

DEC-10-MRZA-D

PDP-10 MONITOR TABLE DESCRIPTIONS

1st Printing October 1970

Copyright © 1970 by Digital Equipment Corporation

The material in this document
is for information purposes
and is subject to change with-
out notice.

The following are trademarks of Digital
Equipment Corporation, Maynard, Massachu-
setts:

DEC

FLIP CHIP

DIGITAL

PDP

FOCAL

COMPUTER LAB

CONTENTS

	<u>Page</u>
AVALTB - Available Devices Table	1
AVLQTB - Requeuing Table for Device Available	2
Buffer Ring	3
Buffer Ring Header	5
CIPWT - Clock Request Queue	6
COMTAB - Commands Table	7
CORTAB - Core Assignment Table	8
DEVDSP - Device Dispatch Table	9
Device Data Block	11
Disk Access Table	15
Disk Device Data Block	16
Disk File Directory	21
DISP - Command Dispatch Table	22
Fragment Table	23
INTTAB - Interrupt Routine Table	24
JBTADR - Job Address Table	25
JBTCHK - Job Checksum Table	26
JBTDIR - Job Directory Table	27
JBTCKT - Job Kilo-Core-Tick Table	28
JBTNAM - Job File Name Table	29
JBTPRV - Job Privileges Table	30
JBTQ - Job Queues Table	31
JBTSGN - Job High Segments Table	32
JBTSTS - Job Status Table	34
JBTSWP - Job Swap Table	37
JBTUFD - Job UFD Table	39
JDA - Job Device Assignments Table	40
JOB DAT - Job Data Area	41
Job Size - Queue Table	43
LINTAB - TTY Line Characteristics Table	44
MQUEUE - Monitor Disk I/O Queue	45
QBITS - Requeuing Table for Wait State Code	46
Quantum Time Table	47
Queue Progression Table	48
REQTAB - Required Sharable Resources Table	49
RIB - Disk Retrieval Information Block	50
RTIME - Job Incremental Run Time Table	53
Scan Table	54
SPCTAB - TTY Special Characters Table	55
Storage Allocation Table	56
SWPTAB - Swapping Space Table	57

CONTENTS (cont.)

Transfer Table	58
TTIME - Job Total Run Time Table	59
TTY Device Data Block	60
TTYTAB - TTY Table	66
UCLJMP - CALL UO Dispatch Table	67
UCLTAB - CALL UO Names Table	68
UUOTAB - UO Dispatch Table	69

AVALTB

Contains flags to indicate that a sharable device has become available.
Each entry referenced by its own label.

RNAVAL	
WSAVAL	
TSAVAL	
STAVAL	
AUAVAl	
MQAVAL	
DAAVAL	
DTAVAL	
DCAVAL	
MTAVAL	

Entry is nonzero if the corresponding sharable resource has become available since last scheduling and some job is waiting for it.

Defined in SCHED.

Referenced in

CLOCK1	Sched	UUCON
DTASRN	DTCSRn	MTCSR6
MTASRX	MTASBX	MTCSR6
DSKSER		

Notes:

1. First three entries are dummy entries which do not represent sharable resources.
2. Table REQTAB has entries corresponding to AVALTB entries.

AVLQTB

Specifies a queue transfer table to remove a job from any sharable resource wait queue when the resource becomes available.

One entry for each sharable device and sharable software queue.
(Three dummy entries at beginning.)

In order of queue number. The Wait State Code in the JBTSTS entry is used as the index into this table.

0	0
-1	0
-2	0
-3	Transfer Table Adr.
-4	"
-5	"
-	"
-	"

LH contains the queue number, always same as the negative of the relative position of the entry.

RH contains the address of the Transfer Table to be used to remove a job from the corresponding sharable resource queue.

Defined in SCHED.

Referenced in SCHED.

BUFFER RING

Contains pointers and storage area for I/O data.
Set up by user programs as needed.

	File Status Bits
U/Buf Size	Next Buf Adr
Bookkeeping	Word Count, N
N data words	
Unused Area (S-N)-1 words	

File Status Bits, First Word										
Bit	Meaning									
18	Improper Mode									
19	Device Detected Error									
20	Data error, e.g., checksum failure									
21	Block too large									
22	End of file									
23	Device is actively transmitting or receiving data.									
24-29	Device dependent parameters									
30	Synchronous input. Stop device after each buffer is filled									
31	Word count supplied by user for output. (Monitor normally computes word count)									
32-35	Data mode									
Second Word										
Bit	Meaning									
0	"Use" bit - U. Set as follows:									
	<table border="1" style="margin-left: auto; margin-right: auto;"> <thead> <tr> <th>I/O</th> <th>In Progress</th> <th>Finished</th> </tr> </thead> <tbody> <tr> <td>Input</td> <td style="text-align: center;">0</td> <td style="text-align: center;">1</td> </tr> <tr> <td>Output</td> <td style="text-align: center;">1</td> <td style="text-align: center;">0</td> </tr> </tbody> </table>	I/O	In Progress	Finished	Input	0	1	Output	1	0
I/O	In Progress	Finished								
Input	0	1								
Output	1	0								
1-17	Size of buffer, not counting first two words									
18-35	Address of second word of next buffer of ring									

Third Word	
Bit	Meaning
0-17	Depends on type of device and mode
18-35	Word count. Normally computed by monitor for the device being used.

Buffer Ring Header

Contains information by which user programs access their I/O buffers.

JBFADR	U	Current Buffer Address
JBFPTR		Byte Pointer
JBFCTR		Byte Count

Labels defined in S.MAC. Label values are relative to the beginning of the buffer ring header.

JBFADR	Bit 0	IOUSE bit -- Will be 1 if buffer ring has been set up but not referenced.
	Bits 18-35	Address of second word of the buffer currently available to the user program.
JBFPTR		Byte pointer for user to access next byte of buffer with ILDB or IDPB.
JBFCTR		Number of bytes remaining, available to user.

Notes:

1. Ring header must be set up within user program.
2. Values are initialized by each INPUT or OUTPUT UVO, for the buffer made available to the user.
3. User program must keep JBFPTR and JBFCTR up to date as it uses the buffer.

CIPWT . Clock Request Queue

Allow a monitor routine to be run after a specific time interval.
One entry for each job number plus three more.
Position of entry in table is of no importance.

CIPWT

	Routine Adr.	Data	Count Down
	"	"	"
	"	"	"
	"	"	"

0 1 18 23 35

Data

Each Count is decremented by Clock Routine. When any countdown reaches 0, the contents of Bits 18-23 are put into bits 30-35 and bit 0 into bit 29 of AC TAC, and a PUSHJ is done to the address in bits 1-17.

CLOCK is a 36 bit byte pointer to the highest in-use entry. When an entry is eliminated from the table, the last entry is copied into its place, and CLOCK is decremented.

Defined in COMMON.

Referenced in CLOCK1, DISSER, DTASRN, DTCSRN, MTASRX, MTASBX, MTCSR6, PTYSRF, SCNSRF, UUOCON.

COMTAB Commands Table

Specifies legal command names. There are corresponding entries in DISP which give routine adr and legality conditions for each command.

One entry for each monitor command.

Position in table is of no significance, but COMTAB and DISP entries must be in corresponding positions.

SIXBIT	/	CMD1	/
SIXBIT	/	CMD2	/
SIXBIT	/	CMD3	/
	-		
	-		
	-		

Defined in COMCON (local symbol)

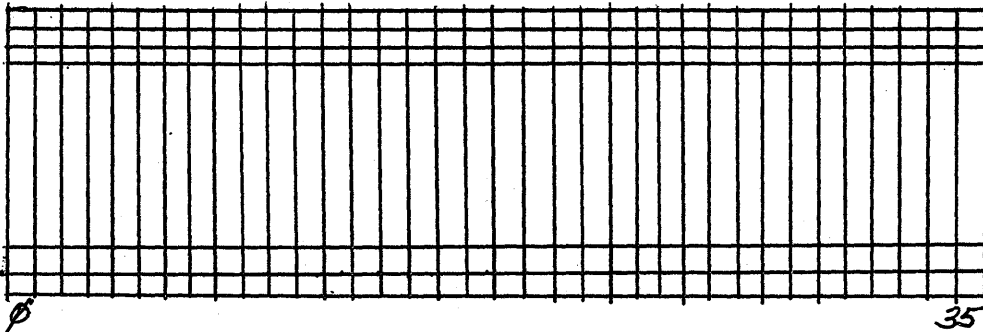
Referenced in COMCON.

CORTAB

Indicates which blocks of core are available for assignment, e.g., to swap a job in or to allow a job to expand.

256 entries of 1 bit each, i.e., 1 for each possible 1K block of core.

Bit N of table corresponds to Nth 1K block. Bit N is 1 if that 1K is not available.



Defined in COMMON.

Referenced in CORE1, SYSINI.

Symbol CORBLK = number of bits in this table.

DEVDSP Device Dispatch Table

Table of JRST's to routines that perform various device dependent functions. For each device, corresponding table entries go to routines to perform a specific function.

The first six entries are present in each device service routine. Tables which contain the additional entries are referred to as long dispatch tables.

