

BASE PAGE CONSTANTS

Loc.	Value	Loc.	Value
40	-64	60	5
41	-10	61	6
42	-9	62	7
43	-8	63	8
44	-7	64	9
45	-6	65	10
46	-5	66	17
47	-4	67	64
50	-3	70	178
51	-2	71	378
52	-1	72	778
53	0	73	1778
54	1	74	3778
55	2	75	1774008
56	3	75	37778
57	4	77	17777008

**EFMP UTIL PROGRAM COMMANDS
(:PR,UTIL[,N])**

- /INIT, pack, dirsz
- /CREATE, file, pack, flgth, rlgth, scoode, ustat
- /DESTROY, file, pack, scoode
- /OPEN, file, pack, rcdno, scoode
- /CLOSE, file, ustat
- /RESET, file, pack, rcdno
- /STATUS, DF, file, pack
- /STATUS, FO, file
- /STATUS, SC, file, pack, scoode
- /STATUS, LR, file, pack
- /STATUS, LF, pack
- /STATUS, NF, pack, statb
- /STATUS, AP, subch, statb
- /REPACK, pack
- /COPY, file, pack
- /CHANGE, ofile, nfile, pack, scoode

/POST

/BRIEF, file, scoode
/END

All of the above are also EFMP EXEC call functions except BRIEF and END.

PROGRAM TYPE AND PAGE LINKING CODES

Assembler NAM Statement:

NAM, name [,type] [,link model]

type is program entry code (in octal):

- 0 = System main-memory resident (default)
- 1 = Disc-resident EXEC module
- 2 = (reserved for system)
- 3 = User program, main
- 4 = Disc-resident device driver
- 5 = User program, segment
- 6 = Library routine
- 7 = Subroutine
- 10 = Relocatable binary
- 11 = ASCII source statements
- 12 = Binary data
- 13 = ASCII data
- 14 = Absolute binary

not acceptable to Relocating Loader

link mode is:

- 0 for current-page linking
- non-0 for base page linking (default)

FORTLAN PROGRAM Statement:

PROGRAM name [,type]

type is program entry code (in octal):

- 3 = main program (default)
- 5 = segment
- 6 or 7 = library

(link mode is always base page linking)

ALGOL Control Statement:

HPAL [L,A,B,P], "name", [error routine] [,type]

type is program entry code (in octal):

- 3 = main program (default)
 - 5 = segment
 - 6 or 7 = utility subroutine or procedure
- (link mode is always base page linking)

PERIPHERAL STATUS RETURNS

Peripheral status information is stored in the device's EOT (word 4, bits 0-7). A user gains access to this information by executing an I/O status EXEC call (IRCODE = 13) in his program.

DISC STATUS RETURNS

Bit Set to 1	Meaning	
	7900, 7901	2883, 2884
0	Any error	
1	Data error	
2	Drive busy	
3	Flagged cylinder	
4	Address error	
5	End of cylinder	
6	Device not ready	
7	(not used)	Drive ID = 1

LINE PRINTER STATUS RETURNS

Bit Set to 1	Meaning		
	2610/2614	2607	2613/2618
0	(not used)		V/F tape hole in chan. 12
1	(not used)		V/F tape hole in chan. 9
2	Power off	Paper out	Not ready
3	PRINT	START	ONLINE
4		Printer idle	
5		(not used)	
6		Last operation top-of-form (this is only bit used with HP 2762)	
7		(not used)	

MAGNETIC TAPE STATUS RETURNS

Bit Set to 1	Meaning
0	Tape unit busy, or in local mode
1	Parity error on last read or write operation
2	No write-enable ring, or the unit is rewinding
3	I/O request rejected
4	Timing error on last read or write operation
5	End-of-tape marker sensed
6	Start-of-tape marker sensed
7	End-of-file record encountered when reading, forward spacing, or back-spacing



24307B
DOS - III
Disc Operating System

pocket guide

HP Computer Museum
www.hpmuseum.net

For research and education purposes only.

