

DECSYSTEM-10 MONITOR

TABLE DESCRIPTIONS

Revision 1 - TOPS-10 Version 7.01

June, 1980

This document reflects:

Operating System:	TOPS-10 V7.01
Assembler:	MACRO%53
Hardware:	DECsystem-1070 through DECsystem-1099

Educational Services
Digital Equipment Corporation
Marlboro, Massachusetts

Copyright (c) 1980 by Digital Equipment Corporation.

The material in this document is for informational purposes and is subject to change without notice; it 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. Digital Equipment Corporation assumes no responsibility for the use or reliability of its software on equipment that is not supplied by Digital or its affiliated companies.

The following are trademarks of Digital Equipment Corporation, Maynard, Massachusetts:

COMPUTER LABS	COMTEX	DBMS-10
DBMS-11	DBMS-20	DDT
DEC	DECCOMM	DECsystem-10
DECSYSTEM-20	DECTape	DECUS
DIBOL	DIGITAL	EDUSYSTEM
FLIPCHIP	FOCAL	INDAC
LAB-8	MASSBUS	OMNIBUS
OS/8	PDP	PHA
RSTS	RSX	TYPESET-8
TYPESET-10	TYPESET-11	TYPESET-20
UNIBUS	DECSYSTEM-2020	

TABLE OF CONTENTS

AABTAB	Access Allowed Bit Table	1
ACC	Access Table	2
AVALTB	Available Resource Table	9
BAF	Bad Allocation File	11
Buffer Ring		16
Buffer Ring Header		18
CDB	CPU Data Block	20
CDT	Connected Device Table	37
CHKTAB	UWO Check Bit Table	38
CHKTBC	CALL and CALLI UWO Check Bit Table	30
CHTABL	Special Character Table	42
CHN	Channel Data Block	50
CIPWT	Clock Request Queue	53
COMTAB	Command Table	55
COMTB2	Set Command Table	56
DDB	Device Data Block	57
Disk DDB	Disk Device Data Block	68
Magtape Disk DDB		84
DEVDSP	Device Dispatch Table	86
DISP	Command Dispatch Table	89
DSCTAB	Data Set Characteristics Table	91
EXE File Format on Disk		93
FETTL	Table of Bits for Monitor Features	95
Fragment Table		96
HOM	Home Block	97

INTTAB	Interrupt Routine Table	102
JB TADR	Job Address Table	104
JB TAD2	Second Job Address Table	106
JB TCHK	Segment Checksum Table	107
JB TCQ	Job Queues Table	108
JB TCSQ	Job Subqueues Table	110
JB TDEV	High Segment Device Table	111
JB TJIL	Just Swapped In List	112
JB TJRQ	Job Requeue List	113
JB TLIM	Job Time Limit Table	114
JB TLOC	Job Location Table	115
JB TNAM	Job Name Table	116
JB TOLS	Job Output List	117
JB TPC	User Mode PC	118
JB TPDB	Process Data Block Table	119
JB TPIA	Software Program Interrupt Table	120
JB TPPN	Project Programmer Number Table	121
JB TPRV	Job Privilege Bits	122
JB TRQT	Job Run-Queue Time Table	124
JB TRSP	Job Response Time Table	125
JB TRTD	Real Time Devices	127
JB TSCD	Job Class Table	128
JB TSFD	Sub File Directory Table	129
JB TSGN	Job Segment Table	130
JB TSPL	Spool Control Table	132
JB TSPS	Second Processor Status	133
JB TSTS	Job Status Table	134

JBTST2	Job Status Table 2	139
JBTSWP	Job Segment Swap Area	142
JBTUPM	UPMP Address Table	144
JBTVRT	Paging Rate Table	146
JBTWCH	Job Watch Table	147
JDA	Job Device Assignment Table	149
JOBDAT	Job Data Area	151
Job Queues and Wait State Codes		154
KON	Disk Controller Data Block	156
LDB	Line Data Block	161
LINTAB	TTY Line Information	177
LOGTAB	Login JBTTAB Pointer Table	179
LVDTBL	Level D Disk Parameter Table	180
Magtape Kontroller Data Block		184
Magtape Unit Data Block		187
MEMTAB	Virtual Memory Page Table	190
MFD	Master File Directory	191
NMB	File Name Block	193
NUMTAB	Table of GETTAB Tables	196
PAGTAB	Page Table	197
PDB	Process Data Block	199
PPB	Project Programmer Number Data Block ..	204
PTYTAB	Pseudo-TTY DDB Table	205
QBITS	Wait State Code Requeue Table	206
Queue Transfer Table		208
Queue Table - Job Scanning		210
Queue Table - Quantum Time		212

Queue Table - Queue Progression	213
REQTAB	Sharable Device Request Table 214
RIB	Retrieval Information Block 216
SAB	Storage Allocation Block 223
SAT	Storage Allocation Table 225
SCNDDDB	Scanner Device Data Block 226
Scheduler Scan Tables	233
SPT	Storage Allocation Pointer Table 235
STR	File Structure Data Block 236
SWPLST	VM Swapper Table 239
SW2LST	SWPLST Entry Table 240
SW3LST	SWPLST Job Number Table 241
SWPTAB	Swapping Unit UDB Address Table 242
TABSTR	File Structure Data Block Address Table 243
Transfer Tables	244
TTFCOM	Forced Commands Table 246
Teletype Buffers	248
TTUUOT	TTCALL Dispatch Table 250
TTYTAB	Attached Terminal Job Table 251
TYPTAB	Device Types Table 252
UCLJMP	CALL and CALLI UUO Dispatch Table 253
UCLTAB	CALL UUO Names Table 254
UDB	Unit Data Block 255
UFB	UFD Data Block 265
UFD	User File Directory 267
UPMP	User Page Map 269
UUOTAB	UUO Dispatch Address Table 272

WSBTAB	Working Set Bit Table	273
--------------	-----------------------------	-----

This page is for notes.

Name: AABTAB Access Allowed Bit Table

Description: Table of bits, located in UPMP from locations 457 to 475, which reflects whether access is allowed to a user page or not. IPCF and the UUO Page UUO handler routines reference this table.

Defined In: S.MAC

Used By: VM SER

Format: UPMP

	.				
	.				
	.				
457					
			.		AABTAB
			.		
			.		
475					

Name: ACC ACCESS TABLE

Description: Contains information needed in order to gain access to a specific "version" of a specific file.

One table for each existing version of each open file.

The access table for the current and possibly a superseding version of the file are linked to the corresponding NMB. Also, each disk device data block contains a pointer to the access table for the file currently being accessed.

Defined In: COMMOD

Used By: FILFND, FILIO, FILUO, SYSINI

See also: NMB, PPB

Format:

ACCALC(0)	Highest Relative Block Number Allocated		
ACCNMB(1)*	NMB or next ACC		See following page
ACCPT1(2)	First Retrieval Pointer to file		
ACCDOR(3)	Next Dormant ACC		Previous Dormant ACC
ACCPB(4)*	See following pages		PPB
ACCADT(5)*	See following pages		
ACCWRT(6)	Highest Relative Block Number Written		
ACCPRV(7)*	Privileges	MODE	Creation Time Lo Creation Date

*Details on following pages.

Special Notes:

1. When all users have closed a file, its access table is considered dormant. Dormant access tables are linked into a doubly linked list through ACCDOR, and are not deleted until their core space is needed. If a table is not dormant, ACCDOR contains zero.
2. ACCWRT and ACCALC do not include the second RIB.

ACCNMB (1) Byte Definitions

+-----+-----+-----+-----+-----+-----+-----+-----+-----+-----+									
	NMB or Next ACC		X		LBS		X	UN1	X X X X
+-----+-----+-----+-----+-----+-----+-----+-----+-----+-----+									
0			17	18	19		26	27	28 31 32 33 34 35

<u>Word</u> <u>Label</u>	<u>Bits</u> <u>Byte</u>	<u>Bit Label or</u> <u>Pointer</u>	<u>Content</u>
ACCNMB	0-17		Address of next ACC for same name and PPN if there is one.
ACCABC	18	ACPABC	This file always has bad checksum.
ACCLBS	19-26	ACYLBS ACZLBS	Number of words in last data block of file.
ACC1PT	27	ACP1PT	File has only one retrieval pointer.
ACCUN1	28-31		Logical unit number, within STR where file begins.
ACCDIR	32	ACPDIR	This file is a directory.
ACCNDL	33	ACPNDL	This file cannot be deleted.
ACCSBC	34	ACPSBC	Sometimes bad checksum.
ACCGRB	35	ACPGRB	Don't grab access table

Notes:

1. Normally there is only one ACC linked to an NMB. But while a file is being superseded, both the old and the new versions of the file have ACC's linked to the NMB.
2. The low order two bits of a pointer to another ACC will be zero. The pointer back to the NMB points to the last word, NMBACC. Hence, its low order

bits will not be zero.

3. There may be an arbitrary number of ACC's for older versions of a file, which are still being read. However, these ACC's are not linked to the NMB. The only pointers to them are in the DDB's of the readers.

ACCPPB (4) Byte Definitions

+-----+-----+-----+-----+-----+				
	FSN		SIM UPDTR'S	
+-----+-----+-----+-----+-----+				
0	5 6		13	18 35

Word Label	Bits	Bit label or Byte Pointer	Content
ACCFSN	0-5	ACYFSN ACZFSN	File structure number of STR to which this file belongs.
ACCWCT	6-13	ACYWCT ACZWCT	Write count for simultaneous updates
ACCPPB	18-35		Core address of project-programmer data block.

ACCADT (5) Byte Definitions

-----+-----																										
	X		Last Access Date						X		CNT/USE				X		X					X		X		
-----+-----																										
0	2	3	17 18 19					27 28 29 30					32 33 34													

Word Label	Bits	Bit Label or Byte Pointer	Content
ACCADT	0-2		High-order 3 bits of creation date. Low-order part is in ACCCDT.
ACCADT	3-17		Date this file was last accessed for more than just LOOKUP.
ACCNIU	18	ACPNIU	File not in UFD
ACCCNT	19-27	ACMCNT	Count of user channels with LOOKUP in force for this version of this file or SFD use count.
ACCREN	28	ACPREN	Rename in progress.
ACCDEL	29	ACPDEL	File to be deleted when all readers finished.
ACCSTS	30-32	ACYSTS	Access Table Status Code
ACCSMU	33	ACPSMU	This file being simultaneously updated.
ACCPAL	34	ACPPAL	Pre-allocated File

No tes :

1. Access Table State Codes

```
ACRCRE      4      File being created ACPCRE (40)
ACRSUP      2      File superseding another ACPSUP (20)
ACRUPD      1      File being updated ACPUPD (10)
```

ACC (cont.)
 ACCPRV Byte Definitions

+-----+-----+-----+-----+									
Privilege		Mode		Creation Time				Creation Date	
+-----+-----+-----+-----+									
0		8 9		12 13				23 24	
								35	

<u>Word</u> <u>Label</u>	<u>Bits</u>	<u>Bit Label or</u> <u>Byte Pointer</u>	<u>Content</u>
ACCPRV	0-8	ACYPRV	Privileges
ACCMOD	9-12		Data mode of file
ACCCTM	13-23		File creation time
ACCCDT	24-35		Low-order 12 bits of file creation date. Upper 3 bits in ACCADT.

Note: Privilege Codes

Bits 0-2 Apply to job which is "owner" of file
 Bits 3-5 Apply to any job with matching project number.
 Bits 6-8 Apply to all other jobs.

A job is considered the owner of a file if:

- A) INDPPN set to 0 at MONGEN (default) and programmer number matches or
- B) INDPPN set to -1 at MONGEN and both project and programmer number match.

Privilege Codes for User Files

Code	Highest Privileges
7	None
6	Execute
5	Read
4	Append (Allocate, Deallocate)
3	Update
2	Write (Supersede, Truncate)
1	Rename (Change attributes)
0	Change privileges

Privilege Codes for Directories

Code	Privilege Given by Bit Being Lit
4	Allows lookups in this directory
2	Allows creates
1	Allow directory to be read as a data file

-- Any combination of these bits may be set.

Name: AVALTB AVAILABLE RESOURCE TABLE

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

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

Defined In: COMMON

Used By: COMMON, CLOCK1, SCHED1, SYSINI

Format:

AUAVAl	
DAAVAl	
CBAVAl	
D1AVAl	
D2AVAl	
DCAVAl	
MMAVAl	
CAAVAl	
EVAVAl	
EQAVAl	

Special Notes:

1. Table REQTAB has entries corresponding to the AVALTB entries.
2. The AVALTB entries are built by the conditionally assembled RWAITS MACRO entries in S.MAC; therefore, all of the above listed entries will not be present in most systems.
3. This table is initialized to zero.

Name: BAF BAD ALLOCATION FILE

Description: Disk block in which all known bad regions in
a STR are recorded.

The BAF is always the next block after the
home block in HOME.SYS.

Defined In: COMMODO

Used By: FILUUO, ONCMOD, REFSTR

Format:

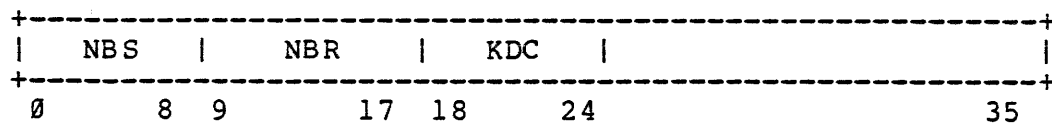
BAFNAM	SIXBIT / BAT /				(0)		
BAFFIR	-No. free words in BAF			Rel adr of first bad region pair	(1)		
BAFNBS*	NBS		NBR		KDC		(2)
BAFCNT	No. pairs added to BAF by monitor					(3)	
BAFREG	Bad Region Pair*						
	Bad Region Pair*						
	.						
	.						
BAFCOD	0		Unlikely Code (606060)			Last t words block	
BAFSLF	0		This block # in unit				

*Details on following pages.

Special Notes:

1. The label BAFREG should not be used by programs which look at this block. They should use RH BAFFIR to determine location of the first Bad Region Pair.
2. The Map Program is a stand alone program which checks all disk blocks and writes an initial BAF. As the monitor finds bad blocks it makes additional entries in the BAF.
3. Both halves of BAFFIR are written by the mapping program and are never changed by the Monitor. LH is always an even number.

BAFNBS Byte Definitions



Word Label	Bits	Byte Pointer	Content
BAFNBS	0-8		Number of bad blocks found by the map program.
BAFNBR	9-17	BAYNBR	Number of bad regions found by the map program. (Number entries in following table.)
BAFKDC	18-24	BAYKDC	Controller device code used by map program.

First Word of Each Bad Region Pair - Byte Definitions

+-----+-----+-----+-----+-----+-----+																		
	NBB				0		PUB				KNM				APN			
+-----+-----+-----+-----+-----+-----+																		
0	8			9	10		17			18	20		22			35		

Word Label	Bits	Byte Pointer or Bit Label	Content
BAFNBB	0-8	BAYNBB	Number of bad blocks -1 in this bad region (not clusters).
BAFOTH	9	BAPOTH	Non-zero if this bad regions is detected on another controller or processor than the one which originally added the entry.
BAFPUB	10-17		Physical unit within controller. Bit 17-N represents unit N, where N = 10-17.
BAFKNM	18-20	BAYKNM	Logical controller number, of this type. From UNIKNM.
	21	BAPNTP	Non-zero if new style BAT block entry.
BAFAPN	22-35	BAYAPN	Serial number of APR running when error was detected.

Second Word of Bad Region Pair

+-----+										
ver	OTR	DTR	HDR	LBN of 1st bad block in region						
+-----+										
0	2	3	4	5	6	8	9	35		

BAFVER 0-2 Version number of entry (presently 0)

BAPOTR 3 Other error (not data or search error)

BAPDTR 4 Data error (parity or ECC hard)

BAPHDR 5 Search error or header compare error

Notes: Values of the word labels are relative to the
beginning of the Bad Region Pair, i.e., 0 and 1.

Name: Buffer Ring

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

Defined in: Status bits defined in S.MAC

Format:

		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 is set as follows:
	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.

Name: Buffer Control Block (Buffer Ring Header)

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

Defined in: S.MAC

Format:

JBFADR	U		Current Buffer Address	
JBFPTR			Byte Pointer	
JBFCTR			Byte Count	
JBFUDX			Univ Dev Index (UDX)	

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.
JBFUDX		UDX for MSGSER

Notes:

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

Name: CDB CPU DATA BLOCK

Description: There is one CPU data block (CDB) for each CPU on a system. The block contains both a constant and a variable data area. Routine addresses, pointers, bit masks and processor dependent instructions are representative of the data found in the constant area. The variable area is cleared at initialization and on 403 restarts. This area contains current status words for a CPU including current job information, protected job data locations, scheduler and swapper parameters, etc.

Reference can be made to a CDB location without indexing. The symbols used in this case will be in the format of .Cn??? where n is the logical number of the CPU (i.e., .C0??? or .C1???).

A 'C' macro is used to generate labels and data for the constant area. All of the 'C' macros are contained in a second macro, CDBCON. CDBCON will be called once for each CDB to be generated after defining the 'C' macro for the call.

The 'V' macro is used for symbol definition are entry counting for the variable area. No data is generated on a 'V' macro call. The count is used to assign absolute values to variable locations in the CDB and also to reserve a block for the variable data.

Defined in: COMMON

Used by: CLOCK1, CP1SER, ERRCON, KISER, KLSER, KSSER, PSISER, SCHED1, SYSINI, UUOCON

GETTAB table no.: 45 (.C0CDB), 47 (.C1CDB)

NOTE

All 'CDB SYMBOL' entries on the following pages require .CP, .CØ or .C1 prefixes. The 'OLD SYMBOL' entry indicates the symbolic reference used prior to 5.Ø4 and retained in some 5.Ø4 modules.

The table as displayed here is a KL dual-processor assembled table. Entries vary both in the constants and variables area if it is a KI, KA, KS or non-VM.

CDB SYMBOL OLD SYMBOL

CDB	Address of next CDB,, 0
ASN	APR serial number
OK	This CPU running OK. If greater than 0, this CPU has stopped running correctly. Contents are No. of jiffies CPU has been stopped.
TOS	Trap offset value Location of Exec page maps for KI, KL
LOG	Logical CPU name SIXBIT/CPUn/ (n=0,1---)
PHY	Physical CPU name SIXBIT/CPxn/ (x=6,A,I,L) (n=0,1,...)
TYP	CPU Type (1-PDP-6, 2-KA10, 3-KI10, 4-KL10, 5-KS10)
MPT	Pointer to bad address subtable bits. Bits 0-8 length, RH=rel adr in CDB
RTC	Real time clock (DK10) DDB
RTD	DK10 DDB if HI prec. time acct. 0 if low (APR clock)
PAR	Pointer to parity subtable bits 0-8 length, RH=rel adr in CDB
RSP	Pointer to response subtable bits 0-8 length, RH=rel adr in CDB
DKX	Number of DK10s on this CPU
EBS	EBOX ticks per second
MBS	MBOX ticks per second

NMT	Pointer to NXM subtable bits 0-8 length, RH=rel adr in CDB
CSB	Pointer to CPU Status Block
DSB	Pointer to Device Status Block
SDP	Pointer to SBDIAG subtable
BPA	Performance analysis subtable ptr
CNP	CPU Number
SK0	Generate SKPCPU(0)
SK1	Generate SKPCPU(1)
OK1	Address of other CPU's OK word
SLF	Pointer to start of CDB
SCN	Scheduler run queue scan list (SSCAN or SSCAN1)
SST	Address of subqueue scanning table
NPD	Null PDL pointer
EPD	Error PDL pointer
NJD	Address of Null job data area (offset by 20)
STO	Scanner once a tick routine
ISR	Scanner once a second routine
DLK	Used to call DSKLOK
SCD	Scheduler interlock flag
RES	Restore after power fail
NBI	Number interrupts broken by BRKLOK

ABK	Return address for CPNBPT
KAF	Keep alive failure dispatch addr
RUN	CPU run flags Bit 0 = don't run jobs Bit 1 = run jobs
NAP	CONSO, CONI Bits all PIs except APR PI in progress
APP	CONSO, CONI BITS APR PI in progress
API	APR PI channel for this CPU
CHL	Address of interrupt PC for interrupt level (AP0CHL OR AP1CHL)
CKL	Addr of interrupt PC for clock lev.
CON	APR CONSO Mask
EEB	Exec enabled bits
IEF	Mask to clear all interrupting APR error flags except memory parity and clock (and sweep done)
CCF	Clear clock flag instruction
HCT	Instruction to skip if clock has ticked.
MPI	Address of CUn parity sweep instruction (CPAMPI## or CPIMPI## or CPLMPI##)
MPS	Bits to request parity sweep,, Address of parity sweep subroutine
NXM	Bits to test for non-existent memory
MPE	Mask to set/clear APR Parity err

DBM	CPU Doorbell word
EBR	Exec base register on this CPU
FEF	Front end flag - set when FE interrupts
CTN	RH = CTY Number LH = DLS Line number on DTE
DTN	Number of DTEs on this CPU
CAC	CRASH AC's
STT	K?SER temp for trap processing
ST1	Save T1 on page traps
ST2	Save T2 on page traps
EJ1	K?SER temp for IME processing
EJ2	K?SER temp for IME processing
EJ3	K?SER temp for IME processing
EJ4	K?SER temp for IME processing
RCT	Real time PI Channel Table
RDT	Real time Dismiss Table
CPI	PI Status
SVA	JSR here to save all AC sets
SVB	Where to go to save AC sets
TRP	MUO during crash
RTT	RTTRP in progress flag
TML	Value of time at last clock tick
OCB	Ø if this CDB isn't owned by a CPU

AID	-1 if restart(vs. initial startup)
DWD	Recursion interlock for DIE
CML	Address of LDB of line on which command is being processed on this CPU
SPR	Instruction to cause parity error Interrupt (KL)
CPR	Clear parity error flag (KL)
SB0	SBDIAG function 0 here on errors
S0A	Answer returned here
SB1	SBDIAG function 1 here on errors
S1A	Answer returned here
TDA	Addr of timeout routine for CPU
TIV	Addr of vector with input routines
EDV	Code "EDV", length of EXEC data vector
ED1	Length of address swapping block
ED2	Relocated contents of .JB SYM
ED3	Relocated contents of .JB USY
ED4	Address of word for DDT to use
HSF	Word for DDT to use
SYB	Length of address swapping block
SY1	Number of words to swap
SY2	First word to swap
SY3	Place where new map may be found
SY4	Place to save old contents

CPU DATA BLOCK Variable Area

Variable areas are cleared at system startup and 403 restarts.

CDB SYMBOL OLD SYMBOL

	CORMAX	Largest Address +1 allowed to user
	CORLST	Pointer to last free block possible
	CORTAL	Free + Dormant + Idle core blocks
	SHFWAT	Job number shuffler has temporarily stopped for its IO devices to become inactive before shuffling
	HOLEF	Absolute address of lowest hole in core
UPT	UPTIME	Uptime for this CPU in clock ticks
	SHFWRD	Total number of words shuffled by system
	STUSER	Job currently using the system tape
	HIGHJB	Highest job number currently assigned
	CLRWRD	Number of words cleared by 'CLRCOR'
LST	LSTWRD	Lost time on this CPU in jiffies
	MEMSIZ	Size of physical memory in words
TPE		Total parity errors
SPE		Total spurious parity errors (did not recur)

MPC		Memory parity continues for this CPU
MPA		Memory parity address of first bad address
MPW		Contents of first bad word found.
MPP		Memory Parity PC exclusive of parity sweep.
	EPOCNT	Number of PDL OVF's at UUO level not recovered
	EPOREC	Number of PDL OVF's at UUO level recovered
	MAXMAX	Highest legal value of CORMAX
	SYSKTM	Count-down timer for set KSYS command
	CORMIN	Lower bound on CORMAX
ABC		Address break count on this CPU
ABA		Address break address on this CPU
LJR		Last Job run on this CPU
ODA		Obsolete
STS	STOPTS	Stop timesharing on this CPU. Contains job no. which did TRPSET UUO.
ORU		Operator controlled scheduling for this CPU. Bit 0 = Run jobs Bit 1 = Don't run jobs
NUL		Null time for this CPU in jiffies
EDI		No. of exec don't care interrupts

JOB	JOB	Current job on this CPU
OHT		Overhead time for this CPU in jiffies
EVM		Max amount of exec space for jobs mapped in exec mode by lock UUO.
EVU		Total EXEC Virt Addr. space currently being used to map user segments
LLC		Lock Loop Count. Number of times this CPU has looped waiting for other CPU
TUC		Total number of UUO's on this CPU
TJC		Total job context-switches
TNE		Total NXM's
SNE		Total non-reproducible NXM's
NJA		Total jobs crashed this NXM
MNA		First address found with NXM
EBJ		EBOX ticks/jiffy
MBJ		MBOX ticks/jiffy
PBA		Physical address with bad parity on last parity trap
TBD		Contents of bad word on last parity trap
TGD		Good contents of word after we recovered parity trap
NPT		Total number of parity traps
AER		Results of RDERA on strange APR INA

PEF	Results of CONI APR, on parity int
PSB	Results of SBDIAG's on parity int.
PPC	PC on last parity trap
PFW	Page fail word on last parity trap
HPT	No. hard AR/ARX parity traps
SPT	No. soft AR/ARX parity traps
PTP	No. page table parity traps
CSN	No. cache sweeps started
CLN	Times job skipped because cache sweep done first
CLT	Time in jiffies the CPU ran the null job because of cache state
CSD	Cache - swapper delay count
CRN	Cache sweep request sweep count
CEC	No. non-recoverable AR/ARX parity errors involving cache
PTR	Retry word for AR/ARX parity trap
TSD	AR/ARX trap routine has already saved APR, ERA, SB diags, etc.
REP	Parity/NXM recovery routines used to determine which type errors should be listed
NDB	Number of times this CPU's doorbell was rung
SBR	LH = unused; RH = bits telling which status blocks have been read on this CPU

BPF	Background Performance Analysis Flag - LT zero is meter enabled
FBI	File blocks input (read)
FBO	File blocks output (written)
SBI	Swapping blocks input (read)
SBO	Swapping blocks output (written)
SNC	Number of CPU stopcodes
SND	Number of DEBUG stopcodes
SNJ	Number of JOB stopcodes
SJN	Job number as last stopcode
SNM	LH = Name of last stopcode RH = PC+1 of last stopcode
SPN	Program running at last stopcode
STN	TTY name at last stopcode
SUP	User PC at time of last stopcode
SUU	UUC at time of last stopcode
EJN	Job number at last parity/NXM err
EPN	Program at last parity/NXM error
PPI	Results of CONI PI, on parity/NXM interrupt
TPI	Results of CONI PI, on parity/NXM trap
	Response sub-tables
	Memory parity subtable
	Memory NXM sub-table