Label of entry (Value Relative to DEVDSP)	Relative Adr	Function
DINI	-2	Device and service routine initialization
DHNG	-1	"Hung device" action
DRL	0	Release (Table Base Adr)
DCL, DCLO	1	Close, Close output
DOU	2	OUTPUT Operation
DIN	3	INPUT operation
DEN	4	ENTER operation
DLK	5	LOOKUP operation
DDO	6	Dump Mode output
DDI	7	Dump Mode input
DSO	10	USETO operation
DSI	11	USETI operation
DGF	12	UGETF operation
DRN	13	RENAME operation
DCLI	14	Close input - dump mode
DCLR	15	Call AC, SIXBIT/UTPCLR/
DMT	16	MTAPE operation

LONG TABLES ONLY

Labels for table entries, relative to the base address, are defined in S.MAC.

The actual tables are defined in the device service routines.
The base address of the device dispatch table is contained in the
corresponding Device Data Block, in RH of DEVSER.

Referenced in SYSINI
 UUOCON
 DISSER

The device dispatch table labels are normally indexed by AC DSER,
which must contain the base address of the appropriate table.

The dispatch is usually performed by a

PUSHJ PDP, Dxxx (DSER)

where Dxxx represents the label of the appropriate table entry.

Before attempting to dispatch to any of the long dispatch table
entries, the monitor checks the DVLNG bit of DEVMOD in the Device
Data Block for that device.

Device Data Block

Contains information needed to perform I/O operations.

One such block for each device or, in the case of multi-file devices, one for each file.

Number of entries in the DDB varies with the device.

The label of a DDB entry is defined as the location of that entry relative to the beginning of the block. Such labels must be indexed by an AC containing the address of the beginning of the DDB. AC DEVDAT is normally used for this purpose in the monitor.

DEVNAM	Physical Device Name *	Device Number *
DEVCHR	See bit definitions	See bit definitions
DEVIOS	See bit definition	See bit definitions
DEVSER	Adr of next DDB	Adr of device dispatch tbl
DEVMOD	See bit definitions	See bit definitions
DEVLOG	Logical Device Name*	
DEVBUF	Output Ring Hdr Adr	Input Ring Hdr Adr
DEVIAD	P	Current input buffer adr
DEVOAD	P	Current output buffer adr

-Additional words defined and used by some device service routines -

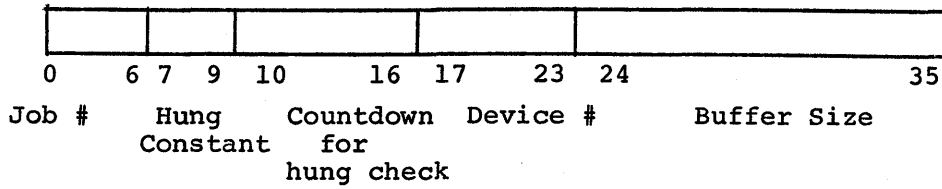
*in SIXBIT code

DEVIAD and DEVOAD point to the buffer currently available to the interrupt routine.

DEVIAD } Bits 14-17 are index register PROG
 DEVOAD }

Device Data Block (cont.)

DEVCHR byte definitions



Byte Pointer Labels, defined in COMMON

PJOBN	Bits 0-6	Job Number of job using this device
PDVTIM	7-9	Hung Constant
PDVCNT	10-16	Countdown for hung device check
PUNIT	17-23	Device Number

Notes:

1. Hung constant = 0 means device cannot be hung. Hung constant of N means device is considered hung after 2^{N-1} seconds without an interrupt.
2. The buffer size in DEVCHR determines the size of buffers set up by the monitor for this device.

DEVIOS Bit Definitions

Device Dependent Bits	I O E N D	I O		I F O S T	I O B E G	I O W	I O I M P M	I O D E R R	I O D T E R	I O B K T L	I O D E N D	I O A C T	I O B O T	I O T E N D	I O P A R	Mag Tape Den- sity	I O N R C K	I O C O N	I O W C	Data Mode (\emptyset - 17)
\emptyset	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26		29	30	31	35

Bit	Label	
12	IOEND	Service routine has transmitted last data
13	IO	Out = 1; In = \emptyset
14		Spare bit
15	IOFST	Next item will be the first item of a buffer
16	IOBEG	Virgin device
17	IOW	I/O Wait
18	IOIMPM	Improper mode detected e.g., write on write-locked device
19	IODERR	Device error, e.g., search error
20	IODTER	Data error, e.g., parity error
21	IOBKTL	Block too large or block # too large
22	IODEND	Data end encountered (for user program)
23	IOACT	Device active
24	IOBOT	Beginning of mag tape
25	IOTEND	End of mag tape
26	IOPAR	1 means write even parity on mag tape
27-28		Mag tape density $\emptyset\emptyset$ = installation standard $\emptyset 1$ = 200 BPI $1\emptyset$ = 556 BPI 11 = 800 BPI
29	IONRCK	Read with no reread check
30	IOCON	1 means stop device between each buffer transferred
31	IOWC	Don't compute word count on OUTPUT

DEVMOD bit definitions

D	D	D	D	T	T	T	D	D	D	D	D	D	D	D	D	A	A																						
V	V	V	V	T	T	T	V	V	V	V	V	V	V	V	V	S	S																						
D	D	C	L	Y	Y	Y	D	L	P	P	P	D	M	T	D	I	O	S	S																				
I	S	D	P	A	U	B	I	N	T	T	T	A	T	T	I	N	U	C	P																				
R	R	R	T	T	S	I	S	G	P	P	R	A	A	Y	R		O	R																					
I				E	U												N	G																					
0		3		6		9		12		15		18		21		24		27		30		33		35															

Bit	Label	Meaning =1
0	DVDIRI	DECTape directory is in core
1	DVDSK	Device is a disk
2	DVCDR	Device is a card reader
3	DVLPT	Device is a line printer
4	TTYATC	This TTY attached to a job
5	TTYUSE	TTY DDB in use (as user console)
6	TTYBIU	TTY DDB in use (as console or I/O device)
7	DVDIS	Device is a display
8	DVLNG	Device has a long device dispatch table
9	DVPTP	Device is a paper tape punch
10	DVPTR	Device is a paper tape reader
11	DVDTA	Device is a DECTape
13	DVMTA	Device is a mag tape
14	DVTTY	Device is a TTY
15	DVDIR	Device has a directory
16	DVIN	Device can do input
17	DVOUT	Device can do output
18	ASSCON	This device has been assigned by a console
19	ASSPRG	This device has been assigned by a program (i.e., INIT done)
20-35		If data mode n is legal, bit (35-n) is set

NOTES:

1. Labels for DDB's are defined in S.MAC. Each device service routine contains a DDB for that device. For devices with more than one DDB, the DDB's are set up dynamically as needed or at system initialization time. The new DDBs are modeled after the one contained in the device service routine.
2. All DDB's in the system are linked together so that routines can scan through the entire group if necessary. Location DEVLST in COMMON contains the address of the first DDB. Then each DDB contains the address of the next in LH of DEVSER. This link is 0 at the end of the chain.
3. Each DDB in use by a job is linked to that Job. The Job # is in bits 0-6 of DEVCHR. Also the DDB address is in the Job Device Assignment table in the Job Data Area for that job.

Disk Access Table - Level C

Provides information to allow simultaneous access of a disk file by several users.

There is one access table for each disk file being accessed at any time. They are dynamically set up and linked as they are needed.

ATPP
 ATNAME
 ATEXT, TBITS
 TDRV, ATLINK

Project #	Programmer #
File Name	
Extension	File Status Bits
Drive for current file	Adr of next Access Table

File Status Bits (RH TBITS)

W	D	R	A	A	R	Count of Readers
T	T	T	T	T	T	
B	B	B	I	C	T	
I	I	I	N	L	B	
T	T	T	D	O	I	
18	19	20	21	22	23	24

Bit	Label	
18	WTBIT	File is being created
19	DTBIT	Delete when through reading
20	RTBIT	File has been renamed
21	ATIND	Name is already in directory
22	ATCLO	Output close to be done
23	RWTBIT	File is being read and written, but not created (Update mode).

All labels above are local symbols in DSKSER.

Disk Device Data Block (Level C disk service)

Contains information needed to access a specific disk file.

There will be one DDB for each open disk file of each user.

The value of each label is the position of the word relative to the beginning of the DDB.