OPERATING DOS - III

LOAD DOS-III USING BMDL

1. Configured DOS-III must be on subchannel 1 (7900/7901) or subchannel 0 (2883/2884).
2. Set starting address of 037750 (16K system), 057750 (24K system), or 077750 (32K system).
3. Enable (unprotect) BMDL.
4. Press PRESET (INTERNAL and EXTERNAL).
5. Press run. Halt = 102077 in Display Register.
6. Protect BMDL (if necessary).
7. Set disc subchannel number of system to be loaded into Switch Register bits 5 through 0.
8. Press run. Result is the initiation message from DOS-III:
INPUT :DATE,XXXXXXXXXX [H,M]

INITIATE DOS-III

INPUT :DATE,XXXXXXXXXX [H,M]
DOS-III system ready

:DATE,day [hour, minute]
Set day [and time, if Time-base Generator is present]

:JOB [name]
Initiate user job [and assign it a name]

:SS [n1,...]
Set system search for file names through all active sub-channels [or through specified subchannel order]

JOB CONTROL

:ABORT
Abort current job before it has run to completion

:COMMENT string
Print message string on console, without interrupting current job

:EA [p1,... p5]
Execute user EXEC module \$EX36 [passing parameters to it]

:EB [p1,... p5]
Execute user EXEC module \$EX37 [passing parameters to it]

:EJOB
Terminate current batch and/or job normally; repack disc

:GO [p1,... p5]
Restart suspended program [passing parameters to it]

:FILE, file
Specify disc source input file for assembly or compilation

:JOB [name]
Initiate user job [and assign it a name]

:OFF
Abort currently-executing program or operation, without terminating job

:PAUSE [comment string]
Suspend current program until :GO is entered [display comment string on console]

:PROG, name [p1,... p5]
Execute system or user program [passing parameters to it]

:RUN, name [time] [N]
Execute a system or user program [maximum minutes] [ignore illegal EXEC request codes]

OPERATING MODES

:BATCH, logical unit
Switch from keyboard to batch mode and/or reassign logical unit as batch input device

:TYPE
Switch from batch to keyboard mode; these directives can be used in keyboard mode only: :ABORT, :DATE, :GO, :IN, :OFF

LIST (DISC PROGRAM SOURCE FILE)

:LIST, S, logical unit, file [m1,n1]
List with sequential line numbers an entire source file [or from statement m [to n] or EOF] on the specified logical unit

:LIST (DISC DIRECTORY)
:LIST, U, logical unit [file1,... file13]
List User Area directory information for all [or specified] user files on specified logical unit

:LIST, X, logical unit [file1,... file13]
List System Area directory information for all [or specified] system files on specified logical unit

LIST (SYSTEM TABLES)

:EQ [eqt#]
List the entire Equipment Table (EOT) [or specified entry] on console

:LU [lu# [eqt#]]
List the entire Logical Unit Table (LUT) [or the EOT number assigned to the specified logical unit number] on console [and assign a new EOT number to the logical unit number]

DUMP (DISC AREA TO LIST DEVICE)

:SA, track, sector [number]
Dump one [or a number of] sectors, starting at track, sector onto list device in ASCII format

:SO, track, sector [number]
Same as above, but octal format

DUMP (DISC TO DISC)

:DD [X]
Dump all [or System Area] of current user disc to another subchannel; follow immediately with a :UD specifying destination disc

:DD, U [file1 [name1],...]
Dump all current user disc files [or specified [optionally renamed] files] to another subchannel; follow immediately with a :UD specifying destination disc

DUMP (DISC FILE TO PERIPHERAL)

:DUMP, logical unit, file [S1 [S2]]
Dump in format appropriate to the specified logical unit the entire named file [or relative sectors S1 to end [or S1 to S2]]

DUMP (CORE PROGRAM TO LIST DEVICE)

:ADUMP [FWA [LWA]] [B] [S]
If next program aborts, dump it with its library routines to list device [starting at address [to another address]] [include base page area] [include system area]

:PDUMP [FWA [LWA]] [B] [S]
If next program runs to normal completion, dump as for :ADUMP, above

FILES (RESERVE SPACE FOR DATA)

:STORE, A, file, sectors
Reserve a number of disc sectors for named ASCII data file

:STORE, B, file, sectors
Reserve a number of disc sectors for named binary data file

FILES (CREATE AND SAVE ON DISC)

:STORE, P [name1,... name14]
Store as permanent files on disc all [or specified] programs generated and relocated by Relocating Loader during current job

:STORE, R, file [logical unit]
Store all relocatable programs from disc Job Binary Area [or input from specified logical unit] into named file

:STORE, S, file, logical unit [C]
Store in named source file a source statement file input from specified logical unit [and allow : input for creating batch file]

:STORE, X, file, logical unit
Store in named binary file the absolute binary file input from specified logical unit

