text{RDR} \end{array} \right\}, \text{FILENAME} = \left\{ \begin{array}{l} \text{filename} \\ \text{'filename'} \\ \text{"filename"} \end{array} \right\}, \text{VSN} = \text{volume}\Delta[\text{DIRECT}][, \text{WAIT}]$$

The PUNCH command allows you to make a punched-card copy of a library file or module, a spool file, a MIRAM file, a tape file, or a unit record file. PUNCH can be run as a background job to allow you to use the workstation for other functions while the file is being punched.

---

---

## REBUILD Command

### REBUILD

The REBUILD command restores messages requiring a response or a GO command to your workstation screen.

---

---

## RECALL Command

RECALL $\Delta$  { LAST $\Delta$ nn  
          { hh:mm:ss - hh:mm:ss } }

The RECALL command lets you display all or part of your workstation log file at your workstation screen. You can view selected portions of the log file by specifying a particular time span. Or, you can indicate the number of messages, prior to the most current one, that you'd like to see.

---

---

## RECOVER Command

RECOVER△MODULE=modulename [ ,TYPE= $\left\{ \begin{array}{l} \text{module-type} \\ \mathbf{s} \end{array} \right\} \text{ ] , FILENAME=$  $\left\{ \begin{array}{l} \text{filename} \\ \text{'filename'} \\ \text{"filename"} \end{array} \right\}$

[ ,RDPASS=password ][ ,WRPASS=password ] ,VSN=volume

The RECOVER command allows you to recover deleted source, procedure, or macro modules in the SAT library. You may also use the command to rename modules not deleted.

---



---

## REMARK Command

REMARK△text

The REMARK command allows you to enter a comment in a stream of commands. It is used principally in card decks for batch processing, but may be used in any situation where a comment needs to be inserted in a command stream.

---

---

## RESUME Command

RESUME

The RESUME command enables you to resume execution of a subsystem, such as the general editor or BASIC, that was suspended when the workstation entered SYSTEM mode.

---

## SCREEN Command

`SCREEN`△ [ `SCROLL` ] [ `UPPER` ] [ `XMIT`= { `VAR` } ] [ `XFER`= { `VAR` } ] [ `SPEED`= { `9600` } ]  
[ `ROLL` ] [ `LOWER` ] [ `CHAN` ] [ `CHAN` ] [ `4800` ]  
[ `NP` ] [ `ALL` ] [ `ALL` ] [ `2400` ]  
[ `WRAP` ] [ `ALL` ] [ `ALL` ] [ `1200` ]  
[ `NOROLL` ] [ `ALL` ] [ `ALL` ] [ `600` ]  
[ `300` ]  
[ `SPACEBAR`= { `DESTRUCT` } ] [ `LINES`= { `24` } ] [ `KEYBOARD`= { `STANDARD` } ] [ `INTENSITY`= { `NORMAL` } ]  
[ `NONDESTRUCT` } ] [ `12` ] [ `KATAKANA` } ] [ `LOW` } ]  
[ `REVERSE` } ]  
[ `LOG`= { `ALL` } ] [ `CENTRAL` } ] [ `NONBURST` } ] [ `CONTINUOUS` } ] [ `SI`= { `TOP` } ]  
[ `COMMANDS` } ] [ `WKSTN` } ] [ `BURST` } ] [ `PAGE` } ] [ `BOTTOM` } ]

The SCREEN command enables you to alter certain characteristics of the workstation or terminal you're using. You may use the SCREEN command to change different characteristics on different terminals acting as workstations; consult the interactive services commands and facilities user guide/programmer reference, UP-9972 (current version) for further information.

---

## STATUS Command

STATUSΔ { TERMINALS  
          { RESOURCES  
          JOBS  
          { FUNCTIONS  
          VOLUMES  
          LIMITS }

The STATUS command enables you to obtain information about various aspects of your system, including usage of terminals and workstations, numbers of jobs running on the system, system main storage resources in use and available, jobs running under your user-id, and which interactive commands and facilities are executing under your user-id. STATUS can also give you a listing of the disk, tape, and diskette volumes currently mounted on your system.

---

---

## TELL Command

TELLΔ [ {user-id } , ] 'text'  
          { ALL }

The TELL command enables you to send messages that do not require a response to the system operator or other workstation users.

---

---

## VTOC Command

VTOCΔ [ 'file-prefix' , ] VSN=volumeΔ [ FREE ]

The VTOC command enables you to produce a listing of the files present on a disk or diskette volume. You may list all the files on a volume, or only those whose file names begin with a prefix you specify. You may also obtain a listing of each free extent available and its size.

---

## INTERACTIVE DATA UTILITIES

To initialize the interactive data utilities, enter the following command:

```
RV△I@DATA[(new-name)][, ,MEM=nnnnn,][ACT=act-no,][DBG={Y}
{N}]
```

- (new-name) Permits the concurrent use of data utilities.
- MEM=nnnnn Specifies, in hexadecimal notation, the minimum main storage needed to run your data utilities job. Default is 8000<sub>16</sub> (32,767<sub>10</sub>).
- ACT=act-no Specifies a 1- to 4-character alphanumeric account number.
- DBG={Y}
{N} Specifies that data utilities run in the debugging mode, used to provide documentation for reporting a software user report (SUR).

---

## JOB CONTROL DIALOG

To initialize the job control dialog, enter the following command:

**RV△JC\$BLD**

---

---

## SCREEN FORMAT GENERATOR

Use the following command to initialize the screen format generator:

**RV△SFGEN**

There are no parameters associated with this command.

---

## USING SCREEN FORMATS IN YOUR PROGRAM

Use the following job control statement to include formatted screen displays in your programs:

```
//[symbol]USE SFS [, {format-file-lfd-1/format-file-lfd-2}] [, initial-screen] [, {nnn}]
 {format-file-id }
 SYSEMT
[, screen-format-1=alias-1... , screen-format-12=alias-12]
```

{format-file-id } Specifies the LFD name(s) of the screen format file.  
**SYSEMT**

initial-screen Specifies the first format name to be used in behalf of the user program.