		CPU Status Block
		Device Status Block
ADR	JOBADR	Same as JBTADR (J) and AC R
REL	USRREL	Highest rel addr for current user
XTM		Time of last switch from monitor cycle to user job or vice versa
LS2		Additional lost time (fractional jiffy)
NL2		Additional null time (fractional jiffy)
OH2		Additional overhead time (fractional jiffy)
TNT		Time interval since last at clock level
HTM		Hung device time check
SEC		Seconds left before doing once a minute code
RCU		Count of real time CONSOS of skip chain
RIT		Real time initialization table
DMI		Real time dismiss instruction
CKF	CLKFLG	Non-zero when CLK interrupt (PI 7) requested
TMF	TIMEF	Non-zero when APR clock ticked
SCF		Force scheduling from exec mode
RTF		Non-zero on realtime reschedule required

ISF	In scheduler flag
SUD	Scan used for this scheduling scan
HQU	Force scheduling to new HPQ
PLT	Potentially lost time
CLF	Potential cache lost time
CL2	Low order cache lost time
SDA	Number cache sweeps for core deallocation
CSR	Cache sweep request flag for this CPU
AEF	APR error flag
SAC	Saved copy of .CPAEF
APC	APR error PC on this CPU while job running on this CPU
NJE	Error in null job if non-zero
SFC	Scheduler fairness count
SQF	Non-zero if current job from subqueues
APR	State of MI prog dis switch
IPI	Interval timer PI assignment
CN1	CONSO mask for APR interrupts user wants to handle.
DT0	last DATA0 to set RELOC and PROT
SP	Place to save P on APR interrupt
S17	Place to save AC17 on CLK interrupt

A00	17 (octal) words to store AC's on parity sweep on PAR PI
A17	20th (octal) word for storing AC's on parity sweep PAR PI
LPP	Last mem par PC
LSB	Las seg BLTed to detect bad parity
LCI	Time of last Parity/NXM interrupt caused by channel reference
PIP	Pointer to real interrupt PC
PSP	Parity/NXM sweep in progress on this CPU
CHE	Channel error reporting in progress on this CPU
TCX	Results of DATAI PAG, on error trp
TCT	Triad counter for 60HZ leap jiffies
PJB	Owner of performance meter (job#)
MJB	Measured job of PERF. UUO.
MJ1	Job enable condition
PMR	# 0 means perf. meter is running
PAE	Used to store perf. analysis enables
PRQ	Semaphore used on testing and giving meter away
APS	Non-zero means ACCT and PERF meters should be kept in sync.
MM0	"Virtual Perf Meter's" High order mem ref count

MM1	Low Order
BPC	Background Performance Analysis (B.P.A.) sample interval (ticks)
BPI	B.P.A. saved PACTAB index
BPT	B.P.A. saved RDTIME at start of current interval
TIM	Clock Interrupt flag for KL10's
ETM	SOSN done on this location every minute to make sure chunks don't stay allocated forever
EAD	LH = address of last KL error chnk RH = addr of first KL error chunk
KPB	KLINIT parameter buffer
20F	LH = flags about RSX20F front end RH = count of characters currently being output
20B	Buffer for 16 bit data to RSX20F
PTH	Trap happened during parity sweep
STE	ERA contents on sweep trap
PTF	Count of page table, parity traps between clock ticks -- used to crash system if too high.
PWF	Power fail AC block save area (holds four sets of ACs)
CTQ	SCNSER output queue header for CTY on this CPU; for KL10, is header for all RSX20F lines
QUE	Queue for I/O requests on other CPUs. Emptied into CPUDSQ once a tick

SWP	Non-zero if swap request for other CPU
QND	End of queue above
TAP	0 if no tape waiting for sweep -1 if no tape IO waiting for sweep 0,, -1 if tape waiting, DSKTIC did sweep
PIS	Save PI states for SYSPIN
DPI	Save PI state for DEVPIN
BTI	Save PI state for BTSOFF
IUT	Uptime a second ago
NTF	NETSER software interrupt flag
QTS	QUESER variables
CPG	Result of DATAI PAG, done by SVSETS with bits set so that DATAO will restore current AC set
ACA	Address of 20 word block in which SVSETS saved the current AC set

* Details follow

Notes:

1. CKF-CLKFLG (175) This location takes the place of a hardware interrupt flag.
2. TMF-TIMFF (176) Set so CLK interrupt routines will know another jiffy has passed.
3. SCF - (177) Normally CLK interrupts are dismissed when they interrupt exec mode. This flag set after monitor detects an error and desires to force rescheduling.

Name: CDT CONNECTED DEVICE TABLE

Description: Used to point to DDB's of devices connected to the Multiplexed I/O Facility (MPX). This table contains the connected device's UDX number and DDB address. This table is sorted by UDX number.

The left half of the DEVXTR word of the MPX DDB points to the CDT table.

Defined In: MSGSER

Used By: MSGSER

Format:

Size of table	No. of free slots
UDX 1	DDB addr.
UDX 2	DDB addr.
UDX 2	DDB addr.
"	"
"	"
"	"

Etc.

Name: CHKTAB UUO CHECK BIT TABLE

Description: Contains bits for checking UUO's which can be executed on CPU1 (UU.CP1), Effective Address (UU.EA) and LOOKUP, ENTER and RENAME (UU.LER).

The table is in the same order as UUOTAB so the UUOTAB index in AC T2 serves to get the CHKTAB entry.

If a check bit is specified for the requested UUO, UUOCON goes to the UUOCHK routine in VMSE.

Defined In: UUOCON

Used By: UUOCON, CPNSER

Format:

+-----+-----+	
Check Bit for 40	Check bit for 41
0	0
0	0
0	Check Bit for 47
"	"
"	"
+-----+-----+	

Etc.

0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17
18	19	20	21	22	23	24	25	26	27	28	29	30	31	32	33	34	35
U							U					U					
U							U					U					
.							.					.					
C							L					E					
P							E					A					
1							R										

Name	Bit	Description
UU.CP1	18	UUO can be executed on CPU1
UU.LER	25	LOOKUP/ENTER/RENAME
UU.EA	29	Check effective address

Name: CHKTBC CALL AND CALLI UO CHECK BIT TABLE

Description: Contains validity check bits for CALL and CALLI UO's.

This table is in the same order as UCLTAB so the UCLTAB index in AC T2 serves to get the CHKTBC entry.

If a check bit is specified for the requested UO, UUOCON goes to the UUOCHK routine in VMSE.

Defined In: UUOCON

Used By: UUOCON, CPNSER

Format:

Check Bit for RESET	Check Bit for DDTIN
Check Bit for SETDDT	Check Bit for DDTOUT
Check Bit for DVCHR	CPOPJ
"	"

Etc.

0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17
18	19	20	21	22	23	24	25	26	27	28	29	30	31	32	33	34	35
U			U	U	U	U	U	U	U	U	U						
U			U	U	U	U	U	U	U	U	U						
.			.	.	.	.	.	.	.	.	.	Number of					
C			N	W	N	L	L	M	C	C	E	Arguments					
P			C	C	A	F	E	N	E	A	A						
1			L	C	L	T	R	S	A	C							

Name		Bit Description
UU.CP1	18	UUO can be executed on CPU1
UU.NCL	21	Negative argument (repeat) count is legal
UU.WCC	22	Working set can change
UU.NAL	23	Not allowed if locked
UU.LFT	24	Use left half
UU.LER	25	LOOKUP/ENTER/RENAME
UU.MNS	26	Minus number of arguments
UU.CEA	27	Use C (effective address)
UU.CAC	28	Use C (Ac)
UU.EA	29	Check effective address

Name: CHTABL Special Character Table

Description: One table entry for each ASCII character, specifying characteristics and, in some cases, address of a special action routine.

Index value is the binary value of the character.

Defined in: SCNSER

Used By: SCNSER, PTYSER

Word Format

0	1	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17
C			C	C	C	C	C	C	C	C	C	C	C	C	C	C
H			H	H	H	H	H	H	H	H	H	H	H	H	H	H
S			P	D	E	V	C	F	D	C	C	R	A	F	U	B
P			U	E	P	P	R	I	E	N	R	I	L	I	A	R
O			N	L	A	O	E	L	L	C	E	A	T	L	E	K
			C	W	R	S		O			T					

Bit	Label	Meaning
0	CHSPO	Requires special checking on output
1-3		Undefined
4	CHPUNC	Punctuation character
5	CHDELW	Character is CONTROL/W
6	CHEPAR	Character is even parity
7	CHVPOS	Vertical positioning simulated with line feeds
8	CHCRE	Gets CRLF after its echoing (see CHUAE)

9	CHFILO	Bit for output filler routine (not in table)
10	CHDEL	This is Delete
11	CHCNC	This is Control C
12	CHCRET	This is a Carriage Return
13	CHRIA	RCV interrupt level action required
14	CHALT	This is an altmode (one of three)
15	CHFIL	Requires fillers at some speeds
16	CHUAE	Echoes as ^A, ^B, ^C, ^D, ^O ^P, ^Q, ^S, ^T, ^U, OR ^Z
17	CHBRK	This is a break character

	C		C	C	C	C	C	C	C	C	C	C	C	C	C	C	DISP	
	H		H	H	H	H	H	H	H	H	H	H	H	H	H	H	ADDR	
	S		P	D	E	V	C	F	D	C	C	R	A	F	U	B	ON	
	P		U	E	P	P	R	I	E	N	R	I	L	I	A	R	RCVINT	
	O		N	L	A	O	E	L	L	C	E	A	T	L	E	K		
			C	W	R	S		O			T							
000	1		1		1							1					RINUL	NULL
001	1		1									1			1		RICA	^A
002	1		1									1			1		RICB	^B
003	1		1		1		1			1		1			1	1	RICC	^C
004	1		1									1			1		RICD	^D
005	1		1		1										1			^E
006	1		1		1										1			^F
007	1		1													1		^G BELL

173			1	1													{
174																	
175	1		1	1							1	1				RIALT	Escape
176	1		1	1							1	1				RIALT	Escape
177	1		1						1			1				1 RIDEL	Delete

NOTES:

1. CHUAE is also set for entries 1, 2, 5, 6, 16, 20, 27, 30, 31, 34, 35, 36, 37.
2. CHEPAR is set for entries whose binary index contains an even number of 1 bits.
3. Table is referenced by the routine SPCHEK in SCNSER.
4. Entries 1, 2, and 20 will only have CHRIA and the dispatch address set if FTMIC is non zero.

Name: CHN CHANNEL DATA BLOCK

Description: Contains information pertaining to all devices on one data channel.

One CHN for each data channel, e.g., CHN3CB for the fourth data channel.

Defined In: COMMOD

Used By: FILIO, VMSE

Format:

.CHBSY	(Number requests on this channel) -1 is IDLE	
.CHSYS	ADR of next CHN Ø if last	Last UDB with error
.CHICW	Initial Control Word on Last Error	
.CHFCW	ICWAD+1 after last error	
.CHCW2	Command word -2 on last error	
.CHCW1	Command word -1 on last error	
.CHCWØ	Command word on last error	
.CHDW2	Data Word -2 on last error	
.CHDW1	Data Word -1 on last error	
.CHDWØ	Data word on last error	
.CHMPE	Number of memory parity errors	
.CHDPE	Number of data parity errors (from device)	
.CHNXM	Number of non-existent memory errors or data late errors (overrun)	

.CHLU

.CHCSR	Bits to request CPU to sweep core	Last DDB Address	.CHLD
.CHTCW	Expected termination control word of last error		

The remaining words apply only to disk channels

CHNECT	Error count on current data transfer		
CHNRCT	Current recalibrate count		
CHNQUE	First DDB in TWQ	Job Number of LH	
CHNIFP*	Initial fairness count for positioning		
CHNCFP	Current fairness count for positioning		
CHNIFT	Initial fairness count for transfers		
CHNCFT	Current fairness count for transfers		
CHNIFS	Initial swapping fairness count		
CHNCFS	Current swapping fairness count		
CHNWAT	Ø if no I/O file I/O pending		
CHNNUM	Number of blocks currently being transferred		
CHNQUL	Length of XFER Wait Queue		
CHNTCW	Expected termination control word if no error indication from controller		

Special Notes:

1. The errors referred to in the Channel data block are both soft and hard errors.
2. TWQ = Transfer Wait Queue

3. CHNECT is -1 while recalibrating.

* Description of CHNIFP Bytes

CHNIFP	+-----+-----+-----+-----+-----+-----+-----+-----+-----+-----+-----+-----+-----+-----+-----+-----+-----+-----+																		
	C	D	R	1	L	2										S	Initial fairness count		
	+-----+-----+-----+-----+-----+-----+-----+-----+-----+-----+-----+-----+-----+-----+-----+-----+-----+-----+																		
	0	1	2	3	4	5										17	18		35

Word Label	Bits	Byte Label	Meaning
CHB22B	0	CP.22B	This is a 22-bit channel
	1	CP.DX1	This is a DX10 channel
	2	CP.RH2	This is an RH20 channel
	3	CP.R11	This is a RH11
	4	CP.LP2	This is a LP20
	5	CP.CD2	This is a CD20
	17	CP.SWF	Cache sweep must be done before interrupt exit in FILIO
CHNIFP	18-35		Initial fairness count for positioning

The bits in the second word are defined as follows:

Bit	Meaning
0	Request is CPU specific
1-3	CPU Number
4	Scanned by CLOCK1
5-35	Data

Name: COMTAB COMMAND TABLE

Description: 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 entires must be in corresponding positions.

Defined In: COMMON

Used By: UUOCON, COMCON

GETTAB Table No.: 30(octal)

Format:

SIXBIT	/	CMD1	/
SIXBIT	/	CMD2	/
SIXBIT	/	CMD3	/
	.		
	.		
	.		

Name: COMTB2 SET COMMAND TABLE

Description: When the command decoder has determined that a SET Command is to be executed it does a table lookup on the SET Command argument to determine the dispatch address. The SET Command argument table is named COMTB2 and is in the same format as COMTAB (q.v.). The dispatch table for the SET Commands is named DISP2. COMTB2 and DISP2 entries must be in corresponding positions.

Defined In: COMMON

Used By: UUOCON, COMCON

GETTAB Table No.: 43(octal)

Format:

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

Name: DDB DEVICE DATA BLOCK

Description: Contains information needed to perform I/O operations. One such block for each device or, in the case of disk, one for each INIT or ASSIGN.

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 F is normally used for this purpose in the monitor.

The start of the DDB chain is accessible as GETTAB [%CNDEV] from a user program, or in location DEVLST in the Monitor. DDB's for some peripherals have their own tags in the Monitor and some sub-chains are also tagged:

TTYLST - Start of TTY sub-chain
 PTYLST - Start of PTY sub-chain
 LPxDDB - Tag for LPTx on I/O bus
 CRxDDB - Tag for CDRx on I/O bus
 FLxDDB - Tag for LPTx off console front end (1091)
 FCxDDB - Tag for CDRx off console front end (1091)

Defined In: Device Service Routines and in S.UNV

Used By: Device Service Routines

Format:

DEVNAM	Physical Device Name *	Device Number *
DEVCHR	See bit definitions	See bit definitions
DEVIOS	See bit definitions	See bit definitions
DEVSER	Adr of next DDB	Adr-device dispatch tbl
DEVMOD	See bit definitions	See bit definitions

DEVLOG	Logical Device Name *	
DEVBUF	Output Ring Hdr Adr	Input Ring Hdr Adr
DEVIAD	R	Current input buffer adr
DEVOAD	R	Current output buffer adr
DEVSTS	Word for Devce CONI	
DEVSTA	See bit definitions	
DEVXTR	Extra word	
DEVEVM	Word to keep track of Exec Vir. Mem	
DEVPSI	Word to keep track of software interrupts	
DEVESE	PSI vector offset	
DEVHCW	Hardware characteristics	
DEVCPU	CPU related information	
DEVJOB	Job owning DDB and unit #	
DEVCTR	Buffer Item Count for Short Dispatch Table Devices	

* - in Sixbit code

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

DEVIAD, DEVOAD
Bits 14-17 are index register R

For long DDBs, DEVCTR is replaced with

DEVFIL	File name
DEVEXT	File extension
DEVPPN	File PPN
DEVNBF	Number buffers swept for Number buffers not swept
DEVSBF	Saved DEVNBF
DEVCSN	Cache sweep serial number when I/O requested

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 dynamically as needed or at system initialization time. The new DDB's are modeled after the one contained in the device service routine.
2. Each DDB in use by a job is linked to that job. The job # is in bits 27-35 of DEVJOB, accessed in the Monitor by the byte pointer PJOBN. Also the DDB address is in the Job Device Assignment table in the Job Data Area for that job.
3. The unit number is in bits 18-26 of DEVJOB, accessed in the Monitor by the byte pointer PUNIT.

DEVCHR byte definitions

+-----+-----+-----+-----+-----+									
Bits		Hung Constant		Countdown for Hung Check				Buffer Size	
+-----+-----+-----+-----+-----+									
0	6	7	9	10	16	17	23	24	35

Byte Name	Bits	Use
DVLPTL	0	Device is a lower case LPT
DVCMDA	1	Device is controlled by mountable device allocator
	2	***free***
DVDIBP	3	Device is a Batch PTY
DVCNET	4	Device is controlled by NETSER
DVOFLN	5	Device was offline the last time the device service routine polled
DVC2IO	6	Device can do simultaneous input and output
PDVTIM	7-9	Hung Constant (2^{n-1} seconds). If 0, the device cannot be hung.
PDVCNT	10-16	Countdown for hung device check
	24-35	Buffer size in words. This determines the size of the buffers set up by the Monitor for this device.

DEVIOS Bit Definitions

Device	I	I	I	I	I	I	I	I	I	I	I	I	I	I	I	I	Mag	I	I	I	Data
Dependent	O	O	O	O	O	O	O	O	O	O	O	O	O	O	O	O	Tape	O	O	O	Mode
Bits	N	T	S	E	M	E	T	K	E	C	O	E	A				Den-	R	O	C	(0-
	D	B	T	G	P	R	E	T	N	T	T	N	R				sity	C	N		17)
		L			M	R	R	L	D									K			

*Device dependent bits - those listed are for the Magtape.

Bit	Label	
12	IOEND	Service routine has transmitted last data
13	IO	Out = 1; In = 0
14	IOSTBL	Device Error Flag (system)
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	Date error; e.g., parity error
21	IOBKTL	Block too large or block # too large
23	IOACT	Device active
24*	IOBOT	Beginning of magtape
25*	IOTENT	End of magtape

26*	IOPAR	1 means write even parity on magtape
27-28*		Magtape density
		00 = installation standard
		01 = 200 BPI
		10 = 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
32-35		Data Mode; e.g., ASCII, ASCII line, DUMP, etc.

DDB (cont.)

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	D	A	M	T	D	I	O
I	S	D	P	A	U	B	I	N	T	T	T	V	T	T	I	N	U
R	K	R	T	T	S	I	S	G	P	R	A	A	A	Y	R	T	O
I				C	E	U					L					N	G
0	3	6	9	12	15	18	20									35	

Bits	Label	Meaning
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
12	DVAVAL	1 if device is available to this job

13	DVMTA	Device is a magtape
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.

DEVSTA Byte Definitions

Word Label -----	Bits -----	Bit Label or Byte Pointer -----	Content -----
DEV SPL	0	DEPSPL	This DDB is for a spooled device
DEV ADV	1	DEPADV	Don't de-advance buffers on error intercept
DEV STA	2	DEPLEN	Device has variable length buffers
DEV STA	3	DEPRAS	Device has restricted assignment
DEV TYP	4-9	PDVTYP	Type of device (see DEVSTA word in Disk DDB)
DEV TYP	10	DEPIBC	Inhibit clearing of output buffers
	11-12		Unusual Spool Bits
DEV SPL	13	.SPCDR	Spooled card reader
DEV SPL	14	.SPCDP	Spooled card punch
DEV SPL	15	.SPPTP	Spooled paper tape punch
DEV SPL	16	.SPPLT	Spooled plotter
DEV SPL	17	.SPLPT	Spooled line printer
DEV AIO	18	DEPAIO	Program is doing asynchronous I/O
DEV AIO	19	DEPIND	Input not yet done (asynchronous)
DEV AIO	20	DEPOND	Output not yet done (asynchronous)
DEV AIO	21	DEPECS	Non-superseding enter done
DEV AIO	22	DEPMMSG	Device is controlled by MSGSER
DEV AIO	23	DEPEVM	Device needs EVM
DEV AIO	24	DEPDER	Disable error recovery
DEV AIO	25	DEPDEL	Disable error logging

DEVPRI	26-29	DEXPRI DEYPRI DEZPRI	Device priority (Disk)
DEVSTA	30-35	PDVSTA	Station number (node) of device

DEVXTR Byte Definitions

Bits	Meaning
0-17	If an MPX DDB, address of connected device table If a device controlled by MSGSER, flags (same definition as LH of USRJDA)
18-35	If an MPX DDB, number of connected devices If a device controlled by MSGSER, address of MPX DDB

DEVEVM Byte Definitions

Bits	Meaning
0-17	UVA of buffer
18-26	Exec. page number
27-35	Number of pages

DEVESE Byte Definitions

Bits	Meaning
0-8	Unused
9-17	PSI Vector Offset
18-26	Unused
27-35	Extended software error status

DEVJOB Byte Definitions

Bits	Byte Pointer	Meaning
0-17		Unused
18-26	PUNIT	Unit number (within device type)
27-35	PJOB	Job number this device assigned to

Name: DDB Disk device data block

Description: Contains information to translate UUO's into specific actions on a disk file. One DDB for each active user software channel. DSKDDB is the address of the prototype DDB.

Defined in: COMMOD, S

DEVNAM	SIXBIT device name	
DEVCHR *	HC Countdown	Buffer size
DEVIOS *	Monitor I/O status	User I/O status
DEVSER	Address of next DDB	Address of dispatch table
DEVMOD *	Device characteristic	Device modes
DEVLOG	SIXBIT logical device name	
DEVBUF	Output buffer header addr	Input buffer header addr
DEVIAD *	COD S L FNC EUN R	Addr current input buffer
DEVOAD *	S L RLC M E F R	Addr current output buffer
DEVSTS	Word for device CONI	
DEVSTA *	S A V R Type Spl Bits A I O N M E D L	Station
DEVXTR *	User virtual address of checksum word	LVL W Num in buf P
DEVEVM *	IEVM IEVMS	OEVM OEVMS
DEVPSI *	PSI interrupt enable bits	PSI deferred int. reason
DEVESE *	PSI interrupt vector off.	Extended err status
DEVHCW	Hardware characteristics word (Not presently implemented for disk)	
DEVCPU	Multi-CPU Information	

DEVLVI

DEVJOB *	Old FSN	Job Number	DEVFSN
DEVFIL	Filename on last LOOKUP/ENTER (arg to FNDFIL)		
DEVEXT	File extension on last LOOKUP/ENTER	Number blocks left in current group	DEVLFN
DEVPPN	PPN on last LOOKUP/ENTER		
DEVNBF	Number buffers swept for	Buffers not swept for	
DEVSBF	Saved value of DEVSBF		
DEVCSN	Cache sweep serial number		
DEVDMF	Current IOWD for dump mode		
DEVRET	0	Addr of current retrieval pointer in DDB	
DEVREL	Relative block number in file to read or write next		DEVDIR
DEVUNI	Addr of original UDB (where RIB is)	Addr of current UDB	
DEVUFB		Addr of UFD data block	
DEVSTF		Addr of NMB for father STF	
DEVBLK	Logical block number in unit to read or write next		
DEVRSU	-Num of unused pointer positions in RIB	Addr of access table entry for user channel	DEVACC
DEVMBF	IOWD to monitor buffer (0 if user channel has none)		
DEVFLR	Block number of file which first in-core ptr points to		
DEVFUN	Addr of UDB to which first in-core pointer points to		
DEVQUE	Addr of next DDB in PWQ or TWQ	Job number of DDB in LH in-core pointer points to	

DEVELB *	Error code First logical block in bad region	
DEVLRL	Last DEVREL Disk priority	DEVPR1
DEVSPN	SIXBIT spooled filename	
DEVSPM	Pointer to spooling parameter block	
DEVTRIB *	E CNT UNIT Cluster address of current RIB	
DEVUPP	"In-your-behalf" PPN	
DEVCUR *	R C Current unit doing I/O for this DDB	
DEVGEN	Generatation number of UDB Addr of core copy of RIBs	
DEVLNM	Current SFD looking for Logical name spec being used	
DEVRL1	First retrieval pointer (Same format as RIB)	
DEVRL2	Second retrieval pointer	
	/ Retrieval pointers (Default total number is 6)	/
	/	/
DEVRLN	Last retrieval pointer in core	

* Byte definitions given below

12	IOEND	End of file reached at interrupt level.
13	IO	Doing output (Ø if doing input).
14	IOSTBL	Device error (system)
15	IOSRST	RESET or RELEASE done on spooled device.
16	IOSHMS	Hung device message already typed.
17	IOW	Job is in IO Wait due to the request in this DDB.
18	IPIMPM	Improper mode (miscellaneous software detected errors).
19	IODERR	Device error.
20	IODTER	Data error; e.g., parity error or checksum fail.
21	IOBKTL	Block too large (also illegal unit, quota overdrawn, or RIB overflow)
22	IODEND	User has reached end of file.
23	IOACT	This DDB contains a current IO request.
29	UDSD	SUSET. UUU request to write disk pack headers.
30	IOCON	Synchronous IO - transfer only one buffer per IN or OUT UUU.
31	IOWC	Do not compute word count on output.
32-35	PIOMOD	Data mode.

Legal modes are

- Ø - ASCII
- 1 - ASCII line
- 2 - Packed image
- 3 - Byte mode
- 10 - Image mode
- 13 - Image binary mode
- 14 - Binary
- 15 - Scope dump mode

- 16 - Dump by records
- 17 - Dump across records

Disk Device Data Block (Continued)

DEVMOD Bit Definitions

[illegible]

Bit	Label	Meaning
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 paper tape reader.
11	DVDTA	Device is DECTape.
13	DVMTA	Device is a magtape.
14	DVVTY	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. See PIOMOD.

Disk Device Data Block (Continued)

DEVIAD Byte Definitions

+-----+ COD FNC EUN Ø R Current Input Buf Adr +-----+										
Ø	2	3	4	5	8	12	13	14	17	35
Word Label	Bits				Label or Byte Pointer		Content			
DEV COD	Ø-2				DEY COD		File status code as in UNISTS. See below.			
DEV SCN	3				DEY SCN		Scanning switch for SFD's.			
DEV LPC	4				DEY LPC		Last RIB pointer is in core.			
DEV FNC	5-8				DEY FNC		Code for highest function permitted.			
DEY EUN	9-12				DEY EUN		Logical unit within STR on which error begins.			
DEY IAD	13						Ø - Indirect addressing bit.			
DEY IAD	14-17						Index field bits for AC R.			
DEY IAD	18-35						Address of second word of buffer to be filled next. (Relative address within job's area.)			

