UPDATE NOTICE

DECSYSTEM-20 Monitor Calls

Reference Manual

AD-4166C-T1

Insert this Update Notice in the Reference Manual to maintain an up-to-date record of changes to this manual.

CHANGED INFORMATION

The change pages contained in this update package reflect Version 3A of the TOPS-20 Software.

Additional copies of this Update Notice to the Reference Manual may be ordered from the Software Distribution Center, Digital Equipment Corporation, Maynard, Massachusetts 01754.

Order Code: AD4166C-T1

Base Manual Order Code: AA4166C-TM

September 1978

INSTRUCTIONS
AD-4166C-T1

The following list specifies which pages are to be placed in the *DECSYSTEM-20 Monitor Calls Reference Manual* as a replacement for, or addition to, current pages. The left bracket ([) means the pages are consecutive.

Title page
Copyright page

v
through
vi.1

2-5
2-6

2-17
2-18

2-29
2-30

2-47
2-48

3-13
through
3-14.1

3-29
3-30

3-55
through
3-56.2

3-65
3-66

3-77
3-78

3-99
3-100

3-103
through
3-106

3-117
3-118

3-125
3-126

3-133
3-134

3-141
through
3-144.1

3-151
through
3-152.5

3-157
through
3-158.2

3-167
through
3-172.1

3-175
3-176

3-179
3-180

3-207
3-208

3-211
3-212

3-223
through
3-226

3-257
3-258

Entire Appendix A

Entire Appendix B

Index-5
Index-6

Index-11
through
Index-16

Reader's Comment Page

September 1978

TYPE AND IDENTIFICATION OF DOCUMENTATION CHANGES

Five types of changes are used to update documents contained in the DECSYSTEM-20 Software manuals. Change symbols and notations are used to specify where, when, and why alterations were made to each updated page. The five types of update changes and the manner in which each is identified are described in the following table.

The Following Symbols and/or Notations	Identify the Following Types of Update
1. Change bar in outside margin; version number change date printed at bottom of page.	1. Changes were required by a new version of the software being described.
2. Change bar in outside margin; change date printed at bottom of page.	2. Changes were required either for the clarification or correction of existing material.
3. Change date printed at the bottom of page.	3. Changes were made for editorial purposes but use of the software is not affected.
4. Bullet (●) in outside margin; version number and change date printed at bottom of page.	4. Data was deleted in order to comply with a new version of the software being described.
5. Bullet (●) in outside margin; change date printed at bottom of page.	5. Data was deleted either to clarify or correct the existing material.

MONITOR CALLS
Reference Manual

Order Numbers: AA-4166C-TM
AA-4166C-T1

September 1978

This manual describes all of the monitor calls that exist in the TOPS-20 system. For easy reference, the monitor call descriptions are arranged alphabetically and presented concisely.

This manual updates the manual of the same name, Order Number: AA-4166C-TM.

OPERATING SYSTEM: TOPS-20 VERSION 3A

To order additional copies of this document, contact the Software Distribution Center, Digital Equipment Corporation, Maynard, Massachusetts 01754

digital equipment corporation · maynard. massachusetts

First Printing, February 1976
Revised: August 1976
Revised: May 1977
Revised: January 1978
Updated: September 1978

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

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

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

The postage-prepaid READER'S COMMENTS form on the last page of this document requests the user's critical evaluation to assist us in preparing future documentation.

The following are trademarks of Digital Equipment Corporation:

DIGITAL	DECsystem-10	MASSBUS
DEC	DECtape	OMNIBUS
PDP	DIBOL	OS/8
DECUS	EDUSYSTEM	PHA
UNIBUS	FLIP CHIP	RSTS
COMPUTER LABS	FOCAL	RSX
COMTEX	INDAC	TYPESET-8
DDT	LAB-8	TYPESET-11
DECCOMM	DECSYSTEM-20	TMS-11
ASSIST-11	RTS-8	ITPS-10
		DECSYSTEM-2020

subsequent references to the file, are also invoked when a file is opened. For example, a file's position pointer is normally reset to the beginning of the file such that the first sequential input operation reads the beginning data of the file.

2.2.5 Sample Program

A sample program follows which acquires JFNs, opens both an input and an output file, and then copies data from the input file to the output file in 7-bit bytes until the end of the input file is encountered.

```
;*** PROGRAM TO COPY INPUT FILE TO OUTPUT FILE. ***
;       (USING BIN/BOUT AND IGNORING NULL'S)

        TITLE FILEIO            ;TITLE OF PROGRAM
        SEARCH MONSYM           ;SEARCH SYSTEM JSYS-SYMBOL LIBRARY

;*** IMPURE DATA STORAGE AND DEFINITIONS ***


INJFN:  BLOCK 1                 ;STORAGE FOR INPUT JFN
OUTJFN: BLOCK 1                 ;STORAGE FOR OUTPUT JFN

        PDLEN=3                 ;STACK HAS LENGTH 3
PDLST:  BLOCK PDLEN             ;SET ASIDE STORAGE FOR STACK

A==1                            ;JSYS AC'S
B==2
C==3
D==4
T1==5                           ;TEMPORARY AC'S
                                ;....
P==17                           ;PUSH DOWN POINTER

;*** PROGRAM INITIALIZATION ***

START:  RESET                   ;CLOSE FILES AND INITIALIZE PROCESS
        MOVE P,[IOWD PDLEN,PDLST] ;ESTABLISH STACK

;*** GET INPUT-FILE ***

INFIL:  HRROI A,[ASCIZ /
INPUT FILE: /]                  ;PROMPT FOR INPUT FILE
        PSOUT                   ;ON CONTROLLING  TERMINAL
        MOVE A,[GJ%OLD+GJ%FNS+GJ%SHT];SEARCH MODES FOR GTJFN
                                ;[EXISTING FILE ONLY , FILE-NR'S IN B
                                ; SHORT CALL ]

        MOVE B,[.PRIIN,,.PRIOU] ;GTJFN'S I/O WITH CONTROLLING TERMINAL
        GTJFN                   ;GET JOB FILE NUMBER (JFN)
         ERCAL [ PUSHJ P,WARN   ;IF ERROR, GIVE WARNING
                 JRST INFIL]    ;AND LET HIM TRY AGAIN
        MOVEM A,INJFN           ;SUCCESS, SAVE THE JFN
```

```
;*** GET OUTPUT-FILE ***

OUTFIL: HRROI A,[ASCIZ /
OUTPUT FILE: /]                 ;PROMPT FOR OUTPUT FILE
        PSOUT                   ;PRINT IT
        MOVE A,[GJ%FOU+GJ%MSG+GJ%CFM+GJ%FNS+GJ%SHT];GTJFN SEARCH MODES
                                ;[DEFAULT TO NEW GENERATION , PRINT
                                ; MESSAGE , REQUIRE CONFIRMATION
                                ; FILE-NR'S IN B , SHORT CALL ]

        MOVE B,[.PRIIN,,.PRIOU] ;I/O WITH CONTROLLING TERMINAL
        GTJFN                   ;GET JOB-FILE NUMBER
         ERCAL [ PUSHJ P,WARN   ;IF ERROR, GIVE WARNING
                 JRST OUTFIL]   ;AND LET HIM TRY AGAIN
        MOVEM A,OUTJFN          ;SAVE THE JFN


;NOW, OPEN THE FILES WE JUST GOT

;   INPUT

        MOVE A,INJFN            ;RETRIEVE THE INPUT JFN
        MOVE B,[7B5+OF%RD]      ;DECLARE MODES FOR OPENF [7-BIT BYTES + INPUT]
        OPENF                   ;OPEN THE FILE
         ERJMP FATAL            ;IF ERROR, GIVE MESSAGE AND STOP

;   OUTPUT

        MOVE A,OUTJFN           ;GET THE OUTPUT JFN
        MOVE B,[7B5+OF%WR]      ;DECLARE MODES FOR OPENF [7-BIT BYTES + OUTPUT]
        OPENF                   ;OPEN THE FILE
         ERJMP FATAL            ;IF ERROR, GIVE MESSAGE AND STOP


;*** MAIN LOOP :COPY BYTES FROM INPUT TO OUTPUT ***

LOOP:   MOVE A,INJFN            ;GET THE INPUT JFN
        BIN                     ;TAKE A BYTE FROM THE SOURCE
        JUMPE B,DONE            ;IF 0, CHECK FOR END OF FILE
        MOVE A,OUTJFN           ;GET THE OUTPUT JFN
        BOUT                    ;OUTPUT THE BYTE TO DESTINATION
         ERJMP FATAL            ;IF ERROR, GIVE MESSAGE AND STOP
        JRST LOOP               ;LOOP, STOP ONLY ON A 0 BYTE (FOUND
                                ;AT LOOP+2)

;*** TEST FOR END OF FILE, ON SUCCESS FINISH UP ***

DONE:   GTSTS                   ;GET THE STATUS OF INPUT FILE.
        TLNN B,(GS%EOF)         ;AT END OF FILE?
        JRST LOOP               ;NO, FLUSH NULL AND CONTINUE COPY

CLOSIF: MOVE A,INJFN            ;YES, RETRIEVE INPUT JFN
        CLOSF                   ;CLOSE INPUT FILE
         ERJMP FATAL            ;IF ERROR, GIVE MESSAGE AND STOP

CLOSOF: MOVE A,OUTJFN           ;RETRIEVE OUTPUT JFN
        CLOSF                   ;CLOSE OUTPUT FILE
         ERJMP FATAL            ;IF ERROR, GIVE MESSAGE AND STOP
        HRROI A,[ASCIZ/
[DONE]/]                        ;SUCCESSFULLY DONE
        PSOUT                   ;PRINT IT
        JRST ZAP                ;STOP
```

13	time integral of number of runnable processes
14	exponential 1-minute average of number of runnable processes
15	exponential 5-minute average of number of runnable processes
16	exponential 15-minute average of number of runnable processes
17	time integral of number of processes waiting for the disk
20	time integral of number of processes waiting for the drum
21	number of terminal input characters
22	number of terminal output characters
23	number of system core management cycles
24	time spent doing postpurging
25	number of forced balance set process removals
26	time integral of number of processes in swap wait
27	scheduler overhead time (same as entry 2) in high precision units
30	idle time (same as entry 0) in high precision units
31	lost time (same as entry 1) in high precision units
32	user time

NOTE

This table is subject to change (usually additions) as measuring routines are added to the system.

QTIMES 0 to n Accumulated runtime of jobs on the n scheduler queues

JOBNAM Job # LH: reserved for DEC
RH: index into the system program tables for the system program being used by this job (determined by the last SETSN call executed by the job)

JOBPNM Job # SIXBIT name of program running in this job

The system program tables SNAMES, STIMES, SPFLTS, SSIZE, and SNBLKS are parallel in that the same entry in each table pertains to the same system program. The system program being run by a specific job may be determined from SNAMES, using an index obtained from table JOBNAM (above).

SNAMES SIXBIT name of system program, or 0 if this entry is unused in this and the corresponding four tables.

STIMES Total runtime of system program

SPFLTS Total number of page faults of system program

SSIZE Time integral of working set size

SNBLKS Number of samples in working set size integral

DBUGSW Debugging information
0 state of operator coverage (0=unattended, 1=attended, 2=debugging)
1 state of BUGCHK handling (0=proceed, 1=breakpoint)

LOGDES	Logging information 0 designator for logging information 1 designator for job 0 and error information
PTYPAR	Pseudo-TTY parameter information 0 LH: number of PTYs in system RH: TTY number of first PTY
SYMTAB	SIXBIT table names of all GETAB tables
DWNTIM	Downtime information 0 date and time when system will be shut down next 1 date and time when system will subsequently be up
BLDTD	Date and time system was generated
APRID	Processor serial number
HQLAV	High queue load averages
LQLAV	Low queue load averages
JBONT	Owning job
NSWPG	Default swapping pages

The following monitor calls are used for obtaining information:

GETER	Returns the last error condition
SETER	Sets the last error condition
ERSTR	Translates an error number to a string
ESOUT	Returns an error string
SYSGT	Returns values for a system table
GETAB	Returns a word from a system table
SETNM	Sets the program's private name
SETSN	Sets the program's system and private names
GETNM	Returns the program name being used by the job
SETJB	Sets a job's parameters
GETJI	Returns job information for specified job
GJINF	Returns job information for current job
STAD	Sets the system's date
GTAD	Returns the system's date
TIME	Returns the time since the system was restarted
TIMER	Sets the runtime limit of a job
RUNTM	Returns the runtime of a job or process
HPTIM	Returns the high-precision clock values
GTDAL	Returns the disk allocation of a directory
GTRPI	Returns the paging trap information
GTRPW	Returns the trap words

2.1 COMMUNICATING WITH DEVICES

The monitor calls in this group are used to communicate with the devices on the system. Some of these devices are line printers, magnetic tapes, terminals, and card readers.

Refer to Section 2.4.3.2 for the explanation of the control character output control (CCOC) words.

ASCII Code	Wakeup Class	CCOC Word(bits)	Character or Control Character
0	C	1(B0,1)	CTRL/@ null,break
1	C	1(B2,3)	CTRL/A
2	C	1(B4,5)	CTRL/B
3	C	1(B6,7)	CTRL/C
4	C	1(B8,9)	CTRL/D
5	C	1(B10,11)	CTRL/E
6	C	1(B12,13)	CTRL/F
7	C	1(B14,15)	CTRL/G bell
10	F	1(B16,17)	CTRL/H backspace
11	P	1(B18,19)	CTRL/I horizontal tab
12	F	1(B20,21)	CTRL/J line feed
13	C	1(B22,23)	CTRL/K vertical tab
14	F	1(B24,25)	CTRL/L form feed
15	F	1(B26,27)	CTRL/M carriage return
16	C	1(B28,29)	CTRL/N
17	C	1(B30,31)	CTRL/O
20	C	1(B32,33)	CTRL/P
21	C	1(B34,35)	CTRL/Q
22	C	2(B0,1)	CTRL/R
23	C	2(B2,3)	CTRL/S
24	C	2(B4,5)	CTRL/T
25	C	2(B6,7)	CTRL/U
26	C	2(B8,9)	CTRL/V
27	C	2(B10,11)	CTRL/W
30	C	2(B12,13)	CTRL/X
31	C	2(B14,15)	CTRL/Y
32	C	2(B16,17)	CTRL/Z
33	all	2(B18,19)	escape (altmode)
34	C	2(B20,21)	FS CTRL/backslash
35	C	2(B22,23)	GS CTRL/right square bracket
36	C	2(B24,25)	RS CTRL/uparrow
37	F	2(B26,27)	US CTRL/backarrow
40	P		space
41	P		!
42	P		"
43	P		#
44	P		$
45	P		%
46	P		&
47	P		'
50	P		(
51	P		)
52	P		*
53	P		+
54	P		,
55	P		-
56	P		.
57	P		/
60-71	A		0-9
72	P		:
73	P		;
74	P		<
75	P		=

ASCII Code	Wakeup Class	CCOC Word(bits)	Character or Control Character
76	P		>
77	P		?
100	P		@
101-132	A		upper case letters A-Z
133	P		[
134	P		\
135	P		]
136	P		^
137	P		_
140	P		accent (grave)
141-172	A		lower case letters a-z
173 [1]	P		left brace
174 [1]	P		vertical bar
175 [1]	P		right brace
176 [1]	P		tilde
177	all		delete (rubout)

NOTE

ESC(33) and DELETE(177) are considered to be in all wakeup classes.

---

[1] If the terminal does not have B31(TT%LIC) on in the JFN mode word, codes 173 through 176 are converted to code 33 on input.

2.4.3.4 **Terminal Characteristics Control** - The various types of terminals have different characteristics for output processing, depending on their type and speed. The characteristics that can be associated with terminals are:

1. mechanical form feed and tab
2. lower case
3. padding after carriage return
4. padding after line feed
5. padding after mechanical tab
6. padding after mechanical form feed
7. page width and length

Instead of setting each of these parameters for his line, the user can specify a terminal type number, which causes the appropriate parameters to be set. Refer to the STTYP monitor call. Tne defined terminal types, along with their characteristics, are listed below.

For conversions between local and internal date and time, the time zone in which the installation is located is normally used, with daylight saving applied from 4AM on the next to last Sunday in April to 3:59:59AM on the next to last Sunday in October.

Two monitor calls in this group, IDTIM and ODTIM, convert date and time between text strings (in core or in a file) and internal format. These should satisfy most users. However, there are four more calls, which are subsets of IDTIM and ODTIM. The calls ODTNC, IDTNC, ODCNV, and IDCNV make available separately the conversion between internal format date and time and separate numbers for local year, month, and day, and the conversion between those numbers and text strings. They also provide additional options, which give the caller more control over the conversion performed than IDTIM and ODTIM.

Time zones occur in the calling sequences of the latter four JSYS's. A time zone is represented internally as a number between -12 and 12 decimal, representing the number of hours west of Greenwich. For example, EST is zone 5. Zones -12 and 12 represent the same time but different days because the zones are on opposite sides of the international date line.

The I/O conversion monitor calls are as follows:

NIN	Inputs integer number
NOUT	Outputs integer number
FLIN	Inputs floating-point number
FLOUT	Outputs floating-point number
DFIN	Inputs double-precision, floating-point number
DFOUT	Outputs double-precision, floating-point number
IDTIM	Inputs date and time, converting to internal format
ODTIM	Outputs date and time, converting from internal format to text
IDTNC	Inputs date and time without converting to internal format
ODTNC	Outputs date and time in internal format
IDCNV	Converts from day, month, year to internal date and time
ODCNV	Converts from internal date and time to day, month, year
GTAD	Gets current date and time in internal format

2.9 PRIVILEGED MONITOR CALLS

The following monitor calls are privileged and require the process to have WHEEL or OPERATOR capability enabled:

ALLOC	Allocates a device to a particular job
BOOT	Performs functions required for loading front-end software
CRDIR	Creates or modifies a directory
GTDIR	Returns directory information
DSKOP	Allows hardware address specification of disk transfers
DIAG	Reserves and releases hardware channels
DSKAS	Assigns specific disk addresses
SJPRI	Sets job priority
SPRIW	Sets process priority
HSYS	Specifies system shutdown times
USRIO	Places program in user I/O mode
MSFRK	Starts a process in monitor mode
NODE	Performs network utility functions
PEEK	Reads monitor data

PLOCK	Locks physical pages
SNOOP	Performs system analysis
SYERR	Records data in the system error file
SMON	Sets various monitor flags
EFACT	Records data in the FACT file
MTALN	Associates magnetic tape drive with logical unit number
TTMSG	Sends a message to a terminal
PMCTL	Controls physical memory
USAGE	Writes entries into the system's accounting data file
UTEST	Tests monitor routines

7 .BTBEL

Block until a signal (doorbell) to the DECSYSTEM-20 is initiated by the communications front end. This function is used to synchronize the caller with the bootstrap program in the front end.

Argument Block

0	.BTDTE	DTE-20 number

10 .BTRMP

Read data from the communications front end using the previously loaded secondary or tertiary bootstrap program. The bootstrap program must abide by the protocol for DTE-20 transfers. The first two bytes of data will be interpreted as a count of the remaining number of bytes of data.

Argument Block

0	.BTDTE	DTE-20 number
1	.BTERR	Error status flags returned on failure of the call
2		Not used and must be zero.
3	.BTFLG	User-supplied flag word
	B0(BT%BEL)	Send a signal (doorbell) to the DECSYSTEM-20 to indicate the transfer is finished.
4	.BTCNT	Maximum number of bytes to transfer. After successful execution of this function, this word is updated to reflect the actual number of bytes transferred.
5	.BTMPT	Pointer to where data is to be placed

11 .BTKML

Load a KMC11 (DECSYSTEM-2020 only). This function will optionally load the CRAM, DRAM, and the four UNIBUS registers. Before the KMC11 is loaded, the system verifies that each bit in UNIBUS registers can be set and cleared. Before the DRAM is loaded, the system verifies that each bit in the entire DRAM can be set and cleared. After the CRAM, DRAM, and registers are loaded, they are verified to ensure that the data was properly loaded. If the register data is not supplied, the UNIBUS registers will be cleared before the KMC11 is started.

Argument Block

0	.BTKMC	KMC11 address
1	.BTKER	Error flags returned
		B0 (BT%CVE) CRAM verify error (right half is bad).
		B1 (BT%DVE) DRAM verify error (right half is bad).
		B2 (BT%RVE) Register verify error (right half is bad).
2	.BTKCC	Count of CRAM data.
3	.BTKCP	Pointer to CRAM data (16-bit data).
4	.BTKDC	Count of DRAM data.
5	.BTKDP	Pointer to DRAM data (8-bit data).
6	.BTKRC	Count of register data.
7	.BTKRP	Pointer to register data (16-bit data).
8	.BTKSA	Right-halfword is starting address.
		B0 (BT%KSA) Right-halfword is set; start KMC11.

12 .BTKMS Dump a KMC11 (DECSYSTEM-2020 only). This function will optionally dump the CRAM, DRAM, and registers if space is provided. The registers are SEL0, SEL2, SEL4, SEL6, INDATA, OUTDATA, INBA, OUTBA, and MISC*400+NPR.

Argument Block

0	.BTKMC	KMC11 address.
1	.BTKER	Error flags returned.
		B0 (BT%CVE) CRAM verify error (right half is bad).
		B1 (BT%DVE) DRAM verify error (right half is bad).
		B2 (BT%RVE) Register verify error (right half is bad).

		2	.BTKCC	Count of CRAM data.
		3	.BTKCP	Pointer to CRAM data (16-bit data).
		4	.BTKDC	Count of DRAM data.
		5	.BTKDP	Pointer to DRAM data (8-bit data).
		6	.BTKRC	Count of register data.
		7	.BTKRP	Pointer to register data (16-bit data).
13	.BTRLC	Return line counters. All counters are positive numbers.		
		Argument Block		
		0	.BTPRT	Port number.
		1	.BTSCC	Status count counter.
		2	.BTSCP	Status count pointer.
		3	.BTRCC	Receive count counter.
		4	.BTRCP	Receive count pointer.
		5	.BTTCC	Transmit count counter.
		6	.BTTCP	Transmit count pointer.
14	.BTCLI	Convert line id to port number.		
		Argument Block		
		1	.BTLID	Pointer to ASCIZ line id.
15	.BTCNP	Convert NSP port number to line id.		
		Argument Block		
		1	.BTLID	Pointer to ASCIZ line id.

The error status flag returned in word .BTERR on failure of a BOOT call are front-end reload status bits recorded in the SYSERR error file. (Refer to the TOPS-10 and TOPS-20 SYSERR Manual for an explanation of these status bits.)

Generates an illegal instruction interrupt on error conditions below.

BOOT ERROR MNEMONICS:

BOTX01: invalid DTE-20 number

BOTX02: invalid byte size

BOTX03: invalid protocol version number

BOTX04: byte count is not positive

1	.CMNUM	Parse a number. Word .CMDAT contains the radix (from 2 to 10) of the number. On a successful return, AC2 contains the number.
2	.CMNOI	Parse a guide word string, but do not return an error if no guide word is input. An error is returned only if a guide word is input that does not match the one expected by the COMND call. A guide word field must be delimited by parentheses. Word .CMDAT contains a pointer to an ASCIZ string. This string does not contain the parentheses of the guide word. Guide words are output if the user terminated the previous field with ESC. Guide words are not output, nor can they be input, if the user has caused parsing into the next field.
3	.CMSWI	Parse a switch. A switch field must begin with a slash and can be terminated with a colon in addition to any of the legal terminators. Word .CMDAT contains the address of a switch keyword symbol table. (Refer to the TBLUK monitor call description for the format of the table.) The entries in the table do not contain the slash of the switch keywords; however, they should end with a colon if the switch requires a value. The data bits CM%INV, CM%NOR, and CM%ABR defined for the .CMKEY function can also be set on this function. On a successful return, AC2 contains the address of the table entry where the switch keyword was found.
4	.CMIFI	Parse an input file specification. This function causes the COMND call to execute a GTJFN call to attempt to parse the specification for an existing file, using no default fields. The .CMGJB address (word 11 in the command state block) must be supplied, but no data should be stored in the block. (Data stored in the block will be overwritten by this COMND call.) On a successful return, AC2 contains the JFN assigned.
5	.CMOFI	Parse an output file specification. This function causes the COMND call to execute a GTJFN call to attempt to parse the specification for either a new or an existing file. The default generation number is the generation number of the existing file plus 1. The .CMGJB address must be supplied, but no data should be stored in the block. On a successful return, AC2 contains the JFN assigned.
6	.CMFIL	Parse a general (arbitrary) file specification. This function causes the COMND call to execute a GTJFN to attempt to parse the specification for the file. The .CMGJB address must be supplied, but no data should be stored in words .GJSRC, .GJCPP, .GJCPC, and .GJRTY of the GTJFN block. Also, the COMND call sets the following flag bits in the GTJFN block: GJ%XTN, G1%RND, G1%RBF, G1%RCM, and G1%RIE. (Refer to the long-form GTJFN

6	.CMFIL (Cont.)	call description for an explanation of these words and flag bits.) The program can set any other words and flag bits in the GTJFN block it supplies. On a successful return, AC2 contains the JFN assigned.
7	.CMFLD	Parse an arbitrary field. This function is useful for fields not normally handled by the COMND call. The input, as delimited by the first nonalphanumeric character, is copied into the atom buffer; the delimiter is not copied. (Hyphens are treated as alphanumerics in this application only.) No application is performed nor is any standard help message available. (See below.)
10	.CMCFM	Confirm. This function waits for the user to confirm the command with a carriage return and should be used at the end of parsing a command line.
11	.CMDIR	Parse a directory name. Login and files-only directories are allowed. Word .CMDAT contains data bits for this function. The currently defined bit is as follows: B0(CM%DWC) Allow wildcard characters to be typed in a directory name. On a successful return, AC2 contains the 36-bit directory number.
12	.CMUSR	Parse a user name. Only login directories are allowed. On a successful return, AC2 contains the 36-bit user number.
13	.CMCMA	Comma. Sets B1(CM%NOP-no parse) in word .CMFLG of the command state block and returns if a comma is not the next item in the input. Blanks can appear on either side of the comma. This function is useful for parsing a list of arguments.
14	.CMINI	Initialize the command line (e.g., set up internal monitor pointers and type the prompt). This function should be used at the beginning of parsing a command line but not when reparsing a line.
15	.CMFLT	Parse a floating-point number. On a successful return, AC2 contains the floating-point number.
16	.CMDEV	Parse a device name. On a successful return, AC2 contains the device designator.
17	.CMTXT	Parse the input text up to the next carriage return, place the text in the atom buffer, and return. If an ESC or CTRL/F is typed, it causes the terminal bell to ring (because recognition is not available with this function) and is otherwise ignored. If a ? is typed, an appropriate response is given, and the ? is not included in the atom buffer. (A ? can be included in the input text if it is preceded by a CTRL/V.)

DIAG JSYS 530

Reserves a channel and either a single device or all devices attached to that channel. This call is also used to release the channel and its devices. When the request is made, no new activity is initiated on the requested channel, and the monitor waits for current activity on all devices connected to the channel to complete. When the channel becomes idle, the process requesting the channel continues running.

The DIAG JSYS can also be used to get and release memory. The get memory function is used by the system program TGHA for performing its spare bit substitution.

Reserving or Releasing a Channel and Device(s).

ACCEPTS IN AC1: length of the argument block in the left half, and address of the argument block in the right half.

RETURNS +1: failure, error code in AC1

+2: success

The format of the argument block is as follows:

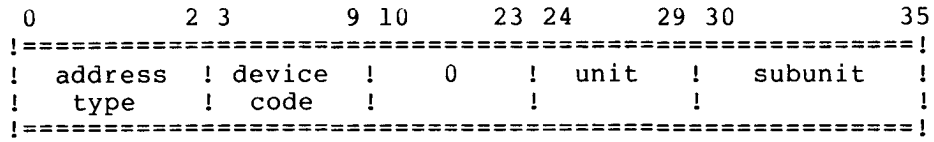
function code
data words for the function

The available functions are as follows:

Function	Symbol	Data Words	Meaning
1	.DGACU	device address time limit in milliseconds	Assign the channel and a single device. Force the device to be released after the time limit specified.
2	.DGACH	device address	Assign the channel and all devices.
3	.DGRCH	device address	Release the channel and all assigned devices.
4	.DGSCP	device address channel control word	Set up the channel program. The data transfer must be in one page. The user page pointed to by the channel control word is locked in memory. The Exec Process Table location corresponding to the channel is updated with the appropriate physical address channel control word.
5	.DGRCP	device address	Release the channel program. The page pointed to by the channel control word for the specified channel is unlocked. This function is not required before specifying a new channel program.

6	.DGGCS	device address word 0 word 1 word 2 word 4	Return the status of the channel. The specified words are the logout area for the channel.

The device address given in the argument block is a machine-dependent specification for the channel and device to be assigned. The devices that can be assigned must be attached to the RH20 controller and must be mounted by a process with the WHEEL, OPERATOR, or MAINTENANCE capability enabled. The format of the device address word is



DIAG ERROR MNEMONICS:

DIAGX1: invalid function

DIAGX2: device is not assigned

DIAGX3: argument block too small

DIAGX4: invalid device type

DIAGX5: WHEEL, OPERATOR, or MAINTENANCE capability required

DIAGX6: invalid channel command list

DIAGX7: illegal to do I/O across page boundary

DIAGX8: no such device

DIAGX9: unit does not exist

DIAG10: subunit does not exist

Getting Memory

ACCEPTS IN AC1: minus count of controllers in left half; address of argument block in right half.

RETURNS +1: failure; error code in AC1

+2: success

The format of the argument block is as follows:

word 0	function code (.DGGEM)
word 1	first page in user address space
word 2	first physical memory page
word 3	number of pages
word 4	user address of AR/ARX parity trap routines

Upon successful return, this function accomplishes the following:

TOPS-20 has requested that all of the front ends refrain from accessing common memory.

The hardware PI system has been turned off; no scheduling can occur.

The time base and interval timer have been turned off.

All DTE byte transfers have completed.

All RH20 activity has ceased.

The designated pages of the process' address space have been set up to address the designated physical memory. Note that this is not the same as your having requested the pages with PLOCK. With the get memory function, the data in the physical memory pages have been retained, and the ownership of the pages is unchanged.

The CST0 entries for each of the designated physical pages have been saved and set as follows:

The age is set to the present age of the requesting process.

The process use field is set to all ones.

The modified bit is set to one.

The entire address space of the requesting process has been locked in memory. (Actually, only the pages that existed at the time of the DIAG call are locked. Therefore, the process must ensure that all of the pages it needs exist and are private when DIAG is executed.)

Releasing Memory

ACCEPTS IN AC1: minus count of controllers left half; address of argument block in right half.

RETURNS +1: failure; error code in AC1

+2: success

The format of the argument block is as follows:

word 0 function code (.DGREM)

DIAG ERROR MNEMONICS

DIAGX1: invalid function

DIAGX3: argument block too small

DIAGX5: WHEEL, OPERATOR, or MAINTENANCE capability required

DIAGX7: illegal to do I/O across page boundary

DIBE JSYS 212

Dismisses the process until the designated file input buffer is empty.

ACCEPTS IN AC1: file designator

RETURNS +1: always

Returns immediately if the designator is not associated with a terminal.

The DOBE monitor call can be used to dismiss the process until the designated file output buffer is empty.

Generates an illegal instruction interrupt on error conditions below.

DIBE ERROR MNEMONICS:

DESX1: invalid source/destination designator

DESX3: JFN is not assigned

DUMPO ERROR MNEMONICS:

DUMPX1: command list error

DUMPX2: JFN is not open in dump mode

DUMPX3: address error (too big or crosses end of memory)

DUMPX4: access error (cannot read or write data in memory)

DUMPX5: no-wait dump mode not supported for this device

DUMPX6: dump mode not supported for this device

DESX1: invalid source/destination designator

DESX2: terminal is not available to this job

DESX3: JFN is not assigned

DESX4: invalid use of terminal designator or string pointer

DESX5: file is not open

IOX2: file is not opened for writing

IOX5: device or data error

IOX11: quota exceeded or disk full

DVCHR JSYS 117

Returns the device characteristics of the specified device.

ACCEPTS IN AC1: JFN or device designator

RETURNS +1: always, with

AC1 containing the device designator (even if a JFN was given).
AC2 containing the device characteristics word.
AC3 containing the job number to which the device is assigned in the left half and the unit number in the right half. If the device is a structure or does not have units, the right half is -1.

The contents of AC3 are -1 if the device is not assigned to any job or -2 if the device allocator has ownership of the UFN or device designator.

Device Characteristics Word

Bit	Symbol	Meaning
0	DV%OUT	device can do output
1	DV%IN	device can do input
2	DV%DIR	device has a directory
3	DV%AS	device is assignable with ASND
4	DV%MDD	device has multiple directories

Device Characteristics Word (Cont.)

5	DV%AV	device is available or assigned to this job		
6	DV%ASN	device is assigned by ASND		
8	DV%MNT	device is mounted		
9-17	DV%TYP	device type		
		0	.DVDSK	disk
		2	.DVMTA	magnetic tape
		7	.DVLPT	line printer
		10	.DVCDR	card reader
		11	.DVFE	front-end pseudo-device
		12	.DVTTY	terminal
		13	.DVPTY	pseudo-terminal
		15	.DVNUL	null device
		16	.DVNET	ARPA network
20-35	DV%MOD	data mode in which device can be opened		
		B20	DV%M17	dump mode
		B27	DV%M10	image mode
		B35	DV%M0	normal mode

Generates an illegal instruction interrupt on error conditions below.

DVCHR ERROR MNEMONICS:

DEVX1: invalid device designator

DESX1: invalid source/destination designator

DESX3: JFN is not assigned

DESX4: invalid use of terminal designator or string pointer

EFACT JSYS 5

Makes an entry in the FACT file. The EFACT monitor call is obsolete and provided only for existing programs that make entries in the FACT file. New programs should use the USAGE monitor call to make entries in the new USAGE file.

ACCEPTS IN AC1: LH: negative size of entry
RH: pointer to beginning of entry (size bits of entry will be updated by the system from the negative size specified)

RETURNS +1: failure, error code in AC1

+2: success

The EFACT call returns successfully without making an entry in the FACT file if the monitor flag SF%FAC (refer to SMON and TMON calls) is not set.

The EFACT monitor call can be executed only by the monitor or by a process that has WHEEL or OPERATOR capability enabled.

Generates an illegal instruction interrupt on error conditions below.

EPCAP ERROR MNEMONICS:

FRKHX1: invalid process handle

FRKHX2: illegal to manipulate a superior process

ERSTR JSYS 11

Translates a TOPS-20 error number to its corresponding text string and writes the string to the specified destination. This error number is the one returned in an AC (usually in AC1) on a JSYS error and is associated with a unique error mnemonic and text string. The error numbers begin at 600010 and are defined in the system file MONSYM.MAC. (Refer to Appendix A for the list of error numbers, mnemonics, and text strings.)

ACCEPTS IN AC1: destination designator

AC2: LH: process handle
RH: error number, or -1 for the most recent error in the specified process

AC3: LH: a negative count of the maximum number of bytes in the string to be transferred, or 0 for no limit
RH: 0

RETURNS +1: failure, undefined error number

+2: failure, string size out of bounds or invalid destination designator

+3: success

Generates an illegal instruction interrupt on error conditions below.

ERSTR ERROR MNEMONICS:

DESX1: invalid source/destination designator

FRKHX1: invalid process handle

IOX11: quota exceeded or disk full

ESOUT JSYS 313

Outputs an error string. This monitor call is used for reporting an error in the input from the primary input stream in order to cause re-synchronization of the input transaction. This mechanism is convenient for communication with a user who made a typing error and may have continued to type ahead. It also standardizes the format of error messages.

ACCEPTS IN AC1: pointer to a string in the caller's address space. The string is terminated with a null character.

RETURNS +1: always, updated string pointer in AC1

The ESOUT call waits for the primary output buffer to empty and then outputs a carriage return, line feed, and question mark to the primary output designator. Next it clears the primary input buffer and outputs the error string to the primary output designator.

Can cause several software interrupts or process terminations on certain file conditions. (Refer to bit OF%HER of the OPENF call description.)

FFFFP JSYS 31

Finds the first free page in the specified file. A free page is one that is marked as not being in use. The FFFFP call is useful for finding a nonused page in a file before a PMAP call is executed that writes into that page.

ACCEPTS IN AC1: JFN

RETURNS +1: always, with the JFN in the left half of AC1 and the page number in the right half of AC1, or -1 if there is no free page.

Generates an illegal instruction interrupt on error conditions below.

FFFFP ERROR MNEMONICS:

DESX1: invalid source/destination designator

DESX3: JFN is not assigned

DESX4: illegal use of terminal designator or string pointer

DESX5: file is not open

12	GJ%OFG (Cont.)	When both B11(GJ%IFG) and B12(GJ%OFG) are on, the GTJFN call parses the specification given, verifying the existence of each field. When a wildcard character appears in a field, the GTJFN call checks the remaining fields for correct punctuation and returns a JFN for the file specification string only. That is, once a wildcard character is seen, the action taken is identical to that taken when only B12(GJ%OFG) is set. If no wildcard character appears in the string, the action is the same as if both bits were off.
13	GJ%FLG	Flags are to be returned in the left half of AC1 on a successful return.
14	GJ%PHY	User logical names specified for the current job are to be ignored and the physical device is to be used.
15	GJ%XTN	This bit is off in the short form of the GTJFN call.
16	GJ%FNS	The contents of AC2 are to be interpreted as follows: 1. If this bit is on, AC2 contains an input JFN in the left half and an output JFN in the right half. The input JFN is used to obtain the file specification to be associated with the JFN. The output JFN is used to indicate the destination for printing the names of any fields being recognized. To omit either JFN, specify .NULIO (377777). 2. If this bit is off, AC2 contains a pointer to an ASCIZ string in memory that specifies the file to be associated with the JFN.
17	GJ%SHT	This bit must be on for the short form of the GTJFN call.
18-35		The generation number of the file. The following values are permitted; however, 0 is the normal case. 0(.GJDEF) to indicate that the next higher generation number of the file is to be used if GJ%FOU (bit 0) is on, or to indicate that the highest existing generation number of the file is to be used if GJ%FOU is off.

18-35 (Cont.)	-1(.GJNHG)	to indicate that the next higher generation number of the file is to be used if no generation number is supplied.
	-2(.GJLEG)	to indicate that the lowest existing generation number of the file is to be used if no generation number is supplied.
	-3(.GJALL)	to indicate that all generation numbers (*) of the file are to be used and that the JFN is to be assigned to the first file in the group, if no generation number is supplied. (Bit GJ%IFG must be set.)
	1-377777	to indicate that the specified generation number of the file is to be used if no generation number is supplied.

The GTJFN monitor call always reads the terminating character after the file specification string. (This character can be obtained by executing the BKJFN call followed by a BIN call.) The valid terminating characters are:

line feed	left parenthesis
CTRL/L	right parenthesis
CTRL/Z	plus sign
carriage return	comma
exclamation point	slash
double quotation marks	equals sign
number sign	at sign (@)
ampersand	space
single quotation mark	ESC

All of these characters except for ESC are also confirmation characters (refer to bit GJ%CFM above) and are called confirming terminators. If a confirming terminator is typed after the string, a confirmation message will not be typed to the user nor will the user be required to confirm the string obtained, regardless of the setting of GJ%MSG and GJ%CFM.

On a successful return, the following flags are returned in the left half of AC1 if flag bit GJ%IFG, GJ%OFG, or GJ%FLG was on in the call.

GJFX34: invalid character "?" in file specification

GJFX35: directory access privileges required

GJFX36: internal format of directory is incorrect

GJFX37: input deleted

GJFX38: file not found because output-only device was specified

GJFX39: logical name loop detected

GJFX40: undefined attribute in file specification

GJFX41: file name must not exceed 6 characters

GJFX42: file type must not exceed 3 characters

GJFX43: more than one ;T specification is not allowed

GJFX44: account string does not match

GJFX45: illegal to request multiple specifications for the same attribute

GJFX46: attribute value is required

GJFX47: attribute does not take a value

GJFX48: GTJFN input buffer is empty

GJFX49: invalid attribute for this device

GFX51: byte size too small

IOX11: quota exceeded or disk full

DESX9: invalid operation for this device

GTJFN JSYS 20
LONG FORM

Returns a JFN for the specified file. Accepts the specification for the file from both a string in memory and from a file. If both are given as arguments, the string is used first, and then the file is used if more fields are needed to complete the specification. This form also allows the program to specify nonstandard values to be used for omitted fields and to request the assignment of a specific JFN.