{nnn } Specifies the number of formats to reside in main storage for use with a given file.

[ , screen-format-1=alias-1... [ , screen-format-12=alias-12 ] ]  
Specifies that a name other than the real name of the format is to be used. Maximum of 12 aliases permitted. More than 12 causes rejection of the job control stream.

---

## MENU GENERATOR

Use the following command to initialize the menu generator:

**MENUGEN**

There are no parameters associated with this command.

---



## USING MENUS IN YOUR PROGRAM

Use the following job control statement to include menus in your program:

```
//[symbol]USE MENU { menu-file-lfd-1/menu-file-lfd-2 } [,initial-menu]
 { SYSFMT/menu-file-lfd-2
 { menu-file-lfd-1/SYSFMT
 { menu-file-lfd-1
 { SYSFMT
[, { nnn }] [,menu-1=alias-1...,menu-12=alias-12]
[1]
```

(continued)

## USING MENUS IN YOUR PROGRAM (cont)

|   |                                 |   |                                                                                                       |
|---|---------------------------------|---|-------------------------------------------------------------------------------------------------------|
| { | menu-file-lfd-1/menu-file-lfd-2 | } | Specifies the lfd name or names<br>for up to two files that will be<br>searched for the correct menu. |
|   | <b>SYSFMT</b> /menu-file-lfd-2  |   |                                                                                                       |
|   | menu-file-lfd-1/ <b>SYSFMT</b>  |   |                                                                                                       |
|   | menu-file-lfd-1                 |   |                                                                                                       |

`initial-menu` Specifies the name of the first or only menu to be used by a program.

`{nnn}` Specifies the number of menus to reside in main storage for use with a given file.  
`{1}`

`[,menu-1=alias-1...[,menu-12=alias-12]`  
Specifies that a name other than the real name of the menu is to be used. Maximum of 12 aliases permitted. More than 12 causes rejection of the job control stream.

---

## INTERACTIVE DUMP/RESTORE HARDWARE UTILITY

The dump/restore hardware utility lets you interactively initiate and control the DMPRST routine from your workstation. The DMPRST routine creates backup copies of your program and data libraries on disk, tape (including streaming tape), or diskette.

To initialize the interactive dump/restore hardware utility, enter the following command:

**HU**

There are no parameters associated with this command.

---

---

## BASIC

BASIC is an interactive programming language you can use from your workstation. To initialize BASIC, enter the following command:

### BASIC

There are no parameters associated with this command.

---

---

## ESCORT

ESCORT is an interactive programming language that uses English statements to create a program. The ESCORT language allows you to generate reports and perform inquiry and update routines through the use of simple, sentence-like programs, entered through your workstation. To use ESCORT, log on and enter the following command:

### ESCORT

There are no parameters associated with this command.

---

## GENERAL EDITOR COMMANDS

The following is a listing of the commands used by the SPERRY Operating System/3 General Editor.

| Command                   | Format                                                    | Explanation                                                                                           |
|---------------------------|-----------------------------------------------------------|-------------------------------------------------------------------------------------------------------|
| EDT Commands              |                                                           |                                                                                                       |
| @                         | @ {line-number [increment]} [:{data<br>+<br>-} {command}] | Sets the current line number and increment for data and command lines keyed in at the workstation     |
| <u>C</u> HANGE            | @C ['search-string'[*n]]TO 'change-string'[*n]            | Replaces an existing string in the current work-space file with a new string                          |
| <u>C</u> OPY<br><u>--</u> | @CO [line-range]['search-string'[*n]]<br>TO destination   | Copies lines in the current work-space file to new line locations without deleting the original lines |

(continued)

---

EDT Commands (cont)

---

DELETE    @D [line-range]['search-string'[\*n]]

Erases specified lines  
and their line numbers  
from the current work-space  
file

FIND        @FIN 'search-string'[\*n]

Locates the first occurrence  
of a string in the work-space  
file and assigns its corres-  
ponding line number to the  
variable ? and the column  
numbers of the first and last  
columns it occupies to  
[ and ] respectively

---

(continued)



---

EDT Commands (cont)

---

FSTATUS @FSA[MO=module-name] [ ,TYPE= { module-type } ]

Creates in the work-space file a list of all modules contained in a specified program library

,FILENAME= { filename  
                  'filename'  
                  "filename" }

[ ,RDPASS=password ],VSN=volume [ ,DEVICE= { did  
                                  DISK  
                                  DISKETTE } ]

(continued)

---

EDT Commands (cont)

---

|                |                                                           |                                                                                                                        |
|----------------|-----------------------------------------------------------|------------------------------------------------------------------------------------------------------------------------|
| <u>I</u> NSERT | @I 'change-string'[*n]                                    | Inserts a specified string into lines in the current work-space file                                                   |
| <u>L</u> IST   | @L [line-range]['search-string'[*n]]△[ <u>I</u> MMEDIATE] | Prints specified lines from the current work-space file on the printer                                                 |
| <u>M</u> OVE   | @M [line-range]['search-string'[*n]]<br>TO destination    | Transfers specified lines to new line locations in the work-space file and deletes the original lines and line numbers |
| <u>N</u> UMBER | @NU 'sequence-string'[*n][BY increment]                   | Inserts sequence numbers into input lines                                                                              |

---

(continued)

---

EDT Commands (cont)

---

|                |                                                            |                                                                                     |
|----------------|------------------------------------------------------------|-------------------------------------------------------------------------------------|
| <u>P</u> PRINT | @P [line-range][search-string[*n]]                         | Displays specified lines from the current work-space file on the workstation screen |
| <u>P</u> PUNCH | @PU [line-range]['search-string'[*n]]△[ <u>I</u> MMEDIATE] | Reproduces specified lines from the current work-space file on cards                |