File Status Codes

Value	Symbol	Meaning
Ø	ICOD	Idle
1	SWCOD	Seek Wait
2	SCOD	Seeking
3	PWCOD	Position Wait
4	PCOD	Positioning
5	TWCOD	Transfer wait
6	TCOD	Doing Transfer

NOTES:

1. Seek means to move the access arms without having a transfer ready to initiate. Position means to move the access mechanism to the position required for a waiting transfer.
2. AC R will contain the absolute address of the job's core area.
3. Function codes for DEVFNC are listed below:

Values	Symbol	Meaning
0		Code not yet stored.
1	FNCExc	Execute only
2	FNCRED	Read
3	FNCALL	Allocate
4	FNCDLL	Deallocate
5	FNCAPP	Append
6	FNCUPD	Update
7	FNCCRE	Create
10	FNCSUP	Supersede
11	FNCTRN	Truncate
12	FNCCAT	Change attributes other than privilege, name, and directory
13	FNCDEL	Delete
14	FNCCNM	Change name
15	FNCCPR	Change privilege

Disk Device Data Block (Continued)

DEVOAD Byte Definitions

+-----+-----+-----+-----+-----+-----+-----+-----+-----+-----+-----+-----+-----+-----+-----+-----+																	
X			RLC			FSN			Ø			R			Cur Out Buf Adr		
+-----+-----+-----+-----+-----+-----+-----+-----+-----+-----+-----+-----+-----+-----+-----+-----+																	
Ø			1			2			8			9			12		
															14		
															17		
															18		

Disk Device Data Block (Continued)

DEVSTA Byte Definitions

+-----+-----+-----+-----+-----+-----+-----+-----+-----+-----+-----+-----+-----+-----+-----+-----+															
SPL	CPT	ADV	LEN	TYPE	IBC	SPOOL	AIO	IND	OND	ECS	MSG	EVM	DER	DEL	STA #
+-----+-----+-----+-----+-----+-----+-----+-----+-----+-----+-----+-----+-----+-----+-----+-----+															
0	1	2	3	4	9	10	11	17	18	19	20	21	22	23	24 25 30 35

Bits	Pointer	Meaning
------	---------	---------

0	DEPSPL	This DDB is for a Spooled Device
1	DEPCPT	Read file in 10/11 compatibility mode
2	DEPADV	Don't advance buffers on error intercept
3	DEPLEN	Variable length buffer capability
4-9	DEYTYP	Type of device.

00 - DSK (.TYDSK)	15 - MPX device (.TYMPX)
01 - DTA (.TYDTA)	16 - PA611R reader (.TYPAR)
02 - MTA (.TYMTA)	17 - PC11 reader/DC44 (.TYPCR)
03 - TTY (.TYTTY)	20 - PA611P on DC44 (.TYPAP)
04 - PTR (.TYPTR)	21 - LPC11 on DC44 (.TYLPC)
05 - PTP (.TYPTP)	22 - PC11 punch/DC44 (.TYPCP)
06 - DIS (.TYDIS)	23 - WTY on DC44 (.TYWTY)
07 - LPT (.TYLPT)	24 - Network TSK (.TYTSK)
10 - CDR (.TYCDR)	25 - DAS78 (.TYD78)
11 - CDP (.TYCDP)	26 - Remote data entry (.TYRDA)
12 - PTY (.TYPTY)	27 - Host command interpreter (.TYEST)

13 - PLT (.TYPLT) 30 - DRA (.TYDRA)
14 - External task (.TYXTR)

10	DEPIBC	Inhibit clearing output buffers
11-17	DEYSPL	Spool Bits
		11 - 12 Unused
		13 CDR
		14 CDP
		15 PTP
		16 PLT
		17 LPT
18	DEPAIO	Doing asynchronous I/O
19	DEPIND	Input not done yet
20	DEPOND	Output not done yet
21	DEPECS	Doing non-superseding enter
22	DEPMSG	This device controlled by MSGSER
23	DEPEVM	Device doesn't need EVM for I/O (KI only)
24	DEPDER	Disable error recovery on this device
25	DEPDEL	Disable error logging on this device
30-35		Station Number

Disk Device Data Block (Continued)

DEVTRIB Byte Definitions

+-----+-----+-----+-----+-----+-----+-----+-----+-----+-----+														
		RBC					RBU			RBA				
+-----+-----+-----+-----+-----+-----+-----+-----+-----+-----+														
0						8	9	12	13				35	

Bits	Label	Content
0		Set if extended RIB
1-8	DEYRBC	Count of RIB's
9-12	DEYRBU	Logical Unit within STR
13-35	DEYRBA	Cluster address within unit

DEVCUR byte definitions

+-----+-----+-----+-----+-----+-----+-----+-----+-----+-----+									
	R		C						Unit doing I/O
+-----+-----+-----+-----+-----+-----+-----+-----+-----+-----+									
0	1							17	18
									35

Word Label	Bits	Bit Label	Content
DEVPRAD	0	DEPRAD	Rename across directories in progress
DEVPRV	1	DEPPRV	Don't check privs in LOOKUP
DEVRRRC	2		Auto-rewrite of RIB on change
	6		On if RIB has changed
	7		On if physical only I/O was specified in INIT
DEVCUR	18-35		Current unit doing I/O for this DDB

Disk Device Data Block (Continued)

Retrieval Pointer

```

+-----+
| Cluster Count |      Checksum      | Cluster Adr |
+-----+

```

Widths of these fields are defined symbolically, and may be different for each file structure. Byte pointer is in STR Data Block.

Field	Byte Pointer
Cluster Count	STYCNP
Checksum	STYCKP
Cluster Address	STYCLP (23 bits maximum)

If cluster count = 0, the word actually is one of the following:

1. Pointer to new unit, if bit 18 = 1. Bits 19-35 specify logical unit number within file structure.
2. EOF flag, if whole word is zero.

Name: Magtape Device Data Block

Description: Contains information to perform I/O on a magtape. One for each tape drive on a system.

Defined in: COMDEV

See also: Magtape Unit Data Block, Magtape Kontroller Data Block

Format:

DEVNAM	SIXBIT Device Name
DEVCHR*	Device Characteristics
DEVIOS*	Monitor I/O Status User I/O Status
DEVSER	Next DDB Device Service Routine
DEVMOD*	Device Characteristics
DEVLOG	Logical Device Name
DEVBUF	Out Buffer Header In Buffer Header
DEVIAD*	Frame Count R Current In Buffer Adr
DEVOAD*	Frame Count R Current Out Buffer Adr
DEVSTS	Word for Device CONI
DEVSTA*	See bit definitions
DEVXTR	# Input Buffers
DEVEVM	Word to keep track of which EVM this device has
DEVPSI	Word to keep track of software interrupts
DEVESE	Extended Software Error Statistics
DEVHCW	Hardware Characteristics

DEVJOB	Job and Unit Number		TDVKDB
TDVUDB	UDB Pointer	KDB Pointer (prime)	
TDVSTS	Unit Status Info		
TDVIOR	IORB to Wait For		
TDVNBFB	Number of Buffers Swept For		
TDVSBFB	Saved TDVNBFB		
TDVCSN	Number Buffers swept for	# I/O requests for bffrs	
TDVSUL	Saved User Upper Limit		
TDVSL	Saved User Lower Limit		
TDVSM	Saved M for Dump Mode		
TDVREM	Remainder for Mode 16		

Name: DEVDSP DEVICE DISPATCH TABLE

Description: 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 nine entries are present in each device service routine. Tables which contain the additional entries are referred to as long dispatch tables.

Defined In: Device Service Routines

Used By: Device Service Routines

Format:

Label of entry (Relative to DEVDSP)	Relative Adr	Function
DOFL	-5	Is device off line ?
DDVO	-4	DEVOP. UUO
DSZ	-3	Return Buffer Size
DINI	-2	Device and service routine initialization
DHNG	-1	"Hung device" action
DRL	0	Release (Table Base Adr)
DCL	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

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

Notes:

1. 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.
2. The device dispatch table labels are normally indexed by AC T4, which must contain the base address of the appropriate table. The dispatch is usually performed by a PUSHJ P, Dxxx(t4) where Dxxx represents the label of the appropriate table entry.
3. Before attemptint 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.

4. The actual names of the tables are xyzDSP where xyz is the three letter name of the device (e.g., PTRDSP for the paper tape reader).

Name: DISP COMMAND DISPATCH TABLE

Description: 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.

Defined In: COMCON (local)

Used By: COMCON

Format:

N	N	E	N	C	T	T	T	N	N	N	C	N	N	I	N	N	N		
O	O	R	O	M	T	T	T	O	O	O	U	B	X	N	O	O	O		
C	J	R	M	W	Y	Y	Y	P	C	I	S	A	O	C	R	A	L		
O	O	F	E	R	R	R	R	E	R	N	T	T	N	O	U	C	O		
R	B	L	S	Q	N	N	N	R	L	C	M	C	L	R	N	T	G		
E	N	G	S		W	C	U		F	K	R	H	Y				I		
																	N		
																		Adr for Cmd1	
																		Adr for Cmd2	
																		Adr for Cmd3	
																		-	
																		-	
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	ERRFLG	Command Error
3	NOMESS	No command response ever
4	CMWRQ	Reque job after command wait
5	TTYRNW	Set TTY to user level and I/O. Wait if in IOW before

6	TTYRNC	Keep TTY in monitor mode and start job.
7	TTYRNU	Set TTY to user and start job
8	NOPER	No printing of period
9	NOCRLF	No printing of CR-LF
10	NOINCK	No check for job initialization (JNA=0)
11	CUSTMR	Reserved for customer definition
12	NBATCH	Not legal for BATCH job
13	NXONLY	Not legal if execute only program
14	INCOR	Job must be in core if it has core
15	NORUN	Job must not be running - C required
16	NOACT	Command must wait until job devices not active
17	NOLOGIN	Job need not be logged in

Bits 0, 1 and 12-17 checked before dispatching to command.
Bits 2-11 checked after return from command.

6	DSCDLF	Dialler Fail - Unsuccessful dialler attempt
7	DSCDLC	Dialler Complete - Successful dialler action
8	DSCEON	End of Number - Sent all digits to dialler
12-17	DSTIMP	Time field - for functions which require timing
18-35		Line number for this data set

Name: EXE Format Save Files

Defined in: S.MAC

Used by: FILFND, FILIO

Description: The old .SAV, .LOW, .SHR, .HGH and .XPN files have been replaced by the .EXE type in order to provide a unified format for saved core image files. This type of file consists of two distinct, but related portions:

- information about the structure of the file
- the data in the file

The file information is organized into a variable number of hunks. Each hunk begins with a word containing an identifier code in the left half and the length of the chunk in the right half. At this time, the defined codes are:

1. Directory: 1776
2. End: 1777

Other hunks may be added later as they become necessary. The save file starts with a directory followed by one or more pages of data. All pages of data are stored on page boundaries in exactly the form they will be loaded into the virtual address space. There is no zero compression within pages; only entire pages will be compressed.

The format of the directory is:

0		1776		size of directory	
1		flags		file page number	
2		repeat		process page number	
/		enough data to describe all pages			
/					

Word 0 has 1776 in the left half and the size of the directory in words in the right half.

Word 1 has flag bits in the leftmost nine bits and a file page number in the right 27 bits. If the file page number is zero then the page is allocated but zero. The flag bits are:

```

bit 0   Set if this is part of the hiseg
bit 1   Set if this page is sharable
bit 2   Set if the page may be written
bit 3   Page is concealed (access by PORTAL only)

```

Word 2 has a repeat count in its leftmost nine bits for one less than the number of consecutive file and memory pages described by this pointer. The rightmost nine bits describe the process address into which this page should be loaded. Descriptors in the same format as words 1 and 2 are provided for every contiguous piece of the process space with identical access bits.

A typical save file directory for a program with a 2K lowseg and a 10K hiseg would contain:

```

001776,,000005
100000,,000001
003000,,000000
600000,,000005
023000,,000400
001777,,000001

```

followed by 506 words of zeroes.

.EXE files were implemented to provide an extendable description of a core image not limited to 256K. This is especially useful in taking crash dumps of systems greater than 256K. Aside from this, the basic advantages include only 1 LOOKUP to get a program as opposed to as many as 3 previously, and the elimination of the overhead associated with expanding zero-compressed core images and compressing them on a SAVE. The disadvantages are that .EXE files require more disk space because of the directory and the elimination of zero compression within pages and that an additional page, the directory, must be read when the core image is fetched.

If bit 21 in the Monitor word CNFST2 is a 1, the SAVE and SSAVE commands generate .EXE files instead of .SAV, etc. files. This bit may be set by the MONGEN parameter M.EXE.

Name: FETTBL TABLE OF BITS FOR MONITOR FEATURES

Description: Contains bits for feature test switches.

Defined In: UUOCON (local symbol)

Used By: UUOCON

GETTAB Table No.: 71(octal)

Format:

Monitor Non I/O Features Which Directly Impact CUSPS
Real Time and Scheduling Features
Command Features
Accounting Features
Non I/O Features Not Related To Debugging
Non I/O Debugging Features
Disk UUO Features
Disk "Invisible" Features
Scanner Features
Other Peripherals
Other Peripherals (second word)
Disk Invisible Features (second word)
Disk UUO Features (second word)
Monitor Non-I/O Features Related to Cusps (2nd word)

Name: Fragment Table

Description: Contains data needed in order to swap in a fragmented core image.

Set up as required when segments are swapped out.

JBTSWP points to first fragment table for a given segment.

Size		U	Address
0	17	21	24
			35

Bits	Content
0-17	Number K in fragment
21-23	Unit index is SWPTAB
24-35	Logical K within unit where fragment starts.

Notes:

1. A zero word indicates end of table.
2. Fragment tables are built from four word blocks. If more than one block is required, and they are not contiguous, the last word of the preceding block contains -1 in the left half and the address of the next table in the right half.

Name: HOM HOME BLOCK

Description: Block on each disk unit or pack which contains vital statistics that cannot be "built in" when a monitor is generated. These are primarily parameters of the unit or pack and the STR to which it belongs.

Defined In: COMMOD

Used By: ONCMOD, REFSTR

Format:

HOMNAM~3	SIXBIT /HOM/	Written by MAP Program.
HOMHID~3	SIXBIT Unit ID	Written by MAP Program.
HOMPHY~3	Physical Disk Adr this Blk on this unit	Physical Disk Adr other Home Blk on this unit
HOMSRC	Position of this STR in System Search List -1 means not in System Search List	
HOMSNM~4	SIXBIT STR Name	Ø = Not in File Structure
HOMNXT~5	Unit ID for next unit in STR	Ø = Last or only unit
HOMPRV~5	Unit ID for previous unit in STR	Ø=Last or only unit
HOMLOG	SIXBIT Logical Unit # within STR	
HOMLUN~5	Logical Unit # within STR	
HOMPPN	Proj-Prog # which refreshed STR under timesharg or Ø	
HOMHOM~6	Logical Blk # for Home Block within unit	Logical Blk # within Unit for extra Home Block
HOMGRP	No blocks per group to try for on output (Not clusters)	
HOMBSC	No blocks per supercluster on this unit	
HOMSCU	No superclusters per unit	

HOMCNP	Byte Ptr for cluster count in Retrieval Pointers	
HOMCKP	Byte Ptr for checksum in Retrieval Pointers	
HOMCLP	Byte Ptr for cluster adr in Retrieval Pointers	
HOMBPC	No Blocks per cluster for this STR	
HOMK4S	No K Words for swapping on this unit Ø means no swapping	
HOMREF~7	Non-Zero if file must be refreshed	
HOMSIC	No SAT Blocks in core	
HOMSID	Unit ID of next unit in active swapping list Ø if last or not in active swapping list	
HOMSUN~8	Logical Unit # in active swapping list	
HOMSLB	First logical block # for swapping on this unit	
HOMCFS	Swapping Class for unit	
HOMSPU	No SAT Blocks per unit	
HOMOVR	Overdraw limit per user on this STR	
HOMGAR	Upper bound on total reserved blocks guaranteed to users	
HOMSAT~9	SAT.SYS*	HOMTAB
HOMHMS	HOME.SYS*	
HOMSWP	SWAP.SYS*	
HOMMNT	MAINT.SYS*	
HOMBAD	BADBLK.SYS*	
HOMCRS	CRASH.EXE*	
HOMSNP	SNAP.SAV*	

HOMRCV	RECOV.SYS*	
HOMSUF	SYS [1,4].UFD*	
HOMPUF	PRINTR [3,3].UFD*	
HOMMFD	MFD [1,1].UFD*	
HOMPT1	First retrieval ptr for MFD for structure this unit is in	
HOMUN1	Logical unit # where MFD starts	
HOMLEN	Table of lengths of files created by refresh - 6 words Needed by CRS, SNP, RCV AND UFDS (in that order)	
HOMUTP	Unit type on which HOME block was written (see UNYUTP)	
HOMRIP	Used by RIPOFF	
HOMKLB	20 words used by console front end in KL10 systems	
HOMKLE	Used to find files for Bootstrap/dump	
HOMK4C	K for CRASH.EXE	
HOMPVS	Word containing bit which says private STR	
HOMSDL	Postion of STR in system dump list	
HOMEND	Last word with valid data	
HOMVID	Volume ID (12 PDP11 bytes)	
HOMOWN	Owner name	
HOMVSY	System type (TOPS-10)	
HOMCOD	0	(Unlikely code) 707070
HOMSLF	0	This Blk # in unit (Not Cluster)

* Logical Block Number within STR of first RIB.

HOMPHY Byte Definitions

+-----+-----+-----+-----+-----+-----+																														
	C								S		A		C								S		A							
+-----+-----+-----+-----+-----+-----+																														
0								7	8	12	13	17	18								25	26	30	31	35					

Byte	Meaning	
C		Cylinder address
S		Surface
A		Sector address

Special Notes:

1. An extra copy of the home block is maintained on each unit in case the original becomes unreadable. These two home blocks which must be at specific places known to the software. Currently, they are blocks 1 and 10 (decimal) of each unit.
2. Each file structure has one home file HOME.SYS. It is a "Sparse File" with retrieval information in first block of each group. However, only the second block of each group (1 cluster) has data, the remaining ones (if any) are 0. The home file is constructed so that each group is on a separate unit where 'Unit' is a side of a Bryant MDF Disk, one cut of up to four Burroughs fixed head disks, etc.
3. These words are the only ones altered by the MAP program. They are not modified by the Monitor.
4. Ignored by Monitor when a pack is mounted. File STRLST.SYS is final authority for the structure name.
5. Checked by Monitor and PULSAR whenever pack is mounted.

6. Note the restriction that home blocks limited to the first 262,000 blocks on a unit. Knowing logical block address of home blocks is useful only to recovery programs when file structure has been clobbered. Most programs will read home blocks via file HOME.SYS.
7. Some parameter for this unit has been changed. Set by disk Once Only code, checked when system is started and by PULSAR when pack mounted.
8. -1 indicates unit is not in Active Swapping List. (May be changed at Once time without requiring refreshing provided swapping space allocated at refresh time.)
9. The rest of the entries in the home block are written by REFRESH.

Name INTTAB INTERRUPT ROUTINE TABLE

Description: 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.

Defined In: COMMON

Used By: SYSCHK, ONCE

Format:

Dev 1 Entry
Dev 2 Entry
:
:
:
Dev N Entry

Format of each entry:

Word 0

Nr DDB's	PI Channel	Int.Rout.Adr
0 1	8 9	17 18 35

Bit 0 = 1 for any type of DEctape routine.

Word 1

+-----+-----+-----+-----+									
Sta. Loc.		DDB Length				DDB Adr.			
+-----+-----+-----+-----+									
0	4	8	9			17	18		35

Bits 0-8 contain the station number of the device.

Bits 9-17 contain length of DDB for devices with multiple units.

Special Notes:

1. INTTAB entries are set up by the ASGINT and ASGSV1 macros in COMMON according to parameters specified in HDWCNF.MAC.
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.
4. The interrupt routine address is the address of the CONSO in the skip chain.

Name: JBTADR JOB ADDRESS TABLE

Description: Tells core address and length for each segment in core. One entry for each job number and each high segment number.

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

Used By: FILIO, FILUO, CDPSE, CLOCK1, COMCON, CORE1,
 DISSER, DTASRN, ERRCON, KISER, KILOCK, METCON,
 PSISER, PTYSER, REMDL, SCHED1, SEGCON, SWPSE,
 TYPSE, UUOCON, VBCSE, VMSE, SYSINI, REFSTR

GETTAB Table No.: 1

Format:

Job 0		
1	Length -1	Core Address
2	"	" "
-		
-		
N		
High Seg N+1	"	" "
N+2		
-		
-		
-		

Special Notes:

1. The entry for a job which is swapped out, or being shuffled, is zero.
2. JBTDAT is the same table.
3. During a BLT of the low or high segment, this word contains the source relocation.
4. The right half of JBTADR contains 371000 unless the segment is locked. JBTADR then contains the core addresses of the segment.

Name: JBTAD2 Second Job Address Table

Description: This table contains the first physical page number for each job's low segment. This table is kept so that the monitor can obtain the page number when the job is not currently mapped or the UPMP is in error. There is one entry per job.

Defined in: COMMON

Format:

Job 0	First Phys. Page		
Job 1			
Job N			
	0	22 23	35

Name: JBTCHK SEGMENT CHECKSUM TABLE

Description: 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.

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

Defined In: COMMON

Used By: SWPSER

Format:

Job 0	
Job 1	Checksum 1
Job 2	Checksum 2
Job N	Checksum N
High Seg N+1	Checksum N+1
High Seg N+2	Checksum N+2
	"

QUEUE	-2		"		"
NUMBER	-1		"		"
JBTCQ	0				
	1		# OF PREV JOB		# OF NEXT JOB
JOB	2		"		"
NUMBER	3		"		"
	4		"		"
	5		"		"

Special 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.

Name: JBTCSQ JOB SUBQUEUE TABLE

Description: Contains the PQ2 Subqueues which partition PQ2 into a number of ordered classes. Each subqueue (or subclass) is an ordered subset of PQ2 job numbers, all of the same class.

Defined In: COMMON

Used By: SCHED1, UUOCON, SYSINI

GETTAB Table No: 125

Format:	-SQN	-----+-----	-----+-----
		Last Job in Subqueue	First Job in Subqueue
		-----+-----	-----+-----
		-----+-----	-----+-----
	-SQ1		
Subqueue		-----+-----	-----+-----
Number	-SQ0		
		-----+-----	-----+-----
	JBTCSQ		
		-----+-----	-----+-----
	1	# of Prev. Job	# of Next Job
		-----+-----	-----+-----
Job	2		
Number	3	-----+-----	-----+-----
		-----+-----	-----+-----

- Special notes:
1. Every job number in PQ2 will be in one and only one subqueue.
 2. It is possible for a subqueue to contain no job numbers. In this case, the subqueue header entry contains a pointer to itself, the negative subqueue number, in both halves.
 3. The number of classes partitioning PQ2 is specified at MONGEN time.
 4. When the Scheduler scans these subqueues, it does so in order, starting with SQ0 (Class 0).

Name: JBTDEV HIGH SEGMENT DEVICE TABLE

Description: One entry for each high segment.

The entry is the high segment's physical
device name or file structure number.

Defined In: COMMON

Used By: SEGCON, UUOCON

GETTAB Table No.: 24(octal)

Format: JBTDEV

Job 0	
	(JOB number entries are not used, and overlap the high end of the JBTPPN table.)
Job N	
High Seg N+1	Physical Device Name
N+2	" " "

Name: JBTJIL JUST SWAPPED IN LIST

Description: Special queues containing jobs in PQ2 which have just been swapped in and have not expired their time slice. The format of the queue table is similar to JBTCQ, that is indexed in the positive direction by job number and the negative direction by queue number. The two queue headers are regular (time/sharing) and background batch.

Defined in: COMMON

Used by: SCHED1, SYSINI

Gettab table no.:

Format:

JBTBBQ		
JBTJIQ	LAST JOB IN QUEUE	FIRST JOB IN QUEUE
JBTJIL		
1		
2		
3	No. OF PREVIOUS JOB	No. OF NEXT JOB
n		

Special notes:

1. Jobs in JBTJIL are also in one of the master queues described by JBTCQ.
2. The scheduler will search the time sharing queue for job selection ahead of PQ2 a certain percentage of the time. This percentage is called the response fairness factor and may be modified from the default value of 10% by the SCHED. UUO.

Name: JBTJRQ Job Requeue List

Description: A singly linked first in last out list of jobs waiting to be requeued. The right half of JBTJRQ points to the first entry in the queue. The remainder of the table is indexed by job number with each entry containing the job number of the next job in the queue.

Defined in: COMMON

Used by: CLOCK1, SCHED1

Gettab Table No.:

Format:

JBTJRQ	No. of 1st JOB IN THE LIST
1	
2	
3	
JOB	

Name: JBTLIM JOB TIME LIMIT TABLE

Description: One entry per job, indexed by Job Number.

Bit definitions are found in S.MAC.

JBTLIM is assembled only if FTTLIM is non-zero.

Defined in: COMMON

Used By: CLOCK1, COMCON, SCNSER, UUOCON

GETTAB Table No.: 40(octal)

Format: JBTLIM:

+-----+-----+-----+-----+-----+-----+-----+-----+-----+-----+-----+-----+												Job 0
+-----+-----+-----+-----+-----+-----+-----+-----+-----+-----+-----+-----+												Job 2
+-----+-----+-----+-----+-----+-----+-----+-----+-----+-----+-----+-----+												Job 3
+-----+-----+-----+-----+-----+-----+-----+-----+-----+-----+-----+-----+												Job N
+-----+-----+-----+-----+-----+-----+-----+-----+-----+-----+-----+-----+												
0	1			9	10	11	12					35

Bit	Label	Content
0		Unused
1-9	JB.LCR	User Core Limit
10	JB.LBT	Batch Job
11	JB.LSY	Set when file gotten from SYS. Used by BATCON
12-35	JB.LTM	Time limit in jiffies (0 = Infinite) Loaded by the SET TIME COMMAND

Name: JBTLOC JOB LOCATION TABLE

Description: One entry per job number; indexed by job
number.

The first entry in the table is the Central
Station Number.

The table is used by the LOCATE command and
by the NEWJOB routine in COMMON to locate a
job at the station number of its controlling
terminal.

JBTLOC is assembled only if FTREM is
non-zero.