UTILITY DIRECTIVES (MISCELLANEOUS)

:CLEAR [logical unit]
Issue a clear command to Job Binary Area [or to the specified logical unit]

:DN, eqt#
Declare EOT number unavailable for remainder of job

:EF [logical unit]
Write EOF on magnetic tape logical unit 8 [or on specified logical unit]

:LU, lu#, eqt#
Assign device specified in EOT number to the logical unit number

:MMGT [subsystem, n,...]
Return values set by previous :MMGT [or assign n words to subsystem]; followed by :RUN or :PROG

:RWND [logical unit]
Rewind magnetic tape logical unit 8 [or specified logical unit]

:TOP [logical unit]
Issue a top of form to list device on logical unit 6 [or to specified logical unit]

:UP, eqt#
Declare EOT number available

UTILITY DIRECTIVES (DISC)

:CLEAR
Clear Job Binary Area of current user disc

:IN, label
Label current user disc if unlabeled; or, if disc is labeled, purge System Area, repack disc, and assign new label

:IN, *
Delete label from current disc and purge System and User Areas; or, if disc is unlabeled, purge user directory and files

:PURGE [file1,... file15]
Flag all [or specified] user files for deletion by :RPACK, :EJOB, or :JOB directives

:RNAME, oldname, newname [type]
Rename user file [changing file type]

:RPACK
Delete flagged and temporary user files, repacking disc

:SS [n1,...]
Set system search for file names through all active sub-channels [or through specified subchannel order]

:TRACKS
Output to console the number (decimal) of next available track after User Area, and number of faulty tracks encountered, if any

:UD
Output current user disc subchannel number and label to console

:UD, label
Assign subchannel with label as current user disc; output subchannel number to console

:UD, label, n
Assign n as current user disc, if it has the specified label

:UD,,n
Output label of subchannel n to console

:UD, *
Assign highest number unlabeled disc as current user disc

:UD, *, n
Assign subchannel n as current user disc, if n is unlabeled; or, if labeled, output label of n to console (making no assignment)

LOADING PROGRAMS

:PROG,LOADR[,P1,P2,P3,P4,P5]

P1 is 0 for loading from Job Binary Area and relocatable library (default); 2 for loading from Job Binary Area, user files, and relocatable library; n for loading from Job Binary Area, user files, relocatable library, and paper tape or magnetic tape (logical unit n)

P2 is list device logical unit number (default is 6)

P3 is 0 for no DEBUG, is non-0 for DEBUG (default is 0)

P4 is 0 for base page linking, is non-0 for current page linking (default is 0)

P5 is 0 for system default program bounds (e.g., lower base page bound to upper base page bound and lower main memory bound to upper main memory bound); is 1 for user-specified program bounds (default is 0)

RELOCATING LOADER ERROR MESSAGES

L01
Checksum error on tape

L02
Illegal record

L03
Memory overflow

L04
Base page overflow

L05
Symbol table overflow

L06
Duplicate main or segment name (may be caused by attempting to run the Loader twice in one job).

L07
Duplicate entry point

L08
No main or segment transfer address

L09
Record out of sequence

L10
Insufficient directory work area or user area space

L11
Program table overflow

L12
User file specified cannot be found

L13
Program name duplication

L14
Non-zero base page length

L15
Segment occurred before main

L16
Program overlay (illegal ORG)

L17
Illegal library record

L18
Illegal octal digit in base page bounds specification; or the lower base page bound is greater than the upper base page bound; or the lower or upper base page bound is greater than 2000g. In keyboard mode, re-enter new base page bounds. In batch mode, Loader aborts.

L19
Illegal octal digit in main memory bounds specification; or the lower program bound is greater than the upper program bound. In keyboard mode, re-enter new program bounds. In batch mode, Loader aborts.

BP BND [L,U]?

Specify the base page bounds desired for the program being loaded. [The bounds, L = lower, U = upper, should be entered as two octal constants, separated by a comma.]

ENTER FILE NAME(S) OR/E
Enter list of relocatable program files. To terminate list, enter /E.

LOADR COMPLETE
Loading has completed. No responses required.

LOADR SUSP

Loader has suspended (usually at End-of-Tape). Type :GO,n to restart the Loader with proper parameter value.

LOADR TERMINATED

Loader has terminated because of an error. Correct input.

LOAD TAPE

In conjunction with LOADR SUSP, this message requests that next relocatable tape be loaded before :GO. Load the next relocatable tape and enter :GO to read next tape or :GO,1 to indicate that all tapes are read in.