---

(continued)

---

EDT Commands (cont)

---

READ

@R△MODULE=module-name, FILENAME= { filename  
                                          'filename'  
                                          "filename" }

Reads a SAT or MIRAM  
library module from disk  
or format label diskette

[, TRUNC= { YES  
              NO } ] [, RDPASS=password], VSN=volume

[, TYPE= { module-type  
              S } ] [, DEVICE= { did  
                                  DISK  
                                  DISKETTE } ]

△ [ { KEY=start-col-no:end-col-no  
      KKEY=start-col-no:end-col-no  
      SHOW△first-col-no:last-col-no } ]

---

(continued)

---

EDT Commands (cont)

---

READ

@RΔFILENAME= { filename  
'filename'  
"filename" } [, RDPASS=password]

Reads a MIRAM data  
file from disk or  
format label diskette

, VSN=volume [ , KEYNO= { n  
0 } ] [ , DEVICE= { did  
DISK  
DISKETTE } ]

[ , BFSZ=n ] [ , TRUNC= { YES  
NO } ]

Δ { KEY=start-col-no:end-col-no  
KKEY=start-col-no:end-col-no  
SHOWΔfirst-col-no:last-col-no }

---

(continued)

EDT Commands (cont)

READ

@R△[JOB=jobname] [ ,HOLD= $\left\{ \begin{array}{l} \text{LOG} \\ \text{NO} \\ \text{YES} \end{array} \right\} ]$

Reads a file from  
the spool file

[ ,FILENAME= $\left\{ \begin{array}{l} \text{filename} \\ \text{'filename' } \\ \text{"filename"} \end{array} \right\} , \underline{QUEUE}= $\left\{ \begin{array}{l} \text{LOG} \\ \text{PRINT} \\ \text{PUNCH} \\ \text{RDR} \end{array} \right\} [ , \underline{ACCT}=acct-no ] [ ,ALL= $\left\{ \begin{array}{l} \text{NO} \\ \text{YES} \end{array} \right\} ]$$$

[ ,SKIP= $\left\{ \begin{array}{l} \text{0} \\ \text{n} \end{array} \right\} ] [ , \underline{TRUNC}= $\left\{ \begin{array}{l} \text{YES} \\ \text{NO} \end{array} \right\} ] \Delta \left[ \left\{ \begin{array}{l} \underline{KEY}=start-col-no:end-col-no \\ \underline{KKEY}=start-col-no:end-col-no \\ \underline{SHOW}△first-col-no:last-col-no \end{array} \right\} \right]$$

(continued)

EDT Commands (cont)

READ

@R△FILENAME= { filename  
'filename'  
"filename" }, VSN=volume

Reads a unit record file  
from a data set label  
diskette or card reader

,DEVICE= { did  
DISKETTE  
RDR } [ ,TRUNC= { YES } ] Δ [ { KEY=start-col-no:end-col-no  
KKEY=start-col-no:end-col-no  
SHOWΔfirst-col-no:last-col-no } ]

@R△FILENAME= { filename  
'filename'  
"filename" } [ ,RDPASS=password ]

Reads a file from  
a tape

, VSN=volume, DEVICE= { did  
TAPE } [ ,BKNO= { YES } ] [ ,TRUNC= { YES } ]

Δ [ { KEY=start-col-no:end-col-no  
KKEY=start-col-no:end-col-no  
SHOWΔfirst-col-no:last-col-no } ]

(continued)

---

EDT Commands (cont)

---

READ      @R      Reads the same module or file last accessed through a previous @READ or @WRITE command

---

READ      @RΔ;Δ [ { KEY=start-col-no:end-col-no  
                  KKEY=start-col-no:end-col-no  
                  SHOWΔfirst-col-no:last-col-no } ]  
                  [valid EDT command]

Reads the same module or file last accessed through a previous @READ or @WRITE command but read now with a previous KEY, KKEY, or SHOW parameter or any valid EDT command specified

---

(continued)



---

EDT Commands (cont)

---

REMOVE @REM 'search-string'[\*n]

Deletes a specified string from lines in the work-space file

SEQUENCE @SEQ { 'sequence-string'[\*n] } BY increment  
                  {\*}

Inserts sequence numbers into existing lines in the current work-space file

UPDATE @U [line-range]['search-string'[\*n]]

Displays specified lines from the work-space file one at a time for you to edit or change

---

(continued)

---

EDT Commands (cont)

---

WRITE

@WΔMODULE=module-name [ ,TYPE={ module-type } ]

Writes a SAT or MIRAM library module to disk or format label diskette

[ ,FILENAME={ filename } [ ,WRPASS=password ] [ ,DEVICE={ did } ]

{ did  
DISK  
DISKETTE }

[ ,VSN=volume [ ,CONTIG={ NO } ] [ ,INC={ n } ] [ ,RCSZ=n ]

{ YES }

{ 1 }

[ ,SIZE=n ] [ ,SAT={ NO } ]

{ YES }

(continued)

EDT Commands (cont)

WRITE

@WΔFILENAME= $\left\{ \begin{array}{l} \text{filename} \\ \text{'filename'} \\ \text{"filename"} \end{array} \right\}$

Writes a MIRAM data file to disk or format label diskette

[, WRPASS=password], VSN=volume

$\left[ \text{, } \underline{\text{CONTIG}} = \left\{ \begin{array}{l} \text{NO} \\ \text{YES} \end{array} \right\} \right] \left[ \text{, } \underline{\text{INC}} = \left\{ \begin{array}{l} \text{n} \\ \text{1} \end{array} \right\} \right] \left[ \text{, } \underline{\text{INIT}} = \left\{ \begin{array}{l} \text{NO} \\ \text{YES} \end{array} \right\} \right] \left[ \text{, } \underline{\text{RCB}} = \left\{ \begin{array}{l} \text{YES} \\ \text{NO} \end{array} \right\} \right] \left[ \text{, } \underline{\text{RCFM}} = \left\{ \begin{array}{l} \text{VAR} \\ \text{FIX} \end{array} \right\} \right]$