Defined In: COMMON

Used By: FILUOO, CLOCK1, COMCON, PTYSER, REMDLX, REMDSX,
UUOCON, SYSINI

GETTAB Table No.: 26(octal)

Format:

+-----+-----+	
Central Site Station Number	
+-----+-----+	
Job Location	Job 1
+-----+-----+	
Job Location	Job 2
+-----+-----+	
Job Location	Job 3
+-----+-----+	
Job Location	Job N
+-----+-----+	

Name: JBTNAM JOB NAME TABLE

Description: Tells name of each segment, typically, the file it came from. One entry for each job number or high seg number. Indexed by job number or high seg number.

Defined In: COMMON

Used By: CLOCK1, COMCON, SCNSER, SEGCON

GETTAB Table No.: 3

Format:

Job 0	
Job 1	Segment Name 1
Job 2	Segment Name 2
Job 3	Segment Name 3
	-
	-
Job N	Segment Name N
High Seg N+1	Segment Name N+1
High Seg N+2	Segment Name N+2
.	.

Special Notes: 1. JBTPRG is the same table.

Name: JBTOLS JOB OUTPUT LIST

Description: Special queues containing jobs in PQ2 which have exceeded their time slice and consequently have become eligible for swap out. The format of the queue table is similar to that of JBTCQ, in that the table is indexed in the positive direction by job number, and the negative direction by queue number. The two queue headers are regular output (time/sharing) and background batch.

Defined in: COMMON

Used by: CLOCK1, SCHED1

Gettab Table No.:

Format:

JBTQBQ		
JBTOLQ	LAST JOB IN QUEUE	FIRST JOB IN QUEUE
JBTOLS		
2		
3		
4		

Special notes:

1. Jobs in JBTOLS are also in one of the master queues described by JBTCQ.
2. Once a job enters JBTOLS it has a higher priority for swap out.

Name: JBTPIA SOFTWARE PROGRAM INTERRUPT TABLE

Description: Contains flags and pointers to program interrupt table for each job which is using the programmed software interrupt facility.

Defined In: COMMON

Used By: CLOCK1, PSISER, UUOCON

Format:

0	1	2	17	18	35
					Job 0
				Adr. of Program Interrupt Table	Job 1
					Job 2
					.
					.
					Job N-1
					Job N

Bit 0 If set, the PI system is turned on.

Bit 1 If set, the PI system is turned off until a DEBRK by the current job.

Name: JBTPPN PROJECT PROGRAMMER NUMBER TABLE

Description: Contains the project programmer number for each logged in job. If the high segment references disk, contains the directory name.

Defined In: COMMON

Used By: FILFND, FILIO, FILUOO, CLOCK1, COMCON, SCNSER, SEGCON, UUOCON

GETTAB Table No.: 2

Format:

Job 0		
Job 1	Project No.	Programmer No.
Job N		
Hi seg 0	Directory	Name
	:	:
Hi Seg M		

14	PVLOCK	Job allowed to use LOCK UUO
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
18-35		Reserved for special customer defined privileges

Special Notes: JBTPRV is assembled only if FTPRV is non-zero.

Name: JBTRQT JOB RUN-QUEUE TIME TABLE

Description: One entry per Job number. Contains a count of the number of ticks a job was in PQ1, PQ2, or PQ3.

Defined In: COMMON

Used By: CLOCK1, SCHED1, UUOCON

GETTAB Table No.: 53 (octal)

Format:

"WANT TO RUN TIME"	Job 0
	Job 1
	Job 2
	Job 3
.	
.	
.	
	Job N

Special Notes: JBTRQT is assembled only if FTRSP is non-zero.

Name: JBTRSP JOB RESPONSE TIME TABLE

Description: This table is assembled if FTRSP is assigned a non-zero value in F.MAC.

One entry per job number including the null job. The null job entry is not referenced.

Defined In: COMMON. Bits are defined in S.UNV.

Used By: CLOCK1, UUOCON

GETTAB Table No.: 50(octal)

Format:

0	1	2	3	4	35
J J J J J					
R R R R R					
.					
R R R R R					
C O I R X					
R R R R R					
					Job 0
				Time job started waiting	Job 1
					.
					.
					Job N

Bit	Name	Description
0	JR.RCR	Recorded first CPU use
1	JR.ROR	Recorded TTY output UUO
2	JR.RIR	Recorded TTY input UUO
3	JR.RRR	Recorded CPU quantum exceeded requeue
4	JR.RXR	Recorded first of above 3 responses (not JR.RCR)

Bits set to 1 when type of response is recorded, to 0 when user types in.

Special Notes: The entries in the left half of this table are the up times at which the user began to wait for system response. If zero, the user is not waiting. Entries are made in the table when the user's job comes out of TTY Input Wait or types a command which runs a job. The entries are used by the scheduler to calculate SYSRSP, the total system response time, and NUMRSP, the number of system responses.

Name: JBTRTD REAL TIME DEVICES

Description: Contains real time status bits in the left half for HPQ's and hibernate-wake. The right half contains a count of the real time devices owned by the job.

Defined In: COMMON

Used By: CLOCK1, IPCSER, PTYSER, RTTRP, SCNSER, UUOCON, SYSINI

GETTAB Table No.: 37(octal)

Format:

0	26	27	35
+-----+-----+-----+			
			Job 0
+-----+-----+-----+			
			Job 1
+-----+-----+-----+			
Status Bits		Count	Job 2
+-----+-----+-----+			
			.
+-----+-----+-----+			
			.
+-----+-----+-----+			
			Job N
+-----+-----+-----+			

Bit	Meaning
0	Only an exec process can wake job
1	IPCF event enable
2-5	Console command setting of HPQ for job
6-9	Current HPQ position of job
10	Wake bit - set if wake job by HIBER
11	I/O activity enable
12	PTY activity enable
13	TTY activity enable - line mode
14	TTY activity enable - character mode
15-17	Hibernate protection code for job
23-26	Temporary HPQ (sharable device owner)
27-35	Count of number of real time devices owned by job

Name: JBTSCD JOB SUBQUEUE (CLASS) TABLE

Description: Contains each job's class indexed by Job number. One entry for each job number.

Defined In: COMMON

Used By: CLOCK1, SCHED1, UUOCON

GETTAB Table No.: 120

Format:

	0	13 . . . 17	35
Job 0		Class #	
Job 1			
Job N			

JBTSFD Sub-File Directory Table

Description: One entry per job holding search list and SFD information.

Defined in: COMMOD

Job 0									
Job 1				DEFAULT SFD NMB					
Job 2									
Job n									
	0	15	18				33	34	35

pts	Label	Meaning
0-15		Address of library PPN PPB
16	JBXPXY	Search NEW before SYS
17	JBPSYS	Search SYS after DSK
18-33		Location of default SFD NMB or 0
34	JBPUBF	Bits 18-33 point to a UFB
35	JBPSCN	Scanning is on

18-35

If SPYSEG set, highest physical address user may see. If no high segment, zero - otherwise, high segment number associated with this job.

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

Meddling means

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

Name: JBT SPL SPOOL CONTROL TABLE

Description: Contains input file name counter, devices being spooled, and disk priority for the job.

Defined In: COMMON

Used By: COMMOD, FILUUO, CLOCK1, COMCON, UUOCON

GETTAB Table No.: 36(octal)

Format:

			Job 0
Input File Name Counter	DSK PR1	Spool DEV	Job 1
			Job 2
			.
			.
			Job N

Name: JBTSPS SECOND PROCESSOR STATUS

Description: Assembled in multi-processor systems to
to indicate second processor status. One
entry per Job number.

Defined In: COMMON

Used By: CP1SER, UUOCON

GETTAB Table No.: 54(octal)

Format:

Bits Set By Monitor	GETTAB Bits for 6 CPU's	Job #
		Job 0
		Job 1
		Job 2
		Job 3

Bit	Label	Meaning
4	SP.NR1	Not runnable on CPU1
5	SP.NR0	Not runnable on CPU0
10	SP.CJ1	Current job on CPU1
11	SP.CJ0	Current job on CPU0
12	SP.ROP	Job forced to policy CPU when policy CPU dies
14	SP.CC1	CTRL/C on CPU1 - stop running job on slave
24-27	-	Reserved for future CPU's
28	SP.SC1	Set CPU command bit for CPU1 (OK as far as user is concerned to run on this CPU even if stopped or not scheduling.)
29	SP.SC0	Set CPU Command bit for CPU0
30-33	-	Reserved for future CPU's
34	SP.CR1	Can run on CPU1
35	SP.CR0	Can run on CPU0

Name: JBTSTS JOB STATUS TABLE

Description: 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.

Defined In: COMMON. Bits defined in S.UNV.

Used By: FILFND, FILIO, FILUWO, CLOCK1, COMCOM, CORE1, CP1SER, DISSER, ERRCON, IPCSER, KISER, KILOCK, KLSER, KSER, METCON, MTXSER, NULSEG, PSISER, PTYSER, REMDSX, RTTRP, SCHED1, SCNSER, SEGCON, UWOCON, VBCSER, VMSE, SYSINI

GETTAB Table No.: 0

See also: AVALTB, QBITS, Job Queues and Wait State Codes

Format:

Job 0	
Job 1	Status Bits
Job 2	" " " " " "
-	
-	
Job N	" " " " " "
High Seg N+1	Status Bits
High Seg N+2	" " " " " "
High Seg N+3	" " " " " "
-	
-	
High Seg M	

See Bit definitions on following pages.

JBTSTS (cont.) LH OF JOB NUMBER ENTRIES

0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17
	R	C	J	J	J	N	S	S	N	C					J	J	J
	U	M	X	N	E	S	H	W	S	N		WAIT		L	R	A	
	N	W	P	A	R	W	F	P	H	T		STATE		O	Q	C	
		B	N		R	P			F	R		CODE		G		C	
									L							T	
								C		(PJBSTS)							

RH of Job number entries

18	19	20	21	22	23	24	25	26	27	28	29	30	31	32	33	34	35
	C	L	J	U	J	J		J	J	J		J	J	J	J	J	J
	L	O	D	T	D	S		S	S	S		S	S	S	S	S	S
	K	K	C	R	C	.		.	.	.		.	.	.	.	.	.
	R			P	O	D		X	R	M		D	A	X	R	S	N
					N	E		O	U	P		P	S	O	Q	F	X
						P			U	E		M	A	R	R	L	M

Bit Number	Label	Meaning
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	JXPN	Job must be swapped out because it is expanding, and there is not enough room in core.
3	JNA	This job number is assigned.
4	JERR	Monitor detected error has occurred.
5	NSWP	Job is not to be swapped.

Bit Number	Label	Meaning
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	CNTRLC	CTRL-C typed while in monitor mode and not in TTY input wait - delay stopping job.
10-14	PJBSTS	Specifies transfer table to requeue job for current conditions. See AVLQTB, QBITS for the names of the tables. These are system dependent values as not all queues will be assembled for all systems. The value here may be used to count 12-bit bytes into STSTBL to get the SIXBIT queue name. See also Job Queues and Wait State Codes.
15	JLOG	Job logged in.
16	JRQ	Job has changed state and must be queued at clock level before rescheduling can take place.
17	JACCT	Privileged system CUSP which cannot be interrupted.
18	CLKR	Job has clock request in.
19	LOK	Job is being locked in core.
20	JDC	Job has typed "DCORE."
21	UTRP	Trap to USER on UUO exit (Reenter DDT).
22	JDCON	Job in CTRL-C state writing to continue from error.
23	JS.DEP	DAEMON Error Pause.
24		Undefined

25	JS.XO	EXECUTE only core image.
26	JS.RUU	RUN UUO or command in progress.
27	JS.MPE	Memory parity error for job.
28		Undefined
29	JS.DPM	DAEMON problem message needed.
30	JS.ASA	Use shadow AC's for UUO ARGS instead of user AC's.
31	JS.XOR	Run UUO or command in progress on execute-only file.
32	JS.RQR	Reset QUANTUM run time.
33	JS.SFL	Stop job if disk is full.
34	JS.NXM	Some page in user's address space is contained in non-existent memory.
35	JS.NTO	Non-blocking TTY output.

LH of High Seg entries

S	S	J	S	S	N	S	N	Hiseg										J	In-core
N	H	X	S	E	S	W	S	Access Privilege Bits										S	count
A	R	P	.	R	W	P	H											.	
S	N	S	R	P		F		Owner	Project	Universe	S								
E	Y																	F	
G	S																	E	
0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	35

Bit	Symbol	Meaning
0	SNA	This High Seg Number assigned.
1	SHRSEG	Sharable Segment.
2	JXPN	High segment is expanding and must be swapped out.
3	SS.SYS	Hiseg came from SYS:.
4	SERR	High Segment Swap Read Error.
5	NSWP	This high Seg is not to be swapped.
7	SWP	1 if segment swapped out or in transit.
8	NSHF	High segment cannot be shuffled.
9-17		Segment access privilege bits - same as disk file.
18	JS.SFE	Hiseg came from an EXE file.
19-35	JBYICC	In-core count of jobs using this high segment.

NOTE: SHRSEG bit is also kept in JBTSGN.

Name: JBTST2 JOB STATUS TABLE 2

Description: Contains status information about each job.
 (Extensions to the JBTSTS table) Indexed by
 job number.

Defined in: COMMON. Bits defined in S.UNV.

Used By: COMCON, SEGCON, UUOCON, CLOCK1, SCHED1,
 KASER, KISER, KLSER, KSER, FILFND

GETTAB Table No.: 117

Format:

	0	16	17	18	19	20	24	25	26	27	32	33	34	35
			E	M	A			E	R					
			X	I	B	EW Code		W	P	Job Queue #		O	R	S
			E	G	P			A	C			O	Q	N
								K						
								E						
Job 0														
Job 1														
Job 2														
Job N														

Bit	Label	Meaning
0	JS.IPQ	On if job is in a processor queue. Must be sign bit.
2-3	JS.DEB	Deferred echo bits
4	JS.OLS	Job owns locked structure
5	JS.SIP	On if swapping I/O is in progress for this job
6	JS.FPS	On if long KA-10 floating point instructions should be simulated FORCE was JXPN to avoid forgetting FORCE was cleared.

7	JS.NNQ	On if not to assign new quanta on swap-in. Set when job GETSEGS a swapped segment.																								
8	JS.BBJ	On if job is from background batch.																								
9	JS.CSQ	On if job is changing subqueue due to SCHED UUO.																								
10	JS.IGS	In GETSEG (doing a GETSEG UUO).																								
11	JS.HNG	Job stayed in FORCE too long, causing temporary or permanent system hang. Cleared when swapped in or out.																								
12	JS.BPR	Bypass program to run checking if set. The current RUN command is allowed despite .PDPGM (used for KJOB).																								
13	JS.FXO	File DAEMON made this core image execute-only.																								
14	JS.CFX	Call File DAEMON on program exit.																								
15	JS.HIB	This job is hibernating																								
16	JS.NCS	On if the cache bits are not to be turned on in this job's map.																								
17	JS.EXE	A new save (EXE file) is underway, or on a "GET", the LOOKUP on the EXE file failed. (Prevents EXE file from being LOOKUP'ed again for the low segment.)																								
18	JS.MIG	Job has migrated to another swapping unit when a disk controller is being taken off-line (FTDHIA).																								
19	JS.ABP	An address break page fault occurred because of a reference to this user's virtual address space during UUO processing.																								
20-24	JBYEWT	ESLEEP reason <table> <tr><td>1</td><td>EV.TKW</td><td>Tape Kontroller</td></tr> <tr><td>2</td><td>EV.REW</td><td>Rewind</td></tr> <tr><td>3</td><td>EV.LBL</td><td>Label processing</td></tr> <tr><td>4</td><td>EV.NET</td><td>Network device</td></tr> <tr><td>5</td><td>EV.IPC</td><td>IPCF system process receive</td></tr> <tr><td>6</td><td>EV.FEI</td><td>Front end device input</td></tr> <tr><td>7</td><td>EV.FEO</td><td>Front end device output</td></tr> <tr><td>10</td><td>EV.D60</td><td>DAS60 device (waiting for the PDP11)</td></tr> </table>	1	EV.TKW	Tape Kontroller	2	EV.REW	Rewind	3	EV.LBL	Label processing	4	EV.NET	Network device	5	EV.IPC	IPCF system process receive	6	EV.FEI	Front end device input	7	EV.FEO	Front end device output	10	EV.D60	DAS60 device (waiting for the PDP11)
1	EV.TKW	Tape Kontroller																								
2	EV.REW	Rewind																								
3	EV.LBL	Label processing																								
4	EV.NET	Network device																								
5	EV.IPC	IPCF system process receive																								
6	EV.FEI	Front end device input																								
7	EV.FEO	Front end device output																								
10	EV.D60	DAS60 device (waiting for the PDP11)																								
25	EWAKEB	EWAKE called (Wakeup waiting)																								
26	JS.RPC	If = 1, run program in .PDPGM on Control-C																								
27-32	PJBST2	Queue Number. Also called PJBST1 (for indexing by T1).																								
33	JS.OOO	User ran out of order. Stop when gives up last resource.																								

34 JS.TFO

Job forced out by timer.

35 JS.SCN

Job was scanned to run by at least one
CPU during last tick.

Special 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.
3. Disk Address Bit Definitions

Bits	Meaning
3-5	Index of Unit in SWPTAB (JBYSUN)
6-17	First Logical K on Unit (JBYLKN)
1-17	Core address of Fragment Table if fragmented.

Name: JBTUPM UPMP ADDRESS TABLE

Description: Contains the address for the user page map page for each job and high segment which is in core. This table is also called JBTHSA.

Defined In: COMMON

Used By: KISER, KILOCK, KLSER, SCHED1, SEGCON, SWPSER, UUOCON, VMSER, SYSINI

GETTAB Table No.: 100(octal)

Format:

0	8	9	17	18	22	23	35	
HSO		LSS		HSS		UPMP ADDR		Job 0
								Job 1
								Job 2
			:					:
								Job N
								Seg N + 1
			:					:
HSO						HSA		Seg M
0	8				23		35	

Description for Low Segment Entry

Bit	Pointer	
0-8	JBYSO	Virtual page number of hi-seg. origin
9-17	JBYSLSS	Number of pages less one in low segment the last time the map was set up
18-22	JBYSHSS	Number of pages less one in high segment
23-35		Physical page number of UPMP

Description of High Segment Entry

0-8	JBYSO	Virtual page number of high-seg. origin
23-35	JBYS	Physical page number of first page of high segment

Name: JBTVRT PAGING RATE TABLE

Description: Virtual Memory per job paging rates are
 ntained in this table.

Defined in: COMMON

Used By: UUOCON, VMSE

Format:

+	-----+	
	PAGE RATE	Job 0
	"	Job 1
		.
		.
	"	Job N
+	-----+	

Name: JBTWCH JOB WATCH TABLE

Description: This table is assembled if FTWATCH is assigned a non-zero value in S.MAC. It determines the specific watch parameters to be displayed. There is one entry per job number including the null job. The null job entry is not referenced.

Defined In: COMMON

Used By: COMCON, MTXSER, UUOCON

GETTAB Table No.: 35(octal)

Format:

	0	1	12	13	35
JBTWCH		CONDITIONS		TIME OF DAY	Job 0
					Job 1
					Job 2
					Job n-
					Job n

Bit	Name	Description
1	JB.WDY	Watch time of day started to wait
2	JB.WRN	Watch runtime when return to command level.
3	JB.WWT	Watch wating time when return to command level
4	JB.WDR	Watch number of 128 word disk block read.
5	JB.WDW	Watch number of 128 word disk blocks written.
6	JB.WVR	Watch Versions
7	JB.WMT	Watch number MTA performace statistics
8	JB.WFL	Watch file activity
9	JB.WLM	Long error messages

10	JB.WNM	Normal error messages
11	JB.WPM	Prefix error messages.
13-35		Time of day in Jiffies user started to wait.

Name: JDA JOB DEVICE ASSIGNMENT TABLE

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

The JDA is part of the job's UPMP. Each JDA contains 16 entries corresponding to the 16 "software channels" of a user job.

Indexed by channel number.

Defined In: COMMON

Used By: FILIO, FILUWO, COMCON, CORE1, DTASRN, MSGSER, PTYSER, SEGCON, UWOCON, VMSER

Format:

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

Bit	Label	Meaning
0	INITB	INIT or OPEN has been done
1	IBUFB	Input ringheader sepcified (by INIT).
2	OBUFB	Output ringheader specified (by INIT).

3	LOOKB	A LOOKUP has been done
4	ENTRB	An ENTER has been doen
5	INPB	An INPUT has been done
6	OUTPB	An OUTPUT has been done
7	ICLOSB	An input CLOSE has been done
8	OCLOSB	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 or SYSPPN on DSK.
12	RENMB	RENAME UUO in progress.
13	RESETB	RESET UUO in progress.
18-35		Address of Device Data Block for I/O on this software channel.

Notes:

1. If both LOOKB and ENTRB are on, the file is being accessed in update mode.
2. It is not valid for a user job to try to get its JDA in .JBJDA since the Monitor does not leave good data here when the user job runs.
3. Extended channel information is kept in a 64-word table in funny space pointed to by the .UPCTA word in the UPMP. If extended channels are not in use, the word is zero. Each entry in the table is formatted the same as entries in the JDA table.

Name: JOBDAT JOB DATA AREA

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

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

Defined In: COMMON

Used By: COMMON

Format:

Relative
Location

Label

0 - 17	.JBBAC	User AC's during UUO (16 words)
20 - 37	.JBBAC	Hardware AC's while job inactive (16 words)
40	.JBUUO	User UUO stored here
41	.JB41	User UUO branch instruction
42	.JBERR	Unused Error cnt for RPG
43	.JBENB	Unused User APR trap flags
44	.JBREL	0 Length of low seg
45	.JBPD1	Push down list (21 words)
72	.JBHCU	Highest I/O channel in use
73	.JBPC	Job PC when job inactive
74	.JBDDT	Unused Start Adr of DDT
75	.JBJDA	Job Device Assignment Table 16 words documented separately

114	.JBSDD	JOBDDT here on SAVE/Protected from I/O	
115	.JBHRL	First free loc in high seg	Length of high seg
116	.JBSYM	Symbol table pointer	
117	.JBUSY	Undefined symbol table pointer	
120	.JBASA	First free loc in low seg- when loaded	Program start address
121	.JBFF	Current first free location in low seg.	
122	.JBS41	.JB41 here on SAVE	
123	.JBEXM	Adr of last D or E command	
124	.JBREN	Address for REENTER command	
125	.JBAPR	Branch loc on user enabled APR error	
126	.JBCNI	APR conditions on APR trap	
127	.JBTPC	PC stored here on APR trap	
130	.JBOPC	Old PC stored here on START, DDT, REENTER, & CSTART commands	
131	.JBCHN	Used for FORTRAN Job chaining (root link)	
132	.JBFDV	DDB adr for FINISH command	
133	.JBCOR	Highest loc in low seg actually loaded	Low seg core assignment
134	.JBINT		Data block adr for error intercept
135	.JBOPS	Reserved for runtime operating system	
136	.JBCST	Reserved for customer	

.JBOV

137	.JBVER	Job version number
140		First loc in user's program area

Special Notes:

1. The actual tables are included at the beginning of each user's area.
2. Many of these words contain different values while a SAVE or GET is in progress, and therefore have several different labels.
3. Starting with 7.01, system sensitive locations for all machines are stored in the UPMP instead of the user's core image. If .JBxxx is stored in location yyy, VJBDAT will define the value of the symbol as -1000+yyy (octal).
4. The areas for storage of accumulators are used differently on a KA-10 monitor (6.03A) which has only one hardware block of accumulators. .JBBAC is used to store the user's accumulators on entry into monitor call (UUO) processing, and .JBDAC is used to store the accumulators (either exec or user depending on the level) when the job is not running.

Name: Job Queues and Wait State Codes

Defined In: (See JBTCQ and JBTSTS)

Used In: (See JBTCQ and JBTSTS)

Job queues keep track of a job's priority to run and to be in core and are maintained in table JBTCQ. Wait State Codes keep track of the runnability of a job and are defined in parallel so that a queue number and a wait state code that have the same meaning will also have the same numeric value. Values will differ at different sites due to conditional assembly.

Symbol	Value	Usage	Feature	Meaning
RNQ	00	N		Ready to run
WSQ	01	U		I/O Wait Satisfied
TSQ	02	U		TTY I/O Wait Satisfied
DSQ	03	U		Disk I/O Wait Satisfied
PSQ	04	U		Paging I/O Wait Satisfied
AUQ	05	R		Alter UFD Wait
DAQ	06	R		Disk Space Allocation Wait
CBQ	07	R		Disk core block scan wait
D1Q	11	R		DEctape Controller 1 Wait (DTA)
D2Q	12	R		DEctape Controller 2 Wait (DTB)
DCQ	13	R		DEctape/Magtape Control Wait
MMQ	14	R	FTMP	Memory Management Wait
CAQ	15	R	FTLOCK	Semi-permanent core Wait (LOCK)
EVQ	16	R	FTKI!FTKL	Exec virtual memory Wait
EQQ	17	R	FTEQDQ	Enqueue-Dequeue Wait
IOWQ	20	C		I/O Wait
TIOWQ	21	CQ		TTY I/O Wait
DIOWQ	22	C		Disk I/O Wait
PIOWQ	23	C		Paging I/O Wait
SLPQ	24	CQ		Sleeping (≥ 1 second)
EWQ	25	CQ		Event Wait (see JBTST2)
NAPQ	26	C		Napping (sleep < 1 second)
NULQ	27	CQ		Unassigned Jobs
JDCQ	30	CQ	FTDAEM	Job Waiting for DAEMON
STOPQ	31	CQ		Job Stopped (^C State)
PQ1	32	Q		Jobs Starting Up or coming out of TTY I/O wait
PQ2	33	Q		Non interactive jobs
CMQ	34	Q		Command Wait for Swapper (^W State)
HPQ1	35	Q	FTHPQ	High Priority (Real time) Jobs
HPQ2	36	Q	FTHPQ	Higher Priority (Real Time) Jobs

HPQ3 37 Q FTHPQ Highest Priority (Real Time) Jobs

The number of HPQ's is determined by the MONGEN dialogue.

Key to Usage

- N No queue header for this WSC. (JBTCQ+0 contains 0). Defined in QUEUES macro.
- U Wait State Codes for jobs that have become unblocked but need to be processed by QREQ in SCHED1 before being considered to run. Defined in QUEUES macro.
- R Wait State Codes for jobs that blocked at UUO level requiring a sharable resource that was unavailable (see REQTAB). Jobs will be unblocked by the Scheduler when the resource becomes available (see AVALTB). Defined in RWAITS macro.
- C Wait State Codes for jobs that are blocked waiting for a monitor event such as I/O complete, clock ticks a command, or DAEMON activity before they will be runnable. Defined in CODES macro.
- Q These Queues in JBTCQ actually hold jobs. Other symbols have queue headers reserved for them, but jobs are never placed in them. RNQ has no queue header at all.