DEVNAM	D S K*	
DEVCHR	See bit definitions	
DEVIOS	See bit definitions	
DEVSER	Adr of next DDB	Adr of DSKDSP
DEVMOD	See bit definitions	
DEVLOG	Logical device name*	
DEVBUF	Output ring hdr adr	Input ring hdr adr
DEVIAD	See bit definitions	P Current input buffer adr
DEVOID	See bit definitions	P Current output buffer adr
DEVFIL	File name*	
DEVEXT	Extension*	Block # of RIB
DEVPPN	Project #	Programmer #
DEVNUM	# of current drive	# of inited drive
DEVACC	Access Table Adr	Current DDB Pointer Adr
DEV CNT	Rel. Block # within UFD	Number blocks in file
DEVBLK	Logical Block # of RIB to read next	Rel. loc of next pointer in that block
DEV BKO	Logical Block # of RIB	Relative Location
DSKBUF	Negative size of Read/Write	Buffer in user area
DSK CNT	Logical Block # for I/O	Error count or Error bit
SETCNT	Block # of UFD RIB	Data Block to be used next
PTR1	Ck Sum x ↓ Ch Sum X+13 ₈	Adr x ↓ Adr x+13 ₈

*SIXBIT code

DEVIAD & DEVOID Bits 14-17 are index register PROG
 DEVEXT RH is Block # of beginning of RIB
 DEVBLK Logical Block # and location within that block of the RIB section to be read in next, at PTR1

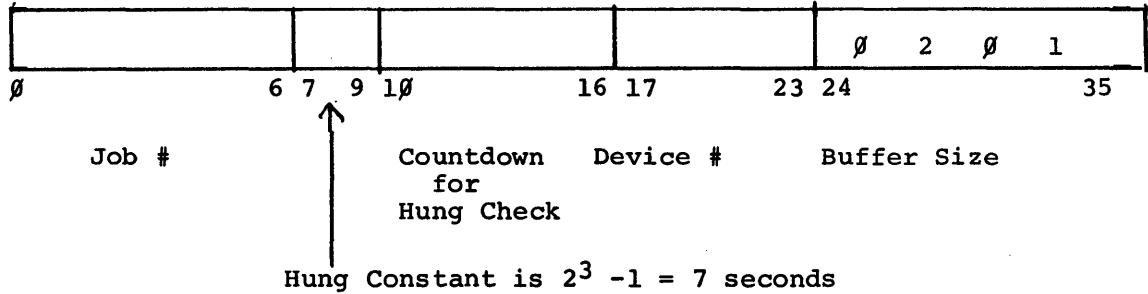
Disk Device Data Block (cont.)

DEVBKO Logical Block # and location within that block
 now in core at PTR1

DEVPPN Project-programmer # from the last LOOKUP, ENTER,
 or RENAME.

SETCNT RH is a relative block number within the file to be
 read or written next.

DEVCHR byte definitions



Byte Pointer Labels:

Bits	Pointer	
0-6	PJOBN	Job #
7-9	PDVTIM	Hung Constant
10-16	PDVCNT	Countdown
17-23	PUNIT	Device #

Disk Device Data Block (cont.)

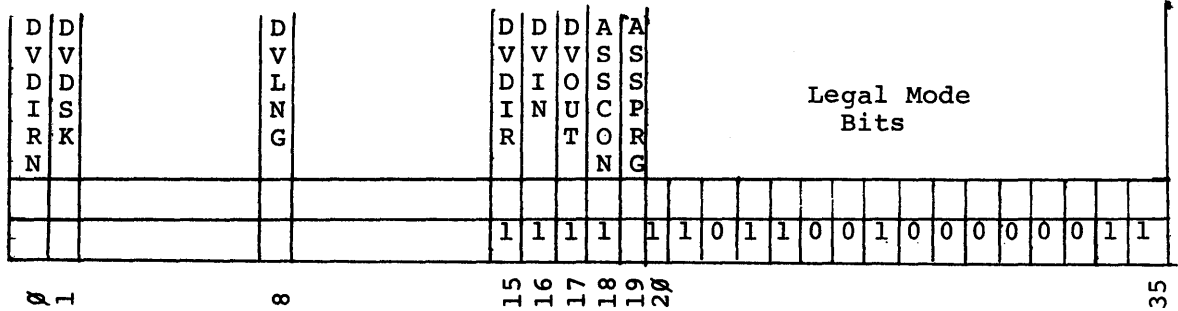
DEVIOS bit definitions

	P R C H G	A U F L G	D A F L G	N C T R L C	U B F U	N O R E L B	U B F S	N M P	L I R	I O		I O F S T	I O B E G	I O W	I O I M P M	I O D E R R	I O D T E R	I O B K T L	I O D E N D	I O A C T			I O W C	Data Mode
0	4	5	6	7	8	9	10	11	12	13		15	16	17	18	19	20	21	22	23			31	35

Bit	Label	Meaning if 1
4	PRCHG	Protection change only is permitted
4	UFDPR	LOOKUP, ENTER: UFD READ/WRITE protected
5	AUFLG	Altering a UFD (process must be interlocked)
6	DAFLG	Using the non-reentrant disk allocation code
7	NCTRLC	Disables the control C at critical times
8	UBFU	Using buffer in user area
9	NORELB	Do not relocate buffer on I/O
10	UBFS	Using buffer in free storage
11	NMP	No more input pointers
12	LIR	Last input requested
13	IO	Output transfer (Ø for Input)
14		Reserved
15	IOFST	Next interrupt will be first of buffer
16	IOBEG	New file
17	IOW	Job in I/O Wait State
18	IOIMPM	Improper mode
19	IODERR	Device error
20	IODTER	Device data error
21	IOBKTL	Data Block Too Large
22	IODEND	End of file (to user)
23	IOACT	Device active (expect interrupt)
	IOWC	Don't compute word count.

Disk Device Data Block (cont.)

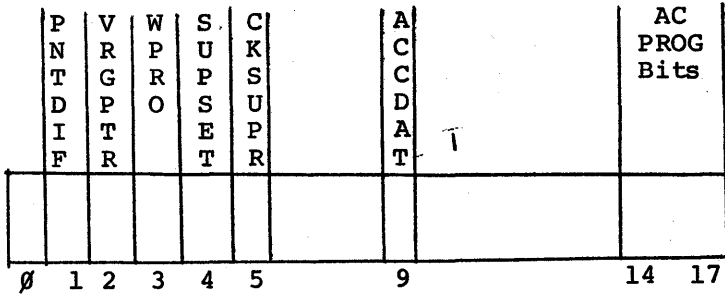
DEVMOD bit definitions



Bit	Label	Description
0	DVDIRN	Directory is in core
1	DVDSK	Device is a disk
8	DVLNG	Long dispatch
15	DVDIR	Directory type device
16	DVIN	Device can do input
17	DVOUT	Device can do output
18	ASSCON	Assigned by console command
19	ASSPRG	Assigned by program (INIT)

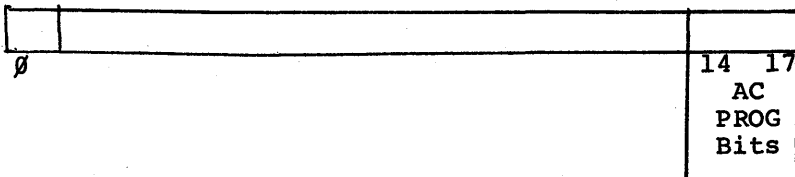
Disk Device Data Block (cont.)

DEVOAD bit definitions



Bit	Label	Meaning
0		Buffered output waiting
1	PNTDIF	Pointers in core differ from pointers on disk
2	VRGPTR	Virgin pointers (i.e., pointers have never been written on disk)
3	WPRO	LOOKUP done; file is in write protect
4	SUPSET	SUPER USET I/O pending
5	CKSUPR	Checksum suppress (reading dump file in non-dump mode)
9	ACCDAT	Access data today's date

DEVIAD Bit definitions



Bit	Meaning
0	Buffered input request waiting

DISK FILE DIRECTORY (Level C)

Tells the location of the Retrieval Information Block for each file belonging to a specific user.

There is one two word entry for each file belonging to that user.

- Structure of Directory Data Block -

File Name 1	
Extension 1	Location of RIB
File Name 2	
Extension 2	Location of RIB
File Name 3	
Extension 3	Location of RIB

File Names and Extensions are in SIXBIT.
Location of RIB is the logical Block # of
the RIB.

Each directory is, itself, a file - having a RIB and one or more data blocks.

There are up to 64 two word entries on each data block of the directory. If the directory has more than one data block, they are not linked. Instead there is an entry in the directory's RIB pointing to each data block of the directory.

DISP Command Dispatch Table

Specifies routine address and legality bits for each monitor command.
 Entries correspond to command names in COMTAB.

One entry for each command.

Position in table is of no significance, but DISP entry and COMTAB entry must be in same relative position.

N	N	N	N	N	N	I	C	N	N	N	N	T	T	N	P		E			
O	O	O	O	O	O	N	M	O	O	O	O	T	T	O	L		R			
C	J	L	A	T	R	C	W	D	I	C	P	Y	Y	M	S		R			
O	O	O	C	R	U	O	R	A	N	R	E	R	R	E	X		R			
R	B	G	T	A	N	R	Q	T	C	L	R	R	N	S	P		F			
E	N	I		N				E	K	F		U	C	S	N		L			
																	G			
																	Adr for Cmd 1			
																	Adr for Cmd 2			
																	Adr for Cmd 3			
																	=			
0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18		35

Bit	Label	Meaning
0	NOCORE	No core needed for command
1	NOJOB	No job number needed for command
2	NOLOGIN	Job need not be logged in
3	NOACT	Command must wait until job devices not active
4	NOTRAN	Command must wait if swapping transfer in progress
5	NORUN	Job must not be running — ↑C required
6	INCOR	Job must be in core if it has core
7	CMWRQ	Requeue job after command wait
8	NODATE	Don't print date during job interrogation
9	NOINCK	No check for job initialization (JNA=0)
10	NOCRLF	No printing of CR-LF
11	NOPE	No printing of period
12	TTYRNU	Set TTY to user and start job
13	TTYRNC	Keep TTY in monitor mode and start job
14	NOMESS	No command response ever
15	PLSXPN	Core must be expanded if still compressed
16		
17	ERRFLG	Command Error

Bits 0-6 and 15 checked before dispatching to command.