ACCEPTS IN AC1: 0 in the left half, and address of the beginning of the argument table in the caller's address space in the right half

AC2: pointer to ASCIZ file specification string in the caller's address space, or 0 if none

RETURNS +1: failure, error code in AC1

+2: success, flags in the left half of AC1, and the JFN assigned in the right half of AC1. (This word is called an indexable file handle and is given to the GNJFN call as an argument.) Updated string pointer in AC2, if pertinent.

All I/O errors can occur. These errors cause software interrupts or process terminations, and only a single return (+1) is given.

The format of the argument table specified by the right half of AC1 is described below. Words 0 through 10 (.GJGEN-.GJJFN) must be supplied in the long form of the GTJFN call. The remaining words are optional, and if they are supplied, B15(GJ%XTN) of word .GJGEN must be on.

Word	Symbol	Meaning
0	.GJGEN	Flag bits in the left half and generation number in the right half. (See below.)
1	.GJSRC	Input JFN in the left half and output JFN in the right half. To omit either JFN, specify .NULIO (377777).
2	.GJDEV	Pointer to ASCIZ string that specifies the default device to be used when none is given. If this word is 0, the user's connected structure will be used.
3	.GJDIR	Pointer to ASCIZ string that specifies the default directory to be used when none is given. If this word is 0, the user's connected directory will be used.
4	.GJNAM	Pointer to ASCIZ string that specifies the default filename to be used when none is given. If this word is 0, either the string or the input JFN must supply the filename.
5	.GJEXT	Pointer to ASCIZ string that specifies the default file type to be used when none is given. If this word is 0, the null file type will be used.
6	.GJPRO	Pointer to ASCIZ string that specifies the default protection to be used when none is given. If this word is 0, the default protection as specified in the directory or the protection of the next lower generation will be used.
7	.GJACT	Pointer to ASCIZ string that specifies the default account to be used when none is given. If this word is 0, the user's LOGIN account (unless changed) will be used.
10	.GJJFN	The JFN to associate with the file specification if flag GJ%JFN is set in word 0 (.GJGEN) of the argument block.

11 .GJF2 Extended argument block if B15(GJ%XTN) is on in the left half of .GJGEN. This word contains a second group of flags in the left half and the count of the number of words following this word in the argument block in the right half. The flags in the left half specify additional control over the GTJFN process. The following flags are defined:

B0(G1%RND) Return to the caller if the filename buffer becomes empty, and the user attempts to delete a character. This can occur if the user, when giving the filename, types a CTRL/U or types a DELETE or CTRL/W and there are no more characters in the buffer.

B2(G1%NLN) Filenames cannot be longer than 6 characters and file types cannot be longer than 3 characters. In addition, the generation number, temporary status, protection, and account fields cannot be specified in the string or the input data.

B3(G1%RCM) Return the confirmation message to the caller by placing it in the destination buffer.

B4(G1%RIE) Return to the caller if the input buffer becomes empty, and the user attempts to delete a character.

12 .GJCPP Pointer to string where GTJFN is to store the exact copy of the user's typescript (destination string pointer). This string will contain logical names, if they were typed by the user, and will not contain the default fields unless they were generated through recognition. This string allows the caller to obtain a true copy of the user's typescript.

13 .GJCPC Number of bytes available in the destination string pointed to by .GTCPP (word 12). If a pointer has been specified but this word is 0, the monitor assumes the string contains 130 bytes.

14 .GJRTY Pointer to the buffer for text to be output when the user types a CTRL/R (i.e., pointer to the CTRL/R buffer). This pointer cannot be equal to the pointer given in AC2. (Refer to the TEXTI call for the definition of the CTRL/R buffer.)

15 .GJBFP Pointer to the beginning of the destination buffer.

16 .GJATR Pointer to the file specification attribute block. This word is reserved for future use.

The flag bits accepted in the left half of .GJGEN (word 0) of the argument block are basically the same as those accepted in the short form of the GTJFN call. The entire set of bits is listed below. (Refer to GTJFN - SHORT FORM for more detailed explanations of these bits.) The flags that are different in the two forms are GJ%JFN, GJ%XTN, GJ%FNS, and GJ%SHT.

Bit	Symbol	Meaning
0	GJ%FOU	Create a new version of the file.
1	GJ%NEW	The file must not exist.
2	GJ%OLD	The file must exist.
3	GJ%MSG	Type a message if the user presses ESC to terminate input.
4	GJ%CFM	Confirmation from the user is required.
5	GJ%TMP	The file is temporary.
6	GJ%NS	Search only the first specification in a multiple logical name definition.
7	GJ%ACC	The JFN cannot be accessed by inferior processes.
8	GJ%DEL	Ignore the file deleted bit in the FDB.
9-10	GJ%JFN	Associate the JFN supplied in .GJJFN (word 10) of the argument block with the file specification. The value of this field is interpreted as follows:

Value	Meaning
0(.GJDNU)	Ignore the JFN supplied.
2(.GJERR)	Attempt to assign the JFN supplied and return an error if it is not available.
3(.GJALT)	Attempt to assign the JFN supplied and, if it is not available, assign an alternate.

11	GJ%IFG	The file specification can contain wildcard characters.
12	GJ%OFG	Associate the JFN with the file specification string and not the file itself.
13	GJ%FLG	Return flags in AC1 on successful completion of the call.
14	GJ%PHY	The physical device is to be used.
15	GJ%XTN	The argument block contains more than 10 (octal) words.
16	GJ%FNS	This bit is ignored for the long form of the GTJFN call.
17	GJ%SHT	This bit must be off for the long form of the GTJFN call.

IIC ERROR MNEMONICS:

FRKHX1: invalid process handle

FRKHX2: illegal to manipulate a superior process

FRKHX3: invalid use of multiple process handle

INLNM JSYS 503

Returns a logical name that is defined either for this job or for the system. (Refer to Section 2.2.2 and CRLNM and LNMST monitor calls.)

ACCEPTS IN AC1: function code in the left half, and index into the table of defined logical names in the right half

AC2: pointer to the string for storing the logical name

RETURNS +1: failure, error code in AC1

+2: success, updated string pointer in AC2

The available functions are:

Code	Symbol	Meaning
0	.INLJB	List the logical names defined for this job
1	.INLSY	List the logical names defined for the system

INLNM ERROR MNEMONICS:

INLNX1: index is beyond end of logical name table

INLNX2: invalid function

JFNS JSYS 30

Returns the file specification currently associated with the JFN.

ACCEPTS IN AC1: destination designator where the ASCIZ string is to be written

AC2: indexable file handle (refer to GTJFN), or pointer to string

AC3: format control bits to be used when returning the string, or 0

AC4: pointer to string containing prefix of file specification attribute

RETURNS +1: always, updated string pointer, if pertinent, in AC1

AC2 can have one of two formats, depending on B26(JS%PTR) in AC3. The first format is a word with either 0 or the flag bits returned from GTJFN in the left half and the JFN in the right half. When the left half is 0, the string returned is the exact specification associated with the JFN. If the given JFN is associated only with a file specification (i.e., it was obtained with B12(GJ%OFG) on in the GTJFN call), the string returned contains null fields for nonexistent fields or fields containing wildcards, and actual values for existent fields. When the left half is nonzero, the string returned contains wildcard characters for appropriate fields and 0, -1, or -2 as a generation number if the corresponding bit is on in the call.

The second format (allowed only if B26(JS%PTR) of AC3 is on) is a pointer to the string to be returned. This string is one field of a file specification. The field is determined by the first nonzero 3-bit field in AC3 or by the setting of B27(JS%ATR) or B28(JS%AT1) in AC3. For example, if bits 6-8 (JS%NAM) of AC3 are nonzero, then the string is interpreted as a filename field. If B27(JS%ATR) is on, the string is interpreted as a file specification attribute. If B28(JS%AT1) is on, the string is concatenated to the string pointed to by AC4, and a colon is inserted between the two strings. In all cases, the string is output to the destination designator, and the appropriate punctuation is added.

AC3 contains control bits for formatting the string being returned. B0-B20 are divided into 3-bit bytes, each byte representing a field in the file specification. The value of the byte indicates the output for that field. The values are:

0	(.JSNOF)	do not output this field
1	(.JSAOF)	always output this field
2	(.JSSSD)	suppress this field if it is the system default

The bits that can be set in AC3 are as follows:

B0-B2(JS%DEV)	output for device field
B3-B5(JS%DIR)	output for directory field
B6-B8(JS%NAM)	output for filename field (2 is illegal)
B9-B11(JS%TYP)	output for file type field (2 is illegal)
B12-B14(JS%GEN)	output for generation number field
B15-B17(JS%PRO)	output for protection field
B18-B20(JS%ACT)	output for account field
B21(JS%TMP)	return ;T if appropriate
B22(JS%SIZ)	return size of file in pages
B23(JS%CDR)	return creation date
B24(JS%LWR)	return date of last write
B25(JS%LRD)	return date of last read
B26(JS%PTR)	AC2 contains pointer to the string being returned
B27(JS%ATR)	return file specification attributes if appropriate
B28(JS%AT1)	return the specific specification attribute whose prefix is indicated by the string pointed to in AC4. This bit is used when a program is processing attributes one at a time. If JS%ATR is also set, all attributes will be returned.
B32(JS%PSD)	punctuate the size and date fields
B33(JS%TBR)	tab before all fields returned, except for first field
B34(JS%TBP)	tab before all fields that may be returned (i.e., fields whose value is given as 1 or 2), except for first field
B35(JS%PAF)	punctuate all fields from device through ;T

IPCF31: invalid page number

IPCF32: page is not private

IPCF34: cannot receive into an existing page

MSEND JSYS 510

Sends an IPCF (Inter-Process Communication Facility) message. The message is in the form of a packet and can be sent to either the specified PID or the system process <SYSTEM>INFO. Refer to the DECsystem-20 Monitor Calls User's Guide for an overview and description of the Inter-Process Communication Facility.

ACCEPTS IN AC1: length of packet descriptor block

AC2: address of packet descriptor block

RETURNS +1: failure, error code in AC1

+2: success. The packet is sent to the receiver's input queue. Word .IPCFS of the packet descriptor block is updated with the sender's PID. This updating is done in case the PID was being defaulted or created by this call.

The format of the packet descriptor block is as follows:

Word	Symbol	Meaning
0	.IPCFL	Flags. (See below.)
1	.IPCFS	PID of sender, or 0 if no PID exists for sender. This word will be 0 if the caller is creating a PID (i.e., flag bit IP%CPD is on).
2	.IPCFR	PID of receiver, or 0 if receiver is <SYSTEM>INFO.
3	.IPCFP	Pointer to message block (length of message in the left half and starting address of message in the right half). When a packet is sent to <SYSTEM>INFO, the message block contains the request being made. (See below.)

The following flags are defined in word .IPCFL of the packet descriptor block. These flags can be set on both the MSEND and MRECV calls.

Flags Set By Caller

B0(IP%CFB) Do not block process if there are no messages in the queue. If this bit is set, an error is given if there are no messages.

B1(IP%CFS) Use, as the sender's PID, the PID obtained from the address specified in word .IPCFS.

B2(IP%CFR) Use, as the receiver's PID, the PID obtained from the address specified in word .IPCFR.

B3(IP%CFO) Allow one send request above the quota. (The default send quota is 2.)

B4(IP%TTL) Truncate the message, if it is larger than the space reserved. If this bit is not set, an error is given if the message is too large.

B5(IP%CPD) Create a PID to use as the sender's PID and return it in word .IPCFS of the packet descriptor block.

B6(IP%JWP) Make the created PID be job wide (i.e., permanent until the job logs out). If this bit is not set, the PID is temporary until the process executes the RESET monitor call. If B5(IP%CPD) is not set, B6 is ignored.

B7(IP%NOA) Do not allow other processes to use the created PID. If B5(IP%CPD) is not set, B7 is ignored.

B18(IP%CFP) The packet is privileged. (This bit can be set only by a process with WHEEL capability enabled.) When a privileged sender sets this bit, the MRECV and MUTIL calls return it set for any reply. An error is given if this bit is set by the sender and the receiver is not privileged.

B19(IP%CFV) The packet is a page of data. Word .IPCFP of the packet descriptor block contains 1000 in the left half and the page number in the right half. The page the packet is being sent to, or is being received into, must be private.

Flags Returned After Call

B20(IP%CFZ) A zero-length message was sent, and the packet consists of only the packet descriptor block.

B24-B29 (IP%CFE) Error code field for errors encountered by <SYSTEM>INFO during a send or receive request.

Code	Symbol	Meaning
15	.IPCPI	insufficient privileges
16	.IPCUF	invalid function
67	.IPCSN	<SYSTEM>INFO needs name
72	.IPCFF	<SYSTEM>INFO free space exhausted
74	.IPCBP	PID has no name or is invalid
75	.IPCDN	duplicate name has been specified
76	.IPCNN	unknown name has been specified
77	.IPCEN	invalid name has been specified

B30-B32 (IP%CFC) System and sender code. This code can be set only by a process with WHEEL capability enabled. The system returns the code so that a nonprivileged user can examine it.

Code	Symbol	Meaning
1	.IPCCC	sent by <SYSTEM>IPCF
2	.IPCCF	sent by system-wide <SYSTEM>INFO
3	.IPCCP	sent by receiver's <SYSTEM>INFO

2 .MSTFL Flag bits in the left half, and the number of units in the structure (.MSTNU) in the right half. The bits that can be set in the left half are:

B0(MS%NFH) If one of the HOME blocks is incorrect, do not fix it and do return an error. If this bit is off and one of the HOME blocks is incorrect, the correct block is copied into the bad HOME block and the mounting procedure continues.

B1(MS%NFB) If one of the BAT (Bad Allocation Table) blocks is incorrect, do not fix it and do return an error. If this bit is off and one of the BAT blocks is incorrect, the correct block is copied into the bad BAT block and the mounting procedure continues.

B2(MS%XCL) Mount the structure for exclusive use by this job. This bit is set by a system program when it initializes or reconstructs a structure. If this bit if off, the structure is mounted for general use.

B3(MS%IGN) Ignore correctable errors in the bit table and in the root directory on this structure. This bit is set by a system program when it reconstructs the root directory on a structure or rebuilds the bit table. If this bit is off and an error is detected, this function returns an error.

3 .MSTUI Beginning of unit information for each unit in the structure. The information is 3 words long per unit, and the symbol for this length is .MSTNO. The first 3-word block is for logical unit 0, and the last 3-word block is for the last logical unit (.MSTNU-1). The offsets into the 3-word block are:

0	.MSTCH	Channel number of unit
1	.MSTCT	Controller number of unit (currently must be -1)
2	.MSTUN	Unit number of unit

After successful completion of this function, the given structure is mounted and available for general use (unless bit MS%XCL was on in word .MSTFL of the argument block).

The following errors are possible on the failure of this function.

MSTRX2: WHEEL or OPERATOR capability required

MSTRX3: argument block too small

MSTRX4: insufficient system resources

MSTRX5: drive is not on line

MSTRX6: home blocks are bad

MSTRX7: invalid structure name

MSTRX8: could not get OFN for ROOT-DIRECTORY

MSTRX9: could not MAP ROOT-DIRECTORY

MSTX10: ROOT-DIRECTORY bad

MSTX11: could not initialize Index Table

MSTX12: could not OPEN Bit Table File

MSTX13: backup copy of ROOT-DIRECTORY is bad

MSTX14: invalid channel number

MSTX15: invalid unit number

MSTX16: invalid controller number

MSTX17: all units in a structure must be of the same type

MSTX19: unit is already part of a mounted structure

MSTX20: data error reading HOME blocks

MSTX23: could not write HOME blocks

MSTX25: invalid number of swapping pages

MSTX27: specified unit is not a disk

MSTX30: incorrect Bit Table counts on structure

MSTX34: unit is write-locked

MSTX35: too many units in structure

MONX01: insufficient system resources

Dismounting a Given Structure - .MSDIS

This function indicates that the given structure can be removed from the system. Any mounted structure other than the public structure PS: can be dismounted with this function. (The public structure PS: is dismounted at system shutdown.)

The format of the argument block is as follows:

Word	Symbol	Meaning
0	.MSUAL	Pointer to ASCIZ string containing the alias of the structure, or device designator of the structure.
1	.MSUFL	Flag bits in the left half and 0 in the right half. The bits that can be set are:
		B0(MS%GTA) Return users who have accessed the structure.
		B1(MS%GTM) Return users who have incremented the mount count.
		B2(MS%GTC) Return users who are connected to the structure.

After successful execution of this function, word 1 through word n+1 (where n is the number of items returned) are updated with the following information.

Word	Symbol	Meaning
1	.MSUFL	Right half contains the number of items (n) being returned. Left half is unchanged.
2	.MSUJ1	Flag bits for the job in the left half, and number of job in the right half.
.		.
.		.
.		.
n + 1		Flag bits for the job in the left half, and number of job in the right half.
		The bits returned for each job are defined as:
		B0(MS%GTA) Job has accessed structure.
		B1(MS%GTM) Job has incremented the mount count for structure.
		B2(MS%GTC) Job has connected to structure.

The following errors are possible on the failure of this function.

MSTRX1: invalid function

MSTRX3: argument block too small

STRX01: structure is not mounted

STDVX1: no such device

ARGX18: invalid structure name

MONX01: insufficient system resources

Specifying word and bits to be modified - .MSHOM

This function allows enabled WHEEL or OPERATOR program to specify word of homeblock of mounted structure to be modified, which bits should be modified, and what the new values should be.

The format of the argument block is as follows:

Word	Symbol	Meaning
0	.MSHNM	Handle on alias such as pointer to string, or device designator.
1	.MSHOF	Offset specifying which word should be changed.
2	.MSHVL	Value for new bits.
3	.MSHMK	Mask showing which bits should be changed.

The following errors are possible on the failure of this function:

MSTRX2: insufficient privileges

MSTRX3: argument block too small

MSTX21: structure not mounted

any errors "MODHOM" routine returns

MTALN JSYS 774

Associates a given serial-numbered magnetic tape drive with the specified logical unit number. This monitor call requires the process to have WHEEL or OPERATOR capability enabled. The MTALN call is a temporary call and may not be defined in future releases.

ACCEPTS IN AC1: slave type in left half; logical unit number of magtape in right half

AC2: decimal serial number of magnetic tape drive

RETURNS +1: always

All units are searched for the specified serial number and slave type. When they are found, the drive is associated with the given logical unit number. The original unit is now associated with the logical unit number that the specified serial-numbered drive had before it was reassigned.

The slaves recognized are

.MTT45	TU45 (The system default)
.MTT70	TU70
.MTT71	TU71
.MTT72	TU72

Generates an illegal instruction interrupt on error conditions below.

MTALN ERROR MNEMONICS:

WHELX1: WHEEL or OPERATOR capability required

DEVX1: invalid device designator

OPNX7: device already assigned to another job

MTOPR JSYS 77

Performs various device-dependent control functions. This monitor call requires that the device either be opened or be assigned to the caller if the device is an assignable device.

Because of the device dependencies of the MTOPR call, programs written with device-independent code should not use this call unless they first check for the type of device.

ACCEPTS IN AC1: JFN of the device

AC2: function code (see below)

AC3: function arguments or address of argument block (see descriptions of individual devices)

RETURNS +1: always

The functions listed for each device apply only to that device. If a function applies to more than one device, its description is repeated for each applicable device.

MTA Functions

The functions available for magnetic tapes (MTA) are described below. Some of these functions accept arguments in AC3 (refer to the individual descriptions).

Code	Symbol	Meaning
0	.MOCLE	Clear any error flags from a previous MTOPR call.
31	.MONOP	Wait for all activity to stop.
1	.MOREW	Rewind the tape. This function waits for activity to stop before winding the tape. If sequential data is being output, the last partial buffer is written before the tape is rewound. Control returns to caller when rewinding begins.
11	.MORUL	Rewind and unload the tape. This function is identical to the .MOREW function and also unloads the tape if the hardware supports tape unloading.
10	.MOEOT	Advance forward until two sequential tape marks are seen and position tape after the first tape mark.

3	.MOEOF	Write a tape mark. This function requires that the magnetic tape be opened for write access. If sequential data is being output, the last partial buffer is written before the tape mark.
6	.MOFWR	Advance over one record in the direction away from the beginning of the tape. If sequential data is being read in the forward direction and not all of the record has been read, this function advances to the start of the next record. If sequential data is being read in the reverse direction and not all of the record has been read, this function positions the tape at the end of that record.
7	.MOBKR	Space backward over one record in the direction toward the beginning of the tape. If sequential data is being read in the forward direction and not all of the record has been read, this function positions the tape back to the start of that record. If sequential data is being read in the reverse direction and not all of the record has been read, this function positions the tape to the end of the record physically preceding that record.
16	.MOFWF	Advance to the start of the next file. This function advances the tape in the direction away from the beginning of the tape until it passes over a tape mark.
17	.MOBKF	Space backward over one file. This function moves the tape in the direction toward the beginning of the tape until it passes over a tape mark or reaches the beginning of the tape, whichever occurs first.
2	.MOSDR	Set the direction of the tape motions for read operations. This function requires AC3 to contain the desired direction. If AC3 = 0, the tape motion is forwards; if AC3 = 1, the tape motion is backwards.
26	.MORDR	Return the direction that the tape is moving during read operations. On a successful return, AC3 = 0 if the direction of the tape motion is forwards, or AC3 = 1 if the direction of the tape motion is backwards.
5	.MOSRS	Set the size of the records. This function requires AC3 to contain the desired number of bytes in the records.
15	.MORRS	Return the size of the records. On a successful return, AC3 contains the number of bytes in the records.

24 .MOSDN Set the density. The function requires AC3 to contain the desired density:

0	.SJDDN	default system density
1	.SJDN2	200 BPI (8 rows/mm)
2	.SJDN5	556 BPI (22 rows/mm)
3	.SJDN8	800 BPI (31 rows/mm)
4	.SJD16	1600 BPI (63 rows/mm)
5	.SJD62	6250 BPI (246 rows/mm)

12 .MORDN Return the current density setting. On a successful return, AC3 contains the current density.

4 .MOSDM Set the hardware data mode to be used when transferring data to and from the tape. This function requires AC3 to contain the desired data mode:

0	.SJDDM	default system data mode
1	.SJDMC	dump mode (36-bit bytes)
2	.SJDM6	SIXBIT byte mode for 7-track drives
3	.SJDMA	ANSI ASCII mode (7 bits in 8-bit bytes)
4	.SJDM8	industry compatible mode
5	.SJDMH	High-density mode for TU70 and TU72 tape drives only (nine 8-bit bytes in two words).

14 .MORDM Return the hardware data mode currently being used in transfers to and from the tape. On a successful return, AC3 contains the current data mode.

20 .MOSPR Set the parity. This function requires AC3 to contain the desired parity:

0	.SJPRO	odd parity
1	.SJPRE	even parity

21 .MORPR Return the current parity. On a successful return, AC3 contains the current parity.

27 .MOSID Set the reel identification of the tape mounted. The process must have WHEEL or OPERATOR capability enabled. This function requires AC3 to contain the desired 36-bit reel ID.

FE Functions

Code	Symbol	Meaning
3	.MOEOF	Send an end of file to the program using the FE device on the front end. This function is used for synchronization between a program running on the DECSYSTEM-20 and a program running on the front end.
4	.MODTE	Assign the specified device to the DTE controller on the front end. This function, which must be performed before I/O is allowed to the device, requires AC3 to contain the device type. The process must have WHEEL or OPERATOR capability enabled.

TTY Functions

25	.MOPIH	Determine if TTY job needs input. On a successful return, AC2 contains 0(.MONWI) if TTY job is not waiting for input or contains -1(.MOWFI) if TTY job is waiting for input.
26	.MOSPD	Set the terminal line speed. This function accepts in AC3 the desired line speed (input speed in the left half and output speed in the right half). The left half of AC2 contains flag bits indicating the type of line being set. If B0(MO%RMT) is on, the line is a remote (dataset) line. If B1(MO%AUT) is on, the line is a remote autobaud line (i.e., is automatically set at 300 baud, and the contents of AC3 are ignored. The process must have WHEEL or OPERATOR capability enabled to set B0(MO%RMT) and B1(MO%AUT).
27	.MORSP	Return the terminal line speed. On a successful return, the left half of AC2 contains flag bits indicating the type of line, and AC3 contains the speed (input speed in the left half and output speed in the right half). If B0(MO%RMT) of AC2 is on, the line is a remote line, and if B1(MO%AUT) is on, the line is a remote autobaud line. AC3 contains the speed or contains -1 if the speed is unknown or is not applicable.
30	.MORLW	Return the terminal page width. On a successful return, AC3 contains the width.
31	.MOSLW	Set the terminal page width. This function requires AC3 to contain the desired width.
32	.MORLL	Return the terminal page length. On a successful return, AC3 contains the length.
33	.MOSLL	Set the terminal page length. This function requires AC3 to contain the desired length.
34	.MOSNT	Specify if terminal line given in AC1 is to receive system messages. This function requires AC3 to contain 0 (.MOSMY) to allow messages or 1 (.MOSMN) to suppress messages.

35	.MORNT	Return a code indicating if terminal line given in AC1 is to receive system messages. On a successful return, AC3 contains 0 (.MOSMY) if messages are being sent to this line or 1 (.MOSMN) if messages are being suppressed to this line.
36	.MOSIG	Specify if input on this terminal line is to be ignored when the line is inactive (i.e., is not assigned or opened). This function requires AC3 to contain 0 if characters on this line are not to be ignored or 1 if characters on this line are to be ignored. When input is being ignored and characters are typed, no CTRL/G (bell) is sent, as is the normal case when characters are typed on an inactive line.

The functions available for DECnet-20 are described below. For a complete description of their application, refer to the TOPS-20 DECnet-20 Programmer's Guide and Operations Manual.

Code	Symbol	Meaning
24	.MOACN	Allow a network task to enable software interrupt channels for any combination of the following work types: o connect event pending o interrupt message available o data available This function requires that AC3 contain three 9-bit fields specifying the changes in the interrupt assignments for this link. These fields are

Field	Symbol	Used to Signal
B0-B8	MO%CDN	Connect event pending
B9-B17	MO%INA	Interrupt message available
B18-B26	MO%DAV	Data available

The contents of the fields are

Value	Meaning
nnn	The number of the channel to be enabled; 0-5 and 23-35 decimal
.MOCIA	Clear the interrupt
.MONCI	No change

25	.MORLS	Read the link status and return a 36-bit word of information regarding the status of the logical link. AC3 contains flag bits in the left half and a disconnect code in the right half. The flag bits are

Symbol	Bit	Meaning
MO%CON	B0	Link is connected
MO%SRV	B1	Link is a server
MO%WFC	B2	Link is waiting for a connect
MO%WCC	B3	Link is waiting for a connect confirm

MO%EOM	B4	Link has an entire message to be read
MO%ABT	B5	Link has been aborted
MO%SYN	B6	Link has been closed normally
MO%INT	B7	Link has an interrupt message available
MO%LWC	B8	Link has been previously connected

The disconnect/reject codes are as follows:

Symbol	Value	Meaning
.DCX0	0	No special error
.DCX1	1	Resource allocation failure
.DCX2	2	Destination node does not exist
.DCX3	3	Node shutting down
.DCX4	4	Destination process does not exist
.DCX5	5	Invalid name field
.DCX11	11	User abort (asynchronous disconnect)
.DCX32	32	Too many connections to node
.DCX33	33	Too many connections to destination process
.DCX34	34	Access not permitted
.DCX35	35	Logical link services mismatch
.DCX36	36	Invalid account
.DCX37	37	Segment size too small
.DCX38	38	Process aborted
.DCX39	39	No path to destination node
.DCX40	40	Link aborted due to data loss
.DCX41	41	Destination process does not exist
.DCX42	42	Confirmation of DISCONNECT INITIATE
.DCX43	43	Image data field too long

If a disconnect code does not apply to the current status of the link, the right half of AC3 will be zeros.

26 .MORHN Return the ASCII name of the host node at the other end of the logical link. This function requires that AC3 contain a string pointer to the location where the host name is to be stored. (If the byte size exceeds eight bits, bytes are truncated to eight bits.)

The monitor call returns with an updated pointer in AC3, and the host name stored as specified.

27 .MORTN Return the unique task name that is associated with your end of the logical link. If you had defaulted the task name in the network file specification, the call returns the monitor-supplied task name. In DECnet-20, the default task name is actually a unique number.

This function requires that AC3 contain a string pointer to the location where the task name is to be stored. (If the byte size exceeds eight bits, bytes are truncated to eight bits.)

The monitor call returns with an updated pointer is AC3 and the task name stored as specified.

30 .MORUS Return the source task user identification supplied in the connect initiate message. This function requires that AC3 contain a string pointer to the location where the user identification is to be stored. (If the byte size exceeds eight bits, bytes are truncated to eight bits.)

The monitor call returns with an updated pointer in AC3 and the user identification stored as specified. If no user identification was supplied by the source task, AC3 continues to point to the beginning of the string, and a null is returned as the only character.

31 .MORPW Return the source task's password as supplied in the connect initiate message. This function requires that AC3 contain a string pointer to the location where the password is to be stored. (Passwords are binary, therefore the string pointer should accomodate 8-bit bytes.)

The monitor call returns with an updated pointer in AC3 and the source task's password stored as specified. AC4 contains the number of bytes in the string; a zero value indicates that no password was supplied by the source task.

32 .MORAC Return the account string supplied by the source task in the connect initiate message. This function requires that AC3 contain a string pointer to the location where the account string is to be stored. (If the byte size exceeds eight bits, bytes are truncated to eight bits.)

The monitor call return with an updated pointer in AC3 and the source task's account number stored as specified. If no account string was supplied by the source task, AC3 continues to point to the beginning of the string, and a null is returned as the only character.

33 .MORDA Return the optional data supplied in any of the connect or disconnect messages. This function requires that AC3 contain a string pointer to the location where the optional user data is to be stored. (This file is binary; the string pointer should specify 8-bit bytes.)

The monitor call returns with an updated pointer in AC3 and the optional data stored as specified. AC4 contains the number of bytes in the data string; a zero value indicates that no optional data was supplied.

34 .MORCN Return the object type that was used by the source task to address this connection. The result indicates whether the local task was addressed by its generic type or its unique network task name.

The monitor call returns with the object type is AC3. A zero object type indicates that the target task was addressed by its unique network task name; a nonzero value indicates that it was addressed by its generic object type.

35 .MORIM Read interrupt message. This function requires that AC3 contain a byte pointer to the receiving buffer. (If the byte size exceeds eight bits, bytes are truncated to eight bits.) The maximum message length is 16 bytes, and the buffer size should be at least 8 bits.

The monitor call returns with an updated pointer in AC3, the message stored in the buffer, and the count of bytes received in AC4.

36 .MOSIM Send an interrupt message. This function requires that AC3 contain a byte pointer to the message (eight bytes maximum) and the AC4 contain a count of the bytes in the interrupt message (sixteen bytes maximum).

37 .MOROD Return the unique identification of the source task. This identification is in the format of object-descriptor, and the contents depend on the DECnet implementation on the remote host. In addition, if the source task is running on a system that provides for group and user codes, this information is also returned. If the source task name is on a DECnet-20 host, the data returned is TASK-taskname. This function requires that AC3 contain a string pointer to the location where the object-descriptor of the source task is to be stored. (If the byte size exceeds eight bits, bytes are truncated to eight bits.)

The monitor call returns with an updated pointer in AC3 and the object-descriptor stored as specified. In addition, if the source host system uses group and user codes (sometimes referred to as project and programmer number or p,pn), AC4 contains the group code in the left half and the user code in the right half. If the source host system does not provide for group or user codes, AC4 contains zeros.

40 .MOCLZ Reject a connection either implicitly or explicitly. If the target task closes its JFN (via the CLOSF monitor call) before accepting the connection either implicitly or explicitly, the local NSM assumes that the connection is rejected and sends a connect reject message back to the source task. The reason given is process aborted (reject code 38). The target task must the reopen its JFN in order to receive subsequent connect initiate messages.

In order to explicitly reject a connect and at the same time return a specific reject reason and set up 16 bytes of user data, the target task must use the .MOCLZ function of the MTOPR monitor call. The .MOLCZ function does not close the JFN.

This function requires that

AC2 contain a reject code in the left half and .MOCLZ in the right half. The reject code is a 2-byte, NSP-defined decimal number indicating the reason that a target task is rejecting a connection. Refer to the description of code 25, .MORLS, for a list of disconnect/reject codes.

AC3 contain a string pointer to any data to be returned. (If the byte size exceeds eight bits, bytes are truncated to eight bits.)

AC4 contain the count of bytes in the data string (maximum=16). A zero indicates no data

41 .MOCC Accept a connection either implicitly or explicitly. Under certain conditions, the local NSP assumes that the connection is accepted and sends a connect confirm message back to the source task. These implicit conditions are

The target task attempts to output to the logical link (issues a SOUT or SOUTR monitor call to the network).

The target task submits a read request to the logical link (issues a SIN or SINR monitor call to the network).

The target task is in input wait state (has enabled itself for a "data available" software interrupt).

In order to explicitly accept a connect and also return a limited amount of data, the target task must use the .MOCC function of the MTOPR monitor call. This function requires that AC3 contain a string pointer to any data to be returned. (If byte size exceeds eight bits, bytes are truncated to eight bits.) AC4 must contain the count of bytes in the data string to a maximum of 16 bytes. A zero indicates no data.

42 .MORSS Returns the maximum segment size that can be sent over this link. This value is the lesser of the maximum segment sizes supported by the remote NSP task and the remote network task. The local task can use this value to optimize the format of data being transmitted over the link.

The monitor call returns the maximum segment size, in bytes, in AC3.

Generates an illegal instruction interrupt on error conditions below.

MTOPR ERROR MNEMONICS:

DESX1:	invalid source/destination designator
DESX2:	terminal is not available to this job
DESX3:	JFN is not assigned
DESX4:	invalid use of terminal designator or string pointer
DESX5:	file is not open
IOX5:	device or data error
MTOX1:	invalid function
MTOX2:	record size was not set before I/O was done
MTOX3:	function not legal in dump mode
MTOX4:	invalid record size
MTOX5:	invalid hardware data mode for magnetic tape
MTOX6:	invalid magnetic tape density
MTOX7:	WHEEL or OPERATOR capability required
MTOX8:	argument block too long
MTOX9:	output still pending
MTOX10:	VFU or RAM file cannot be OPENed
MTOX11:	data too large for buffers
MTOX12:	input error or not all data read
MTOX13:	argument block too small
MTOX14:	invalid software interrupt channel number
MTOX15:	device does not have Direct Access (programmable) VFU

IPCFX4: receiver's PID invalid

IPCFX5: receiver's PID disabled

IPCFX6: send quota exceeded

IPCFX7: receiver quota exceeded

IPCFX8: IPCF free space exhausted

IPCFX9: sender's PID invalid

IPCF10: WHEEL capability required

IPCF11: WHEEL or IPCF capability required

IPCF12: no free PID's available

IPCF13: PID quota exceeded

IPCF14: no PID's available to this job

IPCF15: no PID's available to this process

IPCF16: receive and message data modes do not match

IPCF17: argument block too small

IPCF18: invalid MUTIL JSYS function

IPCF19: no PID for [SYSTEM]INFO

IPCF20: invalid process handle

IPCF21: invalid job number

IPCF22: invalid software interrupt channel number

IPCF23: [SYSTEM]INFO already exists

IPCF24: invalid message size

IPCF25: PID does not belong to this job

IPCF26: PID does not belong to this process

IPCF27: PID is not defined

IPCF28: PID not accessible by this process

IPCF29: PID already being used by another process

IPCF30: job is not logged in

IPCF32: page is not private

IPCF33: invalid index into system PID table

IPCF35: invalid IPCF quota

NIN JSYS 225

Inputs an integer number, with leading spaces ignored. This call terminates on the first character not in the specified radix. If that character is a carriage return followed by a line feed, the line feed is also input.

ACCEPTS IN AC1: source designator

AC3: radix (2-10) of number being input

RETURNS +1: failure, error code in AC3, updated string pointer, if pertinent, in AC1

+2: success, number in AC2 and updated string pointer, if pertinent, in AC1

NIN ERROR MNEMONICS:

IFIXX1: radix is not in range 2 to 10

IFIXX2: first nonspace character is not a digit

IFIXX3: overflow (number is greater than 2**35)

DESX1: invalid source/destination designator

DESX2: terminal is not available to this job

DESX3: JFN is not assigned

DESX5: file is not open

NODE JSYS 567

Performs the following network utility functions: set local node name, get local node name, set local node number, get local node number, set loopback port, clear loopback port, and find loopback port.

ACCEPTS IN AC1: function code

AC2: address of argument block

RETURNS +1: always. If an error occurs, an illegal instruction trap is generated.

The available functions and their argument blocks are described below.

0 .NDSLN Set local node name

Argument Block

0 .NDNOD Pointer to ASCIZ node name.

1 .NDGLN Get local node name

Argument Block

0 .NDNOD Pointer to destination for ASCIZ name of local node.

2 .NDSNM Set local node number

Argument Block

0 Number to set (greater than 2, less than 127)

3 .NDGNM Get local node number.

4 .NDSLP Set loopback port[1]

Argument Block

0 .NDPRT NSP port number.

1 .BTLID Pointer to line id.

5 .NDCLP Clear loopback port[1]

Argument Block

0 .NDPRT NSP port number.

6 .NDFLP Find loopback port[1]

Argument Block

0 .NDPRT NSP port number

1B0 (ND%LPR) Loopback running.

1B1 (ND%LPA) Loopback port assigned.

NODE ERROR MNEMONICS:

ARGX02: Invalid function

ARGX19: Invalid unit number

CAPX2: WHEEL, OPERATOR, or MAINTENANCE capability required

NODX02: Line not turned off

NODX03: Another line already looped

[1] DECSYSTEM-2020 only.

NOUT JSYS 224

Outputs an integer number.

ACCEPTS IN AC1: destination designator

AC2: number to be output

AC3: B0(NO%MAG) output the magnitude. That is, output the number as an unsigned 36-bit number (e.g., output -1 as 777777 777777).

B1(NO%SGN) output a plus sign for a positive number.

B2(NO%LFL) output leading filler. If this bit is not set, trailing filler is output, and bit 3(NO%ZRO) is ignored.

B3(NO%ZRO) output 0's as the leading filler if the specified number of columns (NO%COL) allows filling. If this bit is not set, blanks are output as leading filler if the number of columns allows filling.

PBOUT JSYS 74

Outputs a byte sequentially to the primary output designator. This call is equivalent to a BOUT call with the destination designator given as .PRIOU.

ACCEPTS IN AC1: byte to be output, right-justified

RETURNS +1: always

Can cause several software interrupts or process terminations on certain file conditions. (Refer to bit OF%HER of the OPENF call description.)

PBOUT ERROR MNEMONICS:

DESX1: invalid source/destination designator

DESX2: terminal is not available to this job

DESX3: JFN is not assigned

DESX5: file is not open

IOX2: file is not open for writing

IOX5: device or data error

IOX6: illegal to write beyond absolute end of file

IOX11: quota exceeded or disk full

PEEK JSYS 311

Transfers a block of words from the monitor to the user space. The desired monitor pages must have read access. This monitor call is used to obtain data from the monitor for maintenance and test purposes and should be executed only when GETAB information is not available.

ACCEPTS IN AC1: word count in the left half, and first virtual address of the monitor in the right half

AC2: first user address

RETURNS +1: failure, error code in AC1

+2: success, the desired words are transferred.

The PEEK monitor call requires the process to have the MAINTENANCE, WHEEL, or OPERATOR capability enabled.

PEEK ERROR MNEMONICS:

CAPX1: WHEEL or OPERATOR capability required

PEEKX2: read access failure on monitor page

PLOCK JSYS 561

Acquires physical memory and places a designated section of the process' address space in memory. Allows the process to specify the memory pages to be used, or permits the system to select the pages. The PLOCK monitor call requires the process to have WHEEL, OPERATOR, or MAINTENANCE capability enabled.

ACCEPTS IN AC1: address of first page if acquiring (locking) or -1 if unlocking.

AC2: process handle (currently .FHSLF only) in the left half and number of first page in the right half.

AC3: control flags in the left half and repeat count in the right half. The control flags are

B0 (LK%CNT) right half of AC3 contains a count of the number of pages to lock.

B1 (LK%PHY) value in AC1 is the first page desired. If this bit is off and AC1 is not -1, the system selects pages.

B2 (LK%NCH) pages will not be cached.

B3 (LK%AOL) off-line pages are to be locked.

RETURNS +1: always

If the PLOCK call is unable to honor any one of the requests to unlock any one of the pages specified by the repeat count, it will unlock all of the others.

A page that was locked with the PLOCK call may be unmapped. (Refer to the PMAP call.) This will unlock the process' page and return the now unlocked physical page to its previous state.

The page selected by the user must be capable of being placed off-line for the PLOCK call to acquire it.

Generates an illegal instruction interrupt on error conditions below.

PLOCK ERROR MNEMONICS:

ARGX22: invalid flag

ARGX24 invalid count

PMAP JSYS 56

Maps one or more complete pages from a file to a process (for input), from a process to a file (for output), or from one process to another process. Each of the three uses of PMAP is described below.

Case I Mapping File Pages to a Process

This use of the PMAP call does not actually transfer any data; it simply changes the contents of the process' page map. When changes are made to the page in the process, the changes will also be reflected in the page in the file, if write access has been specified for the file.

ACCEPTS IN AC1: JFN of the file in the left half, and the page number in the file in the right half. This AC contains the source.

AC2: process handle in the left half, and the page number in the process in the right half. This AC contains the destination.

AC3:

B0(PM%CNT)	A count is in the right half of AC3. This count specifies the number of sequential pages to be mapped.
B2(PM%RD)	Permit read access to the page.
B3(PM%WR)	Permit write access to the page.
B4(PM%EX)	Reserved for future use. The symbol PM%RWX can be used to set B2-B4.
B5(PM%PLD)	Preload the page being mapped (i.e., move the page immediately instead of waiting until it is referenced).
B9(PM%CPY)	Create a private copy of the page when it is written into (copy-on-write). If the page is mapped between two processes (Case III below), both processes will receive a private copy of the page.
B18-B35 (PM%RPT)	Number of pages to be mapped if B0(PM%CNT) is set.

RETURNS +1: always

This use of PMAP changes the map of the process such that addresses in the page in the process specified by the right half of AC2 actually refer to the page in the file specified by the right half of AC1. The present contents of the page in the process are removed. If the page in the file is currently nonexistent, it will be created when it is written (i.e., when the corresponding page in the process is written).

This use of PMAP is legal only if the file is opened for at least read access. The access bits specified in the PMAP call are ANDed with the access that was specified when the file was opened. However, copy-on-write is always granted, regardless of the file's access. The access granted is placed in the process' map.

The file cannot be closed while any of its pages are mapped into any process. Thus, before the file is closed, pages must be unmapped from each process by a PMAP call with -1 in AC1 (see below).

Case II Mapping Process Pages to a File

This use of the PMAP call actually transfers data by moving the contents of the specified page in the process to the specified page in the file. The process' map for that page becomes empty.

ACCEPTS IN AC1: process handle in the left half, and the page number in the process in the right half. This AC contains the source.

AC2: JFN of the file in the left half, and the page number in the file in the right half. This AC contains the destination.

AC3: access bits and repetition count. (Refer to Case I.)

RETURNS +1: always

The process page and the file page must be private pages. The ownership of the process page is transferred to the file page. The present contents of the page in the file is deleted.

The access granted to the file page is determined by ANDing the access specified in the PMAP call with the access specified when the file was opened.

When mapping pages from a process to a file, the end-of-file byte pointer and the byte size are not automatically updated in the File Descriptor Block (FDB). To allow the file to be read later via the sequential I/O calls (e.g., BIN, SIN), the process executing the PMAP call should close the file keeping the JFN (CLOSF call, bit CO%NRJ), update the byte pointer and the byte size in the FDB (CHFDB call), and then release the JFN (RLJFN call). (Refer to Section 2.2.8 for the format of the FDB fields.)

Case III Mapping One Process' Pages to Another Process

This use of the PMAP call normally does not transfer any data; it simply changes the contents of the page maps of the processes. When changes are made to the page in one process, the changes will also be reflected in the corresponding page in the other process.

ACCEPTS IN AC1: process handle in the left half, and the page number in the process in the right half. This AC contains the source.

AC2: a second process handle in the left half, and page number in that process in the right half. This AC contains the destination.

AC3: access bits and repetition count. (Refer to Case I.)

RETURNS +1: always

This use of PMAP changes the map of the destination process such that addresses in the page specified by the right half of AC2 actually refer to the page in the source process specified by the right half of AC1. The present contents of the destination page are deleted.

The access granted to the destination page is determined by the access specified in the PMAP call.

Unmapping Pages In a Process

As stated previously, a file cannot be closed if any of its pages are mapped in any process. To unmap the file's pages from a process, a PMAP call is executed with

AC1: -1

AC2: process handle in the left half, and page number in the process in the right half

AC3: B0(PM%CNT) Repeat count. Only the process page numbers are incremented.

B18-B35 Number of pages to remove from process

This format of the PMAP call removes the pages indicated in AC2 from the process.

A page that was locked with the PLOCK call may be unmapped. Doing so will unlock the process' page and return the now unlocked physical page to its previous state.

Illegal PMAP calls

The PMAP call is illegal if:

1. Both AC1 and AC2 designate files.
2. Both AC1 and AC2 are 0.
3. The PMAP call designates a file with write-only access.
4. The PMAP call designates a file with append-only access.

Can cause several software interrupts on certain file conditions.

Generates an illegal instruction interrupt on error conditions below.

PMAP ERROR MNEMONICS:

DESX1: invalid source/destination designator

DESX3: JFN is not assigned

DESX5: file is not open

DESX7: JFN cannot refer to output wildcard designators

PMAPX1: invalid access requested

PMAPX2: invalid use of PMAP

PMAPX3: illegal to move shared page into file

PMAPX4: illegal to move file page into process

PMAPX5: illegal to move special page into file

PMAPX6: disk quota exceeded

PMAPX7: illegal to map file on dismounted structure

FRKHX1: invalid process handle

FRKHX2: illegal to manipulate a superior process

FRKHX3: invalid use of multiple process handle

FRKHX7: process page cannot exceed 777

LNGFX1: page table does not exist and file not open for write

IOX11: quota exceeded or disk full

ARGX06: invalid page number

PMCTL JSYS 560

Controls physical memory. This call allows a privileged program to add or remove most pages of physical memory and to control use of cache memory.

The PMCTL monitor call requires the process to have WHEEL, OPERATOR, or MAINTENANCE capability enabled.

ACCEPTS IN AC1: function code

AC2: length of the argument block

AC3: address of the argument block

RETURNS +1: always

The defined functions and their argument blocks are as follows:

Function	Symbol	Meaning
0	.MCRCE	Return the status of cache memory. The status is returned in word .MCCST of the argument block.
		Argument Block
		0 .MCCST If B35(MC%CEN) is on, the cache is enabled.
1	.MCSCE	Set the status of cache memory.
		Argument Block
		0 .MCCST Enable the cache if B35(MC%CEN) is on.

2 .MCRPS Return the status of the given page(s). The number of the page is given in word .MCPPN, and its status is returned in word .MCPST.

Argument Block

0 .MCPPN Negative count in left half; number of physical page in right half

1 .MCPST Returned page status. The status is represented by one of the following values:

0 .MCPSA Page is available for normal use.

1 .MCPSS Page is in a transition state.

2 .MCPSO Page is off line because it is nonexistent. Nonexistent memory is marked as off line at system startup.

3 .MCPSE Page is off line because the monitor detected an error.

3 .MCSPS Set the status of the given page. The number of the page is given in word .MCPPN, and the status value is given in word .MCPST.

Argument Block

0 .MCPPN Number of physical page.

1 .MCPST Status for page. The status is represented by one of the following values:

0 .MCPSA Mark page available for normal use.

2 .MCPSO Mark page off line because it does not exist.

3 .MCPSE Mark page off line because it has an error.

4 .MCRME Collect information about MOS memory errors. Store the information in block addressed by AC3 and update AC2 on return.

A list of those pages that PMCTL cannot acquire follows:

the EPT
the monitor's UPT
any page containing a CST0 entry
any page containing an SPT entry
the page containing MMAP
any page belonging to the resident free space pool

In certain specialized monitors, for example TOPS-20AN, there are additional pages that cannot be acquired. An estimate of the size of these areas follows:

CST0	one word for every page of memory supported (two to four pages)
SPT	four pages
MMAP	one page
Resident Free Space Pool	two pages minimum

Generates an illegal instruction interrupt on error conditions below.

PMCTL ERROR MNEMONICS:

CAPX2: WHEEL, OPERATOR, or MAINTENANCE capability required

PMCLX1: invalid page state or state transition

PMCLX2: requested physical page is unavailable

PSOUT ERROR MNEMONICS:

DESX1: invalid source/destination designator

DESX2: terminal is not available to this job

DESX3: JFN is not assigned

DESX5: file is not open

IOX2: file is not open for writing

IOX5: device or data error

IOX6: illegal to write beyond absolute end of file

IOX11: quota exceeded or disk full

RCDIR JSYS 553

Translates the given directory string to its corresponding 36-bit directory number. The directory string consists of the structure name or logical name and a colon followed by the directory name enclosed in either square brackets or angle brackets. No spaces can appear between the structure name and the directory name, and each field given must include its punctuation. An example of a directory string is PS:<SMITH>. If the structure name is omitted from the string, the user's connected structure is used. If the directory name is omitted from the string, the user's connected directory is used.

Recognition can be used on the string but only on the directory name field; recognition cannot be used on the structure name field. Partial recognition can be allowed so that a user can employ recognition when typing the name of a subdirectory. When recognition is used on the directory name field and the directory name is not ambiguous, the closing bracket is not required.

The directory name field can contain wildcard characters, and repeated RCDIR calls can be executed to obtain the numbers of the directories whose characters match the given directory. After the first call, each subsequent RCDIR call returns the number of the next directory in the group.

ACCEPTS IN AC1: flag bits in the left half

AC2: pointer to ASCIZ string to be translated, a JFN, a 36-bit user number, or a 36-bit directory number (given for the purpose of checking its validity)

AC3: 36-bit directory number (given when stepping to the next directory in a group of directories)

RETURNS +1: always, with

AC1 containing flag bits in the left half

AC2 containing an updated string pointer (if a pointer was supplied as the argument). If recognition was used, this pointer reflects the remainder of the string that was appended to the original string.

AC3 containing a 36-bit directory number if execution of the call was successful

The flag bits supplied in the left half of AC1 are as follows:

B14 (RC%PAR) Allow partial recognition on the directory name. If the name given matches more than one directory, bit RC%AMB is set on return and the string is updated to reflect the unique portion of the directory name. If bit RC%PAR is not set, the name given matches more than one directory, and recognition is being used, bit RC%AMB is set on return, but the string is not updated.

B15 (RC%STP) Step to the next directory in the group and return the number of that directory. AC1 must have bit RC%AWL set. AC2 must contain a pointer to a string that contains wildcard characters in the directory name field. AC3 must contain a directory number.

B16 (RC%AWL) Allow the directory name to contain wildcard characters. No recognition is performed on a directory name that contains wildcard characters. Also, the directory name must include its terminating bracket. This bit must be set if bit RC%STP is also set.

B17 (RC%EMO) Match the given string exactly. When both the RC%PAR and RC%EMO bits are on, recognition is not used on the string, and the string is matched exactly. If this bit is off, recognition is used on the string.

The flag bits returned in the left half of AC1 are as follows:

On success

B0 (RC%DIR) Directory can be used only by connecting to it (i.e., it is a files-only directory). If this bit is off, the user can also login to (if the directory is on the public structure) or access this directory.

B1 (RC%ANA) Obsolete

B2 (RC%RLM) All messages from <SYSTEM>MAIL.TXT are repeated every time the user logs in. If this bit is off, messages are printed only once.

B6 (RC%WLD) The directory name given contained wildcard characters.

RCUSR JSYS 554

Translates the given user name string to its corresponding 36-bit user number. The user name string consists of the user's name without any punctuation. The string must be associated with a directory on structure PS: that is not a files-only directory.

Recognition can be used on the string. In addition, the string can contain wildcard characters.

ACCEPTS IN AC1: flag bits in the left half

AC2: pointer to ASCII string to be translated

AC3: 36-bit user number (given when stepping to the next user name in a group)

RETURNS +1: always, with

AC1 containing flag bits in the left half

AC2 containing an updated string pointer. If recognition was used, this pointer reflects the remainder of the string that is appended to the original string.

AC3 containing a 36-bit user number if execution of the call was successful

The flag bits supplied in the left half of AC1 are as follows. For additional information on these bits, refer to the RCDIR monitor call description.

B14 (RC%PAR) Allow partial recognition on the user name string.

B15 (RC%STP) Step to the next user name in the group.

B16 (RC%AWL) Allow the user name to contain wildcard characters.

B17 (RC%EMO) Match the given string exactly.

The flag bits returned in the left half of AC1 are as follows. For additional information on these bits, refer to the RCDIR monitor call description.

On success

B1 (RC%ANA) Obsolete

B2 (RC%RLM) User sees all messages from <SYSTEM>MAIL.TXT every time he logs in. If this bit is off, the user sees the messages only once.

B6 (RC%WLD) The user name given contained wildcard characters.

On failure

B3(RC%NOM) No match was found for the string given. This bit will be on if the string given refers to a files-only directory, if there is no directory on PS: that is associated with the user name string, or bit RC%EMO was on in the call and a string was given that matched more than one user.

B4(RC%AMB) The string given was ambiguous because it matched more than one user.

B5(RC%NMD) There are no more user names in the group.

The RCDIR monitor call can be used to translate a directory string to its corresponding directory number. The DIRST monitor call can be used to translate either a user number or a directory number to its corresponding string.

Generates an illegal instruction interrupt on error conditions below.

RCUSR ERROR MNEMONICS:

RCUSX1: insufficient system resources

RCDIX4: monitor internal error

STRX07: invalid user number

STRX08: invalid user name

RDTTY JSYS 523

Reads input from the primary input designator (.PRIIN) into the caller's address space. Input is read until either a break character is encountered or the given byte count is exhausted, whichever occurs first. Output generated as a result of character editing is output to the primary output designator (.PRIOU).

The RDTTY call handles the following editing functions:

1. Delete the last character input (DELETE).
2. Delete back to the last punctuation character (CTRL/W).
3. Delete back to the beginning of the current line or, if the current line is empty, back to the beginning of the previous line (CTRL/U).
4. Retype the current line from its beginning or, if the current line is empty, retype the previous line (CTRL/R).
5. Accept the next character without regard to its usual meaning (CTRL/V).

By handling these functions, the RDTTY call serves as an interface between the terminal and the user program.

.SJDM(2)		Set default for magnetic tape data mode.
	.SJDDM(0)	System default data mode
	.SJDMC(1)	Dump mode
	.SJDM6(2)	SIXBIT byte mode (7-track drives)
	.SJDMA(3)	ANSI ASCII mode (7 bits in 8-bit bytes)
	.SJDM8(4)	Industry compatible mode
.SJRS(3)		Set default for magnetic tape record size in words.
.SJDFS(4)		Set spooling mode.
	.SJSPI(0)	Immediate mode spooling
	.SJSPD(1)	Deferred mode spooling
.SJSRM(5)		Set remark for current job session. AC3 contains a pointer to the session remark, which is updated on a successful return. The first 39 characters of the session remark are placed in the job's Job Storage Block.

The SETJB monitor call requires the process to have WHEEL or OPERATOR capability enabled to set parameters for a job other than the current job.

The GETJI monitor call can be used to obtain the job parameters for a specified job.

Generates an illegal instruction interrupt on error conditions below.

SETJB ERROR MNEMONICS:

SJBX1: invalid function

SJBX2: invalid magnetic tape density

SJBX3: invalid magnetic tape data mode

SJBX4: invalid job number

SJBX5: job is not logged in

SJBX6: WHEEL or OPERATOR capability required

SJBX7: remark exceeds 39 characters

SJBX8: illegal to perform this function

SETNM JSYS 210

Sets the private name of the program being used by the current job. This name is the one printed on SYSTAT listings.

ACCEPTS IN AC1: SIXBIT name used to identify program

RETURNS +1: always

The GETNM monitor call can be used to obtain the name of the program currently being used.

SETSN JSYS 506

Sets either the system name or the private name of the program being used by the current job.

ACCEPTS IN AC1: SIXBIT name to be used as the system name. This name is the one used for system statistics.

AC2: SIXBIT name to be used as the private name. This name is the same as the one set with the SETNM call.

RETURNS +1: failure. (Currently, there are no failure returns defined.)

+2: success

System program usage statistics are accumulated in the system tables SNAMES, STIMES, and SPFLTS. (Refer to Section 2.3.2.) To make this possible, the SETSN call must be executed by each job whenever the system program name is changed. In the usual case, the TOPS-20 Command Language handles this. The argument to SETSN should be: for system programs (programs from directory <SUBSYS>), the filename, truncated to six characters and converted to SIXBIT; for private programs, "(PRIV)."

SEVEC JSYS 204

Sets the entry vector of the specified process. (Refer to Section 2.7.3.)

ACCEPTS IN AC1: process handle

AC2: entry vector word (length in the left half and address of first word in the right half), or 0

RETURNS +1: always

The GEVEC monitor call can be used to obtain the process' entry vector.

SFMOD JSYS 110

Sets the program-related modes for the specified terminal. The modes that can be set by this call are in the following bits of the JFN mode word. (Refer to Section 2.4.3.1.)

```
B0 (TT%OSP)       output suppression control
B18-B23 (TT%WAK)  wakeup control
B24 (TT%ECO)      echoes on
B28-B29 (TT%DAM)  data mode
```

ACCEPTS IN AC1: file designator

AC2: JFN mode word

RETURNS +1: always

The SFMOD call is a no-op if the designator is not associated with a terminal.

The STPAR monitor call can be used to set device-related modes of the JFN mode word, and the RFMOD monitor call can be used to obtain the JFN mode word.

SFMOD ERROR MNEMONICS:

DESX1: invalid source/destination designator

DESX3: JFN is not assigned

DESX5: file is not open

DEVX2: device already assigned to another job

TTYX01: line is not active

SFORK JSYS 157

Starts the specified process. If the process is frozen, the SFORK call changes the PC but does not resume the process. The RFORK call must be used to resume the process.

ACCEPTS IN AC1: process handle

AC2: the PC of the process being started.

RETURNS +1: always

The SFRKV monitor call can be used to start a process at a given position in its entry vector.

Generates an illegal instruction interrupt on error conditions below.

SFORK ERROR MNEMONICS:

FRKHX1: invalid process handle

FRKHX2: illegal to manipulate a superior process

FRKHX3: invalid use of multiple process handle

SFPOS JSYS 526

Sets the position of the specified terminal's pointer. (Refer to Section 2.4.3.4 for information on page lengths and widths of terminals.)

ACCEPTS IN AC1: file designator

AC2: position within a page (i.e., line number) in the left half, and position with a line (i.e., column number) in the right half

RETURNS +1: always

The SFPOS monitor call is a no-op if the designator is not associated with a terminal or is in any way illegal.

The RFPOS monitor call can be used to obtain the current position of the terminal's pointer.

SFPOS ERROR MNEMONICS:

TTYX01: line is not active

SFPTR JSYS 27

Sets the position of the specified file's pointer for subsequent I/O to the file. The SFPTR call specifying a certain byte number, followed by a BIN call, has the same effect as a RIN call specifying the same byte number.

ACCEPTS IN AC1: JFN

AC2: byte number to which the pointer is to be set, or -1 to set the pointer to the current end of the file

RETURNS +1: failure, error code in AC1

+2: success

The RFPTR monitor call can be used to obtain the current position of the file's pointer.

11	.SFBTE	Bit table errors found on startup.
12	.SFCRD	Users can change nonprivileged directory parameters with the CRDIR monitor call.
13	.SFNVT	ARPANET terminal LOGINs are allowed.
21	.SFUSG	USAGE file entries are allowed.
22	.SFFLO	Disk latency optimization using the RH20 backup register is enabled. This feature is not to be enabled unless the M8555 board of the RH20 is at Revision Level D AND either of the KL10-C processor is at Revision Level 10 or KL10-E processor is at Revision Level 2.
44	.SFNTN	Turn ARPANET on.
45	.SFNDU	Reinitialize ARPANET if it is down.
46	.SFNHI	Initialize ARPANET host table.
47	.SFTMZ	Set the local time zone to the value given in AC2.
50	.SFLHN	Set the local ARPANET host number to the value given in AC2.
51	.SFAVR	Account validation will be running on this system.
52	.SFSTS	Enable/disable status reporting.

Function codes 0 through 22 represent a specific monitor flag bit. When the value of the function is 1 (i.e., AC2 contains the value 1), the bit corresponding to the function is set. When the value is 0, the bit is cleared.

The TMON monitor call can be used to obtain the settings of the various monitor flags.

Generates an illegal instruction interrupt on error conditions below.

SMON ERROR MNEMONICS:

SMONX1: WHEEL or OPERATOR capability required

SMONX2: invalid SMON function

SNOOP JSYS 516

Performs system performance analysis. The SNOOP call requires the process to have WHEEL or OPERATOR capability enabled, because the process can patch any instruction in the monitor with this call. For example, the user program can build a PC histogram by patching an instruction in the code for the 1.0-millisecond clock.

The general procedure for using the SNOOP call is as follows:

1. The user program supplies a set of breakpoint routines that are called by the monitor when control reaches one of the patched instructions. These routines are mapped into the monitor's address space into an area selected by the monitor. Thus, the routines must have self-relocating code or must be relocated by the user program to where they will be run, based on the monitor address supplied by the monitor.

2. The user program defines a number of breakpoints, analogous to DDT breakpoints.

3. The user program inserts all of the breakpoints simultaneously.

4. The user program goes to "sleep" or waits for terminal input while its breakpoint routines obtain control.

5. When the user program determines that the routines have completed, it removes the breakpoints.

The user program breakpoint routines run in the monitor address space, which means that the addresses of the code and the data are monitor addresses. The user program must modify these addresses, based on the values returned by the monitor, after the initialization but before the "snooping." The breakpoint routines must preserve any accumulators they use. Also, they must not cause a page fault if at interrupt level or if a patch has been made in the page fault handler or in the scheduler. Thus, the breakpoint routines should test for swappable code being in memory before referencing it. If swappable code needs to be referenced, the swappable monitor can be locked in memory, if desired. When a patch is made to a routine called at many interrupt levels, the program must specify a reentrant instruction to be used for patching.

ACCEPTS IN AC1: function code

AC2: AC3: AC4: arguments for the specified function

RETURNS +1: failure, error code in AC1

+2: success

The following functions are available:

Function Code	Symbol	Meaning
0	.SNPLC	Declare and lock code into the monitor's address space. AC2: number of pages desired AC3: page number in user space of start of breakpoint routines to be locked

On return, the pages are locked contiguously in the monitor's address space, and AC2 contains the monitor page numbers corresponding to the given user page number.

1 .SNPLS Lock the swappable monitor. This function is useful for analyzing swappable data at interrupt level. On return, the entire swappable monitor is locked.

2 .SNPDB Define a breakpoint

AC2: number of breakpoint

AC3: address in monitor space to be patched. The patched instruction can be a skip type instruction or a PUSHJ instruction, and the patching is similar to that in DDT. The routines will receive control before the patched instruction is executed.

AC4: instuction to be executed before the patched instruction is executed. The instruction can be:

JSR LOC where LOC is an address in monitor space of the user's routine.

PUSHJ P,LOC when reentrant or recursive code is patched.

AOS LOC to count frequency of monitor execution points.

The error return is given if breakpoints have already been inserted.

3 .SNPIB Insert all breakpoints and start analyzing.

4 .SNPRB Remove all breakpoints and stop analyzing.

5 .SNPUL Unlock and release all storage, and undefine and remove all breakpoints.

6 .SNPSY Obtain the address of a monitor symbol.

AC2: radix-50 symbol

AC3: radix-50 program name if a local address is desired. If AC3 is 0, the entire symbol table is searched.

On return, AC2 contains the monitor address or value of the symbol.

7 .SNPAD Obtain a monitor symbol.

AC2: 36-bit value of symbol that is to be looked up in the monitor's symbol table.

AC3: radix-50 program name if a local value is desired. If AC3 is 0, the entire symbol table is searched.

On return, AC2 contains the first radix-50 monitor symbol that is closest to and has a value less than the specified value, and AC3 contains the difference between the value of the symbol returned and the specified value.

SNOOP ERROR MNEMONICS:

SNOPX1: WHEEL or OPERATOR capability required

SNOPX2: invalid function

SNOPX3: .SNPLC function must be first

SNOPX4: only one .SNPLC function allowed

SNOPX5: invalid page number

SNOPX6: invalid number of pages to lock

SNOPX7: illegal to define breakpoints after inserting them

SNOPX8: breakpoint is not set on instruction

SNOPX9: no more breakpoints allowed

SNOP10: breakpoints already inserted

SNOP11: breakpoints not inserted

SNOP12: invalid format for program name symbol

SNOP13: no such program name symbol

SNOP14: no such symbol

SNOP15: not enough free pages for snooping

SNOP16: multiply-defined symbol

SNOP17: breakpoint already defined

SNOP18: data page is not private or copy-or-write

B4(TL%SAB) Examine B5(TL%ABS) to determine the setting of the object designator's accept link bit. If this bit is off, B5 is ignored.

B5(TL%ABS) Set the object designator's accept link bit. When B4(TL%SAB) is on, the object designator is accepting links if TL%ABS is on and refusing links if TL%ABS is off.

B6(TL%STA) Examine B7(TL%AAD) to determine the setting of the object designator's accept advice bit. If this bit is off, B7 is ignored.

B7(TL%AAD) Set the object designator's accept advice bit. When B6(TL%STA) is on, the object designator is accepting advice if TL%AAD is on and refusing advice if TL%ADD is off.

B18-B35 (TL%OBJ) Object designator

AC2: remote designator

RETURNS +1: failure, error code in AC1

+2: success

The object and remote designators must be either 4xxxxx or -1. An object designator of -1 indicates the controlling terminal.

The following restrictions apply if the process does not have WHEEL capability enabled:

1. The object designator must specify a terminal assigned to this job.

2. The object-to-remote link must be specified before or at the same time as the remote-to-object link.

If the accept bit of the remote designator is not set, a link from the object-to-remote designator causes the remote designator's bell to ring. If the remote designator does not set the accept bit within 15 seconds, the TLINK call returns an error.

When terminals are linked together and a character is typed on one terminal, the same ASCII character code is sent to all terminals in the link. The character always appears in the output buffers of all terminals regardless of the current mode of each individual terminal. The character is sent according to the data mode and terminal type of the terminal that originates the character. For example, if one terminal originates a TAB and has mechanical tabs set, all terminals in the link receive the ASCII code for a TAB in their output buffers.

TLINK ERROR MNEMONICS:

DESX1: invalid source/destination designator

TLNKX1: illegal to set remote to object before object to remote

TLNKX2: link was not received within 15 seconds

TLNKX3: links full

TTYX01: line is not active

TMON JSYS 7

Tests various monitor flags.

ACCEPTS IN AC1: function code

RETURNS +1: always, value of the function in AC2

The codes for the functions are as follows:

Code	Symbol	Meaning
0	.SFFAC	FACT files entries are allowed.
1	.SFCDE	CHECKD found errors.
2	.SFCDR	CHECKD is running.
3	.SFMST	Manual start is in progress.
4	.SFRMT	Remote LOGINS (dataset lines) are allowed.
5	.SFPTY	PTY LOGINS are allowed.
6	.SFCTY	CTY LOGINS are allowed.
7	.SFOPR	Operator is in attendance.
10	.SFLCL	Local LOGINS (hardwired lines) are allowed.
11	.SFBTE	Bit table errors found on startup.
12	.SFCRD	Users can change nonprivileged directory parameters with the CRDIR monitor call.
13	.SFNVT	ARPANET terminal LOGINs are allowed.
21	.SFUSG	USAGE file entries are allowed.
22	.SFFLO	Disk latency optimization using the RH20 backup register is enabled.
44	.SFNTN	ARPANET is on.
45	.SFNDU	ARPANET will be reinitialized if it is down.
46	.SFNHI	ARPANET host table will be initialized.
47	.SFTMZ	Local time zone is set.
50	.SFLHN	Local ARPANET host number is set.
51	.SFAVR	Account validation is running on this system.
52	.SFSTS	Status reporting is enabled

Functions 0 through 22 represent a specific monitor flag bit. When the value of the function returned in AC2 is 1, the flag corresponding to the function is set. When the value returned is 0, the flag is not set.

The SMON monitor call can be used to set various monitor flags.

Generates an illegal instruction interrupt on error conditions below.

TMON ERROR MNEMONICS:

TMONX1: invalid TMON function

APPENDIX A

MONSYM.MAC

This appendix contains the complete copy of the system file MONSYM.MAC, which defines the symbols used in the manual. The user must include the statement

```
SEARCH MONSYM
```

in his program to have the symbols defined in his assembly.

```
;THIS SOFTWARE IS FURNISHED UNDER A LICENSE AND MAY ONLY BE USED
;  OR COPIED IN ACCORDANCE WITH THE TERMS OF SUCH LICENSE.
;
;COPYRIGHT (C) 1976, 1977, 1978 BY DIGITAL EQUIPMENT CORPORATION, MAYNARD, MASS.

;MONITOR CALL DEFINITIONS AND ERROR MNEMONICS

;NOTE:
;       THE FOLLOWING SYMBOLS ARE RESERVED:
;
;               SYMBOL          RESERVED BY
;               ======          ===========
;
;               .OF???          RMS-20
;               .SZ???          RMS-20
;               .PS???          RMS-20

;MACRO TO DEFINE JSYS NAMES

DEFINE DEFJS (NAME,NUM,SECT,XTRA)<
        OPDEF NAME [104B8+NUM]
        IFDEF .PSECT,<
        INTERN NAME>>

        SALL

IFNDEF REL,<REL==0>             ;ASSEMBLING REL IF NON-0
  IFE REL,<
        UNIVERSAL MONSYM>
   IFN REL,<
        TITLE MONSYM
        IFNDEF .PSECT,<
        .DIRECT .XTABM>
        >
```

```
;JSYS DEFINITIONS WITH 'NIM' AS A FOURTH ARGUMENT ARE CLASSIFIED
; AS 'NOT IN MONITOR'

DEFINE JSLIST <

DEFJS JSYS,0,,NIM

DEFJS LOGIN,1,MSEC1
DEFJS CRJOB,2,MSEC1
DEFJS LGOUT,3,MSEC1
DEFJS CACCT,4,MSEC1
DEFJS EFACT,5,MSEC1
DEFJS SMON,6,MSEC1
DEFJS TMON,7,MSEC1
DEFJS GETAB,10,MSEC1
DEFJS ERSTR,11,MSEC1
DEFJS GETER,12,MSEC1
DEFJS GJINF,13,MSEC1
DEFJS TIME,14,MSEC1
DEFJS RUNTM,15,MSEC1
DEFJS SYSGT,16,MSEC1
DEFJS GNJFN,17,MSEC1
DEFJS GTJFN,20,MSEC1
DEFJS OPENF,21,MSEC1
DEFJS CLOSF,22,MSEC1
DEFJS RLJFN,23,MSEC1
DEFJS GTSTS,24,MSEC1
DEFJS STSTS,25,MSEC1
DEFJS DELF,26,MSEC1
DEFJS SFPTR,27,MSEC1
DEFJS JFNS,30,MSEC1
DEFJS FFFFP,31,MSEC1
DEFJS RDDIR,32,MSEC1    ;OBSOLETE
DEFJS CPRTF,33,,NIM
DEFJS CLZFF,34,MSEC1
DEFJS RNAMF,35,MSEC1
DEFJS SIZEF,36,MSEC1
DEFJS GACTF,37,MSEC1
DEFJS STDIR,40,MSEC1    ;OBSOLETE
DEFJS DIRST,41,MSEC1
DEFJS BKJFN,42,MSEC1
DEFJS RFPTR,43,MSEC1
DEFJS CNDIR,44,,NIM
DEFJS RFBSZ,45,MSEC1
DEFJS SFBSZ,46,MSEC1
DEFJS SWJFN,47,MSEC1
DEFJS BIN,50,MSEC1
DEFJS BOUT,51,MSEC1
DEFJS SIN,52,MSEC1
DEFJS SOUT,53,MSEC1
DEFJS RIN,54,MSEC1
DEFJS ROUT,55,MSEC1
DEFJS PMAP,56,MSEC1
DEFJS RPACS,57,MSEC1
DEFJS SPACS,60,MSEC1
DEFJS RMAP,61,MSEC1
DEFJS SACTF,62,MSEC1
DEFJS GTFDB,63,MSEC1
DEFJS CHFDB,64,MSEC1
```

```
DEFJS DUMPI,65,MSEC1
DEFJS DUMPO,66,MSEC1
DEFJS DELDF,67,MSEC1
DEFJS ASND,70,MSEC1
DEFJS RELD,71,MSEC1
DEFJS CSYNO,72,,NIM
DEFJS PBIN,73,MSEC1
DEFJS PBOUT,74,MSEC1
DEFJS PSIN,75,,NIM
DEFJS PSOUT,76,MSEC1
DEFJS MTOPR,77,MSEC1
DEFJS CFIBF,100,MSEC1
DEFJS CFOBF,101,MSEC1
DEFJS SIBE,102,MSEC1
DEFJS SOBE,103,MSEC1
DEFJS DOBE,104,MSEC1
DEFJS GTABS,105,MSEC1   ;OBSOLETE
DEFJS STABS,106,MSEC1   ;OBSOLETE
DEFJS RFMOD,107,MSEC1
DEFJS SFMOD,110,MSEC1
DEFJS RFPOS,111,MSEC1
DEFJS RFCOC,112,MSEC1
DEFJS SFCOC,113,MSEC1
DEFJS STI,114,MSEC1
DEFJS DTACH,115,MSEC1
DEFJS ATACH,116,MSEC1
DEFJS DVCHR,117,MSEC1
DEFJS STDEV,120,MSEC1
DEFJS DEVST,121,MSEC1
DEFJS MOUNT,122,MSEC1   ;OBSOLETE
DEFJS DSMNT,123         ;OBSOLETE
DEFJS INIDR,124,MSEC1   ;OBSOLETE
DEFJS SIR,125,MSEC1
DEFJS EIR,126,MSEC1
DEFJS SKPIR,127,MSEC1
DEFJS DIR,130,MSEC1
DEFJS AIC,131,MSEC1
DEFJS IIC,132,MSEC1
DEFJS DIC,133,MSEC1
DEFJS RCM,134,MSEC1
DEFJS RWM,135,MSEC1
DEFJS DEBRK,136,MSEC1
DEFJS ATI,137,MSEC1
DEFJS DTI,140,MSEC1
DEFJS CIS,141,MSEC1
DEFJS SIRCM,142,MSEC1
DEFJS RIRCM,143,MSEC1
DEFJS RIR,144,MSEC1
DEFJS GDSTS,145,MSEC1
DEFJS SDSTS,146,MSEC1
DEFJS RESET,147,MSEC1
DEFJS RPCAP,150,MSEC1
DEFJS EPCAP,151,MSEC1
DEFJS CFORK,152,MSEC1
DEFJS KFORK,153,MSEC1
DEFJS FFORK,154,MSEC1
DEFJS RFORK,155,MSEC1
DEFJS RFSTS,156,MSEC1
DEFJS SFORK,157,MSEC1
DEFJS SFACS,160,MSEC1
```

```
DEFJS RFACS,161,MSEC1
DEFJS HFORK,162,MSEC1
DEFJS WFORK,163,MSEC1
DEFJS GFRKH,164,MSEC1
DEFJS RFRKH,165,MSEC1
DEFJS GFRKS,166,MSEC1
DEFJS DISMS,167,MSEC1
DEFJS HALTF,170,MSEC1
DEFJS GTRPW,171,MSEC1
DEFJS GTRPI,172,MSEC1
DEFJS RTIW,173,MSEC1
DEFJS STIW,174,MSEC1
DEFJS SOBF,175,MSEC1
DEFJS RWSET,176,MSEC1
DEFJS GETNM,177,MSEC1
DEFJS GET,200,MSEC1
DEFJS SFRKV,201,MSEC1
DEFJS SAVE,202,MSEC1
DEFJS SSAVE,203,MSEC1
DEFJS SEVEC,204,MSEC1
DEFJS GEVEC,205,MSEC1
DEFJS GPJFN,206,MSEC1
DEFJS SPJFN,207,MSEC1
DEFJS SETNM,210,MSEC1
DEFJS FFUFP,211,MSEC1
DEFJS DIBE,212,MSEC1
DEFJS FDFRE,213,,NIM
DEFJS GDSKC,214,MSEC1
DEFJS LITES,215,MSEC1    ;OBSOLETE
DEFJS TLINK,216,MSEC1
DEFJS STPAR,217,MSEC1
DEFJS ODTIM,220,MSEC1
DEFJS IDTIM,221,MSEC1
DEFJS ODCNV,222,MSEC1
DEFJS IDCNV,223,MSEC1
DEFJS NOUT,224,MSEC1
DEFJS NIN,225,MSEC1
DEFJS STAD,226,MSEC1
DEFJS GTAD,227,MSEC1
DEFJS ODTNC,230,MSEC1
DEFJS IDTNC,231,MSEC1
DEFJS FLIN,232,MSEC1
DEFJS FLOUT,233,MSEC1
DEFJS DFIN,234,MSEC1
DEFJS DFOUT,235,MSEC1

DEFJS CRDIR,240,MSEC1
DEFJS GTDIR,241,MSEC1
DEFJS DSKOP,242,MSEC1
DEFJS SPRIW,243,MSEC1
DEFJS DSKAS,244,MSEC1
DEFJS SJPRI,245,MSEC1
DEFJS STO,246,MSEC1

DEFJS ASNDP,260,,NIM
DEFJS RELDP,261,,NIM
DEFJS ASNDC,262,,NIM
DEFJS RELDC,263,,NIM
DEFJS STRDP,264,,NIM
```

```
DEFJS STPDP,265,,NIM
DEFJS STSDP,266,,NIM
DEFJS RDSDP,267,,NIM
DEFJS WATDP,270,,NIM

DEFJS ATNVT,274,MSEC1          ;TOPS20AN
DEFJS CVSKT,275,MSEC1          ;TOPS20AN
DEFJS CVHST,276,MSEC1          ;TOPS20AN
DEFJS FLHST,277,MSEC1          ;TOPS20AN

DEFJS GCVEC,300,MSEC1
DEFJS SCVEC,301,MSEC1
DEFJS STTYP,302,MSEC1
DEFJS GTTYP,303,MSEC1
DEFJS BPT,304,MSEC1      ;OBSOLETE
DEFJS GTDAL,305,MSEC1
DEFJS WAIT,306,MSEC1
DEFJS HSYS,307,MSEC1
DEFJS USRIO,310,MSEC1
DEFJS PEEK,311,MSEC1
DEFJS MSFRK,312,MSEC1
DEFJS ESOUT,313,MSEC1
DEFJS SPLFK,314,MSEC1
DEFJS ADVIS,315,,NIM
DEFJS JOBTM,316,,NIM
DEFJS DELNF,317,MSEC1
DEFJS SWTCH,320,MSEC1    ;OBSOLETE
DEFJS TFORK,321,MSEC1
DEFJS RTFRK,322,MSEC1
DEFJS UTFRK,323,MSEC1
DEFJS SCTTY,324,MSEC1

DEFJS SETER,336,MSEC1
```

```
;NEW (NOT IN BBN TENEX) JSYS'S ADDED STARTING AT 500

DEFJS RSCAN,500,MSEC1
DEFJS HPTIM,501,MSEC1
DEFJS CRLNM,502,MSEC1
DEFJS INLNM,503,MSEC1
DEFJS LNMST,504,MSEC1
DEFJS RDTXT,505,MSEC1    ;OBSOLETED BY RDTTY AND TEXTI
DEFJS SETSN,506,MSEC1
DEFJS GETJI,507,MSEC1
DEFJS MSEND,510,MSEC1
DEFJS MRECV,511,MSEC1
DEFJS MUTIL,512,MSEC1
DEFJS ENQ,513,MSEC1
DEFJS DEQ,514,MSEC1
DEFJS ENQC,515,MSEC1
DEFJS SNOOP,516,MSEC1
DEFJS SPOOL,517,MSEC1
DEFJS ALLOC,520,MSEC1
DEFJS CHKAC,521,MSEC1
DEFJS TIMER,522,MSEC1
DEFJS RDTTY,523,MSEC1
DEFJS TEXTI,524,MSEC1
DEFJS UFPGS,525,MSEC1
DEFJS SFPOS,526,MSEC1
DEFJS SYERR,527,MSEC1
DEFJS DIAG,530,MSEC1
DEFJS SINR,531,MSEC1
DEFJS SOUTR,532,MSEC1
DEFJS RFTAD,533,MSEC1
DEFJS SFTAD,534,MSEC1
DEFJS TBDEL,535,MSEC1
DEFJS TBADD,536,MSEC1
DEFJS TBLUK,537,MSEC1
DEFJS STCMP,540,MSEC1
DEFJS SETJB,541,MSEC1
DEFJS GDVEC,542,MSEC1
DEFJS SDVEC,543,MSEC1
DEFJS COMND,544,MSEC1
DEFJS PRARG,545,MSEC1
DEFJS GACCT,546,MSEC1
DEFJS LPINI,547,MSEC1
DEFJS GFUST,550,MSEC1
DEFJS SFUST,551,MSEC1
DEFJS ACCES,552,MSEC1
DEFJS RCDIR,553,MSEC1
DEFJS RCUSR,554,MSEC1
DEFJS MSTR,555,MSEC1
DEFJS STPPN,556,MSEC1
DEFJS PPNST,557,MSEC1
DEFJS PMCTL,560,MSEC1
DEFJS PLOCK,561,MSEC1
DEFJS BOOT,562,MSEC1
DEFJS UTEST,563
DEFJS USAGE,564,MSEC1

; HOLE - SLOT 565 AVAILABLE

DEFJS VACCT,566,MSEC1
DEFJS NODE,567,MSEC1
DEFJS ADBRK,570,MSEC1
DEFJS SINM,571,,NIM
DEFJS SOUTM,572,,NIM
```

```
DEFJS SWTRP,573,,NIM

;TEMPORARY JSYS DEFINITIONS

DEFJS SNDIM,750,MSEC1          ;TOPS20AN
DEFJS RCVIM,751,MSEC1          ;TOPS20AN
DEFJS ASNSQ,752,MSEC1          ;TOPS20AN
DEFJS RELSQ,753,MSEC1          ;TOPS20AN

DEFJS THIBR,770,MSEC1
DEFJS TWAKE,771,MSEC1
DEFJS MRPAC,772,MSEC1
DEFJS SETPV,773,,NIM
DEFJS MTALN,774,MSEC1
DEFJS TTMSG,775,MSEC1

>  ;;; END OF DEFINE JSLIST

;NOW EXPAND THE JSYS DEFINITIONS

JSLIST
```

```
;ERROR CONDITION INSTRUCTIONS.  THESE ARE NOP'S UNLESS  IMMEDIATELY
;FOLLOWING A JSYS WHICH FAILS.

OPDEF ERJMP [JUMP 16,0]          ;JUMP ON ERROR
OPDEF ERCAL [JUMP 17,0]          ;CALL ON ERROR (SIMULATE PUSHJ 17,ADR)
   IFNDEF FOR,<
   IFDEF .PSECT,<
INTERN ERJMP,ERCAL
   >>

DEFINE GOPDEF (OP,DEF)<
        OPDEF OP [DEF]
   IFNDEF FOR,<
   IFDEF .PSECT,<
        INTERN OP>>>

; THE FOLLOWING OPCODES ARE USED TO PERFORM THE EXTENDED
; ADDRESSING FUNCTIONS.

GOPDEF XJRSTF,<JRST 5,0>         ;RESTORE FLAGS AND PC
GOPDEF XJEN,<JRST 6,0>           ;RESTORE FLAGS,PC AND DISMISS
GOPDEF XPCW,<JRST 7,0>           ;EXCHANGE FLAGS AND PC
GOPDEF XSFM,<JRST 14,0>          ;SAVE PC FLAGS IN MEMORY
GOPDEF XMOVEI,<SETMI 0,0>        ;EXTENDED MOVEI
GOPDEF XHLLI,<XMOVEI 0,0>        ;INSTRUCTION TO PUT IMMEDIATE ADDRESS IN LH

DEFINE XBLT (A)<
        EXTEND A,[020000,,0]>

IFIW==:1B0                       ;INSTRUCTION FORMAT INDIRECT WORD
EFIW==:0                         ;EXTENDED FORMAT INDIRECT WORD

;THE NO-OPERATION INSTRUCTION (MAY CHANGE FROM PROCESSOR TO PROCESSOR)

GOPDEF NOP,<TRN 0,0>

;SPECIAL LOSEG SYMBOL FOR PAT

.JBHSO==:75                      ; 0 ,, HIGHSEG ORIGIN PAGE NUMBER
```

```
;****************************************
;JSYS SPECIFIC ARGUMENTS
;THE FOLLOWING ARE ORDERED ALPHABETICALLY BY JSYS NAME
;****************************************


;ACCES - ACCESS A DIRECTORY (E.G., BY CONNECTING)

AC%CON==:1B0                    ;CONNECT TO THE SPECIFIED DIRECTORY
AC%OWN==:1B1                    ;GAIN OWNERSHIP
AC%REM==:1B2                    ;REMOVE OWNERSHIP

;OFFSETS IN ARGUMENT BLOCK

.ACDIR==:0                      ;DIRECTORY DESIGNATOR
.ACPSW==:1                      ;POINTER TO PASSWORD STRING
.ACJOB==:2                      ;JOB NUMBER (-1 FOR SELF)


;ADBRK - Address break JSYS function codes and bits

;FUNCTION CODES

.ABSET==:0                      ;SET USER ADDRESS BREAK
.ABRED==:1                      ;READ USER ADDRESS BREAK
.ABCLR==:2                      ;CLEAR USER ADDRESS BREAK
.ABGAD==:3                      ;GET ADDRESS OF TRAPPED INSTRUCTION

;FUNCTION BITS FOR FUNCTION .ABSET

AB%RED==:1B0                    ;READ
AB%WRT==:1B1                    ;WRITE
AB%XCT==:1B2                    ;EXECUTE


;ALLOC JSYS FUNCTION CODES

.ALCAL==:0                      ;ALLOCATE A DEVICE


;ATNVT                  ;TOPS20AN

AN%NTP==:1B2            ;TOPS20AN ;NEW TELNET PROTOCOL


;ATACH

AT%CCJ==:1B0                    ;^C JOB WHEN ATTACHED
AT%NAT==:1B1                    ;NO ATTACH
AT%TRM==:1B2                    ;ATTACH JOB TO TERMINAL IN REGISTER 4
AT%JOB==:777777B35              ;JOB NUMBER
```

```
;BOOT

.BTROM==:0                      ;ACTIVATE ROM BOOT
        .BTDTE==:0              ;DTE-20 NUMBER
.BTLDS==:1                      ;LOAD SECONDARY BOOTSTRAP PROGRAM
        .BTERR==:1              ;ERROR FLAGS
        .BTSEC==:2              ;ADDRESS OF SECONDARY BOOTSTRAP PROGRAM
.BTLOD==:2                      ;LOAD MEMORY (OBSOLETE)
.BTSMP==:2                      ;SEND MOP MESSAGE
        .BTFLG==:3              ;FLAGS
                BT%BEL==:1B0    ;SEND TO -11 DOORBELL AFTER SETUP
        .BTCNT==:4              ;NUMBER OF BYTES TO BE TRANSFERRED
        .BTLPT==:5              ;BYTE POINTER TO DATA TO BE LOADED
.BTDMP==:3                      ;DUMP MEMORY
        .BTDPT==:5              ;BYTE POINTER TO DESTINATION OF DUMPED DATA
.BTIPR==:4                      ;INITIALIZE COMMUNICATIONS PROTOCOL
        .BTPRV==:1              ;PROTOCOL VERSION NUMBER
.BTTPR==:5                      ;TERMINATE COMMUNICATIONS PROTOCOL
.BTSTS==:6                      ;RETURN PROTOCOL STATUS
        .BTCOD==:1              ;STATUS CODE
.BTBEL==:7                      ;WAIT FOR DOORBELL
.BTRMP==:10                     ;READ MOP MESSAGE
        .BTMPT==:5              ;POINTER TO DESTINATION FOR MOP MESSAGE
.BTKML==:11                     ;LOAD KMC11
        .BTKMC==:0              ;KMC11 ADDRESS
        .BTKER==:1              ;ERROR FLAGS
                BT%CVE==:1B0    ;CRAM VERIFY ERROR (RH IS BAD DATA)
                BT%DVE==:1B1    ;DRAM VERIFY ERROR (RH IS BAD DATA)
                BT%RVE==:1B2    ;REG VERIFY ERROR (RH IS BAD DATA)
        .BTKCC==:2              ;COUNT OF CRAM DATA
        .BTKCP==:3              ;POINTER TO CRAM DATA (16 BIT DATA)
        .BTKDC==:4              ;COUNT OF DRAM DATA
        .BTKDP==:5              ;POINTER TO DRAM DATA (8 BIT DATA)
        .BTKRC==:6              ;COUNT OF REGISTER DATA
        .BTKRP==:7              ;POINTER TO REGISTER DATA (16 BIT DATA)
        .BTKSA==:10             ;RH IS STARTING ADDRESS
                BT%KSA==:1B0    ;IS SET RH WANT TO START KMC11
.BTKMD==:12                     ;DUMP KMC11
.BTRLC==:13                     ;RETURN LINE COUNTERS
        .BTPRT==:0              ;PORT NUMBER
                BT%ZRO==:1B0    ;CLEAR COUNTERS AFTER READING
        .BTZTM==:1              ;TIME SINCE COUNTERS HAVE BEEN ZEROED
        .BTSCC==:2              ;STATUS COUNT COUNT
        .BTSCP==:3              ;STATUS COUNT POINTER
        .BTRCC==:4              ;RECEIVE COUNT COUNT
        .BTRCP==:5              ;RECEIVE COUNT POINTER
        .BTTCC==:6              ;TRANSMIT COUNT COUNT
        .BTTCP==:7              ;TRANSMIT COUNT POINTER
.BTCLI==:14                     ;CONVERT LINEID TO PORT NUMBER
        .BTLID==:1              ;POINTER TO ASCIZ LINE-ID
.BTCPN==:15                     ;CONVERT PORT NUMBER TO LINE-ID
```

```
;CFORK

CR%MAP==:1B0                    ;SET MAP FOR NEW FORK TO POINT TO
                                ; THIS PROCESS
CR%CAP==:1B1                    ;MAKE CAPABILITIES IDENTICAL
CR%ACS==:1B3                    ;SET ACS FROM BLOCK
CR%ST==:1B4                     ;START PROCESS AT PC
CR%PCV==:777777B35              ;VALUE OF PC


;CHFDB

CF%NUD==:1B0                    ;NO UPDATE DIRECTORY
CF%DSP==:777B17                 ;FDB DISPLACEMENT
CF%JFN==:777777B35              ;JFN
```

```
;CHKAC JSYS DEFINITIONS

;CHKAC FLAG DEFINITIONS

CK%JFN==:1B0                        ;JFN IS GIVEN AS AN ARGUMENT

;CHKAC ARGUMENT BLOCK OFFSET VALUES

.CKAAC==:0                          ;ACCESS CODE
.CKALD==:1                          ;LOGGED IN USER NUMBER OF USER
.CKACD==:2                          ;CONNECTED DIR NUMBER OF USER
.CKAEC==:3                          ;ENABLED CAPABILITIES OF USER BEING CHK'D
.CKAUD==:4                          ;DIR NUMBER OF DIRECTORY CONTAINING FILE
.CKAPR==:5                          ;PROTECTION OF FILE

;CHKAC ACCESS CODES

.CKARD==:0                          ;READ AN EXISTING FILE
.CKAWT==:1                          ;WRITE AN EXISTING FILE
.CKAWR==:1                          ; (ANOTHER NAME FOR ABOVE)
.CKAEX==:2                          ;EXECUTE AN EXISTING FILE
.CKAAP==:3                          ;APPEND TO AN EXISTING FILE
.CKADL==:4                          ;GET DIR LISTING OF AN EXISTING FILE
.CKADR==:6                          ;READ THE DIRECTORY
.CKAOF==:7                          ;OPEN FILES IN DIR (NOT IMPLEMENTED)
.CKACN==:10                         ;CONNECT TO A DIR
.CKACF==:11                         ;CREATE FILES IN DIR


;CLOSF

CO%NRJ==:1B0                        ;NO RELEASE JFN
CO%WCL==:1B1          ;TOPS20AN     ;WAIT UNTIL MATCHING CLS IS RECEIVED
CO%JFN==:777777B35                  ;JFN


;CLZFF

CZ%NIF==:1B0                        ;NO INFERIOR FORK FILES
CZ%NSF==:1B1                        ;NO SELF FORK FILES
CZ%NRJ==:1B2                        ;NO RELEASE JFN
CZ%NCL==:1B3                        ;NO CLOSE FILE
CZ%UNR==:1B4                        ;UNRESTRICT
CZ%ARJ==:1B5                        ;ALWAYS RELEASE JFN
CZ%ABT==:1B6                        ;ABORT
CZ%NUD==:1B7                        ;NO UPDATE DIRECTORY
CZ%PRH==:777777B35                  ;PROCESS HANDLE
```

```
;CNDIR

CN%CKP==:1B0                    ;CHECK PASSWORD ONLY
CN%NOC==:1B1                    ;NO CONNECT
CN%JOB==:1B2                    ;DOING CONNECT FOR ANOTHER JOB
CN%DIR==:777777B35              ;DIRECTORY NUMBER


;COMND

;COMND - COMMAND STATE BLOCK

.CMFLG==:0                      ;USER FLAGS,,REPARSE DISPATCH ADDRESS
.CMIOJ==:1                      ;INJFN,,OUTJFN
.CMRTY==:2                      ;^R BUFFER POINTER
.CMBFP==:3                      ;PTR TO TOP OF BUFFER
.CMPTR==:4                      ;PTR TO NEXT INPUT TO BE PARSED
.CMCNT==:5                      ;COUNT OF SPACE LEFT IN BUFFER AFTER PTR
.CMINC==:6                      ;COUNT OF CHARACTERS FOLLOWING PTR
.CMABP==:7                      ;ATOM BUFFER POINTER
.CMABC==:10                     ;ATOM BUFFER SIZE
.CMGJB==:11                     ;ADR OF GTJFN ARG BLOCK
 CM%GJB==:777777                ;ADR OF GTJFN ARG BLOCK

;COMND - FUNCTION DESCRIPTOR BLOCK

.CMFNP==:0                      ;FUNCTION AND POINTER
  CM%FNC==:777B8                ;FUNCTION CODE
  CM%FFL==:777B17               ;FUNCTION-SPECIFIC FLAGS
  CM%LST==:777777               ;LIST POINTER TO OTHER BLOCKS
.CMDAT==:1                      ;DATA FOR FUNCTION
.CMHLP==:2                      ;HELP TEXT POINTER
.CMDEF==:3                      ;DEFAULT STRING POINTER


;COMND - MACRO FOR BUILDING FUNCTION DESCRIPTOR BLOCK

DEFINE FLDDB. (TYP,FLGS,DATA,HLPM,DEFM,LST)<
        ..XX==<FLD(TYP,CM%FNC)>+FLGS+<Z LST>
   IFNB <HLPM>,<..XX=CM%HPP!..XX>
   IFNB <DEFM>,<..XX=CM%DPP!..XX>
        ..XX
   IFNB <DATA>,<DATA>
   IFB <DATA>,<0>
   IFNB <HLPM>,<POINT 7,[ASCIZ \HLPM\]>
   IFB <HLPM>,<IFNB <DEFM>,<0>>
   IFNB <DEFM>,<POINT 7,[ASCIZ \DEFM\]>>
```

```
;COMND - FLAGS IN .CMFLG

CM%ESC==:1B0                    ;ESC SEEN
CM%NOP==:1B1                    ;NO PARSE
CM%EOC==:1B2                    ;END OF COMMAND SEEN
CM%RPT==:1B3                    ;REPEAT PARSE NEEDED
CM%SWT==:1B4                    ;SWITCH TERMINATED WITH ":"
CM%PFE==:1B5                    ;PREVIOUS FIELD ENDED WITH ESC
CM%RAI==:1B6                    ;RAISE INPUT
CM%XIF==:1B7                    ;EXCLUDE INDIRECT FILES
CM%WKF==:1B8                    ;WAKEUP AFTER EACH FIELD

;FUNCTION BLOCK FLAGS (IN WORD .CMFNP)

CM%PO==:1B14                    ;PARSE-ONLY
CM%HPP==:1B15                   ;HELP POINTER PRESENT
CM%DPP==:1B16                   ;DEFAULT POINTER PRESENT
CM%SDH==:1B17                   ;SUPPRESS DEFAULT HELP MESSAGE

;FLAGS FOR CMDIR FUNCTION

CM%DWC==:1B0                    ;DIRECTORY WILD CARDING ALLOWED

;FLAGS FOR CMTAD FUNCTION

CM%IDA==:1B0                    ;INPUT DATE
CM%ITM==:1B1                    ;INPUT TIME
CM%NCI==:1B2                    ;NO CONVERT TO INTERNAL

;FLAGS IN KEYWORD TABLE (FIRST WORD OF STRING IF B0-6 = 0)

CM%INV==:1B35                   ;INVISIBLE
CM%NOR==:1B34                   ;NO-RECOGNIZE (PLACEHOLDER)
CM%ABR==:1B33                   ;ABBREVIATION FOR ANOTHER ENTRY
CM%FW==:1B7                     ;FLAG WORD (MUST ALWAYS BE ON)
```

```
;COMND - FUNCTION CODES

.CMKEY==:0                      ;KEYWORD
.CMNUM==:1                      ;NUMBER
.CMNOI==:2                      ;NOISE WORD
.CMSWI==:3                      ;SWITCH
.CMIFI==:4                      ;INPUT FILE
.CMOFI==:5                      ;OUTPUT FILE
.CMFIL==:6                      ;GENERAL FILESPEC
.CMFLD==:7                      ;ARBITRARY FIELD
.CMCFM==:10                     ;CONFIRM
.CMDIR==:11                     ;DIRECTORY NAME
.CMUSR==:12                     ;USER NAME
.CMCMA==:13                     ;COMMA
.CMINI==:14                     ;INIT LINE
.CMFLT==:15                     ;FLOATING POINT NUMBER
.CMDEV==:16                     ;DEVICE NAME
.CMTXT==:17                     ;TEXT TO ACTION CHAR
.CMTAD==:20                     ;TIME AND DATE
.CMQST==:21                     ;QUOTED STRING
.CMUQS==:22                     ;UNQUOTED STRING
.CMTOK==:23                     ;TOKEN
.CMNUX==:24                     ;NUMBER DELIMITED BY NON-DIGIT
.CMACT==:25                     ;ACCOUNT
.CMNOD==:26                     ;NODE NAME
```

```
;CRDIR

CD%LEN==:1B0                    ;FLAGS ,, LENGTH OF CRDIR BLOCK
CD%PSW==:1B1                    ;SET PASSWORD STRING
CD%LIQ==:1B2                    ;SET LOGGED IN QUOTA
CD%PRV==:1B3                    ;SET PRIVILEGES
CD%MOD==:1B4                    ;SET MODE BITS
CD%LOQ==:1B5                    ;SET LOGGED OUT QUOTA
CD%NUM==:1B6                    ;SET DIRECTORY NUMBER FROM PARAM BLK
CD%FPT==:1B7                    ;SET DEFAULT FILE PROTECTION
CD%DPT==:1B8                    ;SET DIRECTORY PROTECTION
CD%RET==:1B9                    ;SET DEFAULT RETENTION COUNT
CD%LLD==:1B10                   ;SET LAST LOGIN DATE
CD%UGP==:1B11                   ;SET USER GROUPS
CD%DGP==:1B12                   ;SET DIRECTORY GROUPS
CD%SDQ==:1B13                   ;SET SUBDIRECTORY QUOTA
CD%CUG==:1B14                   ;SET CREATABLE USER GROUPS
CD%DAC==:1B15                   ;SET DEFAULT ACCOUNT
CD%DEL==:1B17                   ;DELETE DIRECTORY
CD%APB==:777777B35              ;ADDRESS OF PARAMETER BLOCK
.CDLEN==:0                      ;LENGTH OF ARGUMENT BLOCK
        CD%NSQ==:1B0            ;DO NOT UPDATE QUOTAS OF SUPERIOR DIR
        CD%NCE==:1B1            ;DO NOT CHANGE PARAMETERS OF EXISTING DIRS
.CDPSW==:1                      ;POINTER TO PASSWORD STRING
.CDLIQ==:2                      ;LOGGED IN QUOTA
.CDPRV==:3                      ;PRIVILEGE WORD
.CDMOD==:4                      ;MODE WORD
        CD%DIR==:1B0            ;DIRECTORY NAME FOR CNDIR ONLY (FILES ONLY)
        CD%ANA==:1B1            ;ALPHANUMERIC ACCOUNTS
        CD%RLM==:1B2            ;REPEAT LOGIN MESSAGES
.CDLOQ==:5                      ;LOGGED OUT QUOTA
.CDNUM==:6                      ;DIRECTORY NUMBER
.CDFPT==:7                      ;DEFAULT FILE PROTECTION
.CDDPT==:10                     ;DIRECTORY PROTECTION
.CDRET==:11                     ;DEFAULT RETENTION COUNT
.CDLLD==:12                     ;LAST LOGIN DATE
.CDUGP==:13                     ;USER GROUPS
.CDDGP==:14                     ;DIRECTORY GROUPS
.CDSDQ==:15                     ;MAXIMUM NUMBER OF SUBDIRECTORIES
.CDCUG==:16                     ;POINTER TO CREATABLE USER GROUP LIST
.CDDAC==:17                     ;POINTER TO DEFAULT ACCOUNT
```

```
;CRJOB

CJ%LOG==:1B0                    ;ATTEMPT TO LOG IN THE NEW JOB
CJ%NAM==:1B1                    ;USE NAME AND PSWD IN ARG BLK
CJ%ACT==:3B3                    ;WHERE TO GET ACCOUNT
        .CJUCA==:0              ;USE CURRENT ACCT OF CREATOR
        .CJUAA==:1              ;USE ACCOUNT IN ARG BLOCK
        .CJUDA==:2              ;USE DEFAULT ACCOUNT OF NEW USER
CJ%ETF==:1B4                    ;PUT EXEC IN TOP FORK
CJ%FIL==:1B5                    ;GET FILE IN ARG BLOCK
CJ%ACS==:1B6                    ;LOAD THE ACS FROM ARG BLOCK
CJ%OWN==:1B7                    ;RETAIN OWNERSHIP OF NEW JOB
CJ%WTA==:1B8                    ;NEW JOB WAITS TIL ATTACHED
CJ%NPW==:1B9                    ;NO PASSWORD CHECK AT LOGIN TIME
CJ%NUD==:1B10                   ;NO UPDATE OF LAST-LOGIN DATE
CJ%SPJ==:1B11                   ;DO SPJFN IN NEW JOB FROM ARG BLK
CJ%CAP==:1B12                   ;PASS ENABLED CAPABILITIES AS ALLOWED
CJ%CAM==:1B13                   ;CAPABILITY MASK AT LOGIN
CJ%SLO==:1B14                   ;SIGNAL (IPCF) AT LOGOUT TIME
CJ%DSN==:1B17                   ;DISOWN EXISTING JOB # IN 3

.CJNAM==:0                      ;NAME STRING POINTER
.CJPSW==:1                      ;PASSWORD STRING POINTER
.CJACT==:2                      ;ACCOUNT DESIGNATOR/STRING
.CJFIL==:3                      ;FILE NAME STRING POINTER
.CJSFV==:4                      ;SFRKV OFFSET
.CJTTY==:5                      ;TTY DESIGNATOR, OR NULL DESIGNATOR
.CJTIM==:6                      ;TIME LIMIT
.CJACS==:7                      ;ADDRESS OF 16. WORDS OF AC'S
.CJEXF==:10                     ;EXEC FLAGS, FOR EXEC AC1
.CJPRI==:11                     ;PRIMARY JFN'S FOR SPJFN IN NEW JOB
.CJCPU==:12                     ;CPU LIMIT (0 IF NONE)
.CJCAM==:13                     ;CAPABILITY MASK TO APPLY TO LOGIN
.CJSLO==:14                     ;PID TO SIGNAL AT LOGOUT TIME

CR%PRA==:2545                   ;MAGIC # FOR EXEC/CRJOB LINKAGE VIA PRARG


;CRLNM

.CLNJ1==:0                      ;DELETE 1 LOGICAL NAME FROM JOB
.CLNS1==:1                      ;DELETE 1 LOGICAL NAME FROM SYSTEM
.CLNJA==:2                      ;DELETE ALL JOB WIDE LOGICAL NAMES
.CLNSA==:3                      ;DELETE ALL SYSTEM LOGICAL NAMES
.CLNJB==:4                      ;CREATE A JOB WIDE LOGICAL NAME
.CLNSY==:5                      ;CREATE A SYSTEM WIDE LOGICAL NAME
```

```
;DELDF

DD%DTF==:1B0                    ;DELETE TEMPORARY FILES
DD%DNF==:1B1                    ;DELETE NONEXISTENT FILES
DD%RST==:1B2                    ;REBUILD THE SYMBOL TABLE
DD%CHK==:1B3                    ;CHECK THE DIR FOR CONSISTENCY ONLY


;DELF

DF%NRJ==:1B0                    ;DON'T RELEASE JFN
DF%EXP==:1B1                    ;EXPUNGE CONTENTS
DF%FGT==:1B2                    ;FORGET (EXPUNGE W/O DEASSIGNING ADDRESSES)
DF%DIR==:1B3                    ;DELETE, FORGET, AND EXPUNGE A DIRECTORY
                                ; FILE.  (ONLY IF ^E-CREATE KILL FAILED)
DF%JFN==777777B35               ;JFN


;DIAG JSYS DEFINITIONS

DG%ADT==:7B2                    ;ADDRESS TYPE FIELD
DG%DVC==:177B9                  ;DEVICE CODE FIELD
        .DGRH0==:130            ;MBC0
        .DGRH7==:137            ;MBC7
DG%UNI==:77B29                  ;UNIT NUMBER
DG%SUN==:77B35                  ;SUBUNIT NUMBER

;DIAG JSYS FUNCTION CODES

.DGACU==:1                      ;ASSIGN DEVICE
.DGACH==:2                      ;ASSIGN CONTROLLER AND ALL DEVICES
.DGRCH==:3                      ;RELEASE DEVICE(S)
.DGSCP==:4                      ;SETUP CHANNEL PROGRAM
.DGRCP==:5                      ;RELEASE CHANNEL PROGRAM
.DGGCS==:6                      ;GET CHANNEL STATUS

;DIAG MEM CONTROL FUNCTIONS

.DGGEM==:100                    ;LEAVE LARGE HOLE FOR MORE RH20 FUNCTIONS
                                ;GET MEM (FOR TGHA)
.DGREM==:101                    ;RELEASE MEM (FOR TGHA)


;DSKAS

DA%DEA==:1B0                    ;DEASSIGN DISK ADDRESS
DA%ASF==:1B1                    ;ASSIGN FREE PAGE
DA%CNV==:1B2                    ;CONVERT SOFTWARE TO HARDWARE ADDRESS
DA%HWA==:1B3                    ;HARDWARE ADDRESS GIVEN
DA%INI==:1B4                    ;INITIALIZE THE BIT TABLE
DA%WRT==:1B5                    ;WRITE THE BIT TABLE FILE
DA%ADR==:777777B35              ;DISK ADDRESS
```

```
;DVCHR AND DVCH1 BIT DEFINITIONS

DV%OUT==:1B0                    ;DEVICE CAN DO OUTPUT
DV%IN==:1B1                     ;DEVICE CAN DO INPUT
DV%DIR==:1B2                    ;DEVICE HAS A DIRECTORY
DV%AS==:1B3                     ;DEVICE IS ASSIGNABLE
DV%MDD==:1B4                    ;DEVICE IS A MULTIPLE DIRECTORY DEVICE
DV%AV==:1B5                     ;DEVICE IS AVAILABLE TO THIS JOB
DV%ASN==:1B6                    ;DEVICE IS ASSIGNED BY ASND
DV%MDV==:1B7                    ;RESERVED (HISTORICAL)
DV%MNT==:1B8                    ;DEVICE IS MOUNTED
DV%TYP==:777B17                 ;DEVICE TYPE FIELD
DV%MOD==:177777B35              ;DEVICE DATA MODE
DV%M0==:1B35                    ;DEVICE CAN BE OPENED IN MODE 0
DV%M1==:1B34                    ;DEVICE CAN BE OPENED IN MODE 1
DV%M2==:1B33                    ;DEVICE CAN BE OPENED IN MODE 2
DV%M3==:1B32                    ;DEVICE CAN BE OPENED IN MODE 3
DV%M4==:1B31                    ;DEVICE CAN BE OPENED IN MODE 4
DV%M5==:1B30                    ;DEVICE CAN BE OPENED IN MODE 5
DV%M6==:1B29                    ;DEVICE CAN BE OPENED IN MODE 6
DV%M7==:1B28                    ;DEVICE CAN BE OPENED IN MODE 7
DV%M10==:1B27                   ;DEVICE CAN BE OPENED IN MODE 10
DV%M11==:1B26                   ;DEVICE CAN BE OPENED IN MODE 11
DV%M12==:1B25                   ;DEVICE CAN BE OPENED IN MODE 12
DV%M13==:1B24                   ;DEVICE CAN BE OPENED IN MODE 13
DV%M14==:1B23                   ;DEVICE CAN BE OPENED IN MODE 14
DV%M15==:1B22                   ;DEVICE CAN BE OPENED IN MODE 15
DV%M16==:1B21                   ;DEVICE CAN BE OPENED IN MODE 16
DV%M17==:1B20                   ;DEVICE CAN BE OPENED IN MODE 17
D1%SPL==:1B0                    ;DEVICE IS SPOOLED
D1%ALC==:1B1                    ;DEVICE IS UNDER CONTROL OF ALLOCATOR
D1%VVL==:1B2                    ;VOLUME VALID
D1%NIU==:1B3                    ;DEVICE SLOT IS NOT IN USE (FOR STRUCTURES
                                ; NOT YET MOUNTED)
D1%INI==:1B4                    ;DEVICE IS BEING INITIALIZED (STRUCTURE
                                ; IS AVAILABLE ONLY TO THE FORK WHOSE NUMBER
                                ; IS STORED IN SDBSTS)
```

```
;DEVICE TYPE DEFINITIONS

.DVDSK==:0                      ;DISK
.DVMTA==:2                      ;MAGTAPE
.DVDTA==:3                      ;DECTAPE
.DVPTR==:4                      ;PAPER TAPE READER
.DVPTP==:5                      ;PAPER TAPE PUNCH
.DVDSP==:6                      ;DISPLAY
.DVLPT==:7                      ;LINE PRINTER
.DVCDR==:10                     ;CARD READER
.DVFE==:11                      ;FRONT END DEVICE
.DVTTY==:12                     ;TERMINAL
.DVPTY==:13                     ;PTY
.DVNUL==:15                     ;NULL DEVICE
.DVNET==:16                     ;ARPA NETWORK
.DVPLT==:17                     ;PLOTTER
.DVDCN==:22                     ;DECNET ACTIVE COMPONENT
.DVSRV==:23                     ;DECENT PASSIVE COMPONENT
.DVATS==:24                     ;APPLICATIONS TERMINAL SERVICE


;DSKOP

DOP%SA==:1B0                    ;SOFTWARE ADDRESS
DOP%AT==:3B1                    ;ADDRESS TYPE FIELD
 .DOPPU==:1                     ;PHYSICAL CHANNEL AND UNIT
  DOP%CN==:37B6                 ;CHANNEL NUMBER
  DOP%UN==:77B12                ;UNIT NUMBER
  DOP%UA==:37777777             ;UNIT ADDRESS
 .DOPSR==:2                     ;STRUCTURE AND RELATIVE ADDRESS
  DOP%SN==:777B10               ;STRUCTURE NUMBER
  DOP%RA==:177777777            ;RELATIVE ADDRESS
DOP%EO==:1B10                   ;ERROR IF UNIT OFFLINE
DOP%IL==:1B11                   ;INHIBIT ERROR LOGGING
DOP%IR==:1B12                   ;INHIBIT ERROR RECOVERY
DOP%WR==:1B14                   ;WRITE
DOP%CT==:777777B35              ;WORD COUNT


;DUMPI/DUMPO

DM%NWT==:1B0                    ;NO WAIT FOR COMPLETION
DM%FIN==:1B1                    ;FINISH PREVIOUS REQUEST
                                ;***NOT INPLEMENTED YET***
DM%PTR==:777777B35              ;POINTER TO COMMAND LIST
```

```
;DEFINE DECNET DISCONNECT CODES. THESE ARE STIPULATED BY THE NSP SPEC
;AND MAY HAVE MEANININGS NOT IMPLIED BY THE COMMENTS

.DCX0==:0                       ;NO SPECIAL ERROR
.DCX1==:1                       ;RESOURCE ALLOCATION FAILURE
.DCX2==:2                       ;DESTINATION NODE DOES NOT EXIST
.DCX3==:3                       ;NODE SHUTTING DOWN
.DCX4==:4                       ;DESTINATION PROCESS DOES NOT EXIST
.DCX5==:5                       ;INVALID NAME FIELD
.DCX9==:^D9                     ;USER ABORT (ASYNCHRONOUS DISCONNECT)
.DCX11==:^D11                   ;UNDEFINED ERROR CODE
.DCX21==:^D21                   ;CI WITH ILLEGAL DESTINATION ADDRESS
.DCX22==:^D22                   ;CC WITH ILLEGAL DESTINATION ADDRESS
.DCX23==:^D23                   ;CI OR CC WITH ZERO SOURCE ADDRESS
.DCX32==:^D32                   ;TOO MANY CONNECTIONS TO NDOE
.DCX33==:^D33                   ;TOO MANY CONNECTIONS TO DEST. PROCESS
.DCX34==:^D34                   ;ACCESS NOT PERMITTED
.DCX35==:^D35                   ;LOGICAL LINK SERVICES MISMATCH
.DCX36==:^D36                   ;INVALID ACCOUNT
.DCX37==:^D37                   ;SEGSIZE TOO SMALL
.DCX38==:^D38                   ;PROCESS ABORTED
.DCX39==:^D39                   ;NO PATH TO DESTINATION NODE
.DCX40==:^D40                   ;LINK ABORTED DUE TO DATA LOSS
.DCX41==:^D41                   ;DESTINATION PROCESS DOES NOT EXIST
.DCX42==:^D42                   ;CONFIRMATION IF DI
.DCX43==:^D43                   ;IMAGE DATA FIELD TOO LONG


;EFACT - FACT FILE ENTRY DEFINITIONS

.EFHDR==:0                      ;HEADER WORD
        EF%COD==:777B8          ;ENTRY TYPE CODE
        EF%JOB==:777B17         ;JOB NUMBER
        EF%LIN==:7777B29        ;LINE NUMBER
        EF%SIZ==:77B35          ;ENTRY SIZE
.EFUSR==:1                      ;USER NUMBER WORD
.EFTAD==:2                      ;TIME AND DATE OF ENTRY

; FACT FILE ENTRY TYPE CODES

.EFLGI==:501                    ;LOGIN
.EFLGO==:141                    ;LOGOUT
.EFCAC==:502                    ;CHANGE ACCOUNT
.EFATT==:142                    ;CONSOLE ATTACH
.EFDET==:143                    ;CONSOLE DETACH
.EFCHK==:201                    ;CHECKPOINT
.EFSDU==:540                    ;START DISK-UTILIZATION ENTRIES
.EFDSK==:601                    ;DISK SPACE UTILIZATION
.EFTIM==:741                    ;TIME SET
.EFRES==:740                    ;SYSTEM RESTARTED
.EFLPT==:401                    ;LINE PRINTER USAGE
.EFCDR==:402                    ;CARD READER USAGE
```

```
;ENQ/DEQ BIT DEFINITIONS AND FUNCTION CODES

;FUNCTION CODES

.ENQBL==:0                      ;ENQ BLOCK OPTION
.ENQAA==:1                      ;ENQ ALLOCATE ONLY IF AVAILABLE
.ENQSI==:2                      ;ENQ SOFTWARE INTERRUPT WHEN LOCKED
.ENQMA==:3                      ;ENQ MODIFY ACCESS
.DEQDR==:0                      ;DEQ RESOURCE
.DEQDA==:1                      ;DEQ ALL RESOURCES OF THIS FORK
.DEQID==:2                      ;DEQ THIS ID NUMBER
.ENQCS==:0                      ;ENQC STATUS
.ENQCG==:1                      ;ENQC GET ENQ/DEQ QUOTA FOR A JOB
.ENQCC==:2                      ;ENQC CHANGE ENQ/DEQ QUOTA FOR A JOB
.ENQCD==:3                      ;ENQC DUMP LOCKS AND QUEUE ENTRIES


;BIT DEFINITIONS

EN%SHR==:1B0                    ;SHARABLE REQUEST
EN%BLN==:1B1                    ;BYPASS LEVEL NUMBER
EN%NST==:1B2                    ;ALLOW NESTING
EN%LTL==:1B3                    ;LONG TERM LOCK
EN%LVL==:777B17                 ;LEVEL NUMBER
EN%JOB==:777777B35              ;JOB NUMBER
EN%QCE==:1B0                    ;ERROR CODE IN RH OF STATUS WORD
EN%QCL==:1B0                    ;LOCK DUMP (.ENQCD ONLY)
EN%QCO==:1B1                    ;THIS FORK OWNS THE LOCK
EN%QCQ==:1B2                    ;THIS FORK IS IN THE QUEUE FOR THIS LOCK
EN%QCT==:1B2                    ;LOCK CONTAINS A TEXT STRING
EN%QCX==:1B3                    ;THE LOCK IS LOCKED EXCLUSIVELY
EN%QCB==:1B4                    ;USER IS BLOCKED FOR LOCK

;ENQ/DEQ ARGUMENT BLOCK DATA STRUCTURE

.ENQLN==:0                      ;# OF LOCKS ,, LENGTH OF ARGUMENT BLOCK
        .ENHLN==:77B5           ;LENGTH OF HEADER AREA
        .ENNLK==:7777B17        ;NUMBER OF LOCKS
        .ENALN==:777777B35      ;LENGTH OF ARGUMENT BLOCK
.ENQID==:1                      ;PSI CHANNEL # ,, REQUEST ID
.ENQLV==:2                      ;FLAGS & LEVEL NUMBER ,, JFN, -1, -2, OR -3
.ENQUC==:3                      ;STRING POINTER OR USER CODE
.ENQRS==:4                      ;# OF RESOURCES IN POOL ,, # OF RESOURCES WANTED
.ENQMS==:5                      ;ADDRESS OF RESOURCE BLOCK
```

```
;ENQC DUMP DATA STRUCTURE

.ENQDF==:0                      ;FLAGS + LEVEL # ,, OFN, 400000+JOB #, -2, OR -3
                                ;OR:  FLAGS + PSI # ,, JOB # OF Q-ENTRY CREATOR

.ENQDR==:1                      ;TOTAL RESOURCES IN POOL ,, RESOURCES REMAINING
.ENQDT==:2                      ;TIME STAMP OF LAST REQUEST LOCKED
.ENQDC==:3                      ;USER CODE OF LOCK OR START OF TEXT STRING

.ENQDI==:1                      ;GROUP # OR # REQUESTED ,, ENQ ID


;FLOUT/DFOUT
;FORMAT CONTROL WORD

FL%SGN==:3B1                    ;FIRST FIELD SIGN CONTROL
 .FLDIG==:0                     ;DIGIT
 .FLSPC==:1                     ;SPACE
 .FLPLS==:2                     ;PLUS SIGN
 .FLSPA==:3                     ;SPACE
FL%JUS==:3B3                    ;FIRST FIELD JUSTIFICATION CONTROL
 .FLLSP==:0                     ;LEADING SPACES
 .FLLZR==:1                     ;LEADING ZEROS
 .FLLAS==:2                     ;LEADING ASTERISKS
 .FLTSP==:3                     ;TRAILING SPACES
FL%ONE==:1B4                    ;FIRST FIELD NONBLANK
FL%DOL==:1B5                    ;DOLLAR SIGN PREFIX
FL%PNT==:1B6                    ;DECIMAL POINT
FL%EXP==:3B8                    ;THIRD FIELD EXPONENT CONTROL
 .FLEXN==:0                     ;NO EXPONENT
 .FLEXE==:1                     ;E EXPONENT PREFIX
 .FLEXD==:2                     ;D EXPONENT PREFIX
 .FLEXM==:3                     ;*10^ EXPONENT PREFIX
FL%ESG==:3B10                   ;EXPONENT SIGN CONTROL
 .FLDGE==:0                     ;DIGIT
 .FLPLE==:1                     ;PLUS SIGN
 .FLSPE==:2                     ;SPACE
 .FLDGT==:3                     ;DIGIT
FL%OVL==:1B11                   ;COLUMN OVERFLOW
FL%RND==:37B17                  ;DIGIT POSITION FOR ROUNDING
FL%FST==:77B23                  ;FIRST FIELD WIDTH
FL%SND==:77B29                  ;SECOND FIELD WIDTH
FL%THD==:77B35                  ;THIRD FIELD WIDTH
```

```
;GDSTS

;SEE MTOPR FOR CARD READER AND LINE PRINTER STATUS BITS
;SEE GENERAL FIELD AND VALUE SECTION FOR MAGTAPE STATUS BITS
;SEE TOPS20AN SECTION FOR NETWORK STATUS BITS

.GDFSM==:17B3          ;TOPS20AN ;FINITE MACHINE STATE


;GET

GT%ADR==:1B19                    ;USE ADDRESS LIMITS IN AC2
GT%PRL==:1B20                    ;PRELOAD PAGES
GT%NOV==:1B21                    ;DON'T OVERLAY EXISTING PAGES
GT%FL2==:1B22                    ;IF ON, AC3 CONTAINS FLAGS
```

```
;GETAB - TABLE INDICES

.JOBTT==:0                          ;JOB NUMBER TO TTY NUMBER
.JOBRT==:1                          ;JOB RUNTIME
.TICKP==:2                          ;TICKS PER SECOND
.JOBDI==:3                          ;JOB NUMBER TO DIRECTORY NUMBERS (OBS)
.TTYJO==:4                          ;TTY NUMBER TO JOB NUMBER
.NCPGS==:5                          ;NUMBER PHYSICAL CORE PAGES
.DEVNA==:6                          ;DEVICE NAME
.DEVCH==:7                          ;DEVICE CHARACTERISTICS
.DEVUN==:10                         ;DEVICE UNIT NUMBERS
.DSKER==:11                         ;DISK ERROR WORDS
.DRMER==:12                         ;DRUM ERROR WORDS
.SYSVE==:13                         ;VERSION TEXT
.SYSTA==:14                         ;STATISTICS
.QTIME==:15                         ;SCHED QUEUE TIMES
.JOBNA==:16                         ;JOB NUMBER TO PROGRAM NAME
.SNAME==:17                         ;SUBSYSTEM NAME
.STIME==:20                         ; " TIME
.SPFLT==:21                         ; " PAGE FAULTS
.SSIZE==:22                         ; " SIZE INTEGRAL
.SNBLK==:23                         ; " NUMBER WAKEUPS
.DBUGS==:24                         ;DBUGSW, DCHKSW
.LOGDE==:25                         ;LOG, JOB 0 DESIGNATORS
.PTYPA==:26                         ;PTY PARAMETERS
.SYMTA==:27                         ;GTTAB SYMBOL TABLE
.DWNTI==:30                         ;HSYS VARIABLES
.JOBPN==:31                         ;JOB NUMBER TO PROGRAM NAME
.BLDTD==:32                         ;MONITOR BUILD TIME AND DATE
.LSTDR==:33                         ;LAST DIR NUMBER ASSIGNED (OBS)
.APRID==:34                         ;APR SERIAL NUMBER
.HQLAV==:35                         ;HIGH QUEUE LOAD AVERAGES
.LQLAV==:36                         ;LOW QUEUE LOAD AVERAGES
.NETRD==:37          ;TOPS20AN      ;ARPANET STATUS
.IMPHR==:40          ;TOPS20AN      ;HOST READY
.HSTST==:41          ;TOPS20AN      ;DEAD HOST STATUS
.HSTNA==:42          ;TOPS20AN      ;HOST NAMES
.HOSTN==:43          ;TOPS20AN      ;HOST NAME INDEX
.NETLS==:44          ;TOPS20AN      ;LOCAL SOCKET
.NETFS==:45          ;TOPS20AN      ;FOREIGN SOCKET
.NETAW==:46          ;TOPS20AN      ;ARPA CONNECTION ADDRESS
.NETBA==:47          ;TOPS20AN      ;BIT ALLOCATION
.NETST==:50          ;TOPS20AN      ;CONNECTION STATUS
.NETBU==:51          ;TOPS20AN      ;ARPANET BUFFERS
.NETBT==:52          ;TOPS20AN      ;BYTE COUNT STATISTICS
.IMPL1==:53          ;TOPS20AN      ;IMP LINK TABLE ONE
.IMPL2==:54          ;TOPS20AN      ;IMP LINK TABLE TWO
.IMPL3==:55          ;TOPS20AN      ;IMP LINK TABLE THREE
.IMPL4==:56          ;TOPS20AN      ;IMP LINK TABLE FOUR
.LHOST==:57          ;TOPS20AN      ;LOCAL HOST NUMBER
.JBONT==:60                         ;OWNING JOB
.NSWPG==:61                         ;DEFAULT SWAPPING PAGES
```

```
;GETJI

.JIJNO==:0                      ;JOB NUMBER
.JITNO==:1                      ;TERMINAL NUMBER
.JIUNO==:2                      ;USER NUMBER
.JIDNO==:3                      ;DIRECTORY NUMBER
.JISNM==:4                      ;SUBSYS NAME
.JIPNM==:5                      ;PROGRAM NAME
.JIRT==:6                       ;RUN TIME
.JICPJ==:7                      ;CONTROLLING PTY JOB NUMBER
.JIRTL==:10                     ;RUN TIME LIMIT (SET BY TIMER JSYS)
.JIBAT==:11                     ;CONTROLLED BY BATCH
.JIDEN==:12                     ;MAGTAPE DEFAULT DENSITY
.JIPAR==:13                     ;MAGTAPE DEFAULT PARITY
.JIDM==:14                      ;MAGTAPE DEFAULT DATA MODE
.JIRS==:15                      ;MAGTAPE DEFAULT RECORD SIZE
.JIDFS==:16                     ;DEFERRED SPOOLING
.JILNO==:17                     ;LOGGED-IN DIRECTORY NUMBER
.JISRM==:20                     ;POINTER TO JOB SESSION REMARK
.JILLN==:21                     ;LAST LOGIN DATE & TIME
.JISRT==:22                     ;JOB RUNTIME AT START OF THIS ACCOUNTING SESSION
.JISCT==:23                     ;JOB CONSOLE TIME AT START OF THIS SESSION


;GFRKS

GF%GFH==:1B0                    ;GET RELATIVE FORK HANDLES
GF%GFS==:1B1                    ;GET FORK STATUS


;GFUST

.GFAUT==:0                      ;GET FILE AUTHOR
.GFLWR==:1                      ;GET FILE LAST WRITER
```

```
;GTJFN DEFINITIONS

;FLAGS PROVIDED TO GTJFN ON CALL

GJ%FOU==:1B0                    ;FILE IS FOR OUTPUT USE
GJ%NEW==:1B1                    ;NEW FILE ONLY
GJ%OLD==:1B2                    ;OLD FILE ONLY
GJ%MSG==:1B3                    ;PRINT AN APPROPRIATE MESSAGE
GJ%CFM==:1B4                    ;CONFIRMATION IS REQUIRED
GJ%TMP==:1B5                    ;TEMPORARY
GJ%NS==:1B6                     ;DONT SEARCH SEARCH LISTS
GJ%ACC==:1B7                    ;NO ACCESS BY OTHER FORKS
GJ%DEL==:1B8                    ;IGNORE "DELETED" BIT
GJ%JFN==:3B10                   ;JFN USE FIELD
.GJDNU==:0                      ;DO NOT USE JFN PROVIDED
.GJERR==:2                      ;ERROR IF CANNOT USE JFN PROVIDED
.GJALT==:3                      ;USE ALTERNATE IF CANNOT USE GIVEN JFN
GJ%IFG==:1B11                   ;ACCEPT INPUT FILE GROUP DESCRIPTORS
GJ%OFG==:1B12                   ;ACCEPT OUTPUT FILE GROUP DESCRIPTORS
GJ%FLG==:1B13                   ;RETURN FLAGS
GJ%PHY==:1B14                   ;PHYSICAL DEVICE ONLY
GJ%XTN==:1B15                   ;EXTENDED FORMAT (E+11 EXISTS)
GJ%FNS==:1B16                   ;ACCUMULATOR 2 CONTAINS JOB FILE NUMBERS
GJ%SHT==:1B17                   ;SHORT CALL FORMAT

;FLAGS PROVIDED TO GTJFN (IN SECOND FLAG WORD)

G1%RND==:1B0                    ;RETURN ON NULL(IN ALTERNATE FLAG WORD)
G1%RBF==:1B1                    ;^R BUFFER IS DISJOINT (OBSOLETE)
G1%NLN==:1B2                    ;NO LONG NAMES
G1%RCM==:1B3                    ;RETURN CONFIRM MESSAGE
G1%RIE==:1B4                    ;RETURN WHEN MAIN STRING IS EMPTY
```

```
;FLAGS RETURNED BY GTJFN

GJ%DEV==:1B0                    ;ASTERISK WAS GIVEN FOR DEVICE
GJ%UNT==:1B1                    ;ASTERISK WAS GIVEN FOR UNIT
GJ%DIR==:1B2                    ;ASTERISK WAS GIVEN FOR DIRECTORY
GJ%NAM==:1B3                    ;ASTERISK WAS GIVEN FOR NAME
GJ%EXT==:1B4                    ;ASTERISK WAS GIVEN FOR EXTENSION
GJ%VER==:1B5                    ;ASTERISK WAS GIVEN FOR GENERATION
GJ%UHV==:1B6                    ;USE HIGHEST GENERATION
GJ%NHV==:1B7                    ;USE NEXT HIGHER GENERATION
GJ%ULV==:1B8                    ;USE LOWEST GENERATION
GJ%PRO==:1B9                    ;PROTECTION GIVEN
GJ%ACT==:1B10                   ;ACCOUNT GIVEN
GJ%TFS==:1B11                   ;TEMPORARY FILE SPECIFIED (;T)
GJ%GND==:1B12                   ;COMPLEMENT OF GJ%DEL ON CALL

;GTJFN TABLE OFFSETS

.GJGEN==:0                      ;FLAGS ,, GENERATION
.GJDEF==:<Z 0>                  ;DEFAULT GENERATION
.GJNHG==:<Z -1>                 ;NEXT HIGHER GENERATION
.GJLEG==:<Z -2>                 ;LOWEST EXISTING GENERATION
.GJALL==:<Z -3>                 ;ALL GENERATIONS (I.E., ;*)
.GJSRC==:1                      ;SOURCE JFN ,, OUTPUT JFN
.GJDEV==:2                      ;DEFAULT DEVICE
.GJDIR==:3                      ;DEFAULT DIRECTORY
.GJNAM==:4                      ;DEFAULT NAME
.GJEXT==:5                      ;DEFAULT EXTENSTION
.GJPRO==:6                      ;DEFAULT PROTECTION
.GJACT==:7                      ;DEFAULT ACCOUNT
.GJJFN==:10                     ;DESIRED JFN
.GJF2==:11                      ;SECOND GROUP FLAGS,,COUNT
.GJCPP==:12                     ;COPY BUFFER POINTER
.GJCPC==:13                     ;COPY BUFFER COUNT
.GJRTY==:14                     ;RETYPE (^R) POINTER
.GJBFP==:15                     ;TOP OF BUFFER POINTER
.GJATR==:16                     ;POINTER TO ARBITRARY ATTRIBUTE BLOCK

;GNJFN - FLAGS RETURNED

GN%STR==:1B13                   ;STRUCTURE CHANGED
GN%DIR==:1B14                   ;DIRECTORY CHANGED
GN%NAM==:1B15                   ;NAME CHANGED
GN%EXT==:1B16                   ;EXTENSION CHANGED
```

```
;GTRPW

PF%USR==:1B0                    ;PAGE FAIL WORD - USER MODE REFERENCE
PF%WRT==:1B5                    ; " - WRITE REFERENCE
TSW%RD==:1B14                   ;TRAP STATUS WORD - READ
TSW%WT==:1B15                   ; " - WRITE
TSW%WR==:1B15                   ; (ANOTHER NAME FOR ABOVE)
TSW%EX==:1B16                   ; " - EXECUTE
TSW%MN==:1B17                   ; " - MONITOR MODE REFERENCE


;GTSTS BITS RETURNED IN 2

GS%OPN==:1B0                    ;FILE IS OPEN
GS%RDF==:1B1                    ;IF OPEN, FILE IS OPEN FOR READ
GS%WRF==:1B2                    ;IF OPEN, FILE IS OPEN FOR WRITE
GS%XCF==:1B3                    ;IF OPEN, FILE IS OPEN FOR EXECUTE
GS%RND==:1B4                    ;OK TO RESET BYTE POINTER
                                ; (FILE IS NOT APPEND)
GS%APT==:1B5                    ;ACCESS PER PAGE TABLE
                                ; (NOT IMPLEMENTED -- OBSOLETE)
GS%CAL==:1B6                    ;OK TO CALL AS A PROCEDURE
                                ; (NOT IMPLEMENTED -- OBSOLETE)
GS%LNG==:1B7                    ;FILE IS LONG
GS%EOF==:1B8                    ;AT END OF FILE ON READ
GS%ERR==:1B9                    ;FILE MAY BE IN ERROR
GS%NAM==:1B10                   ;FILE HAS A NAME (JFN EXISTS)
GS%AST==:1B11                   ;ONE OR MORE FIELDS OF NAME
                                ; IS WILD
GS%ASG==:1B12                   ;JFN IS BEING ASSIGNED
GS%HLT==:1B13                   ;TERMINATE ON I/O ERROR
GS%FRK==:1B17                   ;FILE IS RESTRICTED TO SOME FORK
GS%MOD==:17B35                  ;DATA MODE
        .GSNRM==:0              ;NORMAL MODE
        .GSIMG==:10             ;IMAGE (BINARY) MODE
        .GSDMP==:17             ;DUMP MODE
```

```
;HPTIM

.HPELP==:0                      ;ELAPSED TIME
.HPRNT==:1                      ;RUN TIME


;IDCNV (ALSO IDTNC AND ODCNV)

IC%DSA==:1B0                    ;DAYLIGHT SAVINGS IF APPROPRIATE
IC%ADS==:1B1                    ;APPLY DAYLIGHT SAVINGS
IC%UTZ==:1B2                    ;USE TIME ZONE GIVEN
IC%JUD==:1B3                    ;USE JULIAN DATE CONVERSION
IC%TMZ==:77B17                  ;TIME ZONE
IC%TIM==777777B35               ;LOCAL TIME


;IDTIM & IDTNC

IT%NDA==:1B0                    ;NO DATE
IT%NNM==:1B1                    ;NO NUMERIC MONTH
IT%SNM==:1B2                    ;SECOND NUMBER IS MONTH
IT%ERR==:1B3                    ;ERROR IF NUMBERS ARE NOT IN SPECIFIED
                                ; ORDER
IT%NTI==:1B6                    ;NO TIME
IT%NIS==:1B7                    ;NO SECONDS
IT%AIS==:1B8                    ;ALWAYS INCLUDE SECONDS
IT%NAC==:1B9                    ;NO COLON ALLOWED BETWEEN HH AND MM
IT%AAC==:1B10                   ;ALWAYS ALLOW COLON
IT%AMS==:1B11                   ;ALWAYS INTERPRET ONE COLON AS HHMM:SS
IT%AHM==:1B12                   ;ALWAYS INTERPRET ONE COLON AS HH:MM
IT%N24==:1B14                   ;NO 24-HOUR FORMAT
IT%NTM==:1B15                   ;NO TIME MODIFIER (AM, PM)
IT%NTZ==:1B16                   ;NO TIME ZONE


;INLNM

.INLJB==:0                      ;GET JOB WIDE LOGICAL NAME FROM INDEX
.INLSY==:1                      ;GET SYSTEM LOGICAL NAME FROM INDEX
```

```
;IPCF BIT DEFINITIONS AND DATA STRUCTURES

;PACKET FORMAT

.IPCFL==:0                      ;FLAGS WORD
IP%CFB==:1B0                    ;DON'T BLOCK READ
IP%CFS==:1B1                    ;INDIRECT SENDER'S PID
IP%CFR==:1B2                    ;INDIRECT RECEIVER'S PID
IP%CFO==:1B3                    ;OVERDRAW SEND
IP%TTL==:1B4                    ;TRUNCATE ON TOO LARGE MESSAGE
IP%CPD==:1B5                    ;CREATE A PID ON THE SEND
IP%JWP==:1B6                    ;MAKE THE CREATED PID BE JOB WIDE
IP%NOA==:1B7                    ;NO ACCESS OF PID BY OTHER FORKS
IP%CFP==:1B18                   ;SENDER IS PRIV'D AND IS ENVOKING PRIVS
IP%CFV==:1B19                   ;PAGE TRANSFER MODE
IP%CFZ==:1B20                   ;ZERO LENGTH MESSAGE WAS SENT
IP%CFE==:77B29                  ;ERROR FIELD

;ERRORS SENT BY INFO

 .IPCPI==:15                    ;INSUFFICIENT PRIVILEGE
 .IPCUF==:16                    ;ILLEGAL FUNCTION
 .IPCSN==:67                    ;SEND INFO YOUR NAME
 .IPCFF==:72                    ;INFO FREE SPACE EXHAUSTED
 .IPCBP==:74                    ;PID HAS NO NAME OR IS ILLEGAL
 .IPCDN==:75                    ;DUPLICATE NAME
 .IPCNN==:76                    ;UNKNOWN NAME
 .IPCEN==:77                    ;ILLEGAL NAME
IP%CFC==:7B32                   ;SYSTEM SENDER CODE
 .IPCCC==:1                     ;SENT BY [SYSTEM]IPCF
 .IPCCF==:2                     ;SENT BY SYSTEM WIDE [SYSTEM]INFO
 .IPCCP==:3                     ;SENT BY RECEIVER'S [SYSTEM]INFO
IP%CFM==:7B35                   ;SPECIAL MESSAGE RETURN FIELD
 .IPCFN==:1                     ;MESSAGE WAS NOT DELIVERED
.IPCFS==:1                      ;PID OF SENDER
.IPCFR==:2                      ;PID OF RECEIVER
.IPCFP==:3                      ;POINTER TO MESSAGE BLOCK
.IPCFD==:4                      ;LOGGED IN DIR OF SENDER
.IPCFC==:5                      ;ENABLED CAPABILITIES OF SENDER
.IPCSD==:6                      ;CONNECTED DIRECTORY NUMBER OF SENDER
.IPCAS==:7                      ;POINTER TO ACCOUNT STRING OF SENDER
.IPCSU==:26                     ;SPOOL MESSAGE CODE FROM IPCC
.IPCSL==:27                     ;LOGOUT MESSAGE CODE FROM IPCC
.IPCSA==:30                     ;RESOURCE ALLOCATOR MESSAGE CODE
.IPCDS==:31                     ;STRUCTURE DISMOUNT MESSAGE CODE FROM IPCC
.IPCLI==:32                     ;LOGIN MESSAGE CODE FROM IPCC
.IPCLO==:33                     ;LOGOUT MESSAGE TO CREATOR FROM IPCC
.IPCKP==:34                     ;DELETED PID MESSAGE FROM IPCC
.IPCCA==:35                     ;CREATE AN APPLICATION (RESERVED FOR TPS USE)

.IPCSS==:15                     ;IPCC REQUEST TO INFO TO DELETE PIDS
```

```
;[SYSTEM] INFO DEFINITIONS

.IPCI0==:0                      ;CODE,,FUNCTION
.IPCIW==:1                      ;FIND PID FOR NAME
.IPCIG==:2                      ;FIND NAME FOR PID
.IPCII==:3                      ;ASSIGN NAME TO PID
.IPCIJ==:4                      ;ASSIGN NAME TO PID
.IPCIS==:15                     ;MONITOR DROP PID FUNCTION
.IPCI1==:1                      ;PID TO GET A COPY OF REPLY
.IPCI2==:2                      ;START OF DATA


;JFNS

JS%DEV==7B2                     ;DEVICE  FIELD OUTPUT CONTROL
JS%DIR==:7B5                    ;DIRECTORY FIELD OUTPUT CONTROL
JS%NAM==:7B8                    ;NAME FIELD OUTPUT CONTROL
JS%TYP==:7B11                   ;FILE TYPE FIELD OUTPUT CONTROL
JS%GEN==:7B14                   ;GENERATION FIELD OUTPUT CONTROL
JS%PRO==:7B17                   ;PROTECTION FIELD OUTPUT CONTROL
JS%ACT==:7B20                   ;ACCOUNT FIELD OUTPUT CONTROL
        ;VALUES FOR ABOVE 7 FIELDS:
        .JSNOF==:0              ;NEVER OUTPUT FIELD
        .JSAOF==:1              ;ALWAYS OUTPUT FIELD
        .JSSSD==:2              ;SUPPRESS IF SYSTEM DEFAULT
JS%TMP==:1B21                   ;RETURN ;T IF TEMP FILE
JS%SIZ==:1B22                   ;RETURN SIZE
JS%CDR==:1B23                   ;RETURN CREATION DATE
JS%LWR==:1B24                   ;RETURN LAST WRITE
JS%LRD==:1B25                   ;RETURN LAST READ
JS%PTR==:1B26                   ;AC 2 HOLDS STRING POINTER NOT JFN
JS%ATR==:1B27                   ;RETURN ATTRIBUTES
JS%AT1==:1B28                   ;RETURN 1 SPECIFIC ATTRIBUTE
JS%PSD==:1B32                   ;PUNCTUATE SIZE AND DATE
JS%TBR==:1B33                   ;TAB BEFORE FIELDS RETURNED
JS%TBP==:1B34                   ;TAB BEFORE POSSIBLE FIELDS
JS%PAF==:1B35                   ;PUNCTUATE ALL FIELDS


;LNMST

.LNSJB==:0                      ;GET JOB WIDE DEFINITION OF A LN
.LNSSY==:1                      ;GET SYSTEM DEFINITION OF A LOGICAL NAME


;LOCK

LK%CNT==:1B0                    ;USE COUNT IN AC3
LK%PHY==:1B1                    ;USE AC1 AS PHYSICAL PAGE NUMBER
LK%NCH==:1B2                    ;MAP PAGES CACHE INHIBITED
LK%AOL==:1B3                    ;ALLOW LOCKING IN OFFLINE PAGES
```

```
;MSTR

.MSRNU==:0                      ;READ STATUS OF NEXT DISK UNIT
.MSRUS==:1                      ;READ STATUS OF A DISK UNIT
  .MSRCH==:0                    ;CHANNEL NUMBER
  .MSRCT==:1                    ;CONTROLLER NUMBER
  .MSRUN==:2                    ;UNIT NUMBER
  .MSRST==:3                    ;STATUS
    MS%MNT==:1B0                ;THIS UNIT IS PART OF A MOUNTED STRUCTURE
    MS%16B==:1B1                ;THIS UNIT WRITTEN IN 16-BIT MODE
                                ; (RESERVED FOR FUTURE)
    MS%DIA==:1B2                ;THIS UNIT IS CURRENTLY IN USE BY AN
                                ;  ON-LINE DIAGNOSTIC
    MS%OFL==:1B3                ;THIS UNIT IS OFF-LINE
    MS%ERR==:1B4                ;THERE WAS AN ERROR READING THIS UNIT
    MS%BBB==:1B5                ;ONE OF THE BAT BLOCKS IS BAD
    MS%HBB==:1B6                ;ONE OF THE HOME BLOCKS IS BAD
    MS%WLK==:1B7                ;UNIT IS WRITE-LOCKED
    MS%TYP==:777B17             ;DISK TYPE CODE
; DEFINED THE SAME AS .UTTXX IN PHYPAR
    .MSRP4==:1                  ;RP04
    .MSRP5==:5                  ;RP05
    .MSRP6==:6                  ;RP06
    .MSRP7==:7                  ;RP07
    .MSRM3==:11                 ;RM03
  .MSRSN==:4                    ;STRUCTURE NAME
  .MSRSA==:5                    ;STRUCTURE ALIAS
  .MSRNS==:6                    ;UNIT # IN STRUCTURE,,# OF UNITS IN STRUCTURE
  .MSRSW==:7                    ;NUMBER OF PAGES FOR SWAPPING
  .MSRUI==:10                   ;UNIT ID
  .MSROI==:13                   ;OWNER ID
  .MSRFI==:16                   ;FILE-SYSTEM ID
  .MSRSP==:21                   ;NUMBER OF SECTORS PER PAGE
  .MSRSC==:22                   ;NUMBER OF SECTORS PER CYLINDER
  .MSRPC==:23                   ;NUMBER OF PAGES PER CYLINDER
  .MSRCU==:24                   ;NUMBER OF CYLINDERS PER UNIT
  .MSRSU==:25                   ;NUMBER OF SECTORS PER UNIT
  .MSRBT==:26                   ;NUMBER OF BIT-WORDS IN BIT TABLE PER CYLINDER
  .MSRLN==:27                   ;MAX LENGTH OF ARGUMENT BLOCK IN WORDS

.MSMNT==:2                      ;MOUNT A STRUCTURE
  .MSTNM==:0                    ;NAME OF STRUCTURE
  .MSTAL==:1                    ;ALIAS NAME
  .MSTNU==:2                    ;NUMBER OF UNITS IN STRUCTURE
  .MSTFL==:2                    ;FLAGS (LHS)
    MS%FLG==:777777,,0          ;MASK FOR .MSTFL
    MS%NFH==:1B0                ;NO FIX BAD HOME BLOCK
    MS%NFB==:1B1                ;NO FIX BAD BAT BLOCK
    MS%XCL==:1B2                ;MOUNT FOR EXCLUSIVE USE BY JOB
    MS%IGN==:1B3                ;IGNORE ERRORS
  .MSTUI==:3                    ;START OF UNIT INFORMATION
    .MSTCH==:0                  ;CHANNEL NUMBER
    .MSTCT==:1                  ;CONTROLLER NUMBER
    .MSTUN==:2                  ;UNIT NUMBER
    .MSTNO==:3                  ;# OF ARGUMENT WORDS/UNIT
```

```
.MSDIS==:3                      ;DISMOUNT A STRUCTURE
  .MSDNM==:0                    ;NAME OF STRUCTURE

.MSGSS==:4                      ;GET STATUS OF A STRUCTURE
  .MSGSN==:0                    ;STRUCTURE NAME (ALIAS)
  .MSGST==:1                    ;STATUS
    MS%PS==:1B0                 ;STRUCTURE IS A PUBLIC STRUCTURE
    MS%DIS==:1B1                ;STRUCTURE IS BEING DISMOUNTED
    MS%DOM==:1B2                ;STRUCTURE IS DOMESTIC
    MS%PPS==:1B3                ;STRUCTURE IS THE PRIMARY PUBLIC STRUCTURE
    MS%INI==:1B4                ;STRUCTURE IS BEING INITIALIZED
    MS%LIM==:1B5                ;STRUCTURE LIMITED TO 2050 SIZES
  .MSGNU==:2                    ;NUMBER OF UNITS IN STRUCTURE
  .MSGMC==:3                    ;MOUNT COUNT
  .MSGFC==:4                    ;OPEN FILE COUNT
  .MSGSI==:5                    ;STRUCTURE ID
  .MSGLN==:6                    ;LENGTH OF ARGUMENT BLOCK

.MSSSS==:5                      ;SET STATUS OF A STRUCTURE
  .MSSSN==:0                    ;STRUCTURE NAME
  .MSSST==:1                    ;NEW STATUS BITS
  .MSSMW==:2                    ;MASK WORD OF BITS TO BE CHANGED
  .MSSLN==:3                    ;LENGTH OF ARGUMENT BLOCK

.MSINI==:6                      ;INITIALIZE A STRUCTURE
  .MSINM==:0                    ;NAME OF STRUCTURE
  .MSIAL==:1                    ;ALIAS NAME
  .MSINU==:2                    ;NUMBER OF UNITS IN STRUCTURE
  .MSIFL==:2                    ;FLAGS (LHS)
                                ;FLAGS DEFINED IN .MSMNT FUNCTION
    MS%FCN==:77B17              ;FUNCTION CODE
      .MSCRE==:1                ;CREATE NEW FILE SYSTEM
      .MSRRD==:2                ;RECONSTRUCT THE ROOT-DIRECTORY
      .MSWHB==:3                ;WRITE THE HOME BLOCKS
      .MSRIX==:4                        ;REBUILD INDEX TABLE (IDXFIL)
  .MSISU==:3                    ;START OF UNIT INFORMATION
    .MSICH==:0                  ;CHANNEL NUMBER
    .MSICT==:1                  ;CONTROLLER NUMBER
    .MSIUN==:2                  ;UNIT NUMBER
    .MSINO==:3                  ;# OF ARGUMENT WORDS/UNIT
  .MSIST==:6                    ;STATUS WORD
  .MSISW==:7                    ;NUMBER OF PAGES FOR SWAPPING ON THIS UNIT
  .MSIFE==:10                   ;NUMBER OF PAGES FOR FRONT-END FILE SYSTEM
  .MSIUI==:11                   ;UNIT ID
  .MSIOI==:14                   ;OWNER ID
  .MSIFI==:17                   ;FILE SYSTEM ID
  .MSIFB==:22                   ;NUMBER OF PAGES FOR  BOOTSTRAP.BIN (OPTIONAL)

.MSIMC==:7                      ;INCREMENT MOUNT COUNT
.MSDMC==:10                     ;DECREMENT MOUNT COUNT
  .MSDEV==:0                    ;DEVICE DESIGNATOR OR STRUCTURE
```

```
.MSGSU==:11                     ;GET STRUCTURE USERS
  .MSUAL==:0                    ;POINTER TO ALIAS OF STRUCTURE
  .MSUFL==:1                    ;FLAGS,,# OF ITEMS RETURNED
    MS%GTA==:1B0                ;GET USERS WHO HAVE ACCESSED STRUCTURE
    MS%GTM==:1B1                ;GET USERS WHO HAVE MOUNTED STRUCTURE
    MS%GTC==:1B2                ;GET USERS WHO ARE CONNECTED TO STRUCTURE
  .MSUJ1==:2                    ;FIRST JOB NUMBER RETURNED

.MSHOM==:12                     ;MODIFY HOMEBLOCK WORD
  .MSHNM==:0                    ;POINTER TO ALIAS, OR DESIGNATOR FOR ALIAS
  .MSHOF==:1                    ;OFFSET INTO HOMEBLOCK OF WORD BEING CHANGED
  .MSHVL==:2                    ;NEW VALUES FOR BITS BEING CHANGED
  .MSHMK==:3                    ;MASK DECLARING WHICH BITS BEING CHANGED
```

```
;MTOPR - FUNCTION CODES

.MOCLE==:0                      ;CLEAR ERRORS
.MONOP==:31                     ;NOP (WAIT FOR ACTIVITY TO STOP)
.MOREW==:1                      ;REWIND
.MOEOF==:3                      ;WRITE EOF
.MODTE==:4                      ;ASSIGN FE DEVICE TO A DTE
.MOFWR==:6                      ;FORWARD SPACE RECORD
.MOBKR==:7                      ;BACKSPACE RECORD
.MORUL==:11                     ;REWIND AND UNLOAD
.MOERS==:13                     ;ERASE TAPE
.MOFWF==:16                     ;FORWARD SPACE FILE
.MOBKF==:17                     ;BACKSPACE FILE
.MOSPD==:26                     ;SET TTY SPEED (FOR KL ONLY)
.MORSP==:27                     ;READ LINE SPEED (FOR KL ONLY)
   MO%RMT==:1B0                 ;FLAG TO SAY LINE IS REMOTE
   MO%AUT==:1B1                 ;FLAG TO SAY LINE IS "AUTO" SPEED
                                ; (RSX20F ONLY)
.MOSDR==:2                      ;SET READ DIRECTION
.MORDR==:26                     ;READ READ DIRECTION
.MOEOT==:10                     ;SKIP TO LOGICAL END OF TAPE
.MOSRS==:5                      ;SET RECORD SIZE
.MORRS==:15                     ;READ RECORD SIZE
.MOSDN==:24                     ;SET DENSITY
.MORDN==:12                     ;READ DENSITY
.MOSDM==:4                      ;SET DATA MODE
.MORDM==:14                     ;READ DATA MODE
.MOSPR==:20                     ;SET PARITY
.MORPR==:21                     ;READ PARITY
.MONRB==:22                     ;GET NUMBER OF REMAINING BYTES IN RECORD
.MOFOU==:23                     ;FORCE OUT RECORD
.MOINF==:25                     ;GET INFORMATION ABOUT TAPE
   .MOICT==:0                   ;COUNT OF ARGUMENTS TO BE RETURNED
   .MOITP==:1                   ;MAGTAPE TYPE CODE
; DEFINED THE SAME AS .UTTXX IN PHYPAR
      .MTT45==:3                ;MAGTAPE TYPE TU45
      .MTT70==:17               ;MAGTAPE TYPE TU70
      .MTT71==:20               ;MAGTAPE TYPE TU71
      .MTT72==:21               ;MAGTAPE TYPE TU72
   .MOIID==:2                   ;MAGTAPE REEL ID
   .MOISN==:3                   ;CHAN,CONTROLLER,UNIT ,, SERIAL #
   .MOIRD==:4                   ;# OF READS DONE
   .MOIWT==:5                   ;# OF WRITES DONE
   .MOIRC==:6                   ;RECORD # FROM BOT
   .MOIFC==:7                   ;FILE COUNT ON TAPE
   .MOISR==:10                  ;# OF SOFT READ ERRORS
   .MOISW==:11                  ;# OF SOFT WRITE ERRORS
   .MOIHR==:12                  ;# OF HARD READ ERRORS
   .MOIHW==:13                  ;# OF HARD WRITE ERRORS
.MOPSI==:27                     ;SET ERROR PSI FOR LPT AND CDR
   MO%MSG==:1B0                 ;SUPPRESS STANDARD CTY MESSAGES
.MOSID==:27                     ;SET REEL I.D.
.MOIEL==:30                     ;INHIBIT ERROR LOGGING
```

```
.MOLVF==:32                     ;LOAD DEVICE'S VFU
.MORVF==:33                     ;READ VFU FILE NAME
.MOLTR==:34                     ;LOAD TRANSLATION RAM
.MORTR==:35                     ;READ RAM FILE NAME
.MOSTS==:36                     ;SET SOFTWARE STATUS
.MORST==:37                     ;READ SOFTWARE STATUS
   MO%LPC==1                    ;PAGE COUNTER OVERFLOW
   MO%LCI==2                    ;CHARACTER INTERRUPT (HARD ERROR)
   MO%LVF==4                    ;VFU ERROR.  PAPER MUST BE RE-ALIGNED
   MO%LVU==20                   ;LINE PRINTER HAS OPTICAL VFU
   MO%RPE==40                   ;RAM PARITY ERROR

   MO%RCK==:1                   ;READ CHECK
   MO%PCK==:2                   ;PICK CHECK
   MO%SCK==:4                   ;STACK CHECK
   MO%HEM==:10                  ;HOPPER EMPTY
   MO%SFL==:20                  ;STACKER FULL

   MO%FNX==:1B17                ;NON-EXISTENT DEVICE
   MO%OL==:1B16                 ;DEVICE IS OFF-LINE
   MO%HE==:1B15                 ;HARDWARE ERROR
   MO%SER==:1B14                ;SOFTWARE ERROR
   MO%IOP==:1B13                ;I/O IN PROGRESS
   MO%EOF==:1B12                ;END OF FILE
;  1B11                         ;RESERVED
   MO%FER==:1B10                ;FATAL ERROR
   MO%LCP==:1B0                 ;LOWER CASE PRINTER
   MO%RLD==:1B1                 ;FRONT-END WAS RELOADED
.MOFLO==:40                     ;FLUSH OUTPUT

;SEE SETJB FOR VARIOUS ARGUMENT VALUES

.MOSNT==:34                     ;SET TTY NON-TERMINAL STATUS
   .MOSMN==:1                   ;NO SYSTEM MESSAGES(I.E. SUPPRESS)
   .MOSMY==:0                   ;YES SYSTEM MESSAGES(DEFAULT)
.MORNT==:35                     ;READ TTY NON-TERMINAL STATUS

;PTY MTOPR NUMBERS

.MOAPI==:24                     ;ASSIGN PTY INTERRUPT CHANNELS
   MO%WFI==:1B0                 ;ENABLE WAITING FOR INPUT
   MO%OIR==:1B1                 ;ENABLE OUTPUT IS WAITING
   MO%SIC==:77B17               ;SOFTWARE INTERRUPT CHANNEL
.MOPIH==:25                     ;TEST PTY INPUT HUNGRY
   .MONWI==:0                   ;NOT WAITING FOR INPUT
   .MOWFI==:-1                  ;WAITING FOR INPUT
.MOBAT==:26                     ;SET BATCH BIT
  .MOJCB==:1                    ;JOB CONTROLLED BY BATCH
  .MONCB==:0                    ;JOB NOT CONTROLLED BY BATCH
```

```
;TTY MODE DEFINITIONS

.MORLW==:30                         ;READ WIDTH
.MOSLW==:31                         ;SET WIDTH
.MORLL==:32                         ;READ LENGTH
.MOSLL==:33                         ;SET LENGTH
.MOSIG==:36                         ;SET "IGNORE INPUT WHEN INACTIVE" BIT
.MORBM==:37                         ;READ 128 CHARACTER BREAK MASK

        MO%WN1==:776117,,777740 ;BIT DEFINITIONS FOR NON-FORMATTING CONTROL
        MO%WN2==:0              ;FOR ASCII CODES 40-777
        MO%WN3==:0              ;FOR ASCII CODES 100-137
        MO%WN4==:20             ;FOR ASCII CODES 137-177

        MO%WF1==:001260,,000420 ;FORMATTING CONTROL BITS
        MO%WF2==:0              ;FOR ASCII CODES 40-77
        MO%WF3==:0              ;FOR ASCII CODES 100-137
        MO%WF4==:20             ;FOR ASCII CODES 140-177

        MO%WP1==:000400,,400    ;PUNCTUATION BIT DEFINITIONS
        MO%WP2==:777774,,001760 ; FOR ASCII CODES 40-77
        MO%WP3==:400000,,000760 ; FOR ASCII CODES 100-137
        MO%WP4==:400000,,000760 ; FOR ASCII CODES 140-177

        MO%WA1==:400            ;ALPHANUMERICS DEFENITIONS
        MO%WA2==:000003,,776000 ; FOR ASCII CODES 40-77
        MO%WA3==:377777,,777000 ; FOR ASCII CODES 100-137
        MO%WA4==:377777,,777020 ; FOR ASCII CODES 140-177
.MOSBM==:40                         ;SET 128 CHARACTER BREAK MASK
.MORFW==:41                         ;READ FIELD WIDTH
.MOSFW==:42                         ;SET FIELD WIDTH

;NET MTOPR NUMBERS

.MOACP==:20             ;TOPS20AN ;ACCEPT CONNECTION ON SOCKET
.MOSND==:21             ;TOPS20AN ;SEND ALL CURENTLY BUFFERED BYTES
.MOSIN==:22             ;TOPS20AN ;SEND INS/INR COMMAND
.MOAIN==:24             ;TOPS20AN ;ASSIGN INS/INR AND FSM PSI CHANNELS
   MO%NIN==:77B5        ;TOPS20AN ;INS/INR SOFTWARE INTERRUPT CHANNEL
   MO%FSM==:77B17       ;TOPS20AN ;FSM CHANGE OF STATE INTERRUPT CHANNEL


;DEFINITIONS FOR DECNET

.MOACN==:24                         ;ASSIGN CONNECT INTERRUPT CHANNEL
   MO%CDN==:777B8                   ;CONNECT INTERRUPT CHANNEL
   MO%INA==:777B17                  ;INTERRUPT MESSAGE CHANNEL
   MO%DAV==:777B26                  ;DATA AVAILABLE CHANNEL
     .MONCI==:777                   ;NO CHANGE
     .MOCIA==:776                   ;CLEAR INTERRUPT ASSIGNMENT

.MORLS==:25                         ;READ LINK STATUS
   MO%CON==:1B0                     ;LINK IS CONNECTED
   MO%SRV==:1B1                     ;LINK IS A SERVER
   MO%WFC==:1B2                     ;WAITING FOR A CONNECT
   MO%WCC==:1B3                     ;WAITING FOR THIS LINK TO CONFIRM
   MO%EOM==:1B4                     ;EOM PRESENT IN INPUT BUFFER
   MO%ABT==:1B5                     ;CONNECTION ABORTED
   MO%SYN==:1B6                     ;SYNCH DI RECIEVED
```

```
   MO%INT==:1B7                 ;INT MESSAGE AVAILABLE
   MO%LWC==:1B8                 ;LINK WAS CONNECTED
.MORHN==:26                     ;READ HOST NAME
.MORTN==:27                     ;READ TASK NAME
.MORUS==:30                     ;READ USER DATA
.MORPW==:31                     ;READ PASSWORD
.MORAC==:32                     ;READ ACCOUNT
.MORDA==:33                     ;READ OPTIONAL DATA
.MORCN==:34                     ;READ CONNECT OBJECT NUMBER
.MORIM==:35                     ;READ INTERRUPT MESSAGE
.MOSIM==:36                     ;SEND INTERRUPT MESSAGE
.MOROD==:37                     ;READ OBJ-DESC OF CONNECTION
.MOCLZ==:40                     ;CLOSE/REJECT A CONNECTION
.MOCC==:41                      ;ACCEPT A CONNECTION
.MORSS==:42                     ;READ SEGMENT SIZE

;DEFINITIONS FOR ATS

;FUNCTION CODES FOR MTOPR ARE IN COLUMN 1

.MOAMO==:1                      ;SET MODE WORD
        .MOAMM==:1              ;MESSAGE MODE
        .MOADM==:2              ;DATA MODE
.MOAAT==:2                      ;ACQUIRE TERMINAL
.MOASI==:3                      ;ENABLE INTERRUPTS
        MO%IFL==:777B8          ;FUNCTION TO BE PERFORMED
          .MOAAI==:0            ;ASSIGN INTERRUPT CHANNEL
          .MOADI==:1            ;DEASSIGN INTERRUPT CHANNEL
        MO%IEV==:777B17         ;EVENT BEING ASSIGNED OR DEASSIGNED
          .MOADT==:0            ;DATA ARRIVAL
          .MOAST==:1            ;STATUS ARRIVAL
        MO%ACH==:777777B35      ;CHANNEL NUMBER
.MORCD==:4                      ;GET STATUS
        MO%WDV==:777B35         ;WHICH DEVICES TO REPORT ON
          .MOALD==:0            ;ALL TERMINALS
          .MOCHG==:1            ;TERMINALS WHOSE STATUS HAS CHANGED
          .MOLST==:2            ;TERMINALS SPECIFIED IN LIST
        MO%ARM==:1B0            ;ASK THE RESOURCE MANAGER
        MO%MDA==:1B1            ;MORE DATA AVAILABLE FOR THIS JFN
.MOADE==:5                      ;DEASSIGN TERMINAL
        MO%AAB==:1B0            ;DON'T SEND REMAINING DATA
```

```
;MUTIL JSYS FUNCTION CODES

.MUENB==:1                      ;ENABLE PID FOR RECEIVING
.MUDIS==:2                      ;DISABLE PID FROM RECEIVING
.MUGTI==:3                      ;GET PID OF [SYSTEM]INFO
.MUCPI==:4                      ;CREATE A PRIVATE INFO FOR A JOB
.MUDES==:5                      ;DESTROY A PID
.MUCRE==:6                      ;CREATE A PID
.MUSSQ==:7                      ;SET SEND AND RECEIVE QUOTAS
.MUCHO==:10                     ;CHANGE OWNER OF A PID
.MUFOJ==:11                     ;FIND OWNER'S JOB NUMBER
.MUFJP==:12                     ;FIND JOB'S PIDS
.MUFSQ==:13                     ;FIND SEND AND RECEIVE QUOTAS
.MUFFP==:15                     ;FIND FORK'S PIDS
.MUSPQ==:16                     ;SET PID QUOTA
.MUFPQ==:17                     ;FIND PID QUOTA
.MUQRY==:20                     ;QUERY
.MUAPF==:21                     ;ASSOCIATE A PID WITH A FORK
.MUPIC==:22                     ;PUT PID ON AN INTERRUPT CHANNEL
.MUDFI==:23                     ;DEFINE PID OF [SYSTEM]INFO
.MUSSP==:24                     ;SET SYSTEM PID TABLE
.MURSP==:25                     ;READ SYSTEM PID TABLE
.MUMPS==:26                     ;GET MAXIMUM PACKET SIZE
.MUSKP==:27                     ;SET PID TO RECEIVE KILLED PID MESSAGE
.MURKP==:30                     ;READ PID THAT RECEIVES KILLED PID MESSAGES


;SYSTEM PID TABLE INDEX VALUES

.SPIPC==:0                      ;PID OF IPCC
.SPINF==:1                      ;PID OF INFO
.SPQSR==:2                      ;PID OF QUASAR
.SPMDA==:3                      ;PID OF QSRMDA
.SPOPR==:4                      ;PID OF OPERATOR JOB (ORION)


;NODE

.NDSLN==:0                      ;SET LOCAL NODE NAME
.NDGLN==:1                      ;GET LOCAL NODE NAME
   .NDNOD==:0                   ;POINTER TO NODE NAME
.NDSNM==:2                      ;SET LOCAL NODE NUMBER
.NDGNM==:3                      ;GET LOCAL NODE NUMBER
.NDSLP==:4                      ;SET LOOPBACK ON PORT
        .NDPRT==:0              ;PORT TO SET IN LOOPBACK
.NDCLP==:5                      ;CLEAR LOOPBACK ON PORT
.NDFLP==:6                      ;FIND LOOPBACK PORT
        ND%LPR==1B0             ;LOOPBACK RUNNING
        ND%LPA==1B1             ;LOOPBACK ASSIGNED TO PORT

;NOUT

NO%MAG==:1B0                    ;OUTPUT MAGNITUDE
NO%SGN==:1B1                    ;OUTPUT SIGN
NO%LFL==:1B2                    ;LEADING FILLER
NO%ZRO==:1B3                    ;FILL WITH ZERO'S
NO%OOV==:1B4                    ;OUTPUT ON COLUMN OVERFLOW
NO%AST==:1B5                    ;OUTPUT ASTERISKS ON OVERFLOW
NO%COL==:177B17                 ;NUMBER OF COLUMNS TO USE
NO%RDX==:777777                 ;RADIX
```

```
;ODCNV -- SEE IDCNV FOR BITS

;ODTIM

OT%NDA==:1B0                    ;DO NOT OUTPUT DATE
OT%DAY==:1B1                    ;OUTPUT DAY OF WEEK
OT%FDY==:1B2                    ;OUTPUT NUMERIC MONTH
OT%NMN==:1B3                    ;OUTPUT NUMERIC MONTH
OT%FMN==:1B4                    ;OUTPUT MONTH IN FULL
OT%4YR==:1B5                    ;OUTPUT 4-DIGIT YEAR
OT%DAM==:1B6                    ;OUTPUT DAY AFTER MONTH
OT%SPA==:1B7                    ;OUTPUT SPACES IN DATE
OT%SLA==:1B8                    ;OUTPUT SLASHES IN DATE
OT%NTM==:1B9                    ;DO NOT OUTPUT TIME
OT%NSC==:1B10                   ;DO NOT OUTPUT SECONDS
OT%12H==:1B11                   ;OUTPUT 12-HOUR FORMAT
OT%NCO==:1B12                   ;DO NOT OUTPUT COLON
OT%TMZ==:1B13                   ;OUTPUT TIME ZONE
OT%SCL==:1B17                   ;SUPPRESS COLUMNIZATION

;ODTNC -- SEE IDCNV FOR BITS

;OPENF

OF%BSZ==:77B5                   ;BYTE SIZE
OF%MOD==:17B9                   ;MODE
OF%HER==:1B18                   ;HALT ON IO ERROR
OF%RD==:1B19                    ;READ
OF%WR==:1B20                    ;WRITE
OF%EX==:1B21                    ;EXECUTE (RESERVED FOR THE FUTURE)
OF%APP==:1B22                   ;APPEND
OF%THW==:1B25                   ;THAWED
OF%AWT==:1B26                   ;ALWAYS WAIT
OF%PDT==:1B27                   ;PRESERVE DATES
OF%NWT==:1B28                   ;NEVER WAIT
OF%RTD==:1B29                   ;RESTRICTED
OF%PLN==:1B30                   ;SET TO DISABLE LINE NUMBER CHECKING FOR
                                ; NON-LINE NUMBER FILES
OF%DUD==:1B31                   ;DON'T UPDATE TO DISK BY DDMP
OF%OFL==:1B32                   ;ALLOW OPENING THE DEVICE EVEN IF OFFLINE
```

```
;PMAP BIT DEFINITIONS

PM%CNT==:1B0                    ;RH WORD CONTAINS A COUNT
PM%MVP==:1B1                    ;MOVE PAGE INSTEAD OF INDIRECT POINTER
                                ; (NOT IMPLEMENTED
PM%RD==:1B2                     ;READ
PM%WT==:1B3                     ;WRITE
PM%WR==:1B3                     ; (ANOTHER NAME FOR ABOVE)
PM%EX==:1B4                     ;EXECUTE (RESERVED FOR THE FUTURE)
PM%RWX==:7B4                    ;CONVENIENT ABBREV FOR RD+WT+EX
PM%PLD==:1B5                    ;PRELOAD PAGES BEING MAPPED
PM%IND==:1B6                    ;USE INDIRECT PTRS (RESERVED FOR THE FUTURE)
PM%TPU==:1B8                    ;TRAP TO USER
                                ; (NOT IMPLEMENTED -- OBSOLETE)
PM%CPY==:1B9                    ;COPY ON WRITE
PM%RPT==:777777B35              ;REPEAT COUNT


;PMCTL - PHYSICAL MEMORY CONTROL

.MCRCE==:0                      ;READ CACHE ENABLE
.MCSCE==:1                      ;SET CACHE ENABLE
 .MCCST==:0                     ;ARGLIST OFFSET FOR CACHE STATE
 MC%CEN==:1                     ;CACHE ENABLED
.MCRPS==:2                      ;READ PAGE STATUS
.MCSPS==:3                      ;SET PAGE STATUS
 .MCPPN==:0                     ;ARGLIST OFFSET FOR PHYSICAL PAGE NUMBER
 .MCPST==:1                     ;ARGLIST OFFSET FOR PAGE STATE
 .MCPSA==:0                     ;PAGE AVAILABLE
 .MCPSS==:1                     ;PAGE IN TRANSITION STATE
 .MCPSO==:2                     ;PAGE OFFLINE
 .MCPSE==:3                     ;PAGE OFFLINE DUE TO ERROR
.MCRME==:4                      ;READ MEMORY ERROR INFORMATION
 .PMMER==:1                     ;MOS MEMORY ERROR
 .PMMTP==:0                     ;ENTRY HEADER AND TYPE
 .PMMRG==:1                     ;ERROR REGISTER
 .PMMSY==:2                     ;SYNDROME
 .PMMBN==:3                     ;BLOCK NUMBER
 .PMMSB==:4                     ;SPARE BIT NUMBER
 .PMMEA==:5                     ;ERROR ADDRESS
 .PMMSN==:6                     ;START OF SERIAL NUMBERS
  .PMMNS==:4                    ;# OF SERIAL NUMBERS TO STORE


;PRARG - PROCESS ARGUMENTS

;FUNCTION CODE DEFINITIONS

.PRARD==:1                      ;READ ARGUMENT BLOCK
.PRAST==:2                      ;SET ARGUMENT BLOCK
```

```
;RCUSR AND RCDIR

; FLAGS SUPPLIED ON CALL

RC%PAR==:1B14                   ;PARTIAL RECOGNITION IS ALLOWED
RC%STP==:1B15                   ;STEP WILDCARD (RCDIR ONLY)
RC%AWL==:1B16                   ;ALLOW WILDCARDS (RCDIR ONLY)
RC%EMO==:1B17                   ;EXACT MATCH ONLY

; FLAGS RETURNED

RC%DIR==1B0                     ;FILES-ONLY DIRECTORY
RC%ANA==1B1                     ;ALPHANUMERIC ACCOUNTS ALLOWED
RC%RLM==1B2                     ;REPEAT LOGIN MESSAGE
RC%NOM==:1B3                    ;NO MATCH FOUND
RC%AMB==:1B4                    ;AMBIGUOUS
RC%NMD==:1B5                    ;NO MORE DIRS - RETURNED IF STP IS REQUESTED
RC%WLD==:1B6                    ;WILDCARD DIR WAS INPUT


;RDTTY AND TEXTI

RD%BRK==:1B0                    ;BREAK ON REGULAR BREAK SET
RD%TOP==:1B1                    ;BREAK ON TOPS10 BREAK SET
RD%PUN==:1B2                    ;BREAK ON PUNCTUATION
RD%BEL==:1B3                    ;BREAK ON END OF LINE
RD%CRF==:1B4                    ;SUPPRESS CR (RETURNS LF ONLY)
RD%RND==:1B5                    ;RETURN IF NOTHING TO DELETE
RD%JFN==:1B6                    ;JFNS GIVEN FOR SOURCE
RD%RIE==:1B7                    ;RETURN ON INPUT (BUFFER) EMPTY
RD%BBG==:1B8                    ;BEGINNING OF (DEST) BUFFER GIVEN
RD%RBF==:1B9                    ;^R BUFFER IS DISJOINT
RD%RAI==:1B10                   ;RAISE LOWERCASE INPUT
RD%SUI==:1B11                   ;SUPPRESS ^U INDICATION
RD%BTM==:1B12                   ;BREAK CHARACTER TERMINATED INPUT
RD%BFE==:1B13                   ;RETURNED BECAUSE BUFFER EMPTY
RD%BLR==:1B14                   ;BACKUP LIMIT REACHED

;TEXTI ARG BLOCK

.RDCWB==:0                      ;COUNT OF WORDS IN BLOCK
.RDFLG==:1                      ;FLAGS
.RDIOJ==:2                      ;IO JFNS
.RDDBP==:3                      ;DEST BYTE POINTER
.RDDBC==:4                      ;DEST BYTE COUNT
.RDBFP==:5                      ;TOP OF BUFFER POINTER
.RDRTY==:6                      ;RETYPE (^R) POINTER
.RDBRK==:7                      ;BREAK SET MASK POINTER
.RDBKL==:10                     ;BACKUP LIMIT POINTER
```

```
;RFSTS

RF%LNG==:1B0                    ;LONG FORM OF RFSTS CALL, ARG BLOCK IN 2
RF%PRH==:777777B35              ;PROCESS HANDLE

;RFSTS ARG BLOCK

.RFCNT==:0                      ;XWD COUNT OF WORDS RETURNED,
                                ; MAXIMUM WORDS TO RETURN
.RFPSW==:1                      ;PROCESS STATUS WORD
.RFPFL==:2                      ;PROCESS' PC FLAGS
.RFPPC==:3                      ;PROCESS' PC
.RFSFL==:4                      ;STATUS FLAGS FOR PROCESS:
  RF%EXO==1B0                   ;PROCESS IS EXECUTE-ONLY

;PROCESS STATUS WORD

RF%FRZ==:1B0                    ;PROCESS IS FROZEN
RF%STS==:377777B17              ;PROCESS STATUS CODE
   .RFRUN==:0                   ;RUNNABLE
   .RFIO==:1                    ;DISMISSED FOR I/O
   .RFHLT==:2                   ;HALTED
   .RFFPT==:3                   ;FORCED PROCESS TERMINATION
   .RFWAT==:4                   ;WAITING FOR INFERIOR PROCESS
   .RFSLP==:5                   ;SLEEP
   .RFTRP==:6                   ;JSYS TRAPPED
   .RFABK==:7                   ;ADDRESS BREAK FREEZE
RF%SIC==:777777B35              ;SOFTWARE INTERRUPT CHANNNEL


;RFTAD/SFTAD

.RSWRT==:0                      ;WRITE DATE WORD
.RSCRV==:1                      ;CREATION DATE WORD
.RSREF==:2                      ;REFERENCE DATE WORD
.RSCRE==:3                      ;INTERNAL SYSTEM WRITE DATE WORD


;RMAP

RM%RD==1B2                      ;READ ACCESS ALLOWED
RM%WR==:1B3                     ;WRITE ACCESS ALLOWED
RM%EX==:1B4                     ;EXECUTE ACCESS ALLOWED
RM%PEX==:1B5                    ;PAGE EXISTS
RM%CPY==:1B9                    ;COPY ON WRITE


;RPACS/SPACS BIT DEFINITIONS

PA%RD==:1B2                     ;READ ACCESS ALLOWED
PA%WT==:1B3                     ;WRITE ACCESS ALLOWED
PA%WR==:1B3                     ; (ANOTHER NAME FOR ABOVE)
PA%EX==:1B4                     ;EXECUTE ACCESS ALLOWED
                                ; (RESERVED FOR THE FUTURE)
PA%PEX==:1B5                    ;PAGE EXISTS
PA%IND==:1B6                    ;INDIRECT POINTER
PA%TPU==:1B8                    ;TRAP TO USER
                                ; (NOT IMPLEMENTED -- OBSOLETE)
PA%CPY==:1B9                    ;COPY ON WRITE
PA%PRV==:1B10                   ;PRIVATE
P1%RD==:1B20                    ;READ ACCESS ALLOWED IN 1ST POINTER
P1%WR==:1B21                    ;WRITE ACCESS ALLOWED IN 1ST POINTER
P1%WT==:1B21                    ; (ANOTHER NAME FOR ABOVE)
```

```
P1%EX==:1B22                    ;EXECUTE ACCESS ALLOWED IN 1ST POINTER
                                ; (RESERVED FOR THE FUTURE)
P1%PEX==:1B23                   ;PAGE EXISTS IN 1ST POINTER
P1%CPY==:1B27                   ;COPY-ON-WRITE IN 1ST POINTER


;RSCAN

.RSINI==:0                      ;MAKE RESCAN BUFFER AVAILABLE FOR INPUT
.RSCNT==:1                      ;COUNT CHARACTERS LEFT TO READ FROM RESCAN BUFFER


;RTIW

RT%DIM==:1B0                    ;DEFERRED TERMINAL INTERRUPT MASK GIVEN
RT%PRH==:377777B35              ;PROCESS HANDLE


;SCTTY

.SCRET==:0                      ;RETURN DESIGNATOR (CTTY) FOR FORK
.SCSET==:1                      ;SET SCTTY FOR FORK
.SCRST==:2                      ;CLEAR FORK CTTY (RESTORE JOB CTTY)

;SCVEC

.SVEAD==:0                      ;ENTRY ADDRESS
.SVINE==:1                      ;INITIAL ENTRY FOR SETUP
.SVGET==:2                      ;ENTRY ADDRESS FOR GET SHARE FILE ROUTINE
.SV40==:3                       ;ADDRESS TO GET LOCATION 40
.SVRPC==:4                      ;ADDRESS TO GET RETURN PC
.SVMAK==:5                      ;ENTRY FOR MAKE SHARE FILE ROUTINE
.SVCST==:6                      ;2 WORD BLOCK FOR CONTROL-C/START PROCESSING


;SDVEC

.SDEAD==:0                      ;ENTRY ADDRESS
.SDINE==:1                      ;INITIAL ENTRY
.SDVER==:2                      ;DMS VERSION
.SDDMS==:3                      ;ADDRESS TO STORE DMS JSYS
.SDRPC==:4                      ;ADDRESS TO STORE RETURN PC
```

```
;SETJB FUNCTION CODES

.SJDEN==:0                      ;SET DEFAULT MAGTAPE DENSITY
 .SJDDN==:0                     ;SYSTEM DEFAULT DENSITY
 .SJDN2==:1                     ;200 BPI
 .SJDN5==:2                     ;556 BPI
 .SJDN8==:3                     ;800 BPI
 .SJD16==:4                     ;1600 BPI
 .SJD62==:5                     ;6250 BPI
.SJPAR==:1                      ;SET DEFAULT MAGTAPE PARITY
 .SJPRO==:0                     ;ODD PARITY
 .SJPRE==:1                     ;EVEN PARITY
.SJDM==:2                       ;SET DEFAULT MAGTAPE DATA MODE
 .SJDDM==:0                     ;SYSTEM DEFAULT DATA MODE
 .SJDMC==:1                     ;CORE DUMP MODE
 .SJDM6==:2                     ;SIX BIT BYTE MODE (FOR 7-TRACK DRIVES)
 .SJDMA==:3                     ;ANSI ASCII MODE (7 BITS IN 8 BIT BYTE)
 .SJDM8==:4                     ;INDUSTRY COMPATIBLE MODE
 .SJDMH==:5                     ;HI-DENSITY MODE (9 EIGHT BIT
                                ; BYTES IN 2 WORDS)
.SJRS==:3                       ;SET DEFAULT MAGTAPE RECORD SIZE
.SJDFS==:4                      ;SET DEFERRED SPOOLING
 .SJSPI==:0                     ;IMMEDIATE MODE SPOOLING
 .SJSPD==:1                     ;DEFERRED MODE SPOOLING
.SJSRM==:5                      ;SET JOB SESSION REMARK


;SFUST

.SFAUT==:0                      ;SET AUTHOR STRING
.SFLWR==:1                      ;SET LAST WRITER STRING
```

```
;SMON FUNCTION CODES AND BIT DEFINITIONS (SYSTEM FLAGS)

.SFFAC==:0                          ;ALLOW FACT ENTRIES
.SFCDE==:1                          ;CHECKDISK FOUND ERRORS
.SFCDR==:2                          ;CHECKDISK RUNNING
.SFMST==:3                          ;MANUAL START IN PROGRESS
.SFRMT==:4                          ;REMOTE LOGINS ALLOWED
.SFPTY==:5                          ;PTY LOGINS ALLOWED
.SFCTY==:6                          ;CTY LOGIN ALLOWED
.SFOPR==:7                          ;OPERATOR IN ATTENDANCE
.SFLCL==:10                         ;LOCAL LOGINS ALLOWED
.SFBTE==:11                         ;BIT TABLE ERRORS FOUND ON STARTUP
.SFCRD==:12                         ;USER CAN CHANGE DIRECTORY CHARACTERISTICS
.SFNVT==:13             ;TOPS20AN   ;NVT LOGIN ALLOWED
.SFWCT==:14                         ;WHEEL LOGIN ON CTY ALLOWED
.SFWLC==:15                         ;WHEEL LOGIN ON LOCAL TERMINALS ALLOWED
.SFWRM==:16                         ;WHEEL LOGIN ON REMOTE TERMINALS ALLOWED
.SFWPT==:17                         ;WHEEL LOGIN ON PTY'S ALLOWED
.SFWNV==:20             ;TOPS20AN   ;WHEEL LOGIN ON NVT'S ALLOWED
.SFUSG==:21                         ;USAGE FILE IN USE
.SFFLO==:22                         ;FULL LATENCY OPTIMIZATION
                                    ;CAUTION: SETTING THIS REQUIRES THAT THE
                                    ;  SYSTEM BE AT REVISION LEVEL 10, AND
                                    ;  THAT RH20 BOARD M8555 BE AT REVISION LEVEL D.
                                    ;  OTHERWISE, THE FILE-SYSTEM MAY BE DAMAGED.

;BELOW ARE FUNCTION CODES WHICH DO NOT MAP DIRECTLY INTO BITS

.SFNTN==:44             ;TOPS20AN   ;NETWORK ON/OFF CONTROL
.SFNDU==:45             ;TOPS20AN   ;NET DOWN/UP REQUEST
.SFNHI==:46             ;TOPS20AN   ;NET HOST TABLE INITIALIZE
.SFTMZ==:47                         ;SET TIME ZONE THIS SYSTEM IS IN
.SFLHN==:50             ;TOPS20AN   ;SET LOCAL HOST NUMBER OF THIS NET SITE
.SFAVR==:51                         ;ACCOUNT VALIDATION ON/OFF
.SFSTS==:52                         ;ENABLE/DISABLE STATUS REPORTING
SF%FAC==:1B<.SFFAC>                 ;FACT ENTRIES ALLOWED
SF%CDE==:1B<.SFCDE>                 ;CHECKDISK FOUND ERRORS
SF%CDR==:1B<.SFCDR>                 ;CHECKDISK RUNNING
SF%MST==:1B<.SFMST>                 ;MANUAL START IN PROGRESS
SF%RMT==:1B<.SFRMT>                 ;REMOTE LOGINS ALLOWED
SF%PTY==:1B<.SFPTY>                 ;PTY LOGINS ALLOWED
SF%CTY==:1B<.SFCTY>                 ;CTY LOGIN ALLOWED
SF%OPR==:1B<.SFOPR>                 ;OPERATOR IN ATTENDANCE
SF%LCL==:1B<.SFLCL>                 ;LOCAL LOGINS ALLOWED
SF%BTE==:1B<.SFBTE>                 ;BIT TABLE ERRORS FOUND ON STARTUP
SF%CRD==:1B<.SFCRD>                 ;USER CAN CHANGE DIRECTORY CHARACTERISTICS
SF%NVT==:1B<.SFNVT>     ;TOPS20AN   ;NVT LOGINS ALLOWED
SF%USG==:1B<.SFUSG>                 ;USAGE FILE IN USE
SF%FLO==:1B<.SFFLO>                 ;FULL LATENCY OPTIMIZATION IN USE
                                    ;CAUTION: SETTING THIS REQUIRES THAT THE
                                    ;  SYSTEM BE AT REVISION LEVEL 10, AND
                                    ;  THAT RH20 BOARD M8555 BE AT REVISION LEVEL D.
                                    ;  OTHERWISE, THE FILE-SYSTEM MAY BE DAMAGED.


;SINM JSYS DEFINITIONS

SI%TMG==:1B0                        ;TRUNCATE MESSAGE
SI%EOM==:1B1                        ;END-OF-MESSAGE FOUND
```

```
;SNOOP JSYS DEFINITIONS

;SNOOP FUNCTION CODES

.SNPLC==:0                      ;LOCK CODE INTO MONITOR VIRT MEMORY
.SNPLS==:1                      ;LOCK DOWN THE SWAPPABLE MONITOR
.SNPDB==:2                      ;DEFINE A BREAK POINT
.SNPIB==:3                      ;INSERT THE BREAK POINTS
.SNPRB==:4                      ;REMOVE THE BREAK POINTS
.SNPUL==:5                      ;UNLOCK AND RELEASE ALL SNOOP RESOURCES
.SNPSY==:6                      ;LOOK UP A MONITOR SYMBOL
.SNPAD==:7                      ;LOOK UP ADDRESS IN SYMBOL TABLE

;SOUTM JSYS DEFINITIONS

SO%WMG==1B0                     ;WRITE END-OF-MESSAGE

;SPOOL JSYS FUNCTION CODES

.SPLDI==:0                      ;DEFINE AN INPUT SPOOLING DEVICE
.SPLSD==:1                      ;SET DIRECTORY OF SPOOLED DEVICE
.SPLRD==:2                      ;READ DIRECTORY OF SPOOLED DEVICE

;FLAGS IN SPOOL MESSAGE ON LOGOUT AND SPOOLED FILE CLOSE

SP%BAT==:1B0                    ;JOB IS A BATCH JOB
SP%DFS==:1B1                    ;SPOOLING IS DEFERRED
SP%ELO==:1B2                    ;JOB EXECUTED LGOUT JSYS ITSELF
SP%FLO==:1B3                    ;JOB FORCED TO LOG OUT BY TRAP IN TOP FK
SP%OLO==:1B4                    ;OTHER JOB AIMED LGOUT AT THIS ONE

;SPOOL ARGUMENT BLOCK

.SPLDV==:0                      ;DEVICE DESIGNATOR
.SPLNA==:1                      ;NAME STRING
.SPLDR==:1                      ;DIRECTORY NUMBER
.SPLGN==:2                      ;GENERATION NUMBER


;SSAVE

SS%NNP==777777B17               ;NEGATIVE NUMBER OF PAGES
SS%CPY==:1B18                   ;ALLOW COPY-ON-WRITE
SS%UCA==:1B19                   ;USE CURRENT ACCESS
SS%RD==:1B20                    ;ALLOW READ ACCESS
SS%WR==:1B21                    ;ALLOW WRITE ACCESS
SS%EXE==:1B22                   ;ALLOW EXECUTE ACCESS
SS%FPN==:777B35                 ;FIRST PAGE NUMBER


;STCMP

SC%LSS==:1B0                    ;T1 LESS THAN T2
SC%SUB==:1B1                    ;T1 SUBSTRING OF T2
SC%GTR==:1B2                    ;T1 GREATER THAN T2
```

```
;STDIR

ST%DIR==:1B0                    ;FILES ONLY DIRECTORY
ST%ANA==:1B1                    ;ALPHANUMERIC ACCOUNTS
ST%RLM==:1B2                    ;REPEAT LOGIN MESSAGE


;STIW

ST%DIM==:1B0                    ;SET DEFERRED INTERRUPT MASK
ST%PRH==:777777B35              ;PROCESS HANDLE

;SWTRP DEFINITIONS

.SWART==:0                      ;SET ARITHMETIC TRAP
.SWRAT==:1                      ;READ ARITHMETIC TRAP
.SWLUT==:2                      ;SET LUUO ADDRESS
.SWRLT==:3                      ;READ LUUO ADDRESS
        .ARPFL==:0              ;OFFSET IN TRAP BLOCK FOR PC FLAGS
        .AROPC==:1              ;OFFSET FOR OLD PC VALUE
        .ARNPC==:2              ;OFFSET FOR NEW PC WORD

;TBLUK

TL%NOM==:1B0                    ;NO MATCH
TL%AMB==:1B1                    ;AMBIGUOUS
TL%ABR==:1B2                    ;LEGAL ABBREVIATION
TL%EXM==:1B3                    ;EXACT MATCH


;TFORK

;FUNCTION CODES IN LH AC1

.TFSET==:0                      ;SET TRAPS AS SPEC'D BY BIT TABLE
.TFRAL==:1                      ;REMOVE ALL TRAPS SET BY THIS FORK
.TFRTP==:2                      ;REMOVE TRAPS SET BY THIS FORK
.TFSPS==:3                      ;SET JSYS TRAP PSI CHAN IN LH(2)
.TFRPS==:4                      ;READ JSYS TRAP PSI CHAN INTO LH(2)
.TFTST==:5                      ;TEST IF SELF MONITORED
.TFRES==:6                      ;REMOVE TRAPS FROM ALL INFERIORS, CLR PSI
.TFUUO==:7                      ;SET UUO TRAPS FOR FORK
.TFSJU==:8                      ;SET BOTH UUO AND JSYS TRAPS
.TFRUU==:9                      ;REMOVE UUO TRAPS

;TIMER DEFINITIONS

.TIMRT==:0                      ;SET TIME LIMIT
.TIMEL==:1                      ;SET ELAPSED TIME CLOCK
.TIMDT==:2                      ;SET DATE & TIME CLOCK
.TIMDD==:3                      ;DELETE AN EXPLICT DATE & TIME CLOCK
.TIMBF==:4                      ;DELETE ALL ENTIRES BEFORE D&T
.TIMAL==:5                      ;DELETE ALL (INCLUDES TIME LIMIT)
```

```
;TLINK

TL%CRO==:1B0                    ;CLEAR REMOTE TO OBJECT LINK
TL%COR==:1B1                    ;CLEAR OBJECT TO REMOTE LINK
TL%EOR==:1B2                    ;ESTABLIST OBJECT TO REMOTE LINK
TL%ERO==:1B3                    ;ESTABLISH REMOTE TO OBJECT LINK
TL%SAB==:1B4                    ;SET ACCEPT BIT FOR OBJECT
TL%ABS==:1B5                    ;ACCEPT BIT STATE
TL%STA==:1B6                    ;SET OR CLEAR ADVICE
TL%AAD==1B7                     ;ACCEPT ADVICE
TL%OBJ==:777777B35              ;OBJECT DESIGNATOR


;UTEST FUNCTION CODES

.UTSET==:0                      ;START TESTING
.UTCLR==:1                      ;STOP TESTING AND RETURN RESULTS

;UTEST ARGUMENT BLOCK

.UTADR==:0                      ;STARTING ADDRESS OF CODE
.UTLEN==:1                      ;LENGTH OF CODE
.UTMAP==:2                      ;START OF BIT MAP

;USAGE

.USENT==:0                      ;WRITE ENTRY
.USCLS==:1                      ;CLOSE OUT CURRENT FILE
.USCKP==:2                      ;PERFORM CHECKPOINT
.USLGI==:3                      ;LOGIN
.USLGO==:4                      ;LOGOUT
.USSEN==:5                      ;SESSION END
.USCKI==:6                      ;SET CHECKPOINT INTERVAL
.USENA==:7                      ;ENABLE ACCOUNT VALIDATION
.USCAS==:10                     ;CHANGE ACCOUNTING SHIFT NOW
.USSAS==:11                     ;SET AUTOMATIC ACCOUNTING SHIFT CHANGE TIMES
.USRAS==:12                     ;READ AUTOMATIC ACCOUNTING SHIFT CHANGE TIMES
                                ;TABLE ENTRY FORMAT FOR .USSAS/.USRAS:
 US%DOW==:177B6                 ;DAY-OF-WEEK BITS
 US%SSM==:777777                ;TIME IN SECONDS SINCE MIDNIGHT

;UTFRK

UT%TRP==:1B0                    ;ITRAP (OR DO ERJMP/ERCAL) TRAPPED JSYS
```

```
;SCHEDULER CONTROL FLAGS (JSYS NOT YET DEFINED)

SK%CYT==:1B18                   ;CYCLE TIME
SK%IOC==:1B19                   ;IO QUANTUM CHARGE
SK%HTF==:3B21                   ;BALSET HOLD TIME
SK%HQR==:1B22                   ;HIGH QUEUE RESERVE
SK%LQR==:1B23                   ;LOW QUEUE RESERVE
SK%BQE==:1B24                   ;BALSET QUEUE ON ENTRY
SK%BQR==:1B25                   ;BALSET QUEUE ON REQUEUE
SK%RQ1==:1B26                   ;REQUEUE TO QUEUE 1
SK%TTP==:1B27                   ;TTY PREFERENCE
SK%WCF==:1B28                   ;WAIT CREDIT PROPORTIONAL TO LOAD AV
```

```
;************************************************
;GENERAL FIELD AND VALUE DEFINITIONS
;USED BY MANY JSYSES
;************************************************


;GENERAL FORK HANDLES

.FHSLF==:400000                 ;SELF
.FHSUP==:<Z -1>                 ;SUPERIOR
.FHTOP==:<Z -2>                 ;TOP IN JOB
.FHSAI==:<Z -3>                 ;SELF AND INFERIORS
.FHINF==:<Z -4>                 ;INFERIORS
.FHJOB==:<Z -5>                 ;ALL IN JOB


;FIELDS OF JFN MODE WORD

TT%OSP==:1B0                    ;OUTPUT SUPPRESS
TT%MFF==:1B1                    ;MECHANICAL FORMFEED PRESENT
TT%TAB==:1B2                    ;MECHANICAL TAB PRESENT
TT%LCA==:1B3                    ;LOWER CASE CAPABILITIES PRESENT
TT%LEN==:177B10                 ;PAGE LENGTH
TT%WID==:177B17                 ;PAGE WIDTH
TT%WAK==:17B23                  ;WAKEUP FIELD
TT%WK0==:1B18                   ;WAKEUP CLASS 0 (UNUSED)
TT%IGN==:1B19                   ;IGNORE TT%WAK ON SFMOD
TT%WKF==:1B20                   ;WAKEUP ON FORMATING CONTROL CHARS
TT%WKN==:1B21                   ;WAKEUP ON NON-FORMATTING CONTROLS
TT%WKP==:1B22                   ;WAKEUP ON PUNCTUATION
TT%WKA==:1B23                   ;WAKEUP ON ALPHANUMERICS
TT%ECO==:1B24                   ;ECHOS ON
TT%ECM==:1B25                   ;ECHO MODE
TT%ALK==:1B26                   ;ALLOW LINKS
TT%AAD==:1B27                   ;ALLOW ADVICE (NOT IMPLEMENTED)
TT%DAM==:3B29                   ;DATA MODE
.TTBIN==:0                      ;BINARY
.TTASC==:1                      ;ASCII
.TTATO==:2                      ;ASCII AND TRANSLATE OUTPUT ONLY
.TTATE==:3                      ;ASCII AND TRANSLATE ECHOS ONLY
TT%UOC==:1B30                   ;UPPER CASE OUTPUT CONTROL
TT%LIC==:1B31                   ;LOWER CASE INPUT CONTROL
TT%DUM==:3B33                   ;DUPLEX MODE
.TTFDX==:0                      ;FULL DUPLEX
.TT0DX==:1                      ;NOT USED, RESERVED
.TTHDX==:2                      ;HALF DUPLEX (CHARACTER)
.TTLDX==:3                      ;LINE HALF DUPLEX
TT%PGM==:1B34                   ;PAGE MODE
TT%CAR==:1B35                   ;CARRIER STATE
```

```
;DIRECTORY PROTECTION DEFINITIONS (3 6-BIT FIELDS: OWNER, GROUP, WORLD)

DP%RD==:40                      ;READING DIRECTORY IS ALLOWED
DP%CN==:10                      ;CONNECT TO DIR, OR CHANGE PROT/ACCOUNT
DP%CF==:4                       ;CREATING FILES IN DIR IS ALLOWED

;FILE PROTECTION DEFINITIONS (3 6-BIT FIELDS: OWNER, GROUP, WORLD)

FP%DIR==:2                      ;DIRECTORY LISTING
FP%APP==:4                      ;APPEND
FP%EX==:10                      ;EXECUTE
FP%WR==:20                      ;WRITE
FP%RD==:40                      ;READ


;INPUT AND OUTPUT IDENTIFIERS

.PRIIN==:100                    ;PRIMARY INPUT
.PRIOU==:101                    ;PRIMARY OUTPUT
.NULIO==:377777                 ;NULL DESIGNATOR
.CTTRM==:777777                 ;JOB'S CONTROLLING TERMINAL
.DVDES==:600000                 ;UNIVERSAL DEVICE CODE
.TTDES==:400000                 ;UNIVERSAL TERMINAL CODE


;MAGTAPE DEVICE STATUS BITS

MT%ILW==:1B18                   ;ILLEGAL WRITE
MT%DVE==:1B19                   ;DEVICE ERROR
MT%DAE==:1B20                   ;DATA ERROR
MT%SER==:1B21                   ;SUPPRESS ERROR RECOVERY PROCEDURES
MT%EOF==:1B22                   ;EOF (FILE MARK)
MT%IRL==:1B23                   ;INCORRECT RECORD LENGTH
MT%BOT==:1B24                   ;BEGINNING OF TAPE
MT%EOT==:1B25                   ;END OF TAPE
MT%EVP==:1B26                   ;EVEN PARITY
MT%DEN==:3B28                   ;DENSITY (0 IS 'NORMAL')
.MTLOD==:1                      ;LOW DENSITY (200 BPI)
.MTMED==:2                      ;MEDIUM DENSITY (556 BPI)
.MTHID==:3                      ;HIGH DENSITY (800 BPI)
MT%CCT==:7B31                   ;CHARACTER COUNTER


;DEVICE DATA MODES

.DMASC==:1                      ;ASCII
.DMIMG==:10                     ;IMAGE
.DMIMB==:13                     ;IMAGE BINARY
.DMBIN==:14                     ;BINARY
```

```
;DEFINED PSI CHANNELS

RADIX 5+5

.ICAOV==:6                      ;ARITHMETIC OVERFLOW
.ICFOV==:7                      ;FLOATING OVERFLOW
.ICPOV==:9                      ;PDL OVERFLOW
.ICEOF==:10                     ;END OF FILE
.ICDAE==:11                     ;DATA ERROR
.ICQTA==:12                     ;QUOTA/DISK EXCEEDED
.ICTOD==:14                     ;TIME OF DAY (NOT IMPLEMENTED)
.ICILI==:15                     ;ILLEG INSTRUCTION
.ICIRD==:16                     ;ILLEGAL READ
.ICIWR==:17                     ;ILLEGAL WRITE
.ICIEX==:18                     ;ILLEGAL EXECUTE (NOT IMPLEMENTED)
.ICIFT==:19                     ;INFERIOR FORK TERMINATION
.ICMSE==:20                     ;MACHINE SIZE EXCEEDED
.ICTRU==:21                     ;TRAP TO USER (NOT IMPLEMENTED)
.ICNXP==:22                     ;NONEXISTENT PAGE REFERENCED
```

```
;TERMINAL TYPE NUMBERS

.TT33==:0                       ;MODEL 33
.TT35==:1                       ;MODEL 35
.TT37==:2                       ;MODEL 37
.TTEXE==:3                      ;EXECUPORT
.TTDEF==:^D8                    ;DEFAULT
.TTIDL==:^D9                    ;IDEAL
.TTV05==:^D10                   ;VT05
.TTV50==:^D11                   ;VT50
.TTL30==:^D12                   ;LA30
.TTG40==:^D13                   ;GT40
.TTL36==:^D14                   ;LA36
.TTV52==:^D15                   ;VT52

;DEFINED TERMINAL CODES

.TICBK==:0                      ;BREAK
.TICCA==:1                      ;^A
.TICCB==:2                      ;^B
.TICCC==:3                      ;^C
.TICCD==:4                      ;^D
.TICCE==:5                      ;^E
.TICCF==:6                      ;^F
.TICCG==:7                      ;^G
.TICCH==:8                      ;^H
.TICCI==:9                      ;^I
.TICCJ==:10                     ;^J
.TICCK==:11                     ;^K
.TICCL==:12                     ;^L
.TICCM==:13                     ;^M
.TICCN==:14                     ;^N
.TICCO==:15                     ;^O
.TICCP==:16                     ;^P
.TICCQ==:17                     ;^Q
.TICCR==:18                     ;^R
.TICCS==:19                     ;^S
.TICCT==:20                     ;^T
.TICCU==:21                     ;^U
.TICCV==:22                     ;^V
.TICCW==:23                     ;^W
.TICCX==:24                     ;^X
.TICCY==:25                     ;^Y
.TICCZ==:26                     ;^Z
.TICES==:27                     ;ESC
.TICRB==:28                     ;RUBOUT
.TICSP==:29                     ;SPACE
.TICRF==:30                     ;CARRIER OFF
.TICTI==:31                     ;TYPEIN
.TICTO==:32                     ;TYPEOUT
```

```
RADIX 8

;CAPABILITIES

SC%CTC==:1B0                    ;CONTROL-C
SC%GTB==:1B1                    ;GETAB
SC%MMN==:1B2                    ;MAP MONITOR
SC%LOG==:1B3                    ;LOGGING FUNCTIONS
SC%MPP==:1B4                    ;MAP PRIVILEGED PAGES
SC%SDV==:1B5                    ;SPECIAL DEVICES
SC%SCT==:1B6                    ;ASSIGN TTY AS CONTROLLING FOR FORK (SCTTY)

SC%SUP==:1B9                    ;SUPERIOR ACCESS

SC%FRZ==:1B17                   ;FREEZE ON TERMINATING CONDITIONS

SC%WHL==:1B18                   ;WHEEL
SC%OPR==:1B19                   ;OPERATOR
SC%CNF==:1B20                   ;CONFIDENTIAL INFORMATION ACCESS
SC%MNT==:1B21                   ;MAINTENANCE
SC%IPC==:1B22                   ;IPCF PRIVILEGES
SC%ENQ==:1B23                   ;ENQ/DEQ PRIVILEGES
SC%NWZ==:1B24         ;TOPS20AN ;NET WIZARD PRIVILEGES (ASNSQ, ETC.)
SC%NAS==:1B25         ;TOPS20AN ;NETWORK ABSOLUTE SOCKET PRIVILEGE


;OUTMODED NAMES FOR BITS IN DIRECTORY MODE WORD - USE CD%XXX
;EQUIVALENTS

MD%FO==:CD%DIR                  ;FILES ONLY DIRECTORY
MD%SA==:CD%ANA                  ;STRING ACCOUNT ALLOWED
MD%RLM==:CD%RLM                 ;REPEAT LOGIN MESSAGE
```

```
;FDB DEFINITIONS

FB%TMP==:1B0                    ;FILE IS TEMPORARY
FB%PRM==:1B1                    ;FILE IS PERMANENT
FB%NEX==:1B2                    ;FILE DOES NOT HAVE AN EXTENSION YET
FB%DEL==:1B3                    ;FILE IS DELETED
FB%NXF==:1B4                    ;FILE IS NONEXISTENT
FB%LNG==:1B5                    ;FILE IS A LONG FILE
FB%SHT==:1B6                    ;FILE HAS COMPRESSED PAGE TABLE
FB%DIR==:1B7                    ;FILE IS A DIRECTORY FILE
FB%NOD==:1B8                    ;FILE IS NOT TO BE DUMPED BY BACKUP SYSTEM
FB%BAT==:1B9                    ;FILE HAS AT LEAST ONE BAD PAGE IN IT
FB%SDR==:1B10                   ;THIS DIRECTORY HAS SUBDIRECTORIES
FB%FCF==:17B17                  ;FILE CLASS FIELD
 .FBNRM==:0                     ;NON-RMS
 .FBRMS==:1                     ;RMS FILES

.FBHDR==:0                      ;HEADER WORD
.FBCTL==:1                      ;FLAGS
.FBEXL==:2                      ;LINK TO FDB OF NEXT EXTENSION
.FBADR==:3                      ;DISK ADDRESS OF INDEX BLOCK
.FBPRT==:4                      ;PROTECTION OF THE FILE
.FBCRE==:5                      ;TIME AND DATE OF LAST WRITE
.FBUSE==:6                      ;LAST WRITER ,, AUTHOR (OBS)
.FBAUT==:6                      ;POINTER TO AUTHOR STRING
.FBGEN==:7                      ;GENERATION ,, DIR #
 FB%GEN==:777777B17             ;GENERATION NUMBER
.FBDRN==:7                      ;GENERATION ,, DIR #
 FB%DRN==:777777                ;DIR NUMBER
.FBACT==:10                     ;ACCOUNT
.FBBYV==:11                     ;RETENTION+BYTE SIZE+MODE ,, # OF PAGES
 FB%RET==:77B5                  ;RETENTION COUNT
 FB%BSZ==:77B11                 ;BYTE SIZE
 FB%MOD==:17B17                 ;LAST OPENF MODE
 FB%PGC==:777777                ;PAGE COUNT
.FBSIZ==:12                     ;EOF POINTER
.FBCRV==:13                     ;TIME AND DATE OF CREATION OF FILE
.FBWRT==:14                     ;TIME AND DATE OF LAST USER WRITE
.FBREF==:15                     ;TIME AND DATE OF LAST NON-WRITE ACCESS
.FBCNT==:16                     ;# OF WRITES ,, # OF REFERENCES
.FBBK0==:17                     ;BACKUP WORDS (5)
.FBBK1==:20
.FBBK2==:21
.FBBK3==:22
.FBBK4==:23
.FBUSW==:24                     ;USER SETTABLE WORD
.FBGNL==:25                     ;LINK TO NEXT GENERATION FILE
.FBNAM==:26                     ;POINTER TO NAME BLOCK
.FBEXT==:27                     ;POINTER TO EXTENSION BLOCK
.FBLWR==:30                     ;POINTER TO LAST WRITER STRING

.FBLN0==:30                     ;LENGTH OF VERSION 0 FDB
.FBLN1==:31                     ;LENGTH OF VERSION 1 FDB
.FBLEN==:31                     ;LENGTH OF THE FDB
```

```
;CARD READER DEFINITIONS

.CRILC==:"\"                        ;ILLEGAL CHARACTER CODE

;A WORD IS DISTINGUISHED FROM A BYTE POINTER BY THE VALUE 5 IN BITS 0-2
;USE THESE DEFINITIONS TO TEST FOR A NUMBER AS FOLLOWS:
;       LOAD AC,NMFLG,LOC
;       CAIE AC,NUMVAL

NMFLG==:7B2
NUMVAL==:5



;DEFINITIONS FOR COMMUNICATIONS PROTOCOLS

;DEFINE THE SUPPORTED PROTOCOL TYPES

.VN20F==:0                          ;RSX20F PROTOCOL
.VNMCB==:1                          ;MCB DECNET PROTOCOL
.VNDDC==:2                          ;DDCMP PROTOCOL
.VNMOP==:3                          ;MOP (DDCMP MAINTENANCE) MODE
.VNCNL==:4                          ;CONTROLLER LOOPBACK
.VNCBL==:5                          ;CABLE LOOPBACK

;DEFINE BITS USED WHEN RELOADING AN -11

RM%ROM==:1B0                        ;IF SET, ACTIVATE ROM
```

```
;*********************************************
;GENERAL FIELD AND VALUE DEFINITIONS
;USED BY TOPS20AN JSYS'S
;*********************************************

;STATES OF A CONNECTION IN ARPANET NCP
; RETURNED IN B0-B3 OF GDSTS ON A NET CONNECTION
; ALSO AVAILABLE IN A GETAB, BUT THAT'S NOT THE PREFERRED WAY
; TO READ THEM, IF YOU HAVE A JFN FOR THE CONNECTION.

.NSCZD==:01                     ;CLOSED
.NSPND==:02                     ;PENDING
.NSLSN==:03                     ;LISTENING
.NSRCR==:04                     ;REQUEST FOR CONNECTION RECEIVED
.NSCW1==:05                     ;CLOSE WAIT SUB ONE (NCP CLOSE)
.NSRCS==:06                     ;REQUEST FOR CONNECTION SENT
.NSOPN==:07                     ;OPENED
.NSCSW==:10                     ;CLOSE WAIT (NCP CLOSE)
.NSDTW==:11                     ;FINAL DATA WAIT
.NSRF1==:12                     ;RFNM WAIT SUB ONE (NORMAL NCP CLOSE)
.NSCZW==:13                     ;CLOSE WAIT (PROGRAM CLOSE)
.NSRF2==:14                     ;RFNM WAIT SUB TWO (UNEXPECTED NCP CLOSE)
.NSFRE==:16                     ;FREE
```

```
;ERROR CODE DEFINITIONS

        .ERBAS==:600000         ;BASE VALUE FOR ALL ERROR CODES

DEFINE .ERCOD <

.ERR (10,LGINX1,<Invalid account identifier>)
.ERR (11,LGINX2,<Directory is "files-only" and cannot be logged in to>)
.ERR (12,LGINX3,<Internal format of directory is incorrect>)
.ERR (13,LGINX4,<Invalid password>)
.ERR (14,LGINX5,<Job is already logged in>)
.ERR (20,CRJBX1,<Invalid parameter or function bit combination>)
.ERR (21,CRJBX2,<Illegal for created job to enter MINI-EXEC>)
.ERR (22,CRJBX3,<Reserved>)
.ERR (23,CRJBX4,<Terminal is not available>)
.ERR (24,CRJBX5,<Unknown name for LOGIN>)
.ERR (25,CRJBX6,<Insufficient system resources>)
.ERR (26,CRJBX7,<Reserved>)
.ERR (35,LOUTX1,<Illegal to specify job number when logging out own job>)
.ERR (36,LOUTX2,<Invalid job number>)
.ERR (45,CACTX1,<Invalid account identifier>)
.ERR (46,CACTX2,<Job is not logged in>)
.ERR (50,EFCTX1,<WHEEL or OPERATOR capability required>)
.ERR (51,EFCTX2,<Entry cannot be longer than 64 words>)
.ERR (52,EFCTX3,<Fatal error when accessing FACT file>)
.ERR (55,GJFX1,<Desired JFN invalid>)
.ERR (56,GJFX2,<Desired JFN not available>)
.ERR (57,GJFX3,<No JFN available>)
.ERR (60,GJFX4,<Invalid character in filename>)
.ERR (61,GJFX5,<Field cannot be longer than 39 characters>)
.ERR (62,GJFX6,<Device field not in a valid position>)
.ERR (63,GJFX7,<Directory field not in a valid position>)
.ERR (64,GJFX8,<Directory terminating delimiter is not preceded by a valid
     beginning delimiter>)
.ERR (65,GJFX9,<More than one name field is not allowed>)
.ERR (66,GJFX10,<Generation number is not numeric>)
.ERR (67,GJFX11,<More than one generation number field is not allowed>)
.ERR (70,GJFX12,<More than one account field is not allowed>)
.ERR (71,GJFX13,<More than one protection field is not allowed>)
.ERR (72,GJFX14,<Invalid protection>)
.ERR (73,GJFX15,<Invalid confirmation character>)
.ERR (74,GJFX16,<No such device>)
.ERR (75,GJFX17,<No such directory name>)
.ERR (76,GJFX18,<No such filename>)
.ERR (77,GJFX19,<No such file type>)
.ERR (100,GJFX20,<No such generation number>)
.ERR (101,GJFX21,<File was expunged>)
.ERR (102,GJFX22,<Insufficient system resources (Job Storage Block full)>)
.ERR (103,GJFX23,<Directory full>)
.ERR (104,GJFX24,<File not found>)
.ERR (107,GJFX27,<File already exists (new file required)>)
.ERR (110,GJFX28,<Device is not on line>)
.ERR (111,GJFX29,<Device is not available to this job>)
.ERR (112,GJFX30,<Account is not numeric>)
.ERR (113,GJFX31,<Invalid wildcard designator>)
.ERR (114,GJFX32,<No files match this specification>)
.ERR (115,GJFX33,<Filename was not specified>)
.ERR (116,GJFX34,<Invalid character "?" in file specification>)
.ERR (117,GJFX35,<Directory access privileges required>)
.ERR (120,OPNX1,<File is already open>)
.ERR (121,OPNX2,<File does not exist>)
.ERR (122,OPNX3,<Read access required>)
.ERR (123,OPNX4,<Write access required>)
.ERR (124,OPNX5,<Execute access required>)
```

```
.ERR (125,OPNX6,<Append access required>)
.ERR (126,OPNX7,<Device already assigned to another job>)
.ERR (127,OPNX8,<Device is not on line>)
.ERR (130,OPNX9,<Invalid simultaneous access>)
.ERR (131,OPNX10,<Entire file structure full>)
.ERR (133,OPNX12,<List access required>)
.ERR (134,OPNX13,<Invalid access requested>)
.ERR (135,OPNX14,<Invalid mode requested>)
.ERR (136,OPNX15,<Read/write access required>)
.ERR (137,OPNX16,<File has bad index block>)
.ERR (140,OPNX17,<No room in job for long file page table>)
.ERR (141,OPNX18,<Unit Record Devices are not available>)
.ERR (142,OPNX19,<IMP is not up>) ;TOPS20AN
.ERR (143,OPNX20,<Host is not up>) ;TOPS20AN
.ERR (144,OPNX21,<Connection refused>) ;TOPS20AN
.ERR (145,OPNX22,<Connection byte size does not match>) ;TOPS20AN
.ERR (150,DESX1,<Invalid source/destination designator>)
.ERR (151,DESX2,<Terminal is not available to this job>)
.ERR (152,DESX3,<JFN is not assigned>)
.ERR (153,DESX4,<Invalid use of terminal designator or string pointer>)
.ERR (154,DESX5,<File is not open>)
.ERR (155,DESX6,<Device is not a terminal>)
.ERR (156,DESX7,<JFN cannot refer to output wildcard designators>)
.ERR (157,DESX8,<File is not on disk>)
.ERR (160,CLSX1,<File is not open>)
.ERR (161,CLSX2,<File cannot be closed by this process>)
.ERR (165,RJFNX1,<File is not closed>)
.ERR (166,RJFNX2,<JFN is being used to accumulate filename>)
.ERR (167,RJFNX3,<JFN is not accessible by this process>)
.ERR (170,DELFX1,<Delete access required>)
.ERR (175,SFPTX1,<File is not open>)
.ERR (176,SFPTX2,<Illegal to reset pointer for this file>)
.ERR (177,SFPTX3,<Invalid byte number>)
.ERR (200,CNDIX1,<Invalid password>)
.ERR (202,CNDIX3,<Invalid directory number>)
.ERR (204,CNDIX5,<Job is not logged in>)
.ERR (210,SFBSX1,<Illegal to change byte size for this opening of file>)
.ERR (211,SFBSX2,<Invalid byte size>)
.ERR (215,IOX1,<File is not opened for reading>)
.ERR (216,IOX2,<File is not opened for writing>)
.ERR (217,IOX3,<File is not open for random access>)
.ERR (220,IOX4,<End of file reached>)
.ERR (221,IOX5,<Device or data error>)
.ERR (222,IOX6,<Illegal to write beyond absolute end of file>)
.ERR (240,PMAPX1,<Invalid access requested>)
.ERR (241,PMAPX2,<Invalid use of PMAP>)
.ERR (245,SPACX1,<Invalid access requested>)
.ERR (250,FRKHX1,<Invalid process handle>)
.ERR (251,FRKHX2,<Illegal to manipulate a superior process>)
.ERR (252,FRKHX3,<Invalid use of multiple process handle>)
.ERR (253,FRKHX4,<Process is running>)
.ERR (255,FRKHX6,<All relative process handles in use>)
.ERR (260,SPLFX1,<Process is not inferior or equal to self>)
.ERR (261,SPLFX2,<Process is not inferior to self>)
.ERR (262,SPLFX3,<New superior process is inferior to intended inferior>)
.ERR (267,GTABX1,<Invalid table number>)
.ERR (270,GTABX2,<Invalid table index>)
.ERR (271,GTABX3,<GETAB capability required>)
.ERR (273,RUNTX1,<Invalid process handle -3 or -4>)
.ERR (275,STADX1,<WHEEL or OPERATOR capability required>)
.ERR (276,STADX2,<Invalid date or time>)
.ERR (300,ASNDX1,<Device is not assignable>)
.ERR (301,ASNDX2,<Illegal to assign this device >)
.ERR (302,ASNDX3,<No such device>)
```

```
.ERR (320,ATACX1,<Invalid job number>)
.ERR (321,ATACX2,<Job already attached>)
.ERR (322,ATACX3,<Incorrect user number>)
.ERR (323,ATACX4,<Invalid password>)
.ERR (324,ATACX5,<This job has no controlling terminal>)
.ERR (332,STDVX1,<No such device>)
.ERR (335,DEVX1,<Invalid device designator>)
.ERR (336,DEVX2,<Device already assigned to another job>)
.ERR (337,DEVX3,<Device is not on line>)
.ERR (345,MNTX1,<Internal format of directory is incorrect>)
.ERR (346,MNTX2,<Device is not on line>)
.ERR (347,MNTX3,<Device is not mountable>)
.ERR (350,TERMX1,<Invalid terminal code>)
.ERR (351,TLNKX1,<Illegal to set remote to object before object to remote>)
.ERR (352,ATIX1,<Invalid software interrupt channel number>)
.ERR (353,ATIX2,<Control-C capability required>)
.ERR (356,TLNKX2,<Link was not received within 15 seconds>)
.ERR (357,TLNKX3,<Links full>)
.ERR (360,TTYX1,<Device is not a terminal>)
.ERR (361,RSCNX1,<Overflowed rescan buffer, input string truncated>)
.ERR (362,RSCNX2,<Invalid function code>)
.ERR (363,CFRKX3,<Insufficient system resources>)
.ERR (365,KFRKX1,<Illegal to kill top level process>)
.ERR (366,KFRKX2,<Illegal to kill self>)
.ERR (367,RFRKX1,<Processes are not frozen>)
.ERR (370,HFRKX1,<Illegal to halt self with HFORK>)
.ERR (371,GFRKX1,<Invalid process handle>)
.ERR (373,GETX1,<Invalid save file format>)
.ERR (374,GETX2,<System Special Pages Table full>)
.ERR (375,TFRKX1,<Undefined function code>)
.ERR (376,TFRKX2,<Unassigned fork handle or not immediate inferior>)
.ERR (377,SFRVX1,<Invalid position in entry vector>)
.ERR (407,NOUTX1,<Radix is not in range 2 to 36 >)
.ERR (410,NOUTX2,<Column overflow>)
.ERR (411,TFRKX3,<Fork(s) not frozen>)
.ERR (414,IFIXX1,<Radix is not in range 2 to 10>)
.ERR (415,IFIXX2,<First nonspace character is not a digit>)
.ERR (416,IFIXX3,<Overflow (number is greater than 2**35 )>)
.ERR (424,GFDBX1,<Invalid displacement>)
.ERR (425,GFDBX2,<Invalid number of words>)
.ERR (426,GFDBX3,<List access required>)
.ERR (430,CFDBX1,<Invalid displacement>)
.ERR (431,CFDBX2,<Illegal to change specified bits>)
.ERR (432,CFDBX3,<Write or owner access required>)
.ERR (433,CFDBX4,<Invalid value for specified bits>)
.ERR (440,DUMPX1,<Command list error>)
.ERR (441,DUMPX2,<JFN is not open in dump mode>)
.ERR (442,DUMPX3,<Address error (too big or crosses end of memory)>)
.ERR (443,DUMPX4,<Access error (cannot read or write data in memory)>)
.ERR (450,RNAMX1,<Files are not on same device>)
.ERR (451,RNAMX2,<Destination file expunged>)
.ERR (452,RNAMX3,<Write or owner access to destination file required>)
.ERR (453,RNAMX4,<Quota exceeded in destination of rename>)
.ERR (454,BKJFX1,<Illegal to back up terminal pointer twice>)
.ERR (460,TIMEX1,<Time cannot be greater than 24 hours>)
.ERR (461,ZONEX1,<Time zone out of range>)
.ERR (462,ODTNX1,<Time zone must be USA or Greenwich>)
.ERR (464,DILFX1,<Invalid date format>)
.ERR (465,TILFX1,<Invalid time format>)
.ERR (466,DATEX1,<Year out of range>)
.ERR (467,DATEX2,<Month is not less than 12>)
.ERR (470,DATEX3,<Day of month too large>)
.ERR (471,DATEX4,<Day of week is not less than 7>)
.ERR (472,DATEX5,<Date out of range>)
```

```
.ERR (473,DATEX6,<System date and time are not set>)
.ERR (516,SMONX1,<WHEEL or OPERATOR capability required>)
.ERR (530,SACTX1,<File is not on multiple-directory device>)
.ERR (531,SACTX2,<Insufficient system resources (Job Storage Block full)>)
.ERR (532,SACTX3,<Directory requires numeric account>)
.ERR (533,SACTX4,<Write or owner access required>)
.ERR (540,GACTX1,<File is not on multiple-directory device>)
.ERR (541,GACTX2,<File expunged>)
.ERR (544,FFUFX1,<File is not open>)
.ERR (545,FFUFX2,<File is not on multiple-directory device>)
.ERR (546,FFUFX3,<No used page found>)
.ERR (555,DSMX1,<File(s) not closed>)
.ERR (560,RDDIX1,<Illegal to read directory for this device>)
.ERR (570,SIRX1,<Table address is not greater than 20>)
.ERR (600,SSAVX1,<Illegal to save files on this device>)
.ERR (601,SSAVX2,<Page count is not less than or equal to 1000>)
.ERR (610,SEVEX1,<Entry vector is not less than 1000>)
.ERR (614,WHELX1,<WHEEL or OPERATOR capability required>)
.ERR (615,CAPX1,<WHEEL or OPERATOR capability required>)
.ERR (617,PEEKX2,<Read access failure on monitor page>)
.ERR (620,CRDIX1,<WHEEL or OPERATOR capability required>)
.ERR (621,CRDIX2,<Illegal to change number of old directory>)
.ERR (622,CRDIX3,<Insufficient system resources (Job Storage Block full)>)
.ERR (623,CRDIX4,<Superior directory full>)
.ERR (624,CRDIX5,<Directory name not given>)
.ERR (626,CRDIX7,<File(s) open in directory>)
.ERR (640,GTDIX1,<WHEEL or OPERATOR capability required>)
.ERR (641,GTDIX2,<Invalid directory number>)
.ERR (650,FLINX1,<First character is not blank or numeric>)
.ERR (651,FLINX2,<Number too small>)
.ERR (652,FLINX3,<Number too large>)
.ERR (653,FLINX4,<Invalid format>)
.ERR (660,FLOTX1,<Column overflow in field 1 or 2>)
.ERR (661,FLOTX2,<Column overflow in field 3>)
.ERR (662,FLOTX3,<Invalid format specified>)
.ERR (670,HPTX1,<Undefined clock number>)
.ERR (700,FDFRX1,<Not a multiple-directory device>)
.ERR (701,FDFRX2,<Invalid directory number>)
.ERR (710,ATNX1,<Invalid receive JFN>) ;TOPS20AN
.ERR (711,ATNX2,<Receive JFN not opened for read>) ;TOPS20AN
.ERR (712,ATNX3,<Receive JFN not open>) ;TOPS20AN
.ERR (713,ATNX4,<Receive JFN is not a NET connection>) ;TOPS20AN
.ERR (714,ATNX5,<Receive JFN has been used>) ;TOPS20AN
.ERR (715,ATNX6,<Receive connection refused>) ;TOPS20AN
.ERR (716,ATNX7,<Invalid send JFN>) ;TOPS20AN
.ERR (717,ATNX8,<Send JFN not opened for write>) ;TOPS20AN
.ERR (720,ATNX9,<Send JFN not open>) ;TOPS20AN
.ERR (721,ATNX10,<Send JFN is not a NET connection>) ;TOPS20AN
.ERR (722,ATNX11,<Send JFN has been used>) ;TOPS20AN
.ERR (723,ATNX12,<Send connection refused>) ;TOPS20AN
.ERR (724,ATNX13,<Insufficient system resources (No NVT's)>) ;TOPS20AN
.ERR (727,CVHST1,<No string for that Host number>) ;TOPS20AN
.ERR (730,CVSKX1,<Invalid JFN>) ;TOPS20AN
.ERR (731,CVSKX2,<Local socket invalid in this context>) ;TOPS20AN
.ERR (732,SNDIX1,<Invalid message size>) ;TOPS20AN
.ERR (733,SNDIX2,<Insufficient system resources (No buffers available)>) ;TOPS20AN
.ERR (734,SNDIX3,<Illegal to specify NCP links 0 - 72>) ;TOPS20AN
.ERR (735,SNDIX4,<Invalid header value for this queue>) ;TOPS20AN
.ERR (736,SNDIX5,<IMP down>) ;TOPS20AN
.ERR (737,NTWZX1,<NET WIZARD capability  required>) ;TOPS20AN
.ERR (740,ASNSX1,<Insufficient system resources (All special queues in use)>) ;TOPS20AN
.ERR (741,ASNSX2,<Link(s) assigned to another special queue>) ;TOPS20AN
.ERR (742,SQX1,<Special network queue handle out of range>) ;TOPS20AN
.ERR (743,SQX2,<Special network queue not assigned>) ;TOPS20AN
```

```
.ERR (750,RNAMX5,<Destination file is not closed>)
.ERR (751,RNAMX6,<Destination file has bad page table>)
.ERR (752,RNAMX7,<Source file expunged>)
.ERR (753,RNAMX8,<Write or owner access to source file required>)
.ERR (754,RNAMX9,<Source file is nonexistent>)
.ERR (755,RNMX10,<Source file is not closed>)
.ERR (756,RNMX11,<Source file has bad page table>)
.ERR (757,RNMX12,<Illegal to rename to self>)
.ERR (760,GJFX36,<Internal format of directory is incorrect>)
.ERR (770,ILINS1,<Undefined operation code>)
.ERR (771,ILINS2,<Undefined JSYS>)
.ERR (772,ILINS3,<UUO simulation facility not available>)
.ERR (1000,CRLNX1,<Logical name is not defined>)
.ERR (1001,INLNX1,<Index is beyond end of logical name table>)
.ERR (1002,LNSTX1,<No such logical name>)
.ERR (1003,MLKBX1,<Lock facility already in use>)
.ERR (1004,MLKBX2,<Too many pages to be locked>)
.ERR (1005,MLKBX3,<Page is not available>)
.ERR (1006,MLKBX4,<Illegal to remove previous contents of user map>)
.ERR (1007,VBCX1,<Display data area not locked in core>)
.ERR (1010,RDTX1,<Invalid string pointer>)
.ERR (1011,GFKSX1,<Area too small to hold process structure>)
.ERR (1013,GTJIX1,<Invalid index>)
.ERR (1014,GTJIX2,<Invalid terminal line number>)
.ERR (1015,GTJIX3,<Invalid job number>)
.ERR (1016,IPCFX1,<Length of packet descriptor block cannot be less than 4>)
.ERR (1017,IPCFX2,<No message for this PID>)
.ERR (1020,IPCFX3,<Data too long for user's buffer>)
.ERR (1021,IPCFX4,<Receiver's PID invalid>)
.ERR (1022,IPCFX5,<Receiver's PID disabled>)
.ERR (1023,IPCFX6,<Send quota exceeded>)
.ERR (1024,IPCFX7,<Receiver quota exceeded>)
.ERR (1025,IPCFX8,<IPCF free space exhausted>)
.ERR (1026,IPCFX9,<Sender's PID invalid>)
.ERR (1027,IPCF10,<WHEEL capability required>)
.ERR (1030,IPCF11,<WHEEL or IPCF capability required>)
.ERR (1031,IPCF12,<No free PID's available>)
.ERR (1032,IPCF13,<PID quota exceeded>)
.ERR (1033,IPCF14,<No PID's available to this job>)
.ERR (1034,IPCF15,<No PID's available to this process>)
.ERR (1035,IPCF16,<Receive and message data modes do not match>)
.ERR (1036,IPCF17,<Argument block too small>)
.ERR (1037,IPCF18,<Invalid MUTIL JSYS function>)
.ERR (1040,IPCF19,<No PID for [SYSTEM] INFO>)
.ERR (1041,IPCF20,<Invalid process handle>)
.ERR (1042,IPCF21,<Invalid job number>)
.ERR (1043,IPCF22,<Invalid software interrupt channel number>)
.ERR (1044,IPCF23,<[SYSTEM] INFO already exists>)
.ERR (1045,IPCF24,<Invalid message size>)
.ERR (1046,IPCF25,<PID does not belong to this job>)
.ERR (1047,IPCF26,<PID does not belong to this process>)
.ERR (1050,IPCF27,<PID is not defined>)
.ERR (1051,IPCF28,<PID not accessible by this process>)
.ERR (1052,IPCF29,<PID already being used by another process>)
.ERR (1053,IPCF30,<Job is not logged in>)
.ERR (1054,GNJFX1,<No more files in this specification>)
.ERR (1055,ENQX1,<Invalid function>)
.ERR (1056,ENQX2,<Level number too small>)
.ERR (1057,ENQX3,<Request and lock level numbers do not match>)
.ERR (1060,ENQX4,<Number of pool and lock resources do not match>)
.ERR (1061,ENQX5,<Lock already requested>)
.ERR (1062,ENQX6,<Requested locks are not all locked>)
.ERR (1063,ENQX7,<No ENQ on this lock>)
.ERR (1064,ENQX8,<Invalid access change requested>)
```

```
.ERR (1065,ENQX9,<Invalid number of blocks specified>)
.ERR (1066,ENQX10,<Invalid argument block length>)
.ERR (1067,ENQX11,<Invalid software interrupt channel number>)
.ERR (1070,ENQX12,<Invalid number of resources requested>)
.ERR (1071,ENQX13,<Indirect or indexed byte pointer not allowed>)
.ERR (1072,ENQX14,<Invalid byte size>)
.ERR (1073,ENQX15,<ENQ/DEQ capability required>)
.ERR (1074,ENQX16,<WHEEL or OPERATOR capability required>)
.ERR (1075,ENQX17,<Invalid JFN>)
.ERR (1076,ENQX18,<Quota exceeded>)
.ERR (1077,ENQX19,<String too long>)
.ERR (1100,ENQX20,<Locked JFN cannot be closed>)
.ERR (1101,ENQX21,<Job is not logged in>)
.ERR (1102,IPCF31,<Invalid page number>)
.ERR (1103,IPCF32,<Page is not private>)
.ERR (1104,PMAPX3,<Illegal to move shared page into file>)
.ERR (1105,PMAPX4,<Illegal to move file page into process>)
.ERR (1106,PMAPX5,<Illegal to move special page into file>)
.ERR (1107,PMAPX6,<Disk quota exceeded>)
.ERR (1110,SNOPX1,<WHEEL or OPERATOR capability required>)
.ERR (1111,SNOPX2,<Invalid function>)
.ERR (1112,SNOPX3,<.SNPLC function must be first>)
.ERR (1113,SNOPX4,<Only one .SNPLC function allowed>)
.ERR (1114,SNOPX5,<Invalid page number>)
.ERR (1115,SNOPX6,<Invalid number of pages to lock>)
.ERR (1116,SNOPX7,<Illegal to define breakpoints after inserting them>)
.ERR (1117,SNOPX8,<Breakpoint is not set on instruction>)
.ERR (1120,SNOPX9,<No more breakpoints allowed>)
.ERR (1121,SNOP10,<Breakpoints already inserted>)
.ERR (1122,SNOP11,<Breakpoints not inserted>)
.ERR (1123,SNOP12,<Invalid format for program name symbol>)
.ERR (1124,SNOP13,<No such program name symbol>)
.ERR (1125,SNOP14,<No such symbol>)
.ERR (1126,SNOP15,<Not enough free pages for snooping>)
.ERR (1127,SNOP16,<Multiply defined symbol>)
.ERR (1130,IPCF33,<Invalid index into system PID table>)
.ERR (1131,SNOP17,<Breakpoint already defined>)
.ERR (1132,OPNX23,<Disk quota exceeded>)
.ERR (1133,GJFX37,<Input deleted>)
.ERR (1134,CRLNX2,<WHEEL or OPERATOR capability required>)
.ERR (1135,INLNX2,<Invalid function>)
.ERR (1136,LNSTX2,<Invalid function>)
.ERR (1137,ALCX1,<Invalid function>)
.ERR (1140,ALCX2,<WHEEL or OPERATOR capability required>)
.ERR (1141,ALCX3,<Device is not assignable>)
.ERR (1142,ALCX4,<Invalid job number>)
.ERR (1143,ALCX5,<Device already assigned to another job>)
.ERR (1144,SPLX1,<Invalid function>)
.ERR (1145,SPLX2,<Argument block too small>)
.ERR (1146,SPLX3,<Invalid device designator>)
.ERR (1147,SPLX4,<WHEEL or OPERATOR capability required>)
.ERR (1150,SPLX5,<Illegal to specify 0 as generation number for first file>)
.ERR (1151,CLSX3,<File still mapped>)
.ERR (1152,CRLNX3,<Invalid function>)
.ERR (1153,ALCX6,<Device assigned to user job, but will be given to allocator
     when released>)
.ERR (1154,CKAX1,<Argument block too smal>)
.ERR (1155,CKAX2,<Invalid directory number>)
.ERR (1156,CKAX3,<Invalid access code>)
.ERR (1157,TIMX1,<Invalid function>)
.ERR (1160,TIMX2,<Invalid process handle>)
.ERR (1161,TIMX3,<Time limit already set>)
.ERR (1162,TIMX4,<Illegal to clear time limit>)
.ERR (1163,SNOP18,<Data page is not private or copy-on-write>)
.ERR (1164,GJFX38,<File not found because output-only device was specified>)
```

```
.ERR (1165,GJFX39,<Logical name loop detected>)
.ERR (1166,CRDIX8,<Invalid directory number>)
.ERR (1167,CRDIX9,<Internal format of directory is incorrect>)
.ERR (1170,CRDI10,<Maximum directory number exceeded;  index table needs expanding>)
.ERR (1171,DELDX1,<WHEEL or OPERATOR capability required>)
.ERR (1172,DELDX2,<Invalid directory number>)
.ERR (1173,GACTX3,<Internal format of directory is incorrect>)
.ERR (1174,DIAGX1,<Invalid function>)
.ERR (1175,DIAGX2,<Device is not assigned>)
.ERR (1176,DIAGX3,<Argument block too small>)
.ERR (1177,DIAGX4,<Invalid device type>)
.ERR (1200,DIAGX5,<WHEEL, OPERATOR, or MAINTENANCE capability required>)
.ERR (1201,DIAGX6,<Invalid channel command list>)
.ERR (1202,DIAGX7,<Illegal to do I/O across page boundary>)
.ERR (1203,DIAGX8,<No such device>)
.ERR (1204,DIAGX9,<Unit does not exist>)
.ERR (1205,DIAG10,<Subunit does not exist>)
.ERR (1206,SYEX1,<Unreasonable SYSERR block size>)
.ERR (1207,SYEX2,<No buffer space available for SYSERR>)
.ERR (1210,MTOX1,<Invalid function>)
.ERR (1211,IOX7,<Insufficient system resources (Job Storage Block full)>)
.ERR (1212,IOX8,<Monitor internal error>)
.ERR (1213,MTOX5,<Invalid hardware data mode for magnetic tape>)
.ERR (1214,DUMPX5,<No-wait dump mode not supported for this device>)
.ERR (1215,DUMPX6,<Dump mode not supported for this device>)
.ERR (1216,IOX9,<Function legal for sequential write only>)
.ERR (1217,CLSX4,<Device still active>)
.ERR (1220,MTOX2,<Record size was not set before I/O was done>)
.ERR (1221,MTOX3,<Function not legal in dump mode>)
.ERR (1222,MTOX4,<Invalid record size>)
.ERR (1223,MTOX6,<Invalid magnetic tape density>)
.ERR (1224,OPNX25,<Device is write locked>)
.ERR (1225,GJFX40,<Undefined attribute in file specification>)
.ERR (1226,MTOX7,<WHEEL or OPERATOR capability required>)
.ERR (1227,LOUTX3,<WHEEL or OPERATOR capability required>)
.ERR (1230,LOUTX4,<LOG capability required>)
.ERR (1231,CAPX2,<WHEEL, OPERATOR, or MAINTENANCE capability required>)
.ERR (1232,SSAVX3,<Insufficient system resources (Job Storage Block full)>)
.ERR (1233,SSAVX4,<Directory area of EXE file is more than one page>)
.ERR (1234,TDELX1,<Table is empty>)
.ERR (1235,TADDX1,<Table is full>)
.ERR (1236,TADDX2,<Entry is already in table>)
.ERR (1237,TLUKX1,<Internal format of table is incorrect>)
.ERR (1240,IOX10,<Record is longer than user requested>)
.ERR (1241,CNDIX2,<WHEEL or OPERATOR capability required>)
.ERR (1242,CNDIX4,<Invalid job number>)
.ERR (1243,CNDIX6,<Job is not logged in>)
.ERR (1244,SJBX1,<Invalid function>)
.ERR (1245,SJBX2,<Invalid magnetic tape density>)
.ERR (1246,SJBX3,<Invalid magnetic tape data mode>)
.ERR (1247,TMONX1,<Invalid TMON function>)
.ERR (1250,SMONX2,<Invalid SMON function>)
.ERR (1251,SJBX4,<Invalid job number>)
.ERR (1252,SJBX5,<Job is not logged in>)
.ERR (1253,SJBX6,<WHEEL or OPERATOR capability required>)
.ERR (1254,GTJIX4,<No such job>)
.ERR (1255,ILINS4,<UUO simulation is disabled>)
.ERR (1256,ILINS5,<RMS facility is not available>)
.ERR (1257,COMNX1,<Invalid COMND function code>)
.ERR (1260,COMNX2,<Field too long for internal buffer>)
.ERR (1261,COMNX3,<Command too long for internal buffer>)
.ERR (1262,COMNX4,<Invalid character in input>)
.ERR (1263,PRAX1,<Invalid PRARG function code>)
.ERR (1264,PRAX2,<No room in monitor data base for argument block>)
```

```
.ERR (1265,COMNX5,<Invalid string pointer argument>)
.ERR (1266,COMNX6,<Problem in indirect file>)
.ERR (1267,COMNX7,<Error in command>)
.ERR (1270,PRAX3,<PRARG argument block too large>)
.ERR (1271,CKAX4,<File is not on disk>)
.ERR (1272,GACCX1,<Invalid job number>)
.ERR (1273,GACCX2,<No such job>)
.ERR (1274,MTOX8,<Argument block too long>)
.ERR (1275,DBRKX1,<No interrupts in progress>)
.ERR (1276,SJPRX1,<Job is not logged in>)
.ERR (1277,GJFX41,<File name must not exceed 6 characters>)
.ERR (1300,GJFX42,<File type must not exceed 3 characters>)
.ERR (1301,GACCX3,<Confidential Information Access capability required>)
.ERR (1302,TIMEX2,<Downtime cannot be more than 7 days in the future>)
.ERR (1303,DELFX2,<File cannot be expunged because it is currently open>)
.ERR (1304,DELFX3,<System scratch area depleted; file not deleted>)
.ERR (1305,DELFX4,<Directory symbol table could not be rebuilt>)
.ERR (1306,DELFX5,<Directory symbol table needs rebuilding>)
.ERR (1307,DELFX6,<Internal format of directory is incorrect>)
.ERR (1310,DELFX7,<FDB formatted incorrectly; file not deleted>)
.ERR (1311,DELFX8,<FDB not found; file not deleted>)
.ERR (1312,FRKHX7,<Process page cannot exceed 777>)
.ERR (1313,DIRX1,<Invalid directory number>)
.ERR (1314,DIRX2,<Insufficient system resources>)
.ERR (1315,DIRX3,<Internal format of directory is incorrect>)
.ERR (1316,UFPGX1,<File is not open for write>)
.ERR (1317,LNGFX1,<Page table does not exist and file not open for write>)
.ERR (1320,IPCF34,<Cannot receive into an existing page>)
.ERR (1321,COMNX8,<Number base out of range 2-10>)
.ERR (1322,MTOX9,<Output still pending>)
.ERR (1323,MTOX10,<VFU or RAM file cannot be OPENed>)
.ERR (1324,MTOX11,<Data too large for buffers>)
.ERR (1325,MTOX12,<Input error or not all data read>)
.ERR (1326,MTOX13,<Argument block too small>)
.ERR (1327,MTOX14,<Invalid software interrupt channel number>)
.ERR (1330,SAVX1,<Illegal to save files on this device>)
.ERR (1331,MTOX15,<Device does not have Direct Access (programmable) VFU>)
.ERR (1332,MTOX16,<VFU or Translation Ram file must be on disk>)
.ERR (1333,LPINX1,<Invalid unit number>)
.ERR (1334,LPINX2,<WHEEL or OPERATOR capability required>)
.ERR (1335,LPINX3,<Illegal to load RAM or VFU while device is OPEN>)
.ERR (1336,MTOX17,<Device is not on line>)
.ERR (1337,LGINX6,<No more job slots available for logging-in>)
.ERR (1340,DESX9,<Invalid operation for this device>)
.ERR (1341,ACESX1,<Argument block too small>)
.ERR (1342,ACESX2,<Insufficient system resources>)
.ERR (1343,DSKOX1,<Channel number too large>)
.ERR (1344,DSKOX2,<Unit number too large>)
.ERR (1345,MSTRX1,<Invalid function>)
.ERR (1346,MSTRX2,<WHEEL or OPERATOR capability required>)
.ERR (1347,MSTRX3,<Argument block too small>)
.ERR (1350,MSTRX4,<Insufficient system resources>)
.ERR (1351,MSTRX5,<Drive is not on-line>)
.ERR (1352,MSTRX6,<Home blocks are bad>)
.ERR (1353,MSTRX7,<Invalid structure name>)
.ERR (1354,MSTRX8,<Could not get OFN for ROOT-DIRECTORY>)
.ERR (1355,MSTRX9,<Could not MAP ROOT-DIRECTORY>)
.ERR (1356,MSTX10,<ROOT-DIRECTORY bad>)
.ERR (1357,MSTX11,<Could not initialize Index Table>)
.ERR (1360,MSTX12,<Could not OPEN Bit Table File>)
.ERR (1361,MSTX13,<Backup copy of ROOT-DIRECTORY is bad>)
.ERR (1362,MSTX14,<Invalid channel number>)
.ERR (1363,MSTX15,<Invalid unit number>)
.ERR (1364,MSTX16,<Invalid controller number>)
```

```
.ERR (1365,DSKX01,<Invalid structure number>)
.ERR (1366,DSKX02,<Bit table is being initialized>)
.ERR (1367,DSKX03,<Bit table has not been initialized>)
.ERR (1370,DSKX04,<Bit table being initialized by another job>)
.ERR (1371,GFUSX1,<Invalid function>)
.ERR (1372,GFUSX2,<Insufficient system resources>)
.ERR (1373,SFUSX1,<Invalid function>)
.ERR (1374,SFUSX2,<Insufficient system resources>)
.ERR (1375,SFUSX3,<No such user name>)
.ERR (1376,RCDIX1,<Insufficient system resources>)
.ERR (1377,RCDIX2,<Invalid directory specification>)
.ERR (1400,RCDIX3,<Invalid structure name>)
.ERR (1401,RCDIX4,<Monitor internal error>)
.ERR (1402,RCUSX1,<Insufficient system resources>)
.ERR (1403,TDELX2,<Invalid table entry location>)
.ERR (1404,TIMX5,<Invalid software interrupt channel number>)
.ERR (1405,LSTRX1,<Process has not encountered any errors>)
.ERR (1406,SWJFX1,<Illegal to swap same JFN>)
.ERR (1407,MTOX18,<Invalid software interrupt channel number>)
.ERR (1410,OPNX26,<Illegal to open a string pointer>)
.ERR (1411,DELFX9,<File is not a directory file>)
.ERR (1412,CRDIX6,<Directory file is mapped>)
.ERR (1413,COMNX9,<End of input file reached>)
.ERR (1414,STYPX1,<Invalid terminal type>)
.ERR (1415,PMAPX7,<Illegal to map file on dismounted structure>)
.ERR (1416,DSKOX3,<Invalid structure number>)
.ERR (1417,DESX10,<Structure is dismounted>)
.ERR (1420,DSKOX4,<Invalid address type specified>)
.ERR (1421,MSTX17,<All units in a structure must be of the same type>)
.ERR (1422,MSTX18,<No more units in system>)
.ERR (1423,MSTX19,<Unit is already part of a mounted structure>)
.ERR (1424,MSTX20,<Data error reading HOME blocks>)
.ERR (1425,MSTX21,<Structure is not mounted>)
.ERR (1426,MSTX22,<Illegal to change specified bits>)
.ERR (1427,CRDI11,<Invalid terminating bracket on directory>)
.ERR (1430,MSTX23,<Could not write HOME blocks>)
.ERR (1431,ACESX3,<Password is required>)
.ERR (1432,ACESX4,<Function not allowed for another job>)
.ERR (1433,ACESX5,<No function specified for ACCES>)
.ERR (1434,STRX05,<No such user name>)
.ERR (1435,ACESX6,<Directory is not accessed>)
.ERR (1436,STRX01,<Structure is not mounted>)
.ERR (1437,STRX02,<Insufficient system resources>)
.ERR (1440,IOX11,<Quota exceeded or disk full>)
.ERR (1441,IOX12,<Insufficient system resources (Swapping space full)>)
.ERR (1442,STRX03,<No such directory name>)
.ERR (1443,STRX04,<Ambiguous directory specification>)
.ERR (1444,PPNX1,<Invalid PPN>)
.ERR (1445,PPNX2,<Structure is not mounted>)
.ERR (1446,PPNX3,<Insufficient system resources>)
.ERR (1447,PPNX4,<Invalid directory number>)
.ERR (1450,SPLX6,<No directory to write spooled files into>)
.ERR (1451,CRDI12,<Structure is not mounted>)
.ERR (1452,GFUSX3,<File expunged>)
.ERR (1453,GFUSX4,<Internal format of directory is incorrect>)
.ERR (1454,RNMX13,<Insufficient system resources>)
.ERR (1455,SJBX8,<Illegal to perform this function>)
.ERR (1456,DECRSV,<DEC reserved bits not zero>)

; ERROR CODES 1457-1534 ARE AVAILABLE******

.ERR (1535,TIMX6,<Time has already passed>)
.ERR (1536,TIMX7,<No space available for a clock>)
.ERR (1537,TIMX8,<User clock allocation exceeded>)
```

```
.ERR (1540,TIMX9,<No such clock entry found>)
.ERR (1541,TIMX10,<No system date and time>)

.ERR (1550,SCTX1,<Invalid function code>)
.ERR (1551,SCTX2,<Terminal already in use as controlling terminal>)
.ERR (1552,SCTX3,<Illegal to redefine the job's controlling terminal>)
.ERR (1553,SCTX4,<SC%SCT capability required>)

; Error codes 1554-1677 are available ******

.ERR (1700,SFUSX4,<File expunged>)
.ERR (1701,SFUSX5,<Write or owner access required>)
.ERR (1702,SFUSX6,<No such user name>)
.ERR (1703,GETX3,<Illegal to overlay existing pages>)
.ERR (1704,FILX01,<File is not open>)
.ERR (1705,ARGX01,<Invalid password>)
.ERR (1706,CAPX3,<WHEEL capability required>)
.ERR (1707,CAPX4,<WHEEL or IPCF capability required>)
.ERR (1711,CAPX6,<ENQ/DEQ capability required>)
.ERR (1712,CAPX7,<Confidential Information Access Capability required>)
.ERR (1713,ARGX02,<Invalid function>)
.ERR (1714,ARGX03,<Illegal to change specified bits>)
.ERR (1715,ARGX04,<Argument block too small>)
.ERR (1716,ARGX05,<Argument block too long>)
.ERR (1717,ARGX06,<Invalid page number>)
.ERR (1720,ARGX07,<Invalid job number>)
.ERR (1721,ARGX08,<No such job>)
.ERR (1722,ARGX09,<Invalid byte size>)
.ERR (1723,ARGX10,<Invalid access requested>)
.ERR (1724,ARGX11,<Invalid directory number>)
.ERR (1725,ARGX12,<Invalid process handle>)
.ERR (1726,ARGX13,<Invalid software interrupt channel number>)
.ERR (1727,MONX01,<Insufficient system resources>)
.ERR (1730,MONX02,<Insufficient system resources (JSB full)>)
.ERR (1731,MONX03,<Monitor internal error>)
.ERR (1732,MONX04,<Insufficient system resources (Swapping space full)>)
.ERR (1733,ARGX14,<Invalid account identifier>)
.ERR (1734,ARGX15,<Job is not logged in>)
.ERR (1735,FILX02,<Write or owner access required>)
.ERR (1736,FILX03,<List access required>)
.ERR (1737,DEVX4,<Device is not assignable>)
.ERR (1740,FILX04,<File is not on multiple-directory device>)
.ERR (1741,ARGX16,<Password is required>)
.ERR (1742,ARGX17,<Invalid argument block length>)
.ERR (1743,ARGX18,<Invalid structure name>)
.ERR (1744,DEVX5,<No such device>)
.ERR (1745,DIRX4,<Invalid directory specification>)
.ERR (1746,FILX05,<File expunged>)
.ERR (1747,STRX06,<No such user number>)
.ERR (1750,MSTX24,<Illegal to dismount the Public Structure>)
.ERR (1751,MSTX25,<Invalid number of swapping pages>)
.ERR (1752,MSTX26,<Invalid number of Front-End-Filesystem pages>)
.ERR (1753,LOUTX5,<Illegal to log out job 0>)
.ERR (1754,GJFX43,<More than one ;T specification is not allowed>)
.ERR (1755,MTOX19,<Invalid terminal line width>)
.ERR (1756,MTOX20,<Invalid terminal line length>)
.ERR (1757,MSTX27,<Specified unit is not a disk>)
.ERR (1760,MSTX28,<Could not initialize bit table for structure>)
.ERR (1761,MSTX29,<Could not reconstruct ROOT-DIRECTORY>)
.ERR (1763,DSKX05,<Disk assignments and deassignments are currently prohibited>)
.ERR (1764,DSKX06,<Invalid disk address>)
.ERR (1765,DSKX07,<Address cannot be deassigned because it is not assigned>)
.ERR (1766,DSKX08,<Address cannot be assigned because it is already assigned>)
.ERR (1767,COMX10,<Invalid default string>)
```

```
.ERR (1770,MSTX30,<Incorrect Bit Table counts on structure>)
.ERR (1771,LOCKX1,<Illegal to lock other than a private page>)
.ERR (1772,LOCKX2,<Requested page unavailable>)
.ERR (1773,LOCKX3,<Attempt to lock too much memory>)
.ERR (1774,ILLX01,<Illegal memory read>)
.ERR (1775,ILLX02,<Illegal memory write>)
.ERR (1776,ILLX03,<Memory data parity error >)
.ERR (1777,ILLX04,<Reference to non-existent page>)
.ERR (2000,MSTX31,<Structure already mounted>)
.ERR (2001,MSTX32,<Structure was not mounted>)
.ERR (2002,MSTX33,<Structure is unavailable for mounting>)
.ERR (2003,STDIX1,<The STDIR JSYS has been replaced by RCDIR and RCUSR>)
.ERR (2004,CNDIX7,<The CNDIR JSYS has been replaced by ACCES>)
.ERR (2005,PMCLX1,<Illegal page state or state transition>)
.ERR (2006,PMCLX2,<Requested physical page is unavailable>)
.ERR (2007,PMCLX3,<Requested physical page contains errors>)
.ERR (2010,DLFX10,<Cannot delete directory; file still mapped>)
.ERR (2011,DLFX11,<Cannot delete directory file in this manner>)
.ERR (2012,GJFX44,<Account string does not match>)
.ERR (2013,UTSTX1,<Invalid function code>)
.ERR (2014,UTSTX2,<Area of code too large to test>)
.ERR (2015,UTSTX3,<UTEST facility in use by another process>)
.ERR (2016,BOTX01,<Invalid DTE-20 number>)
.ERR (2017,BOTX02,<Invalid byte size>)
.ERR (2020,DCNX1,<Invalid network file name>)
.ERR (2021,DCNX5,<No more logical links available>)
.ERR (2022,DCNX3,<Invalid object>)
.ERR (2023,DCNX4,<Invalid task name>)
.ERR (2024,DCNX9,<Object is already defined>)
.ERR (2025,DCNX8,<Invalid network operation>)
.ERR (2026,DCNX11,<Link aborted>)
.ERR (2027,DCNX12,<String exceeds 16 bytes>)
.ERR (2030,TTYX01,<Line is not active>)
.ERR (2031,BOTX03,<Invalid protocol version number>)
.ERR (2032,MONX05,<Insufficient system resources (no resident free space)>)
.ERR (2033,ARGX19,<Invalid unit number>)
.ERR (2034,IOX69,<General temporary TAPE error code>)
.ERR (2035,COMX11,<Invalid CMRTY pointer>)
.ERR (2036,COMX12,<Invalid CMBFP pointer>)
.ERR (2037,COMX13,<Invalid CMPTR pointer>)
.ERR (2040,COMX14,<Invalid CMABP pointer>)
.ERR (2041,COMX15,<Invalid default string pointer>)
.ERR (2042,COMX16,<Invalid help message pointer>)
.ERR (2043,COMX17,<Invalid byte pointer in function block>)
.ERR (2044,NPXAMB,<Ambiguous>)
.ERR (2045,NPXNSW,<Not a switch - does not begin with slash>)
.ERR (2046,NPXNOM,<Does not match switch or keyword>)
.ERR (2047,NPXNUL,<Null switch or keyword given>)
.ERR (2050,NPXINW,<Invalid guide word>)
.ERR (2051,NPXNC,<Not confirmed>)
.ERR (2052,NPXICN,<Invalid character in number>)
.ERR (2053,NPXIDT,<Invalid device terminator>)
.ERR (2054,NPXNQS,<Not a quoted string - does not begin with double quote>)
.ERR (2055,NPXNMT,<Does not match token>)
.ERR (2056,NPXNMD,<Does not match directory or user name>)
.ERR (2057,NPXCMA,<Comma not given>)
.ERR (2060,GJFX45,<Illegal to request multiple specifications for the same attribute>)
.ERR (2061,GJFX46,<Attribute value is required>)
.ERR (2062,GJFX47,<Attribute does not take a value>)
.ERR (2063,MSTX34,<Unit is write-locked>)
.ERR (2064,GJFX48,<GTJFN input buffer is empty>)
.ERR (2065,GJFX49,<Invalid attribute for this device>)
.ERR (2077,SJBX7,<Remark exceeds 39 characters>)
.ERR (2100,DELF10,<Directory still contains subdirectory>)
```

```
.ERR (2101,CRDI13,<Request exceeds superior directory working quota>)
.ERR (2102,CRDI14,<Request exceeds superior directory permanent quota>)
.ERR (2103,CRDI15,<Request exceeds superior directory subdirectory quota>)
.ERR (2104,CRDI16,<Invalid user group>)
.ERR (2105,ENACX1,<Account validation data base file not completely closed>)
.ERR (2106,ENACX2,<Cannot get a JFN for <SYSTEM>ACCOUNTS-TABLE.BIN>)
.ERR (2107,ENACX3,<Account validation data base file too long>)
.ERR (2110,ENACX4,<Cannot get an OFN for <SYSTEM>ACCOUNTS-TABLE.BIN>)
.ERR (2111,VACCX0,<Invalid account>)
.ERR (2112,VACCX1,<Account string exceeds 39 characters>)
.ERR (2113,USGX01,<Invalid USAGE entry type code>)
.ERR (2114,BOTX04,<Byte count is not positive>)
.ERR (2115,NODX01,<Node name exceeds 6 characters>)
.ERR (2116,USGX02,<Item not found in argument list>)
.ERR (2117,CRDI17,<Illegal to create non-files-only subdirectory under
     files-only directory>)
.ERR (2120,ENQX23,<Mismatched mask block lengths>)
.ERR (2121,ENQX22,<Invalid mask block length>)
.ERR (2122,DCNX2,<Interrupt message must be read first>)
.ERR (2123,ABRKX1,<Address break not available on this system>)
.ERR (2124,USGX03,<Default item not allowed>)
.ERR (2125,IPCF35,<Invalid IPCF quota>)
.ERR (2126,VACCX2,<Account has expired>)
.ERR (2127,CRDI18,<Illegal to delete logged-in directory>)
.ERR (2130,CRDI19,<Illegal to delete connected directory>)
.ERR (2132,BOTX05,<Protocol initialization failed>)
.ERR (2133,CRDI20,<WHEEL, OPERATOR, or requested capability required>)
.ERR (2134,COMX18,<Invalid character in node name>)
.ERR (2135,COMX19,<Too many characters in node name>)
.ERR (2136,CRDI21,<Working space insufficient for current allocation>)
.ERR (2137,ACESX7,<Directory is "files-only" and cannot be accessed>)
.ERR (2140,CRDI22,<Subdirectory quota insufficient for existing subdirectories>)
.ERR (2141,CRDI23,<Superior directory does not exist>)
.ERR (2142,STRX07,<Invalid user number>)
.ERR (2143,STRX08,<Invalid user name>)
.ERR (2144,CRDI24,<Invalid subdirectory quota>)
.ERR (2146,ATSX01,<Invalid mode>)
.ERR (2147,ATSX02,<Illegal to declare mode twice>)
.ERR (2150,ATSX03,<Illegal to declare mode after acquiring terminal>)
.ERR (2151,ATSX04,<Invalid event code>)
.ERR (2152,ATSX05,<Invalid function code for channel assignment>)
.ERR (2153,ATSX06,<JFN is not an ATS JFN>)
.ERR (2154,ATSX07,<Table length too small>)
.ERR (2155,ATSX08,<Table lengths must be the same>)
.ERR (2156,ATSX09,<Table length too large>)
.ERR (2157,ATSX10,<Maximum applications terminals for system already assigned>)
.ERR (2160,ATSX11,<Byte count is too large>)
.ERR (2161,ATSX12,<Terminal not assigned to this JFN>)
.ERR (2162,ATSX13,<Terminal is XOFF'd>)
.err (2163,ATSX14,<Terminal has been released>)
.ERR (2164,ATSX15,<Terminal identifier is not assigned>)
.ERR (2165,PMCLX4,<No more error information>)
.ERR (2166,ATSX16,<Invalid Host Terminal Number>)
.ERR (2167,ATSX17,<Output failed -- monitor internal error>)
.ERR (2170,FRKHX8,<Illegal to manipulate an execute-only process>)
.ERR (2171,ARGX20,<Invalid arithmetic trap argument>)
.ERR (2172,ARGX21,<Invalid LUUO trap argument>)
.ERR (2173,ARGX22,<Invalid flags>)
.ERR (2174,ATSX18,<ATS input message too long for internal buffers>)
.ERR (2175,ATSX19,<Monitor internal error - ATS input message truncated>)
.ERR (2176,ATSX20,<Illegal to close JFN with terminal assigned>)
.ERR (2177,ARGX23,<Invalid section number>)
.ERR (2200,ARGX24,<Invalid count>)
.ERR (2201,MSTX35,<Too many units in structure>)
.ERR (2202,DCNX13,<Node not accessible>)
```

```
.ERR (2203,DCNX14,<Previous interrupt message outstanding>)
.ERR (2204,DCNX15,<No interrupt message available>)
.ERR (2205,GJFX50,<Invalid argument for attribute>)
.ERR (2206,KDPX01,<KMC11 not running>)
.ERR (2207,NODX02,<Line not turned off>)
.ERR (2210,NODX03,<Another line already looped>)
.ERR (2211,GJFX51,<Byte count too small>)
.ERR (2212,COMX20,<Invalid node name>)
>                                ;END OF .ERCOD DEFINITION


;DEFINE THE ERROR CODE VALUES

DEFINE .ERR (N,E,S) <
        E=:.ERBAS+N
        IFG <N-.ERMAX>,<.ERMAX==:N>>

        .ERMAX==:0

.ERCOD
```

```
;THIS SECTION CONSISTS OF SPECIAL CODE TO WRITE THE ERRMES.BIN FILE
; THE CODE IS ONLY ASSEMBLED IF .ERBLD IS PREVIOUSLY
; DEFINED TO BE NON-ZERO.

IFNDEF .ERBLD,<.ERBLD==0>

IFN .ERBLD,<

.ERGO:  MOVSI 1,(GJ%FOU!GJ%SHT) ;GET A JFN ON ERROR FILE
        HRROI 2,[ASCIZ/ERRMES.BIN/]
        GTJFN
         JRST .ERER
        MOVE 2,[440000,,OF%WR]
        OPENF                   ;OPEN THE FILE FOR WRITE
         JRST .ERER
        MOVNI 3,.ERSTE-.ERTAB   ;GET LENGTH OF FILE
        MOVE 2,[POINT 36,.ERTAB]
        SOUT                    ;OUTPUT THE ERROR FILE DATA
        CLOSF                   ;CLOSE THE FILE
         JRST .ERER
        HALTF                   ;DONE

.ERER:  MOVEI 1,101             ;TYPE OUT ERROR CODE
        HRLOI 2,400000
        SETZ 3,
        ERSTR
         JFCL
         JFCL
        HALTF

LIT

DEFINE .ERR (N,E,S) <
        .ERQQ==<.-.ERTAB>*5
        .ERQQ2==N&37777
        .ERRM1 \.ERQQ2,N,.ERQQ
        ASCII \S'@\
>

DEFINE .ERRM1 (NN,N,.ERQQ)<
   IF1,<IFDEF EZ'NN,<
        PRINTX ERROR N=NN HAS ALREADY BEEN USED
   >>
   EZ'NN==1
   RELOC .ERTAB+NN
        .ERQQ
   RELOC
>

.ERTAB: .ERMAX                  ;FIRST WORD OF TABLE IS THE LENGTH
                                ; OF THE TABLE FOR ERSTR TO USE AS
                                ; A BOUNDS CHECK.
        BLOCK .ERMAX            ;LEAVE ROOM FOR POINTERS

.ERST:  .ERCOD                  ;BUILD STRINGS AND .ERTAB
.ERSTE:                         ;END OF STRINGS

        END .ERGO

>                               ;END OF IFN .ERBLD CONDITIONAL

PURGE .ERR,REL

END
```

APPENDIX B

ACTSYM.MAC

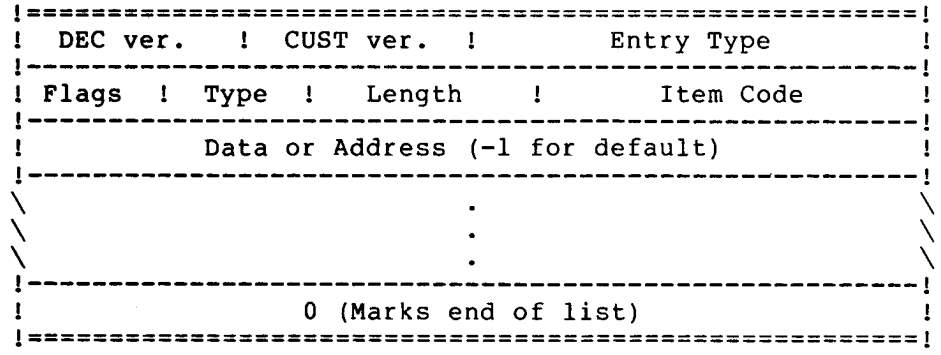
This appendix contains the complete copy of the system file ACTSYM.MAC, which defines the symbols used in the manual. The user must include the statement

```
SEARCH ACTSYM
```

in his program to have the symbols defined in his assembly.

```
        UNIVERSAL ACTSYM - SYMBOL FILE FOR ACCOUNTING
        SUBTTL B.A. HUIZENGA/BAH/TAH - 6-JUN-77

;THIS SOFTWARE IS FURNISHED UNDER A LICENSE AND MAY ONLY BE USED
;  OR COPIED IN ACCORDANCE WITH THE TERMS OF SUCH LICENSE.
;
;COPYRIGHT (C) 1976, 1977, 1978 BY DIGITAL EQUIPMENT CORPORATION, MAYNARD, MASS.


;PARAMETERS FOR USAGE ITEM DESCRIPTORS

;FIELDS IN DATA ITEM DESCRIPTOR

US%FLG==:77B5                   ;FLAGS
    US%IMM==:1B0                ; 1 - IMMEDIATE DATA ITEM
                                ; 0 - ADDRESS OF DATA ITEM
US%TYP==:77B11                  ;TYPE CODE
    .USASC==:0                  ;ASCII
    .USSIX==:1                  ;SIXBIT
    .USOCT==:2                  ;OCTAL
    .USDEC==:3                  ;DECIMAL
    .USDAT==:4                  ;DATE-TIME
    .USTAB==:5                  ;TABLE (SPECIAL FORM)
    .USVER==:6                  ;VERSION NUMBER
    .USSPC==:7                  ;SPACE FILL

US%LEN==:777B20                 ;LENGTH
US%COD==:77777B35               ;ITEM CODE

;RECORD TYPE CODES

        RADIX 10                ;**** NOTE RADIX 10 ****

.UTRST==:1                      ;SYSTEM RESTART ENTRY
.UTSEN==:2                      ;SESSION ENTRY
.UTCKP==:3                      ;CHECKPOINT ENTRY (SYSTEM RESTART)
.UTUSB==:4                      ;FIRST ENTRY OF USAGE FILE (SAME AS .UTRST)
.UTTAD==:5                      ;DATE-TIME CHANGE
.UTBAT==:6                      ;BATCH PROCESSOR
.UTINP==:7                      ;INPUT SPOOLER ENTRY
.UTOUT==:8                      ;OUTPUT SPOOLER ENTRY
.UTFLU==:9                      ;FILE USAGE DIRECTORY ENTRY
.UTDSU==:10                     ;DISK SPINDLE USAGE ENTRY
.UTMNT==:11                     ;STRUCTURE MOUNT ENTRY
.UTMMT==:12                     ;TAPE MOUNT ENTRY
.UTDMT==:13                     ;DECtape MOUNT ENTRY
.UTFCM==:14                     ;FILE COMMAND ENTRY

.UTUSR==:5000                   ;USER-DEFINED ENTRY TYPES ARE 5000-9999

        RADIX 8                 ;**** END OF RADIX 10 ****
```

```
COMMENT &

        The format of the data to be passed to the accounting system
will consist of a list of items describing the entries in a single
record.

The record descriptor list will have a header containing the record
type code and the record version information.

Format of a record descriptor:
```




```
        The generation of these tables will be controlled by the
UITEM. macro. All known data items will have a name generated by
the use of this macro. If any application dependent items are needed
the UITEM. macro may be used to generate the new item. The USENT.
macro may be used to generate the first word of the entry descriptor
table.

All USAGE entry headers and the system-defined USAGE entry types use the
specific item types and these items are ordered by the system.

Installation-defined USAGE entries (with entry types above .UTUSR - 5000 to
9999) use the arbitrary data items (USUAS., USUSX., USUDC., USUOC., USUVR.,
USUDT., and USUSP.) in the order in which they are to be written into the
USAGE entry record.  Each arbitrary data record must be preceded by a
USUAR. item.

Example of installation-defined USAGE entry:

;The following code writes a USAGE entry for a fictitious "file access count"
; in a user program.  This program must be running as an enabled OPERATOR or
; WHEEL.

;Here to write USAGE entry for file access count

        MOVEI T1,.USENT         ;USAGE function to write entry
        MOVEI T2,FILRDB         ;Address of Record Descriptor Block
        USAGE                   ;Write the entry
         ERJMP USGERR           ;Failed to write entry-- do something else
        JRST USGOK              ;Entry written-- go on
```

```
;Record descriptor block for file access count accounting

FILRDB:
        USENT. (.UTUSR+12,1,1)  ;Entry type 5012= file access count.
        USPVR. (<BYTE(3)VWHO(9)VMAJOR(6)VMINOR(18)VEDIT>,US%IMM) ;Version
                                ;  of this program (for header record).

        USUAR.                  ;Start of first arbitrary record.
        USUAS. ([ASCII \This appears in every entry\],,27) ;Text.
        USUSP. (,,5)            ;Space fill, 5 characters.
        USUDC. (FILCNT,,6)      ;Count of file accesses, 000000-999999.

        USUAR.                  ;Start of second arbitrary record.
        USUSX. (<SIXBIT \FILE: \>,US%IMM,6) ;SIXBIT text for filename.
        USUAS. (FILNAM,,200)    ;File name, 200 characters.

        EXP 0                   ;End of entry.

;Storage

FILCNT: BLOCK 1                 ;File access count
FILNAM: BLOCK ^D<200/5>         ;File name text

&       ;;; End of comment
```

```
        SUBTTL UITEM. / USENT. / USAGE. DEFINITIONS

        SALL

DEFINE UITEM. (NAME,TYPE,LEN) <
    DEFINE US'NAME'. (DATA<-1>,IMMED<0>,ULEN<LEN>) <
        USAGE. (.US'NAME,ULEN,TYPE,IMMED,<DATA>)
    >
>

DEFINE USENT. (ETYPE,DVER,CVER) <
        BYTE (9) ^D<DVER>,^D<CVER> (18) ^D<ETYPE>
>

DEFINE USAGE. (CODE,LENGTH,TYPE,FLAGS,DATUM) <
        FLAGS+<TYPE>B11+<^D<LENGTH>>B20+CODE
        DATUM
>

DEFINE USDSK. (TABLE) <
        USAGE. (.USDST,0,.USTAB,US%IMM,<TABLE>)
>
```

```
        SUBTTL USAGE. ITEM-CODE DEFINITIONS

DEFINE USLIST <

DEFUS (JNO,0,.USDEC,4)          ;JOB NUMBER
DEFUS (TAD,1,.USDAT,14)         ;CURRENT DATE/TIME
DEFUS (TRM,2,.USASC,1)          ;TERMINAL DESIGNATOR
DEFUS (LNO,3,.USOCT,4)          ;LINE NUMBER
DEFUS (PNM,4,.USSIX,6)          ;PROGRAM NAME (CALLER)
DEFUS (PVR,5,.USVER,15)         ;PROGRAM VERSION
DEFUS (AMV,6,.USVER,15)         ;ACCOUNTING MODULE VERSION
DEFUS (NOD,7,.USSIX,6)          ;CALLER'S LOCATION
DEFUS (PPN,10,.USOCT,12)        ;PROJECT / PROGRAMMER NUMBER (TOPS10 ONLY)
DEFUS (NM1,11,.USASC,12)        ;NAME OF USER (TOPS10)
DEFUS (SNM,12,.USASC,39)        ;SYSTEM NAME
DEFUS (MVR,13,.USVER,15)        ;MONITOR VERSION NUMBER
DEFUS (MBD,14,.USDAT,14)        ;MONITOR BUILD DATE
DEFUS (MUP,15,.USDEC,18)        ;MONITOR UPTIME (IN SECONDS)
DEFUS (ACT,16,.USASC,39)        ;ACCOUNT STRING
DEFUS (LCK,17,.USDAT,14)        ;TIME OF LAST CHECKPOINT
DEFUS (RTM,20,.USDEC,9)         ;RUNTIME IN MS
DEFUS (CTI,21,.USDEC,11)        ;CORE-TIME INTEGRAL (TOPS10 ONLY)
DEFUS (SST,22,.USDAT,14)        ;SESSION START TIME
DEFUS (JTY,23,.USDEC,1)         ;JOB TYPE (BATCH / TIMESHARING)
DEFUS (BJN,24,.USSIX,6)         ;BATCH JOB NAME
DEFUS (BSN,25,.USDEC,6)         ;BATCH SEQUENCE NUMBER
DEFUS (COM,26,.USASC,39)        ;USER COMMENT
DEFUS (DKR,27,.USDEC,8)         ;DISK READS
DEFUS (DKW,30,.USDEC,8)         ;DISK WRITES
DEFUS (VTI,31,.USDEC,11)        ;VIRTUAL CORE-TIME INTEGRAL
DEFUS (EBX,32,.USDEC,9)         ;EBOX MEGACOUNTS (CYCLES * 10^6)
DEFUS (MBX,33,.USDEC,9)         ;MBOX MEGACOUNTS (CYCLES * 10^6)
DEFUS (MCL,34,.USDEC,6)         ;MONITOR CALLS
DEFUS (MCM,35,.USDEC,6)         ;MONITOR COMMANDS
DEFUS (SCL,36,.USDEC,3)         ;SCHEDULING CLASS
DEFUS (TYI,37,.USDEC,6)         ;TTY INPUT CHARACTERS
DEFUS (TYO,40,.USDEC,6)         ;TTY OUTPUT CHARACTERS
DEFUS (TYW,41,.USDEC,6)         ;TTY WAKEUPS
DEFUS (CPN,42,.USDEC,1)         ;NUMBER OF CPUS
DEFUS (CP0,43,.USDEC,4)         ;SERIAL NUMBER OF CPU0
DEFUS (CP1,44,.USDEC,4)         ;SERIAL NUMBER OF CPU1
DEFUS (CP2,45,.USDEC,4)         ;SERIAL NUMBER OF CPU2
DEFUS (CP3,46,.USDEC,4)         ;SERIAL NUMBER OF CPU3
DEFUS (CP4,47,.USDEC,4)         ;SERIAL NUMBER OF CPU4
DEFUS (CP5,50,.USDEC,4)         ;SERIAL NUMBER OF CPU5
DEFUS (RQQ,51,.USDEC,11)        ;RUN QUEUE QUOTIENT (TOPS10 ONLY)
DEFUS (NM2,52,.USASC,39)        ;NAME OF USER (TOPS20)
DEFUS (CCT,53,.USDEC,7)         ;CONSOLE CONNECT TIME (SECONDS)
DEFUS (DTL,54,.USDAT,14)        ;DATE/TIME BEFORE CHANGE (STAD)
```

```
;DISK UTILIZATION RECORD ENTRIES

DEFUS (NRF,55,.USDEC,3)          ;NUMBER OF RECORDS FOLLOWING
DEFUS (TAL,56,.USDEC,10)         ;TOTAL ALLOCATED STORAGE
DEFUS (TUS,57,.USDEC,10)         ;TOTAL STORAGE USED
DEFUS (TNF,60,.USDEC,5)          ;TOTAL NUMBER OF FILES
DEFUS (STR,61,.USASC,6)          ;STRUCTURE NAME
DEFUS (STP,62,.USDEC,1)          ;STRUCTURE TYPE CODE
DEFUS (KTP,63,.USDEC,3)          ;CONTROLLER TYPE
DEFUS (DTP,64,.USDEC,3)          ;DEVICE TYPE
DEFUS (LIQ,65,.USDEC,6)          ;LOGGED IN QUOTA
DEFUS (LOQ,66,.USDEC,6)          ;LOGGED OUT QUOTA
DEFUS (LLI,67,.USDAT,14)         ;LAST LOGGED IN DATE/TIME
DEFUS (LAT,70,.USDAT,14)         ;LAST DISK ACCOUNTING DATE/TIME
DEFUS (EXP,71,.USASC,1)          ;EXPIRED DIRECTORY (Y/N)
DEFUS (DIR,72,.USASC,39)         ;DIRECTORY NAME
DEFUS (ALC,73,.USDEC,10)         ;ALLOCATED STORAGE
DEFUS (USG,74,.USDEC,10)         ;STORAGE USED
DEFUS (FIL,75,.USDEC,5)          ;NUMBER OF FILES
DEFUS (FON,76,.USASC,1)          ;FILES ONLY INDICATOR (Y/N)

;SPOOLER INFORMATION RECORD ENTRIES

DEFUS (SRT,77,.USDEC,9)          ;SPOOLER RUNTIME
DEFUS (SCI,100,.USDEC,11)        ;CORE-TIME INTEGRAL
DEFUS (SDR,101,.USDEC,8)         ;SPOOLER DISK READS
DEFUS (SDW,102,.USDEC,8)         ;SPOOLER DISK WRITES
DEFUS (JNM,103,.USSIX,6)         ;JOB NAME
DEFUS (QNM,104,.USSIX,3)         ;QUEUE NAME
DEFUS (SDV,105,.USSIX,6)         ;PROCESSING DEVICE
DEFUS (SSN,106,.USDEC,6)         ;SEQUENCE NUMBER
DEFUS (SUN,107,.USDEC,6)         ;SPOOLER UNITS PROCESSED
DEFUS (CRT,110,.USDAT,14)        ;CREATION DATE/TIME OF REQUEST
DEFUS (DSP,111,.USSIX,6)         ;DISPOSITION
DEFUS (TXT,112,.USASC,39)        ;OPR OR SYSTEM TEXT
DEFUS (PRI,113,.USDEC,2)         ;PRIORITY
DEFUS (SNF,114,.USDEC,5)         ;NUMBER OF FILES PROCESSED
DEFUS (SCD,115,.USDAT,14)        ;SCHEDULED DATE/TIME
DEFUS (FRM,116,.USSIX,6)         ;FORMS TYPE

;DATE/TIME CHANGE RECORD ENTIRES

DEFUS (OFD,117,.USDEC,7)         ;OFFSET IN DAYS
DEFUS (OFS,120,.USDEC,7)         ;OFFSET IN SECONDS
DEFUS (ODT,121,.USDAT,14)        ;OLD DATE/TIME

;ARBITRARY RECORD ITEM TYPES

DEFUS (UAR,122,.USSPC,0)         ;USER-DEFINED ARBITRARY RECORD DELIMITER
DEFUS (UAS,123,.USASC,0)         ;USER-DEFINED ASCII STRING
DEFUS (USX,124,.USSIX,0)         ;USER-DEFINED SIXBIT STRING
DEFUS (UOC,125,.USOCT,0)         ;USER-DEFINED OCTAL NUMBER
DEFUS (UDC,126,.USDEC,0)         ;USER-DEFINED DECIMAL NUMBER
DEFUS (UDT,127,.USDAT,14)        ;USER-DEFINED DATE AND TIME
DEFUS (UVR,130,.USVER,15)        ;USER-DEFINED VERSION (STANDARD FORMAT)
DEFUS (USP,131,.USSPC,0)         ;USER-DEFINED SPACE FILL

>;;; END OF USLIST
```

```
;MACRO TO DEFINE ALL USAGE. ITEM CODES

DEFINE DEFUS (NAM,VAL,TYP,LEN) <
    IF1,<IFDEF .US'NAM,<
        PRINTX .US'NAM ALREADY DEFINED
    >>
        .US'NAM==:VAL
        UITEM. (NAM,TYP,LEN)
>

;EXPAND ALL DEFINITIONS

USLIST

;SPECIAL ITEM TYPE CODE DEFINITIONS

.USDSX==:7776           ;STRUCTURE/DIRECTORY INFO WORD (SPECIAL)
.USDST==:7777           ;DISK STATISTICS TABLE POINTER


        END
```

DECSYSTEM-20
Monitor Calls
Reference Manual
AD-4166C-T1

READER'S COMMENTS

NOTE: This form is for document comments only. DIGITAL will use comments submitted on this form at the company's discretion. Problems with software should be reported on a Software Performance Report (SPR) form. If you require a written reply and are eligible to receive one under SPR service, submit your comments on an SPR form.

Did you find errors in this manual? If so, specify by page.

_____

_____

_____

_____

_____

_____

Did you find this manual understandable, usable, and well-organized? Please make suggestions for improvement.

_____

_____

_____

_____

_____

_____

Is there sufficient documentation on associated system programs required for use of the software described in this manual? If not, what material is missing and where should it be placed?

_____

_____

_____

_____

_____

_____

Please indicate the type of user/reader that you most nearly represent.

- ☐ Assembly language programmer
- ☐ Higher-level language programmer
- ☐ Occasional programmer (experienced)
- ☐ User with little programming experience
- ☐ Student programmer
- ☐ Non-programmer interested in computer concepts and capabilities

Name _____ Date _____

Organization _____ Telephone _____

Street _____

City _____ State _____ Zip Code or Country _____

Please cut along this line.

------Do Not Tear - Fold Here and Tape------

digital

No Postage
Necessary
if Mailed in the
United States

BUSINESS REPLY MAIL
FIRST CLASS PERMIT NO.33 MAYNARD MASS.

POSTAGE WILL BE PAID BY ADDRESSEE

SOFTWARE PUBLICATIONS
200 FOREST STREET MR1-2/E37
MARLBOROUGH, MASSACHUSETTS 01752

----- Do Not Tear - Fold Here and Tape------

Cut Along Dotted Line