, RCSZ=n  $\left[ \text{, } \underline{\text{SCSZ}} = \left\{ \begin{array}{l} \text{n} \\ \text{256} \end{array} \right\} \right]$

$\left[ \text{, } \underline{\text{DEVICE}} = \left\{ \begin{array}{l} \text{did} \\ \text{DISK} \\ \text{DISKETTE} \end{array} \right\} \right] \left[ \text{, } \underline{\text{SIZE}} = \text{n} \right] \left[ \text{, } \underline{\text{EXTEND}} = \left\{ \begin{array}{l} \text{YES} \\ \text{NO} \end{array} \right\} \right] \left[ \text{, } \underline{\text{BFSZ}} = \text{n} \right]$

$\left[ \text{, } \underline{\text{KEY}} \text{i} = \left\{ \begin{array}{l} \text{start-col-no: end-col-no} \\ \left( \text{start-col-no: end-col-no}, \left\{ \begin{array}{l} \text{DUP} \\ \text{NDUP} \end{array} \right\}, \left\{ \begin{array}{l} \text{CHG} \\ \text{NCHG} \end{array} \right\} \right) \end{array} \right\} \right]$

(continued)

---

EDT Commands (cont)

---

WRITE

@W△[JOB=jobname][HOLD= $\left\{ \begin{array}{l} \text{NO} \\ \text{YES} \end{array} \right\}$ ]

Writes a file to the  
the spool file

[FILENAME= $\left\{ \begin{array}{l} \text{filename} \\ \text{'filename'} \\ \text{"filename"} \end{array} \right\}$ ]

[ACCT=acct-no],QUEE= $\left\{ \begin{array}{l} \text{PRINT} \\ \text{PUNCH} \\ \text{RDR} \end{array} \right\}$ ][COPIES= $\left\{ \begin{array}{l} n \\ 1 \end{array} \right\}$ ]

---

(continued)

EDT Commands (cont)

WRITE    @W△FILENAME= { filename } ,VSN=volume  
                               'filename'  
                               "filename"  
               , DEVICE= { did } [ ,RCFM= { **FIX** } [ ,RCSZ=n ]  
                               DISKETTE            { VAR }  
                               PRINT  
                               PUNCH

Writes a unit record file to a printer, punch, or diskette

WRITE    @W△FILENAME= { filename } [ ,WRPASS=password ]  
                               'filename'  
                               "filename"  
               , VSN=volume, DEVICE= { did } [ ,BFSZ=n ] [ ,INIT= { YES } ] [ ,EXTEND= { YES } ]  
                                           TAPE            { NO }            { NO }  
               [ ,BKNO= { YES } ] [ ,RCFM= { **FIXUNB** } [ ,RCSZ=n ]  
                                                           { FIXBLK  
                                                           VARBLK  
                                                           VARUNB  
                                                           UNDEF

(continued)

---

EDT Commands (cont)

---

WRITE     @W

Writes to the same module or file last accessed through a previous @READ or @WRITE command

---

WRITE     @WΔ;Δvalid EDT command

Writes to the same module or file last accessed through a previous @READ or @WRITE command but written now with any valid EDT command specified

---

---

EDT Variable Commands

---

ASSIGN    @AS△Gn=  $\left. \begin{array}{l} \text{'string'[*n]} \\ n(x:y) \\ n[\pm m] \\ Gm \\ \text{LEN}(n) \end{array} \right\}$     Assigns values to EDT variables

---

DISPLAY    @DI△  $\left. \begin{array}{l} \text{'string'[*n]} \\ n(x:y) \\ n[\pm m] \\ Gm \\ \text{LEN}(n) \end{array} \right\}$     Displays a specified expression or the value of a specified expression from the work-space file on the workstation screen

---

IF    @IF.condition.command  
or  
@IF expression relation expression command

---

General Editor Procedure File Commands

|              |                                                                                                                                |                                           |
|--------------|--------------------------------------------------------------------------------------------------------------------------------|-------------------------------------------|
| <u>DO</u>    | @DO proc-number $\left[ \begin{array}{l} \text{PRINT} \\ \text{NOPRINT} \\ \text{REVERT} \\ \text{—} \end{array} \right]$      | Executes a procedure file                 |
| <u>END</u>   | @E                                                                                                                             | Terminates procedure file definition      |
| <u>GOTO</u>  | @G^ $\left\{ \begin{array}{l} \text{line} \\ \text{label} \end{array} \right\}$                                                | Permits branching within a procedure file |
| <u>INPUT</u> | @INP△file-parameters $\left[ \begin{array}{l} \text{PRINT} \\ \text{NOPRINT} \\ \text{REVERT} \\ \text{—} \end{array} \right]$ | Loads and executes a procedure file       |

(continued)



---

General Editor Procedure File Commands (cont)

---

|               |                    |                                                                    |
|---------------|--------------------|--------------------------------------------------------------------|
| <u>NOP</u>    | @NOP△[comment]     | Enters extra lines for branching or comments into a procedure file |
| <u>PROC</u>   | @PRO [proc-number] | Begins procedure file definition                                   |
| <u>RETURN</u> | @RET               | Terminates procedure file execution                                |