NOTES:

1. RNQ, IOWQ, DIOUQ, PIOUQ, WSQ, TSQ, DSQ, and PSQ never actually hold jobs. The queues are defined only to define the corresponding Wait State Codes.
2. The values of PQ1, PQ23, CMQ, and STOPQ are never used as wait state codes. Jobs in any of the PQ's have wait state codes of 000. When jobs are put into CMQ or STOPQ they retain their previous codes, so that they can be returned to their previous queues.

Name: KON CONTROLLER DATA BLOCK

Description: There is one KON per disk controller. It contains information specific to that controller, e.g., dispatch addresses into the controller dependent routines. Names for KONS are in the form xxyINT where xx is the controller type and y the controller number (e.g., RPlINT for the second RH10 or RH20).

Defined In: COMMOD

Used by: DPXKON, DTASRN, FH XKON, FILIO, FSXKON, ONCE, ONCMOD, SYSCHK, SYSINI, VM SER

Format:

xxnINT	CONSO skip chain entry for this controller*
KONTAB	UDB Address Table - contains address of each UDB associated with this controller - 8 words
KONCOM	Negative of CCWMAX KONLST
KONLST	Channel command list for this controller Length is CCWMAX (KA)
	0,,0 (end of command list)
KONRED	Table of controller-dependent dispatch instructions
KONCUA	UDB address of unit doing transfer (or last one)
KONIOC	Address of controller-channel pair in low core
KONCHN	Address of channel data block for this controller
KONPTR	Indirect pointer to index KONTAB (P3)
KONDMP	Holds DEV DMP during dump-mode I/O
KONERR	Dispatch to controller-dependent error-recovery prog.
KONECC	Used for ECC mask and position

KONRRG	Used to hold drive registers
KONECR	Control register on error
KONEDB	Data buffer register on error
KONREG	Length of KONEBK
KONEBK	Place to save drive registers on error

* format of these instructions is:

```

xxnINT:      CONSO xxn,xx'INF      ;SKIP IF INTERRUPT FROM THIS KON
              JRST                  ;RESOLVED AT ONCE TIME
IFN FTDHIA,<
              CONSO xxn,7           ;IF NO PI UP,
              JRST  .-2             ;PROBABLY DIAG FIDDLING WITH KON
>;END DTDHIA
              JSR   ssn'SAV         :SAVE INTERRUPT AC'S
              JSP   J,xx'xINT       ;BRANCH INTO CONTROLLER-
                                   ;DEPENDENT CODE

```

Bit 0 of KONTAB is set when the controller is busy.

KONRED Controller-dependent dispatch addresses

KONRED	PI	Address of read program
KONRDS		Address of read, stop-on-error program
KONRDF		Read header and data
KONRDC		Read in 10/11 compatibility mode
KONWRT		Entry point to write program
KONWTS		Entry point to write, stop on error
KONWTF		Write format
KONWTC		Write in 10/11 compatibility mode
KONUNL		Unload drive
KONPOS	F	Position heads-entry zero for fixed-head disk
KONLTM		Compute rotational latency time
KONUPA	U	Test if controller is on-line
KONCPY		Determine type and capacity of unit
KONRCL		Recalibrate
KONSTP		Stop on hung unit

Word	Bit	Function
KONRED	0-2	PI level for controller
KONPOS	0	1 - implies fixed-head devices
KONUPA	0	1 - implies controller is down
	1	1 - controller can seek during transfer

The following notation is used in expressing some of the KON symbols.

X - Controller number (which is really a letter) - A,B,...
 XX - Controller type - DP, RP, FS, FH
 XXN - Controller symbol - e.g., DP0

KON (cont.)

KONTAB UDB Address Table within KON

Contains UDB Address for each unit on this controller.

One entry for each unit on the controller.

Indexed by physical unit number with controller.

	0	UDB 1	
	0	UDB 2	
	0	UDB 3	
	0	UDB 4	
	0	UDB 5	
	0	UDB 6	
	0	UDB 7	
	0	UDB 10	
0	13	18	35

Notes:

1. Bit 0 of first word set when controller is busy.
2. If controller has fewer than eight units, the additional entries are present, but blank.

KON (cont.)

KONTROLLER DEPENDENT INSTRUCTIONS

The four word block preceding a KONTROLLER DATA BLOCK contains the KONTROLLER dependent instructions used to link the controller to a PI CONSO skip chain.

X - KONTROLLER number (e.g., A,B)
 XX - KONTROLLER type (e.g., DP, FH)
 XXN - KNOTROLLER symbol (e.g., DP0, FH1)

```
XXN'INT:  CONSO XXN,XX'INF
          JRST
          JSR  XXN'SAV
          JSP  J,XX'XINT
```

NAME: LDB Line Data Block

DESCRIPTION: Contains data pertaining to one TTY Line.
One LDB per line; including scanner lines,
PTY's and CTY. LINTAB serves as directory.

DEFINED IN: SCNSER

USED BY: MSGSER, NETSER, COMCON, XTCSEr, PTYSEr, SYSINI,
COMMON, D76INT, CLOCK1, PSISER, UUOCON

FORMAT:

LDBDDB	See LDBDDB word	!Address of Line's attached ! DDB
LDICLR	Start clearing here on Restart	
LDBBKU	Copy of LDBTIP at last break RECINT	
LDBTIP	T2 to put characters in input buffer	
LDBTIT	T2 to take characters from input buffer	
LDBTIC	Count of echoed characters in input buffer	
LDBBKC	Count of break characters in input buffer	
LDBTOP	T3 to put characters in output buffer	
LDBTOT	T2 to take characters from output buffer	
LDBTOC	Count of characters from output buffer	
LDBECT	T2 to take characters from input for echoing	
LDBECC	Count of characters to echo	
LDBXNP	XON class character pointer for output	
LDBFLP	Filler character pointer for output	
LDBPBK	Up to 4 break characters for Packed Image Mode (PIM)	
LDBHPS	Horizontal Position Counter	
LDICLE(1)	Clear through here on Restart	
LDBDCH	See LDBDCH word	

LDBOFL	!	-----	!
LDBBYT	!	See LDBBYT word	!
LDBBY2	!	See LDBBY2 word	!
LDBPAG	!	See LDBPAG word	!
LDBISR	!	See LDBISR word	!
LDBQUE	!	See LDBQUE word	!
LDBTTW	!	See LDBTTW word	!
LDBBCD(2)	!	Flags for 2741 support	!
LDBREM(3)	!	See LDBREM+0 word	!
LDBREM(3)	!	See LDBREM+1 word	!
LDBREM(3)	!	See LDBREM+2 word	!
LDBREM(3)	!	See LDBREM+3 word	!
LDBREM(3)	!	See LDBREM+4 MCR word	!
LDBREM(3)	!	See LDBREM+4 VTM word	!
LDBTTD(4)	!	See LDBTTD word	!
LDBLAR(4)	!	Current line allocation remaining	!
LDBSS1(4)	!	First word of line speed message sent to front-end	!
LDBSS2(4)	!	Second word of line speed message	!
LDBMIC(5)	!	See LDBMIC word	!
LDBLOT(6)	!	Log taker !Count of characters to log	!
LDBLOC(6)	!	Count of characters to log	!
LDBLEN	!	Size of a data block for a line	!
	+	-----	+

- (1) If FTRSP (measure user TTY input response times) and FTACCT (include support for accounting) are set, the following three words would come directly after LDBHPS and be cleared on a restart:

```

LDBBCT  +-----+
        !   Total break character count since system (re)start  !
        !-----!
LDBICT  !   Total input character count since system (re)start  !
        !-----!
LDBOCT  !   Total output character count since system (re)start  !
        +-----+

```

- (2) If FTD10H (2741 lines on a DC10 -- Reassemble DLSINT) the flags are for 2741 support
- (3) If FTNET (include NCL network software) then five words containing network related LDB fields for remote station terminal status
- (4) If FTKL10 (include KL10 support) the definitions in this word are for support of RSX-20F terminals
- (5) If FTMIC (include Macro Command Processor) this word for MIC use
- (6) If FTMLOG (include MIC Log File Support) two words for MIC logging

LDBDDB

```

+-----+
!L!L!L!L!      !   L   !           !           !
!D!D!D!D!      !   D   !           !           !
!B!B!B!B!      !   P   !           !   Address of line's attached DDB   !
!C!C!C!D!      !   C   !           !           !
!M!M!M!E!      !   M   !           !           !
!R!F!K!T!      !   X   !           !           !
!-----!
!0!1!2!3!4-----8!9-----12!13-----17!18-----35!
+-----+

```

Bits in LDBDDB	Value	Bit Label or Byte Pointer	Description
0	4000000	LDBCMR	Command Request Bit
1	2000000	LDBCMF	Command forced
2	1000000	LDBCMK	Forcing KJOB Command
3	400000	LDBDET	Job detached from this line during command processing
4-8			
9-12		LDPCMXX	Pointer index for forced command
13-17			
18-35			

4	GTLLCP	Local copy (copy of LDLLCP)
2	GTLXON	XON is true
	LDPLCT (1)	Pointer to lower case bit
	LDPTAB (2)	Pointer to hardware tabs bit
	LDPFPM (3)	Pointer to hardware form feed bit
	LDPOSU (4)	Pointer to output suppression bit

LDBBYT

+-----+															
!L!	L!	!	P!		!	L	!	L!	L!	L	!	L!	L!	L!	!
!D!	D!	!	O!		!	D	!	l!	l!	D	!	l!	l!	l!	!
!B!	P!	!	H!	LDPSVC	!	P	!	R!	R!	P	!	R!	R!	R!	LDPECK
!O!	F!	!	P!		!	T	!	M!	C!	C	!	D!	D!	D!	!
!F!	L!	!	O!		!	I	!	I!	H!	P	!	E!	E!	E!	!
!L!	C!	!	S!		!	M	!	F!	P!	U	!	L!	C!	M!	!
+-----+															
!0!1 2!3—5!6-----14!15-----19!20!21!22-24!25!26!27!28-----35!															
+-----+															

Bits in LDBBYT	Value	Bit Label or Byte Pointer	Description
0		LDBOFL (1)	-1 if front end for this line is down
1-2		LDPFLC	Count of number of fillers by class
3-5		POHPOS	Old horizontal position. Needed for tab simulation
6-14		LDPSVC	Save character during free CTLF, Control "X", etc.
15-19		LDPTIM	Timeout on image input
20		LlRMIF	Mic interlock flag
21		LlRCHP	Change hardware parameters queue bit
22-24		LDPCPU	CPU number
25		LlRDEL (3)	Echo may echo 1 line if deferred
26		LlRDEC (3)	Echo may echo 1 character if deferred
27		LlRDEM (2)	Deferred echo bit. Set by Set Terminal Defer
28-35		LDPECK (4)	Echo check for half-duplex lines
(1)	400000	LlLOFL	The off-line bit
(2)		LDPDEM	Pointer to LlRDEM
(3)		LDPDEB	Pointer to deferred echo bits
(4)		LDPECK	If FTHDPX (support certain half-duplex terminals) or FTTRACK (leaves useful information in locations in COMMON to aid in debugging crash)

LDBBY2

```

+-----+
!L!L!L!L!L!L!L!L!L!      ! L! L!      !      !
!2!2!2!2!2!2!2!2!2!      ! 2! 2!      !      !
!L!L!L!L!L!L!L!L!L!      LDPDSC      ! R! R!      LDPWID      ! LDPRCS      !
!D!C!H!H!H!H!H!S!T!      ! X! E!      !      !
!E!C!D!D!D!D!D!N!A!      ! O! C!      !      !
!L!S!1!2!3!4!5!D!P!      ! N! S!      !      !
+-----+
!0!1!2!3!4!5!6!7!8!9-----17!18!19!20-----27!28-----35!
+-----+

```

Bits in LDBBY2	Value	Bit Label or Byte Pointer	Description
0	400000	L2LDEL	Last character in was a delete
1	200000	L2LCCS	Last character in was a Control "C"
2	100000	L2LHD1	XMT done flag seen this character on HDX line
3	40000	L2LHD2	RCV done flag seen this character on HDX line
4	20000	L2LHD3	Ignoring RCV interrupts due to echo check error on HDX line
5	10000	L2LHD4	Next RCV interrupt will be q'd after echo check
6	4000	L2LHD5	Receive echo was in fact not same as XMT character
7	2000	L2LSND	Send allowed while busy
8	1000	L2LTAP	Control "Q" from Keyboard turns on L2RXON
9-17		LDPDSC (1)	Dataset Control Table Index back pointer
18	400000	L2RXON	XON is true (paper tape input)
19	200000	L2RECS	Eat command sync
20-27		LDPWID	Width of terminal carriage
28-35		LDPRCS (2)	Most recent received character
(1)		LDPDSC	If modem control code (FTMODM)
(2)		LDPRCS	If FTTRACK (leaves useful information in locations in COMMON to aid in debugging cras

LDBPAG

!L!L!L!L!L!L!L!L!L! !	!	!	!	!
!P!P!P!P!P!P!P!P!P! !	!	!	!	!
!L!L!L!L!L!L!L!L!L! !	LDPACR	!	LDPPFF	!
!I!P!X!A!B!S!S!D!P! !	!	!	LDPPSZ	!
!R!A!O!L!L!L!A!I!O! !	!	!	!	!
!M!G!F!T!K!F!P!S!K! !	!	!	!	!
!-----!				
!0!1!2!3!4!5!6!7!8!9!10-----17!18-----23!24-----29!30-----35!				
!-----!				

Bits in LDBPAG	Value	Bit Label or Byte Pointer	Description
0	400000	LPLIRM	Terminal not heard from this second
1	200000	LPLPAG	Set terminal page command was executed
2	100000	LPLXOF	Sent XOFF, always send XON later
3	40000	LPLALT (1)	Altmode conversion bit
4	20000	LPLBLK	Suppress blank lines
5	10000	LPLSLF	Suppress line feeds
6	4000	LPLSAP	Send all pending
7	2000	LPLDIS (2)	Set if this is a display terminal
8	1000	LPLPOK	Forcing XMIT start via TOPOKE
9			
10-17		LDPACR	Auto CRLF counter
18-23		LDPPFF	Number LF's remaining on VT and FF simulation
24-29		LDPPSZ	Page size
30-35		LDPPCT	Page counter
(1)		LDPALT (1)	Pointer to altmode conversion bit
(2)		LDPDIS (2)	Pointer to display terminal bit

LDBISR

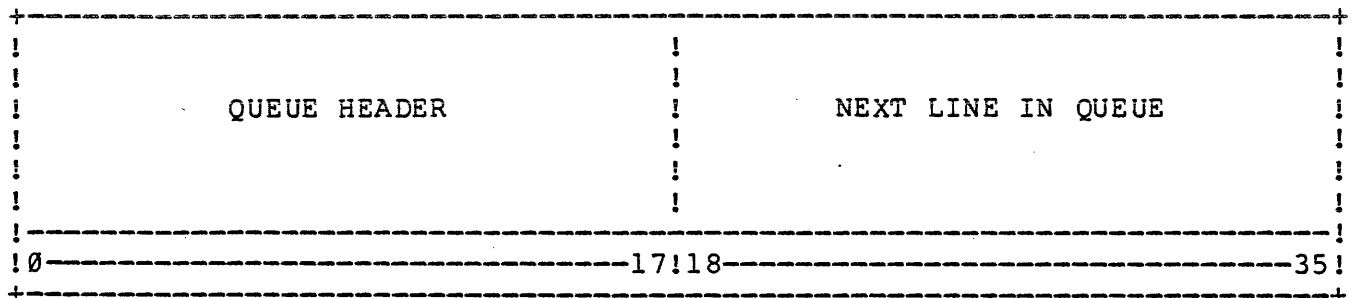
```

+-----+
!L!  L  !  L  !L! L! L! L!      !
!I!  D  !  D  !D! D! D! D!      !
!L!  P  !  P  !P! P! P! P!      !
!C!  T  !  R  !A! D! R! T!      !
!F!  S  !  S  !P! B! T! D!      !
!E!  P  !  P  !L! K! C! Y!      !
!-----!
!0!1--4!5--8!9!10!11!12!13--17!18-----35!
+-----+

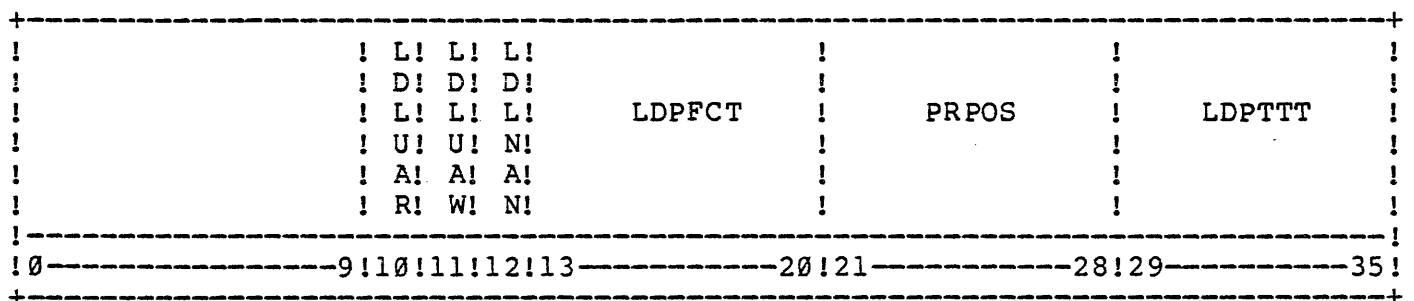
```

Bits in LDBISR	Value	Bit Label or Byte Pointer	Description
0		LILCFE	1 if front end is clever, 0 if dump
1-4		LDPTSP (1)	Transmit speed
5-8		LDPRSP (1)	Receive speed
9		LDPAPL	APL Mode
10		LDPDBK	Line has debreak
11		LDPRTC	Control "R", Control "T" compatibility
12		LDPTDY	User said "SET TERMINAL TIDY"
13-17			T1 for @LDBISR(U)
18-35			Address of ISR Dispatch Table
	4000000	LILCFE	Clever front end
	3600000	LILTSP	Transmit Speed
	0170000	LILRSP	Receive Speed
	0004000	LILAPL	APL Mode
	0002000	LILDBK	Line has debreak
	0001000	LILRTC	Control "R", Control "T" passed to program
	0000400	LILTDY	Do not expand output
(1)		LDPSPD	Both speeds (If FTCAFE, support the DC76)

LDB QUE



LDB TTW



Bits in LDBTTW	Value	Bit Label or Byte Pointer	Description
0-9			Free
10	200	LDLUAR	On if processing Control "R"
11	100	LDLUAW	On if processing Control "W"
12	40	LDLNAN	On if alpha-numeric seen while processing Control "W"
13-20		LDPFCT	Filler repeat counter
21-28		PRPOS	Position off prompt
29-35		LDPTTT	Terminal type as specified by TTY type commar LDBREM+0

- LDBREM+1

! ! ! ! ! ! ! ! ! !

LAST CHARACTERISTICS MESSAGE SENT

! ! ! ! ! ! ! ! ! !

! 0 —————— 35

LDBREM+2

LDPSLA	LDPDLA	LDPRLN
12!13	25!26	35!

Bits in LDBREM	Value	Bit Label or Byte Pointer	Description
0-12		LDPSLA	Our Source Link Address
13-25		LDPDLA	Our Destination Link Address
26-35		LDPRLN	Line number at Remote Station

LDBREM+3

!	L	!	!	!	!	!
!	D	!	!	!	!	!
!	P	!	LDPDRQ	!	LDPEPM	!
!	E	!	!	!	!	LDPRNN
!	L	!	!	!	!	LDPRNF
!	E	!	!	!	!	!
!-----3!4-----11!12-----19!20-----35!						

Bits in LDBREM	Value	Bit Label or Byte Pointer	Description
0-3		LDPELE	2741 element number
4-11		LDPDRQ	Number of data requests from remote
12-19		LDPEPM	Serial number of last EPM from remote
20-35		LDPRNN (U)	Number of node owning this TTY
20-35		LDPRNF (F)	Same as LDPRNN except indexed by "F"

LDBREM+4
(MCR)

!	!	!	!	!
!	!	!	!	!
!	LDPCHR	!	LDPADT	!
!	!	!	!	!
!	!	!	!	!
!	!	!	!	!
!-----8!9-----17!18-----26!27-----35!				

Bits in LDBREM	Value	Bit Label or Byte Pointer	Description
0-8		LDPCHR	If LRLTTO=1, contains the next output character
9-17		LDPADT	Auto-disconnect timer
18-26		LDPJOB	Pointer to job (only for connects)
27-35			Unused

LDBREM+4
(VTM)

[illegible]

Bits in LDBREM	Value	Bit Label or Byte Pointer	Description
0-17		LDPDST	Delayed-status-message for VTM
18-35		LDBVTQ	VTM-queue-link

LDBTTD

```
-----!
!                                     !
!                                     !
! Line information for support of RSX-20F terminals      !
!                                     !
!                                     !
!                                     !
! -----!
! Ø -----35!
```

Bits in LDBTTD	Value	Bit Label or Byte Pointer	Description
	740000		Remembered transmit speed
	36000		Remembered receive speed
	1000	LTLXOF	Sent XOFF to -20F
	400	LTLRBS	Remote bit sent for -20F datasets
	200	LTLCTO	Need to send flush output to -20F

LDBMIC

```

+-----+
| ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! |
| ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! |
| ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! |
| ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! |
| ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! |
| ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! |
+-----+
! 0!1!2!3!4!5!6!7!8!9!10!11!12!13!14!15-----21!22-----28!29-----35!
+-----+

```

Bits in LDBMIC	Value	Bit Label or Byte Pointer	Description
0	4000000	LDLCHK	Set if any bit 1-14 set
1	2000000	LDL.CC	Set if Control "C" has been typed
2	1000000	LDL.OP	Set if operator character seen in column 1
3	400000	LDL.ER	Set if error character seen in column 1
4	200000	LDL.CP	Set if Control "P" has been typed
5	100000	LDL.CB	Set if Control "B" has been typed
6	40000	LDL.XX	Silence this line
7	20000	LDL.MM	Line in Monitor Mode
8	10000	LDL.TI	Line in User Mode and TI wait or in Monitor Mode and can accept commands
9	4000	LDL.TO	Line is in column 1 on output
9	400	LDLCL1	Column 1 bit
10	200	LDL.CA	Set if Control "A" has been typed (abort)
11	100	LDL.RS	Error response
12	40	LDL.SY	Response code sync
13	20	LDL.LG	MIC is logging
14			
15-21			ASCII character to be treated as operator character
22-28			ASCII character to be treated as error character
29-35			MIC Master Job Numbers -- enables more than one MIC to run

Name: LINTAB Line Table

Description: Contains data pertaining to each TTY Line

One entry per line (including scanner, CTY, and PTY lines.)

Indexed by line number

Defined: COMDEV

Used in: COMMON, CLOCK1, COMCON, CPNSER, D76INT, DL0INT
DL1INT, DTESER, ERRCON, MSGSER, NETMCR, PTYSER
SCNSER, TTDINT, UUOCON, SYSINI, ONCE

Form:

Line 0	Characteristics	0	LDB Address
Line 1	Bits*		
Line 2			
-			
-			
-			
Line N			
	0	12 13 - 17 18	35

*Details on following page

Note:

The device characteristics bits in 0-12 are copied into various bytes of the LDBDCH word of the corresponding LDB during initialization.

LINTAB is located in COMDEV; bit definitions are in SCNSER and COMDEV

LINTAB (Cont'd.)

LINTAB Byte Definitions

[illegible]

Bits

0-1 Copied into bits 1-2 of LDBBYT

2-7 Copied into bits 12-17 of LDBDCH

8-12 Copied into bits 20-24 of LDBDCH

NOTE: These bits are used to initialize the specified areas of Line Data Block. The bits may be changed in the Line Data Block, but not in LINTAB.

See LDBBYT and LDBDCH for bit definitions.

Name: LOGTAB LOGIN JBTTAB POINTER TABLE

Description: Table of pointers to those job tables in which the LOGIN UUO stores statistics.

Defined In: UUOCON (Local Symbol)

Used By: UUOCON

Format:

+-----+-----+-----+			
AC J		JBTPPN##	; PPN
+-----+-----+-----+			
AC J		JBTPRV##	; Privilege Bits
+-----+-----+-----+			
AC W		PDNM1##	; 1st half user name
+-----+-----+-----+			
AC W		PDNM2##	; 2nd half user name
+-----+-----+-----+			
AC W		PDCNO##	; Charge no. this job
+-----+-----+-----+			
0	17	18	35

Special Notes:

1. LOGIN UUO used only by LOGIN or LOGOUT CUSPS
2. User program does:
 LOGIN AC,
 or
 CALLI AC,15
 where AC contains:
 XWD No.-of-entries,location
 which is a list of job
 statistics to be stored in the
 monitor tables listed in LOGTAB.

Name: LVDTBL LEVEL D DISK PARAMETER TABLE

Description: Contains parameters for the Level D disk routines. Each entry accessed by its own label.

Defined In: COMMOD

Used By: UUOCON, FILFND, FILUO, REFSTR, CLOCK1,
COMCON, SCNSER, ONCMOD, FILIO, SYSINI,
COMMON.