Bits 7-14 and 17 checked after return from command.

Defined in COMCON (all symbols local).

Referenced in COMCON.

Fragment Table (Level C Disk Service)

Tells where the different fragments of a swapped out segment are stored.

There is one fragment table for each swapped-out segment that could not be written in one contiguous block of disk space.

Number 1K blocks	Disk Address
"	"
"	"
Ø	Link or Ø

A zero entry indicates end of table.

An entry with LH = Ø, RH ≠ Ø is a linkage to a continuation of the table. The RH is the core address where the table is continued.

Fragment tables are set up as needed, by routine FULL in SCHED. Space is allocated in blocks of four words. If table is longer, another four words are used.

INTTAB Interrupt Routine Table

Contains descriptive information about each interrupt routine.
One entry, of two words, for each interrupt routine.

Position of an entry in the table is of no significance.

INTTAB:

—	PTY Entry	—
—	CDR Entry	—
	.	
	.	
—	Clock Entry	—

Format of each entry:

	Nr DDB's	PI Channel	Int. Rout. Adr.
Ø	1 8	9 17	18 35

Bit Ø = 1 for any type of DEctape routine.

DDB Length or Ø	DDB Adr.
Ø	17 18 35

Bits Ø-17 contain length of DDB for devices with multiple units.

Defined in COMMON.

Referenced in ONCE,COMMON.

Notes:

1. INTTAB entries are set up by the ASGINT and ASGSV1 macros in COMMON according to parameters specified in CONFIG.
2. Table is used by INTLOP in ONCE, to build the interrupt routine chain for each channel. Also, used in the long ONCE dialog to print out I/O configuration.
3. Used by Once Only code in COMMON to set up multiple Device Data Blocks.

JBTADR

Tells core address and length for each segment in core.

One entry for each job number and each high segment number.

Indexed by job number or high segment number.

Job 0			
1	Length -1	Core Address	
2	"	"	"
-			
-			
High Seg 64	"	"	"
65	"	"	"
-			
-			
-			

For low segments the length -1 is the protection setting and the address in core is the relocation factor.

For high segments, protection and relocation are computed from the table entries.

Defined in COMMON.

Referenced in	CDPSER	CLOCK1	COMCON	COREL
	DISSER	DTASRN	DTCSRN	ERRCON
	MTCSR6	SEGCON	SYSINI	UUOCON
	DSKSER	SCHED	MOVIE	

Note:

The entry for a job which is swapped out, or being shuffled, is zero.

JBTCHK

Used to assure correct read-in of swapped segments.

One entry for each job number and high seg number.

Indexed by job number or high segment number.

Job 0	
Job 1	Checksum 1
Job 2	Checksum 2
High Seg 64	Checksum 64
High Seg 65	Checksum 65

Each entry contains the checksum of the corresponding segment when that segment is swapped out.

Defined in COMMON.

Referenced in SCHED.

JBTDIR (PRJPRG)

Tells project-programmer number for each job, directory name for each high segment.

One entry for each job number and high segment number.

Indexed by job number or high segment number.

Job 0	Project Number	Programmer Number
Job 1	" "	" "
Job 2	" "	" "
-		
-		
-		
High Seg 64	Directory Name	
High Seg 65	" "	" "
-		
-		

Defined in COMMON.

Referenced in SEGCON UUOCON
 CLOCK1 DSKSER
 COMCON

JBTKCT

Contains total of (run time) x (job size) for each job.

One entry for each job number.

Indexed by job number.

↓

Job 0	
Job 1	KCT 1
Job 2	KCT 2
Job N	KCT N

KCT represents a total computed by adding the job's current size (in 1K blocks) to the previous total each clock tick. Hence it is in units of "kilo-core - ticks."

Defined in COMMON.

Referenced in CLOCK1, COMCON, SEGCON, UUOCON.

JBTNAM (JBTPRG)

Tells file each segment came from.

One entry for each job number or high seg number

Indexed by Job number or High seg number.

Job 0
Job 1
Job 2
Job 3

High Seg 64
High Seg 65

File Name 1
File Name 2
File Name 3
-
-
-
File Name 64
File Name 65
≡

Defined in COMMON

Referenced in SEGCON
COMCON
UUOCON

JBTPRV Job Privilege Bits

Tells privileges allowed each job.
 One entry for each job number.
 Indexed by job number.

	P V N C T C	P V T R P S	P V S P Y A	P V S P Y M
Job 0				
Job 1				
Job 2				
-				
-				
-				
-				
Job N				

0 15 16 17

Bit	Label	Use
0	PVNCTC	Disable Control C.
1-14		Reserved for expansion by DEC.
15	PVTRPS	Job allowed to use TRPSET UUO.
16	PVSPYA	Job allowed to spy at all of core using SPY/PEEK UUO's.
17	PVSPYM	Job allowed to spy at monitor using SPY/PEEK UUO's.
18-35		Reserved for special customer defined privileges.

Defined in COMMON.

Referenced by COMCON, UUOCON, DSKSER, SCHED.

Bits are set by LOGIN from ACCT.SYS File as modified by customer for his particular installation.

JBTQ Job Queues Table

Contains the job queues. Each queue is an ordered list of job numbers, corresponding to all the jobs in some particular status.

The index of each entry, relative to location JBTQ, is the (positive) job number, or (negative) queue number which that entry represents.

QUEUE NUMBER	↑	-3	LAST JOB IN QUEUE	FIRST JOB IN QUEUE
		-2	"	"
		-1	"	"
		JBTQ	∅	
JOB NUMBER	↓	1	# OF PREV JOB	# OF NEXT JOB
		2	"	"
		3	"	"
		4	"	"
		5	"	"

Each queue is an ordered list of job numbers. In JBTQ, each queue is represented by a forward and backward linked ring of table entries. Each ring begins and ends with a "queue header" entry at the position equal to the negative queue number. A job number in the queue is represented by the entry at the position equal to the job number. This entry contains pointers to the preceding entry and the following entry, thus establishing a unique position for that job number within the queue.

Defined in COMMON.

Referenced in SCHED.

Notes:

1. Every job number will be in one and only one queue.
2. It is possible for a queue to contain no job numbers. In this case, the queue header entry contains a pointer to itself, the negative queue number, in both halves.

JBTSGN

Tells which high segment, if any, each job is using.

One entry for each job number.

Indexed by job number.

	S P Y S E G	S H R S E G	U W P O F F	M E D D L E	C O R C O U N T		
Job 0							High Seg # for this job
Job 1							
Job 2							
-							
-							
-							
Job N							
	0	1	2	3	4		18 36

Bit	Label	Meaning
0	SPYSEG	High segment is physical core (see SPY UUO)
1	SHRSEG	High segment is sharable
2	UWPOFF	User mode write protect is off
3	MEDDLE	User had meddled with sharable program (see comments)
4	CORCNT	High segment's In-Core Count has been incremented.
18-35		If SPYSEG set, highest physical address user may see. If no high segment, 0. Otherwise, high segment # associated with this job.

Defined in COMMON.

Referenced in	COMCON	UUOCON
	ERRCON	MOVIE
	SEGCON	

JBTSGN (cont.)

1. If the MEDDLE bit is set, program cannot trust itself to turn off UWP or change high seg core assignment.

Meddling means

1. START N, or D Command has been executed.
 2. RUN UWO with greater than 1 starting increment.
 3. GETSEG UWO.
 4. High seg is physical core (SPY UWO).
-
2. The SHRSEG bit is also kept in the JBTSST entry for that high segment.

JBTSTS

Contains status information about each job and high segment.

One entry for each job number and each high segment number.

Indexed by Job number or High Segment number.

Job 0		
Job 1	Status Bits	Quantum Run Time left
Job 2	" "	" " " "
-		
-		
-		
High Seg 64	Status Bits	In Core Count
High Seg 65	" "	" " "
High Seg 66	" "	" " "
High Seg 67	" "	" " "
-		
-		
-		

See Bit definitions on following pages

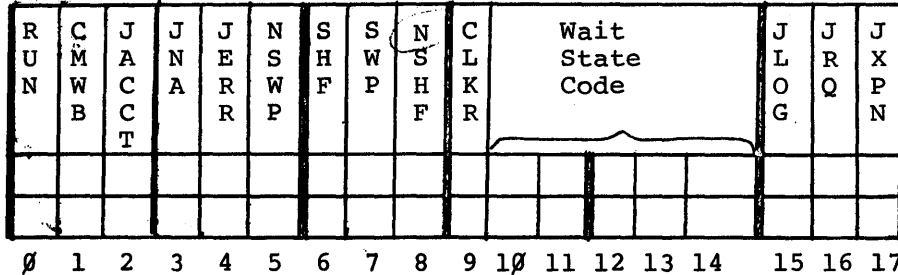
Defined in COMMON.