---

---

General Editor Directives

---

|                                                       |                            |                                                                                  |
|-------------------------------------------------------|----------------------------|----------------------------------------------------------------------------------|
| <u>C</u> <u>C</u> <u>H</u> <u>E</u> <u>C</u> <u>K</u> | @CHE△<br>[ OFF ]<br>[ ON ] | Determines whether processed lines are to be displayed on the workstation screen |
| <u>C</u> <u>O</u> <u>B</u> <u>O</u> <u>L</u>          | @COB                       | Activates the COBOL editor                                                       |
| <u>D</u> <u>R</u> <u>O</u> <u>P</u>                   | @DR                        | Deletes all lines in the entire EDT work-space file                              |
| <u>E</u> <u>F</u> <u>F</u>                            | @EFP                       | Activates the error file processor                                               |

---

(continued)

---

General Editor Directives (cont)

---

|             |                                                               |                                                                                                                                                                                                                                                        |
|-------------|---------------------------------------------------------------|--------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| FORMAT      | @FORMAT parameter-string (for RPGEDT)<br>@FORMAT (for COBEDT) | Used only in conjunction with either RPGEDT or COBEDT. For information on the @FORMAT directive, see the current version of the RPG II editor user guide/ programmer reference, UP-8803 or the COBOL editor user guide/ programmer reference, UP-9106. |
| <u>HALT</u> | @H                                                            | Terminates the EDT session                                                                                                                                                                                                                             |
| RPG         | @RPG                                                          | Activates the RPG II editor                                                                                                                                                                                                                            |

---

---

General Editor Directives (cont)

---

SET (a)S△ [ CHAR=tab-character, TABS={columns} ]  
                        { OFF }  
[ , LINE=length ] [ , EXCLUDE={exclusion-character} ]  
                        { OFF }  
[ , ATSIGN=command-trigger ] [ , COLON=range-separator ]  
  
[ , ENCOL=end-column ] [ , BUFFER={record-size} ]  
                        { OFF }  
  
[ , WIDTH=device-size ] [ , CLEAR ] [ , STRIP={ON } ] [ , DISPLAY ]  
                        { OFF }  
  
[ , SCRDSPLY={ TRUNCATE } ] [ , ROLL={ 15 (if SCRDSPLY=TRUNCATE) } ]  
                        { FOLD }                                    { 8 (if SCRDSPLY=FOLD) }  
                                                                                        { 1-15 } ] ]

Defines various parameters  
to EDT that collectively  
make up your EDT environment

(continued)

---

General Editor Directives (cont)

---

[, MODE = { LINE  
SCREEN } ] [ , LANGUAGE = { FREEFORM  
FORTRAN  
COBOL  
RPG } ] [ , RECENTRY = { SINGLE  
MULT } ]

[ , SCRFORM = { UNDERLINE  
BLANK } ]

---

SYSTEM @SY△[workstation-command]

Permits workstation commands to be issued during an EDT session or temporarily returns you to system mode

---

---

## General Editor Screen Commands

---

|                |                          |                                                                                                            |
|----------------|--------------------------|------------------------------------------------------------------------------------------------------------|
| <u>B</u> LOCK  | @BL                      | Displays a free-form screen that allows you to switch to block mode for entering multiple commands or data |
| <u>H</u> ELP   | @HE△[error message code] | Displays help screens for any EDT error messages                                                           |
| <u>P</u> ARAMS | @PA                      | Displays a screen showing the parameters on the @SET directive (those that make up your EDT environment)   |

---

(continued)

---

General Editor Screen Commands (cont)

---

PROMPT    @PROM△[EDT command]

Displays the EDT command menu screen or help screens for any EDT command (meaning EDT commands, modifiers, directives, procedure file commands, variables, and screen commands)

---

RESTORE    @RES

Returns you to the point in your EDT session where you originally entered a screen command

---

ROLL    @RO

Displays free-form screens showing the EDT work-space file, where you can update lines or simply view them

---

---

## EFP Commands

---

EFP

To correct and display COBOL and RPG II errors and FORTRAN IV errors for one source module at a time, use:

@EF[X]△[program-unit-name]△  
[error-range]△['search-string']

To correct and display FORTRAN IV errors for compilations that process multiple source modules, use:

@EF△SOURCE△source-module-name,source-file-name,vsn

Displays errors in your error file along with the source lines that contain those errors. Note that EFP is both an EDT directive and an EFP command.

---

(continued)





## RPG II EDITOR

## COBOL EDITOR

The RPG II and COBOL editors are actually subeditors of EDT. Therefore, you must first activate the General Editor before you can use a language editor. Once in a language editor session you can use any of the EDT commands.

## RPG II EDITOR

To activate the RPG II editor, key in the following command:

EDT△@RPG

The following are commands you use in conjunction with the RPG II editor:

| Command       | Format                             | Explanation                                                                                                           |
|---------------|------------------------------------|-----------------------------------------------------------------------------------------------------------------------|
| <u>FORMAT</u> | @FO specification-type             | Allows you to change the display format type and/or switch the RPG II editor from the update mode to the create mode. |
|               | @FO specification-type,format-type |                                                                                                                       |
|               | @FO ,format-type                   |                                                                                                                       |
|               | @FO ,format-type,CMD               |                                                                                                                       |
| <u>HALT</u>   | @H                                 | Terminates the RPG II (and EDT) session.                                                                              |

---

## COBOL EDITOR

To activate the COBOL editor, key in the following command:

EDT△@COBOL

The following are commands you use in conjunction with the COBOL editor:

| Command       | Format         | Explanation                                    |
|---------------|----------------|------------------------------------------------|
| <u>FORMAT</u> | @F0 parameters | Changes the display format type.               |
| <u>HALT</u>   | @H             | Terminates the COBOL editor (and EDT) session. |

---

## DDP COMMANDS

The following is a list of special commands to perform DDP functions.