GETTAB Table No.: 16 (.GTLVD)

Format:

LVDTBL:

MFDPPN	(PROJ, PROG) FOR MFD'S [1,1]	
SYSPPN	(PROJ, PROG) for device SYS [1,4]	
FSFPPN	(PROJ, PROG) for FAILSAFE [1,2]	
HELPPN	(PROJ, PROG) for HELP and SYSTAT [2,5]	
PNTPPN	(PROJ, PROG) for printer spooling [3,3]	
SYSPPB	First PPB in System	PPB to start scan for grabable NMB *2
SYSSTR	First STR in System	Index in structure data block for pointer to next structure data block
SYSUNI	First UDB in System	Index in unit data block for pointer to next unit data block
SWPUNI	First swapping UDB	Index in unit data block for pointer to next swapping unit data block
CORNUM	Number of 4 word blocks	
STNPRT	Standard Privilege	
UFDPRT	Standard UFD Privilege	
MBFNUM	Number of monitor buffers	

QUESTR	SIXBIT Name of STR for Queuing Programs
CRUPPN	UFD for dumping crashes
SFDLVL	Number nested SFD levels allowed
SPLPRT	Protection for Spooled Output
SYSPRT	Protection for most System Files
SYSPRY	Protection for System .SYS Files
MUSTMX	Negative Maximum Extended Rib Argument for USETI
MAXTRN	Maximum number of blocks to transfer in one operation
XSYPN	PPN of experimental System [1,5]
OLDPN	PPN of old System [1,3]
UMDPN	User mode diagnostics PPN [6,10]
NUMBF	Default number of disk buffers
MAXSWP	Maximum number of units in active swapping list
ALGPPN	ALGOL library PPN [5,4]
BLIPN	BLISS library PPN [5,5]
FORPPN	FORTTRAN library PPN [5,6]
MACPPN	MACRO library PPN [5,7]
UNVPPN	UNIVERSAL library PPN [5,17]
PUBPPN	User maintained SYS: [1,6]
TEDPPN	TEXT editor library PPN [5,10]
RELPPN	REL file library PPN [5,11]
RNOPN	RUNOFF library PPN [5,11]
SNOPN	SNOBOL library PPN [5,13]

DOCPPN	DOC file library PPN [5,14]
FAIPPN	FAIL library PPN [5,15]
MUSPPN	Music library PPN [5,16]
DECPPN	Library for DEC distributed software PPN [10,7]
TABSWP	Pointer to active swap list
BASPPN	BASIC library PPN [5,1]
COBPPN	COBOL library PPN [5,2]
MXIPPN	PDP-11 library PPN [5,3]
NELPPN	NELIAC library PPN [5,20]
DMPPPN	DUMP library PPN [5,21]
POPPPN	POP2 library PPN [5,22]
TSTPPN	Test library PPN [5,23]
ALLOVR	If non-zero, log soft errors if recover from overrun on 1 try
MASERR	DAEMON's pointers to massbuss error locations
BATCHN	DAEMON's pointers to BAT block and channel error info
DBSPPN	DBMS library PPN [5,24]
EXPCHN	Offset of the expected channel terminal word in CHN
MICPPN	MIC library PPN [5,25]
TPSPPN	Text processing system library PPN [5,26]
CTLPPN	CTL file library PPN [5,27]
GAMPPN	Game library PPN [5,30]
ACTPPN	System accounting library PPN [1,7]
APLPPN	APL library PPN [5,31]

RIBECT	RIB error threshold
RIBTOT	Total RIB errors
SYSDOR	Dormant access table pointer
SYSCOR	Free core pointer
INTFNC	Number of times front end (RSX20F) had disk
D60PPN	DAS60 log file area [5,32]
ERTLOC	Starting location of the queue table for DAEMON error reports
ERTPT1	Starting pointer for DAEMON error extract
ERTPT2	Starting pointer for DAEMON error insert
ERTLTH	Length of DAEMON error table
ERTCDA	Offset of UNICDA in UDB
ERTDES	Offset of UNIDES in UDB
SYSPTR	Pointer to in-core copies of retrieval ptrs
MAXSSL	Max. # in system search list MAX # in job search list
ERTSLB	Offset of UINSLB into UDBs
UTPPPN	UETP Area

Special Notes:

1. MBFNUM may be changed at Once Only time.
2. If RH=0 core grabber should start over at beginning of PPB list (i.e., LH of SYSPPB).

Name: Magtape Kontroller Data Block

Description: Kontroller dependent information. One KDB for each tape controller on a system. Tags for magtape KDB's are MT'x'KDB, where 'x' is the kontroller number. Tabs for the CONSO instructions are MT'x'INT.

Defined in: COMDEV

Used by: TAPUUO, TM1KON, TMXKON

See also: Magtape Device Data Block, Magtape Unit Data Block

Format:

TKBCSO	CONSO MT'x'S,Ø
	JRST .-1
	JSR MT'x'SAV
	JSP W,TAPINT
TKBNAM	SIXBIT Controller Name
TKBCNT	Number of records done on this operation
TKBKDB	Link to next magtape KDB, Ø if none
TKBIUN	Initial Unit AOBJN Pointer
TKBCUN	Current Unit AOBJN Pointer
TKBDSP	Kontroller dispatch
TKBSTS	Kontroller status Consecutive ops remaining
TKBICP	Pointer to ICPC/CHL info
TKBTIM	Timer for spacing operations
TKBCDB	Pointer to channel (Ø if TM11A)

TKBJOB	Job # of maintenance mode owner
TKBCCL	Channel command list (5 words long)
TKBERB	IORB for error recovery (4 words long)
TKBFCT	Fairness count for Queued I/O
TT2C01	CONO MTxS, (T1) (TM02 only)
TT2CI2	CONI MTxS, T2 (TM02 only)
TT2C03	CONO MTxS, (T3) (TM02 only)
TT2DI2	DATAI MTxS, T2 (TM02 only)
TT2D02	DATAO MTxS, T2 (TM02 only)
TKBUDB	Pointers to units on this Kontroller (8 words long, found by TKBIUN)

MTAKDB (contd.)

Dispatch Offsets for TKBDSP

Value	Symbol	Meaning
0	TPKINI	Initialization code
1	TPKRES	Reset active transfer
2	TPKSIO	Start I/O
3	TPKINT	Interrupt routine
4	TPKCMD	Set device command in list (DX10 only)
5	TPKIDL	Set device idle
6	TPKONL	Skip if kontroller online
7	TPKSCH	Cause schedule cycle

TKBSTS Bits (These same bits are also defined for TUBSTS in the UDB)

Bit	Symbol	Meaning
12	TKSMNT	Controller is in maint mode
13	TKSSIL	Request silence about offline condition
14	TKSSCH	Requested scheduled interrupt
15	TKSSTD	Started
16	TKSSEL	Selected
17	TKSOFL	Offline

Name: Magtape Unit Data Block

Description: Tape unit dependent information. One UDB for each tape unit on each kontroller on a system. UDB's can be found by tracing them through the pointers in their KDB's (TKBUDB pointed to by TKBIUN) or by tags formed by concatenating "..U" with kontroller number and unit number, e.g., ..U12 for the third unit on the second kontroller (MTB2).

Defined in: COMDEV

Used by: COMMON, TAPUUO

See also: Magtape Device Data Block, Magtape Kontroller Data Block

Format:

TUBNAM	SIXBIT Unit Name		
TUBKDB		Pointer to kontroller	
TUBADR		Unit addr on kontroller	
TUBAKA	Current unit address	Current kontroller	
TUBDDB	Pointer to DDB's		
TUBCUR	Pointer to current DDB		
TUBSTS*	Unit status	Configuration info	TUBCNF
TUBQUE	Queue pointer for IORB's		
TUBERR	Error recovery info		
TUBRID	Reelid		
TUBFIL	# of files from BOT		
TUBREC	# of records from EOF		
TUBCRD	# of characters read since unload		

TUBCWR	# of characters written since unload	
TUBSRE	# of soft read errors	
TUBHRE	# of hard read errors	
TUBSWE	# of soft write errors	
TUBHWE	# of hard write errors	
TUBTME	Total media errors since unload	
TUBTDE	Total device errors since reload	
TUBTUN	Total unloads	
TUBTRY	Retries to resolve error	
TUBCCR	Character count on last record	
TUBPBE	Position before error file record	
TUBFES	Final error state word	
TUBCHR	Statistics for MTCHR. UUU	
TUBDDA	Shadow area fro DAEMON 12 (octal) words	TUBDDE
TUBPGM	Program name on error	
TUBUID	PPN using drive on error	
TUBCNI	Error status for next record	FTRDBA
TUBMSG	Label message length Label message function code	FTTLAB
TUBPHY	Physical name for label PCS	FTTLAB
TUBLBL	Label status word	FTTLAB
TUBRFM	Record format (for labels only)	FTTLAB

TUBRCC	Record size (for labels only)	FTTLAB
TUBBKL	Block size (for labels only)	FTTLAB
TUBEXP	Expiration date (for labels only)	FTTLAB
TUBPRT	Protection (for labels only)	FTTLAB

TUBSTS Bits

Bit	Symbol	Meaning
0	TUSNS	Do not schedule this unit (sign bit)
1	TUSBOT	Beginning of tape
2	TUSWTL	Write locked
3	TUSREW	Tape rewinding
12	TKSMNT	Controller is in maint mode (also in TKBSTS)
13	TKSSIL	Req. silence about offline cond. (also in TKBSTS)
14	TKSSCH	Requested scheduled interrupt (also in TKBSTS)
15	TKSSTD	Started (also in TKBSTS)
16	TKSSEL	Selected (also in TKBSTS)
17	TKSOFL	Offline (also in TKBSTS)

TUBCNF Bits

Bit	Symbol	Meaning
18	TUC7TK	Seven track
19	TUCIRD	Interrupts when rewind done
20	TUCDMS	Diagnostic mode set
21	TUCSNS	Force sense
23	TUCD62	Drive can do 6250 BPI
24	TUCD16	Drive can do 1600 BPI
25	TUCD80	Drive can do 800 BPI
26	TUCD55	Drive can do 556 BPI
27	TUCD20	Drive can do 200 BPI

Name: MEMTAB VIRTUAL MEMORY PAGE TABLE

Description: Contains one word per page of core. It is used during swap/page requests in conjunction with the UPMP to keep track of where pages end up on the swapping area, and which page to transmit next.

Defined In: S.MAC

Used By: VM SER

Format:

0	5	17	21	35
		nxt vir pg		disk address
				(unit # - Pg #)

Notes:

- Bit 0 is on for the last page in a fragment
- Bits 5-17 - next virtual page in this fragment
- Bits 21-23 - unit number in active swapping list
- Bits 24-35 - Page number on unit.

Name: MFD MASTER FILE DIRECTORY

Description: Disk file which tells location of each UFD, User File Directory, in a file structure. There is one MFD included in each file structure

There is one entry in the MFD for each [Proj, Prog] which has files in the structure.

Position of an entry is of no significance.

Defined In: COMMOD

Used By: FILFND, FILIO, FILUUD

Format: Format of MFD Data Block

Entry for User A
Entry for User B
.
.
.
Entry for User Z

Format of each entry

Project	Programmer
SIXBIT/UFD/	CFP

Special Notes:

1. The MFD is actually the UFD for [1,1]. Project-Programmer Number [1,1] is used only for this purpose.
2. The CFP, Compressed File Pointer, specifies the relative supercluster within the STR where the RIB of the corresponding file can be found.
3. The first entry in the MFD is a pointer to itself (i.e., to [1,1].UFD).

CFP=	Relative Unit # within STR	x	Number of Superclusters + per Unit	Relative Block # within that unit	.	Number Blocks per Supercluster
------	-------------------------------------	---	---	--	---	---

The number of clusters in a "supercluster" is determined by

number-clusters-in-structure

2(18)

rounded up by 1. Hence, every STR will have fewer than 2^{18} superclusters, and the CFP will fit in 18 bits. The concept of a supercluster is used only in connection with Compressed File Pointers.

Name: NMB FILE NAME BLOCK

Description: Used to remember file name in a project programmer number across all file structures
One NMB for each open file of each PPN regardless of how many versions of the files are in existence. Linked into list for each PPN.

Defined In: COMMOD

Used By: FILFND, FILUUD, FILIO

Format:

NMBNAM	File Name in SIXBIT		
NMBPPB	Next NMB	X X	
NMBRNG	Ptr to SFD NMB 1st	Compressed File Pointer	NMBCFP
NMBACC	First ACC Block	File EXT in SIXBIT	NMBEXT
NMBFSN	FSN		
NMBKNO	Know bits for this file		
NMBYES	Yes bits for this file		
NMBCNT	Use Countdd		

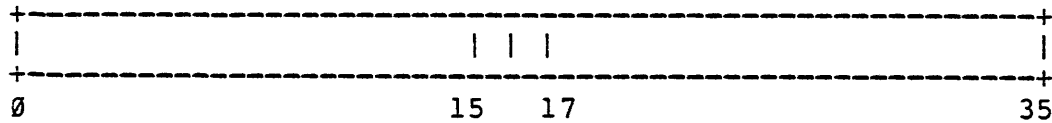
Notes:

1. If a file is being superseded, there are Access Blocks corresponding to each existing version of the file, but only one NMB.
2. The compressed file pointer in NMBEXT is copied from the second word of the UFD entry for this file. Its value is the logical super-cluster number within the STR of the file's RIB (See UFD).
3. If this NMB is for an SFD (NMPSFD (bit 34) set in NMBSFD), NMBEXT is a pointer to a NMB list of files under the SFD and the extension is assumed to be 'SFD'.
4. NMBKNO - Bit 36-N set if we know whether or not file exists in STR n (i.e., can be either way, and NMBYES tells which is true).
5. NMBYES - Bits 36-N set if file definitely exists in

STR n.

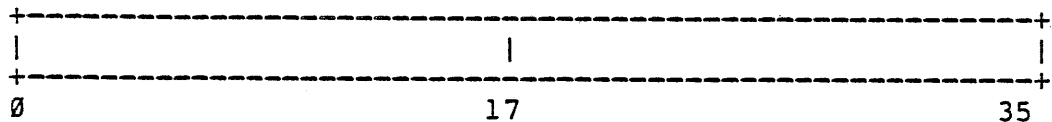
6. Access tables are linked into a ring for each file, starting and ending at NMBACC. All addresses which point to ACC blocks are even multiples of 4; the link back to NMBACC is not. If there are no ACC blocks in the ring at some time, the left half of NMBACC does not contain zero, but rather its own address.
7. NMBFSN holds the file structure number is bits 0-5.

NMBPPB Byte Definitions.



Word Label	Bits	Bit Label	Content
NMBPPB	0-15		Address of next NMB for this directory (low order 2 bits=0)
	16	NMPUPT	If this bit is on, NMBPPB is the location of the father SFD for this list of NMB's (only set in the last NMB in the list)
	17	NMPSFU	Bit on in AC, never in core. if the location returned (by UFORSF) is an SFD ACC. Off if the location is a UFB.

NMBACC Byte Definitions



Word Label	Bits	Bit Label	Contents
NMBACC	0-17		First ACC in access ring with this filename/PPN. If ring is empty, the bit is not 0, instead it points to itself. In this way no special checking is needed to add or delete access blocks from ring.

NMBEXT 18-35 Holds the left justified SIXBIT

'file extension

NAME: NUMTAB TABLE OF GETTAB TABLES

Description: Contains monitor table address and bits which indicate whether there may be segment data or process data.

This is the table which is referenced by the GETTAB UUO to pass back the information requested.

There are 163(octal) GETTAB tables in 7.01 monitors.

Defined In: UUOCON

Used By: UUOCON, COMMON

GETTAB Table No.: 23(octal)

Format:

Bits*	GETTAB Table Address
	JBTSTS
	JBTADR
	JBTTPN
"	"

Special Notes: ABSTAB (loc 410) in COMMON contains absolute address of NUMTAB

Bits 0-8 hold the maximum size of the table if it is a regular table

Bits 9-11 are interpreted as follows:

- 0 - undefined in this monitor
- 1 - index by item type
- 2 - index by job number
- 3 - index by job or segment
- 4 - index by job data in PDB
- 5 - index by negative and positive offsets

Name: PAGTAB PAGE TABLE

Description: This table is used to keep track of user core. It contains one word for each page of physical core. PAGPTR, defined in COMMON, contains the starting address for the linked list of free pages. In addition, each segment in core has its own linked list of pages and is addressed through the EPMP at location 412.

Defined In: S.MAC

Used By: KISER, KILOCK, KLSER, VMSE, SYSINI

Format:

Phys PG	0	1	2	3	4	5	6	7	17	18	35
	+-----+-----+										
0	Pg Bits								Ptr to next Phys PG of List		
1	-----										
2	-----										

	:										
	:										
	+-----+-----+										

Notes:

BIT	LABEL	MEANING
0	FREPAG	On if page is not in some job's addressing space.
1	LOKPAB	On if this page is contained in a segment which is locked in physically contiguous memory
2	LOKEVB	On if this page is contained in segment which is locked virtually contiguous in the exec addressing space
3	LOKIPB	On if this page is contained in a segment which is locked in place
4	NXMBIT	On if this page is below MEMSIZ but is non-existent.

- 5 MONTRB On if this page is contained in the
 Monitor
- 6 IPCBIT On if this page is owned by IPCF
- 7 TNCSHB On if this page is temporarily uncached

Name: PDB PROCESS DATA BLOCK

Description: One PDB for each active job. Setup by CREPDB routine in DATMAN when the job is initialized.

Defined In: DATMAN (modify and find routines), COMMON (prototype)

Used By: KISER, UUOCON, COMCON, CORE1, SEGCON, VMSE, QUESER, IPCSER, COMMON, SCHED1, FILFND, COMMOD, CLOCK1, NETSER, KLSER

GETTAB Table No.: Varies depending upon the entry.

Format:

Conditional
Assembly

.PDIPT	* ICPT (MCU) QUANTUM RUN TIME	.PDQNT
.PDCNO	USER'S CHARGE NUMBER	FTCNO
.PDKCT	KILO-CORE TICKS FOR THE JOB	FTKCT
.PDNM1	FIRST HALF OF USER'S NAME IN SIXBIT	FTUNAME
.PDNM2	SECOND HALF OF USER'S NAME IN SIXBIT	FTUNAME
.PDRTM	JOB'S INCREMENTAL RUNTIME	FTTIME
.PDTTM	JOB'S TOTAL RUNTIME	FTTIME
.PDTT2	ADDITIONAL RUNTIME IN FRACTIONAL JIFFIES	FTTIME
.PDEBT	TOTAL EBOX TIME IN JIFFIES	FTKL10,FTTIME
.PDEB2	REMAINDER IN EBOX COUNTS	FTKL10,FTTIME
.PDMBT	TOTAL MBOX TIME IN JIFFIES	FTKL10,FTTIME
.PDMB2	REMAINDER IN MBOX COUNTS	FTKL10,FTTIME
.PDPGM	PROGRAM TO RUN ON CONTROL-C OR RUN	FTSET
.PDABS	ADDR. BREAK SETTINGS BREAK ADDRESS	

.PDCVL	CVPL CPPL	
.PDMVL	MVPL MPPL	
.PDDVL	POINTER TO TABLE OF DDB'S WITH LOG. NAMES	FTHSLN
.PDIPC	FIRST PACKET SEND/RECEIVE CTRS	FTIPCF
.PDIPA	IPCF STATISTICS	FTIPCF
.PDIPQ	FLAGS AND QUOTAS	FTIPCF
.PDIPI	PID OF THIS JOB'S SYS:INFO	FTIPCF
.PDIPN	Last entry in IPCF Q	FTIPCF
.PDEQJ	Ø POINTER TO JOB QUEUE	FTEQDQ
.PDEQQ	FLAGS ENQ QUOTA	FTEQDQ
.PDJSL	JOB SEARCH LIST	FTSTR
.PDNET	DDB WITH ACTIVE CON/DIS ADDR OF SAVED CONTEXT	.PDSCX
.PDEPA	ADDR OF EXEC IPCF PACK	FTIPCF
.PDDIA	LOCATION OF DIAG. DDB FOR JOB	FTDHIA
.PDSTR	STRUCTURE THE PROGRAM CAME FROM	
.PDNAM	NAME OF THE PROGRAM	
.PDDIR	DIRECTORY THE PROGRAM CAME FROM	
.PDSPD	PATH TO PROGRAM	
.PDDFL	WORD CONTAINING USER DEFINED DEFAULTS	FTSET
.PDCAP	MAXIMUM PRIVILEGES ALLOWED	FTPRV
.PDACS	ACCOUNT STRING EIGHT WORDS	FTACCT

.PDVVC	VIRTUAL TIME-CORE INTERVAL	FTKCT, FTACCT
.PDUUC	COUNT OF UUOS DONE BY THIS JOB	FTACCT
.PDHZF	HPQ FIT FLAG	FTHPQ
.PDPST	NEGATIVE OF SWAPOUT TIME	FTPSCD
.PDOBI	OPERATOR/BATCH INFORMATION	

NOTES:

1. .PDCHK is used for debugging and stored in JBTCHK when the PDB is swapped out.
2. Bit 0 of .PDIPT is the PDMSWP bit which is set to indicate the expiration of the MCU (minimal core utilization)
3. .PDABS - Address Break Settings

1B0	OC.BCI	Break on instruction fetch
1B1	OC.BCD	Break on data fetch
1B2	OC.BCW	Break on write
1B3	OC.BCM	Break on MUUO reference to address (software)
1B4	OC.ABE	Address break enabled
1B5	OC.FEP	Follow exec paging
1B6	OC.FUP	Follow user paging
1B7	OC.BSU	Break address and conditions set by UUO.
4. .PDCVL - Bit 18 is set is the CPPL is a limit rather than a guideline.
5. .PDDVL - Pointer to table of pointers to DDB's owned by this job and having logical names, or zero if no such DDB's, or -1 if too many to fit in the table (4 words = 8DDB's).
6. .PDIPC - LH

RH 18-26	Packets sent and not received
RH 27-35	Packets waiting to be received
7. .PDIPA LH Count of sends since LOGIN
 RH Count of receives since LOGIN
8. .PDIPQ 1B0 IP.DBS Receiver is disabled
 1B1 IP.HBS Quotas have been set
 1B2 IP.DPR At least 1 PID dropped on RESET
 1B3 IP.DPL At least 1 PID dropped on LOGOUT
 18-26 Send quota
 27-35 Receive quota
9. .PDEQQ 1B0 EQ.HBS The Quota has been set

10. .PDJSL The number of words in the PDB is a function of the maximum number of file structures in a search list (.SLMXJ= D10). For each file structure there will be a 9 bit byte plus an additional two bytes for the fence and stop markers. For file structure bytes the following definitions exist.

1B0	Spare bit
1B1 FS.NCR	No-create
1B2 FS.WLK	Software writelock
B3-B8	File structure number

System search list is maintained in COMMOD starting at location SYSSL and also consists of 9 bit bytes.

The fence marker will have a value 1 greater than the maximum file structure number. The stop marker will have a value 1 greater than the fence marker.

11. Process Data Blocks are swappable but the current default setting of FTPDBS is 0, therefore PDB's remain resident.

12. .PDDFL contains the following fields:

Bits	Description
0-8	Default file protection
9	Non-zero is default protection was specified
10	Non-zero if file daemon specified protection
18-26	File protection from file daemon
27-35	Default number of disk buffers

13. Bits for .PDOBI are:

Bit	Description
0-1	Write to operator values
2-4	Operator values
10	Batch stream number set
11-17	Batch stream number

14. For the values for .PDCAP, see JBTPRV.

Name: PPB PROJECT PROGRAMMER NUMBER DATA BLOCK

Description: Contains information pertaining to all files belonging to one (PROJ,PROG)

One PPB for each (PROJ,PROG) with any active files.

Linked into list for system starting at SYSPPB and also available through JBTPPB.

Defined In: COMMOD

Used By: FILEND, FILUUD

Format:

PPBNAM	Project Number	Programmer number	
PPBSYS	Next PPB in system		
PPBUFB	First UFB		
PPBNMB	First NMB		PPBNLG
PPBCNT	Use Count for the PPB		
PPBKNO	KNO bits for UFD		
PPBYES	YES bits for UFD		
PPBLOK	Bits N+1=1 if UFD for FSN is interlocked		

Notes:

1. Bit 35 PPBNMB is the PPPNLG bit, [project, programmer] not logged in. This bit when set indicates the PPN is logged in. Used to flush PPB immediately when last file becomes dormant in PPB.
2. PPBKNO - Bit 36-N set if Monitor knows whether or not UFD for this PPN exists in STR N.
3. PPBYES - Bit 36-N is set if the UFD for this PPN definitely exists in STR N.

Name: PTYTAB

Description: Table of PTY DDB's. One entry for each PTY in the system.

Defined in: COMMON

Referenced in: SCNSER, SYSINI, PTYSER

See Also: TTYTAB

PTYTAB:	+-----+-----+			
		DDB ADDRESS	PTY0	
		" "		PTY1
		" "		PTY2
		DDB ADDRESS	PTYN	

Name: QBITS WAIT STATE CODE REQUEUE TABLE

Description: Specifies a requeue dispatch address and a transfer table address for requeueing 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 of the table.

Defined In: SCHED1

Used By: SCHED1

Wait State	Dispatch Address	Transfer Table Address
0	QRNT	QRNW
1	QWST	QWSW (-1)
2	QTST	QTSW
3	QDST	QDSW (-1)
4	QPST	QPSW (-1)
5	QAUT	QAUW (-1)
6	QDAT	QDAW (-1)
7	QCBT	QCBW (-1)
10	QD1T	QD1W (-1)
11	QD2T	QD2W (-1)
12	QDCT	QDCW (-1)
13	QCAT	QCAW (-1)
14	QMMT	QMMW
15	QEV T	QEVW

16	QEQT		QEQW
17	QIOWT		QIOWW (-1)
20	QTOWT		QTOWW
21	QDIOWT		QDIOWW (-1)
22	QPIOWT		QPIOWW (-1)
23	QSLPT		QSLPW
24	QEW T		QEW W
25	QNAPT		QNAPW (-1)
26	QNULT		QNULW
27	QJDCT		QJDCW
30	QSTOPT		QSTOPW

Note:

1. A transfer table address of (-1) implies that the code at the dispatch address will just modify the wait state code and no physical queue transfer will take place, i.e., a short term state change.

Name: Queue Transfer Table

Description: Contains input parameters for Queue Transfer routine. There is one such table for each different type of queue transfer.

Defined in: SCHED1

Format:

PLACE	FUNCTION
QUANT	DEST

Note:

PLACE>0 Transfer to beginning of destination queues.

PLACE<0 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 Job Size-Queues 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<0, no change to quantum runtime.

If QUANT>0, reset the quantum run time as follows:

1. If FUCNTION 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 SCHED1.

Name: Queue Table - Job Scanning (ISCAN, OSCAN, SSCAN, SSCAN1)

Description: Determines manner in which job queues are scanned by routine QSCAN in SCHED1.

One entry for each queue to be considered.

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

Defined in: COMMON

Referenced by: SCHED1

Format:

Queue	Scan Code
"	"
"	"
"	"
Ø	

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

QFOR	Scans whole queue forward, first in-core then out-core
QBAK	Scans whole queue backward, first out-core then in-core
IQFOR	Scans in-core queue forward
IQBAK	Scans in-core queue backward
IQFOR1	Scans in-core queue for first member
IQBAK1	Scans in-core queue backward (all but first member)
OQFOR	Scans out-core queue forward
OQBAK	Scans out-core queue backward
OQFOR1	Scans out-core queue for first member
OQBAK1	Scans out-core queue backward (all but first member)
SQFOR	Scans out-core subqueues (PQ2 class swap-in scan)
BGFOR	Scans out-core background batch subqueue (PQ2 class swap-in)
ISSFOR	Scans in-core subqueues (PQ2 class scheduling scan)
IBBFOR	Scans in-core background batch subqueue (PQ2 class scheduling)
OSSFOR	Scans out-core subqueues (PQ2 class lost-time scan)
IRRFOR	Scans just swapped in queue, then QP2 in-core queue
IGFOR	Scans just swapped in queue and jobs waiting for high segment

OLFOR Scans background batch, B.B. JIL, regular output
 queue, PQ2 in-core

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

A zero entry terminates the table.