Referenced in

CLOCK1 SEGCON
 COMCON SYSINI
 CORE1 UUOCON
 DISSER DSKSER
 DTASRN SCHED
 DRCSRN MOVIE
 NULSEG
 SCNSRF

JBTSTS (cont)

LH of each Job entry in JBTSTS



<u>Bit number</u>	<u>Label</u>	<u>Meaning</u>
0	RUN	Set if user wants job to run.
1	CMWB	Job in Command Wait (e.g., swapped out job has typed command which requires core)
2	JACCT	Privileged system CUSP which cannot be interrupted.
3	JNA	This job number is assigned.
4	JERR	Monitor detected error has occurred.
5	NSWP	Job is not to be swapped.
6	SHF	Monitor is waiting to shuffle or swap out this job.
7	SWP	1 if job swapped out or in transit.
8	NSHF	Job is not to be shuffled.
9	CLKR	Job has clock request in.
10 - 14	Wait State Code	Specifies transfer table to requeue job for current conditions. See AVLQTB, QBITS.
15	JLOG	Job logged in.
16	JRQ	Job has changed state and must be requeued at clock level before rescheduling can take place.
17	JXPN	Job must be swapped out because it is expanding and there is not enough room in core.

NSWP and NSHF are set only by display service routine (4S72 Monitor).

JBTSTS (cont)

LH of each High Seg entry in JBTSTS

S N A	S H R S E G				N S W P		S W P		Access Privilege Bits								
									Owner			Project			Universe		
0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17

<u>Bit</u>	<u>Symbol</u>	<u>Meaning</u>
0	SNA	This High Seg Number assigned.
1	SHRSEG	Sharable Segment.
5	NSWP	This High Seg is not to be swapped.
7	SWP	1 if segment swapped out or in transit.
9-17		Access privilege bits - same as disk.

Note: SHRSEG bit is also kept in JBTS GN.

JBTSWP

Contains information used when swapping segments.

One entry for each job number and high seg number.

Indexed by job number or high seg number.

	F R G S E G	S W P C L R			
Job 0	In-Core Protect		Out-Core Size		In-Core Size
Job 1	Time or Disk Adr		" " "		" " "
Job 2			" " "		" " "
Job 3					
-					
-					
High Seg 64			" " "		" " "
High Seg 65			" " "		" " "
	0 1	18	27		35

LH contains In-Core Protect Time in Jiffies if job is not swapped out.

If job is swapped out and not fragmented, FRGSEG = 0.

Bits 1-17 = Disk Address where swapped out.

If job is swapped out and fragmented, FRGSEG = 1.

Bits 1-17 = Core Address of Fragment Table.

SWPCLR Bit 18 = 1 Job Data Area should be cleared after swap in.
Set by 140 restart.

Defined in COMMON

Referenced in CORE1
 SEGCON
 UUOCON
 SCHED

JBTSWP (cont)

NOTES:

1. Fragment Table documented separately.
2. For swapped out segments, Out-Core Size is the size of the area in which the segment is written on the swapping device, in 1K blocks; In-Core Size is the size of the core area which it should be read into when it is swapped in. These are usually, but not always, the same. One case in which they would be different is when a segment is expanding in size.

JBTUFD Job UFD Table

Contains location of UFD for each job.

One entry for each job number.

Indexed by job number.

Job #	Disk Drive #	UFD RIB Block #
1	"	"
2	"	"
3	"	"
N		

Note: This table was added to Monitor 4S72 by Patch AB. It is not included in the 5 series monitors.

Associates a device or file with each active channel in a user job, and tells which UUO's have been done on that channel.

The JDA is a part of each Job Data Area. Each JDA contains 16 entries corresponding to the 16 "software channels" of a user job.

Indexed by channel number

	I N I T B	I B U F B	O B U F B	L O O K B	E N T R B	I N P B	O U T P B	I C L O S B	O C L O S B	I N B F B	O U T B F B	S Y S D E V	
Channel 0													Device Data Blk Adr
1													"
2													"
-													
-													
17													
	0									11		18	35

<u>Bit</u>	<u>Label</u>	<u>Meaning</u>
0	INITB	INIT or OPEN has been done.
1	IBUFB	Input ringheader specified (by INIT).
2	OBUFB	Output ringheader specified (by INIT).
3	LOOKB	A LOOKUP has been done.
4	ENTRB	An ENTER has been done.
5	INPB	An INPUT has been done.
6	OUTPB	An OUTPUT has been done.
7	ICLOS B	An input CLOSE has been done.
8	OCLOS B	An output CLOSE has been done.
9	INBFB	An input buffer ring has been set up.
10	OUTBFB	An output buffer ring has been set up.
11	SYSDEV	This is the system tape device.
18-35		Adr of Device Data Block for I/O on this channel.

NOTE: The JDA for the current job is kept at USRJDA in COMMON, where it is protected from access by the job. USRJDA is referenced by CLOCK1, COMCON, CORE1, and UUOCON.

Storage area for items of interest to both the monitor and the user.

There is one Job Data Area for each job which is running. It occupies the first 140 locations of the job's core area, and is swapped out along with the job.

<u>Relative Location in octal</u>	<u>Label</u>	
0-17	JOBAC	User AC's during UUU (16 words)
20-37	JOB DAC	Hardware AC's while job inactive (16 words)
40	JOB UUU	User UUU stored here
41	JOB 41	User UUU branch instruction
42	JOB ERR	Unused Error cnt for RPG
43	JOB ENB	Unused User APR trap flags
44	JOB REL	0 Length of low seg
45	JOB PD1	Push Down List (21 words)
72	JOB HCU	Highest I/O channel in use
73	JOB PC	Job PC when job inactive
74	JOB DDT	Unused Start Adr of DDT
75	JOB JDA	Job Device Assignment Table 16 words documented separately
114	JOB SDD	JOB DDT here on SAVE
115	JOB HRL	First free loc in high seg Length of high seg
116	JOB SYM	Symbol table pointer
117	JOB USY	Undefined symbol table pointer
120	JOB SA	First free loc in low seg-when loaded Program start adr.
121	JOB FF	Current first free location in low seg
122	JOB S41	JOB 41 here on SAVE
123	JOB EXM	Adr of last D or E command
124	JOB REN	Address for REENTER command
125	JOB APR	Branch loc on user enabled APR error
126	JOB CNI	APR conditions on APR trap
127	JOB TPC	PC stored here on APR trap
130	JOB OPC	Old PC stored here on START, DDT, REENTER, & STARTC commands
131	JOB CHN	Used for FORTRAN Job chaining
132	JOB FDV	DDB adr for FINISH command
133	JOB COR	Highest loc in low seg actually loaded low seg core assignment

continued on next page

Relative
Location
in octal

Label

134-136

Unused - 3 words

137

JOBVER

Programmer #

Version # of CUSP

140

JOBDA

First location not used by Job
Data Area.

Labels are defined in the file JOBDAT.MAC. The actual tables are included at the beginning of each user's area.

NOTE: Many of these words contain different values while a SAVE or GET is in progress, and therefore have several different labels. See a listing of JOBDAT for a complete list of these labels

Job Size - Queue Table

Specifies the queue for a job to be put into as a function of the job's size.

A Job Size - Queue Table is specified in the Transfer Table for Job - Size type queue transfers.

Job Size 1	Dest. Queue 1
Job Size 2	Dest. Queue 2
"	"
"	"
"	"
Ø	Ø

Job Size is in 1K units. Table must be in order of increasing Job Size.

Destination Queue is a queue number.

A zero entry indicates end of table.

See QSTAB in SCHED.

Referenced in SCHED (QJSIZ).

LINTAB (before Monitor 5.02)

Contains characteristics of each TTY line

One entry per line and Pseudo TTY line

Indexed by Line #

P T Y L I N	C T Y L I N	D I S L I N	D S D T L I N	H L F D P X	T T Y R M T	T 3 5	F U L T W X	Line #
								Ø
								1
								2
								N

Ø 1 2 3 5 6 14 15 18 35

<u>Bit</u>	<u>Label</u>	<u>Meaning</u>
Ø	PTYLIN	Pseudo TTY line
1	CTYLIN	Console TTY
2	DISLIN	Keyboard display
3	DSDTLN	Data set line
5	HLFDPX	Half duplex line
6	TTYRMT	Remote TTY line
14	T35	TTY has hardware tabs and formfeed
15	FULTWX	Self echoing line

Bit labels defined in SCNSRF

Defined in COMMON

Referenced in SCNSRF, PTYSRF

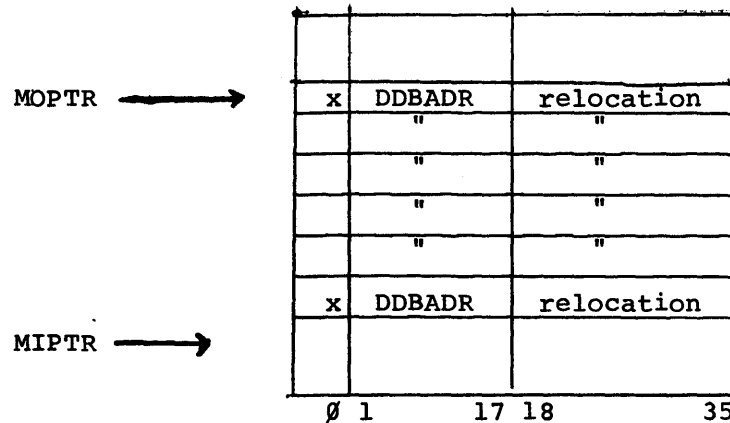
NOTE: LINTAB is set up according to parameter values in CONFIG.MAC. These values are set by the MONGEN dialog according to answers to questions about the TTY's.

MQUEUE (Level C Disk Service)

Tasks to be performed by monitor disk I/O routines.

One entry for each job #.

Ring structure. Two pointers are defined in DSKSER. MOPTR contains address of next task to be performed. MIPTR contains address of next available position in the queue.



Bit 0 = 1 for input
 = 0 for output

Bits 1-17 contain the absolute address of the DDB to be used for the I/O.

Bits 18-35 contain the relocation factor to be added to DSKBUF (DEV DAT) before doing I/O. Relocation is 0 if transfer to monitor core, contents of PROG if transfer to user core.

Defined in COMMON.

Referenced in DSKSER.

Only those locations after C(MOPTR) -1 and before C(MIPTR), considering wraparound at end of table, contain current information.

QBITS

Specifies the type of queue transfer to be performed for a job as a function of its wait state code.

There is one entry for each wait state code.

The value of the wait state code is the index factor for the table.

Wait State Code

Ø
1
2
-
-
-

Transfer Table Adr.
"
"
"
"
"

Each entry in QBITS specifies a transfer table to be used to requeue a job having the corresponding Wait State Code.

Defined in SCHED.

Referenced in SCHED.

Quantum Time Table

Specifies value to which a job's quantum run time is reset on certain types of queue transfers.

Quantum Time 1
Quantum Time 2
-
-
-
∅

Quantum time is in jiffies ($1/60^{\text{th}}$ of a second)

A ∅ entry indicates end of table.

See QQSTAB in SCHED.

Referenced in SCHED (QX2).

Queue Progression Table

Specifies the queue for a job to be put into as a function of the queue it is in.

A Queue Progression Table is specified in the Transfer Table for Link type queue transfers. (Ref. Transfer Table).

Source Queue 1	Destination Queue 1
Source Queue 2	Destination Queue 2
-	-
-	-
-	-
Ø	Ø

Each entry is a queue number.

A 0, 0 entry indicates end of table.

See QTTAB in SCHED.

Referenced in SCHED (QLINK).

REQTAB

Tells how many jobs require use of each sharable device.

Each entry referenced by its own label.

RNREQ	
WSREQ	
TSREQ	
STREQ	
AUREQ	
MQREQ	
DAREQ	
DTREQ	
DCREQ	
MTREQ	

Entry contains -1 if no job wants that resource. If a job uses the device, the entry is incremented to 0. Each additional job which asks for the device while it is in use increments the entry by one, and must be requeued to the corresponding sharable resource wait queue.

Defined in SCHED.

Referenced in	SCHED	CLOCK1	COMCON
	DTASRN	UOCON	DSKSER
	DTCSRN	MTCSR6	MTASRX
	MTASBX		

NOTES:

1. Entries in this table are tested and incremented simultaneously, as follows:

AOSE	XXREQ
PUSHJ	XXWAIT
-	
-	
-	

If the resource was available, the routine continues. Otherwise the job must be requeued to wait for it.

2. The first three entries are dummy entries and are never referenced.

3. Table AVALTB has entries corresponding to the entries in REQTAB.

RIB Retrieval Information Block (Level C)

Provides information needed by the disk service routine to read the various blocks of disk that make up a file.

Consists of four words plus one word per block of the file plus two linkage words for each extension block of the RIB.

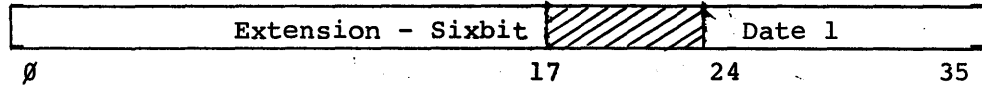
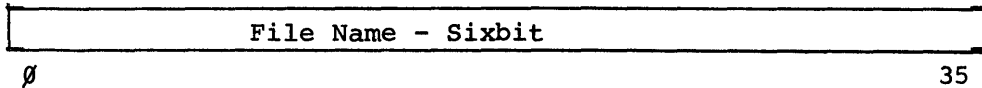
Entries are ordered as their blocks appear in the file.

Preamble	File Name (Sixbit)			
	Extension			Date 1
	Prot.	M	Time	Date 2
	Size		Programmer #	
	Ck Sum 1		Adr 1	
	Ck Sum 2		Adr 2	
Data Block Pointers				
	Ck Sum 122		Adr 122	
	Next Blk		Ø	
			This Blk	
Linkage				

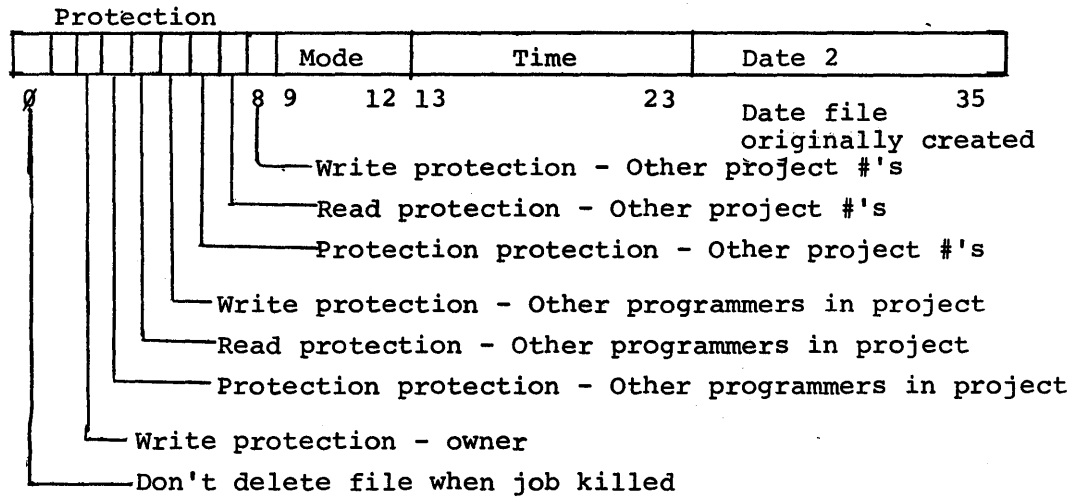
Data Block Pointers	Ck Sum 123		Adr 123	
	Ck Sum 124		Adr 124	
Linkage				
	Ck Sum N		Adr N	
	Ø		Prev Blk	
			This Blk	

RIB (cont.)

Preamble - 4 words which appear at beginning of the first block of retrieval information for each file.



Date file last referenced.
(See format following)

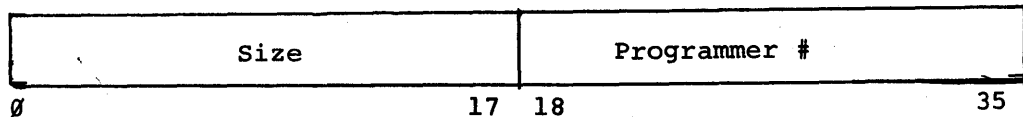


Mode is standard 4 bit data mode.

Time is the 24 hour time, in minutes, that the file was created.

Dates are 12 bit numbers computed as follows:

$$\text{Date} = [(\text{Year} - 1964) \times 12 + (\text{month} - 1)] \times 31 + (\text{day} - 1)$$



Size If file contains less than 2^{17} words, this is the negative of the word count for the file.
If file contains more than 2^{17} words, this is the positive block count.

RIB (cont)

File data block pointers

Check Sum	Address
∅	17 18 35

Address is the logical block # of the corresponding data block of the file.

Check sum is computed as 36 bit, end-around carry, sum, followed by 18 bit folded sum.

Linkages - 2 word blocks following the pointers on each block of retrieval information

∅	Previous Block #
Next Block #	This Block #

Previous Block # is ∅ for first block.

Next Block is ∅ for last block.

RTIME

Contains incremental run time for each job, i.e., run time since last RUNTIME command.

One entry for each job number.

Indexed by job number.

Job 0	
Job 1	Time 1
Job 2	Time 2
-	
-	
-	
Job N	Time N

Time is in units of clock ticks, or "jiffies", usually $1/60^{\text{th}}$ second.

Defined in COMMON.

Referenced in CLOCK1, COMCON.

Scan Table

Determines manner in which job queues are scanned by routine QSCAN in SCHED.

One entry for each queue to be considered.

Entries are in the order that the corresponding queues are to be considered

Queue #	Scan Code
"	"
"	"
"	"
Ø	

The Scan Code is the address of a routine in QSCAN. There are four possible values:

QFOR	Scan entire queue forward
QFOR1	Look at first entry only
QBAK	Scan entire queue backward
QBAK1	Scan queue backward except first entry.

Queues are scanned, in specified manner, in the order in which their entries appear in the table.

A zero entry terminates the table.

Examples in SCHED (ISCAN, OSCAN, SSCAN).

Referenced in SCHED (QSCAN).

SPCTAB Special Character Table (before Monitor 5.02)

Tells characteristics of special characters read in from TTY.

35 entries, corresponding to ASCII characters 000-040 and 175-177.

Position in table is same as binary value of the character for values 000-040 and value-135, for 175-177.

Binary value of Char.	S	S	B	S	F	E	Address of special action routine	
	P	P	R	P	C	C		
ACTN	A	O	E	H	S	H		
000	T	U	A	P	B	S		
001	N	T	K	O	R	U		
002			B	S	K	P		
003								
-								
-								
177								

Ø
1
2
3
4
5
6
7
8
17
18
35

Bit	Label	Meaning
Ø	SPACTN	Special action to be taken (routine address in RH)
2	SPOUT	Special handling on UO output
4	BREAKB	Break character
5	SPHPOS	Character affects horizontal position
6	FCSBRK	Break character in Full Character Set mode.
7	ECHSUP	Suppress echo of character itself
18-35		Address of routine for special action, if any.

Table entries are in order of the binary value of the characters which they represent. Entry for 175 follows entry for 040.

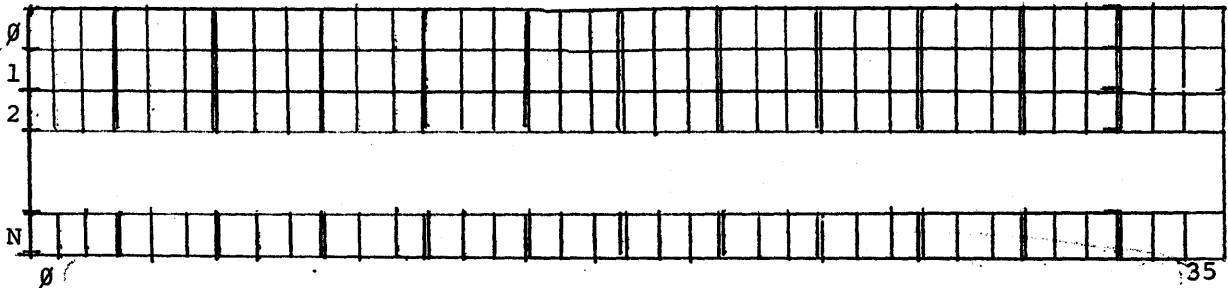
Defined in SCNSRF.

Referenced in SCNSRF.

Storage Allocation Table (Level C Disk Service)

Tells which blocks of a disk are in use.

Each disk device has its own file of storage allocation blocks named *SAT*.SYS. The file may consist of more than one block, if more than one are necessary to account for the blocks of the device.



The n^{th} bit of the table represents the n^{th} disk block, where $n=0, 1, 2, \dots$. Bit is 1 if block is use. All bits of a SAT block corresponding to nonexistent blocks of the device are set to 1.

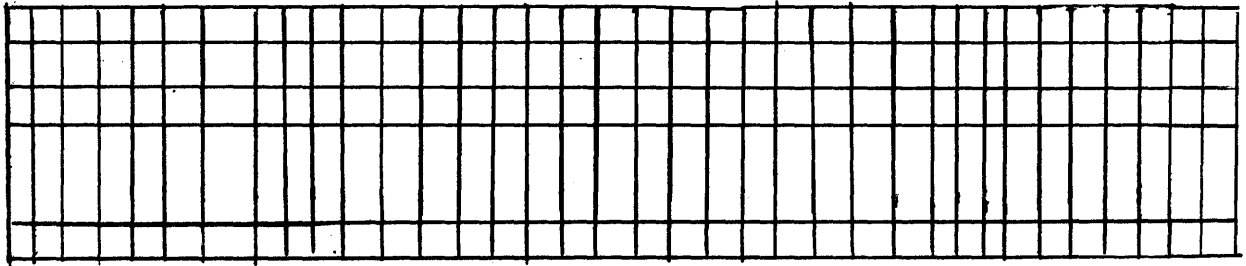
When a disk block is needed, the monitor searches the SAT for a free block. When a file is deleted the corresponding bits are cleared.

SWPTAB (Level C Disk Service)

Tells which blocks of swapping space are in use.

One entry, of 1 bit, for each 1K block of swapping space.

Each bit corresponds to a 1K block of swapping space. Bit=1, if that block is in use. Bit=0 if that block is available.

A large empty grid representing the SWPTAB table structure. The grid consists of 35 columns and 4 rows, forming a rectangular array of cells.

0

35

Defined in SCHED.

Referenced in DSKSER (GETBIT, CLRBIT).

Referenced indirectly in SCHED.

This table is referenced by the GETBIT and CLRBIT routines in DSKSER. A three word pointer table, SWPENT is set up just in front of SWPTAB for use by these routines. SCHED calls those routines to set and clear bits in the table, according to input parameters.

TRANSFER TABLE

Contains input parameters for Queue Transfer routine.

There is one such table for each different type of queue transfer.

PLACE	FUNCTION
QUANT	DEST

PLACE $\geq \emptyset$ Transfer to beginning of destination queue.
PLACE $< \emptyset$ Transfer to end of destination queue.
FUNCTION is the address of the QXFER routine to be used. It defines the type of transfer. There are three possible values:
1. QFIX - Destination queue specified in this table.
2. QLINK- Destination queue is a function of source queue.
3. QJSIZ- Destination queue is a function of job size.
DEST specifies the destination queue.
1. If FUNCTION is QFIX, DEST is the destination queue number.
2. If FUNCTION is QLINK, DEST is the address of a Queue Progression Table which specifies destination queue as a function of source queue.
3. If FUNCTION is QJSIZ, DEST is the address of a Job Size-Queue Table which specifies destination queue as a function of job size.
QUANT specifies the change to the job's quantum run time as follows:
If QUANT $< \emptyset$, no change to quantum run time.
If QUANT $> \emptyset$, reset the quantum run time as follows:
1. If FUNCTION is QFIX, set quantum run time to the value of QUANT.
2. If FUNCTION is QLINK or QJSIZ, QUANT is the address of a Quantum Time Table, which has entries corresponding to the entries in the table used to determine the destination queue. The entry in the same position as the selected destination queue is used to reset the quantum run time.

See BQFIX and following in SCHED.

Referenced in SCHED.

TTIME

Contains total run time for each job.

One entry for each job number

Indexed by job number

Job 0	
Job 1	Time 1
Job 2	Time 2
-	
-	
-	
Job N	Time N

Time is in units of clock ticks, or "jiffies", usually $1/60^{\text{th}}$ second.

Defined in COMMON

Referenced in CLOCK1, UUOCON, COMCON.

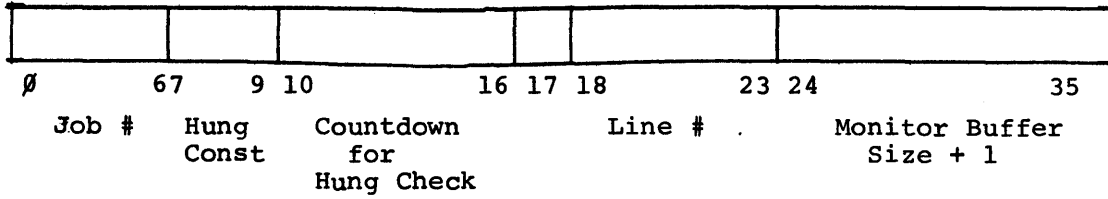
TTY Device Data Block (before Monitor 5.02)

Contains information necessary for I/O to a TTY.

DEVNAM	Physical Name - Dynamically set up		
DEVCHR	See bit definitions		
DEVIOS	See bit definitions		
DEVSER	Adr of next DDB	Device Dispatch Tbl	
DEVMOD	See bit definitions		
DEVLOG	Logical device name		
DEVBUF	Output ring hdr adr	Input ring hdr adr	
DEVIAD	HP	P	Current in buf adr.
DEVOAD	Out buf sz	P	Current out buf adr
TTYPTR	See bit definitions		
TISYNC	Number of break characters in input buf		
TTIBUF	Input buffer size	Input buffer adr	
TIPUTR	Pointer to put character into input buf		
TIPCTR	No of Bytes before end for TIPUTR		
TITAKR	Pointer to take character from in buf		
TITCTR	No of bytes before end for TITAKR		
TIFCTR	Free character count for Input Buf		
TIBF	Input Buffer - 16 words -		
TTOBUF	Output buffer size	Output buffer adr	
TOPUTR	Pointer to put character into output buf		
TOPCTR	No of bytes before end for TOPUTR		
TOTAKR	Pointer to take characters from out buf		
TOTCTR	No of bytes before end for TOTAKR		
TOFCTR	Free character count for Output Buffer		
TOBF	Output Buffer - 16 words -		

TTY DDB (cont.)

DEVCHR Definitions



Byte Pointer Labels

Job # PJOBN
Hung Const PDVTIM
Countdown PDVCNT
Line # PUNIT

TTY DDB (cont)

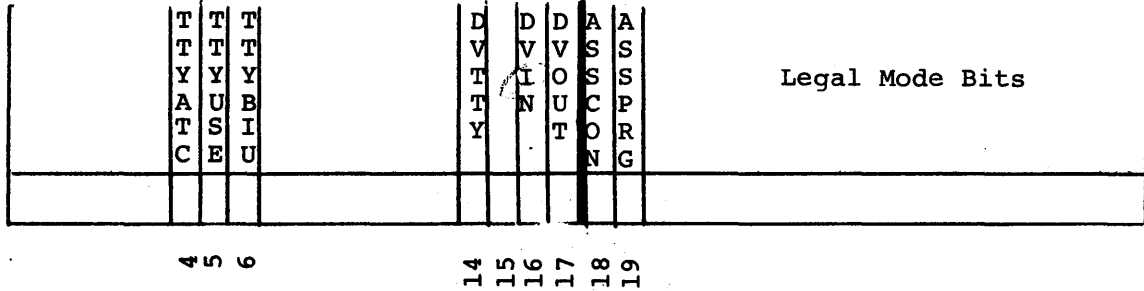
DEVIOS Definitions

TTYIOW		SYNC	TTYDTC	DDTM	TPMON	IOSUPR	USRB	IOEND	IO	IOFST	IOBEG	IOW	MERTPO	IGNO	ECHO							DLRSUP	NOECHO	FCS						
0	2	5	8	11	14	17	20	23	26	29	32	35																		

<u>Bit</u>	<u>Label</u>	<u>Meaning</u>
0	TTYIOW	TTY Input Wait
4	SYNC	Break character seen
7	TTYDTC	DDB is detached from line
8	DDTM	DDT Mode
9	TPMON	TTY is in monitor command mode
10	IOSUPR	Suppress all output until next input or INIT.
11	USRB	Set TTY to user mode when output finished.
12	IOEND	Set at interrupt level when End of File seen.
13	IO	Direction of transfer; Out = 1; In = 0
15	IOFST	Next interrupt will be first item of buffer.
16	IOBEG	Virgin device bit
17	IOW	Job in I/O wait
18	MERTPO	Monitor error timeout
19	IGNORE	Ignore all input for 1/2 sec, echo check.
20	ECHO	Echo failure.
23	IOACT	Device Active
27	DLRSUP	Suppress \$ for Altmode
28	NOECHO	Suppress echo print
29	FCS	Full character set

TTY DDB (cont)

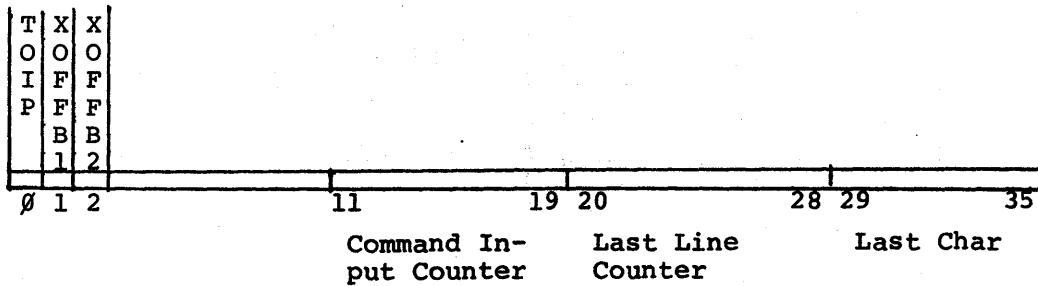
DEVMOD bit definitions



<u>Bit</u>	<u>Label</u>	<u>Meaning</u>
4	TTYATC	This TTY attached to a job
5	TTYUSE	This DDB in use (as user console)
6	TTYBIU	This DDB in use (as console or I/O device)
14	DVTTY	Device is a TTY
16	DVIN	Device can do input
17	DVOUT	Device can do output
18	ASSCON	Device has been assigned by a console
19	ASSPRG	Device has been assigned by a program (i.e., INIT)

TTY DDB (cont)

TTYPTR definitions



<u>Bits</u>	<u>Label</u>	<u>Byte Pointer</u>	
0	TOIP		Type Out in Progress- Set to 1 while output in progress
1	XOFFB1		Force XOFF on full buffer
2	XOFFB2		Force XOFF on full buffer
3-10			Unused
11-19		PCOMIC	For rescanning console command
20-28		PLSTLC	For ↑U deletion
29-31		PLASTC	For echo checking on half duplex lines

TTY DDB (cont.)

Note that there is a group of six parameter words preceding each of the two buffers in the TTYDDB. These groups are identical in structure. SCNSRF defines labels to use with either group, depending on the contents of the index register used with them.

<u>Label</u>	<u>Value</u>	
BUF	Ø	Pointer to beginning of buffer
PUTR	1	Byte pointer for putting characters into the buffer
PCTR	2	Number of bytes before end for PUTR
TAKR	3	Byte pointer for taking characters from buffer
TCTR	4	Number of bytes before end of TAKR
FCTR	5	Free character count

These labels are normally indexed by AC DAT. DAT contains the address of TTIBUF for the symbols to apply to input buffer parameter words. DAT contains the address of TTOBUF for the symbols to apply to the output buffer parameter words.

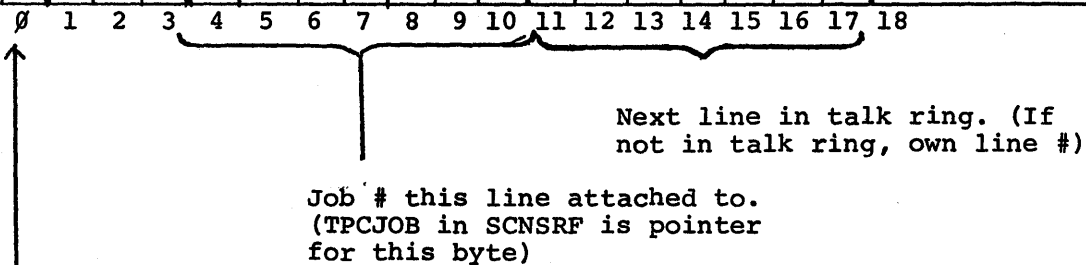
TTYTAB

Gives the connection between physical line numbers and TTY DDBs. Also provides some information about events that have occurred on each line.

One entry for each line, followed by entries for each PTY.

Indexed by line number.

Line #	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18		
Line 0																					Adr of DDB
Line 1																					for this line
Line 2																					"
Line 3																					"
=																					
Line N																					"



Set when break character typed on TTY in monitor command mode. Searched for on Channel 7 level by TTYCOM.

Defined in COMMON

- Referenced in BTHINT
- DLSINT
- CCIINT
- PTYSRF
- COMCON
- SCNSRF
- DCSINT
- UUOCON

UCLJMP

Contains dispatch addresses for CALL and CALLI UWO's.

One entry for each two routines. LH contains address for even numbered routines; RH, odd numbered.

Indexed by one half the CALLI argument.

UCLJMP:

Customer Adr 4	Customer Adr 3
Customer Adr 2	Customer Adr 1
DEC Adr 0	DEC Adr 1
DEC Adr 2	DEC Adr 2

For CALL UWO's, a table lookup is done in UCLTAB to get routine number; for CALLI UWO's the number is supplied directly. One half this routine number is used as the table index. The LH is taken for even numbers; RH for odd. The CALL routine then dispatches to that address.

Defined in UWOCON (Local Symbol)

Referenced in UWOCON

NOTE: UCLJMP has entries corresponding to entries in UCLTAB. Table entries may be added in the negative direction by customers, and in the positive direction by DEC. Once a table entry is established, its position can never be changed without invalidating those programs which use the corresponding CALLI.

UCLTAB

Contains names of the CALL UO's

One entry for each CALL function.

Indexed by corresponding CALLI value. Customer defined CALL's have negative index values; DEC CALL's have positive values.

UCLTAB

Customer CALL 3
Customer CALL 2
Customer CALL 1
DEC CALL 0
DEC CALL 1
DEC CALL 2

Table entries are SIXBIT expressions of the CALL names. There are corresponding dispatch addresses in Table UCLJMP.

Defined in UUOCON (Local Symbol)

Referenced in UUOCON

NOTE:

Customers may extend the table in the negative direction with as many of their own CALL's as desired. DEC may add entries in the positive direction.

The value specified in a CALLI UO corresponds to the position of the CALL UO name in this table. Hence, once an entry is established, its position in the table can never be changed without invalidating any existing programs which use that CALLI.

UUOTAB

Contains address of operator-dependent UVO routines.

Table is in order of UVO Op Code, with two addresses per entry.
Entry N contains addresses corresponding to Op Codes $40 + 2N$,
 $41 + 2N$.

Adr for Op Code 40	Adr for Op Code 41
Adr for Op Code 42	Adr for Op Code 43
Adr for Op Code 76	Adr for Op Code 77

Defined in UUOCON.

Referenced in UUOCON.

Entries corresponding to invalid Op Codes contain the address of UUOERR. Some of these are reserved for future use by Digital, others for customers. See current listing for specific examples.