### Command

### Explanation

#### DDP Commands

DDP△CREATE△FILE={ { host-id } :: } file-id  
                          { local-host-id }

[△BLOCK\_SIZE={ nnnnnnnnn } ] [△DENSITY={ 200  
                                          556  
                                          800  
                                          1600  
                                          6250  
                                          host-SYSGEN-option } ]

The CREATE command:

- Establishes a file on a receiving host
- Allocates space for the file
- Catalogs the file in your online system catalog
- Records the file in the volume table of contents (VTOC) of the volume at the remote host on which the file is created

(continued)

DDP Commands (cont)

$\left[ \begin{array}{l} \Delta \text{DEVICE\_CLASS} = \left\{ \begin{array}{l} \text{DISK} \\ \text{TAPE} \\ \text{DISKETTE} \end{array} \right\} \end{array} \right] \left[ \begin{array}{l} \Delta \text{FILE\_TYPE} = \left\{ \begin{array}{l} \text{SEQUENTIAL} \\ \text{INDEXED} \\ \text{LIBRARY} \\ \text{UNDEFINED} \end{array} \right\} \end{array} \right]$

$\left[ \begin{array}{l} \Delta \text{INCREMENT\_SIZE} = \left\{ \begin{array}{l} \text{nnnnnnnnn} \\ \text{3 cyl} \end{array} \right\} \end{array} \right] \left[ \begin{array}{l} \Delta \text{INITIAL\_SIZE} = \left\{ \begin{array}{l} \text{nnnnnnnnn} \\ \text{3 cyl} \end{array} \right\} \end{array} \right]$

$\left[ \begin{array}{l} \text{KEY} \left[ \begin{array}{l} - \{n\} \\ - \{1\} \end{array} \right] = (\text{size, location}) \left( \begin{array}{l} \left[ \begin{array}{l} \Delta \left\{ \begin{array}{l} \text{DUPLICATES} \\ \text{NO\_DUPLICATES} \end{array} \right\} \right] \\ \left[ \begin{array}{l} \Delta \left\{ \begin{array}{l} \text{CHANGE} \\ \text{NO\_CHANGE} \end{array} \right\} \right] \end{array} \right) \end{array} \right]$

$\left[ \begin{array}{l} \Delta \text{PARITY} = \left\{ \begin{array}{l} \text{ODD} \\ \text{EVEN} \end{array} \right\} \end{array} \right] \left[ \begin{array}{l} \Delta \text{RECORD\_FORM} = \left\{ \begin{array}{l} \text{FIXED} \\ \text{VARIABLE} \\ \text{UNDEFINED} \end{array} \right\} \end{array} \right]$

NOTE:

*The default for INCREMENT SIZE and INITIAL SIZE is three cylinders. If more or less than three cylinders is needed, the size must be entered in number of blocks (nnnnnnnnn).*

(continued)

---

DDP Commands (cont)

---

[ $\Delta$ RECORD\_SIZE={nnnnn}] [ $\Delta$ REGISTER={VTOC  
CATALOG}]

---

DDP $\Delta$ COPY $\Delta$ FROM= [{originating-host-id}::  
local-host-id] originating-file-id

$\Delta$ TO= [{destination-host-id}::  
local-host-id] destination-file-id

[ $\Delta$ ELEMENT\_TYPE={SYMBOLIC  
RELOCATABLE  
ABSOLUTE  
MACRO  
PROC  
COMPILED\_JOB  
SCREEN\_FORMAT}]

The COPY command permits you to copy a file or module from one system to another. You may copy a file or module from one remote system to another, from your local system to a remote system, or vice versa. You may also use the COPY command to copy a file within your local system.

(continued)

DDP Commands (cont)

[KEY [-{n}] = /size, location [Δ{DUPLICATES  
NO DUPLICATES}]]  
[Δ{CHANGE  
NO CHANGE}]

[MODE = {DIRECT  
WAIT  
INDIRECT}] [ΔPOSITION = {EOF  
SOF}]

[ΔTRANSLATE = {ASCII  
EBCDIC  
NONE}]

DDPΔPURGEΔFILE = [{host-id  
local-host-id}]::] file-id

The PURGE command allows you to physically remove a file, and all references to it, from a host system. (continued)



---

DDP Commands (cont)

---

DDP△SUBMIT△FILE={ {originating-host-id} :: } file-id  
                  { local-host-id }

[△ELEMENT\_TYPE={ SYMBOLIC  
                  { COMPILED\_JOB } }]

[△HOST={ destination-host-id }  
          { local-host-id }]

The SUBMIT command allows you to send a file of job control streams to a host system for execution. You can also use it to initiate a file of job control streams already at the host system or to bring a job control stream to your local system for execution.

---

DDP△CANCEL△JOB={ {host-id } :: } jobname  
                  { local-host-id }

[△OUTPUT={ DISCARD } ]△COMMAND={ host-id  
                                  { local-host-id }

The CANCEL command allows you to terminate a job either executing or scheduled for execution on a host system.

---

DDP△SUBMIT△REQUEST=statement [△HOST={ host-id  
                                  { local-host-id } }]

The SUBMIT REQUEST command allows you to send a statement, such as an operator

(continued)

---

## DDP Commands (cont)

---

or interactive command, to a host system. The following statements (commands) cannot be used: DISPLAY, DELETE, BREAKPOINT, FILE, IN, SU, TU, PD.

---

```
DDP△STATUS△ { COMMAND=work-order-number
 FILE=[{host-id }::]file-id;
 [{local-host-id }]
 [keyword parameter]
 HOST=host-id
 JOB= host-id }::]jobname
 {local-host-id }
 USER=[{host-id }::]user-id
 [{local-host-id }]
 }
```

The STATUS command enables you to obtain information about:

- Commands entered
- Host systems in your DDP system
- Jobs you have submitted
- Files in your DDP system
- Other users on your DDP system

---

DDP Commands (cont)

---

DDP△TALK△MESSAGE='string'

△USER= [ { host-id } :: { OPERATOR } [△WAIT]  
          { local-host-id } { user-id } ]

---

The TALK command allows you to send a message to a remote operator or user.

## INTERACTIVE TERMINALS

OS/3 interactive services can be accessed through four types of terminals: the workstation (UTS 20/20D or UTS 40/40D), the System 80 console workstation, the UNISCOPE 100 and 200, and the UTS 400. The following table shows the different procedures used to access interactive services through these different types of terminals:

*NOTE:*

*You do not need to enter the \$\$\$SON command to log onto either a workstation or the System 80 console workstation. However, some system programs such as the Information Management System (IMS) require the use of ICAM to connect to workstations and terminals. In such cases, log on normally and then enter \$\$\$SON to sign onto IMS or other similar programs.*

| Procedure        | Workstation                       | UNISCOPE Terminals                                                                               | UTS 400                                                            | System 80 Console Workstation     |
|------------------|-----------------------------------|--------------------------------------------------------------------------------------------------|--------------------------------------------------------------------|-----------------------------------|
| \$\$\$ON         | NO                                | YES                                                                                              | YES                                                                | NO                                |
| LOGON            | YES                               | YES                                                                                              | YES                                                                | YES                               |
| SYSTEM mode      | Press FUNCTION and SYS MODE keys. | Press MESSAGE WAITING key.                                                                       | Press MSG WAIT key.                                                | Press FUNCTION and SYS MODE keys. |
| WORKSTATION mode | Press FUNCTION and WS MODE keys.  | Press MESSAGE WAITING key. Then press W then S keys; then press TRANSMIT.                        | Press MSG WAIT key. Then press W then S keys; then press TRANSMIT. | Press FUNCTION and WS MODE keys.  |
| Function keys    | Press FUNCTION and F1-F22 keys.   | Press MESSAGE WAITING key. Then press F1-F4; for rest, press F, the pound symbol (#), then 5-22. | Press F1-F22 UPPER FUNCTION when required.                         | Press FUNCTION and F1-F22 keys.   |

(continued)

| Procedure                       | Workstation                         | UNISCOPE Terminals                                                                                         | UTS 400                                                    | System 80 Console Workstation       |
|---------------------------------|-------------------------------------|------------------------------------------------------------------------------------------------------------|------------------------------------------------------------|-------------------------------------|
| Function keys (cont.)           |                                     | To simulate the MESSAGE WAITING key, enter system mode by pressing the MESSAGE WAITING key, then type MSG. |                                                            |                                     |
| Message waiting indicator       | SYS MSG displayed on indicator line | MESSAGE WAITING light is lit and the audible alarm sounds.                                                 | MESSAGE WAITING light is lit and the audible alarm sounds. | SYS MSG displayed on indicator line |
| Save and restore SYS mode lines | YES                                 | NO                                                                                                         | NO                                                         | YES                                 |
| LOGOFF                          | YES                                 | YES                                                                                                        | YES                                                        | YES                                 |
| \$\$\$OFF                       | NO                                  | YES                                                                                                        | YES                                                        | NO                                  |

## FUNCTION KEY SUMMARY

This summary provides an overview of the function keys used by interactive services and the various interactive facilities.

| Software Component   | Key | Function                                                                                                                                                         |
|----------------------|-----|------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| Interactive services | F15 | Informs the system you have no more data to input from the workstation (end of file)                                                                             |
|                      | F17 | Temporarily halts the workstation display. (See F19.)                                                                                                            |
|                      | F19 | Restarts workstation display after it has been temporarily stopped using F17 or when the screen is full of data                                                  |
| General editor       | F1  | Suppresses any printing options associated with a command. (Same as F18)                                                                                         |
|                      | F2  | Terminates processing of a command                                                                                                                               |
|                      | F3  | Displays a screen showing the parameters on the @SET directive, or those that make up your EDT environment. F3 is the same as issuing the PARAMS screen command. |

| Software Component    | Key | Function                                                                                                                                                                                                                                         |
|-----------------------|-----|--------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| General editor (cont) | F4  | Displays a freeform screen through which you can switch to block mode for entering multiple commands or data. F4 is the same as issuing the BLOCK screen command.                                                                                |
|                       | F5  | Displays freeform screens, showing the EDT work-space file where you can update lines or simply view them. F5 is the same as issuing the ROLL screen command.                                                                                    |
|                       | F6  | Displays help screens for any EDT error messages. F6 is the same as issuing the HELP screen command.                                                                                                                                             |
|                       | F12 | Shows a previously displayed additional help screen for a specific command when that command requires several help screens to fully describe it.                                                                                                 |
|                       | F13 | Displays the EDT command menu screen and help screens for any of the EDT commands. F13 is the same as issuing the PROMPT screen command. From a help screen, F13 also lets you see subsequent help screens needed to fully describe the command. |
|                       | F14 | Returns you to the point in your EDT session where you originally entered a screen command. F14 is the same as issuing the RESTORE screen command.                                                                                               |



| Software Component     | Key | Function                                                                                                                                                                                                                                                                                           |
|------------------------|-----|----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| General editor (cont)  | F15 | Recognized as EOF indicator. Halts an EDT session run in batch mode or produces an error message in interactive mode.                                                                                                                                                                              |
|                        | F18 | Suppresses printing option associated with a command. (Same as F1)                                                                                                                                                                                                                                 |
|                        | F19 | Restarts workstation display after it has been temporarily stopped using F17 or when the screen is full of data                                                                                                                                                                                    |
| Screen format services | F1  | Returns the screen format generator to the HOME screen. Returning home deletes all work done up to that point.                                                                                                                                                                                     |
|                        | F5  | Causes the screen format generator to breakpoint the spool file to a printer. All records currently in the spooled printer output are directed to a printer. Records added to the spool file after the breakpoint are held until a subsequent breakpoint is requested or until the end of the job. |
|                        | F13 | Displays a HELP screen appropriate to the step you're at when generating a screen format. (See F14.)                                                                                                                                                                                               |