See SSCAN, ISCAN, OSCAN, in COMMON.

Name: QQSTAB - Quantum Time Queue Table

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

Defined in: COMMON

Referenced in: SCHED1

FORMAT:

Quantum Time 1
Quantum Time 2
-
-
-
Ø

Quantum time is in jiffies (power line frequency).

A Ø entry indicates end of table.

Name: QTTAB - Queue Progression Queue Table

Description: 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).

Defined in: COMMON

Referenced in: SCHED1

Format:

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

Each entry is a queue number.

A Ø,,Ø entry indicates end of table.

Name: REQTAB SHARABLE DEVICE REQUEST TABLE

Description: Tells how many jobs require use of each sharable device. Each entry referenced by its own label.

An 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: SCHED1

Used By: CLOCK1, SYSINI

Format:

AUREQ	
DAREQ	
CBREQ	
D1REQ	
D2REQ	
DCREQ	
CAREQ	
MMREQ	
EVREQ	
EQREQ	

Special Notes:

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

AOSE	XXREQ
PUSHJ P,	XXWAIT
-	
-	
-	

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

2. Table AVALTB has entries corresponding to the entries in REQTAB.
3. The AVALTB entries are built by the conditionally assembled RWAITS MACRO entries in S.MAC; therefore all of the above listed entries will not be present in most systems.

Name: RIB RETRIEVAL INFORMATION BLOCK

Description: Disk block containing descriptive information about a file. There is one prime RIB for each file. If a file needs more retrieval pointers than can fit in a single RIB, a second (extended) RIB block is allocated to hold the additional pointers (and so on). The last block(s) of a file is (are) a copy of the prime RIB, called the redundant RIB.

Defined In: COMMOD

Used By: FILFND, FILUUD, FILIO, REFSTR, ONCMOD

Format:

RIBFIR	-Number of retrieval pointers First pointer address
RIBPPN	Project # Programmer #
RIBNAM	File Name in SIXBIT
RIBEXT*	File Extension Access Date
RIBPRV*	Access Mode Creation Time Creation Date
RIBSIZ	File length in words
RIBVER	Version number (as in .JBVER)
RIBSPL	Possible user file name when spooled
RIBEST	Estimated length of file in blocks
RIBALC	Number of blocks allocated for file (Including RIB's)
RIBPOS	Logical Block # in structure
RIBFT1	Word for future use by DEC
RIBNCA	Non-privileged word for customer to define
RIBMTA	Tape label if File on magtape

RIBDEV	Name of structure containing file	
RIBSTS*	Status Bits	
RIBELB	Logical Block # where bad region begins	
RIBEUN	Err Unit # in structure	Number bad blks in region
RIBQTF	FCFS Quota for this PPN in this structure (UFD only)	
RIBQTO	Logged Out Quota this PPN in this STR (UFD only)	
RIBQTR	Reserved Quota this PPN in this STR (UFD only)	
RIBUSD	Nr blocks used when job was logged out (UFD only)	
RIBAUT	Author - PP# writing the file	
RIBNXT	Next STR for this file	
RIBPRD	Prev STR for this file	
RIBPCA	Privileged Argument for customer use	
RIBUFD	Blk # in STR of UFD Data Block with ptr to this RIB	
RIBFLR	Relative block # of 1st block in RIB	
RIBXRA	Address of next RIB in chain	
RIBTIM	Creation date and time in new format	
RIBLAD	Last accounting date (UFD only)	
RIBDED	Directory expiration date (UFD only)	
RIBACT	AOBJN pointer for accounting string (UFD only)	
	Retrieval Pointers (details on following pages)	

RIBACS	Account string (pointer in RIBACT, UFD only)	
RIBCOD	0	777777
RIBSLF	0	Self Block number

RIBEXT Byte Definitions

Extension		Access Date	
0	17 18 23 24		35

Bits	Contents
0-17	Filename extension in SIXBIT
24-35	Last access date

RIB (cont.)

RIBPRV Byte Definitions

+-----+-----+-----+-----+-----+				
Access		Mode	Creation Time	Creation Date
+-----+-----+-----+-----+-----+				
0	8	12	23	35
0-8	Access Code			
9-12	Data mode of file			
13-23	File creation time			
24-35	Low-order twelve bits of file creation date			

NOTES:

1. Access Codes

- Bits 0-2 Apply to any job with matching programmer number.
- Bits 3-5 Apply to any job with matching project number.
- Bits 6-8 Apply to all other jobs.

Privilege Codes for User Files

Code	Highest Privileges
7	None (but owner may read)
6	Execute-only (but owner may read)
5	Read
4	Append (Allocate, Deallocate)
3	Update
2	Write (Supersede, Truncate)
1	Rename (Change attributes)
0	Change privileges

Privilege Codes for Directories

Code	Privilege
4	Allow lookups in this directory
2	Allow creates
1	Allow directory to be read as a data file
--	Any combination of these bits may be set.

2. This word is maintained in ACYPRV while the file is being accessed.
3. If the monitor is assembled for File DAEMON, and FILDAE is running, an owner privilege greater than or equal to 4 invokes FILDAE, giving extended access protection modes. See the FILDAE Specification for details.

RIB (cont.)

RIBSTS Byte Definition

```
+-----+
|               Status Bits               |
+-----+
```

LH bits apply to some file in UFD.

RH bits apply to this specific file.

Bit	Label	Meaning
0	RIPLOG	(LH only) User logged in
9,27	RIPSCE	File has had checksum error
10,28	RIPHWE	File has had hard write error
11,29	RIPHRE	File has had hard read error
14,32	RIPBFA	File found bad by FAILSAFE during restore
15,33	RIPCHR	File closed after crash
17,35	RIPBDA	File found bad by assessment cusp
18	RIPDIR	This is a directory
19	RIPNDL	No deletion of this file by any user
21	RIPNFS	Not to be dumped by FAILSAFE
22	RIPABC	Always bad checksum (SWAP.SYS, SAT.SYS)
23	RIPCBS	Compress bit set
7,24	RIPABU	Always BACKUP this UFD/file
31	RIPPAL	Pre-allocated file
25	RIPNQC	Non Quota-checked file

RIB (cont.)

Retrieval Pointers

```

+-----+
| Cluster Count |      Checksum      | Cluster Adr |
+-----+

```

Widths of these fields are defined symbolically, and may be different for each file structure. Byte pointer is defined in the HOM block, kept in STR Data Block while the structure is mounted.

Field	Byte Pointer
Cluster Count	STYCNP
Checksum	STYCKP
Cluster Address	STYCLP (23 bits maximum)

If cluster count = 0, the word actually is one of the following:

1. Pointer to new unit, if bit 18 = 1. Bits 19-35 specify logical number within file structure.
2. EOF flag, if whole word is zero.
3. Cluster Count is number of clusters in group.

RIBELB Byte Definitions

+-----+		
	Error Bits	LBN where Bad Region starts
+-----+		

Bits 0 through 8 give the type of error that occurred. Bits 9 through 35 give the logical block number on the unit where the error occurred.

Bit	Meaning
3	Error other than listed below
4	Data error (parity or ECC hard)
5	Search or header compare error

RIBXRA Byte Definitions (same format as DEVRIB)

Bits	Byte Pointer	Content
0		Set to 1
1-8	DEYRBC	Number of RIB (first extended RIB is 1, etc.)
9-12	DEYRBU	Logical unit on which extended RIB exists
13-35	DEYRPA	Cluster address on unit of extended RIB

Name: SAB STORAGE ALLOCATION BLOCK

Description: Table describing allocation of clusters of blocks for a file structure.

Each allocation bit represents a corresponding cluster of physical blocks within the STR.

Defined In: COMMOD

Used By: FILEND, FILIO, ONCMOD, REFSTR

Format:

SABRNG	Core adr of next SAB for unit				No free clusters in this SAT	SABTAI
SABSCN	No words in SAT Buffer				Adr to start scan for free clusters	
SABNDX*	A	B	NDX		First Cluster	SABCLA SABFIR
SABHOL	Number of blocks in largest hole					
SABBIT	One data block of SAT.SYS					

*Details on following page.

NOTE:

See SAT.SYS and SPT

SAB (cont.)

SABNDX Byte Defintions

A		B		NDX				First Cluster																		
0		1	2					12	13																35	

Word Label	Bits	Bit Label or Byte Pointer	Content
SABFIR	0	SAPDIF	Set if table in core different from disk
SABFIR	1	SAPBAD	Set if SAT Block is on a bad block in disk
SABNDX	2-12	SAYNDX this SAT	Index value for SPT entry representing
SABCLA	13-35	SAYCLA SAT	Cluster address within unit of first cluster represented in this

NOTE:

1. The NDX and CLA fields are related by the formula.

$$\text{CLA} = \frac{\text{Number of Clusters per SAT}}{\text{Number of SATs}} \times (\text{NDX})$$

2. Bits 2-35 are set -1 when the file structure is created to force SAT to be read into SAB.

SAB (cont.)

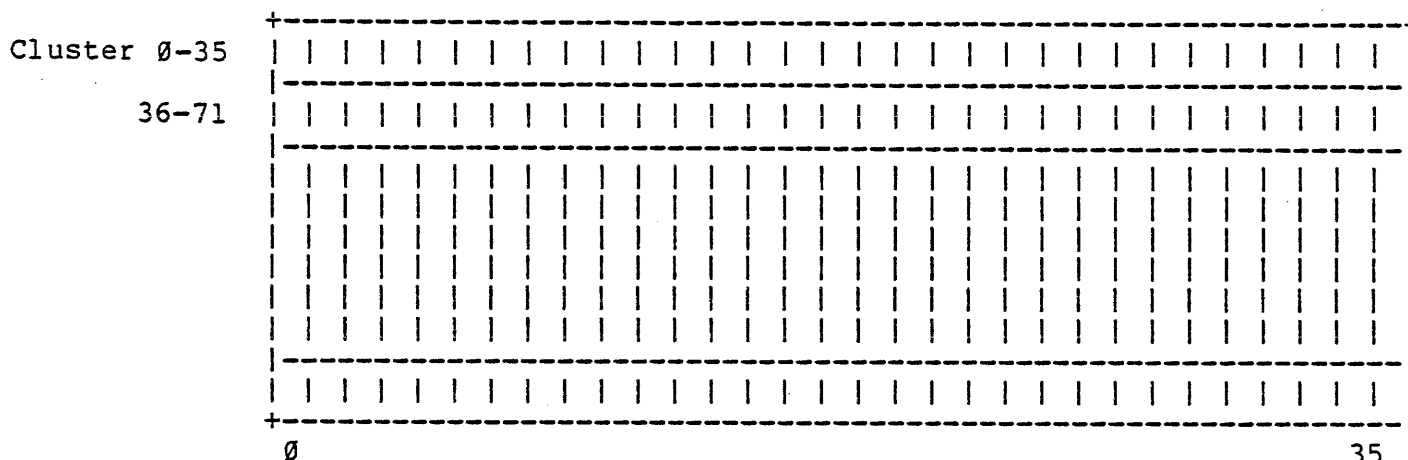
SAT.SYS

Disk file describing the allocation of all clusters of blocks on the file structure.

Contains one bit for each cluster of the file structure.

Bits are in the same order as the clusters which they represent.

Data Block Format



NOTES:

1. SAT blocks are always on the same unit as the clusters which they represent.
2. If more than one SAT block is needed for a single physical unit, each block will be near the clusters which it represents. Hence, only the first block in each group (1 cluster) of SAT.SYS contains data.
3. Programs should not look at bits corresponding to nonexistent clusters. They may or may not be set. (Function of hardware sector length)
4. Each SAT block has a corresponding entry in the Storage Allocation Pointer Table (SPT) for that unit.
5. The unused bits in the final word of each SAT must be set to 1, as the Monitor depends on this condition when searching for holes.

Name: Scanner (TTY) Device Data Block

Description: The SCANNER DDB contains information needed to perform I/O operations. There is one SCN DDB per job. It is set up and linked into the DDB chain by ONCE.

Defined in: S.MAC

	Left Half	!	Right Half
DEVNAM	Device Mnemonic	!	Device Number
DEVCHR	See DEVCHR		
DEVIOS	See DEVIOS		
DEVSER	Next Device Data Block	!	Device Service Dispatch Table
DEVMOD	See DEVMOD		
DEVLOG	Logical Name for Job Device		
DEVBUF	Address of 3-word output buffer header	!	Address of 3-word input buffer header
DEVIAD	R in Index Field (bits 14-17)	!	Address of input buffer the service routine is filling
DEVOAD	R in Index Field (bits 14-17)	!	Address of output buffer the service routine is emptying
DEVSTS	Word for Device CONI		
DEVSTA	See DEVSTA		
DEVXTR	Extra Word		
DEVEVM	Word to keep track of executive virtual memory this device has		
DEVPSI	Enable Cond.	!	Pending Ints.
DEVESE	See DEVESE		
DEVHCW	Hardware Characteristics Word		

DEVCPU	See DEVCPU
DEVJOB	Job and Unit Numbers
DEVLSD	**Length of Short DDB**
DEVFIL	File Name in SIXBIT
DEVEXT	Extension ! Unused
DEVPPN	Ø
	If KL1Ø and Master-Slave
DEVNBF	Number of buffers swept (QPR only)
DEVSBF	Saved DEVNBF
DEVCSN	Cache Sweep Number (QPR)

Notes:

1. The Line Data Block address is set when a specific TTY line is attached to the job. The DDB is always associated with the job, and the LDB is always associated with the line. The linkages may be changes by the ATTACH or DETACH commands.

DEVCHR
Device Characteristics Word

```

+-----+
!D!D!D!D!D!D! ! H !           !           !           !
!V!V!V!V!V!V! ! U !           !           !           !
!L!C!D!D!C!O! ! N !   PDVCNT !           !   BUFFER SIZE !
!P!M!A!I!N!F! ! G !           !           !           !
!T!D!T!B!E!L! ! S !           !           !           !
!L!A!J!P!T!N! ! T !           !           !           !
!-----!
!0!1!2!3!4!5!6!7-9!10-----16!17-----23!24-----35!
+-----+

```

Bits in DEVCHR	Value	Bit Label or Byte Pointer	Description
0	4000000	DVLPTL	Lower case LPT
1	2000000	DVCMDA	Device controlled by mountable device allocator
2	1000000	DVDATJ	Device allocated to job in DEVJOB
3	400000	DVDIBP	Device is a batch PTY
4	200000	DVCNET	Device controlled by NETSER
5	100000	DVOFLN	Device off-line last time service routine polled
6			Free
7-9	400	HUNGST	Code for hung device timeout
	0	PDVTIM	0 means device cannot be hung
	1-7		means hung time is 2**N-1 seconds
10-16		PDVCNT(1)	Countdown timer for the hung device
17-23			Free
24-35			Buffer Size

- (1) Number of seconds to go before considering device hung.
 Initialized every time the device is serviced, using the hung
 constant to determine the value.

DEVIOS
Device Input/Output Status Word

```

+-----+
!      ! I! I! I! I! I! I! I! I! I! I! I! I! I! I! I!      ! I! I! I! P !
!      ! O! O! /! O! O! O! O! O! O! O! O! O! O! O! O!      ! O! O! O! I !
!      ! S! E! O! S! F! B! W! I! D! D! B! D! A! B! T! P!      ! N! C! W! O !
!      ! D! N!   ! T! S! E!   ! M! E! T! K! E! C! O! E! A!      ! R! O! C! M !
!      ! M! D!   ! B! T! G!   ! P! R! E! T! N! T! T! N! R!      ! C! N!   ! O !
!      ! R!   !   ! L!   !   !   ! M! R! R! L! D!   !   ! D!   !      ! K!   !   ! D !
!-----!
!0-----10!11!12!13!14!15!16!17!18!19!20!21!22!23!24!25!26!27 28!29!30!31!32-35!
+-----+

```

Bit in DEVIOS	Value	Bit Label or Byte Pointer	Description
0-10			Device Dependent Bits
11	100	IOSDMR	TSKSER only, disable message reassembly
12	40	IOEND	Service routine has transmitted last data
13	20	I/O	Out = 1; In = 0
14	10	IOSTBL	Device error flag
15	4	IOFST	Next item will be the first item of a buffer
16	2	IOBEG	Virgin Device
17	1	IOW	Input/Output Wait
18	400000	IOIMPM	Improper Mode
19	200000	IODERR	Device Error
20	100000	IODTER	Data Error
21	40000	IOBKTL	Block Too Large
22	20000	IODEND	Data End Encountered
23	10000	IOACT	Device Active
24	4000	IOBOT	Beginning of Magnetic Tape
25	2000	IOTEND	End of Magnetic Tape
26	1000	IOPAR	Write even parity if 1 on magnetic tape
27-28			Density of Magnetic Tape
	00		Installation Standard
	01		200 BPI
	10		556 BPI
	11		800 BPI
29		IONRCK	Read with no reread check
30		IOCON	Continuous
31		IOWC	Don't compute word count
32-35		PIOMOD	Data Modes
	0	A	ASCII
	1	AL	ASCII Line
	2	PIMMOD	Packed Image Mode
	3	BYTMOD	Byte Mode
	10	I	Image
	13	IB	Image Binary

14	B	Binary
15	SD	Scope Dump Mode
16	DR	Dump by Records
17	D	Dump across records

DEVMOD
Device Modes Word

```

-----+
D!D!D!D!T!T!T!D!D!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! V! V! S! S!      !
D!D!C!L!Y!Y!Y!D!L!P! P! D! A! M! T! D! I! O! S! S!      !
I!S!D!P!A!U!B!I!N!T! T! T! V! T! T! I! N! U! C! P!      !
R!K!R!T!T!S!I!S!G!P! R! A! A! A! Y! R! ! T! O! R!      !
I! ! ! !C!E!U! ! ! ! ! ! L! ! ! ! ! ! N! G!      !
-----+
!1!2!3!4!5!6!7!8!9!10!11!12!13!14!15!16!17!18!19!20-----35!
-----+

```

is in MOD	Value	Bit Label or Byte Pointer	Description
	400000	DVDIRI	DECTape Directory in core if 1
	200000	DVDSK	Is a disk
	100000	DVCDR	Is a card reader
	40000	DVLPT	Is a line printer
	20000	TTYATC	TTY attached to job if 1
	10000	TTYUSE	TTYDDB in use flag
	4000	TTYBIU	TTYDDB in use
	2000	DVDIS	Is a display
	1000	DVLNG	Device has long dispatch table
	400	DVPTP	Is a paper tape punch
	200	DVPTR	Is a paper tape reader
	100	DVDTA	Is a DECTape
2	40	DVAVAL	1 if device available to this job
3	20	DVMTA	Is a magnetic tape (rewind)
4	10	DVTTY	Is a TTY
5	4	DVDIR	Has a directory
6	2	DVIN	Input Device
7	1	DVOUT	Output Device
18	400000	ASSCON	Device Assigned by Console Command ASSIGN
19	200000	ASSPRG	Device Assigned by Program (INIT UO)
10-35			If data mode n is legal, bit (35-n) is set

DEVSTA
Device Station Word

```

+-----+
!D!D!D!D!      ! D!  ! .! .! .! .! .! D! D! D! D! D! D! D! D! D! D!  !  !
!E!E!E!E!      ! E!  ! S! S! S! S! S! S! E! E! E! E! E! E! E! E! E!  !  !
!P!P!P!P! DEVTYP ! P!  ! P! P! P! P! P! P! P! P! P! P! P! P! P! V!  !  !
!S!A!L!R!      ! I!  ! C! C! P! P! L! A! I! O! E! M! E! D! D! D!  !  !
!P!D!E!A!      ! B!  ! D! D! T! L! P! I! N! N! C! S! V! E! E! P!  !  !
!L!V!N!S!      ! C!  ! R! P! P! T! T! O! D! D! S! G! M! R! L! R!  !  !
!-----!
!0!1!2!3!4-----9!10!11 ! ! ! ! !17!18!19!20!21!22!23!24!25!26--29!30--35!
+-----+

```

Bits in DEVSTA	Value	Bit Label or Byte Pointer	Description
0	4000000	DEPSPL	DDB is for a spooled device
1	2000000	DEPADY	1 if a disk in 10/11 compatibility mode
2	1000000	DEPLEN	1 if variable length buffers
3	400000	DEPRAS	1 if device has restricted assignment
4-9		DEVTYP	Class
10		DEPIBC	Inhibit Clearing output Buffers
11-17			Spool bits
	37	.SPALL	Change SPALL if add new spooling devices
13	20	.SPCDR	Card Reader Spool Bit
14	10	.SPCDP	Card Punch Spool Bit
15	4	.SPPTP	Paper Tape Punch Spool Bit
16	2	.SPPLT	Plotter Spool Bit
17	1	.SPLPT	Line Printer Spool Bit
18	4000000	DEPAIO	Doing Asynchronous Input/Output
19	2000000	DEPIND	Input Not Yet done
20	1000000	DEPOND	Output Not Yet done
21	400000	DEPECS	On if a non-superscede enter
22	400000	DEPMMSG	This device controlled by MSGSER
23	200000	DEPEVM	If 1, device doesn't need EVM
24	100000	DEPDER	Disable error recovery on this device
25	40000	DEPDEL	Disable error logging on this device
16-29	2000	DEVDPRI	Disk Priority
30-35			Station Number

DEVESE
Device Extended Software Error Word

[illegible]

Bit in DEVESE	Value	Bit Label or Byte Pointer	Description
0-17			PSISER Device Link
18-26			PSISER Vector Interrupt
27-35			Extended Software Error Status

DEV CPU
Device CPU Word

				D!	
				E!	
				P!	
				C!	
				P!	
				S!	

!0---	2!3---5!6---	8!9---	11!12!13----	17!18--	--35!

Bits in DEVCPU	Value	Bit Label or Byte Pointer	Description
0-2			CPU number for 1st CPU owning device
3-5			CPU number for 2nd CPU owning device
6-8			Queue-protocol
9-11			CPU on which in or out last done
12		DEPCPS	1 if bits 30-32 are valid (device on 2 CPU's)
13-17			0
18-35			Address of interlock word

Name: Scheduler Scan Tables

Defined in: COMMON

Referenced in: SCHED1

SSCAN Used by Master processor for selecting a job to run.

SSCAN	-HPQN	IQFOR
	-HPQ1	IQFOR
	-PQ1	IQFOR
	-PQ2	IRRFOR(RR)
	-PQ2	ISSFOR(class)
	-PQ2	IBBFOR(class)

SSCAN1 Used by second processor for selecting a job to run

SSCAN	-HPQN	IQFOR
	-HPQ1	IQFOR
	-PQ2	IRRFOR(RR)
	-PQ2	ISSFOR(class)
	-PQ1	IQFOR
	-PQ2	IBBFOR(class)

*SQSCAN Used by SQFOR code

	Subqueue #	Ptrs to Quota Left
SQSCAN	-SQ0	CLSQTA
	-SQ1	CLSQTA+1
	-SQ2	CLSQTA+2
	-SQN	CLSQTA+N

SQFOR - Scan subqueues forward according to SQSCAN table if RRFLAG = 0, (count of classes with non-zero quotas), else scan PQ2 via QFOR routine.

DCSCAN - Used by the scheduler for selecting jobs for ICPT decrementing. This table is not processed by QSCAN but by specific code in the IPCT maintenance routines.

DCSCAN	-----	
		-EWQ

		-SLPQ

		-PQ2

		-PQ1

		-HPQn

Name: SPT Storage Allocation Pointer Table

Description: Contains pointers to all SAT blocks for a unit, whether in core or not.

One entry for each SAT block on a unit, in order of the cluster address which they represent.

Zero entry indicates end of table.

Defined in: COMMOD

Referenced in: FILFND, ONCMOD

SPTFIR	Free Cluster Cnt		Cluster Adr in Unit	
	Free Cluster Cnt		Cluster Adr in Unit	
	0		0	
	0		12 13 35	
Bits	Byte Pointer		Content	
0-12	SPYTAL		Number free clusters represented in this SAT Block.	
13-35	SPYCLA		Cluster address within unit for this SAT Block.	

Notes:

1. Each Unit Data Block, UDB, contains a pointer to its SPT.
2. Each SAT block which is in core is in a SAB, Storage Allocation Block. The SAB contains the index value for the entry in this table corresponding to the SAT block which it currently contains.
3. The last word in the SPT table will always be zero.

Name: STR File Structure Data Block

Description: Contains descriptive information about a file structure.

There is a STR Data Block for each STR defined in the system.

Defined in: COMMOD

Referenced in: COMMON, FILIO, FILEND, FILUVO, IPCSER, ONCMOD

Format:

STRNAM	SIXBIT STR Name		
STRSYS	Next STR in system	This STR Number	STRFSN
STRUNI	First UDB for this STR	K for CRASH.SAV	STRK4C
STRREF	Nonzero if STR needs to be refreshed	NR units in this STR	STRUNM
STRHGH	Highest logical BLK # in this STR		
STRSIZ	Size of STR in 128 word blocks		
STRGAR	Limit on total blocks reserved in STR		
STRRES	Nr reserved blocks remaining free		
STRALT	Alter number for this structure		
STRTAL	No. first-come-first-serve free blocks on structre		
STROVR	Overdraw limit - per user		
STRMNT	Mount count for this STR		
STRPT1	First retrieval ptr for MFD		
STRTRY*	TRY /////////////////////////	X UN1 /////////	STRUN1
STRBPU	(Maximum) Nr Blocks per unit		
STRBSC	Nr blocks per supercluster	(maximum) Nr super-clusters per unit	

STRJOB	ACCESS	Job # having access or 0
STYCNP	Byte ptr to RIB cluster count in AC T2	
STYCKP	Byte ptr to RIB checksum field in AC T2	
STYCLP	Byte ptr for cluster adr in AC T2	
STRPPN	PPN of the structure owner	

*Details following.

NOTE:

1. All STR Data Blocks are set up by the Once Only Code, according to information found in the Home Blocks. No information pertaining to STR's is coded into the monitor.
2. STRALT is incremented each time a SAT block is written for this structure.
3. ACCESS is -1 if the job in RH is only job with the structure mounted, and it is not single access.

STR (continued)

STRTRY BYTE DEFINITIONS

+-----+-----+-----+-----+-----+-----+-----+-----+										
	TRY	RETRY	RECAL			X		UNIT		X
+-----+-----+-----+-----+-----+-----+-----+-----+										
0	6	12	18	27	31	34	35			

Word Label	Bits	Byte Label	Content
STRTRY	0-5	STYTRY	Number of times to retry before error considered hard.
STRTRY	6-11	STYSER	Number of times to retry on search and data errors
STRTRY	12-17	STYRCL	Number of recalibrates for search and data errors
STRlPT	27	STPlPT	Set if STRPTl is only retrieval pointer for MFD.
STRUNl	28-31	STYUNl	Logical unit number within this file structure where MFD begins.
STRPVS	35	STYPVS	Non-zero if this is a private structure.