NO PROGRAMS LOADED

No programs were loaded. Loader terminates.

PROG BND [L,U]?

Enter program bounds for the program being loaded. [The bounds, L = lower, U = upper, should be entered as two octal constants, separated by a comma.]

UNDEFINED EXTS

Undefined external references exist in programs loaded. The external references are listed one per line. To load additional programs from paper tape, enter :GO,0,logical unit.

EDIT COMMANDS (:ED, FILE, LU [,NEW FILE])

/DELETE, m [n]

Delete line m [through n].

/END

Terminate EDIT operation.

/INSERT, m

Insert line(s) after line m.

/MERGE [k], secondary file [m],n]

Add secondary file [line m [through n]] to primary file at its end [or merge after line k].

/REPLACE, m [n]

Replace line m [through n] with line(s).

/SUPPRESS

Suppress echo of EDIT operations if logical unit number used is not the Console.

/UNSUPPRESS

Resume echo.

DEBUG COMMANDS (:PROG,LOADR,,1)

COMMANDS FOR 2100 RTE/DOS DEBUG AND 21MX RTE/DOS DEBUG

B,n

Instruction breakpoint at octal address n (Note: if n = USB EXEC, a memory protect violation occurs).

D,A,n1[n2]

ASCII dump of octal main memory address n1 or from n1 through n2.

D,B,n1[n2]

Binary dump of octal main memory address n1 or from n1 through n2.

M,n

Set absolute base of relocatable program unit at octal address n.

R[,n]

Execute user program starting at octal address n or execute starting at next location in user program (used after a breakpoint or to initiate the program at the transfer point in the user program).

S,n,d

Set octal value d in octal address n.

S,n,d1,d2,...,dn

Set octal values d1 through dn in successive memory locations beginning at octal address n.

W,A,d

Set A-register to octal value d

W,B,d

Set B-register to octal value d.

W,E,d

Set E-register to octal value d (0 = off; non-zero = on).

W,O,d

Set Overflow to octal value d (0 = off; non-zero = on).

X,n

Clear breakpoint at octal address n.

A

Abort DEBUG operation.

COMMANDS FOR 21MX RTE/DOS DEBUG ONLY

W,X,d

Set X-register to octal value d

W,Y,d

Set Y-register to octal value d

ALGOL, ASSEMBLER, AND FORTRAN :PROG DIRECTIVES

:PR, [ALGOL [ASMB [FTN4 [P1,P2,P3,P4,99]

P1 is the logical unit number of input device (default is 5; set to 2 for source file input indicated by a JFILE directive).

P2 is the logical unit number of the list device (default is 6).

P3 is the logical unit number of punch device (default is 4)

P4 is the lines/page on the source listing (default is 56).

99 if present, the object program is stored in the Job Binary Area for later loading. Any requested punch output still occurs. The 99 may occur anywhere in the parameter list, but terminates the list.

SYSTEM ASSIGNMENTS

Device or Accessory	Driver Is	Uses DMA?	Memory Resident?	STD LU Assign	Actual LU	Actual EOT	Actual I/O Slot
System Console (TTY controller)	DVR05	No	Yes	1	1		
Disc	DVR31	Yes	Yes	2,3	2,3		
7900, 7901, ..., 883, 8884							
Paper Tape PUNCH (2885, 2753)	DVR02	No	No	4			
Paper Tape Reader	DVR01	No	No	5			
(2748)	DVR12	No	No	6			
Line Printer (2610, 2013, 4, 2607, 2767)	DVR11	Yes	Yes	7			
Card Reader (7892)	DVR23	Yes	Yes	8,9			
Mapletape Tape Unit (7970)	DVR00	No	No				
Terminal Printer (2749)	DVR26	No	No				
(2762, 2615)	DVR04	No	No				
Display Terminal (2616)	DVR33	Read No Print No Write No	No				
Windup Control Store							
Timebase Generator (12539C)							
Privileged Interrupts (12936A)							
12897B Asynchronous Data Set Interface	DVR12	No	Yes				
12920B Asynchronous Multiplexer Interface	DVR73	No	Yes				
12889B High speed Serial Interface	DVR67	Yes	Yes				
1261BB Synchronization Data Set Interface	DVR70	No	Yes				

DEVICE EOT FORMAT

EOT word	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
1	Comments															
2	Driver Initiation Section Address															
3	Driver Continuation Section Address															
4	Equip. Type															
5	Request Return Address (Reserved)															
6	Request Return Address (Reserved)															
7	Request Return Address (Reserved)															
8	Request Return Address (Reserved)															
9	Request Buffer Address															
10	Request Buffer Length															
11	Temporary, or Disc Track #															
12	Temporary, or Starting Sector #															
13	Upper Memory Address - Main Driver Area															
14	Upper Memory Address - Driver Linkage Area															
15	Starting Track #															
16	Starting Sector #															
17	Units															

PROGRAMMING DOS - III

SUMMARY OF EXEC CALLS

FORTRAN call format:
CALL EXEC (Request code [other parameters])

- 19 [not allowed from FORTRAN]
Store value into base page location (first put value in A-reg; address in B-reg)
- 1 .ICNWD, IBUFR, IBUFL, ITRAK, ISECT]
Read from a peripheral or work area of disc.
- 2 .ICNWD, IBUFR, IBUFL, ITRAK, ISECT]
Write to a peripheral or work area of disc.

3 .ICNWD [IPRM1]
Execute an I/O control operation.

6
Terminate program

7
Suspend program

8 .NAMS [IPRM1 . . . 5]
Load segment of program.

10 .NAMP [IPRM1 . . . 5]
Transfer main program into main memory.

11 .IARAY
Request time of day.

13 .LUN, ISTAT [ITLOG]
Request device status.

14 .ICNWD, IBUFR, IBUFL, NAMEF, IR [IPRM1]
Read user disc data file.

15 .ICNWD, IBUFR, IBUFL, NAMEF, IR [IPRM1]
Write user disc data file

16 .NTRAK, IREQT, ISTRK
Ascertain if n contiguous work tracks are available.

17 .IFTRK, LTRAK, ISIZ [IDISC]
Return first and last tracks of work area.

18 .NAMEF, NSECT [IPRM1]
Search for file in directory

23 .LABEL [ICHNL]
Change current user disc subchannel.

24 .IEFMP (see EFMF in DOS III Manual)
Execute EFMF function call.

27 [IPRM1 . . . 5]
Execute user EXEC module \$EX36.

28 [IPRM1 . . . 5]
Execute user EXEC module \$EX37.

29 [IPRM1 . . . 5]
Execute subroutine-like return from a segment to a main program.

30 .MPTK
Control memory protect from user program.

32 .ISTAT, NAMEF, ITY, NSECT
Create disc file under program control.

33 .ISTAT [NAMEF]
Purge disc file under program control.

34 .ISTAT, NAMO, NAMN [ITYPE]
Change file name under program control.

35 .ISTAT, IWDS, ISAD, IBID [IPRM1]
Reserve block of memory under block name identifier (IBID).

36 .IWDS [IBID]
Return number of words [in BID] or unallocated.

38 .ISTAT, IWDS, ISAD, ID [IBID]
Allocate buffer [in BID] or from available memory.

41 .ISTAT, ID
Release reserved buffer space. If buffer was under BID, buffer is released but logical address space remains reserved (IRCDCE 35).

EXEC CALL PARAMETER KEYS

- IARAY**
Time value array (BSS 5).
- IBID**
Block name ID.
- IBUFL**
Buffer length (decimal).
- IBUFR**
Buffer address (BSS n)
- ICHNL**
Disc subchannel.
- ICNWD**
I/O control word (octal).
- ID**
Buffer ID.
- IDISC**
0 = system disc non-0 = user disc.
- IEFMP**
EFMP function number.
- IFTRK**
First track number available.
- IPRM1**
Parameter
- IREQT**
Desired first track number.
- IR**
Relative sector number (decimal)
- ISAD**
Starting address
- ISECT**
Sector number (decimal)
- ISIZ**
Number of sectors/track.
- ISTAT**
Status
- ISTRK**
First track number available.
- ITLOG**
Transmission log
- ITRAK**
Track number (decimal)
- ITYPE**
Program type
- IWDS**
Number of words
- LABEL**
Disc label (ASCII)
- LTRAK**
Last track number available
- LUN**
Logical unit number
- MPTK**
Memory protect parameter flag (decimal)
- NAMEF**
File name (ASCII)
- NAMN**
New file name
- NAMO**
Old file name
- NAMP**
Program name (ASCII)
- NAMS**
Segment name (ASCII)
- NSECT**
Number of sectors
- NTRAK**
Number of tracks (decimal)