| Software Component            | Key | Function                                                                                                                                                                                                                                                                                                                                                                                                                                                              |
|-------------------------------|-----|-----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| Screen format services (cont) | F14 | Removes the HELP screen displayed by using F13; returns the screen format generator to the point it was at before display of the HELP screen.                                                                                                                                                                                                                                                                                                                         |
|                               | F15 | Indicates end of input data                                                                                                                                                                                                                                                                                                                                                                                                                                           |
|                               | F16 | Indicates input data cannot be entered properly                                                                                                                                                                                                                                                                                                                                                                                                                       |
|                               | F20 | Restores the screen to its original contents for the current pass if it has inadvertently been destroyed.                                                                                                                                                                                                                                                                                                                                                             |
| BASIC                         | F1  | <p>Pauses or terminates execution of a BASIC program. When no I/O operation is in progress, the system displays the message:</p> <p style="text-align: center;">EXECUTION PAUSED AT LINE xxxx CONTINUE (Y,N)?</p> <p>Key in Y to continue (resume) execution. Key in N to terminate the program.</p> <ul style="list-style-type: none"> <li>■ If a BASIC program is requesting output, you must press XMIT after pressing F1 to display the above message.</li> </ul> |

| Software Component | Key | Function                                                                                                                                                                                                                                                         |
|--------------------|-----|------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| BASIC<br>(cont)    | F1  | <ul style="list-style-type: none"> <li>■ If the BASIC program is sending output, the workstation screen must be full of data for F1 to be recognized. Press F19 to display the above message.</li> </ul>                                                         |
|                    | F19 | See F1.                                                                                                                                                                                                                                                          |
| ESCORT             | F1  | Signals the end of input and returns to master menu. (Used only with the structure processor)                                                                                                                                                                    |
|                    | F2  | Cancels display output. (Used only with the structure processor)                                                                                                                                                                                                 |
|                    | F3  | Cancels the current screen and returns to the previous screen. (Used only for program and tutorial modes)                                                                                                                                                        |
|                    | F4  | <ul style="list-style-type: none"> <li>■ Structure processor: aborts structure and returns to menu</li> <li>■ Program mode: ends free-form input and returns to previous menu</li> <li>■ Run-time processor: terminates program and returns to caller</li> </ul> |

# COMMAND INDEX

This index alphabetically lists each interactive services command and its page number.

| Command  | Page No. | Command      | Page No. | Command           | Page No. |
|----------|----------|--------------|----------|-------------------|----------|
| ALLOCATE | 33       | BRKPT        | 30       | COMMENT           | 36       |
| ASK      | 35       | BRKPT LOG    | 35       | CONNECT           | 21       |
| BASIC    | 79       | CANCEL       | 20       | COPY              |          |
| BEGIN    |          | CHANGE       | 17       | library modules   | 37       |
| JBQ      | 15       | CHANGE SPOOL | 29a      | MIRAM data files  | 38       |
| jobname  | 15       |              |          | spool files       | 40       |
| SPL      | 28       |              |          | tape files        | 41       |
|          |          |              |          | unit record files | 43       |

| Command  | Page No. |
|----------|----------|
| DEFKEY   |          |
| (delete) | 44       |
| DISPLAY  | 45       |
| DELETE   |          |
| JBQ      | 18       |
| jobname  | 18       |
| SPL      | 29       |
| DISPLAY  |          |
| ACT      | 25       |
| JBQ      | 19       |
| JS       | 21       |
| LOG      | 45       |
| SPL      | 26       |

| Command           | Page No. |
|-------------------|----------|
| DLOAD             | 14       |
| EDT               | 46, 82   |
| ENTER             |          |
| card reader       | 51       |
| diskette          | 49       |
| library file      | 47       |
| spooled file      | 48       |
| tape              | 50       |
| ERASE             |          |
| library and MIRAM |          |
| data files        | 53       |
| library module    | 52       |

| Command | Page No. |
|---------|----------|
| ESCORT  | 80       |
| EXECUTE | 13       |
| FILE    | 8        |
| FREE    | 22       |
| FSTATUS | 54       |
| GO      | 23       |
| HELP    | 55       |

| Command | Page No. | Command          | Page No. | Command          | Page No. |
|---------|----------|------------------|----------|------------------|----------|
| HOLD    |          | MENUGEN          | 75       | PUNCH            |          |
| JBQ     | 16       |                  |          | library module   | 61       |
| jobname | 16       | PAUSE            | 22       | MIRAM data file  | 62       |
| SPL     | 27       |                  |          | spool file       | 63       |
| HU      | 78       | PR/PU            | 31       | tape file        | 64       |
|         |          |                  |          | unit record file | 65       |
| I@DATA  | 72       | PRINT            |          | REBUILD          | 66       |
|         |          | library module   | 56       |                  |          |
| JC\$BLD | 73       | MIRAM data file  | 57       | RECALL           | 66       |
|         |          | spool file       | 60       |                  |          |
| LOGOFF  | 6        | tape file        | 58       | RECOVER          | 67       |
|         |          | unit record file | 59       |                  |          |
| LOGON   | 6        |                  |          | REMARK           | 68       |
| MENU    | 55       |                  |          | RESUME           | 68       |

| Command    | Page No. | Command | Page No. |
|------------|----------|---------|----------|
| RP         | 32       | SI/SC   | 11, 12   |
| RUN        | 9        | STATUS  | 70       |
| RV         | 10       | STOP    | 23       |
| RV I@DATA  | 72       | TELL    | 71       |
| RV JC\$BLD | 73       | ULD     | 13       |
| RV SFGEN   | 73       | UNLOAD  | 14       |
| SCREEN     | 69       | VTOC    | 71       |
| SFGEN      | 73       |         |          |