NAME: SWPLST

Description: Table used by the VM swapper in conjunction with MEMTAB to keep track of jobs being swapped or having paging I/O in progress.

Defined In: COMMON

Used By: SCHED1, SWPSER, VM SER

Format:

0	17 18	26 27	35
Bits	Start Phys. Pg.	# of Pg.*	

Notes:

SL.FRG = 1B0 Fragmented Entry
 SL.DIO = 1B1 Direction of IO (1 = out)
 SL.SIO = 1B2 Swapping/paging (1 = swapping)
 SL.IOP = 1B3 IO in Progress
 SL.IOD = 1B4 IO Done (This swap list entry is done)
 SL.IPC = 1B5 On if an IPCF Page
 SL.CHK = 1B11 Swapping Checksum Error
 SL.ERR = 1B12 IO Error (IODTER, IODERR, or IOIMPM)
 SL.CHN = 1B13 Channel Error (IOCHMP or IOCHNX)

if not a fragmented entry

Bits 14-26 - starting physical page number
 (used as an index into MEMTAB)

Bits 27-35 - number of pages

* if fragmented

Bits 18-35 - address of fragment table.
 The fragment table is linked
 the same way the JBTSWP entry
 is, but the entries are as above.

Name: SW2LST

Description: The original SWPLST entry is stored here for cleanup purposes, since SWPLST is modified while I/O is progressing.

Defined In: COMMON

Used By: VM SER

Format:

SW2LST	+-----+

	. : . :

Name: SW3LST

Description: The right half of this table is used to store the job number of the job being swapped or doing paging I/O, and left half holds the contents of SWPOUT.

Defined In: COMMON

Used By: SCHED1, SWPSER, VM SER

Format:

SW3LST		
	.	
	.	
	(SWPOUT)	Job Number

Name: SWPTAB Swapping Table

Description: Contains addresses of the Unit Data Blocks of
all units available for swapping.

Defined in: COMMOD

Referenced in: FILEND, FILIO, CPNSER, SEGCON, SWPSER, VMSE,
SYSINI, ONCMOD

UDB 1
UDB 2
UDB 3
UDB n

NOTE:

This table specifies the Active Swapping List.

Name: TABSTR Structure Table

Description: Contains addresses of all File Structure Data Blocks in the system.

File structures are in order of access speed, fastest to slowest.

Index value for each entry is the File Structure Number, FSN.

Defined in: COMMOD

Referenced in: COMMON, FILIO, FILUOO, FILFND, CPNSER, SYSINI, ONCMOD

STRAOB:	+	-----	+
		-.SLMAX-1 . .FSMIN	
STRTAB:		-----	
		STR 1	

		STR 2	

		STR N	

	+	-----	+

NOTE:

Number of entries is STRNUM. This value cannot exceed 36.

The first word is an AOBJN pointer to the rest of the table.

Name: Transfer Tables

Description: These tables are used in requeing process to determine the destination queue and quantum runtime for jobs being requeued by the Scheduler.

Defined in: SCHED1

Format:

Although these tables are defined by Macros in COMMON they are represented here in their expanded form.

```
EQFIX== 400000,,QFIX ;SPECIFIES FIX TRANSFER TO END OF Q
EQLNKX==400000,,QLNKZ ;SPECIFIES REQUEING AND QUANTUM RUN
; TIME BASED ON CURRENT QUEUE AND
; JOB SIZE
```

```
QNULW:      EXP EQFIX ;NULL QUEUE
            XWD -1,-NULQ
```

```
QSTOP::
QSTOPW: EXP EQFIX           ;STOP QUEUE
        XWD -1,-STOPQ
```

```
IFN FTDAEM,
QJDCW: EXP EQFIX
        XWD -1,-JDCQ
; End of conditional assembly on FTDAEM
```

```
QCMW:: EXP EQFIX           ;COMMAND WAIT QUEUE
        XWD -1,-CMQ
```

```
QTSW:           ;TTY I/O WAIT SATISFIED
QRNW: EXP EQFIX   ;JOBS JUST BECAME RUNNABLE
        XWD QADTAB##,-PQ1
```

```
QRNW1: EXP EQFIX           ;BACK OF PQ1, NO QUANTUM CHANGE
        XWD -1,-PQ1
```

```
QRNW2: EXP EQFIX           ;BACK OF PQ2, NO QUANTUM CHANGE
        XWD -1,-PQ1
```

```
QTIOWW: EXP EQFIX           ;TTY I/O WAIT SATISFIED
        XWD -1,TIOWQ
```

```
QSLPW: EXP EQFIX           ;SLEEP FOR .GE. 1 SECOND
        XWD -1,-SLPQ
```

```
QEWV: EXP EQFIX           ;EVENT WAIT
```

XWD -1,-EWQ

QTIME: EXP EQLNKZ
XWD 0,QRQTBL

;WHEN QUANTUM TIME EXCEEDED

Name: TTFCOM Forced Commands Table

Description: Allows SCNSER to force a specified command to be executed for a job without having to put the command into the TTY buffer.

One entry, in SIXBIT format, for each command which SCNSER might want to force.

Defined in: SCNSER

Used in: COMCON, UUOCON, SYSINI, ONCE, NETMCR, NETVTM, CLOCK1, NETSER

Label	Content	Command
TTFCXC	HALT	Control-C
TTFCXD	.BYE	Dataphone disconnect
TTFCXH	.HELLO	Dataphone connect
TTFCXR	.RESTA	System restart
TTFCXK	KJOB	Kill job
TTFCXI	INITIA	Call Initializing CUSP
TTFCXJ	.FCONT	Forced Continue
TTFCXU	.DUMP	CCL Run of DUMP
TTFCXT	.TYPE	Retype line
TTFCXW	USESTA	CONTROL/T
TTFCXL	.NETLD	Network Reload
TTFCXS	HALT	CONTROL/C with no trapping
TTFCXX	CTEST	(For patching)

NOTE:

Value of each label is the position of that entry within the table. This value is plugged into LDBDDB when a command is to be forced.

The entries in TTFCOM are conditionally assembled; therefore all of the above listed entries will not be in every Monitor.

NAME: Teletype Buffers (Chunks)

Description: Used to hold characters which need to be typed on a TTY, or characters received from a TTY and not yet read by a program.

Buffers are set up dynamically, as needed, from a pool of monitor free core reserved for that purpose.

The association between a buffer and a line depends on pointers in the Line Data Block.

Buffers are built from four word "chunks" which are linked together as necessary.

Defined in: SCNSER

Format:

```

+-----+
! Prev Chunk Addr ! Next Chunk Addr !
+-----+
!           !           !           !
+-----+
!           !           !           !
+-----+
!           !           !           !
+-----+

```

Notes:

1. The ninth bit in each byte of data is used to indicate image mode.
2. If there is not another chunk in a given direction, the corresponding linkage will be zero.
3. All chunks which are not part of a buffer are linked together to form the "free list". The word TTFREE points to the first chunk of the free list.
4. The size of an individual buffer is limited by program action. An output buffer cannot exceed 80 characters. If an input buffer exceeds 80 characters (value of symbol TIWRNN) each receive interrupt will force the output of an XOFF. If an input buffer contains 90 characters (symbol TISTPN) no additional characters will be accepted from that line. A bell will be substituted for the echo of a

lost character.

5. See Description of Line Data Block for various pointers which are used with the TTY buffer.

Name: TTUUOT - TTCALL Dispatch

Description: This table contains pre-check and dispatch information for TTCALL UUOs. The bits in the LH are checked before dispatching. There is one entry for each TTCALL UUO.

Defined in: SCNSER

TTUUOT:	Pre-Check Bits			Dispatch Address	TTCALL 0
					TTCALL 1
					TTCALL 2
				Dispatch address	TTCALL 17
	0	4	18	35	

BIT	LABEL	MEANING
0	TC.ADR	This function must be address checked
1	TC.USR	This function must be at USER level else return
2	TC.USW	This function must be at USER level else wait
3	TC.ATW	This functin must be attached else wait
4	TC.ATR	This funtion must be attached else return

Name: TTYTAB TTY Table

Description: One entry per job, indexed by Job Number.

Table of controlling (attached) terminals for each job.

Defined in: COMMON

Reference in: COMDEV, FILIO, CLOCK1, COMCON, CPNSER, ERRCON,
IPCSER, PSISER, PTYSER, SCNSER, UUOCON

TTYTAB:			Job 0
		DDB ADDRESS	Job 1
		DDB ADDRESS	Job 2
		DDB ADDRESS	Job 3
		DDB ADDRESS	Job N

A zero entry indicates no attached terminal otherwise RH is controlling DDB for the job. There is always a TTY DDB for every job, even though no TTY need be attached. Thus UUOs look through TTYTAB(n) for a DDB since UUOs come from jobs.

Name: TYPTAB DEVICE TYPES TABLE

Description: Contains three letter generic device type
 prefix in SIXBIT format for all possible
 devices in the system.

 Used by UUOCON subroutine which finds a DDB
 given its universal I/O index.

Defined In: UUOCON (local symbol)

Used By: UUOCON

Format:

SIXBIT/DSK/
SIXBIT/DTA/
SIXBIT/MTA/
.
.

Name: UCLJMP CALL and CALLI UUO Dispatch Table

Description: Contains dispatch addresses for CALL and CALLI UUO's.

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

Indexed by one half the CALLI argument.

For CALL UUO's, a table lookup is done in UCLTAB to get routine number; for CALLI UUO'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: UUOCON (local symbol)

Used In: UUOCON

Format:

--	--
----	----
Customer Adr 4	Customer Adr 3
Customer Adr 2	Customer Adr 1

DEC Adr 0	DEC Adr 1

DEC Adr 2	DEC Adr 2
--	--
----	----

Special Notes:

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.

Name: UCLTAB CALL UWO Names Table

Description: Contains names of the CALL UWO'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. Table entries are SIXBIT expressions of the CALL names. There are corresponding dispatch addresses in the UCLJMP table.

Format:

CCLTAB:	Customer CALL M	
	Customer CALL 2	
	Customer CALL 1	
UCLTAB:	DEC CALL 0	
	DEC CALL 1	
	DEC CALL 2	
	DEC CALL N	

Special Notes:

1. Customers may extend the table in the negative direction with as many of their own CALL's as desired.
2. The value specified in a CALLI UWO corresponds to the position of the CALL UWO 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.
3. ALL CALL's above CALLI AC,55 do not have a corresponding CALL with a SIXBIT argument. In the future only CALLI's will be added by DEC.

Name: UDB Unit Data Block

Description: One UDB for each physical disk drive on the system (two if the drive is dual ported). Names are formed by concatenating the unit name with "CB", e.g., RPA0CB, DPB3CB.

Defined in: COMMOD

Referenced in: FILFND, FILIO, FILUO, ONCMOD, ERRCON, RPXKON, VMSE, SYSINI, KLSER, CPNSER, COMMON, DPXKON, FHXKON, FSXKON

*Indicates that details are shown on following pages

UNINAM	SIXBIT physical unit name
UNIOLOG	SIXBIT logical name within structure (HOMLOG)
UNIHID	SIXBIT home block ID name (HOMHID)
*UNISYS	Next UDB in system SIC LUN
UNISTR	Next UDB for STR STR Data Block
UNICHN	Next UDB on Channel CHN Data Block
UNIKON	Next UDB on Controller KON Data Block
*UNISWP	Next UDB for swapping CFS K4S
UNIHCT	Hard disk error statistics
UNISCT	Soft and hard error statistics
UNIMCT	Monitor detected error statistics
UNIERR	Device CONI at time of last hard error
UNISOF	CONI at time of last error before recovery
UNIHBN	Last logical block number on hard or soft error
UNIBRC	Number of buffered mode blocks read on unit
UNIBWC	Number of buffered mode blocks written on unit

UNIDRC	Number of dump mode blocks read on unit		
UNIDWC	Number of dump mode blocks written on unit		
UNIMRC	Number of Monitor blocks read on unit		
UNIMWC	Number of Monitor blocks written on unit		
UNIICT	Number of blocks swapped in from unit		
UNIOCT	Number of blocks swapped out to unit		
UNIMSC	Number of Monitor + swap seeks on this unit		
UNIUSC	Number of user mode seeks on this unit		
UNIPCT	Number of positioning failures	Number of soft+hard seek incomplete failures	
UNIFKS	Free K for swapping on this unit		
UNISDI	Last DATAI status before recovery attempted		
UNIHDI	Last DATAI status after first recovery failed		
UNIECT	# times error status returned for last operation		
UNIHNG	Hung timeout counters		
*UNISTS	Status code for unit		
*UNICCT	BCT	# of channel termination errors	
UNIHOM	1st home block address	redundant home blk adr	
UNIQUE	addr. of 1st FWQ DDB	Job NO of FWQ DDB	
UNIGRP	# blocks to try for on output	DDB with operation in progress	UNICDA
UNIBPU	# logical blocks per unit (returned by DSKCHR)		
UNIBPM	# of logical blocks/unit incl. maint. cyls.		
UNIPCI	# of blocks paged in from unit		

UNIPCO	# of blocks paged out to unit			
*UNICHR	BPC		BPT	Blocks per cylinder
*UNICPS	WPS		SPU	CPS
UNICYL	Current physical cylinder number			
UNIBLK	Logical block number within unit			
UNISAB	Addr 1st SAB in ring Addr of SPT table			UNISPT
UNITAL	# of free blocks on unit (reserved + FCFS)			
*UNIDES	Unit description bits for DSKCHR			
UNIPTR	-length swap SAT table Addr of swap SAT table			
UNISLB	1st logical block for swapping on unit			
UNIXRA	# blks read using extended ribs		# blks written using extended ribs	
UNICDA	Previous cont. of RH Addr of positioning DDB			
UNICOR	Addr first word for SAB, SPT tables if in non-free core		Addr last word for SAB, SPT tables if in non-free core	
UNIRCV	# of hung unit retries without success			
UNISWA	addr of current SWPLST entry		distance to swap block	
UNISWD	Distance to swap cylinder			
UNIQL	Length of position wait queue			
UNIBUC	# of blocks in 10/11 compatibility mode			
UNIDIA	Job # of job shutting down I/O (DIAG. UUO)		Addr of DDB of job shutting down I/O (DIAG.)	
UNISER	Drive serial number			
UNI2ND	0,,alternate if main/-1,, main if alternate			

UNITIM	Hung-Timer
UNIJOB	Previous RH UNIJOB Job no of RH UNICDA
*UNIAJB	DA resource status word
UNIGEN	Generation number of UDB
UNIDS2	Bit 0=1 is non-removable media
UNILAS	Last command issued to massbus device
UNISCR	Contents of control register at first error
UNIHCR	Contents of control register at end
UNISDR	Contents of data register at first error
UNIHDR	Contents of data register at end
UNIEBK	Drive registers saved here on error. LH has last error, RH has first error. Last word in block is command which cause the error. (0-16 words, determined by X'ERNO)

NOTES

UNISYS word (3)

+-----+-----+-----+-----+-----+				
Next UDB in system		SIC		LUN
0		17 18	25	30 35
Word	Bits	Byte	Description	
UNISYS	0-17		Core address of next UDB in system. Zero indicates last unit.	
UNISIC	18-25	UNYSIC	Number of SAT blocks in core for this unit.	
UNILUN	30-35	UNYLUN	Logical Unit Number within file structure for unit	

UNISWP word (7)

+-----+ Next UDB for swapping CFS K for swapping +-----+			
0	17	21	23 24 35
UNISWP	0-17	Address of next UDB for swapping	
UNIFCS	21-23	UNYCFS	Swapping Class of unit
UNIK4S	24-35	UNYK4S	Number of K for swapping on the unit

UNISTS word (36)

+-----+ code +-----+			
0			35
ICOD	0	Unit idle	
SWCOD	1	Seek wait; i.e., waiting to position with no implied transfer following	
SCOD	2	Seeking	
PWCOD	3	Position wait	
PCOD	4	Positioning	
TWCOD	5	Waiting to transfer data	
TCOD	6	Transferring data	
MDACOD	7	Unit useable only by MDA (mountable device allocator)	
OWCOD	10	Operator wait, file in transfer state	
OCOD	11	Operator wait, no file active	
OW2COD	12	Same as OWCOD but no message once a minute	
O2COD	13	Same as OCOD, but no message once a minute	

UNICCT word (37)

+-----+-----+-----+-----+			
	BCT	# of channel termination errors	
+-----+-----+-----+-----+			
0	8 9	17 18	35

UNIBCT 9-17 UNYBCT Number of slots left in BAT block for unit

UNICCT 18-35 Number of channel termination errors on this unit

UNICHR word (47)

+-----+-----+-----+-----+			
	BPC	BPT	blocks per cylinder
+-----+-----+-----+-----+			
0	8 9	17 18	35

UNIBPC 0-8 UNYBPC Number of blocks per cluster

UNIBPT 9-17 UNYBPT Number of blocks per track

UNIBPY 18-35 UNYBPY Number of blocks per cylinder

UNICPS word (50)

+-----+-----+-----+-----+			
	WPS	SPU	CPS
+-----+-----+-----+-----+			
0	8 9 10	17 18	35

UNIWPS 0-8 UNYWPS Number of words per SAT block

UNISPU 9-17 UNYSPU Number of SAT blocks on the unit

UNICPS 18-35 UNYCPS Number of clusters per SAT

UNIDES word (55)

+-----+ +-----+			
-------------------------	--	--	--

Word Label	Bit	Bit Label or Byte Pointer	Content
UNIDES	0	UNPRHB	Monitor must reread home block to ensure Pack ID correct. Set when a pack goes offline.
UNIDES	1	UNPOFL	Unit is offline.
UNIDES	2	UNPHWP	Unit is hardware write protected
UNIDES	3	UNPSWP	Unit is in a structure which is software write-protected.
UNIDES	4	UNPSAF	Unit is in a single access structure.
UNIDES	5	UNPZMT	STR mount count is zero.
UNIPRF	6	UNPPRF	Unit is in a private structure.
UNIUST	7-8		Unit Status, as follows:
		UNVPIM	0 - unit is up and pack mounted.
		UNVPBM	1 - unit is up and pack is being mounted.
		UNVNPM	2 - unit is up, but pack is not mounted.
		UNVDWN	3 - unit is down.
UNIUST	9	UNPMSB	Unit has more than one SAT block.
UNIUST	10	UNPNNA	No new access on STR.
UNIAWL	11	UNPAWL	STR is write protected for all jobs.
UNIAWL	12	UNPFUS	Unit got a file-unsafe.
UNIWMD	13	UNPWMD	Unit waiting for MDA to do something.

UNIWMD	14	UNPALT	Unit is dual-ported.
	15-17*		Code for argument passed to Monitor on DSKCHR UUO.
UNISCN	18-20		Data Channel Number
UNIKTP	21-26	UNYKTP	Controller type, as follows:
		TYPDR	0 - DR (Future drum, if any)
		TYPFH	1 - FH RC10 (Burroughs disk or Bryant drum)
		TYPDP	2 - DP RP10 (RP01-03 disks)
		TYPMD	3 - MD Bryant mass disk
		TYPFS	4 - FS RH10 with RS04
		TYPRP	5 - RP RH10 with RP04-06
UNIKNM	27-29	UNYKNM	Controller number within type.
UNIUTP	30-32	UNYUTP	Unit type
UNIPUN	33-35	UNYPUN	Physical unit number within controller.

*Not present in UDB

NOTES:

1. This word is returned by the DSKCHR UUO. Those items marked with an asterisk are not in the UDB, but are set in the word returned by the DSKCHR UUO.
2. DSKCHR Codes - specify argument passed to monitor

Codes

0	Generic device DSK
1	STR name abbreviation
2	STR name
3	Unit within a STR (e.g., DSKA0, DSKB1)
4	Kontroller class name (e.g., DP, RP)
5	Kontroller name (e.g., DPA, RPC)
6	Physical unit name (e.g., RPA0, RPC1)

3. KONTROLLER type starts at zero (e.g., DPA=0, DPB=1 ...).

UNI2ND Word

+-----+-----+-----+		
	Dual port flag	UDB address
+-----+-----+-----+		
0	17 18	35

Entire word is zero if this unit is not being accessed through dual ports. For units which are dual ported, UNI2ND will be:

XWD 0, UDB-addr-of-alternate If this is the main port
for the unit
XWD -1, UDB-addr-of-main If this is the alternate port

UNIAJB Word

1. Is -1 if no DA in progress on this unit
2. Is +N is Job N is allocating but no other jobs are waiting.
3. Is N,,N is JOB N is allocating and others are waiting to use the DA resource.

Name: UFB UFD DATA BLOCK.

Description: One data block for every UFD-File Structure pair which has an active file. All blocks for a file structure are linked together.

Defined In: COMMOD

Used By: FILUUG, FILEND, FILIO

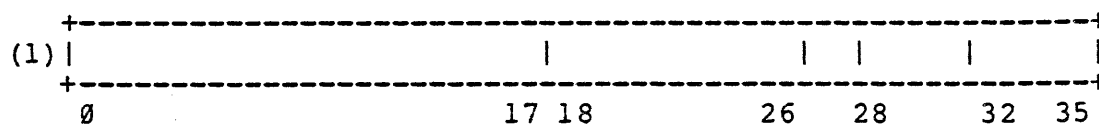
Format:

UFBTAL	Total of reserved + free blocks left in this UFD		
*UFBPPB	Next UFB, this user	privileges	UN1
UFBPT1	First retrieval pointer to UFD		
UFBWRT	FCFS Quote		Nr. Blks
UFBFSN	FSN		
UFBAUJ	Equals N is Job N owns the AU for this UFB		
UFBWAT	Non-zero is a job is waiting for the AU resource		
UFBZRB	Bit=1 is UFD has empty data blocks		

Notes:

1. UFBTAL - This count will go negative if the user has exceeded quota and is using overdraw. No new ENTER's allowed if this is 0 or negative. Total includes RIB's

2. UFBPPB byte definitions.



Word Label	Bits	Bit Label	Content
UFBPPB	0-17		Core address of next UFD data block for this PPN. (i.e., in another file structure).
UFBPRV	18-26		Access privileges for this UFD in this structure. Byte pointer UFYPRV used to load this byte into AC. UFRXLK = 4 Can do LOOKUPS UFRXCR = 2 Can create files UFRXRD = 1 Can read diretory
UFB1PT	27	UFP1PT	Set if UFBPT1 is the only retrieval pointer for this UFD.
UFBUN1	28-31		Logical unit number within file structure associated with first retrieval pointer.

3. UFBWRT - Bits 0-26= logged in first come first-served quota. This is never decremented. Bits 27-35=number of blocks written in the UFD itself.

Name: UFD User File Directory

Description: Contains the locations for all files in the STR belonging to a particular project programmer number. One UFD in each STR for each project programmer number having any files in that STR.

Defined in: COMMOD

Referenced in: FILUUD, FILFND, FILIO

Format:

File 1 Entry
File 2 Entry
File 3 Entry

Format of each entry

UFDNAM	File Name in Sixbit	
UFDEXT	Extension	CFP
UFDCFP		

Note: The CFP (Compressed File Pointer) specifies the relative supercluster within the STR where the RIB of the file can be found.

$$\text{CFP} = \begin{array}{l} \text{Relative Unit \#} \\ \text{within STR} \end{array} \times \begin{array}{l} \text{Number of} \\ \text{Superclusters} \\ \text{Per unit} \end{array} + \begin{array}{l} \text{Relative} \\ \text{Block \#} \\ \text{within that} \\ \text{unit} \end{array} / \begin{array}{l} \text{Number} \\ \text{Blocks per} \\ \text{Supercluster} \end{array}$$

The number of clusters in a "supercluster" is determined by
number clusters in STR.
 $2(18)$

Hence, every STR will have fewer than $2(18)$ superclusters, and the CFP will fit in 18 bits.

The concept of a supercluster is used only in connection with Compressed File Pointers, in directories.

Name: UPMP User Page Map Page

Description: Contains the Virtual Memory data base for each job and many other important job specific locations.

Defined in: S.MAC

Referenced in: VMSE, COMMON

Format:

User page fault trap instruction	!.UPPFT
User arithmetic trap instruction	!.UPAOT
User PDL overflow trap instruction	!.UPPDT
User trap 3 instruction	!.UPØ3T
MUO	!.UPMUO
MUO PC	!.UPMUP
Exec page fail word	!.UPEPF
User page fail word	!.UPUPF
.	!
.	!
.	!
WSBTAB	!WSBTAB
Bit Table for working set	!
AABTAB	!AABTAB
Bit table for access allowed bits	!
High order part of EBOX meter	!.LMEBH
Low order part of EBOX cycle meter	!.LMEBL

High order part of MBOX cycle meter	! .LMMBH
Low Order part of MBOX cycle meter	! .LMMBL
Push down List	! .JOBPDO
Protected Job data area	! .JOBPRO
Saved pointer to current virtual page number	! .UPLPS
! SWPLST index	! .UPSIX
Temporary locations used for swapping	! .UPTMP
Block of 20 ACs used to save job's ACs while getting PFH.EXE	! .UPUAC
Virtual time of first page fault	! .UPFFT
Virtual time of last fault	! .UPLFT
Real page faults ! Faults when page in core	! .UPVCT
Highest location gotten by CORE UUO or command	! .UPREL
Interval for virtual time interrupts	! .UPTMI
Count-down counters for virtual time interrupts	! .UPTMC
! =-1 if paging I/O pages not give back, =0 if given back!	
Job number	! .UPJOB
Total virtual memory a job has	! .UPMEM
Virtual address of the end of the high segment	! .UPHSE
Virtual address at the start of the high segment	! .UPHSS

! Non-zero if job virtual; LH for hi-seg, RH for low-seg	!.UPVRT
! Random collection of bits	!.UPBTS
! Count of non-accessible pages	!.UPANA
! Incremental count of page faults	!.UPICT
! Copy of .JBPFH on swap out	!.UPPFH
! Used by FILOP. to recover from a page fail	!.UPFOP
! Spy pages table	!.SPYTBL
! Used for address checking at interrupt level	!.UPHVA
! Ptr to swappable DDBs !	!.UPLST
! Header for cached free space	!.UPFCC
! Header for uncached free space	!.UPFCU
! Core loc. of SWITCH.INI !	!.UPSWI
! Ptr to logical name space ! T4	!.UPLNM
! Extended channel table loc.	!.UPCTA

Name: UUOTAB UUO Dispatch Address Table

Description: Contains address of operator-dependent UUO routines.

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

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.

Defined In: UUOCON (Local Symbol)

Used By: UUOCON

Format:

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

Name: WSBTAB WORKING SET BIT TABLE

Description: This bit table is found in the UPMP from location 440 to 456. If a bit is on in this table, then the relative page is in core.

Defined In: S.MAC

Used By: VMSE

Format:

