

File No. S370-36
Order No. SY20-0884-3

Systems

IBM Virtual Machine Facility/370: Data Areas and Control Block Logic

| Release 6 PLC 1

This publication, together with the *VM/370 System Logic and Problem Determination Guide, Volumes 1, 2, and 3*, is intended for use by system programmers responsible for updating VM/370. This publication contains descriptions of the major data areas and control blocks used by three of the components of VM/370, the Control Program (CP), the Conversational Monitor System (CMS), and the Remote Spooling Communications Subsystem (RSCS).

To use this publication effectively and to understand it thoroughly, the following publications are prerequisite:

IBM System/370 Principles of Operation

Order No. GA22-7000

IBM OS/VS, DOS/VS, and VM/370 Assembler Language,

Order No. GC33-4010

The IBM logo, consisting of the letters 'IBM' in a bold, sans-serif font, with horizontal lines through the letters.

| **Fourth Edition** (March 1979)

| This is a major revision of, and obsoletes, SY20-0884-2 and Technical
| Newsletters SN25-0413, SN25-0453, and SN25-0466. This edition applies
| to Release 6 PLC 1 (Program Level Change) of the IBM Virtual Machine
| Facility/370, and to all subsequent releases unless otherwise indicated
| in new editions or Technical Newsletters.

Technical changes and additions to text and illustrations are indicated
by a vertical bar to the left of the change.

Changes are periodically made to the information herein; before using
this publication in connection with the operation of IBM systems,
consult the latest IBM System/370 Bibliography, Order No. GC20-0001, for
the editions that are applicable and current.

Publications are not stocked at the address given below; requests for
copies of IBM publications should be made to your IBM representative or
to the IBM branch office serving your locality.

A form for readers' comments is provided at the back of this
publication. If the form has been removed, comments may be addressed to
IBM Corporation, VM/370 Publications, Dept. D58, Bldg. 706-2, P.O. Box
390, Poughkeepsie, New York 12602. IBM may use or distribute any of the
information you supply in any way it believes appropriate without
incurring any obligation whatever. You may, of course, continue to use
the information you supply.

This publication contains descriptions of major data areas and control blocks used by the three major components of VM/370. The three components are:

- The Control Program (CP)
- The Conversational Monitor System (CMS)
- The Remote Spooling Communications Subsystem (RSCS)

There are three sections and five appendixes, as follows:

- "Section 1. CP Data Areas and Control Blocks" contains information about CP data areas and control blocks.
- "Section 2. CMS Data Areas and Control Blocks" contains information on CMS data areas and control blocks.
- "Section 3. RSCS Data Areas and Control Blocks" contains information on RSCS data areas and control blocks.
- "Appendix A. CP and RSCS Equate Symbols" contains assembler language equate symbols used by CP and RSCS to reference data.
- "Appendix B. RSCS Control Areas" contains RSCS control areas that define constants and variables used during execution.
- "Appendix C. RSCS Request Elements" contains RSCS request elements that are the tables used by RSCS for task-to-task communication.
- "Appendix D. CMS Equate Symbols" contains CMS equate symbols.
- "Appendix E. Data Areas and Control Block References" contains information on the modules that reference data areas and control blocks.

OTHER VM/370 DATA AREAS AND CONTROL BLOCKS

Some data areas and control blocks that affect VM/370 service and support programs are not included in this publication. Information on these data areas and control blocks can be found in the IBM Virtual

Machine Facility/370: Service Routines Program Logic, Order No. SY20-0882.

RELATED PUBLICATIONS

This publication should be used in conjunction with:

IBM Virtual Machine Facility/370:

System Logic and Problem Determination Guide,

Volume 1 Control Program (CP), Order No. SY20-0886

Volume 2 Conversational Monitor System (CMS), Order No. SY20-0887

Volume 3 Remote Spooling Communication Subsystem (RSCS), Order No. SY20-0888

System Programmer's Guide, Order No. GC20-1807

Glossary and Master Index, Order No. GC20-1813.

For information on how to use the fourth component -- interactive problem control system -- and its facilities, the hardware and software support personnel or the installation system programmer should use:

IBM Virtual Machine Facility/370: Interactive Problem Control System (IPCS) User's Guide, Order No. GC20-1823.

HOW TO USE THIS PUBLICATION

This publication addresses and describes the major control blocks associated with CP, CMS, and RSCS. Generally, data areas, or scratch areas that are created and exist only during the execution of a particular module are not described in this publication. In this publication, the data areas and control blocks are arranged in alphabetical order by DSECT name.

The CMS and RSCS components operate under control of CP. Each component creates, updates, and erases its own control blocks and data areas.

Control blocks and data areas are blocks of related information applicable to one or more system functions. They are usually defined by the DSECT instruction. The blocks can reflect current status, history information, or combinations of both, applicable to VM/370 functions. Control blocks and data areas provide the linkage and information for the user, the hardware, and the programs to work as one entity for the successful execution of a job, task, or process.

For every data area or control block, a statement is given that defines the use of the data area or control block. This statement is followed by a formatted block showing the fields defined in the data area or control block and the displacement into the DSECT of that field.

The formatted blocks for CP and CMS control areas are 8 bytes wide, showing two fullwords per line. RSCS control blocks are 4 bytes wide.

Note: One exception to this width rule is the formatting for PSA, where the control areas are given in 16-byte width.

When the name of a field is too large to fit into the formatted line, a pointer to the definition of the field is used instead of the name of the field. This pointer usually takes the form A*1, A*2, etc. When there is a particularly large field (one that uses more than three or four lines of the formatted block), ellipses are used in the block to show that the displacement of this field is larger than can be shown in the block.

The use of slashes in a field indicates that the field is reserved for IBM's use.

The formatted block is followed by listing-related information such as the hexadecimal displacement of the field into the DSECT, the name of the field and its definition in the listing, and a brief description of the contents and meaning of the field.

The following terms in this publication, refer to the indicated support devices:

- "2305" refers to IBM 2305 Fixed Head Storage, Models 1 and 2.
- "270x" refers to IBM 2701, 2702, and 2703 Transmission Control Units or the Integrated Communications Adapter (ICA) on the System/370 Model 135.
- "2741" refers to the IBM 2741 and the 3767, unless otherwise specified.
- "3270" refers to a series of display devices, namely, the IBM 3275, 3276, 3277, and 3278 Display Stations. A specific device type is used only when a distinction is required between device types.

Information about display terminal usage also applies to the IBM 3138, 3148, and 3158 Display Consoles when used in display mode, unless otherwise noted.

Any information pertaining to the IBM 3284 or 3286 Printer also pertains to the IBM 3287, 3288, and 3289 printers, unless otherwise noted.

- "3330" refers to the IBM 3330 Disk Storage, Models 1, 2, or 11; the IBM 3333 Disk Storage and Control, Models 1 or 11; and the 3350 Direct Access Storage operating in 3330/3333 Model 1 or 3330/3333 Model 11 compatibility mode.
- "3340" refers to the IBM 3340 Disk Storage, Models A2, B1, and B2, and the 3344 Direct Access Storage Model B2.
- "3350" refers to the IBM 3350 Direct Access Storage Models A2 and B2 in native mode.
- "370x" refers to IBM 3704 and 3705 Communications Controllers.
- The term "3705" refers to the 3705 I and the 3705 II unless otherwise noted.

Contents

SUMMARY OF AMENDMENTS	ix	MN602: VM/370 Monitor DASTAP Utilization Record	52
SECTION 1. CP DATA AREAS and CONTROL BLOCKS	1	MN700: VM/370 Monitor Seeks Class Record	53
ACCTELOC: User Accounting Block	2	MN802: VM/370 Monitor System Profile Class	54
ACNTBLOK: Accounting Card Buffer Block	2	MNDEVLST: VM/370 Monitor Class 6 (DASTAP) Device List	55
ALOCBLOK: DASD Cylinder Allocation Block	4	MCNCOM: VM/370 Monitor Communications Area	56
BSCBLOK: Binary Synchronous Communication Control Block	6	MSSCOM: MSS Communications Control Block	59
BUFFER	8	NCPTBL: Named 370X Control Program Table	60
CCHREC: Channel Check Handler Record	9	NICBLOK: Network Interface Control Block	61
CCPARM: Communications Controller Parameter List	11	NPRFBL: Named 3800 Image Library Table	63
CHXBLOK and CHYBLOK: Virtual Channel-to-Channel Adapter Control Blocks	12	OBRECN: Unit Check Error Record (long Outboard Record)	64
CKPBLOK: Telecommunications Checkpoint Block	14	OBRECN: Unit Check Error Record (Short Outboard Record)	66
CONTASK: Console I/O Package	15	OWNDLIST: CP-Owned Volumes List	67
CORTABLE: Storage Allocation Table	17	PAGTABLE: Translation Page Table	67
CPEXELOC: CP Execute Block	18	PGBLOK: Pseudo Page Fault Stack Block	68
DDRREC: Reconfiguration Macro	19	PSA: Prefix Storage Area (Low Storage Locations)	69
DMPINREC: Dump File Information Record	20	PWDIBLOK: Password Invalid Block	78
DMPKYREC: Dump File Key Storage Record	21	REAL I/O CONTROL BLOCKS	79
DMPTRREC: Dump File Symbol Table Record	21	Real Channel Control Blocks	79
ECBLOK: Extension to VMBLOK for Virtual Machine with Relocate	22	Real Control Unit Blocks	79
ERRBLOK: Error Block Used to Build OBR/MDR	24	Real Device Control Blocks	79
ICBLOK: I/O Task Control Block	25	Input/Output Blocks	79
ICERBLOK: I/O Error Information Block	27	Network Interface Control Block	80
IRMBLOK: Intensive Error Recording Mode Block	31	RCHBLOK: Real Channel Block	81
JPSCBLOK: Journaling and Password Suppression Control Block	32	RCUBLOK: Real Control Unit Block	82
LOCKBLOK: Userid Lock Control Block	33	RCWTASK: Translated Virtual I/O CCW	84
MCHAREA: Machine Check Save Area	34	RDEVBLOK: Real Device Block	85
MCRECORD: Machine Check Handler Record	37	RECBLOK: DASD Page (Slot) Allocation Block	89
MDRREC: Miscellaneous Data Recording Record	38	RECPAG: Error Recording Page Record	90
MICBLOK: Virtual Machine Pointer List for VM/370 Hardware Assist	39	RSECTL: Real Spooling Control Block	91
MIHREC: Missing Interrupt Handler Error Record	40	RSPXBLOK: Real Device Extension Block	91
MNHDR: VM/370 Monitor Record Header	41	SAVEAREA	92
MN000: VM/370 Monitor Perform Class Record	42	SAVTABLE: First Page on Saved System DASD	93
MN001: VM/370 Monitor Perform Class Record	45	SDRBLOK: Statistical Data Recording Block	94
MN097: VM/370 Monitor Header Records	46	SEGTABLE: Translation Segment Table	95
MN098: VM/370 Monitor Trailer Record	46	SFBLOK: Spool File Block	96
MN099: VM/370 Monitor Suspension Record	47	SHCBLOK: Spool Hold Queue Block	98
MN10X: VM/370 Monitor Response Class Records	47	SHRTABLE: Named-Shared Segment Systems Table	99
MN20X: VM/370 Monitor Schedule Class Records	48	SPLINK: Spool Page Buffer Linkage Block	100
MN400: VM/370 Monitor User Class Record	49	SWPTABLE: Swap Table for Virtual Machine Paging	101
MN500: VM/370 Monitor Instruction Simulation Class Record	50	SYSLOC: System Low Storage Information Block	102
MN600: VM/370 Monitor DASTAP I/O Count Records	51	SYSTBL: Named System Table	103
		TNSREC: "T" Type Record Format (Environmental Recording)	104
		TREXT: Virtual Machine Tracing Extension to VMBLOK	105
		TRCBLOK: TOD Clock Comparator Request	107
		UDBFBLOK: User Directory Buffer Block	108
		UDEVBLOK: User Device Block	109
		UDIRBLOK: User Directory Block	110

UMACBLOK: User Machine Block111
VIRTUAL I/O CONTROL BLOCKS113
Virtual Channel Blocks113
Virtual Control Unit Blocks.113
Virtual Device Blocks.113
VCHBLOK: Virtual Channel Block114
VCONCTL: Virtual Console Control Block .115	
VCUBLOK: Virtual Control Unit Block. . .116	
VDEVBLOK: Virtual Device Block117	
VFCBLOK: Virtual Forms Control Buffer Block120
VMABLOK: Shared Systems Control Addition to VMBLOK.120
VMBLOK: Virtual Machine Control Block. .121	
VMCBLOK: Virtual Machine Communication Block128
VMCMHDR: VMCF Message Header129	
VMCPARM: VMCF Parameter List130	
VRRBLOK: Virtual Reserve/Release Block .131	
VSPLCTL: Virtual Spooling Control Block.132	
VSPXELOK: Virtual Spool Extension Block.133	
XINTEBLOK: External Interrupt Block . .134	
XOBR3211: Extended Outboard Recording Block135
SECTION 2. CMS DATA AREAS AND CONTROL BLOCKS.	
AETAB: Abend Termination Option Table. .138	
AEWSECT: Abend Recovery Workspace. . . .139	
ACTSECT: Active Disk Table140	
AFTSECT: Active File Table143	
ANCHSECT: Anchor Table145	
BATLSECT: CMS Batch User Job Limits. . .146	
BBOX: Boundary Box146	
BGCOM: DOS/VS Partition Communication Region.147
CMSTAXE: Terminal Attention Exit Element149
CVTSECT: Communication Vector Table as supported by CMS.150
DEGSECT: Debug Work Area152	
DEVSECT: Device Table DSECT.156	
DEVTAB: Device Table157	
DIOSECT: Disk I/O Work Area.160	
DMSCCB: Command Control Block.162	
DOSSECT: DOS Simulation Control Block. .164	
EDCB: Edit Control Block166	
ERDSECT: Error Handling Routine DSECT. .174	
EXTSECT: External Interrupt Work Area. .177	
EXTUAREA: External User Area179	
FCBSECT: Simulated OS Control Blocks . .180	
FCHTAB: Fetch Table.184	
FICL: First In Class Block185	
FRDSECT: Free Chain Element Header Blocks.186
FSCBD: File System Control Block188	
FSTD: File Status Table Entry DSECT. . .189	
FSTSECT: File Status Table190	
FVSECT: Fixed Variable Storage Work Area for CMS File System.191
IHADECB: Data Event Control Block. . . .195	
IOSECT: I/O Interrupt Save Area.196	
KEYSECT: Disk Key Table DSECT for BDAM Simulation.197
LDRST: Loader Storage Area198	
LUETAB and LUEPR: Logical Unit Block Table202

NICL: Number in Class.204
NUCON: Nucleus Constant Area205	
OESECT: Major CSECT for all I/O Operation Lists219
OSFST: OS File Status Table.222	
OVSECT: Describes the First Few Locations of DMSOVS224
PCTAB: Program Check Option Table. . . .224	
PDSSECT: Directory Table for EPAM Simulation.225
PGNSECT: Program Interrupt Work Area . .226	
PIBADR: Program Information Block. . . .227	
PIB2TAB: Program Information Block Extension228
PUBADR: Physical Unit Block Table. . . .229	
PUBOWNER: Physical Unit Block Ownership Table230
SSAVE: System Save Area.231	
SUBSECT: Subset Work Area.233	
SVCSECT: SVC Interrupt Storage234	
SVEARA: LTA and PP Save Area DSECT . . .238	
SYSCOM: System Communication Region. . .239	
SYSNAMES: Saved Systems Names.242	
TSCBLKS: TSO Control Blocks.243	
USAVE: User Save Area.245	
USERSECT: User Work Area245	

SECTION 3. RSCS DATA AREAS AND CONTROL BLOCKS.	
ASYNE: Asynchronous Exit Element248	
BUPDSECT: SML Telecommunications Buffer.249	
COMDSECT: Address Constants as Pointers.250	
DEVTABLE: NPT Device Table251	
FREEE: A Free Element on the Supervisor Element Queue252
GIVE Request Table253	
GIVEE: A GIVE Element.253	
IOE: An I/O Element.254	
IOTABLE: An I/O Table.255	
LINKTABL: Link Table256	
REQBLOCK: NPT Request Block.258	
ROUTE: Routing Table Entry259	
SVECTORS: Low Storage Definitions. . . .260	
TAG: RSCS File Descriptor.263	
TAGAREA.265	
TAKE Request Table265	
TANKDSEC: SML Unit Record Tank266	
TAREA: A Task Save Area.267	
TASKE: A Task Element.269	
TCTDSECT: Task Control Table270	

APPENDIXES273

APPENDIX A. CP and RSCS EQUATE SYMBOLS .275	
VM/370 Device Classes, Types, Models, and Features.276
VM/370 EQUATE SYMBOLS -- Machine Usage .278	
VM/370 EQUATE SYMBOLS -- Extended Control Registers279
VM/370 EQUATE SYMBOLS -- CP Usage. . . .280	
VM/370 Registers283	

APPENDIX E. RSCS CONTROL AREAS285	
AXS Monitor Control Area286	
REX Monitor Control Area287	
SML Monitor Control Area288	

APPENDIX C. RSCS REQUEST ELEMENTS.291	Message Request Element.303
Command ALERT Element Format A1.292	Operational Notes.303
Operational Notes.292	Print Table304
Command ALERT Element Format A2.293	Operational Notes.304
Operational Notes.293	Terminate Request Element.305
Command ALERT Element Format L0.294	Operational Notes.305
Operational Notes.294	APPENDIX D. CMS EQUATE SYMBOLS307
Command ALERT Element Format L1.296	CMS Usage Equates.308
Operational Notes.296	CMS Register Equates309
Command ALERT Element Format L2.297	APPENDIX E. DATA AREAS AND CONTROL	
Operational Notes.297	BLOCK REFERENCES.311
Command ALERT Element Format L3 (also		CF Control Block References.312
Message Alert Element).298	CMS Control Block References321
Operational Notes.298	RSCS Control Block References.326
Command Request Element.299		
Operational Notes.299		
File Request Element300		
Operational Notes.300		
Line Alert Element302		
Operational Notes.302		

FIGURES

Figure 1. CP Control Block Relationships.....1
Figure 2. CMS Control Block Relationships.....137

3800-1 PRINTER SUPPORT

New: Documentation and Program Support

VM/370 now offers support for the 3800-1 unit as a dedicated virtual machine printer. The 3800-1 is also supported as a VM/370 spooling device.

3850-2 VIRTUAL MACHINE MSS SUPPORT

New: Program and Documentation

VM/370 now supports the 3850-2 MSS to permit most operating systems that are running in the virtual machine environment access to data on MSS virtual volumes.

PASSWORD-ON-THE-COMMAND-LINE SUPPRESSION

New: Program Feature

VM/370 now supports the suppression of the entering of passwords on the command line for LOGON, AUTOLOG, and LINK. The intent is to force passwords to be typed upon a mask. The new support is specified via the SYSJRL macro in DMKSYS. It is optional and must be implemented at system generation time. Privilege class A users can use the JOURNAL operand of either the SET or QUERY commands.

MULTIPLE ALTERNATE CONSOLE SUPPORT

New: Program and Documentation

VM/370 supports the specification of multiple alternate consoles at system generation time.

MONITOR ENHANCEMENTS SUPPORT

New: Program and Documentation

VM/370 supports the enhancement to the Monitor module which permits the analyst the option to specify periodic closing

of the active Monitor spool file frequently enough to support real time data reduction and display.

SECURITY JOURNALING SUPPORT

New: Program Feature

VM/370 now supports the journaling of LOGONS and AUTOLOGS specifying invalid passwords and the journaling of all linkages. This is accomplished via the generation of type 04, 05, and 06 accounting records. The new support is specified in the SYSJRL macro in DMKSYS.

4331 AND 4341 PROCESSOR SUPPORT

New: Program and Documentation

VM/370 supports 4331 and 4341 processors offering compatibility with the new model IDs as well as the S/370 RAS function subset.

MISCELLANEOUS

New: Documentation and Program

The following features and enhancements are now supported by VM/370.

- 3203-5 Unit
- Special Messages facility
- Trace Table size as a system generation option
- Modification of Shared Segment handling
- 3031 Alternate Processor
- 12 and 16 Megabyte Processors
- Directory hooks

Summary of Amendments
for SY20-0884-2
as updated by SN25-0461
VM/370 Release 5 PLC 12

VARY PROCESSOR SUPPORTED BY VM/370

New: Documentation and Program Support

When a system has been generated for attached processor operations, use of a new command, VARY PROCESSOR ONLINE/OFFLINE, facilitates the transition to or from uniprocessor mode on the main processor. This command can be used to vary a specified processor offline or online without any serious disruption to system users.

ALTERNATE TRACK FOR 3340/3344
NOW SUPPORTED BY VM/370

New: Documentation and Program Support

New code has been added for VM/370 support of 3340/3344 alternate track facility. These changes affected the following modules:

IOBLOK
IOERBLOK

Section 1. CP Data Areas and Control Blocks

This section contains descriptions of the major CP data areas and control blocks. Figure 1 shows the relationships of control blocks to each other.

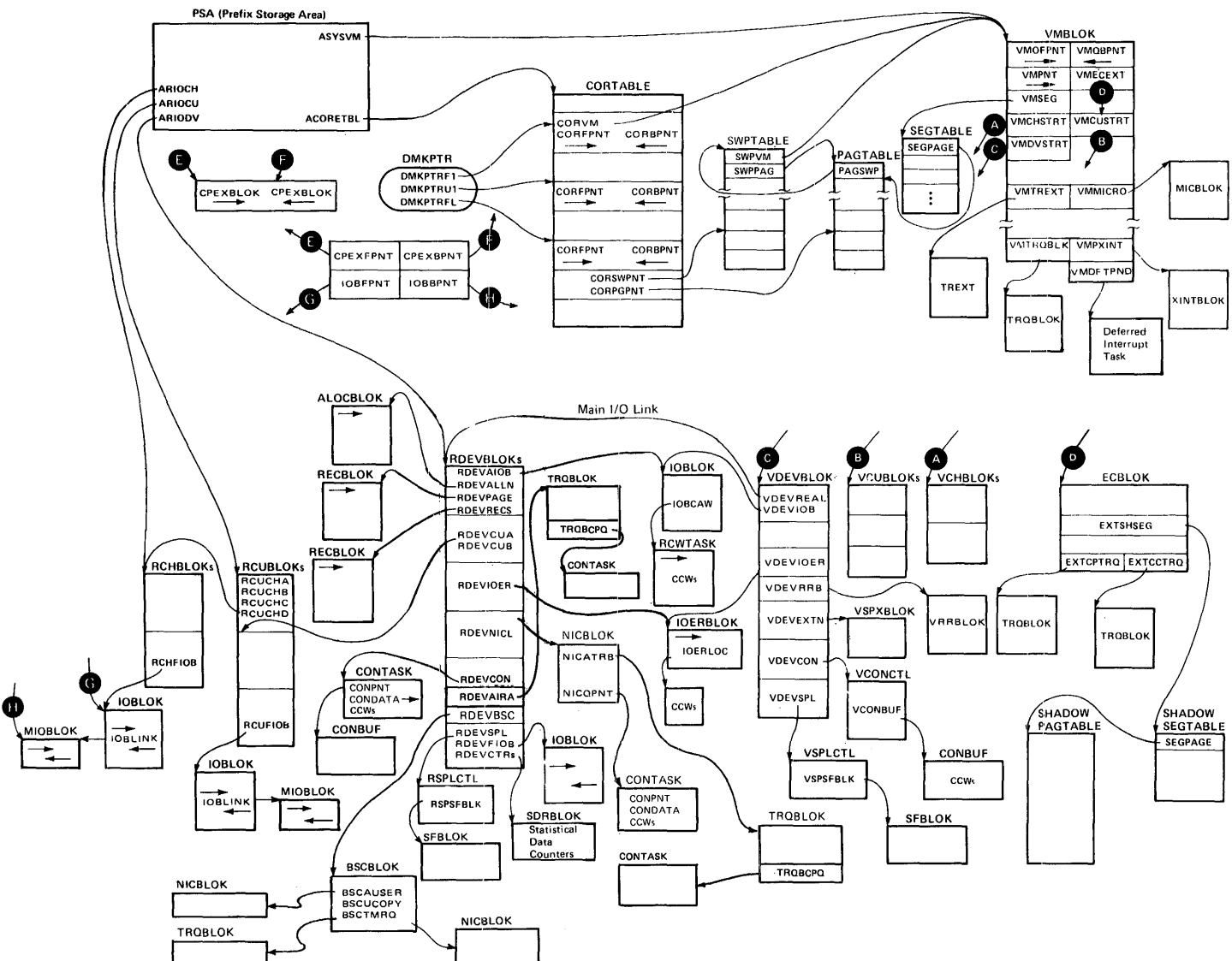
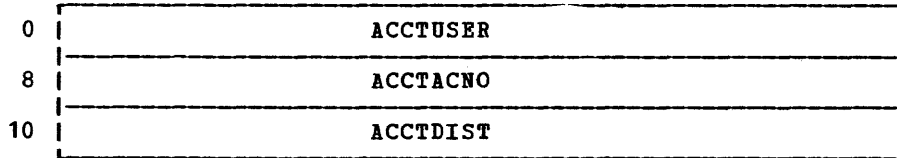


Figure 1. CP Control Block Relationships

ACCTBLOK, ACNTBLOK

ACCTBLOK: USER ACCOUNTING BLOCK

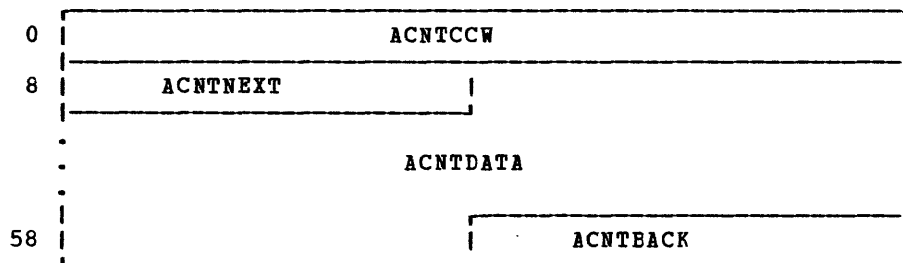
ACCTBLOK provides header information for spool files. The VMACCUNT field in the VMELCK points to ACCTBLOK.



Hexadecimal Displacement	Field Name			Field Description, Contents, Meaning
0	ACCTUSER DS	CL8		Virtual machine identification
8	ACCTACNO DS	CL8		Virtual machine accounting number
10	ACCTDIST DS	CL8		Virtual machine distribution number
	ACCTLENG EQU	(*-ACCTBLOK)/8		Size of ACCTBLOK in doublewords (X'03')

ACNTBLOK: ACCOUNTING CARD BUFFER BLOCK

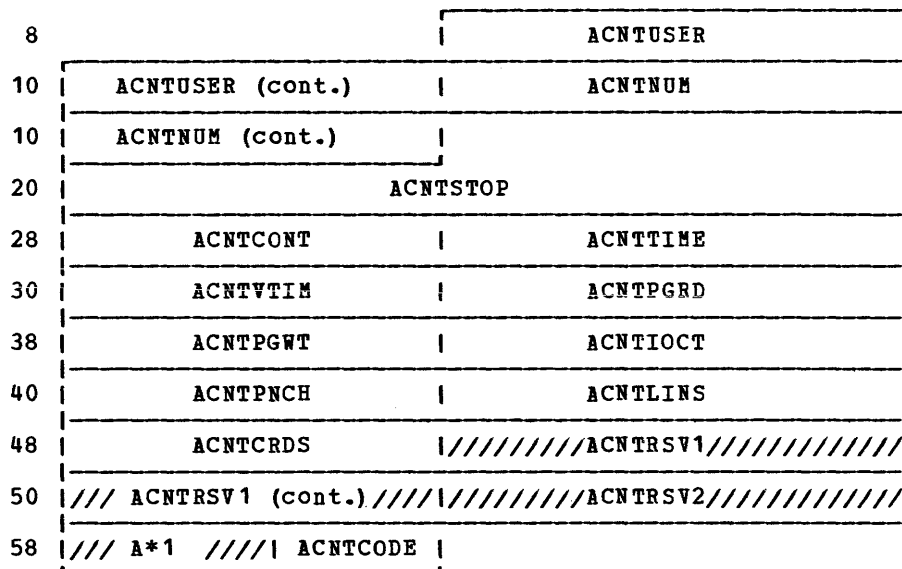
ACNTBLOK provides accounting and statistical information on each user that has used VM/370 facilities. The ARSPAC field in the Prefix Storage Area (PSA) points to the start of the chain of ACNTBLOKs.



Hexadecimal Displacement	Field Name			Field Description, Contents, Meaning
0	ACNTCCW DS	D		Punch CCW for accounting card
8	ACNTNEXT DS	F		Address of next ACNTBLOK in chain
C	ACNTDATA DS	CL80		Accounting information (see "Format for User Cards")
5C	ACNTBACK DS	F		Address of previous ACNTBLOK in chain
	ACNTSIZE EQU	(*-ACNTBLOK)/8		Size of ACNTBLOK in doublewords (X'0C')

• Format for User Cards

The fields below represent the 80 bytes defined by ACNTDATA in the ACNTBLCK data area.



Hexadecimal Displacement	Field Name	Field Description, Contents, Meaning
	ORG ACNTDATA	
C	ACNTUSER DS CL8	Virtual machine identification
14	ACNTNUM DS CL8	Virtual machine accounting number
1C	ACNTSTOP DS CL12	Date and time of accounting MMDDYYHHSS
28	ACNTCONT DS 1F	Number of seconds connected
2C	ACNTTIME DS 1F	Milliseconds of processor time used
30	ACNTVTIM DS 1F	Milliseconds of virtual processor time used
	ORG ACNTTIME	
2C	ACNTDEVC DS XL4	Device code (CTFM); see the DEVTYPE copy file
30	ANCTNCYL DS 1H	Number of cylinders of T-disk space
34	ACNTPGRD DS 1F	Total page reads
38	ACNTPGWT DS 1F	Total page writes
3C	ACNTIOCT DS 1F	Virtual SIO count for nonspooled I/O
40	ACNTPNCH DS 1F	Virtual card count for spooled punch
44	ACNTLINS DS 1F	Virtual line count for spooled printer
48	ACNTRCRDS DS 1F	Virtual card count for spooled reader
4C	ACNTRSV1 DS 2F	Reserved for IBM use
54	ACNTRSV2 DS XL6	Reserved for IBM use
5A	ACNTCODE DS 1H	Accounting card identification code
	<u>Card Codes for ACNTCODE</u>	
	DC C'0'	User formatted accounting card
	DC C'x1'	User virtual machine accounting card
	DC C'x2'	User dedicated device accounting card
	DC C'x3'	User temporary disk space accounting card

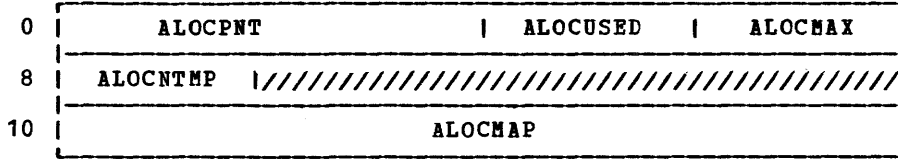
where:

x = C if the card is initiated via a DIAGNOSE Code X'4C'
 x = 0 if the card is initiated via CP command processing

ALOCBLOK

ALOCBLOK: DASD CYLINDER ALLOCATION BLOCK

ALOCBLOK provides information on the temporary disk space available to a virtual machine. The RDEVALLN field in the RDEVBLK points to the ALOCBLOK.



Hexadecimal Displacement	Field Name				Field Description, Contents, Meaning
0	ALOCPNT	DS	1F		Pointer to next ALOCBLOK on chain
4	ALOCUSED	DS	1H		Number of nontemporary cylinders
6	ALOCMAX	DS	1H		Maximum number of cylinders available
8	ALOCNTMP	DS	1H		Number of nontemporary cylinders
A		DS	3H		Reserved for IBM use
10	ALOCHMAP	DS	0F		Cylinder allocation bit map

Bits defined in ALOCHMAP
 0 = Cylinder is available
 1 = Cylinder has been assigned

Note: The size of ALOCHMAP is variable and depends upon the number of cylinders on the device. Generally, the size of the ALOCBLOK is determined by the following formula:

$$ALOC\text{SIZE (doublewords)} = \frac{(ALOC\text{MAX}+63)}{64} + 2 = \frac{\text{No. of Cylinder}}{\text{Bits per doubleword}} + \text{header}$$

where:

- ALOCMAX for 2305-1 = 48 cylinders
- for 2305-2 = 96 cylinders
- for 2314 = 203 cylinders
- for 3330-1 = 404 cylinders
- for 3330-2 = 404 cylinders
- for 3330-11 = 808 cylinders
- for 3333-1 = 404 cylinders
- for 3333-11 = 808 cylinders
- for 3340-35 = 349 cylinders
- for 3340-70 = 698 cylinders
- for 3350 = 555 cylinders
- for all others = 1 cylinder

Note that any bits in the map that represent cylinders not present on the device are set to 1.

For Temporary Disk Allocation Blocks

		ORG	ALOCUSED	
4	ALOCYL1	DS	1H	First cylinder of T-disk area
6	ALOCYL2	DS	1H	Last cylinder of T-disk area

Bytes defined in ALOCHMAP
 X'00' = Cylinder is available

X'AA' = Cylinder has been allocated

Note: The size of the T-disk ALOCCMAP is variable and depends upon the number of cylinders in the range ALOCCYL1 to ALOCCYL2. Generally, the size of a given block is determined by the following formula:

$$\text{ALOC SIZE (doublewords)} = \left\{ \frac{(\text{ALOCCYL2} - \text{ALOCCYL1} + 1) * 7}{8} \right\} + 2 =$$

$$\left\{ \frac{\text{Number of Cylinder (inclusive)}}{\text{Bytes per doubleword}} \right\} + \text{header}$$

Note that bytes for cylinders that are not available are marked assigned.

BSCBLOK

BSCBLOK: BINARY SYNCHRONOUS COMMUNICATION CONTROL BLOCK

BSCBLOK provides status, control information buffers (necessary for polling and addressing), and channel programs for 3270 remote equipment. The RDEVESC field in the RDEVBLOK points to the BSCBLOK.

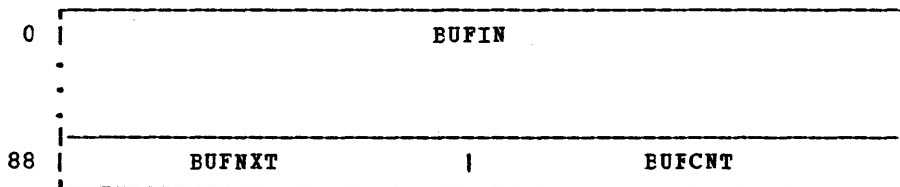
0	BSCSCCW1			
8	BSCSCCW2			
10	BSCSCCW3			
18	BSCPCCW1			
20	BSCPCCW2			
28	BSCPCCW3			
30	BSCPCCW4			
38	BSCECCW1			
40	BSCECCW2			
48	BSCUECCW			
50	BSCSEL			B*1
58	B*2	B*3	BSCINDEX	//////////BSCRESVD//////////
60	BSCSPTR			BSCAUSER
68	BSCUCOPY			BSCRSTRT
70	BSCCNT	BSCSENSE	BSCRCVD	BSCSEND
78	//////////BSCUSER1//////////			BSCRROBN
80	BSCMRQ		BSCRESP	
88	BSCREAD			

Hexadecimal Displacement	Field Name			Field Description, Contents, Meaning
0	BSCSCCW1	DS	1D	CCW for write reset operation
8	BSCSCCW2	DS	1D	CCW for addressing or selection
10	BSCSCCW3	DS	1D	CCW for read response to selection
18	BSCPCCW1	DS	1D	CCW for write reset operation
20	BSCPCCW2	DS	1D	CCW for general/specific polling
28	BSCPCCW3	DS	1D	CCW for NOP command
30	BSCPCCW4	DS	1D	CCW for read text
38	BSCGCCW1	DS	1D	CCW for write error response
40	BSCGCCW2	DS	1D	CCW to transfer control to read CCW
48	BSCUECCW	DS	1D	CCW for read response on time-out
50	BSCSEL	DS	7X	Addressing/polling entry
57	BSCFLAG	DS	1X	B*1 BSCBLOK flags
<u>Bits defined in BSCFLAG</u>				
	BSCRVI	EQU	X'80'	Sending RVI response
	BSCENQ	EQU	X'40'	Enqueued in data from station
	BSCCOPY	EQU	X'20'	COPY function is active
	BSCOPIED	EQU	X'10'	Initiate COPY function
	BSCREGEN	EQU	X'08'	Regeneration error
	BSCSTRQ	EQU	X'04'	Ignore input processing
	BSCLOG	EQU	X'02'	Bypass FORCE message at logoff
	BSCSCAN	EQU	X'01'	Second scan for write request
58	BSCFLAG1	DS	1X	B*2 BSCBLOK flags
<u>Bits defined in BSCFLAG1</u>				
	BSCFTB	EQU	X'80'	Station transmitted block record
	BSCIGN	EQU	X'40'	Ignore block record
	BSCPA1	EQU	X'20'	Indicator to call DMKCFMBK
	BSCINBID	EQU	X'10'	Initial bid sequence required
	BSCFORCE	EQU	X'08'	User FORCE in progress
	BSCHALT	EQU	X'04'	Halt I/O has been issued for this device
59	BSCLINE	DS	1X	B*3 Line coordinate for input area
5A	BSCINDEX	DS	1H	Index value for available space in input buffer
5C	BSCRESVD	DS	4X	Reserved for IBM use
60	BSCSPTR	DS	1F	Write CCW string address and/cr address of buffer
64	BSCAUSER	DS	1F	Address of active resource
68	BSCUCOPY	DS	1F	Address of COPY requestor's NICBLOK
6C	BSCRSTRT	DS	1F	Address of restart CCW string
70	BSCCNT	DS	1H	Retry count
72	BSCSENSE	DS	1H	Sense bytes from remote station
74	BSCRCVD	DS	1H	Expected received ACK (ACK-0/ACK-1)
76	BSCSEND	DS	1H	Sending ACK (ACK-0/ACK-1)
78	BSCUSER1	DS	1F	Reserved for IBM use
7C	BSCRROBN	DS	1F	Address of active user in queue
80	BSCMRQ	DS	1F	Pointer to TRQBLOK for poll delay
84	BSCRESP	DS	1H	Response buffer for selection
86	BSCREAD	DS	CL264	Read buffer for polling
	BSCSIZE1	EQU	*(BSCREAD+1)	Read buffer size in bytes
	BSCSIZE2	EQU	(BSCREAD-BSCBLOK)	ESC header size in bytes
	BSCSIZE	EQU	(*-BSCBLOK+7)/8	BSC blocksize in doublewords (X'19')

BUFFER

BUFFER

BUFFER is a buffer area that contains console input to be used by CP. The VCONREUF field in the VCONCTRL block points to BUFFER.



Hexadecimal Displacement	Field Name			Field Description, Contents, Meaning
0	BUFIN	DS	CL136	Input line
88	BUFNXT	DS	1F	Pointer to next byte in BUFFER
8C	BUFCNT	DS	1F	Count of characters in input line
<u>Bits defined in BUFCNT</u>				
	BUFINLTH	EQU	L'BUFIN'	Size of input line in bytes (136)
	BUFSIZE	EQU	(* - BUFFER) / 8	Size of console buffer in doublewords (X'12')

CCHREC: CHANNEL CHECK HANDLER RECORD

CCHREC provides statistical data for error recovery and/or error recording related to a previously performed channel operation that did not successfully complete.

0	C*1	C*2	C*3	C*4	///CCSW2REV//	C*5	/C*6//
8	CCDATE						
10	CCCPUID			CCHMDL	CCHMCEL		
18	CCPROGID						
20	FAILADD						
30	FAILCCW						
38	FAILCSW						
40	FAILECSW			CCDEV TYP			
48	C*7	CCHCUA		CCHCHCUA	CCHCLOGL		
50	CCHLOG						

Hexadecimal Displacement	Field Name	Field Description, Contents, Meaning
0	CCRECTYP DS 1X	C*1 Record type
1	CCOPSYS DS 1X	C*2 Operating system
2	CCSW1 DS 1X	C*3 Switch 1
3	CCSW2 DS 1X	C*4 Switch 2
4	CCSW2REV DS 2X	Reserved for IBM use
6	CCRECNT DS 1X	C*5 Record count
7	CCRECNT1 DS 1X	C*6 Reserved for IBM use
8	CCDATE DS 1D	Date and time
10	CCCPUID DS 1F	Processor identification
14	CCHMDL DS 1H	Processor model number (for example, 0158, 0168, etc.)
16	CCHMCEL DS 1H	Maximum length of machine extended logout area (model dependent)
18	CCPROGID DS 1D	User identification
20	FAILADD DS 8H	Active I/O units
30	FAILCCW DS 1D	Failing CCW
38	FAILCSW DS 1D	Failing CSW
40	FAILECSW DS 1F	Failing ECSW
40	ORG FAILECSW IGPRGFLG DS CL1	Program flag bits
<u>Bits defined in IGPRGFLG</u>		
	CCHSIOB EQU X'80'	Start I/O bit
	CCHINTB EQU X'40'	Interrupt bit
	CCHTIO EQU X'20'	Test I/O bit
	CCHHIO EQU X'10'	Halt I/O bit
	CCHSNSB EQU X'04'	Sense data stored bit
	CCHCNTB EQU X'02'	Count valid bit
	CCHNRYB EQU X'01'	No retry bit

CCHREC

Hexadecimal Displacement	Field Name		Field Description, Contents, Meaning
41	IGBLAME DS	CL1	Probable source of error
	<u>Bits defined in IGBLAME</u>		
	CCHCPU EQU	X'80'	Processor is source of error
	CCHCHNL EQU	X'40'	Channel is source of error
	CCHSCUB EQU	X'20'	Storage control unit is source of error
	CCHSTG EQU	X'10'	Storage is source of error
	CCHINTFC EQU	X'08'	Control unit is source of error
42	IGVALIDB DS	CL1	Validity indicator bits
	<u>Bits defined in IGVALIDB</u>		
	CCHINTFV EQU	X'80'	Interface address valid
	CCHRCV EQU	X'10'	Sequence code valid
	CCHUSV EQU	X'08'	Unit status valid
	CCHCMDV EQU	X'04'	Command address valid
	CCHCAV EQU	X'02'	Channel address valid
	CCHDAV EQU	X'01'	Device address valid
43	IGTERMSQ DS	CL1	Termination/sequence code bits
	<u>Bits defined in IGTERMSQ</u>		
	COMP SYS EQU	X'C0'	System reset
	COMPSEL EQU	X'80'	Selective reset
	COMPRES EQU	X'40'	Stop, Stack, or normal termination
	CCHIOH EQU	X'10'	I/O interface inoperative
	COMPID EQU	X'00'	Interface disconnect
	CCHDI EQU	X'08'	I/O error alert
	<u>Sequence Code Bits</u>		
	RTCODE0 EQU	X'00'	Retry code values for the constructed ECSW
	RTCODE1 EQU	X'01'	
	RTCODE2 EQU	X'02'	
	RTCODE3 EQU	X'03'	
	RTCODE4 EQU	X'04'	
	RTCODE5 EQU	X'05'	
	RTCODE6 EQU	X'06'	
	RTCODE7 EQU	X'07'	
44	CCDEVTYP DS	1F	CP device type
48	CCHANID DS	CL1	C*7 Channel identification
49	CCHCUA DS	CL3	Actual failing device address
4C	CCHCHCUA DS	2X	Address from machine location X'EA'
4E	CCHCLOGL DS	2X	Length of channel logout
	CCHSIZE EQU	(*-CCHREC)/8	Size in doublewords (X'0B')
50	CCHLOG80 DS	0CL112	2880 channel - 112 bytes
50	CCHLOG70 DS	0CL24	2870 channel - 24 bytes
50	CCHLOG60 DS	0CL24	2860 channel - 24 bytes
50	CCHADDR DS	1F	Unit address stored by integrated channel
	CCHSIZE1 EQU	(*-CCHREC)	Size in bytes for integrated channel
54	CCHLOG45 DS	0CL96	Model 145 integrated channel (96 bytes)
54	CCHLOG35 DS	0CL24	Model 135 integrated channel (24 bytes)

CCPARN: COMMUNICATIONS CONTROLLER PARAMETER LIST

CCPARN provides control information used for loading and controlling the 370X Communication Controller Network Control Program, Emulation Program, and Partitioned Emulation Program and their attached resources.

0	CCPNAME								
8	CCPADDR				CCPSIZE				
10	CCPPSIZE				CCPENTRY				
18	C*1		C*2		C*3	/C*4//	CCPESTOR		
20	CCPHBFSZ		CCPHBFNO		C*5		C*6		CCPMAxid
28	CCPRESID								
.	.								
.	.								
.	.								

Hexadecimal Displacement	Field Name				Field Description, Contents, Meaning
0	CCPNAME	DS	CL8		CPNAME specified in NAMENCP macro
8	CCPADDR	DS	1F		Origin of control program image
2	CCPSIZE	DS	1F		Control program size in bytes
10	CCPPSIZE	DS	1F		Parameter list size in bytes
14	CCPENTRY	DS	1F		Control program entry point address
18	CCPTYPE	DS	1X	C*1	Control program type flag
<u>Bits defined in CCPTYPE</u>					
	CCPTNCP	EQU	X'01'		Network control program
	CCPTPE	EQU	X'02'		270x emulation program
	CCPTPEP	EQU	X'03'		Partitioned emulation program
19	CCPCAONE	DS	1X	C*2	First channel adapter type flag
1A	CCPCATWO	DS	1X	C*3	Second channel adapter type flag
<u>Bits defined in CCPAONE and CCPATWC</u>					
	CCPTYPE1	EQU	X'01'		Channel adapter type 1
	CCPTYPE2	EQU	X'02'		Channel adapter type 2
1B	CCPRSV1	DS	1X	C*4	Reserved for IBM use
1C	CCPSTOR	DS	1F		370x storage size specified (in bytes)
20	CCPHBFSZ	DS	1H		Buffer size from HOST macro
22	CCPHBFNO	DS	1H		Number of buffers in read list
24	CCPPAD0	DS	1X	C*5	First buffer pad count (in bytes)
25	CCPPAD1	DS	1X	C*6	Subsequent buffer pad count
<u>HOST Values Required by VM/370 Support for 370x</u>					
	CCPVPAD0	EQU	34		34-byte pad in first bisynchronous transmitter buffer
	CCPVPAD1	EQU	34		34-byte pad in subsequent buffers
26	CCPMAxID	DS	1H		Highest resource ID defined
28	CCPRESID	DS	1F		Resource ID definition
		ORG	CCPRESID		Definition breakdown
28	CCPRSTYP	DS	1X		Resource type flag
29	CCPRSTAT	DS	1X		Resource initial status flags
2A	CCPRSTEP	DS	1H		Subchannel address when in EF mode

CHXBLOK,CHYBLOK

CHXBLOK AND CHYBLOK: VIRTUAL CHANNEL-TO-CHANNEL ADAPTER CONTROL BLOCKS

CHXBLOK and CHYBLOK provide the necessary control for a virtual machine using a virtual channel-to-channel adapter (CTCA). The VDEVREAL field in the VDEVBLCK points to CHXBLOK and CHYBLOK for virtual CTCAs.

0	CHXOTHR				CHYOTHR			
8	X*1	X*2	X*3	X*4	Y*1	Y*2	Y*3	Y*4
10	CHXNCCW				CHYNCCW			
18	CHXRCNT				CHYRCNT			
20	CHXSTAT		CHXYADD		CHYSTAT		CHYXADD	
28	CHXIDAW				CHYIDAW			
30	CHXCNCT				CHYCNT			
38	CHXDATN				CHYDATN			

Note: As indicated in the illustrated block, the CHXBLOK and CHYBLOK are interleaved with a 4-byte displacement. The X-side VDEVBLCK points to the +0 slot, the Y-side VDEVBLCK points to the +4 slot; however, once the virtual connection is made, either side can be the X-side or the Y-side since this interleaved arrangement makes the control block references completely symmetrical. The dual DSECT definition allows the active adapter (defined to be the X-side, arbitrarily) to reference both adapter sides concurrently without knowing whether it is at +0 or +4.

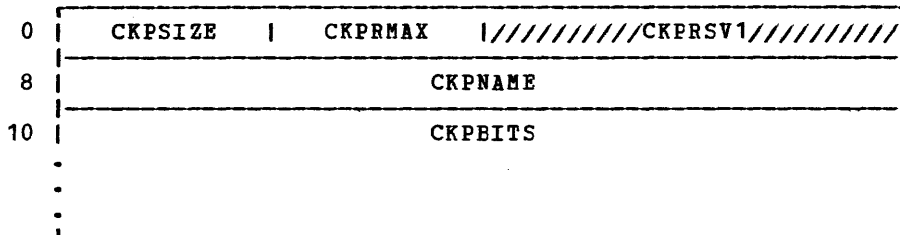
Hexadecimal Displacement	Field Name	Field Description, Contents, Meaning		
	CHXBLOK	X-side channel adapter block		
0	CHXOTHR DS 2F			VMELCK address of Y-side adapter user
8	CHXFLAG DS 1X	X*1		Internal processing flags
	<u>Bits defined in CHXFLAG</u>			
	CHBMNOP EQU X'80'			Modified NOP command issued (also in CMDT)
	CHBM370 EQU X'40'			CTCA operating in System/370 mode
	CHBATTN EQU X'20'			Attention pending from Y-side
	CHBREST EQU X'10'			CTCA has been reset X-side and Y-side
	CHBEOFL EQU X'08'			Force EOF to next READ instruction
	CHBHIO EQU X'04'			Halt I/O or halt device issued
	CHBWAIT EQU X'02'			CPEXELCK available for channel reconnect
	CHBCENT EQU X'01'			Channel end has been preserved on SIO
9	CHXCMDB DS 1X	X*2		Active CCW command byte buffer
A	CHXCMDT DS 1X	X*3		Active CCW command type (RD, WR, etc.)
	<u>Bits defined in CHXCMDT</u>			
	CHBCTNL EQU X'40'			Control, other than NOP
	CHBRDBK EQU X'20'			Read backward
	CHBWEof EQU X'10'			Write EOF
	CHBSCMD EQU X'08'			Sense command byte
	CHBSADS EQU X'04'			Sense adapter status
	CHBREAD EQU X'02'			Read
	CHBWRIT EQU X'01'			Write

Hexadecimal Displacement	Field Name				Field Description, Contents, Meaning
B	CHXPKEY	DS	1X	Y*4	Virtual CAW protection key
C		DS	4X		"Other" adapter control flags
10	CHXNCCW	DS	2F		Next CCW fetch address (real)
18	CHXRCNT	DS	2F		Remaining CCW data count
20	CHXSTAT	DS	1H		Device status accumulation field
22	CHXYADD	DS	1H		Virtual address of Y-side adapter
24		DS	2H		"Other" adapter control flags
28	CHXIDAW	DS	2F		Active indirect data list word
30	CHXCNCT	DS	2F		CPEXELOK for channel reconnect
38	CHXDATN	DS	2F		IOELCK address for deferred I/O interrupt
	CHBSIZE	EQU	(*CHXBLOK)/8		Total block size in doublewords (X'08')
	CHYBLOK				Y-side channel adapter block
0	CHYOTHR	DS	2F		VMELCK address of X-side adapter user
8	CHYFLAG	DS	1X	Y*1	Internal processing flags
	<u>Bits defined in CHYFLAG</u>				
	CHBMNOP	EQU	X'80'		Modified NOP command issued (also in CMDT)
	CHBM370	EQU	X'40'		CTCA operating in System/370 mode
	CHBATTN	EQU	X'20'		Attention pending from X-side
	CHBREST	EQU	X'10'		CTCA has been reset X-side and Y-side
	CHBEOF	EQU	X'08'		Force EOF to next READ instruction
	CHBHIO	EQU	X'04'		Halt I/O or halt device issued
	CHBWAIT	EQU	X'02'		CPEXELOK available for channel reconnect
	CHBEENT	EQU	X'01'		Channel end has been preserved on SIO
9	CHYCMDB	DS	1X	Y*2	Active CCW command byte buffer
A	CHYCMDT	DS	1X	Y*3	Active CCW command type (RD, WR, etc.)
	<u>Bits defined in CHYCMDT</u>				
	CHBCNTL	EQU	X'40'		Control, other than NOP
	CHBRDBK	EQU	X'20'		Read backward
	CHBWEOF	EQU	X'10'		Write EOF
	CHBSCMD	EQU	X'08'		Sense command byte
	CHBSADS	EQU	X'04'		Sense adapter status
	CHBREAD	EQU	X'02'		Read
	CHBWRT	EQU	X'01'		Write
B	CHYPKEY	DS	1X	Y*4	Virtual CAW protection key
C		DS	4X		"Other" adapter control flags
10	CHYNCCW	DS	2F		Next CCW fetch address (real)
18	CHYRCNT	DS	2F		Remaining CCW data count
20	CHYSTAT	DS	1H		Device status accumulation field
22	CHYXADD	DS	1H		Virtual address of X-side adapter
24		DS	2H		"Other" adapter control flags
28	CHYIDAW	DS	2F		Active indirect data list word
30	CHYCNCT	DS	2F		CPEXELOK for channel reconnect
38	CHYDATN	DS	2F		IOELCK address for deferred I/O interrupt

CKPBLOK

CKPBLOK: TELECOMMUNICATIONS CHECKPOINT BLOCK

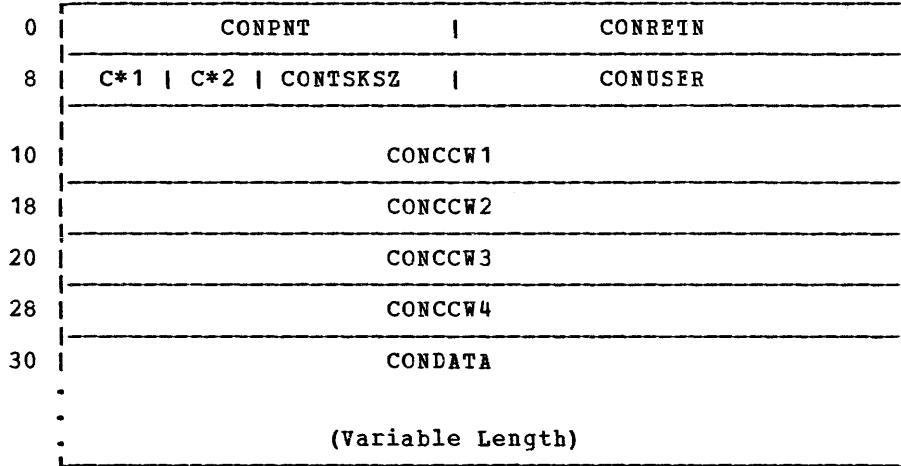
CKPBLOK provides checkpoint information needed for VM/370 warm start recovery for 3704/3705 Communication Controllers and enabled lines and resources. The RDEVCKPT field of the RDEVBLK points to CKPBLOK.



Hexadecimal Displacement	Field Name	Field Description, Contents, Meaning
0	CKPSIZE DS 1H	Size of checkpoint block in doublewords
2	CKPRMAX DS 1H	Number of resources checkpointed
4	CKPRSV1 DS 1F	Reserved for IBM use
8	CKPNAME DS CL8	370x control program reference name
10	CKPBITS DS 0D	Bit map of enabled lines or resources
	CKPBKSZ EQU (CKPBITS-CKPBLOK)/8	Header size in doublewcrds

CONTASK: CONSOLE I/O PACKAGE

CONTASK contains data and control information pertinent to the control and communication between virtual and real terminal console tasks and command streams. The RDEVCON field of the RDEVBLK and the NICQPNT field of the NICELK point to CONTASK.



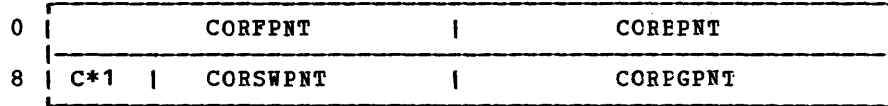
Hexadecimal Displacement	Field Name	Field Description, Contents, Meaning		
0	CONPNT DS 1F	Pointer to next CONTASK		
4	CONRETN DS 1F	Pointer to SAVEAREA for return		
8	CONSTAT DS 1X	C*1	CONTASK status control flags	
	<u>Bits defined in CONSTAT</u>			
	CONOUTPT EQU X'80'	Generate CONTASK output		
	CONRESP EQU X'40'	Response expected from this CCNTASK		
	CONACTV EQU X'20'	CONTASK is active on real device		
	CONCNTL EQU X'10'	This is a control CONTASK only		
	CONESCP EQU X'08'	CONTASK contains device dependent data		
	CONRTRY EQU X'04'	Retry operation in progress		
	CONSPLT EQU X'02'	Output data being split via RDEVLEN		
	CONSYNC EQU X'01'	CONTASK for synchronization only		
9	CONPARM DS 1X	C*2	DMKQCN parameter flags (see "Appendix A. CP and RSCS Equate Symbols")	
A	CONTSKSZ DS 1H	CONTASK size in doublewords		
C	CONUSER DS 1F	Address of VMBLK for destination user		
10	CONCCW1 DS 1D	First console I/O CCW		
	ORG CONCCW1			
10	CONADDR DS 1F	CCW data address		
14	CONFLAG DS 1X	CCW flag bits		
15	CONDWC DS 1X	DIAGNOSE write control		
16	CONCNT DS 1H	CCW byte count		
	ORG CONADDR			
10	CONCOMND DS 1X	CCW command code		
18	CONCCW2 DS 1D	Second console I/O CCW		
20	CONCCW3 DS 1D	Third console I/O CCW		
28	CONCCW4 DS 1D	Fourth console I/O CCW		
30	CONDATA DS 0C	Output data area (variable length)		
	CONTSIZE EQU (*-CONTASK)/8	CONTASK size in doublewords		

CONTASK

Hexadecimal Displacement	Field Name	Field Description, Contents, Meaning		
<u>Bits redefined in CONCCW for 370x Network Control Program</u>				
		ORG	CONCCW3+2	
2A	CONSRID	DS	1H	Source identifier
2C	CONDEST	DS	1H	Destination resource ID
2E	CONRTAG	DS	1H	Request tag for this CONTASK
30	CONSYSR	DS	1X	370x system response byte
31	CONEXTR	DS	1X	370x extended response byte
32	CONTCMD	DS	1H	Bisynchronous terminal command modifier
34	CONFUNC	DS	1X	Basic device function control flags
35	CONDFLG	DS	1X	Basic device data control flags
36	CONDCNT	DS	1H	Text data length
<u>Bits redefined for 3270 Remote Support</u>				
		ORG	CONCCW4	
30	CONLABEL	DS	1X	Return index value
31	CONSTX	DS	1X	Start text character
33	CONESC	DS	1X	Escape character
33	CONCMD	DS	1X	Command code for remote station
34	CONWCC	DS	1X	Write control character
35	CONSBA	DS	1X	Start buffer address
36		DS	1H	Buffer address

CORTABLE: STORAGE ALLOCATION TABLE

CORTABLE maintains the status and ownership of each page frame of real storage for use by page management routines. The ACORETBL field of the PSA points to CORTAELE.

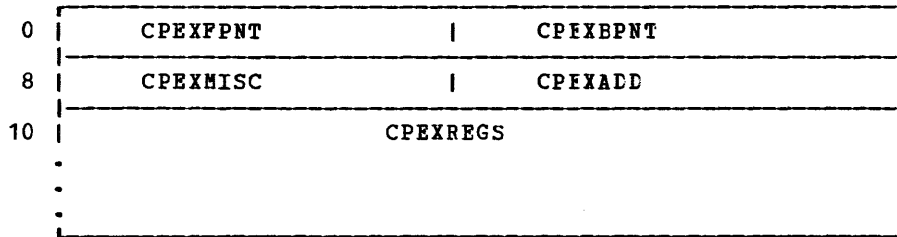


Hexadecimal Displacement	Field Name	Field Description, Contents, Meaning		
0	CORFPNT DS 1F	Pointer to next CORTABLE entry in queue		
<u>Entry definition for Page Ownership</u>				
	CORVM DS 1F	ORG CORFPNT	Pointer to VMBLOK of page owner	
4	COREPNT DS 1F	Pointer to previous CORTABLE entry in queue		
8	CORSWPNT DS 1F	Pointer to SWPTABLE for page		
C	CORPGPNT DS 1F	Pointer to FAGTABLE for page		
8	CORFLAG DS 1X	ORG CORSWPNT	C*1 CORTAELE entry status flags	
<u>Bits defined in CORFLAG</u>				
	CORIOLOCK EQU X'80'	Page locked for I/C or CORLCNT is greater than 0		
	CORCFLCK EQU X'40'	Page locked by console function		
	CORFLUSH EQU X'20'	Page is in FLUSH list		
	CORFREE EQU X'10'	Page is in FREE list		
	CORSHARE EQU X'08'	Page is shared		
	CORRSV EQU X'04'	Page is reserved		
	CORCP EQU X'02'	Page belongs to the control program		
	CORDISA EQU X'01'	Page disabled, not available		
<u>Entry Definition if Page Is Locked</u>				
4	CORLCNT DS 1F	ORG COREPNT	Page lock count for CORIOLOCK	
<u>Entry Definition if Page Is in Transit</u>				
8	CORCODE DS 1X	ORG CORFLAG	C*1 DASD operation code for DMKPAGE	

CPEXBLOK

CPEXBLOK: CP EXECUTE BLOCK

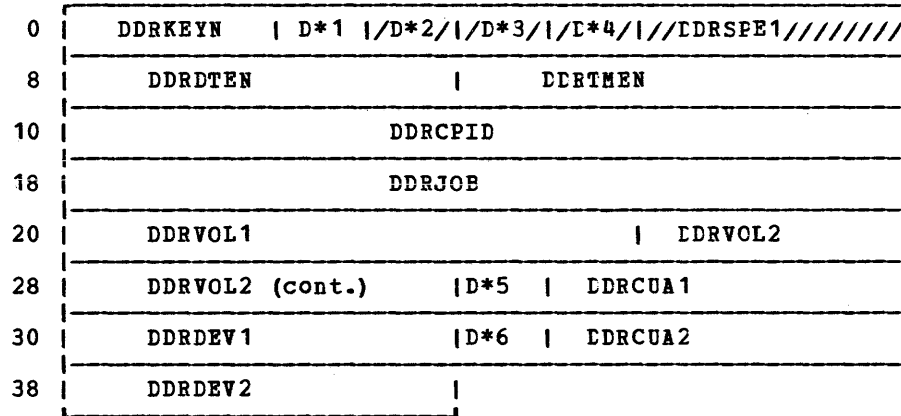
CPEXBLOK maintains register values and addressing information (module address or entry point address) to handle a delayed transfer of control. Stacked CPEXBLOCKS are queued off DMKDSRQ.



Hexadecimal Displacement	Field Name				Field Description, Contents, Meaning
0	CPEXFPNT DS	1F			Pointer to next CPEXBLOK
4	CPEXPNT DS	1F			Pointer to previous CPEXBLOK
8	CPEXMISC DS	1F			Use varies with stacker
<u>Definition for DMKSTK and DMKDSP</u>					
	ORG CPEXMISC				
	CPEXTYPE DS	1X			Type of block on CPEXBLOK chain
<u>Bits defined in CPEXTYPE</u>					
	CPEXDEFR EQU	X'80'			Deferred interrupt request
	CPEXPRIO EQU	X'40'			CPEXBLOK with priority
	CPEXLPSW EQU	X'20'			Load PSW to go to execution address
	DS	1X			Reserved for IBM use
	CPEXPROC DS	1H			Address of processor related to block
C	CPEXADD DS	1F			Return address
10	CPEXREGS DS	16F			Execute registers
<u>For CPEXREGS Area</u>					
	ORG CPEXREGS				
10	CPEXR0 DS	1F			Registers 0 through 15
14	CPEXR1 DS	1F			
18	CPEXR2 DS	1F			
1C	CPEXR3 DS	1F			
20	CPEXR4 DS	1F			
24	CPEXR5 DS	1F			
28	CPEXR6 DS	1F			
2C	CPEXR7 DS	1F			
30	CPEXR8 DS	1F			
34	CPEXR9 DS	1F			
38	CPEXR10 DS	1F			
3C	CPEXR11 DS	1F			
40	CPEXR12 DS	1F			
44	CPEXR13 DS	1F			
48	CPEXR14 DS	1F			
4C	CPEXR15 DS	1F			
	CPEXSIZE EQU	(*-CPEXBLOK)/8			Size in doublewords (X'0A')

DDRREC: RECONFIGURATION MACRO

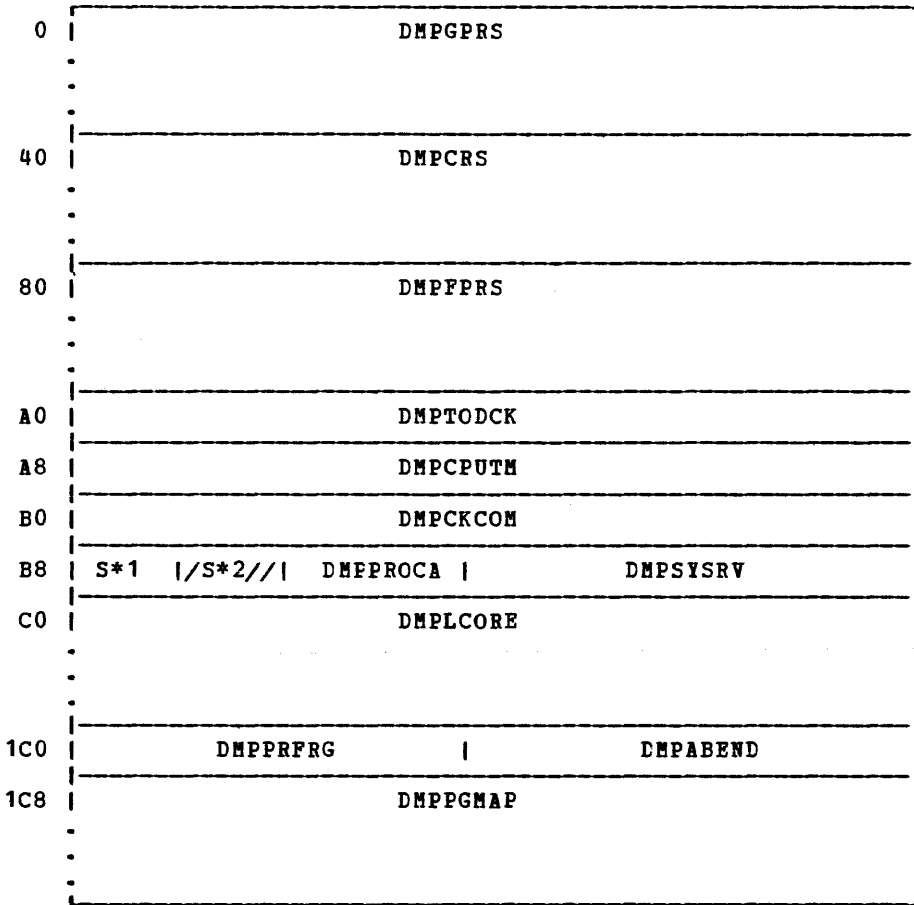
DDRREC is used in the SVC 76-initiated error recording process for type 60 DASD dump restore (DDR) dynamic device reallocation records. The reallocation records contain the replacement of the virtual "FROM" and "TO" control unit addresses (CUA) by the real addresses of the real DASD devices.



Hexadecimal Displacement	Field Name	Field Description, Contents, Meaning		
0	DDRKEYN DS 1H			Type and operating system
2	DDRSWS1 DD 1C	D*1		Switch byte
3	DDRSWS2 DS 1C	D*2		Reserved for IBM use
4	DDRSWS3 DS 1C	D*3		Reserved for IBM use
5	DDRRECNT DS 1C	D*4		Reserved for IBM use
6	DDRSPE1 DS 1H			Reserved for IBM use
8	DDRTEN DS 1F			Date
C	DDRTMEN DS 1F			Time
10	DDRCPID DS 2F			Processor identification and model number
<u>Device Dependent Data</u>				
18	DDRJOB DS 8X			Job using FROM device
20	DDRVOL1 DS 6X			Volume serial FROM device
26	DDRVOL2 DS 6X			Volume serial TO device
2C	DDRDEVP1 DS 1X	D*5		Device identification of FROM DASD
2D	DDRCUA1 DS 3X			Primary CUA of FROM device
30	DDRDEV1 DS 4X			Device type FROM device
34	DDRDEVP2 DS 1X	D*6		Device identification TO DASD
35	DDRCUA2 DS 3X			Primary CUA of TO device
38	DDRDEV2 DS 4X			Device type of TO device
	DDRSIZE EQU (*-DDRREC)			DDR record size

DMPINREC: DUMP FILE INFORMATION RECORD

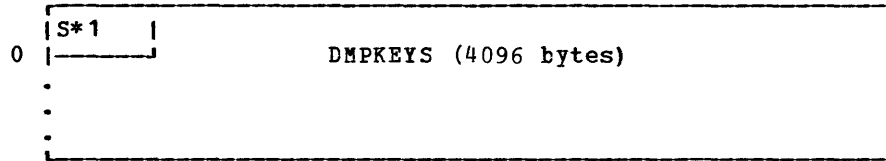
DMPINREC retains vital system register and storage location values necessary for the CPDUMP file. See also DMPKYREC.



Hexadecimal Displacement	Field Name	Field Description, Contents, Meaning		
0	DMPGPRS DS 16F	16 general registers		
40	DMPCRS DS 16F	16 control registers		
80	DMPFPRS DS 4D	Four floating-point registers (if floating-point feature is installed on machine)		
A0	DMPTODCK DS 1D	Time-of-day clock		
A8	DMPCPUTM DS 1D	Processor timer		
B0	DMPCKCOM DS 1D	Time-of-day clock comparator		
B8	DMPFLAG DS 1X	S*1	Flag byte	
<u>Bits defined in DMPFLAG</u>				
	HALFPAGE EQU X'80'	When on, last record in DUMP file is 2K		
B9	DMPRSV1 DS 1X	S*2	Reserved for IBM use	
BA	DMPPROCA DS 1H	Abending processor address		
BC	DMPSYSRV DS 1F	System generated storage size		
C0	DMPLCORE DS 256X	Absolute storage locations 0 through 255		
1C0	DMPPRFRG DS 1F	Prefix register		
1C4	DMPABEND DS 1F	Abend code for failing processor		
1C8	DMPPGMAP DS 4096B	Bit map indicating which pages appear in the DUMP file (each bit represents a 4K block)		

DMPKYREC: DUMP FILE KEY STORAGE RECORD

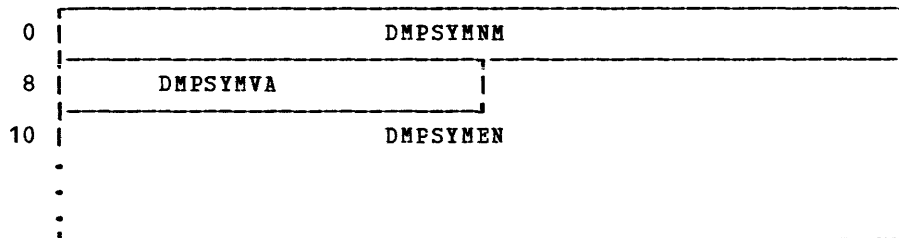
DMPKYREC contains the storage keys of each 2K block of main storage at the time of SVC 0 or a PSW restart condition. DMPKYREC and DMPINREC are used for debugging operations.



Hexadecimal Displacement	Field Name	Field Description, Contents, Meaning
0	DMPKEYS DS 4096X	Main storage keys
0	DMPKEY DS 1X S*1	Storage key for each 2K block

DMPTBREC: DUMP FILE SYMBOL TABLE RECORD

DMPTBREC is a listing of all entry points in the system and their locations.



Hexadecimal Displacement	Field Name	Field Description, Contents, Meaning
C	DMPSYMEN DS 341XL12	Symbol table entries
0	DMPSYMNM DS CL8	CSECT or entry point name
8	DMPSYMVA DS A	Location in main storage of this symbol

ECBLOK

ECBLOK: EXTENSION TO VMBLOK FOR VIRTUAL MACHINE WITH RELOCATE

ECBLOK provides an extension to the VMBLOK for virtual machine operation in System/370 extended control mode. The VMEEXT field of the VMBLOK points to ECECLK.

0	EXTCR0		EXTCR1
8	EXTCR2		EXTCR3
10	EXTCR4		EXTCR5
18	EXTCR6		EXTCR7
20	EXTCR8		EXTCR9
28	EXTCR10		EXTCR11
30	EXTCR12		EXTCR13
38	EXTCR14		EXTCR15
40	EXTSHCR0		EXTSHCR1
48	EXTSHLEN EXTVSEGS		EXTSTOLD
50	EXTSHSEG		EXTSEGLN EXTARCH
58	EXTPERAD		EXTPERCD EXTCOPY
60	EXTCPTMR		
68	EXTCPTRQ		EXTCCTRQ

Hexadecimal Displacement	Field Name			Field Description, Contents, Meaning
0	EXTCR0	DS	1F	Virtual control register 0; architecture controls
4	EXTCR1	DS	1F	Virtual control register 1; segment table pointer
8	EXTCR2	DS	1F	Virtual control registers 2 through 15
C	EXTCR3	DS	1F	
10	EXTCR4	DS	1F	
14	EXTCR5	DS	1F	
18	EXTCR6	DS	1F	
1C	EXTCR7	DS	1F	
20	EXTCR8	DS	1F	
24	EXTCR9	DS	1F	
28	EXTCR10	DS	1F	
2C	EXTCR11	DS	1F	
30	EXTCR12	DS	1F	
34	EXTCR13	DS	1F	
38	EXTCR14	DS	1F	
3C	EXTCR15	DS	1F	

Hexadecimal Displacement	Field Name			Field Description, Contents, Meaning
40	EXTSHCRO	DS	1F	Shadow control register 0
44	EXTSHCR1	DS	1F	Shadow control register 1
48	EXTSHLEN	DS	1H	Length of shadow SEGTABLE in bytes
4A	EXTVSEGS	DS	1H	Length of virtual SEGTABLE in bytes
4C	EXTSTOLD	DS	1F	Control register 1 value corresponding to tables
50	EXTSHSEG	DS	1F	Real address of shadow SEGTABLE
54	EXTSEGLN	DS	1H	Length of shadow SEGTABLE in doublewords
56	EXTARCH	DS	1H	Architecture control index
58	EXTPERAD	DS	1F	Address of instruction PER interrupt
5C	EXTPERCD	DS	1H	PER code to be reflected
5E	EXTCOPY	DS	1H	Length code from active SEGTABLE entry
60	EXTCPTMR	DS	1D	Virtual processor timer
68	EXTCPTRQ	DS	1F	Address of TRQBLOK for processor timer
6C	EXTCCTRQ	DS	1F	Address of TRQBLOK for clock comparator
	EXTSIZE	EQU	(*-ECBLOK)/8	ECBLOK size in doublewords (X'0E')

IOBLCK: I/O TASK CONTROL BLOCK

IOBLCK contains information required to perform I/O operations. The I/O request initiator for the I/O operation is either a CP-initiated or virtual machine-initiated event. There are five pointers to the IOBLCK: RCHFICB field of the RCHBLCK, RCHFICE field of the RCUBLOK, RDEVAIOB field of the RCFVBLOK, VDEVFIOB field of the VDEVBLOK, RDEVFIOB field of the RDEVBLOK.

0	IOBRADD	I*1	I*2	IOBLINK
8	IOBFPNT			IOBEPNT
10	IOBCYL	IOBVADD		IOBMISC
18	IOBUSER			IOBIRA
20	IOBCAW			IOBRCAW
28	IOBCSW			
30	IOBIOER			IOBMISC2
38	I*3	I*4	//IOBRV2// //////////IOERSV3//////////	

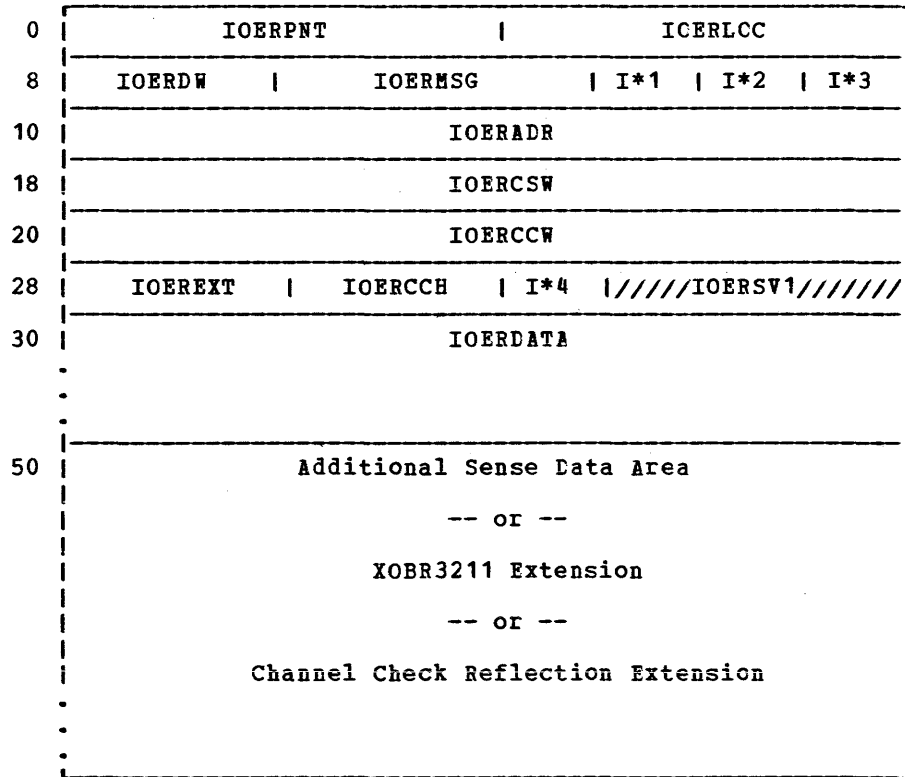
Hexadecimal Displacement	Field Name			Field Description, Contents, Meaning
0	IOBRADD	DS	1H	Real device address for SIC
2	IOBFLAG	DS	1X	I*1 IOBLCK flags
	<u>Bits defined in IOBFLAG</u>			
	IOBCP	EQU	X'80'	CP-generated I/O operation
	IOBRSTRT	EQU	X'40'	Restarted operation - IOBRCAW
	IOBSPLT	EQU	X'20'	DASD - CP split seek operation
	IOBPAG	EQU	X'10'	IOBLCK created for paging I/O
	IOBRELCU	EQU	X'08'	Control unit released at initiation
	IOBERP	EQU	X'04'	I/O task is under control of ERP
	IOBRES	EQU	X'02'	I/O task has been reset
	IOBHVC	EQU	X'01'	I/O initiated via DIAGNOSE instruction
3	IOBSTAT	DS	1X	I*2 IOBLCK status
	<u>Bits defined in IOBSTAT</u>			
	IOBFATAL	EQU	X'80'	Unrecoverable error in this I/O operation
	IOBFLT	EQU	X'40'	IOBLCK queued pending completion of a MSS cylinder fault
	IOBPATHF	EQU	X'20'	Path is fixed, use IOBRADD value
	IOBMINI	EQU	X'08'	This is a mini-IOBLCK
	IOBALTSK	EQU	X'04'	DASD channel program has seek to alternate track
	IOBCC3	EQU	X'03'	Processing CC 3, not available
	IOBCC2	EQU	X'02'	Processing CC 2, channel busy
	IOBCC1	EQU	X'01'	Processing CC 1, CSW stored
	IOBCC0	EQU	X'00'	Processing I/O interrupt

IOBLOK

Hexadecimal Displacement	Field Name			Field Description, Contents, Meaning
4	IOBLINK	DS	1F	Pointer for multipath IOELCK chain
8	IOBFPNT	DS	1F	Pointer to next IOBLOK in queue
C	IOBBPNT	DS	1F	Pointer to previous IOBLOK in queue
	IOBMSIZE	EQU	(*IOBLOK)/8	Multiple path IOBLOK size in doublewrds (X'02')
10	IOBCYL	DS	1H	DASD - seek cylinder for this IOBLOK
12	IOBVADD	DS	1H	Virtual device address
14	IOBMISC	DS	1F	Use varies according to caller
18	IOBUSER	DS	1F	Pointer to VMBLCK of user
1C	IOBIRA	DS	1F	IOELCK interrupt return address
20	IOBCAW	DS	1F	Pointer to CCW chain
24	IOBRCAW	DS	1F	Pointer to restart CCW chain
28	IOBCSW	DS	1D	Real CSW for I/O operation
30	IOBIOER	DS	1F	Pointer to IOERBLOK with sense byte
34	IOBMISC2	DS	1F	Use varies according to caller
38	IOBSPEC	DS	1X	I*3 IOELCK special requests flag
	<u>Bits defined in IOBSPEC</u>			
	IOBTIO	EQU	X'80'	IOELCK request for a TIO
	IOBHIO	EQU	X'40'	IOELCK request for a HIO
	IOBSIOF	EQU	X'20'	Virtual SIO fast release
	IOBIMSTK	EQU	X'10'	Shut down SDR function
	IOBUNSL	EQU	X'08'	IOELCK resulting from unsolicited interrupt
	IOBCOPY	EQU	X'04'	I/O block associated with a COPY request
	IOBSENS	EQU	X'02'	Sense operation for COPY request
	IOBTRPND	EQU	X'01'	Virtual trace pending on this I/O block
39	IOBSPEC2	DS	1X	I*4 IOELCK special requests flag second byte
	<u>Bits defined in IOBSPEC2</u>			
	IOBWRAP	EQU	X'80'	Input/output task for AUTOPLL wrap list
	IOBCLN	EQU	X'40'	VDEVELOK locked when CCW gct control
	IOBUNREL	EQU	X'20'	Input/output task contains release, DMKUNT must process
	IOBUC	EQU	X'10'	Unit check status
	IOBSNSIO	EQU	X'08'	Normal sense operation in progress
	IOBRELE	EQU	X'04'	Channel program contains CF release
3A	IOBRV2	DS	1H	Reserved for IBM use
3C	IOBRV3	DS	1F	Reserved for IBM use
	IOBSIZE	EQU	(*IOBLOK)/8	IOELCK size in doublewords (X'08')
	<u>For CP IOBLOKs</u>			
12	IOBRCNT	DS	1H	Retry count

IOERBLOK: I/O ERROR INFORMATION BLOCK

IOERBLOK contains information related to I/O and channel errors. This entails error retry, operator message information, and SDR (Statistical Data Recording) IOERBLOK related to I/O equipment. There are three pointers to the IOERBLOK: RDEVIOER field of the RDEVBLOK, VDEVIOER field of the VDEVBLOK, IOPIOER field of the ICBLCK.



Hexadecimal Displacement	Field Name	Field Description, Contents, Meaning		
0	IOERPNT DS 1F	Pointer to next IOERBLOK		
4	IOERLOC DS 1F	Address of CCWs used in recovery		
8	IOERDW DS 1H	Size (in doublewords) of storage needed to construct CCWs		
A	IOERMSG DS XL3	Communications with error recording processor and message writer		
A	IOERNUM DS 1X	ORG IOERMSG	Breakdown for DMKMSW	
B	IOERIND3 DS 1X	Message number for message routine		
		Indicators for message routine		
	<u>Bits defined in IOERIND3</u>			
	IOERIGN EQU X'80'	Allow IGNORE response		
	IOERRETRY EQU X'40'	Allow RETRY response		
	IOERCAN EQU X'20'	Allow CANCEL response		
	IOEREC EQU X'10'	Error occurred during recovery action		
	IOERDASD EQU X'08'	Home address is present		
	IOERDEC EQU X'04'	Operator decision is necessary		
	IOERINFO EQU X'02'	Informational message		
	IOERACT EQU X'01'	Operator action is required		

IOERBLOK

Hexadecimal Displacement	Field Name			Field Description, Contents, Meaning
C	IOERIND4	DS	1X	Indicators for message writer
	<u>Bits defined in IOERIND4</u>			
	IOERIGNR	EQU	X'80'	Operator responded "ignore"
	IOERSTRT	EQU	X'40'	Operator responded "retry"
	IOERCNCL	EQU	X'20'	Operator responded "cancel"
D	IOERFLG1	DS	1X	I*1 Flag byte for error recovery routines
	<u>Bits defined in IOERFLG1</u>			
	IOERPEND	EQU	X'80'	Pending device end interrupt from interrupt request
	IOERCLN	EQU	X'40'	Tape cleaning in progress
	IOERERP	EQU	X'40'	Spooling - error routine in control
	IOERFSR	EQU	X'20'	Forward space record being executed
	IOERDEPD	EQU	X'20'	Spooling - waiting for device end
	IOERBSR	EQU	X'10'	Backspace record being executed
	IOERDERD	EQU	X'10'	Spooling - device end received
	IOERERG	EQU	X'08'	Erase gap command in progress
	IOERXERP	EQU	X'08'	Spooling - error routine getting OBR data
	IOERORA	EQU	X'04'	Opposite recovery action in progress
	IOERSUPP	EQU	X'02'	CCW has suppress data transfer bit on
	IOERVLD	EQU	X'01'	Read opposite recovery successful
E	IOERFLG2	DS	1X	I*2 Second flag byte for error recovery program use
	<u>Bits defined in IOERFLG2</u>			
	IOERSTAT	EQU	X'80'	Statistical data being unloaded
	IOERHA	EQU	X'40'	DASD home address being read
	IOERCAL	EQU	X'20'	Standalone recalibrate being executed
	IOERECF	EQU	X'10'	Error correction function
	IOERRBK	EQU	X'10'	Read backward command
	IOERREW	EQU	X'08'	Tape rewind being executed
	IOERCYLR	EQU	X'04'	Cylinder (in sense byte) has been relocated
	IOERMSW	EQU	X'04'	Message writer is active
	IOERCEND	EQU	X'02'	Intensive recording mode
	IOERVOL1	EQU	X'01'	DASD valid being read
F	IOERWRK	DS	1X	I*3 Miscellaneous work byte
10	IOERADR	DS	1D	Home address for DASD devices
18	IOERCSW	DS	1D	CSW associated with error
20	IOERCCW	DS	1D	Sense CCW used to sense the real device
		ORG	IOERCCW	Breakdown of sense CCW field
20	IOERVSR	DS	CL6	Volume serial number for statistical data
26	IOERLEN	DS	1H	Number of sense bytes present
2C	IOERFLG3	DS	1X	I*4 Third flag byte for SDR use
	<u>Bits defined in IOERFLG3</u>			
	IOERREAD	EQU	X'80'	SDR READ operation flag
	IOERALTR	EQU	X'40'	Alternate track retry is in progress
	IOERRDRO	EQU	X'20'	Read home address, read only is in progress
2D	IOERSV1	DS	XL3	Reserved for IBM use
30	IOERSNSZ	EQU	32X	Number of sense bytes supported
30	IOERDATA	DS	(IERSNSZ/8) D	Sense bytes associated with error
		ORG	IOERDATA	Breakdown for channel check handler
30	IOERCSW	DS	1F	FCSW information from channel logout
34	IOERCHAN	DS	1X	Channel type flag
	IOEREND	DS	0F	Label for end of IOERBLOK

Hexadecimal Displacement	Field Name	Field Description, Contents, Meaning
50	XOBR3211	Area where additional sense bytes may be appended to IOERBLCK. The format for one of these is described in "XOBR3211: Extended Outboard Recording Block" later in this section.

• Channel Check Reflection Extension

50	IOERCCRL		IOERCCRA
58	IOERCUID		IOERCMDL IOERMCEL
60	IOERUID		
68	IOERFADD		
.	.		
78	IOERCCCW		
80	IOERCCSW		
88	IOERZCSW		IOERDTYP
90	I*5 IOERCCUA		IOERMCUA IOERLOGL
98	IOERCLOG		
.	.		
.	.		

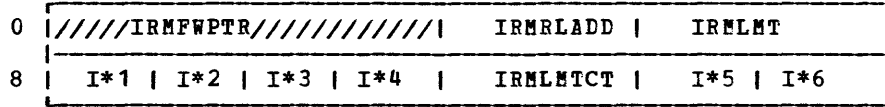
Hexadecimal Displacement	Field Name	Field Description, Contents, Meaning
50	IOERCEXT DS OD	Extension used only for channel check reflection
50	IOERCCRL DS F	Length of channel check record
54	IOERCCRA DS F	Address of channel check record
58	IOERCUID DS F	Processor identification
5C	IOERCMDL DS H	Processor model number (for example, 0158, 0168, etc.)
5E	IOERMCEL DS H	Maximum length of machine check extended logout area (model-dependent)
60	IOERUID DS D	User identification
68	IOERFADD DS 8H	Active input/output units
78	IOERCCCW DS D	Failing CCW
80	IOERCCSW DS D	Failing CSW
88	IOERZCSW DS F	Failing ECSW
8C	IOERDTYP DS F	CP device type
90	IOERCHID DS CL1 I*5	Channel identification
<u>Channels defined in IOERCHID</u>		
	IOER2860 EQU X'05'	Standalone selector (2860)
	IOER2870 EQU X'06'	Standalone multiplexer (2870)
	IOERB80 EQU X'07'	Standalone block multiplexer (2880)
	IOERS80 EQU X'08'	Selector channel (2880)
91	IOERCCUA DS CL3	Actual failing device address
94	IOERMCUA DS CL2	Address from processor location X'BA'
96	IOERLOGL DS CL2	Length of channel logout

IOERBLOK

Hexadecimal Displacement	Field Name			Field Description, Contents, Meaning
98	IOERCLOG	DS	F	Channel logout area
98	IOERLG80	ORG DSCL112	IOERCLOG	2880 channel (112 bytes)
98	IOERLG70	ORG DS	IOERCLOG CL24	2870 channel (24 bytes)
98	IOERLG60	ORG DS	IOERCLOG CL24	2860 channel (24 bytes)
98	IOERADDR	ORG DS	IOERCLOG F	Unit address stored by integrated channel
9C	IOERLG33	DS	CL640	3033 channel (640 bytes)
9C	IOERLG45	ORG DS	IOERLG33 CL96	145 integrated channel (96 bytes)
9C	IOERLG35	ORG DS	IOERLG33 CL24	135 integrated channel (24 bytes)
	IOERSIZE	EQU	(*-IOERBLOK)/8	IOERBLOK size in doublewords

IRMBLOK: INTENSIVE ERROR RECORDING MODE BLOCK

IRMBLOK provides the information necessary for the implementation of intensive recording mode via CP SET RECORD command. Intensive recording mode allows the recording of unit check errors from a specified device whose sense data matches the values selected.

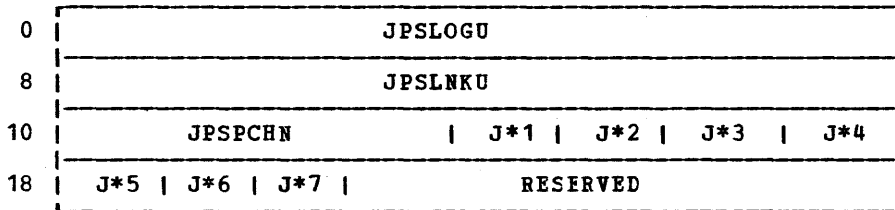


Hexadecimal Displacement	Field Name	Field Description, Contents, Meaning
0	IRMFWPTR DS 1F	Reserved for IBM use
4	IRMLADD DS 1H	Device address to be monitored
6	IRMLMT DS 1H	Limit count - record every <u>n</u> th error
8	IRMBYT1 DS 1X	I*1 First sense byte specified
9	IRMBIT1 DS 1X	I*2 Sense bit within first sense byte
A	IRMBYT2 DS 1X	I*3 Second sense byte specified
B	IRMBIT2 DS 1X	I*4 Sense bit within second sense byte
C	IRMLMTCT DS 1H	Summary count for limit detection
E	IRMMAXCT DS 1X	I*5 Count of recordings made for this request
F	IRMFLG DS 1X	I*6 Flag byte
<u>Bits defined in IRMFLG</u>		
	IRMAND EQU X'80'	AND condition specified
	IRMOR EQU X'40'	OR condition specified
	IRMSIZE EQU (*-IRMBLOK)/8	IRMBLOK size in doublewords (X'02')

JPSCBLOK

| JPSCBLOK: JOURNALING AND PASSWORD SUPPRESSION CONTROL BLOCK

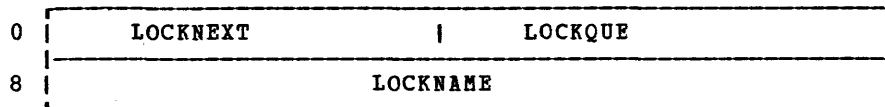
| The JPSCBLOK is a part of DMKSYS. It is referenced via the EXTERNAL symbol DMKSYSJR. It is used to control the LOGON/AUTOLOG/LINK Journaling and Password-on-the-command-line Suppression functions.



Hexadecimal Displacement	Field Name			Field Description, Contents, Meaning
0	JPSLOGU	DS	CL8	Userid for LOGON/AUTOLOG threshold messages
8	JPSLNKU	DS	CL8	Userid for LINK threshold messages
10	JPSPCHN	DS	1F	Anchor for PWDIBLOKs
14	JPSFLAGS	DS	1X	J*1 Control flags
<u>Bits defined in JPSFLG</u>				
	LOGONJRL	EQU	X'80'	LOGON/AUTOLOG journaling ON
	LINKJRL	EQU	X'40'	Successful LINK journaling ON
	LINKJRL1	EQU	X'20'	Invalid password LINK journaling ON
	JRLSQOK	EQU	X'10'	SET/QUERY JCURNAL enabled
	MASKLOG	EQU	X'08'	Suppress LOGON password
	MASKLINK	EQU	X'04'	Suppress LINK password
15	JPSLOGAR	DS	1X	J*2 LOGON/AUTOLOG accounting record threshold messages
16	JPSLOGMS	DS	1X	J*3 LOGON/AUTOLOG message threshold
17	JPSLOGDS	DS	1X	J*4 LOGON/AUTOLOG disable threshold
18	JPSLNKAR	DS	1X	J*5 LINK accounting record threshold
19	JPSLNKMS	DS	1X	J*6 LINK message threshold
1A	JPSLNKDS	DS	1X	J*7 LINK disable threshold
1B		DS	XL5	Reserved for IBM use

LOCKBLOK: USERID LOCK CONTROL BLOCK

LOCKBLOK is used to synchronize execution for sections of nonreenterable code. Locked users are returned to the CPEXBLOK queue when the function being executed completes or no longer requires nonreenterable resources. LOCKBLOKs are queued off DMKSYSLE.

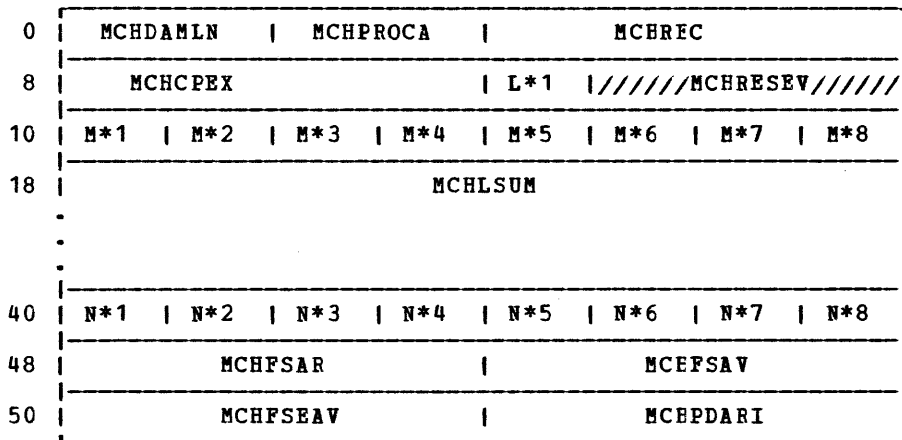


Hexadecimal Displacement	Field Name			Field Description, Contents, Meaning
<hr/>				
0	LOCKNEXT DS	1F		Pointer to the next lock control block
4	LOCKQUE DS	1F		Pointer to CPEXBLOK queue
8	LOCKNAME DS	1D		The name being locked
	LOCKSIZE EQU	(*-LOCKBLOK)/8		LOCKBLOK size in doublewords (X'02')

MCHAREA

MCHAREA: MACHINE CHECK SAVE AREA

MCHAREA provides CP with statistical data that relates to malfunctions of the real processor, to its buffers, to processor storage for damage assessment, and to the recovery of VM/370.



Hexadecimal Displacement	Field Name		Field Description, Contents, Meaning
0	MCHDAMGE DS	0H	Damage assessment area
0	MCHDAMLN DS	1H	Length of damage assessment area
2	MCHPROCA DS	XL2	Processor address
4	MCHREC DS	1F	Machine check record address
8	MCHCPEX DS	1F	Machine check CPEXBLCK address
C	MCHMODEL DS	1X	L*1 Model number for the machine

Bits defined in MCHMODEL

MODEL240 EQU	X'18'	ID number for the 240 machine
MODEL210 EQU	X'18'	ID number for the 210 machine
MODEL200 EQU	X'18'	ID number for the 200 machine
MOD3033 EQU	X'14'	ID number for the 3033 processor
MOD3032 EQU	X'14'	ID number for the 3032 processor
MOD3031 EQU	X'14'	ID number for the 3031 processor
MODEL168 EQU	X'10'	ID number for the 168 machine
MODEL165 EQU	X'10'	ID number for the 165 machine
MODEL158 EQU	X'0C'	ID number for the 158 machine
MODEL155 EQU	X'0C'	ID number for the 155 machine
MODEL148 EQU	X'08'	ID number for the 148 machine
MODEL145 EQU	X'08'	ID number for the 145 machine
MODEL138 EQU	X'04'	ID number for the 138 machine
MODEL135 EQU	X'04'	ID number for the 135 machine
NOMODEL EQU	X'00'	No support for machine

D	MCHRESEV DS	3X	Reserved for IBM use
10	MCHDAMFL DS	OBL8	Damage assessment data
10	MCHFLAG0 DS	1X	M*1 System status

Bits defined in MCHFLAG0

MCHOHDWR EQU	X'80'	Hardware recovery
MCHOSFTR EQU	X'40'	Software recovery
MCHOUSAD EQU	X'20'	User abnormally terminated
MCH1GERR EQU	X'10'	Channel inoperative
MCHOTERM EQU	X'08'	Operating system termination
MCHOQUIT EQU	X'04'	Quiet mode in effect

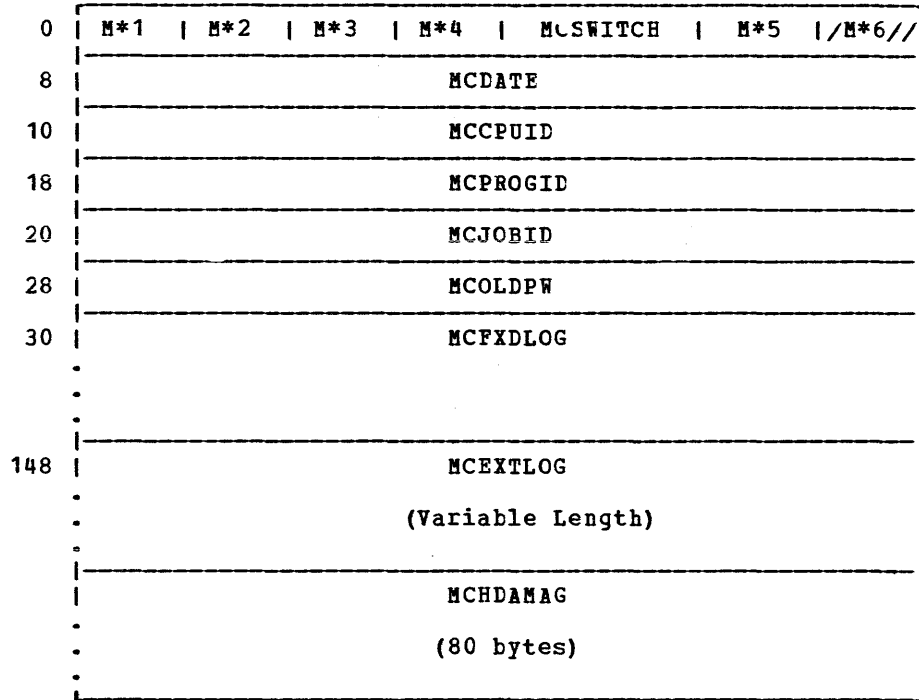
Hexadecimal Displacement	Field Name			Field Description, Contents, Meaning
11	MCHFLAG1 DS	1X	M*2	Damage area
	<u>Bits defined in MCHFLAG1</u>			
	MCH1MAIN EQU	X'80'		Main storage
	MCH1BUFF EQU	X'40'		Buffer
	MCH1COST EQU	X'20'		Control storage
	MCH1GERR EQU	X'10'		Indicates channel is inoperative if bit is on
	MCH1PROC EQU	X'08'		Processor
	MCH1TODC EQU	X'02'		Time-of-day clock
	MCH1SYSD EQU	X'01'		System damage
12	MCHFLAG2 DS	1X	M*3	Damage area (continued)
13	MCHFLAG3 DS	1X	M*4	Error type
	<u>Bits defined in MCHFLAG3</u>			
	MCH3INTE EQU	X'80'		Intermittent error
	MCH3SOLD EQU	X'40'		Solid error
	MCH3DATA EQU	X'20'		Data error
	MCH3PROT EQU	X'10'		Protect error
14	MCHFLAG4 DS	1X	M*5	RMS action data
	<u>Bits defined in MCHFLAG4</u>			
	MCH4TOLO EQU	X'80'		Time-out loop
	MCH4REPA EQU	X'40'		Repair
	MCH4STRE EQU	X'20'		Storage reconfiguration
	MCH4BURE EQU	X'10'		Buffer reconfiguration
15	MCHFLAG5 DS	1X	M*6	RMS information status
	<u>Bits defined in MCHFLAG5</u>			
	MCH5INLG EQU	X'80'		Invalid hardware logout
	MCH5INMC EQU	X'40'		Invalid machine check interrupt code
	MCH5IFSA EQU	X'20'		Invalid failing storage address
16	MCHFLAG6 DS	1X	M*7	RMS wait state suffix
17	MCHFLAG7 DS	1X	M*8	DMKMCH status
	<u>Bits defined in MCHFLAG7</u>			
	MCH7SMCR EQU	X'80'		Second machine check recursion
	MCH7VRTM EQU	X'40'		Terminate the virtual machine
	MCH7SYST EQU	X'20'		Terminate the system
	MCH7OPSW EQU	X'10'		Machine check old PSW in problem state
	MCH7VEQR EQU	X'08'		Terminate the virtual-real user
	MCH7SUP EQU	X'04'		DMKMCH under the global system lock
18	MCHLSUM DS	XL40		Summary
40	MCHPDAR DS	0BL8		
40	MCHPDAR0 DS	1X	N*1	Action taken
41	MCHPDAR1 DS	1X	N*2	Failure type
	<u>Bits defined in MCHPDAR1</u>			
	MCHP1SDE EQU	X'80'		Solid storage data error
	MCHP1IDE EQU	X'40'		Intermittent storage data error
	MCHP1SKE EQU	X'20'		Solid SPF key error
	MCHP1IKE EQU	X'10'		Intermittent SPF key error
	<u>Machine Check Communication Area Must Be Cleared Above This Line</u>			
42	MCHPDAR2 DS	1X	N*3	Operating system status
43	MCHPDAR3 DS	1X	N*4	Location of failure
44	MCHPDAR4 DS	1X	N*5	Location of failure
45	MCHPDAR5 DS	1X	N*6	Requested operator awareness
46	MCHPDAR6 DS	1X	N*7	Flag byte

MCHAREA

Hexadecimal Displacement	Field Name			Field Description, Contents, Meaning
	<u>Bits defined in MCHPDAR6</u>			
	MCHP6CBA	EQU	X'80'	Change bit active
47	MCHPDAR7	DS	1X N*8	Flag byte
	<u>Bits defined in MCHPDAR7</u>			
	MCH7STCK	EQU	X'80'	Interfaces for STACK routine
	MCH7GSTR	EQU	X'40'	Interfaces for GETMAIN routine
	MCH7PURG	EQU	X'20'	Interfaces for PURGE routine
	MCH7LOGO	EQU	X'10'	Interfaces for V=R logoff routine
	MCH7EXIT	EQU	X'08'	Interfaces for exit to CP
	MCH7RSRE	EQU	X'04'	Interfaces for RELEASE and RESET routines
	MCH7IOEM	EQU	X'02'	Interfaces for the recorder
48	MCHFSAR	DS	1F	Failing location real address
4C	MCHPSAV	DS	1F	Instruction address at failure
50	MCHPSEAV	DS	1F	End of the failing location
54	MCHPDARI	DS	1F	End of the failing virtual storage address
	MCHLEN1	EQU	*MCHDAMAGE	Length of damage assessment area
	MCHLEN	EQU	*MCHRESEV	Length of area to be cleared
	MCHFIX	EQU	280+48	Length of the fixed logout and the header record for machine check handler

MCRECORD: MACHINE CHECK HANDLER RECORD

MCRECORD provides the necessary extended logout information for error recording of processor and main storage.

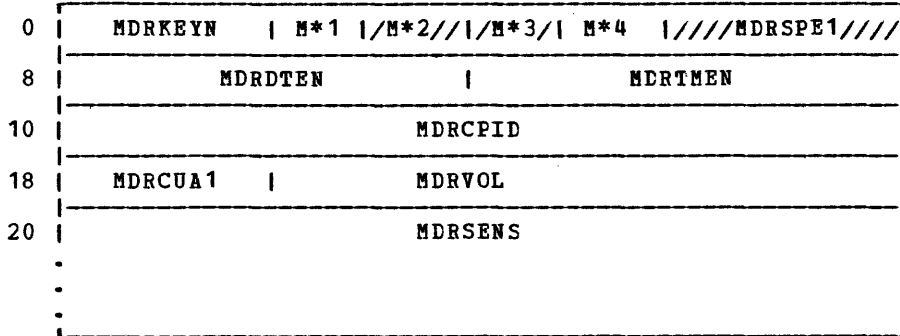


Hexadecimal Displacement	Field Name			Field Description, Contents, Meaning
0	MCREC	DS	0D	
0	MCRECTYP	DS	1X	M*1 Machine check record type
1	MCOPSYS	DS	1X	M*2 Operating system
2	MCSWONE	DS	1X	M*3 Record independent switch
3	MCSWTWO	DS	1X	M*4 Record dependent switch
4	MCSWITCH	DS	2X	Unused switches
6	MCRECCNT	DS	1X	M*5 Record count
7	MCRECCC	DS	1X	M*6 Reserved for IBM use
8	MCDATE	DS	XL8	Date and time
10	MCCPUID	DS	XL8	Processor identification and model number
18	MCPRGID	DS	XL8	Program identity
20	MCJOBID	DS	XL8	Job identity (unused)
28	MCOLDPW	DS	XL8	Machine check old PSW
30	MCFXDLOG	DS	35D	Machine check fixed logout
	FXDLGLH	EQU	(*MCFXDLOG)	Length of machine check fixed logout area
148	MCEXTLOG	EQU	*	Beginning of machine check extended logout (the extended logout length is variable length and machine dependent)
	MCHDAMAG	EQU	*	Damage assessment area (80 bytes)

MDRREC

MDRREC: MISCELLANEOUS DATA RECORDING RECORD

MDRREC retains information for the VM/370 error recording cylinders.



Hexadecimal Displacement	Field Name				Field Description, Contents, Meaning
<u>24-Byte Header</u>					
0	MDRKEYN	DS	1H		Type and operating system
2	MDRSWS1	DS	1C	M*1	Switch byte 0
3	MDRSWS2	DS	1C	M*2	Reserved for IBM use
4	MDRSWS3	DS	1C	M*3	Reserved for IBM use
5	MDRRECNT	DS	1C	M*4	Sequence (4 bits) and record (4 bits)
<u>Bits as used in MDRRECNT</u>					
					0 through 3 Used for sequence number
					4 through 7 Indicates the record total
6	MDRSPE1	DS	1H		Reserved for IBM use
8	MDRDTEN	DS	1F		Date
C	MDRTMEN	DS	1F		Time
10	MDRCPID	DS	2F		Processor identification and model number
<u>End of 24-Byte Header</u>					
<u>Device Dependent Data</u>					
18	MDRCUA1	DS	2X		Primary CUA of device
1A	MDRVOL	DS	6X		Volume serial number of device
20	MDRSENS	DS	24X		Sense byte data
	MDRSIZE	EQU	(*-MDRREC)		MDR record size
<u>For 3270 Remote Support</u>					
	ORG		MDRCUA1		
18		DS	2X		Line address
1A		DS	1X		Control unit address
1B		DS	1X		Device address
1C		DS	2X		Sense and status information
1E		DS	2X		Resource identification
20		DS	2X		Reserved for IBM use
<u>For 3705 Remote Support</u>					
	ORG		MDRCUA1		
18		DS	2X		Device address
1A		DS	8X		Userid
22		DS	2X		Resource identification

MICBLOK: VIRTUAL MACHINE POINTER LIST FOR VM/370 HARDWARE ASSIST

MICBLOK contains pointers to control registers, the segment table, and other values required by the virtual machine assist feature and the VM/370 Extended Control-Program Support (ECPS). This information is needed for the handling of certain instructions and privileged operations requested by the virtual machine. The VMMICRO field of the VMBLCK points to MICBLOK.

0	MICRSEG		MICCREG
8	MICVPSW		MICWORK
10	MICVTMR		MICACF

Hexadecimal Displacement	Field Name			Field Description, Contents, Meaning
0	MICRSEG	DS	1F	Real segment table pointer
4	MICCREG	DS	1F	Virtual control register pointer
8	MICVPSW	DS	1F	Virtual PSW pointer
8	MICVIP	ORG DS	MICVPSW 1X	Virtual interrupt pending bit
	<u>Bits defined in MICVIP</u>			
	MICPEND	EQU	X'80'	Virtual interrupt is pending; therefore, the virtual machine assist feature is not to handle change of PSW channel masks or external mask from disabled to enabled. All other bits in this byte must be 0.
9		DS	3X	Address of virtual PSW
C	MICWORK	DS	1F	Workspace pointer
10	MICVTMR	DS	1F	Location to be decremented when the virtual interval timer assist feature of VM/370 ECPS is being used
14	MICACF	DS	1F	Assist controls
14	MICEVMA	ORG DS	MICACF 1X	Expanded virtual machine assist control bits
	<u>Bits defined in MICEVMA</u>			
	MICLPSW	EQU	X'80'	LPSW simulation
	MICPTLB	EQU	X'40'	PTLE simulation
	MICSCSP	EQU	X'20'	SCKC, SPT simulation
	MICSIO	EQU	X'10'	SIO simulation
	MICSTSM	EQU	X'08'	STNSM, STOSM, and SSM simulation
	MICSTPT	EQU	X'04'	STPT simulation
	MICTCH	EQU	X'02'	TCH simulation
15		DS	3X	Reserved for IBM Use
	MICSIZE	EQU	(*-MICBLOK)/8	Size of DSECT in doublewords (X'03')

MIHREC

MIHREC: MISSING INTERRUPT HANDLER ERROR RECORD

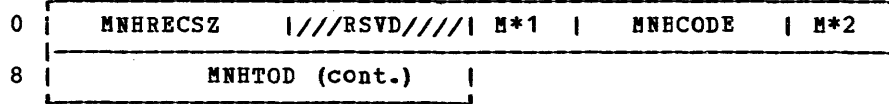
MIHREC is used in the SVC 76-initiated error recording process of type 70 MIH (Missing Interrupt Handler) records.

0	MIHKEYN	A*1	/A*2// /A*3// /A*4// //MIHSPE1//
8	MIHDTEN		MIHTMEN
10	MIHCPID		
18	MIHJOB		
20	MIHCUA2		MIHCUA1 MIHVOL
28	MIHVOL (cont.)		MIHDEVT
30	MIHINT		

Hexadecimal Displacement	Field Name			Field Description, Contents, Meaning
<u>24-Byte Header</u>				
0	MIHKEYN	DS	1H	Type and operating system
2	MIHSWS1	DS	1C	A*1 Switch byte 0
3	MIHSWS2	DS	1C	A*2 Reserved for IBM use
4	MIHSWS3	DS	1C	A*3 Reserved for IBM use
5	MIHRECNT	DS	1C	A*4 Reserved for IBM use
6	MIHSPE1	DS	1H	Reserved for IBM use
8	MIHDTEN	DS	1F	Date
C	MIHTMEN	DS	1F	Time
10	MIHCPID	DS	2F	Processor identification and model number
<u>Device Dependent Data</u>				
18	MIHJOB	DS	8X	Job whose I/O request is pending
20	MIHCUA2	DS	3X	CUA used to address the device
23	MIHCUA1	DS	3X	Primary device address
26	MIHVOL	DS	6X	Volume serial number of device
2C	MIHDEVT	DS	4X	Device type
30	MIHINT	DS	8X	Time interval used to check pending interrupt
	MIHSIZE	EQU	(*-MIHREC)	MIH record size in doublewords (X'07')

MNHDR: VM/370 MONITOR RECORD HEADER

MNHDR provides header information for following monitor records.



Hexadecimal Displacement	Field Name			Field Description, Contents, Meaning
0	MNBHDLEN EQU	4		Length of the block header (VB format)
0	MNHRECSZ DS	1H		Record size
2	RSVD DS	1H		Reserved for IBM use
4	MNHCLASS DS	1X	M*1	Monitor class
5	MNHCODE DS	XL2		Monitor code
7	MNHTOD DS	XL5	M*2	Current TOD value
	MNHDLEN EQU	*-MNHDR		Length of header record

MN000: VM/370 MONITOR PERFORM CLASS RECORD

MN000 provides an area for the accumulation of records dealing with privilege operations, paging, dispatching, and interrupt activity.

0	MN000WID			E0	MN000HI		MN000HD
8	MN000WPG			E8	MN000TC		MN000MN
10	MN000WIO			C0	MN000MO		MN000LR
18	MN000PRB			C8	MN000CP		MN000CH
20	MN000PSI		MN000CPA	DC	MN000TE		MN000CE
28	MN000NFL		MN000PSN	D8	MN000CT		MN000PE
30	MN000PRC		MN000RPC	E0	MN000PT		MN000EP
38	MN000SPC		MN000FLF	E8	MN000IP		MN000PB
40	MN000CPT		MN000SS	F0	MN000RR		MN000TCL
48	MN000PFF		MN000PRF	F8	MN000LCL		MN000CS
50	MN000PCS		MN000NXR	100	MN000CD		MN000HDI
58	MN000CPR		MN000CVI	108	MN000NDU		MN000NAU
60	MN000CCW		MN000ITI	110	MN000PRD		MN000PWR
68	MN000PTI		MN000CKI	118	MN000NPP		MN000SWS
70	MN000CSV		MN000CPG	120	MN000Q1N		MN000Q2N
78	MN000CIO		MN000CDS	128	MN000Q1E		MN000Q2E
80	MN000CDA		MN000CDB	130	MN000PPC		//////////
88	MN000CSC		MN000EK	138	MN000GTM		MN000DQM
90	MN000IK		MN000MS	140	MN000SWP		MN000EXT
98	MN000LP		MN000DI	148	MN000NXT		MN000ATT
A0	MN000SI		MN000SF	150	MN000CNT		
A8	MN000TI		MN000CI				

Hexadecimal Displacement	Field Name			Field Description, Contents, Meaning
0	MN000WID DS	XL8		Total system idle wait time
8	MN000WPG DS	XL8		Total system page wait time
10	MN000WIO DS	XL8		Total system I/O wait time
18	MN000PRB DS	XL8		Total system problem state time
20	MN000PSI DS	1F		No. of paging SIOs
24	MN000CPA DS	1F		No. of calls to DMKPAG
28	MN000NFL DS	1F		No. of page frames currently on free list
2C	MN000PSN DS	1F		No. of pages currently being swapped
30	MN000PRC DS	1F		No. of pages flushed but reclaimed

Hexadecimal Displacement	Field Name		Field Description, Contents, Meaning
34	MN000RPC DS	1F	No. of reserved pages
38	MN000SPC DS	1F	No. of shared system pages
3C	MN000FLF DS	1F	No. of times the free list was empty
40	MN000CPT DS	1F	No. of calls to DMKPTRFR
44	MN000SS DS	1F	No. of pages stolen from in-queue users
48	MN000PFF DS	1F	No. of pages swapped from the flush list
4C	MN000PRF DS	1F	No. of pages examined in stealing a page
50	MN000PCS DS	1F	No. of full scans done in stealing pages
54	MN000NXR DS	1F	No. of real external interruptions
58	MN000CPR DS	1F	No. of calls to DMKPRVLG
5C	MN000CVI DS	1F	No. of calls to DMKVIOEX
60	MN000CCW DS	1F	No. of calls to DMKCCW from DMKVIO
64	MN000ITI DS	1F	No. of interval timer interruptions reflected
68	MN000PTI DS	1F	No. of processor timer interruptions reflected
6C	MN000CKI DS	1F	No. of clock comparator interruptions reflected
70	MN000CSV DS	1F	No. of SVC interruptions reflected
74	MN000CPG DS	1F	No. of program interruptions handled
78	MN000CIO DS	1F	No. of I/O interruptions handled
7C	MN000CDS DS	1F	No. of calls to DMKDSP (main entry)
80	MN000CDA DS	1F	No. of fast reflects in DMSDSP
84	MN000CDB DS	1F	No. of dispatches for new PSWs
88	MN000CSC DS	1F	No. of calls to DMKSCDDL
8C	MN000EK DS	1F	Instruction count for X'08' SSK
90	MN000IK DS	1F	Instruction count for X'09' ISK
94	MN000MS DS	1F	Instruction count for X'80' SSM
98	MN000LP DS	1F	Instruction count for X'82' LFSW
9C	MN000DI DS	1F	Instruction count for X'83' DIAG
A0	MN000SI DS	1F	Instruction count for X'9CX0' SIO
A4	MN000SF DS	1F	Instruction count for X'9CX1' SIOF
A8	MN000TI DS	1F	Instruction count for X'9DX0' TIO
AC	MN000CI DS	1F	Instruction count for X'9DX1' CLRIO
B0	MN000HI DS	1F	Instruction count for X'9EX0' HIC
B4	MN000HD DS	1F	Instruction count for X'9EX1' HDV
B8	MN000TC DS	1F	Instruction count for X'9F' TCH
BC	MN000MN DS	1F	Instruction count for X'AC' STNSM
C0	MN000MO DS	1F	Instruction count for X'AD' STOSM
C4	MN000LR DS	1F	Instruction count for X'E1' LRA
C8	MN000CP DS	1F	Instruction count for X'E202' STIDP
CC	MN000CH DS	1F	Instruction count for X'E203' STIDC
D0	MN000TE DS	1F	Instruction count for X'E204' SCK
D4	MN000CE DS	1F	Instruction count for X'E206' SCKC
D8	MN000CT DS	1F	Instruction count for X'E207' STCKC
DC	MN000PE DS	1F	Instruction count for X'E208' SPT
E0	MN000PT DS	1F	Instruction count for X'E209' STPT
E4	MN000EP DS	1F	Instruction count for X'E20A' SPKA
E8	MN000IP DS	1F	Instruction count for X'E20B' IPK
EC	MN000PB DS	1F	Instruction count for X'E20D' PTLB
F0	MN000RR DS	1F	Instruction count for X'E213' RRB
F4	MN000TCL DS	1F	Instruction count for X'E6' STCTL
F8	MN000LCL DS	1F	Instruction count for X'E7' LCTL
FC	MN000CS DS	1F	Instruction count for X'EA' CS
100	MN000CD DS	1F	Instruction count for X'EB' CDS
104	MN000HDI DS	1F	Diagnose disk I/O simulation count
108	MN000NDU DS	1F	No. of users dialed to a virtual machine
10C	MN000NAU DS	1F	No. of users logged on
110	MN000PRD DS	1F	No. of page reads
114	MN000PWR DS	1F	No. of page writes
118	MN000NPP DS	1F	No. of system pageable pages

Hexadecimal Displacement	Field Name		Field Description, Contents, Meaning
11C	MN000SWS DS	1F	Sum of working sets of in-queue users
120	MN000Q1N DS	1F	No. of users in Q1
124	MN000Q2N DS	1F	No. of users in Q2
128	MN000Q1E DS	1H	No. of users eligible for Q1
12A	MN000Q2E DS	1H	No. of users eligible for Q2
12C	MN000INT DS	1H	MONITOR sampling interval (in seconds)
12E	MN000PPA DS	1H	Pseudo-cylinders ¹ of allocated temporary space
130	MN000PPC DS	1H	Pseudo-cylinders ¹ of system temporary space
132	DS	1H	Reserved for IBM use
134	MN000ISD DS	1F	Count of mini I/O block stack depletes
138	MN000GTM DS	1F	Count of mini I/O blocks in the queue
13C	MN000DMQ DS	1F	Count of mini I/O blocks processed
140	MN000SWP DS	1F	Count of SIOs on alternate path
144	MN000EXT DS	1F	Count of free storage extensions
148	MN000NXT DS	1F	Count of release of free storage extensions
14C	MN000ATT DS	1F	Count of TRYSPLIT attempt
150	MN000CNT DS	1F	Count of subpool splits
	MN000LEN EQU	*-MN000	Record length in bytes

¹A pseudo-cylinder comprises 100 available page slots for all system-owned volumes. The total available pseudo-cylinders per device is determined by the formula:

$$\frac{(\text{number of cylinders} \times \text{number of records per cylinder for a device})}{100}$$

MN001: VM/370 MONITOR PERFORM CLASS RECORD

MN001 gives information on the performance of the Attached Processor.

0	MN001WID	
8	MN001WPG	
10	MN001WIO	
18	MN001PRB	
20	MN001NXR	MN001CSV
28	MN001PRD	MN001PWR
30	MN001SSY	MN001NSY
38	MN001SFR	MN001NFR
40	MN001SRN	MN001NRN
48	MN001STM	MN001NTM
50	MN001SDP	MN001NDP
58	MN001NFL	MN001NFS
60	MN001NSD	MN001NVD
68	MN001NRU	

Hexadecimal Displacement	Field Name		Field Description, Contents, Meaning
0	MN001WID	DS	L8 Attached processor idle wait time
8	MN001WPG	DS	L8 Attached processor page wait time
10	MN001WIO	DS	L8 Attached processor I/C wait time
18	MN001PRB	DS	L8 Attached processor problem state time
20	MN001NR	DS 1F	Number of external interrupts to the attached processor
24	MN001CSV	DS 1F	Number of SVCs reflected by the attached processor
28	MN001PRD	DS 1F	Number of page reads by attached processor
2C	MN001PWR	DS 1F	Number of page writes by the attached processor
30	MN001SSY	DS 1F	Total time spin on system lock
34	MN001NSY	DS 1F	Total number of spins for system lock
38	MN001SFR	DS 1F	Total time spin on DMKPRE lock
3C	MN001NFR	DS 1F	Total number of spins for DMKPRE lock
40	MN001SRN	DS 1F	Total time spin on RUNLIST lock
44	MN001NRN	DS 1F	Total number of spins for RUNLIST lock
48	MN001STM	DS 1F	Total time spin on timer request lock
4C	MN001NTM	DS 1F	Total number of spins for timer request lock
50	MN001SDP	DS 1F	Total time spin on displacement lock
54	MN001NDP	DS 1F	Total number of spins for displacement lock
58	MN001NFL	DS 1F	Number of times CPFRELK set
5C	MN001NFS	DS 1F	Number of times CPFRESW set
60	MN001NSD	DS 1F	Number of times system lock request deferred
64	MN001NVD	DS 1F	Number of times VMBLCK lock deferred
68	MN001NRU	DS 1F	Number of DMKDSPRU entries
	MN001LEN	EQU *-MN001	Length of record

MN097, MN098

MN097: VM/370 MONITOR HEADER RECORD

MN097 provides header information for a file that contains data accumulated by VM/370 Monitor. This is the first record of the file.

0	MN097CPU	
8	MN097LEV	
10	MN097DAT	
18	MN097TIM	
20	MN097UID	
28	MN097CR8	MN097NUC
30	MN097FSS	MN097DPA
38	MN097TTS	MN097VR
40	MN097CPL	MN097APL

Hexadecimal Displacement	Field Name		Field Description, Contents, Meaning
0	MN097CPU DS	XL8	Processor serial number and model number
8	MN097LEV DS	CL8	Program level change
10	MN097DAT DS	CL8	Current date
18	MN097TIM DS	CL8	Current time
20	MN097UID DS	CL8	Userid of user who invoked MONITOR
28	MN097CR8 DS	1F	Value of control register 8
2C	MN097NUC DS	1F	Size of nucleus
30	MN097FSS DS	1F	Size of free storage
34	MN097DPA DS	1F	Size of dynamic paging area
38	MN097TTS DS	1F	Size of trace table
3C	MN097VR DS	1F	Size of V=R area
40	MN097CPL DS	1H	Logical address of main processor
42	MN097APL DS	1H	Logical address of alternate processor
	MN097LEN EQU	*-MN097	Length of header record

MN098: VM/370 MONITOR TRAILER RECORD

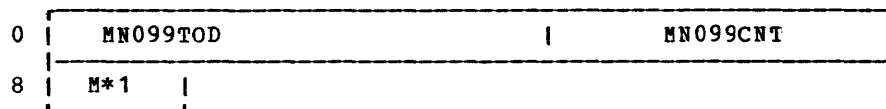
MN098 contains the userid of the user who has terminated current VM/370 Monitor activity. This is the last record of the file.

0	MN098UID	
---	----------	--

Hexadecimal Displacement	Field Name		Field Description, Contents, Meaning
0	MN098UID DS	CL8	ID of user stopping the VM/370 Monitor
	MN098LEN EQU	*-MN098	Length of trailer record

MN099: VM/370 MONITOR SUSPENSION RECORD

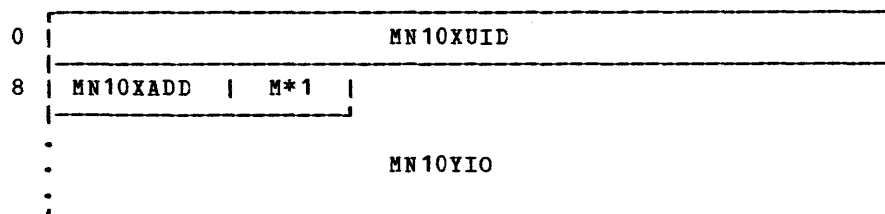
| MN099 contains information recorded when VM/370 Monitor activity is suspended because all
| buffers are full and are queued for output.



Hexadecimal Displacement	Field Name	Field Description, Contents, Meaning		
0	MN099TOD DS XL5	TOD clock value at suspensicn		
5	MN099CNT DS XL4 M*1	Count of suspensions		
	MN099LEN EQU *-MN099	Length of the suspension record		

MN10X: VM/370 MONITOR RESPONSE CLASS RECORD

MN10X contains information on the number of input or cutput console line transmissions for a given userid.



Hexadecimal Displacement	Field Name	Field Description, Contents, Meaning		
0	MN10XUID DS CL8	Userid		
8	MN10XADD DS 1H	Terminal line address		
	MN10XLEN EQU *-MN10X	Short record length		
A	MN10YCNT DS 1X M*1	Byte count for the following line		
	MN10YLEN EQU *-MN10X	Long record, header length		
B	MN10YIO EQU *	Input/output line starts here		

MN20X

MN20X: VM/370 MONITOR SCHEDULER CLASS RECORD

MN20X contains VM/370 Monitor data on CP's scheduler activity, dispatch queues, paging, and spool activity.

0	MN20XUID				
8	MN20XNPP			MN20XSWs	
10	MN20XQ1N			MN20XQ2N	
18	MN20XQ1E		MN20XQ2E		MN20XWSS M*1 M*2
20	MN20YTTI				
28	MN20YVTI				
30	MN204PRI		MN202PGR		MN202APR MN202REF
38	MN202RES		MN202PST		MN202IOC
40	MN202PNC			MN202LIN	
48	MN202CRD			M*3	

Hexadecimal Displacement	Field Name	Field Description, Contents, Meaning		
0	MN20XUID DS CL8	Userid being added and/or dropped from queue		
8	MN20XNPP DS 1F	No. of system pageable pages		
C	MN20XSWs DS 1F	No. of working sets of in-queue users		
10	MN20XQ1N DS 1F	No. of users in Q1		
14	MN20XQ2N DS 1F	No. of users in Q2		
18	MN20XQ1E DS 1H	No. of users eligible for Q1		
1A	MN20XQ2E DS 1H	No. of users eligible for Q2		
1C	MN20XWSS DS 1H	User's projected working set size		
1E	MN20XQNM DS 1X	M*1	Queue wherein additions and/or deletions are being made	
1F	MN20XPRC DS 1X	M*2	CP TRACE table processor identification	
	MN203LEN EQU *-MN20X	Length of class 2 code 3 record (Add queue)		
20	MN20YTTI DS XL8	Current VMTIME (CP simulation time)		
28	MN20YVTI DS XL8	Current VMVTIME (user virtual time)		
30	MN204PRI DS 1F	Eligible list priority		
	ORG MN204PRI			
30	MN202PRI DS 1H	Dispatch priority		
32	MN202PGR DS 1H	Pages read while in queue		
	MN204LEN EQU *-MN20X	Length of class 2 code 4 record (Drop queue)		
34	MN202APR DS 1H	Sum of pages read that are resident at each paging operation		
36	MN202REF DS 1H	No. of pages referenced while in queue		
38	MN202RES DS 1H	Current number of pages resident		
3A	MN202PST DS 1H	No. of pages stolen while in queue		
3C	MN202IOC DS 1F	Virtual nonspooled SIO count		
40	MN202PNC DS 1F	Virtual cards punched		
44	MN202LIN DS 1F	Virtual lines printed		
48	MN202CRD DS 1F	Virtual cards read		
4C	MN202LPR DS 1X	M*3	Last processor on which execution took place	
	MN202LEN EQU *-MN20X	Length of class 2 code 2 record (add to the eligible list)		

MN400: VM/370 MONITOR USER CLASS RECORD

MN400 provides user virtual machine statistics.

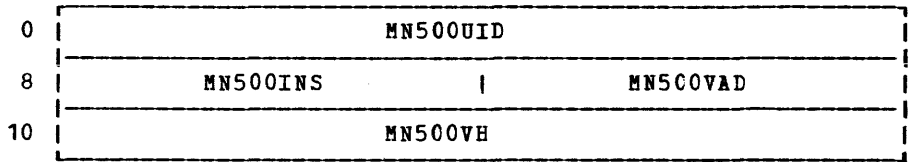
0	MN400UID								
8	MN400TTI								
10	MN400VTI								
18	MN400PGR					MN400PGW			
20	MN400IOC					MN400PNC			
28	MN400LIN					MN400CRD			
30	M*1	M*2	M*3	M*4	M*5	M*6	M*7	M*8	
38	M*9	M*10	M*11	M*12	M*13	/M*14/	MN400RES		
40	MN400WSS		MN400PDR		MN400PDK		MN400INT		
48	M*15								

Hexadecimal Displacement	Field Name			Field Description, Contents, Meaning
0	MN400UID	DS	CL8	Userid
8	MN400TTI	DS	XL8	Current VMTIME (in VMBLCK); CP simulation time
10	MN400VTI	DS	XL8	Current VMVTIME (in VMBLCK); user virtual time
18	MN400PGR	DS	1F	Total page reads for this user
1C	MN400PGW	DS	1F	Total page writes for this user
20	MN400IOC	DS	1F	Virtual nonspoiled SIO count
24	MN400PNC	DS	1F	Virtual cards punched
28	MN400LIN	DS	1F	Virtual lines printed
2C	MN400CRD	DS	1F	Virtual cards read
30	MN400RST	DS	1X	M*1 User running status
31	MN400DST	DS	1X	M*2 User dispatch status
32	MN400OST	DS	1X	M*3 User operating status
33	MN400QST	DS	1X	M*4 User queuing status
34	MN400PST	DS	1X	M*5 User processing status
35	MN400EST	DS	1X	M*6 User execution status
36	MN400TST	DS	1X	M*7 User tracing control status
37	MN400MLV	DS	1X	M*8 User message level
38	MN400QLV	DS	1X	M*9 User queue level
39	MN400CLV	DS	1X	M*10 User command level
3A	MN400TLV	DS	1X	M*11 User timer level
3B	MN400PND	DS	1X	M*12 Interrupt pending status
3C	MN400UPR	DS	1X	M*13 Directory or SET priority
3D	MN4RSV1	DS	1X	M*14 Reserved for IBM use
3E	MN400RES	DS	1H	Number of pages resident
40	MN400WSS	DS	1H	Estimated working set size
42	MN400PDR	DS	1H	Drum allocated page frames
44	MN400PDK	DS	1H	Disk allocated page frames
46	MN400INT	DS	1H	Monitor sampling interval (in seconds)
48	MN400LPR	DS	1X	M*15 Last processor on which execution took place
	MN400LEN	EQU	*-MN400	Length of class 4 code 0 record

MN500

MN500: VM/370 MONITOR INSTRUCTION SIMULATION CLASS RECORD

MN500 provides data on instructions simulated by CP.



Hexadecimal Displacement	Field Name	Field Description, Contents, Meaning
0	MN500UID DS CL8	Userid
8	MN500INS DS 1F	Privileged instruction
C	MN500VAD DS 1F	Virtual storage address of the instruction
10	MN500VH DS XL8	Current total of CP simulation time
	MN500LEN EQU *-MN500	Length of class 4 code 0 record

| MN600: VM/370 MONITOR DASTAP I/O COUNT RECORD

| • Header Record

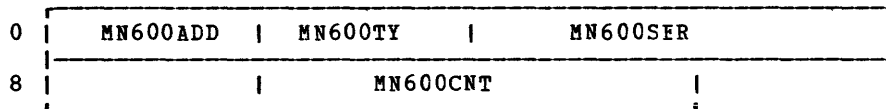
| MN600HDR header record provides the number of device data packages.



Hexadecimal Displacement	Field Name	Field Description, Contents, Meaning
0	MN600NUM DS 1H	Number of device data packages that follow length of header
	MN600HLN EQU *-MN600HDR	Length of header

| • I/O Count Record

| MN600DEV input/output count record provides information for each device in the device data packages.



Hexadecimal Displacement	Field Name	Field Description, Contents, Meaning
0	MN600ADD DS 1H	Device address
2	MN600TY DS 1H	VM/370 device type and/or codes
4	MN600SER DS CL6	Volume serial number of device
A	MN600CNT DS XL4	Device accumulated I/O count
	MN600DLN EQU *-MN600DEV	Length of each data record
	MN600MAX EQU (4096-MNBHDLEN-MNHRLLEN-MN600HLN)/MN600DLN	Maximum device count

| MN602: VM/370 MONITOR DASTAP UTILIZATION RECORD

- | • Header recrd

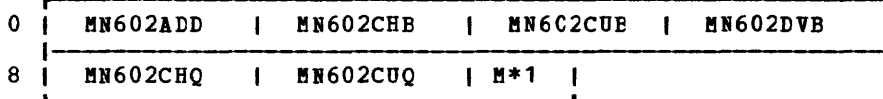
| MN602HDR provides the number of samples for intervals of device packages.



Hexadecimal Displacement	Field Name		Field Description, Contents, Meaning
0	MN602SAM DS	1H	Number of samples for interval

- | • Utilization Record

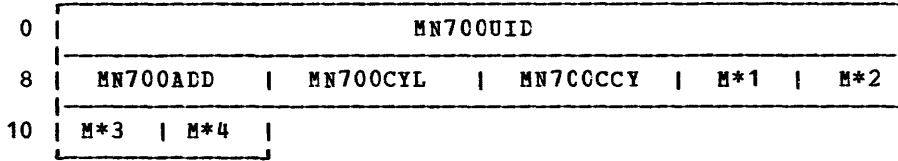
| MN602 provides, via CP MONITOR command, utilization data for DASD and tape devices.
 | There is one recrd for each device.



Hexadecimal Displacement	Field Name		Field Description, Contents, Meaning
0	MN602ADD DS	1H	Address of the device
2	MN602CHB DS	1H	Number of times channel is busy
4	MN602CUB DS	1H	Number of times control unit is busy
6	MN602DVB DS	1H	Number of times device is busy
8	MN602CHQ DS	1H	Input/output tasks queued on channel
A	MN602CUQ DS	1H	Input/output tasks queued on the control unit
C	MN602DVQ DS	1C	M*1 Input/output tasks queued on device
	MN602DLN EQU	*-MN602DEV	Length of device portion in dcublewords

MN700: VM/370 MONITOR SEEKS CLASS RECORD

MN700 provides, via CP MONITOR, the I/O tasks and cylinder seek activity of a specified DASD.

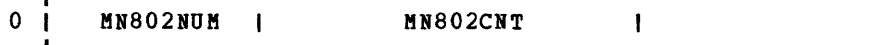


Hexadecimal Displacement	Field Name				Field Description, Contents, Meaning
0	MN700UID DS	CL8			Userid
8	MN700ADD DS	1H			Device address
A	MN700CYL DS	1H			Cylinder being sought
C	MN700CCY DS	1H			Current cylinder
E	MN700QDV DS	1X	M*1		I/O tasks queued on the device
F	MN700QCU DS	1X	M*2		I/O tasks queued on the control unit
10	MN700QCH DS	1X	M*3		I/O tasks queued on the channel
11	MN700DIR DS	1X	M*4		Seek direction: 00=lower, 01=higher
	MN700LEN EQU	*-MN700			Length of class 7 code 0 record

| MN802: VM/370 MONITOR SYSTEM PROFILE CLASS

- | • Header Recrd

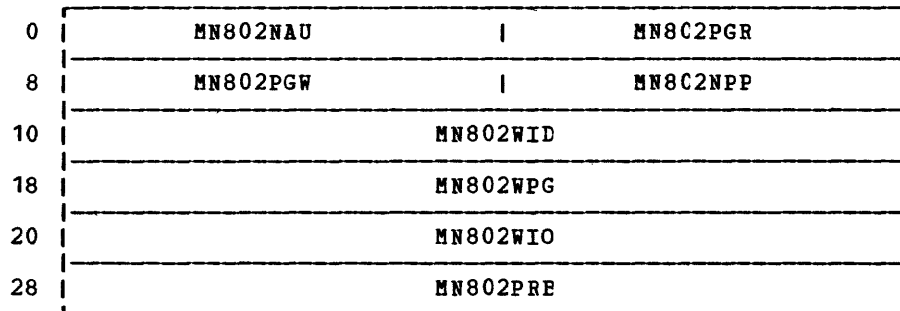
| MN802HDR provides the number of device block counters.



Hexadecimal Displacement	Field Name	Field Description, Contents, Meaning
0	MN802NUM DS 1H that follow	Number of device block counters
2	MN802CNT DS XL4	Device I/O count
	MN802DLN EQU *-MN802CTR	Length of the header

- | • System Profile Data

MN802CTR provides, via CP MONITOR command, additional system profile data. The monitor data includes: the I/O activity for each device, the number of logged on users, number of page read/writes, and the total system I/O, page wait, and problem state times.



Hexadecimal Displacement	Field Name	Field Description, Contents, Meaning
0	MN802NAU DS 1F	No. of logged on users
4	MN802PGR DS 1F	Total system page reads
8	MN802PGW DS 1F	Total system page writes
C	MN802NPP DS 1F	No. of system pageable pages
10	MN802WID DS XL8	Total system idle wait time
18	MN802WPG DS XL8	Total system page wait time
20	MN802WIO DS XL8	Total system I/O wait time
28	MN802PRB DS XL8	Total system problem time
	MN802CLN EQU *-MN802CTR	Length of each data entry

| MNDEVLST: VM/370 MONITOR CLASS 6 (DASTAP) DEVICE LIST

| MNDEVLST provides information on device activity. Pointed to by MCNDVLST, it (MNDEVLST) contains a list of RDEVBLK addresses to be used by class 6 (DASTAP) data collection, together with bins, for each device, in which the high frequency sampler (DMKENTTI) can accumulate control unit and device busy counts during each monitor interval. Note that channel busy counts are accumulated in a separate area pointed to by MONCHPTR, 5 doublewords and 2 bytes per entry.

```
|
|           0 |-----| | |
|           | MNRDEVB | MNCUESY | MNDVBSY |
|           |-----|
|
```

Hexadecimal Displacement	Field Name	Field Description, Contents, Meaning
0	MNRDEBV DS 1F	Address of RDEVBLK
4	MNCUBSY DS 1H	Control unit busy count
6	MNDVBSY DS 1H	Device busy count
	MNDEVLEN EQU *-MNDEVLST	Device entry size in doublewords (X'01')

MONCOM

MONCOM: VM/370 MONITOR COMMUNICATIONS AREA

MONCOM provides the control link for CP's monitoring activity, the user, and the tape drive.

0	MONARDB	M*1 M*2	MONDVNUM
8	MONDVLST		////////MONRSV1////////
10	MONAIOB		MONATRB
18	MONCLOCK		
20	MONSUSCK		
28	MONSUSCT		////////MONRSVD1////////
30	MONSAVE1		
.			
.			
70	MONSAVE2		
.			
.			
B0	MONUSER		
B8	MONSPLCT		MONSFB
C0	MONCURV		MONNXTV
C8	MONCURR		MONNITR
D0	MONDASA		MONDASB
D8	MONDAS	M*3 M*4	MONBUFNO
E0	MONCURBF		MONCRSLT
E8	MONIOBF		MONIOSLT
F0	MONSKLST		MONSACT
F8	MONCHPTR		MONUTRB
100	MONBUF1		
	MONBUF1V		
.			
.			
	(Variable number of buffers)		
			(Last address field)

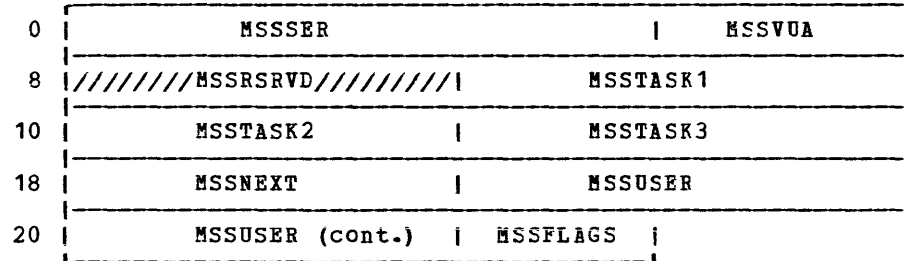
Hexadecimal Displacement	Field Name			Field Description, Contents, Meaning
0	MONARDB	DS	1F	Address of monitor tape real device block
4	MONFLAG1	DS	1X	M*1 Monitor flag
	<u>Bits defined in MONFLAG1</u>			
	MONSYSVM	EQU	X'80'	Flag used by user class routine
	CFSTOP	EQU	X'20'	MONITOR STOP command has been issued
	TRUN	EQU	X'10'	Tape rewind-unload CCW has been scheduled
	ERROR	EQU	X'08'	Tape error has occurred -- stcp VM/370 monitor
	MONTIINT	EQU	X'04'	Handling timer interruption
	MONLSTBK	EQU	X'02'	Handling the last block
	MONIBUF	EQU	X'01'	Only one buffer for VM/370 monitoring
5	MONFLAG2	DS	1X	M*2 Work byte
	<u>Bits defined in MONFLAG2</u>			
	SUSPEND	EQU	X'80'	VM/370 monitor has been suspended
6	MONDVNUM	DS	1H	Number of entries in real device list
8	MONDVLST	DS	1F	Address of the real device list
C	MONRSV1	DS	1F	Reserved for IBM use
10	MONAIOB	DS	1F	Address of monitor tape I/C block
14	MONATRB	DS	1F	Address of monitor timer request block
18	MONCLOCK	DS	1D	TOD clock stamp for each record
20	MONSUSCK	DS	1D	TOD clock value at last suspension
28	MONSUSCT	DS	1F	Suspension count
2C	MONRSVD1	DS	1F	Reserved for IBM use
30	MONSAVE1	DS	16F	Monitor internal save area for main processor
70	MONSAVE2	DS	16F	Monitor internal save area for attached processor
E0	MONUSER	DS	8C	User starting/stopping the VM/370 monitor
B8	MONSPLCT	DS	1F	Number of records on spool file
BC	MONSFB	DS	1F	Address of SFBLCK for spool file
C0	MONCURV	DS	1F	Virtual address of first virtual buffer
C4	MONNXTV	DS	1F	Virtual address of second virtual buffer
C8	MONCURR	DS	1F	Real address of first virtual buffer
CC	MONNXTR	DS	1F	Real address of second virtual buffer
D0	MONDASA	DS	1F	Address of next DASD buffer
D4	MONDASB	DS	1F	Address of previous DASD buffer
D8	MONDAS	DS	1F	Address of DASD buffer for the spool file
DC	MONEEX	DS		M*3 Flag byte
	<u>Bits defined in MONEEX</u>			
	CLCMD	EQU	X'80'	Spool file closed by command
DD	MONFLAG3	DS		M*4 Flag byte
	<u>Bits defined in MONFLAG3</u>			
	CLSUS	EQU	X'80'	Suspend during close
	EXHAUST	EQU	X'40'	Spool DASD slots exhausted
	CL	EQU	X'20'	Suspension necessary
	SPOOLED	EQU	X'10'	Monitor to spool active
DE	MONBUFNO	DS	1H	Reserved for IBM use
E0	MONCURBF	DS	1F	Address of current VM/370 monitor buffer
E4	MONCRSLT	DS	1F	Corresponding slot address
E8	MONIOBF	DS	1F	Address of VM/370 monitor buffer going to tape
EC	MONIOSLT	DS	1F	Corresponding slot address
F0	MONSKLST	DS	1F	Address for device list seeks
F4	MONSACT	DS	1F	Limit count for real time monitor
F8	MONCHPTR	DS	1F	Address of channel sampling data
FC	MONUTRB	DS	1F	Address of I/O utilization
100	MONBUF1	DS	1F	First VM/370 monitor buffer address

MONCOM

Hexadecimal Displacement	Field Name	Field Description, Contents, Meaning
104	MONBUF1V EQU MONBUF1+4	First buffer address of remaining variable number of buffers. There is one 4-byte entry for each monitor buffer. The last field contains X'FFFFFFFF'
	MONSIZE EQU (*-MONCOM)/8	Size of DSECT in doublewords
<u>Associated Monitor Control Flags</u>		
<u>Flags in DMKSYSAT</u>		
AUTOGO EQU X'80'		SYSMON setting for AUTODISK on
AUTOSPL EQU X'40'		Stop monitor when spool file record limit is reached
MONSLMT EQU X'20'		Sampling for real time Monitor
<u>Monitor Buffer Control Flag in Byte 3 of Buffer for Tape and in Byte 9 of Buffer for Spool</u>		
MONBUPIO EQU X'00'		Not collecting; being used for output
MONBUFAV EQU X'01'		Available for use
MONBUFAC EQU X'03'		Current active collector
<u>Monitor Buffer Control Flag in Byte 10 of Spool Buffer</u>		
TRAP EQU X'80'		Last buffer queued for I/O
UNFIN EQU X'40'		Close occurred before the buffer was full
<u>Flag in Spool File Control Block (SFBFLAG2)</u>		
SFBMON EQU X'01'		Monitor spool file identifier

MSSCOM: MSS COMMUNICATIONS CONTROL BLOCK

MSSCOM contains information necessary to request a MSS volume mount, request a MSS volume demount, or complete processing when a pack change interrupt is received on a MSS device. The MSSCOM blocks are chained from location DMKSSMQ in module DMKSSS.



Hexadecimal Displacement	Field Name			Field Description, Contents, Meaning
0	MSSSER	DS	CL6	Volume serial number of the MSS volume to be mounted or demounted
6	MSSVUA	DS	XL2	Device address for the volume
8	MSSRSRVD	DS	1F	Reserved for IBM use
C	MSSTASK1	DS	1F	Pointer to a CPEXBLOK for a pending MSS pack change interrupt
10	MSSTASK2	DS	1F	Pointer to a CPEXBLOK for a pending MSC return on mount or demount
14	MSSTASK3	DS	1F	Pointer to a CPEXBLOK for an I/C request to a volume being mounted
18	MSSNEXT	DS	1F	Next entry in the chain, or zero
1C	MSSUSER	DS	CL8	Name of the virtual machine that requires the MSS activity
24	MSSFLAGS	DS	X12	Binary flags representing the status of the request

Bits defined in MSSFLAGS

MOUNT	EQU	X'8000'	Mount volume MSSSER on address MSSVUA
DEMOUNT	EQU	X'2000'	Demount MSSSER from MSSVUA
MSSERR	EQU	X'400'	The MSC detected an error while attempting the requested action
RQENT	EQU	X'80'	This request is waiting to be passed to the MSC
MQENT	EQU	X'40'	This request has been passed to the MSC, and is awaiting a pack change interrupt
INPROC	EQU	X'20'	This request being processed by the MSC
MSGPROC	EQU	X'10'	The MSC has completed for this request and message DMKSSS088I is being sent
MSSSIZE	EQU	(*-MSSSER)/8	MSSCOM size in doublewords

NCPTBL

NCPTBL: NAMED 370X CONTROL PROGRAM TABLE

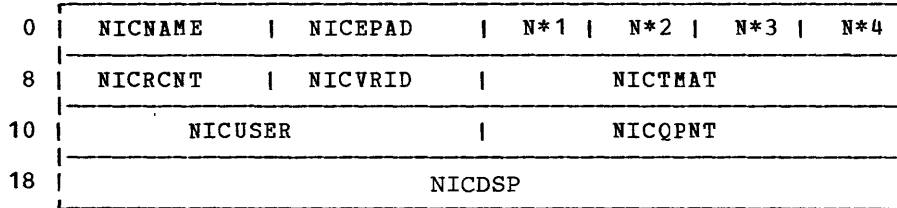
NCPTBL entries provide description information on 370x control program images saved on CP-owned volumes.

0	NCPNT	1	NCPSIZE
8	NCPNAME		
10	NCPVOL	N*1	N*2
18	NCPSTART	1	NCPAGCT

Hexadecimal Displacement	Field Name	Field Description, Contents, Meaning		
0	NCPNT DS 1F	Displacement to next entry		
4	NCPSIZE DS 1F	370x storage size required for load		
8	NCPNAME DS CL8	Control program reference name		
10	NCPVOL DS CL6	Volume identification of DASD containing saved image		
16	NCPFLAG DS 1X	N*1 CPTYPE flag byte		
<u>Bits defined in NCPFLAG</u>				
	NCPTNCP EQU X'01'	Network Control Program		
	NCPTCEP EQU X'02'	270x Emulation Control Program		
	NCPTPEP EQU X'03'	Partitioned Emulation Program		
17	NCPRSV1 DS 1X	N*2 Reserved for IBM use		
18	NCPSTART DS 1F	Pointer to first page (CCPL) on NCPVOL of saved NCP		
1C	NCPAGCT DS 1F	Total number of pages saved		
	NCPSIZE EQU (*-NCPSIZE)/8	NCPTBL size in doublewords (X'04')		

NICBLOK: NETWORK INTERFACE CONTROL BLOCK

NICBLOK contains control information related to 3704/3705 resources, teleprocessing lines, and display screen status information. The RDEVNICL field of RDEVLOK points to NICBLOK.



Hexadecimal Displacement	Field Name			Field Description, Contents, Meaning
0	NICNAME	DS	1H	370x NCP resource name
2	NICEPAD	DS	1H	Subchannel address when in EP mode
4	NICSTAT	DS	1X	N*1 Resource status flags
<u>Bits defined in NICSTAT</u>				
	NICERLK	EQU	X'80'	Device error lock is set
	NICNTRL	EQU	X'40'	Control operation is active
	NICDISA	EQU	X'20'	Resource inactive (offline)
	NICSWEP	EQU	X'10'	Resource is switchable to EP mode
	NICEPMD	EQU	X'08'	Resource now in emulator mode
	NICLTRC	EQU	X'02'	NCP line trace active
	NICDED	EQU	X'01'	Resource is dedicated
	NICTRQ	EQU	X'80'	Graphic device - timer request pending
	NIC HOLD	EQU	X'10'	Graphic device - screen full; in HOLD status
	NICMORE	EQU	X'08'	Graphic device - screen full; in MORE status
	NICRUNN	EQU	X'04'	Graphic device - screen in running status
	NICREAD	EQU	X'02'	Graphic device - read pending for screen input
	NICCPNA	EQU	X'01'	Graphic device - last input not accepted
5	NICFLAG	DS	1X	N*2 Interface control flags
<u>Bits defined in NICFLAG</u>				
	NICSESN	EQU	X'80'	Session is active for this device
	NICATTN	EQU	X'40'	Attention handling in progress
	NICPSUP	EQU	X'20'	Resource has print suppress feature
	NICATOF	EQU	X'10'	Suppress attention signal character
	NICENAB	EQU	X'08'	Resource is active and enabled
	NICDISB	EQU	X'02'	Resource to be disabled as soon as possible
	NICMTA	EQU	X'01'	Multiple terminal access resource
	NICFMT	EQU	X'80'	Graphic device - screen formatted VM/370 online
	NICDIAG	EQU	X'40'	Graphic device - screen written with DIAGNOSE
	NICALRM	EQU	X'10'	Graphic device - screen has an alarm message
	NICCARD	EQU	X'04'	Graphic device - data from card reader
	NICPROCN	EQU	X'01'	Graphic device - process control task now
6	NICLLEN	DS	1X	N*3 Terminal output line length
7	NICTYPE	DS	1X	N*4 Resource type and/or features
<u>Bits defined in NICTYPE</u>				
	NICCTLR	EQU	X'00'	Resource is the 370x
	NICLINE	EQU	X'80'	Resource is a teleprocessing line
	NICTERM	EQU	X'40'	Resource is a terminal device
	NICLGRP	EQU	X'20'	Resource is a logical line group

NICBLOK

Hexadecimal Displacement	Field Name		Field Description, Contents, Meaning
<u>Bits defined in NICTYPE (cont.)</u>			
	NICSDLC EQU	X'08'	LINE - Synchronous data link control
	NICLBSC EQU	X'04'	LINE - Binary synchronous line control
	NICSWCH EQU	X'02'	LINE - Switched line interface
	NICMLTP EQU	X'01'	LINE - Multiple-drop leased line
	NICTELE EQU	X'10'	TERM - Telegraph line adapter
	NICCIBM EQU	X'08'	TERM - Selectric-based terminal
	NICRCPU EQU	X'04'	TERM - Bisynch remote computer
	NICRSPL EQU	X'02'	TERM - Bisynch remote spool device
	NICGRAF EQU	X'01'	TERM - Bisynch remote graphics
	NIC3271 EQU	X'08'	Graphic device - 3271 control unit
	NIC3275 EQU	X'04'	Graphic device - 3275 standalone display station
	NICOPRDR EQU	X'10'	Graphic device - card reader feature
8	NICRCNT DS	1H	Retry count for BTU errors
A	NICVRID DS	1H	Virtual resource ID when dedicated
C	NICTMAT DS	1F	TOD clock value when attached
10	NICUSER DS	1F	VMBLOCK address of associated user
14	NICQPNT DS	1F	Pointer to input BTU chain
18	NICDSP DS	1D	Remote 3270 information
	ORG NICDSP		
	NICRSV1 DS	1X	Reserved for IBM use
	NICDTYPE DS	1X	Display station type
<u>Bits define in NICDTYPE</u>			
	NICD3277 EQU	X'04'	3277 display station
	NICD3275 EQU	X'02'	3275 display station
	NICMDL DS	1X	Display station model
	NICRSV2 DS	1X	Reserved for IBM use
	NICRSV3 DS	1X	Reserved for IBM use
	NICSIZE EQU	(*-NICBLOK)/8	Size of block in doublewords (X'03')
<u>Device Dependent Data - 3270 on Binary Synchronous Lines</u>			
	ORG NICEPAD		
2	NICCORD DS	1X	Current line coordinates
3	NICTMCD DS	1X	Terminal mode
<u>Bits defined in NICTMCD</u>			
	NICTABF EQU	X'80'	Second scan of screen's input area
	NICSIO EQU	X'40'	DIAGNOSE issued to input area
	NICAPL EQU	X'20'	APL cn for 3270 remote
	NICTEXT EQU	X'10'	Text feature on for 3270 remote
	ORG NICRCNT		
8	NICSELT DS	1H	Remote station selection characters
A	NICPOLL DS	1H	Remote station polling characters
C	NICATRB DS	1F	Timer request block address
<u>Equate Symbols for VM/370 Support of the 370x</u>			
	WRITBRK EQU	X'09'	Write break CCW operation code
	RDBUFLN EQU	96	Length of host read buffers
	RDBUFNO EQU	6	Number of host read buffers
<u>Sense Bits (sense byte 0) Peculiar to the 370x</u>			
	IPLREQ EQU	X'02'	IPL required--3705 inactive
	ABORT EQU	X'01'	Buffer depletion--transfer terminated

| NPRTEL: NAMED 3800 IMAGE LIBRARY TABLE

| NPRTEL lists by name all pages saved and indicates the DASD volume that contains the saved image.

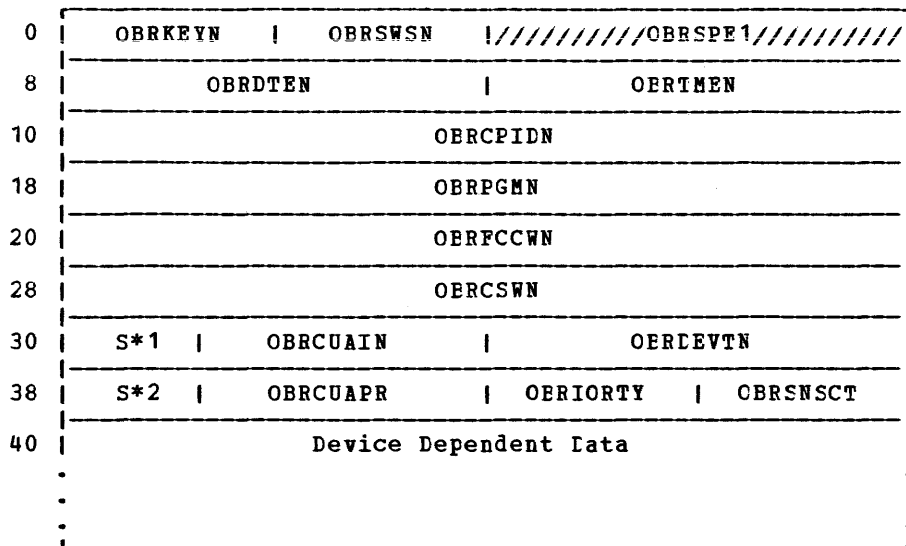
0	NPRPNT		NPRRSV1
8	NPRNAME		
10	NPRVOL		N*1 N*2
18	NPRSTART		NPRPAGCT

Hexadecimal Displacement	Field Name	Field Description, Contents, Meaning		
0	NPRPNT DS 1F	Chain pointer to next entry		
4	NPRRSV1 DS 1F	Reserved for IBM use		
8	NPRNAME DS CL8	Reference name for image library		
10	NPRVOL DS CL6	Volume of DASD containing the saved image		
16	NPRCNT DS 1X	N*1	Number of 3800s active on this image	
17	NPRRSV2 DS 1X	N*2	Reserved for IBM use	
18	NPRSTART DS 1F	CCPI of first page on NPRVOL		
1C	NPRPAGCT DS 1F	Number of pages saved		

OBRREC (Long OBR)

OBRREC: UNIT CHECK ERROR RECORD (LONG OUTBOARD RECORD)

OBRREC provides error, sense, and other statistical data needed for error recording on a specified channel-attached I/O device.



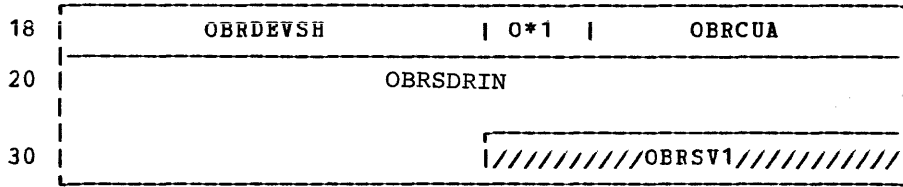
Hexadecimal Displacement	Field Name	Field Description, Contents, Meaning		
<u>24-Byte Header Record</u>				
0	OBRKEYN DS 1H	Record type		
2	OBRSWSN DS 1H	Switches		
<u>Bits defined in OBRSWSN</u>				
Byte 0	OBRMORE EQU X'80'	More records to follow		
	OBRTOE EQU X'40'	TOE clock instruction issued		
<u>Bits defined in OBRSWSN</u>				
Byte 1	OBRTOE EQU X'80'	SDR counters dumped at TOE		
	OBRTEMP EQU X'40'	Temporary error		
	OBRSHOBR EQU X'20'	Short record		
	OBRDEMNT EQU X'04'	Volume demounted		
4	OBRSP1 DS 1F	Reserved for IBM use		
8	OBRDTEN DS 1F	Date		
C	OERTMEN DS 1F	Time		
10	OERCPIDN DS 2F	Processor identifier and serial number		
	OBRHSIZE EQU (*-OBRREC)	Size of OBR header		
<u>End of 24-Byte Header Record</u>				
18	OBRPGMN DS 2F	Job identification		
20	OBRFCCWN DS 2F	Failing CCW		
28	OBRCSWN DS 2F	Failing CSW		
30	OBRDDCNT DS 1X	S*1	Number of doublewords in record	
31	OBRCUAIN DS 3C	Address of failing device		

Hexadecimal Displacement	Field Name		Field Description, Contents, Meaning
34	OBRDEVTN DS	1F	Device type
38	OBRSDRCT DS	1X	S*2 Number of SDR work area bytes
39	OBRUCUAPR DS	3C	Primary unit address
3C	OBRRIORTY DS	2X	Number of retries
3E	OBRSNSCT DS	2X	Number of sense bytes
	OBR1SIZE EQU	(*-OBRRECEN)	Size of long OBR record base
<u>Format of Device Dependent Data</u>			
<u>All DASD Units</u>			
40	OBRVOLN DS	8C	Volume identification
48	OBRLSKN DS	8X	Last seek address
50	OBRHAN DS	8X	Home address
<u>2314/2319 Format</u>			
58	OBRSDRWK DS	10X	SDR work area
62	OBRSENSN DS	6C	Sense data
	OBR2SIZE EQU	(*-OBRRECEN)	Maximum size of 2314/2319 record
<u>3350/3340/3330/2305 Format</u>			
	ORG OBRSDRWK		
58	OBR33SNS DS	24C	3350/3340/3330/2305 sense data
	OBR3SIZE EQU	(*-OBRRECEN)	Maximum size of 3330/3340/2305 record
<u>Unit Record Format</u>			
	ORG OBRVOLN		
40	OBRURST DS	10X	SDR work area
4A	OBRURSNS DS	1C	Unit record sense data
<u>3505/3525 Format</u>			
	ORG OBRVOLN		
40	OBR3505S DS	1C	3505/3525 sense data
<u>3203 Format</u>			
	ORG OBRVOLN		
40	OBRCORRL DS	1X	Correlation number
41	DS	7X	Reserved for IBM use
48	OBRSDRO3 DS	10X	SDR work area
52	OBR3203S DS	24C	3203 sense data
<u>3211 Format</u>			
	ORG OBRVOLN		
40	OBRCORL DS	1X	Correlation number
41	DS	7X	Reserved for IBM use
48	OBRSDR32 DS	10X	SDR work area
52	OBR3211S DS	6C	3211 sense data
<u>2400 Tape Format</u>			
	ORG OBRLSKN		
48	OBRTAPST DS	10X	SDR work area
52	OBRTAPSN DS	24C	Tape sense data
<u>3420/3410 Tape Format</u>			
	ORG OBRLSKN		
48	OBRDVDEP DS	16C	Device dependent data
58	OBR342ST DS	20X	SDR work area
6C	OBR3420S DS	24C	3420 sense data

OBRREC (Short OBR)

OBRREC: UNIT CHECK ERROR RECORD (SHORT OUTEOARE RECORD)

OBRREC provides error, sense, and other statistical data needed for error recording on a specified channel-attached I/O device.



Hexadecimal Displacement	Field Name	Field Description, Contents, Meaning
--------------------------	------------	--------------------------------------

24-Byte Header Record

Note: The 24-byte Header Record for the Short OBR is identical to that of the Long OBR, the description of which precedes this block.

18	OBRDEVSH DS	1F	Device type
1C	OBRSDRSH DS	1X	0*1 Number of SDR work area bytes
	OBRSIZE EQU	(*-OBRREC)	Size of short OBR record base
	OBRSIZE1 EQU	(*-OBRREC+7)/8	Size in doublewords (X'04')
1D	OBRCUA DS	3X	Channel and unit address
20	OBRSDRIN DS	20X	SDR work area
	ORG OBRSDRIN		
20	OBRSSDR1 DS	10X	SDR work area
2A	OBRSSDR2 DS	10X	SDR work area
34	OBRSV1 DS	1F	Reserved for IBM use
	OBRSIZE2 EQU	(*-OBRREC+7)/8	Size in doublewords (X'07')

OWNDLIST: CP-OWNED VOLUMES LIST

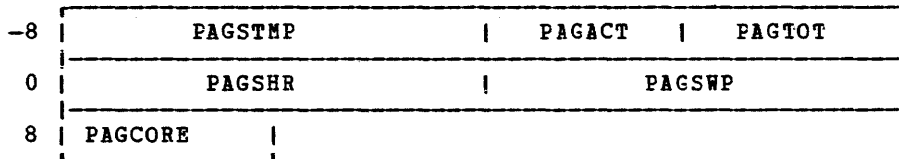
OWNDLIST contains a list of all the system owned DASD volumes that are used for paging, spooling, and temporary disk storage activity. Each entry specifies the volume identity and its preferred use (that is, paging/spooling/I-disk space). This block is generated by the SYSOWN macro at system generation.



Hexadecimal Displacement	Field Name	Field Description, Contents, Meaning
0	OWNDVSER DS CL6	Volume serial number
6	OWNRDEV DS 1H	Displacement of RDEVBLK for the volume
6	ORG OWRDEV OWNDPREF DS 1H	Allocation preference

PAGTABLE: TRANSLATION PAGE TABLE

PAGTABLE is used by CP for allocating and referencing storage. It is referenced by the segment table (SEGTABLE) data area and contains a pointer to the swap table (SWPTABLE) which, in turn, is related to a DASD cylinder location. The SWPPAG field of SWPTABLE points to PAGTABLE.



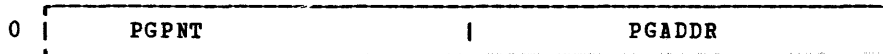
Hexadecimal Displacement	Field Name	Field Description, Contents, Meaning
	<u>8-Byte Header</u>	
-8	PAGSTMP DS 1F	Page table time stamp
-4	PAGACT DS 1H	Count of active segment table entries for this segment
-2	PAGTOT DS 1H	Count of total segment table entries for this segment
0	PAGSHR DS 1F	Pointer to NAME table (SHRTABLE)
4	PAGSWP DS 1F	Pointer to swap table (SWPTABLE)

PAGTABLE, PGBLCK

Hexadecimal Displacement	Field Name			Field Description, Contents, Meaning
8	PAGCORE	DS	1H	Real page address (start of page table for hardware)
<u>Bits defined in PAGCORE+1</u>				
	PAGINVAL	EQU	X'08'	PAGTAELE entry invalid
	PAGREF	EQU	X'01'	Page has been referenced
<u>Equates Used in Attached Processor Support for Shared Segments</u>				
	PAGTSWP	EQU		(PAGCORE-PAGSTMP+16*I'PAGCCRE) length of a full 16 entry page table
	PAGBMP	EQU		(PAGTSWP+(SWPFLAG-SWPVM)+16*(SWPCODE-SWPFLAG+1)*8) length of a contiguous page and swap table

PGBLCK: PSEUDO PAGE FAULT STACK BLOCK

PGBLCK is used by VM/VS Handshaking. The block is created and stacked when a multiprogramming or multitasking VS1 program interrupt occurs when a referenced page is not available in storage. The VMGPNT field in the VMBLCK points to PGBLCK.



Hexadecimal Displacement	Field Name			Field Description, Contents, Meaning
0	PGPNT	DS	1F	Pointer to next page block on the stack
4	PGADDR	DS	1F	Virtual page fault address
	PGBSIZE	EQU	(*-PGBLCK+7)/8	PGBLCK size in doublewords (X'01')

PSA: PREFIX STORAGE AREA (LOW STORAGE LOCATIONS)

PSA is the primary control block for controlling CP and virtual machine activity. This control block contains the normal low core IPL, logout, and PSW information; the processor model and type and features of the processor; and save areas used by EALR and FREE. This block also contains monitor and trace data and the necessary linkages to virtual machines, real devices, and spool files.

Note: All fields reside in real PSA unless otherwise specified. Fields residing in absolute PSA are specifically identified. For uniprocessor operation, real PSA equals absolute PSA (or 0). If the system was running in AF mode when a catastrophic error occurred, the Attached Processor will no longer be running. System recovery is in uniprocessor mode and the real PSA will no longer be zero.

Page 0, Machine Usage

0	IPLPSW		IPLCCW1	37C	CPID		CPABEND		P*3		P*4		ASYSVM
10	IPLCCW2		EXOPSW	38C	ARSPPR		ARSPFU		ARSEFD		ARICPU		
20	SVCOPSW		PROPSW	39C	ARIOPR		ARIORD		P*5		P*6		ARSPAC
30	MCOPSW		IOOPSW	3AC	AVMREAL		ASYSABND		ASYSLC		ASYSOP		
40	CSW		CAW		QUANTUMR	3EC	ARIOCT		ARIOCH		ARICCU		ARICDV
50	TIMER		QUANTUM		EXNPSW	3CC	ARIOCC		ARIOUC		ARICDC		ACORETEL
60	SVCNPSW		PRNPSW	3DC	APAGCF		CPCREG0		CPCREG6		CPCREG8		
70	MCNPSW		IONPSW	3EC	TINEDISP		ASVCLIST		AVMALIST		LASTUSER		
80			CPULOG	3FC	PAGECUR		MONNEXT		PAGEND		PAGENXT		
100			FXDLOG	40C	TRACEFLG		TTSEGCNT		////////////////////////////////				
160			FPRLOG						////////////////////////////////				
180			GPRLOG						////////////////////////////////				
1C0			CRLOG	43C	INSTWRD1		INSTWRD2		INSTWRD3		INSTWRD4		
200			TEMPSAVE	44C					Constants Pool				
240			BALRSAVE	.									
280			FREESAVE	.									
2C0			FREWORK	4TC	APTRLK		NOADD		X4CFPS		XRIGHT24		
2F0	DATE		TODATE	4FC	XPAGNUM		XRIGHT16		AFREE		AFRET		
300	STARTIME		CPUID	4FC	AQCNT		ADSPCH		AFTRAN		X2048END		
310	IDLEWAIT		PAGEWAIT	50C					DUMPSAVE				
320	IONTWAIT		PROBTIME	.									
330	RUNPSW		RUNUSER		DSPLPSW	54C			SIGSAVE				
340	RUNCRO		RUNCR1		CPSTAT		CPRESTR						
350	PGREAD		PGWRITE		PGWAITIM	58C			LCKSAVE				
360	////PGWAITPG////		PSASVCCT		P*1		P*2						

PSA

5C0	MFASAVE	6E0	CHGREGS /////RESERVED/////
600	SWTHSAVE	6C0	UNSHRVM P*10 P*11 /////RESERVED/////
640	LOCKSAV	6D0	STACKVM UNSHRVM2 ADMKCFE RESERVED/
650	SVCREGS	6E0	//////////RESERVED (ccnt)//////////
660	PREFIXA PREFIXB PSACPYBF //RESVD//	6F0	ALOKUM RESERVE ALCKSE AEXTSP
670	WAITSTRT WAITEND	700	ATMRSN //////////RESERVED//////////
680	PWTPAGES ACTIVTRQ EMSPEND EMSREC	710	MCMREGS
690	XCPEND P*7 P*8 P*9 AFSTATUS	750	LORSAVE2
6A0	AMCHARFA SHRLKCNT PROBSTRT		

Hexadecimal Displacement	Field Name			Field Description, Contents, Meaning
0	<u>Machine Usage</u> IPLPSW	DS	1D	IPL start PSW
		ORG	IPLPSW	
	RSRTNPSW	DS	1D	Restart new PSW
	RSRTOPSW	DS	1D	Restart old PSW
8	IPLCCW1	DS	1D	IPL CCW
10	IPLCCW2	DS	1D	IPL CCW
		ORG	IPLCCW1	
8	PSARSV3	DS	1F	Reserved for IBM use
C	TRACSTRT	DS	1F	Address of start of trace table. Note that TRACSTRT is in absolute PSA
10	TRACEND	DS	1F	Address of end of trace table. Note that TRACEND is in absolute PSA
14	TRACCURR	DS	1F	Address of next available trace table entry. Note that TRACCURR is in absolute PSA
18	EXOPSW	DS	1D	External old PSW
20	SVCOPSW	DS	1D	SVC old PSW
28	PROPSW	DS	1D	Program old PSW
30	MCOPSW	DS	1D	Machine check old PSW
38	IOOPSW	DS	1D	I/O old PSW
40	CSW	DS	1D	Channel status word
48	CAW	DS	1F	Channel address word
4C	QUANTUMR	DS	1F	Interval timer value at last interrupt
50	TIMER	DS	1F	13-microsecond interval timer
54	QUANTUM	DS	1F	Interval timer value at last dispatch
58	EXNPSW	DS	1D	External new PSW
60	SVCNPSW	DS	1D	SVC new PSW
68	PRNPSW	DS	1D	Program new PSW
70	MCMPSW	DS	1D	Machine check new PSW
78	IONPSW	DS	1D	I/O new PSW

Hexadecimal Displacement	Field Name			Field Description, Contents, Meaning
80	CPULOG	DS	16D	Processor and storage logout area
		ORG	CPULOG	
80		DS	1F	Reserved for IBM use
84	INTEXF	DS	1F	External interrupt code (fullword)
	<u>Bits defined in INTEXF</u>			
86	INTEX	EQU	INTEXF+2	External interrupt code (halfword)
88	INTSVCL	DS	1H	SVC instruction length code (ILC)
8A	INTSVC	DS	1H	SVC interrupt code
8C	INTPRL	DS	1H	Program instruction length code (ILC)
8E	INTPR	DS	1H	Program interrupt code
90	TREXADD	DS	1F	Translation exception address
94	MONCLASS	DS	1H	Monitor class
96	PERCODE	DS	1H	PER code to be reflected
98	PERADD	DS	1F	Address of instruction causing PER interrupt
9C	MONCODE	DS	1F	Monitor code
A0		DS	1D	Reserved for IBM use
A8	CHANID	DS	1F	Channel ID
AC	IOELPNTR	DS	1F	I/O extended logout (IOEL) pointer
B0	ECSWLOG	DS	1F	Limited channel logout (ECSW)
B4		DS	1F	Reserved for IBM use
B8	INTRKFLIN	DS	1F	I/O interrupt key, flags, and interface address
	<u>Bits defined in INTRKFLIN</u>			
BA	INTTIO	EQU	INTRKFLIN+2	I/O interrupt device address (halfword)
BC		DS	11F	Reserved for IBM use
E8	INTMC	DS	1D	Machine check interrupt code
F0		DS	1F	Reserved for IBM use
F4	INTRC	DS	1X	External damage reason code
F5		DS	3X	Reserved for IBM use
F8	FAILSTAD	DS	1F	Failing storage address
FC	REGNCODE	DS	1F	Region code
100	FXDLOG	DS	12D	Fixed logout area
160	FPRLOG	DS	4D	Floating-point register logout area
180	GRLOG	DS	16F	General register logout area
1C0	CRLOG	DS	16F	Control register logout area
200	CPUSAGE	DS	0H	End of machine usage; start of CP usage
		ORG	CPUSAGE	
200	TEMPSAVE	DS	16F	Temporary save area
		ORG	TEMPSAVE	
200	TEMPR0	DS	1F	Registers 0-15
204	TEMPR1	DS	1F	
208	TEMPR2	DS	1F	
20C	TEMPR3	DS	1F	
210	TEMPR4	DS	1F	
214	TEMPR5	DS	1F	
218	TEMPR6	DS	1F	
21C	TEMPR7	DS	1F	
220	TEMPR8	DS	1F	
224	TEMPR9	DS	1F	
228	TEMPR10	DS	1F	
22C	TEMPR11	DS	1F	
230	TEMPR12	DS	1F	
234	TEMPR13	DS	1F	
238	TEMPR14	DS	1F	
23C	TEMPR15	DS	1F	

Hexadecimal Displacement	Field Name			Field Description, Contents, Meaning
240	BALRSAVE	DS	16F	BALR linkage save area
		ORG	BALRSAVE	
240	BALR0	DS	1F	Registers 0-15
244	BALR1	DS	1F	
248	BALR2	DS	1F	
24C	BALR3	DS	1F	
250	BALR4	DS	1F	
254	BALR5	DS	1F	
258	BALR6	DS	1F	
25C	BALR7	DS	1F	
260	BALR8	DS	1F	
264	BALR9	DS	1F	
268	BALR10	DS	1F	
26C	BALR11	DS	1F	
270	BALR12	DS	1F	
274	BALR13	DS	1F	
278	BALR14	DS	1F	
27C	BALR15	DS	1F	
280	FREESAVE	DS	16F	DMKFRF save area
		ORG	FREESAVE	
280	FREER0	DS	1F	Registers 0-15
284	FREER1	DS	1F	
288	FREER2	DS	1F	
28C	FREER3	DS	1F	
290	FREER4	DS	1F	
294	FREER5	DS	1F	
298	FREER6	DS	1F	
29C	FREER7	DS	1F	
2A0	FREER8	DS	1F	
2A4	FREER9	DS	1F	
2A8	FREER10	DS	1F	
2AC	FREER11	DS	1F	
2B0	FREER12	DS	1F	
2B4	FREER13	DS	1F	
2B8	FREER14	DS	1F	
2BC	FREER15	DS	1F	
2C0	FREWORK	DS	12F	DMKFRF work area
2F0	DATE	DS	CL8	Date - mm/dd/yy - edited EECDIC
2F8	TODATE	DS	1D	TOD clock at hh.mm.ss today - local time
300	STARTIME	DS	1D	Date and time started - TOD clock value
308	CPUID	DS	1D	Processor identification field
		ORG	CPUID	
308	CPUVERSN	DS	1X	Version code
309	CPUSER	DS	3X	Processor serial number - packed unsigned
30C	CPUMODEL	DS	2X	Processor model number
30E	CPUMCELL	DS	1H	Maximum length in bytes of MCEL
310	IDLEWAIT	DC	X'7FFFFFFFFFFFFFF00'	Total system idle wait time
318	PAGWAIT	DC	X'7FFFFFFFFFFFFFF00'	Total system page wait time
320	IOWAIT	DC	X'7FFFFFFFFFFFFFF00'	Total system I/O wait time
328	PROBTIME	DC	X'7FFFFFFFFFFFFFF00'	Total system problem state time
330	RUNPSW	DS	1D	PSW last loaded by dispatcher
338	RUNUSER	DS	1F	Address of dispatched VMELCK
33C	DSPLPSW	DS	1F	Load PSW instruction used to dispatch
340	RUNCRO	DS	1F	Control register 0 at dispatch
344	RUNCR1	DS	1F	Control register 1 at dispatch

Hexadecimal Displacement	Field Name			Field Description, Contents, Meaning
348	CPSTAT	DS	1F	CP running status
		ORG	CPSTAT	
348	CPSTATUS	DS	1X	CP running status
	<u>Bits defined in CPSTATUS</u>			
	CPWAIT	EQU	X'80'	CP in wait state
	CPRUN	EQU	X'40'	CP running user in RUNUSER
	CPEX	EQU	X'20'	CP executing stacked request
	CPFVRUN	EQU	X'10'	Reserved for IBM use
	CPSUPER	EQU	X'08'	Processor is executing in supervisor state
349	XTNDLOCK	DC	1X	System extending free storage if it is equal to X'FF'. Note that XTNDLOCK is in absolute PSA.
34A	CPSTAT2	DC	1X	Flag byte
	<u>Bits defined in CPSTAT2</u>			
	CPMICAVL	EQU	X'80'	Virtual machine assist available on processor
	CPMICON	EQU	X'40'	Virtual machine assist is on for system
	CPSHRLK	EQU	X'20'	CP processing shared named system page
	CPASTAVL	EQU	X'08'	CP assist available on processor
	CPASTON	EQU	X'04'	CP assist is on for system
34B	CPSTAT3	DS	1X	Wait time accounting flag
	<u>Bits defined in CPSTAT3</u>			
	CPTIDLE	EQU	X'80'	Timer contains idle time
	CPTPAGE	EQU	X'40'	Timer contains page wait time
	CPTIONT	EQU	X'20'	Timer contains I/O wait time
34C	CPRESTR	DS	1F	Restart address if external interrupt marks page invalid
350	PGREAD	DS	1F	Total number of page reads
354	PGWRITE	DS	1F	Total number of page writes
358	PGWAITIM	DS	1D	Time spent in page wait, multiplied by number of pages waiting
360	PGWAITPG	DS	1D	Reserved for IBM use
368	PSASVCT	DS	1F	Total number of user SVCs
36C	PAGELoad	DS	1H	P*1 Page wait percent, last measurement
36E	PAGERATE	DS	1H	P*2 Paging rate, pages per second Note that PAGERATE is in absolute PSA.
370	PSENDCLR	DS	0F	End of area cleared by DMKCPINT
370	CPID	DS	1F	CP running identifier. Note that CPID is in absolute PSA.
374	CPABEND	DS	1F	CP abend code
378	PSTARTSV	DS	0F	Start of save/restored code
378	SYSIPLDV	DS	1H	P*3 Device address of system IFL device
37A	PGSRATIO	DC	H'0'	P*4 Page steals/total replenished
37C	ASYSVM	DC	V(DMKSYSVM)	Address of system VMBLOCK
380	ARSPPR	DC	V(DMKRSPPR)	Address of system printer file chain.
384	ARSPPU	DC	V(DMKRSPPU)	Address of system punch file chain.
388	ARSPRD	DC	V(DMKRSPRD)	Address of system reader file chain.
38C	ARIOPU	DC	V(DMKRIOPU)	Address of system punch table.
390	ARIOPR	DC	V(DMKRIOPR)	Address of system printer table.
394	ARIORD	DC	V(DMKRIORD)	Address of system reader table.
398	IPUADDR	DS	1H	P*5 Instruction processing address
39A	PSAMSS	DS	1H	P*6 Address of MSS volume
	<u>Bits defined in PSAMSS</u>			
	MSSPRES	EQU	X'80'	The MSS is online and the MSS communicator has been initialized

PSA

Hexadecimal Displacement	Field Name			Field Description, Contents, Meaning
39C	ARSPAC	DC	V(DMKRSPAC)	Address of system accounting chain
3A0	AVMREAL	DC	A(0)	VMELCK address of virtual=real user. Note that AVMREAL is maintained in both PSAs
3A4	ASYSABND	DC	A(0)	Address of system abend printer
3A8	ASYSLC	DC	V(DMKSYSYC)	Address of SYSLOCS information
3AC	ASYSOP	DC	V(DMKSYSOP)	Address of system operator VMELCK
3B0	ARIOCT	DC	V(DMKRIOCT)	Address of real channel index table
3B4	ARIOCH	DC	V(DMKRIOCH)	Address of first RCHBLCK
3B8	ARIOCU	DC	V(DMKRIOCU)	Address of first RCUBLOK
3BC	ARIODV	DC	V(DMKRIODV)	Address of first RDEVBLCK
3C0	ARIOCC	DC	V(DMKRIOCC)	Address of count of real system channels
3C4	ARIOUC	DC	V(DMKRIOUC)	Address of count of real system control units
3C8	ARIODC	DC	V(DMKRIODC)	Address of count of real system devices
3CC	ACORETBL	DC	V(DMKSYSCS)	Address of system CORTABLE
3D0	APAGCP	DC	A('FFFFFF')	Address of first pageable program
3D4	CPCREG0	DC	X'808008C0'	CP architecture control and external mask
3D8	CPCREG6	DC	F'0'	CP assist and virtual machine assist mask
3DC	CPCREG8	DC	F'0'	MONITOR CALL enable mask
3E0	TIMEDISP	DS	1F	Timer displacement for charge
3E4	ASVCLIST	DC	V(DMKSVCS)	Address of CP assist pointer list
3E8	AVMALIST	DC	V(DMKPRVMA)	Address of expanded virtual machine assist pointer list
3EC	LASTUSER	DC	V(DMKSYSVM)	Last user to be dispatched
3F0	PAGECUR	DS	1F	Current monitor buffer page address. Note that PAGECUR is in absolute PSA.
3F4	MONNEXT	DS	1F	Next available address in monitor buffer. Note that MONNEXT is in absolute PSA.
3F8	PAGEND	DS	1F	Last address in current monitor buffer page. Note that PAGEND is in absolute PSA.
3FC	PAGENXT	DS	1F	Alternate monitor buffer page address. Note that PAGENXT is in absolute PSA.
400	TRACEFLG	DS	1F	Trace table flags
		ORG	TRACEFLG	
400	TRACFLG1	DS	1X	Trace table flag
	<u>Bits defined in TRACFLG1</u>			
	TRAC01	EQU	X'80'	External interrupt tracing on
	TRAC02	EQU	X'40'	SVC interrupt tracing on
	TRAC03	EQU	X'20'	Program interrupt tracing on
	TRAC04	EQU	X'10'	Machine check tracing on
	TRAC05	EQU	X'08'	I/O interrupt tracing on
	TRAC67	EQU	X'04'	FREE/FRET call tracing on
	TRAC08	EQU	X'02'	Enter dispatch tracing on
	TRAC09	EQU	X'01'	Queue drop tracing on
401	TRACFLG2	DS	1X	Trace table flag
	<u>Bits defined in TRACFLG2</u>			
	TRAC0A	EQU	X'80'	Run user tracing on
	TRAC0C	EQU	X'40'	Unstack I/O interrupt tracing on
	TRAC0D	EQU	X'20'	Virtual CSW stored tracing on
	TRACBEF	EQU	X'10'	SIO, TIO, and HDV tracing on
	TRAC10	EQU	X'08'	Unstack IOBLOK or TRQBLOK tracing on
	TRAC11	EQU	X'04'	Trace BTU activity for 370x NCP
	TRAC12	EQU	X'02'	Lock spin tracing active
	TRAC13	EQU	X'01'	Signal processor tracing active
402	TRACFLG3	DS	1H	Reserved for IBM use
404	TTSEGCNT	DS	1F	Count of total page/swap tables in system. Note that TTSEGCNT is in absolute PSA.
408	PSARSV15	DS	5D	Reserved for IBM use
430	INSTWRD1	DC	F'0'	Reserved for installation use

Hexadecimal Displacement	Field Name		Field Description, Contents, Meaning	
434	INSTWRD2	DC	F'0'	Reserved for installation use
438	INSTWRD3	DC	F'0'	Reserved for installation use
43C	INSTWRD4	DC	F'0'	Reserved for installation use
<u>List of Frequently Used Constants</u>				
440	ZEROES	DC	6D'0'	
470	BLANKS	DC	8X'40'	
478	PFS	DC	8X'FF'	Also = -1
440	F0	EQU	ZEROES	
480	F1	DC	F'1'	
484	F2	DC	F'2'	
488	F3	DC	F'3'	
48C	F4	DC	F'4'	
490	F5	DC	F'5'	
494	F6	DC	F'6'	
498	F7	DC	F'7'	
49C	F8	DC	F'8'	
4A0	F9	DC	F'9'	
4A4	F10	DC	F'10'	
4A8	F15	DC	F'15'	Also = X'CCCC000F'
4AC	F16	DC	F'16'	
4B0	F20	DC	F'20'	
4B4	F24	DC	F'24'	
4B8	F60	DC	F'60'	Also = X'CCCC003C'
4BC	F240	DC	F'240'	Also = X'CCCC00F0' = C'0'
4C0	F255	DC	F'255'	Also = X'CCC000FF'
4C4	F256	DC	F'256'	Also = X'CCC00100'
4C8	F4095	DC	F'4095'	Also = X'CCC00FFF'
4CC	F4096	DC	F'4096'	Also = X'CCC01000'
4D0	APTRLK	DC	V(DMKPTRLK)	Entry to lock a page in storage
4D4	NOADD	DC	X'FF000000'	Frequently used work value
4D8	X4OFFS	DC	X'40FFFFFF'	Frequently used work value
4DC	XRIGHT24	DC	X'00FFFFFF'	Isolate right 24 bits
4E0	XPAGNUM	DC	X'00FFFF000'	Isolate the page number
4E4	XRIGHT16	DC	X'0000FFFF'	Isolate the right 16 bits
4E8	AFREE	DC	V(DMKFREE)	Entry to allocate free storage
4EC	AFRET	DC	V(DMKFRET)	Entry to release free storage
4F0	AQCNWT	DC	V(DMKQCNWT)	Entry to write a terminal message
4F4	ADSPCH	DC	V(DMKDSPCH)	Entry to the VM/370 dispatcher
4F8	APTRAN	DC	V(DMKPTRAN)	Entry to the paging supervisor
4FC	X2048BND	DC	X'00FFFF800'	Locate a half-page boundary
500	PSBCLR2	DS	0F	Start of second area cleared by CP initialization (DMKCPI)
500	DUMPSAVE	DS	16F	Save area for dump routine
540	SIGSAVE	DS	16F	Save area for DMKEXT
580	LOKSAVE	DS	16F	DMKLCK save area
5C0	MFASAVE	DS	16F	Save area for malfunction alert
600	SWTHSAVE	DS	16F	DMKVMASW save area
640	LOCKSAV	DS	4F	LOCK macro save area
650	SVCREGS	DS	4F	SVC save area
660	PREFIXA	DC	F'0'	Prefix value of this processor
664	PREFIXB	DC	F'0'	Prefix value of other processor
668	PSACPYBP	DC	A(0)	Address of CPEXBLOK for switch SVC. Note that PSACPYBP is in absolute PSA.
66C	RESVD	DS	1F	Reserved for IBM use
670	WAITSTRT	DS	D	Timer value at start of wait
678	WAITEND	DS	D	Timer value at end of wait
680	PWTPAGES	DC	F'0'	Pages in PGWAIT at start of wait
684	ACTIVTRQ	DC	A(0)	Address of active transit queue

PSA

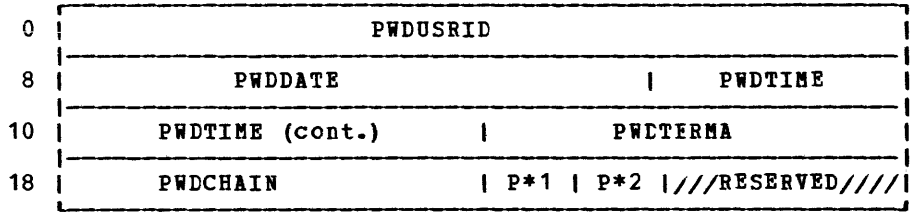
Hexadecimal Displacement	Field Name			Field Description, Contents, Meaning
688	EMSPEND	DC	F'0'	Emergency signal pending flags
	<u>Bits defined in EMSPEND</u>			
	EMSPQUI	EQU	X'80'	Quiesce pending
	EMSPEXT	EQU	X'40'	Extend pending
	EMSPSYNC	EQU	X'20'	Synchronization pending
	EMSPSHD	EQU	X'10'	Shutdown pending
	EMSPCLKC	EQU	X'08'	High order TOD synchronization pending
	EMSINQSC	EQU	X'01'	Processor is quiesced
68C	EMSREC	DC	F'0'	Emergency signal received flags
	<u>Bits defined in EMSREC</u>			
	EMSRQUI	EQU	X'80'	Quiesce request received
	EMSREXT	EQU	X'40'	Extend request received
	EMSRSYNC	EQU	X'20'	Synchronization request received
	EMSRSHD	EQU	X'10'	Shutdown request received
	EMSRCLKC	EQU	X'08'	High order TOD synchronization received
690	XCPEND	DC	F'0'	External call pending flags
	<u>Bits defined in XCPEND</u>			
	XCAPR	EQU	X'80'	Automatic processor recovery pending
	XCRES	EQU	X'40'	Resume request pending
	XCWAK	EQU	X'20'	Wakeup request pending
	XCDISP	EQU	X'10'	Dispatch request pending
694	IPUADDRX	DC	H'0'	P*7 Processor address of other processor
696	LPUADDR	DC	H'0'	P*8 Logical address of this processor
698	LPUADDRX	DC	H'0'	P*9 Logical address of other processor
69A	APSTATUS	DS	6X	Attached processor status bytes
		ORG	APSTATUS	
69A	APSTAT1	DC	X'00'	Attached processor status
	<u>Bits defined in APSTAT1</u>			
	APUOPER	EQU	X'80'	Attached processor operational
	PROCIO	EQU	X'40'	Processor has I/O capability
	APUNONLN	EQU	X'20'	System generated for attached processor mode but running in uniprocessor mode
	MPFEAT	EQU	X'10'	Multiprocessing feature is installed
	CPINITD	EQU	X'01'	System initialization complete
69B	APSTAT2	DC	X'00'	Second flag byte
	<u>Bits defined in APSTAT2</u>			
	CPMCHLK	EQU	X'10'	Machine check processing pending (for ECPS only)
	CPPTLBR	EQU	X'02'	PTLE required for processor
69C	CPTERMLK	DC	X'00'	DMKMCT system termination is in progress. Note that CPTERMLK is in absolute PSA.
69D	CPFRELK	DC	X'00'	Free storage extend pending. Note that CPFRELK is in absolute PSA.
69E	FRLKPROC	DC	X'00'	Logical processor identification for CPFRELK. Note that FRLKPROC is in absolute PSA.
69F	CPFRESW	DC	X'00'	DMKFRE must transfer execution to the attached processor. Note that CPFRESW is in absolute PSA.
6A0	AMCHAREA	DC	F'0'	Address of DMKMCH work area
6A4	SHRLKCNT	DC	F'0'	Count of times CPSHRLK is set (used to clear CPSHRLK)
6A8	PROBSTRT	DS	1D	Virtual machine time out queue at dispatch

Hexadecimal Displacement	Field Name				Field Description, Contents, Meaning
6B0	CHGREGS	DS	2F		Save area for charge synchronization
6B8	RESERVED	DS	2F		Reserved for IBM use
6C0	UNSHRVM	DC	A(0)		VMBLCK for pending DMKVMAPs call (attached processor only)
6C4	TRACPROC	DC	X'00'	P*10	Processor identifier for CF trace table entries
6C5	APSTAT	DS	3X		More attached processor status flag bytes
6C5	APSTAT3	DC	X'00'		Third attached processor status flag byte
	<u>Bits defined in APSTAT3</u>				
	CPSYSLK	EQU	X'80'		Other processor is spinning on lock
6C6	APSTAT4	DC	X'00'	P*11	Fourth attached processor status flag byte
	<u>Bits defined in APSTAT4</u>				
	CPLOKFL	EQU	X'80'		DMKLOCK enabled for external interrupts
	RECMODE	EQU	X'40'		Processor will record soft machine checks
	CPMCHSE	EQU	X'20'		Machine check processing pending (CPMCHSE is the replacement of CPMCHLK in APSTAT2)
	PROCSCHK	EQU	X'10'		TOD synchronous check received
	CPAPRPND	EQU	X'08'		Automatic processor recovery pending
	POFFLINE	EQU	X'04'		Vary processor function in use
6C8	RESERVE	DS	2F		Reserved for IBM use
6D0	STACKVM	DC	A(0)		R11 for dispatcher unstacking
6D4	UNSHRVM2	DC	A(0)		R2 value for pending VMAPs call (attached processor only)
6D8	ADMKCPE	DC	V(DMKCPE)		Address of DMKCPE for IPCS use
6DC	RESERVED	DS	5F		Reserved for IBM use
	PSECLR2	DS	0F		End of second area cleared by CP initialization (DMKCPI)
6F0	ALOKVM	DC	V(DMKLOKVM)		Entry to lock VMBLCK
6F4	RESERVED	DC	1F		Reserved for IBM use
6F8	ALOKSP	DC	V(DMKLOKSP)		Entry to spin on lock
6FA	AEXTSP	DC	V(DMKEXTSP)		Entry to signal processor routine
700	ATMRSN	DC	V(DMKTMRSN)		Entry to charge synchronization routine
704	RESERVED	DC	3F'0'		Reserved for IBM use
710	MONREGS	DC	16F		Register save area for Monitor call
750	LOKSAVE2	DC	16F		Save area for switching to virtual machine in DMKLOCK
	PSAEND	DS	0D		End of page 0 usage

PWDIBLOK

| PWDIBLOK: PASSWORD INVALID BLOCK

| The PWDIBLOK is used to retain information about invalid passwords supplied with LOGCN
| and AUTOLOG commands.



Hexadecimal Displacement	Field Name	Field Description, Contents, Meaning		
0	PWDUSRID DS CL8	Userid attempting LOGON or AUTCLCG		
8	PWDDATE DS CL6	Date (mmddy)		
E	PWDTIME DS CL8	Time (hh:mm:ss)		
14	PWDTERMA DS CL4	Terminal address		
18	PWDCHAIN DS F	Address of next PWDIBLOK		
1C	PWDINVCT DS 1X	P*1	Invalid password count	
1D	PWDFLAGS DS 1X	P*2	Flags	
<u>Bits defined in PWDFLAGS</u>				
	PWDLOG EQU X'80'	This block for LOGCN		
	PWDALOG EQU X'40'	This block for AUTCLCG		
1E	DS XL2	Reserved for IBM use		

Real I/O Control Blocks

In order to control the activity of the I/O devices of the system and schedule I/O requests upon them, I/O control uses several types of control blocks. These blocks can be separated into two basic types:

- Static blocks that describe the components of the I/O system.
- Dynamic blocks that represent active and pending requests for I/O operations.

The I/O components of the real system are described by one control block for each channel, control unit, and device available to the control program. Units present but not represented by control blocks are not available for either user-initiated or control program-initiated operations.

REAL CHANNEL CONTROL BLOCKS

For each channel attached to the system there exists a Real Channel Control Block (RCHBLOK) which contains:

- The channel portion of the address of its attached units,
- Status flags indicating the channel's availability for scheduling.
- A two-way queue anchor pointing to the list of I/O requests waiting to use the channel.

In addition, each RCHBLOK contains 32 halfword indexes, arranged in ascending address order, that represent the displacement into the Real Control Unit table of the control blocks for the control units attached to the channel. The 32 entries are required because the control unit address may be made up of five bits from the unit address. To locate the control block for a given unit:

1. Index into the table in the RCHBLOK a displacement equal to twice the control unit address.
2. Load the index value.
3. Add the value to the base address of the Real Control Unit Table.

REAL CONTROL UNIT BLOCKS

The Real Control Unit Table is composed of Real Control Unit Blocks (RCUBLOK), one for each control unit on the system. These blocks are similar to the RCHBLOK in that they contain the control unit portion of the address and status flags, and a pointer to a queue of I/O requests. In addition, the RCUBLOK contains a pointer to the RCHBLOK for the channel to which it is attached. The RCUBLOK contains a table of 16 halfword entries that represent the displacement into the Real Device Table of its attached devices. This table is referenced in the same manner as the table in the RCHBLOK.

REAL DEVICE CONTROL BLOCKS

Each device and 3270 remote communications line in the system is represented by a Real Device Control Block (RDEVBLCK), contains the device portion of the unit address and status flags similar to those in RCHBLOK and RCUBLOK. There is also a pointer for those operations that are waiting for the device to become available. Fields that appear in the RDEVBLCK and not in the other blocks include a pointer to the I/O request that is currently active on the device, SIC counts, and a pointer to error and sense information. The RDEVBLCK contains a pointer to the RCUBLOK for the control unit to which it is attached and fields of device dependent information which do not affect the operation of I/O control.

If the RDEVBLCK is associated with 3270 remote communications line, then the RDEVBLCK contains a pointer to NICBLOKS that represents the resources on that line.

INPUT/OUTPUT BLOCKS

I/O requests that are active in the system are represented by Input/Output Blocks (ICBLOK). There is one ICBLOK for each operation (that is, channel program) to be executed. The ICBLOK is constructed by the requesting task and contains such information as:

- The identity of the requester
- The address of the channel program to be executed

The address to which control is to be returned upon completion of the operation

In addition, the IOBLOK contains status flags that indicate the current state of the operation (such as, whether or not an error has occurred, if an error recovery procedure (ERP) is in control, and the condition returned from the SIO) and the CSW associated with the interrupt that signals the end of the operation. Since IOBLOKS are queued off various I/O control blocks, they also contain forward and backward queue pointers. DMKIOS builds in them the real device address of the unit on which the operation is started.

In general, the IOBLOK representing a given operation progresses through the system by being queued, in turn, from device, control unit, and channel blocks until a path is at last free to the device. A SIO is then issued. After the operation is complete, the IOBLOK is dequeued from the RDEVBLOK and stacked on a queue maintained in the dispatcher, DMKDSP. Each time the dispatcher is entered, the entries on the queue are unstacked and control is

passed to the point specified in the Interrupt Return Address (IOBIRA). After I/C control stacks the ICBLCK for the given task, it attempts to restart all of the components that have been freed by the completion of the operation.

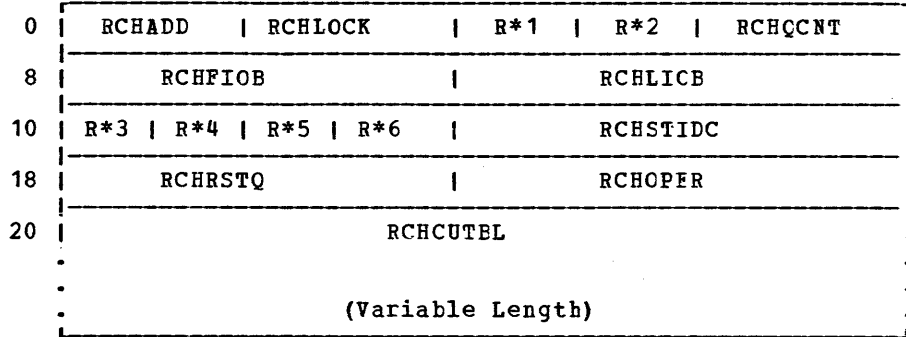
NETWORK INTERFACE CONTROL BLOCK

There is one Network Interface Control Block (NICBLOK) for each defined 370x and each resource attached to a 3270 bisynchronous line. The NICBLOK provides the correspondence between the line or device address and the physical resource connected to that line. This block not only defines the identity of the terminal type, line, or control unit but it also contains flags and status information pertaining to that resource. If the defined resource is a remote 3270 component, the NICBLOK contains the current line coordinates, polling and selection characters information as well.

The remainder of this section describes the real I/O control blocks.

RCHELCK: REAL CHANNEL BLOCK

RCHELCK contains status and type information for the specified channel. The linkage to I/O tasks operated on by that channel and to the control units attached to that channel is also maintained. The ARIODV field of the PSA points to the first RCHELCK, which is generated in contiguous storage.

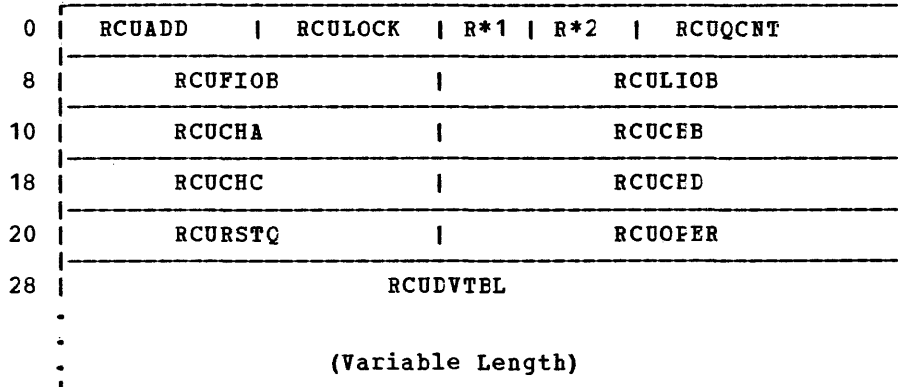


Hexadecimal Displacement	Field Name					Field Description, Contents, Meaning
0	RCHADD	DS	1H		Channel address	
2	RCHLOCK	DS	1H		Channel lock	
4	RCHSTAT	DS	1X	R*1	Channel status	
	<u>Bits defined in RCHSTAT</u>					
	RCHBUSY	EQU	X'80'		Channel busy	
	RCHSCED	EQU	X'40'		IOE scheduled on channel	
	RCHDED	EQU	X'01'		Channel dedicated	
5	RCHTYPE	DS	1X	R*2	Channel type	
	<u>Bits defined in RCHTYPE</u>					
	RCHSEL	EQU	X'80'		Selector channel	
	RCHBMX	EQU	X'40'		Elock multiplexer channel	
	RCHMPX	EQU	X'20'		Eyte multiplexer channel	
	RCH370	EQU	X'01'		S/370 type channel (S/370 I/C instruction support)	
6	RCHQCNT	DS	1H		Number of ICBLOCKS queued off channel	
8	RCHFIOB	DS	1F		Pointer to first IOBLOCK queued	
C	RCHLIOB	DS	1F		Pointer to last IOBLOCK queued	
10	RCHDTCK	DS	1X	R*3	Channel data check count	
11	RCHCCCK	DS	1X	R*4	Channel control check count	
12	RCHIFCC	DS	1X	R*5	Interface control check count	
13	RCHCHCK	DS	1X	R*6	Channel chaining check count	
14	RCHSTIDC	DS	1F		Result of STIDC instruction issued at CP initialization; if cc = 3, the content is X'FFFFFFFF'	
18	RCHRSTQ	DS	1F		Address of channel to be restarted	
1C	RCHOPER	DS	1F		IOELCK operational on channel time	
20	RCHCUTBL	DS	32H		Control units attached - RCUSTART index (The index values must be multiplied by 8 and added to the beginning of the RDEVELOK table (ARIODV).)	
	RCHSIZE	EQU	(*-RCHBLCK)/8		RCHELCK size in doublewords (X'0D')	

RCUBLOK

RCUBLOK: REAL CONTROL UNIT BLOCK

RCUBLOK provides control and status information on a defined real control unit. Linkages are provided to queued IOBLOKs. The ARIOCU field of the PSA points to the first RCUBLOK, which is generated in contiguous storage.



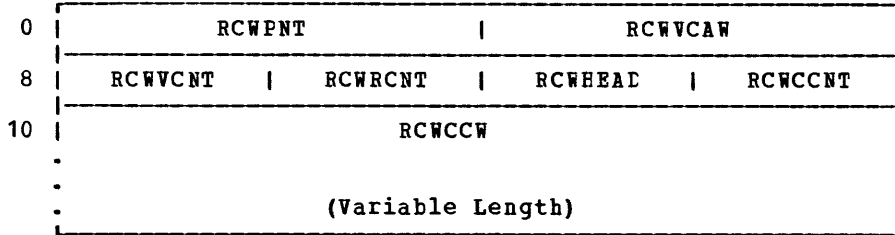
Hexadecimal Displacement	Field Name				Field Description, Contents, Meaning
0	RCUADD	DS	1H		Control unit address
2	RCULOCK	DS	1H		Control unit lock
4	RCUSTAT	DS	1X	R*1	Control unit status
<u>Bits defined in RCUSTAT</u>					
	RCUBUSY	EQU	X'80'		Control unit busy
	RCUSCED	EQU	X'40'		IOE scheduled on control unit
	RCUDISA	EQU	X'20'		Control unit disabled
	RCUCHAOF	EQU	X'08'		RCUCEA to RCHBLOK path is not available
	RCUCHBOK	EQU	X'04'		RCUCEB to RCHBLOK path is not available
	RCUCHCOF	EQU	X'02'		RCUCHC to RCHBLOK path is not available
	RCUCHDOF	EQU	X'01'		RCUCHD to RCHBLOK path is not available
5	RCUTYPE	DS	1X	R*2	Control unit type
<u>Bits defined in RCUTYPE</u>					
	RCUSHRD	EQU	X'80'		This control unit can be attached to only one subchannel
	RCUSUB	EQU	X'40'		This is a subordinate control unit
	RCU2703	EQU	X'03'		TCU is a 2703
	RCU2702	EQU	X'02'		TCU is a 2702
	RCU2701	EQU	X'01'		TCU is a 2701
6	RCUQCNT	DS	1H		Number of IOBLOKs queued off control unit
8	RCUFIOB	DS	1F		Pointer to first IOBLOK queued
C	RCULIOB	DS	1F		Pointer to last IOBLOK queued
10	RCUCHA	DS	1F		Pointer to RCHBLOK - path A
10	RCUPRIME	DS	1F	ORG RCUCHA	Pointer to the primary control unit

Hexadecimal Displacement	Field Name			Field Description, Contents, Meaning
14	RCUCHE	DS	1F	Pointer to RCHBLOK - path E
18	RCUCHC	DS	1F	Pointer to RCHBLOK - path C
1C	RCUCHD	DS	1F	Pointer to RCHBLOK - path D
20	RCURSTQ	DS	1F	Address of control unit to be restarted
24	RCUOPER	DS	1F	IOELCK operational on control unit time
28	RCUDVTBL	DS	16H	Devices attached - RDVSTART index (The index values must be multiplied by 8 and added to the beginning of the RDEVBLCK table (ARIODV).)
	RCUSIZE	EQU	(*RCUBLOK)/8	RCUBLOK size in doublewords (X'08')

RCWTASK

RCWTASK: TRANSLATED VIRTUAL I/O CCW

RCWTASK contains the virtual-to-real CCW translation and other data related to a virtual machine's I/O operation. A pointer is maintained to the virtual CCW operation. The first CCW-16 points to the beginning of RCWTASK.



Hexadecimal Displacement	Field Name			Field Description, Contents, Meaning
0	RCWPNT	DS	1F	Pointer to next RCWTASK
4	RCWCAW	DS	1F	Virtual address of CCW chain
8	RCWVCNT	DS	1H	Virtual CCW count
A	RCWRCNT	DS	1H	Real CCW count
C	RCWHEAD	DS	1H	RCWTASK header mark X'FFFF'
E	RCWCCNT	DS	1H	RCWTASK control word count
10	RCWCCW	DS	1D	One or more CCWs for device I/O
		ORG	RCWCCW	
10	RCWADDR	DS	1F	CCW data address
14	RCWFLAG	DS	1X	CCW flag bits
15	RCWCTL	DS	1X	CCW CP-control bits
	<u>Bits defined in RCWCTL</u>			
	RCWIO	EQU	X'80'	I/O data page locked
	RCWGEN	EQU	X'40'	CP-generated CCW
	RCWHMR	EQU	X'20'	DMKUNT must relocate home address/record R0
	RCWREL	EQU	X'10'	CCW address relocatable if CCWs moved
	RCWISAM	EQU	X'08'	ISAM modifying CCW
	RCW2311	EQU	X'04'	TYP2311T-E pseudo 2311 on 2314
	RCWINVL	EQU	X'02'	CCW operation code or address is invalid
	RCWSHR	EQU	X'01'	Shared user page was copied
16	RCWCNT	DS	1H	CCW byte count
		ORG	RCWADDR	
10	RCWCOMND	DS	1X	CCW command code

RDEVBLCK: REAL DEVICE BLOCK

RDEVBLCK is generated by the RDEV macro at system generation. There is one RDEVBLCK for each real device and one for each binary synchronous line. The block contains status and device parameters applicable to I/O instruction processing. The ARIOEV field of the PSA and the VDEVREAL field of the VDEVBLCK point to the first RDEVBLCK, which is generated in contiguous storage.

0	RDEVADD	RDEVLOCK	R*1	R*2	R*3	R*4
8	RDEVFIQB			RDEVLIQB		
10	RDEVCUA			RDEVQUB		
18	RDEVQUB					
20	RDEVIOCT			RDEVAIOB		
28	RDEVUSER		RDEVATT		RDEVCYL	
30	RDEVSR				RDEVLNKS	
38	RDEVTCTL (8 device dependent bytes)					
40	RDEVTMAT		R*5	R*6	R*7	R*8
48	RDEVIOER			RDEVCTRS		
50	RDEVNAME	///RDEVRSV1///		RDEVIOBL		

Hexadecimal Displacement	Field Name					Field Description, Contents, Meaning
0	RDEVADD	DS	1H		Device address	
2	RDEVLOCK	DS	1H		Device lock	
4	RDEVSTAT	DS	1X	R*1	Device status	
<u>Bits defined in RDEVSTAT</u>						
	RDEVBUSY	EQU	X'80'		Device busy	
	RDEVSCED	EQU	X'40'		IOB scheduled on device	
	RDEVDISA	EQU	X'20'		Device disabled (offline)	
	RDEVRSVD	EQU	X'10'		Device reserved	
	RDEVIRM	EQU	X'08'		Device in intensive error recording mode	
	RDEVNRDY	EQU	X'04'		Device intervention required	
	RDEVWAI	EQU	X'02'		GRAF - IOBLCK pending, queue requests	
	RDEVDED	EQU	X'01'		Dedicated device (attached to a virtual machine)	

RDEVELOK

Hexadecimal Displacement	Field Name	Field Description, Contents, Meaning		
5	RDEVFLAG DS 1X R*2	Device flags, device dependent		
<u>Bits defined in RDEVFLAG</u>				
	RDEVSKUP EQU X'80'	DASD - ascending order seek queuing		
	RDEVPREF EQU X'40'	DASD - volume preferred for paging		
	RDEVSYS EQU X'20'	DASD - volume attached to system		
	RDEVOWN EQU X'10'	DASD - CP-owned volume		
	RDEVMOUT EQU X'08'	DASD - volume mounted, not attached		
	RDEV333V EQU X'04'	DASD - volume dedicated as 3330V		
	RDEVSEL EQU X'02'	DASD - device selected for MSS mount		
	RDEVPSUP EQU X'80'	Console - terminal has print suppress		
	RDEVPREP EQU X'40'	Console - terminal executing PREPARE command		
	RDEVACTV EQU X'20'	Console - ICBLOCK pending; queue request		
	RDEVIDNT EQU X'10'	Console - 2741 terminal code identified		
	RDEVENAB EQU X'08'	Console - device is enabled		
	RDEVHIO EQU X'04'	Console - next interrupt from a Halt I/O		
	RDEVDISB EQU X'02'	Console - device is to be disabled		
	RDEVPMDD EQU X'01'	Console - 370x NCP resource in EP mode		
	RDEVDRAN EQU X'80'	Spooling - device output drained		
	RDEVTERM EQU X'40'	Spooling - device output terminated		
	RDEVACNT EQU X'20'	Spooling - device busy with accounting		
	RDEVSPAC EQU X'10'	Spooling - force printer to single space		
	RDEVRSTR EQU X'08'	Spooling - restart current file		
	RDEVBACK EQU X'04'	Spooling - backspace the current file		
	RDEVSEP EQU X'02'	Spooling - print/punch job separator		
	RDEVLOAD EQU X'01'	Spooling - UCS buffer verified		
	RDEVLNCP EQU X'80'	Special - network control program active		
	RDEVLCEP EQU X'40'	Special - 270x Emulation program active		
	RDEVSLOW EQU X'20'	Special - 370x in buffer slowdown mode		
	RDEVAUTO EQU X'10'	Special - automatic dump/load enabled		
	RDEVWAIT EQU X'08'	Special - ICBLOCK pending; queue requests		
	RDEVPLN EQU X'04'	Special - emulator lines in use by system		
	RDEVRCVY EQU X'02'	Special - automatic dump/load process active		
	RDEVTBTU EQU X'01'	Special - BTU trace requested		
6	RDEVTPC DS 1X R*3	Device type class (see "Appendix A. CP and RSCS Equate Symbols")		
7	RDEVTYPE DS 1X R*4	Device type (see "Appendix A. CP and RSCS Equate Symbols")		
8	RDEVFIOB DS 1F	Pointer to first ICBLOCK queued		
C	RDEVLIOB DS 1F	Pointer to last IOELCK queued		
10	RDEVCUA DS 1F	Pointer to RCUBLOCK - interface A		
14	RDEVSUB DS 1F	Pointer to RCUBLOCK - interface B		
18	RDEVQUED DS 1D	IOELCK queued time - TOD clock units		
20	RDEVIOCT DS 1F	Device I/O count		
24	RDEVAIOB DS 1F	Active IOELCK		
28	RDEVUSER DS 1F	Pointer to VMBLOCK of dedicated user		
2C	RDEVATT DS 1H	Attached virtual address		
2E	RDEVCYL DS 1H	DASD - current cylinder location		
30	RDEVSER DS CL6	Device volume serial number		
36	RDEVLNKS DS 1H	DASD - number of links to this disk		
38	RDEVCTL DS 8X	8 device-dependent terminal control bytes		
40	RDEVTMAT DS 1F	Device attached time - TCD clock word 0		
44	RDEVQCNT DS 1X R*5	Number of queued IOELCKs		

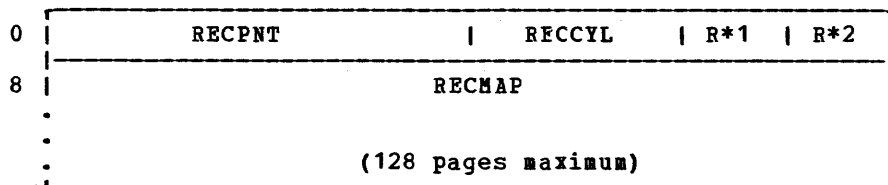
Hexadecimal Displacement	Field Name			Field Description, Contents, Meaning
45	RDEVSTA2 DS	1X	R*6	Device status (byte 2)
	<u>Bits defined in RDEVSTA2</u>			
	RDEVRACT EQU	X'80'		Active device is being reset
	RDEVBUCH EQU	X'40'		Device is busy with the channel
	RDEVCONC EQU	X'20'		Contingent connection present
	RDEVDROP EQU	X'10'		Logdrop or loghold indicated
	RDEVALT EQU	X'08'		Alternate path device
	RDEVSIBC EQU	X'04'		Attention signal during active I/O
	RDEVPURG EQU	X'02'		3800 - Purge files in error
	RETRYSW EQU	X'01'		Retry count switch
46	RDEVMDL DS	1X	R*7	Device model number
47	RDEVFTR DS	1X	R*8	Device feature code
48	RDEVIOER DS	1F		Pointer to IOERBLCK for last CP error
4C	RDEVCTRS DS	1F		Pointer to error counter control block
50	RDEVNAME DS	1H		Real device name
52	RDEVRSV1 DS	1H		Reserved for IBM use
54	RDEVIOBL DS	1F		IOBLCK queue lock
	RDEVSIZE EQU	(*-RDEVBLCK)/8		RDEVBLCK size in doublewords (X'0B')
	<u>For CP-owned Devices</u>			
	ORG	RDEVUSER		
28	RDEVALLN DS	1F		Anchor for ALOCBLCK chain for this device
2C	RDEVCODE DS	1H		Device code - SYSOWNED index
	ORG	RDEVTCTL		
38	RDEVPAGE DS	1F		Anchor for RECBLOCK chain for paging
3C	RDEVRECS DS	1F		Anchor for RECBLOCK chain for spooling
40	RDEVPNT DS	1F		Pointer to next RDEVBLCK for allocation
	<u>For Slotted 2301 Paging Drums</u>			
	ORG	RDEVRECS		
3C	RDEVDCtrl DS	1F		Pointer to DRUMTABL control block
	<u>For Graphic Devices</u>			
	ORG	RDEVCYL		
2E	RDEVCORD DS	1H		Current line coordinates
30	RDEVGRTB DS	1F		Address of table of CCWs and data streams
	<u>For Spooling Unit Record Devices</u>			
	ORG	RDEVQUED		
18	RDEVSPL DS	1F		Pointer to active RSPLCTL block
1C	RDEVCLAS DS	4C		Device class(es)
	ORG	RDEVUSER		
28	RDEVDELP DS	A		Anchor for delay purge queue
2C	RDEVCURP DS	1X		Current page length in half inches
2D	DS	1X		Reserved for IBM use
2E	RDEVMAXP DS	1X		Maximum number of entries in delay purge queue
2F	RDEVFSEP DS	1X		Function control block for separator page (6, 8, or 12)
30	RDEVXSEP DS	CL4		Name of character arrangement table for the separator page
34	RDEVEXTN DS	1A		Pointer to the 3800 extension
38	RDEVIMAG DS	CL8		Name of current image library
40	RDEVVOLY DS	CL4		Name of current forms overlay
	<u>For Terminal Devices</u>			
	ORG	RDEVQUED		
18	RDEVCON DS	1F		Pointer to CONTASK list
1C	RDEVAIRA DS	1F		Attention interrupt return address
	ORG	RDEVTCTL		
38	RDEVRCNT DS	1H		Start/stop line retry count

RDEVBLK

Hexadecimal Displacement	Field Name		Field Description, Contents, Meaning
3A	RDEVTFLG	DS 1X	Additional terminal flags
	<u>Bits defined in RDEVTFLG</u>		
	RDEVLOG	EQU X'80'	TERM and GRAF - Logon process has been initiated
	RDEVREST	EQU X'40'	TERM - Terminal in reset process
	RDEVATOF	EQU X'20'	TERM - Suppress attention signal
	RDEVLOG	EQU X'80'	GRAF - Logon process has been initiated
	RDEVMORE	EQU X'40'	GRAF - Screen full; more data waiting
	RDEVRUN	EQU X'20'	GRAF - Screen in running status
	RDEVREAD	EQU X'10'	GRAF - Read pending for screen input
	RDEVCPNA	EQU X'08'	GRAF - Last input not accepted
	RDEVTRQ	EQU X'04'	GRAF - Timer request pending
	RDEVCTL	EQU X'02'	GRAF - Control function interrupt pending
	RDEVHOLD	EQU X'01'	GRAF - Screen full; in hold status
3B	RDEVGRTY	DS 1X	Display alternate screen size index
3C	RDEVLEN	DS 1X	Device line length
3D	RDEVATNC	DS 1X	Device attention count
3E	RDEVBASE	DS 1H	370x base address for emulator line
3F	RDEVRSV3	DS 1X	Reserved for IBM use
		ORG RDEVMDL	
46	RDEVTMCD	DS 1X	Terminal code
	<u>Bits defined in RDEVTMCD</u>		
	RDEVTEXT	EQU X'20'	3270 Text character set
	RDEVUSC8	EQU X'10'	ASCII-8 level keyboard
	RDEVAPLC	EQU X'0C'	APL Correspondence keyboard
	RDEVAPLP	EQU X'08'	APL PTTC/EBCD keyboard
	RDEVCORR	EQU X'04'	Correspondence keyboard
	RDEVPTTC	EQU X'00'	PTTC/EBCD keyboard
47	RDEVSADN	DS 1X	Terminal set-address number
	<u>For Real 3704/3705 Communications Controller</u>		
		ORG RDEVAIRA	
1C	RDEVEPDV	DS 1F	Start of free RDEVBLK list for EP line
		ORG RDEVCYL	
2E	RDEVMAX	DS 1H	Highest valid NCP resource name
30	RDEVNCP	DS CL8	Reference name of active 3704 NCP
38	RDEVNICL	DS 1F	Pointer to network control list
3C	RDEVCKPT	DS 1F	Pointer to CKPBLOK for re-enable
	<u>For 3270 Remote Support</u>		
		ORG RDEVNCP	
30	RDEVBSC	DS 1F	Pointer to binary synchronus control block
34	RDEVPDLY	DS 1F	Poll delay timer interval

RECBLOK: DASD PAGE (SLOT) ALLOCATION BLOCK

RECBLOK maintains the correlation of DASD storage pages to a specific cylinder location. Also maintained is a bit map to indicate the page slots available for data page storage. The RDEVRECS field and the RDEVPAGE field of the RDEVELCK point to RECBLOK.



Hexadecimal Displacement	Field Name					Field Description, Contents, Meaning
0	REC PNT	DS	1F			Pointer to next RECBLOK on chain
4	RECCYL	DS	1H			Cylinder address for pages in this block
6	RECUSED	DS	1X	R*1		Number of pages currently in use
7	REC MAX	DS	1X	R*2		Maximum number of pages available
8	RECMAP	DS	1L			Page allocation bit map (128 pages maximum)

Bits defined in RECMAP

0 - Page is available
 1 - Page has been assigned

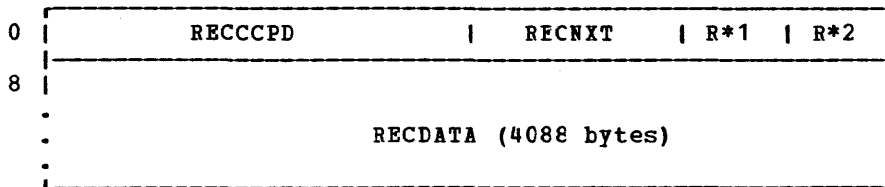
Note: Although the size of RECMAP is fixed, the maximum number of pages available on a cylinder is device dependent. Bits corresponding to pages not physically present on a cylinder are set to 1.

RECSIZE EQU (*-RECBLOK)/8 RECBLOK size in doublewords

RECPAG

RECPAG: ERROR RECORDING PAGE RECORD

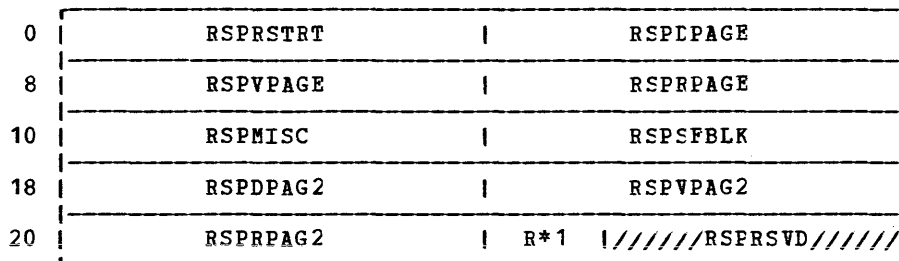
RECPAG retains up to 4K bytes of error recording data for eventual placement on the specified error recording cylinder.



Hexadecimal Displacement	Field Name	Field Description, Contents, Meaning
0	RECCCPD DS 4X	CCPD of the record
4	RECNEXT DS 2X	Displacement to next error record
6	RECFLAG1 DS 1X R*1	Record usage flags
	<u>Bits defined in RECFLAG1</u>	
	RECPAGIU EQU X'80'	Page contains valid data
	RECPAGFR EQU X'40'	Page is cleared
	RECPAGFL EQU X'20'	Page is full of error records
	RECPAGER EQU X'10'	Next page is unreadable (I/C error)
	RECPAGFA EQU X'08'	Page contains page frame records
7	RECFLAG2 DS 1X R*2	Record format flag
	<u>Bits defined in RECFLAG2</u>	
	RECPAGFM EQU X'80'	Set in page 1 of a recording cylinder when the cylinder is being formatted. This flag bit is reset when all pages are cleared.
	RECPAGDN EQU X'00'	Cylinder formatted
8	RECDATA DS 4088C	Data area
	RECPAGSZ EQU (*-RECPAG)/8	Size of page in doublewords (X'512')

RSPCTL: REAL SPOOLING CONTROL BLOCK

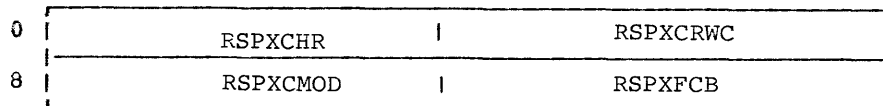
RSPCTL is used in conjunction with SFBLK for processing closed spool files. The RDEVSP field of RDEVBLK points to RSPCTL.



Hexadecimal Displacement	Field Name		Field Description, Contents, Meaning
0	RSPRSTRT DS	1F	Restart CAW - CCW address
4	RSPDPAGE DS	1F	DASE location (DCHR) of current page buffer
8	RSPVPAGE DS	1F	Virtual address of page buffer
C	RSPRPAGE DS	1F	Real address of page buffer
10	RSPMISC DS	1F	Use varies according to caller
14	RSPSFBLK DS	1F	Pointer to SFBLK for file
18	RSPDPAG2 DS	1F	DASE location of second page buffer if any
1C	RSPVPAG2 DS	1F	Virtual address of second page buffer if any
20	RSPRPAG2 DS	1F	Real address of second page buffer if any
24	RSPFLAG1 DS	1X	Real spooling control flag byte
	<u>Bits defined in RSPFLAG1</u>		
	RSPBF1IO EQU	X'80'	I/O pending on buffer 1
	RSPBF2IO EQU	X'40'	I/O pending on buffer 2
	RSPBF1VL EQU	X'20'	Buffer 1 is full of data to print
	RSPBF2VL EQU	X'10'	Buffer 2 is full of data to print
25	RSPRSVD DS	3X	Reserved for IBM use
	RSPSIZE EQU	(*-RSPCTL)/8	Size in doublewords (X'05')

RSPXBLOK: REAL DEVICE EXTENSION BLOCK

RSPXBLOK is used with the RDEVBLK to accommodate the 3800 requirements.

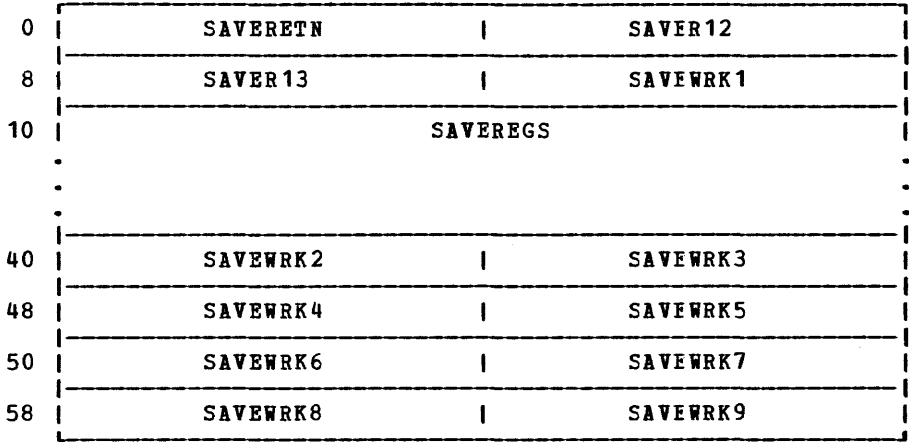


Hexadecimal Displacement	Field Name		Field Description, Contents, Meaning
0	RSPXCHR DS	CL4	Character arrangement table now in 3800
4	RSPXCRWC DS	4X	Current contents of WCGMs
8	RSPXCMOD DS	CL4	Copy module now in 3800
C	RSPXFCB DS	CL4	FCB now in 3800
	RSPXSIZE EQU	(*-RSPXBLOK)/8	RSPXBLOK size in doublewords (X'01')

SAVEAREA

SAVEAREA

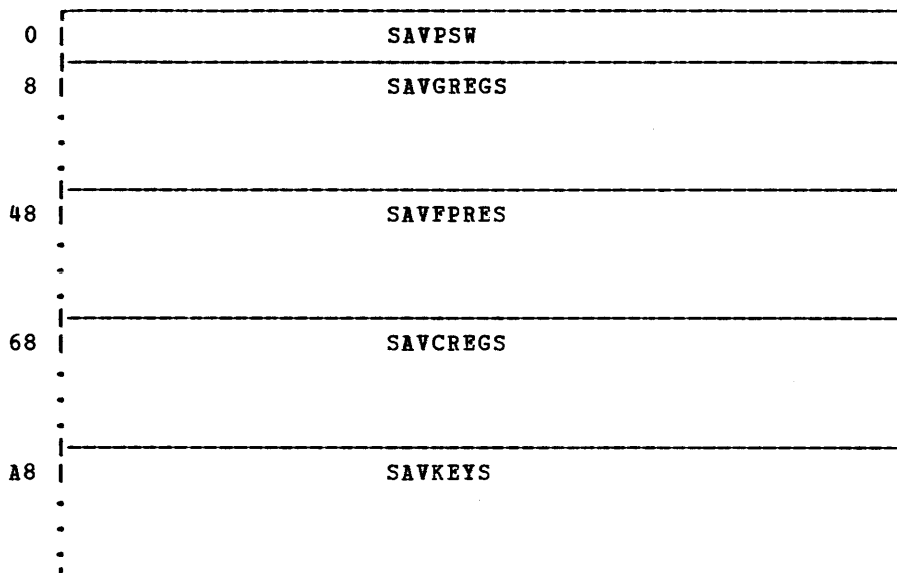
SAVEAREA is used to save the registers of a module when that module has called another module. Register 13 points to SAVEAREA.



Hexadecimal Displacement	Field Name			Field Description, Contents, Meaning
0	SAVERETN DS	1F		Active SAVEAREA (caller's return address)
<u>Redefinition of SAVERETN To Include Processor Address</u>				
		ORG	SAVERETN	
0	SAVEPROC DS	1X		Active SAVEAREA -- processor address
1	SAVERTN DS	3X		Active SAVEAREA -- return address
		ORG	SAVERETN	
0	SAVENEXT DS	1F		Inactive SAVEAREA (next SAVEAREA address)
4	SAVER12 DS	1F		Caller's base (R12)
8	SAVER13 DS	1F		Caller's SAVEAREA (R13)
C	SAVEWRK1 DS	1F		Called routine's work area
10	SAVEREGS DS	12F		Caller's registers (R0 to R11)
		ORG	SAVEREGS	
10	SAVER0 DS	1F		
14	SAVER1 DS	1F		
18	SAVER2 DS	1F		
1C	SAVER3 DS	1F		
20	SAVER4 DS	1F		
24	SAVER5 DS	1F		
28	SAVER6 DS	1F		
2C	SAVER7 DS	1F		
30	SAVER8 DS	1F		
34	SAVER9 DS	1F		
38	SAVER10 DS	1F		
3C	SAVER11 DS	1F		
40	SAVEWRK2 DS	1F	}	Called routine's work area (8 fullwords)
44	SAVEWRK3 DS	1F		
48	SAVEWRK4 DS	1F		
4C	SAVEWRK5 DS	1F		
50	SAVEWRK6 DS	1F		
54	SAVEWRK7 DS	1F		
58	SAVEWRK8 DS	1F		
5C	SAVEWRK9 DS	1F		
	SAVESIZE EQU	(*-SAVEAREA)/8 Size in doublewords (X'0C')		

SAVTABLE: FIRST PAGE ON SAVED SYSTEM DASD

SAVTABLE is used in the initial program loading of saved virtual machine named systems. It is created by the name system generation process (SAVESYS macro/SAVESYS command).

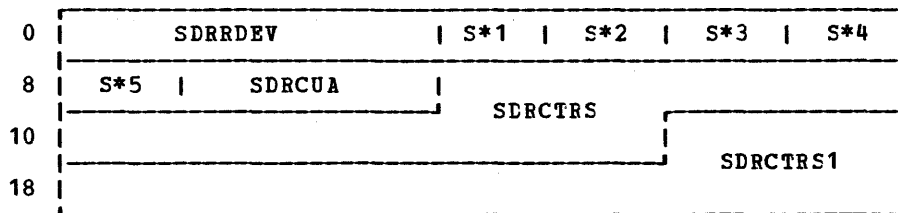


Hexadecimal Displacement	Field Name			Field Description, Contents, Meaning
0	SAVPSW	DS	1D	PSW of virtual machine at SAVSYS time
8	SAVGREGS	DS	16F	General registers
48	SAVFPRES	DS	4D	Floating-point registers
68	SAVCREGS	DS	16F	Control registers
A8	SAVKEYS	DS	1H	Two-byte entry for each saved page containing storage keys for each page

SDRBLOK

SDRBLOK: STATISTICAL DATA RECORDING BLOCK

Contains counters to record temporary errors on a given I/O device.



Hexadecimal Displacement	Field Name				Field Description, Contents, Meaning
0	SDRRDEV	DS	1F		Address of associated RDEVELCK
4	SDRFLAGS	DS	1X	S*1	SDRELOK flags
<u>Bits defined in SDRFLAGS</u>					
	SDRSHT	EQU	X'80'		Short OBRRECEN to be written
	SDRRECD	EQU	X'40'		Long OBRRECEN to be recorded on counter overflow
	SDRMAX	EQU	X'20'		Maximum numbers of SDR counters handled
5	SDRPRMCT	DS	1X	S*2	Parameter list counter
6	SDRFLCT	DS	1X	S*3	Full byte counter
7	SDROVFWK	DS	1X	S*4	Statistical update work byte
8	SDRLNGTH	DS	1X	S*5	Length (bytes) of SDR counters
9	SDRCUA	DS	3X		Primary CUA of device being used
	SDRBSIZE	EQU	(*-SDRBLOK)		SDRELOK base size, in bytes
C	SDRCTRS	DS	10X		SDR error counters
	SDRSIZE	EQU	(*-SDRELOK+7)/8		Size in doublewords (X'03')
16	SDRCTRS1	DS	10X		Additional SDR error counters for devices that use more than 10 SDR counters.
	SDRSIZE1	EQU	(*-SDRBLOK)/8		Size in doublewords (X'04')

SEGTABLE: TRANSLATION SEGMENT TABLE

SEGTABLE is used in conjunction with the page table (PAGTABLE) and swap table (SWPTABLE) by the page management routines. The VMSEG field of the VMELOK points to SEGTABLE.



Hexadecimal Displacement	Field Name				Field Description, Contents, Meaning
0	SEGPAGE	DS	1F		Pointer to page table (PAGTABLE)
0	SEGPLN	DS	BL.4	S*1	Page table length (Total pages - 1) (in left half of byte)
	<u>Bits defined in SEGPAGE+3</u>				
	SEGINV	EQU	X'01'		Segment invalid
	SEGMIG	EQU	X'10'		Segment migrated if pointer is equal to 0
	SEGENQ	EQU	X'40'		Segment is enqueued if pointer is equal to 0

SFBLOK

SFBLOK: SPOOL FILE BLOCK

SFBLOK retains all the information relating to a spool file. A pointer provides a linkage to the next SFBLOK in the chain. The VSPSFBLK field of VSPICTL and the RSPSFBLK field of RSPLCTL point to the SFBLOK.

0	SFBPNT		SFBSTART
8	SFBUSER		
10	SFBORIG		
18	SFBRECNO		SFERECSZ SFBFILID
20	S*1 S*2 SFBMISC1		SFERECS
28	SFBFNAME		
34	SFBFTYPE		
40	SFBDATE		
48	SFBTIME		
50	SFBLAST		SFECOPY S*3 S*4
58	SFBDIST		
60	SFBFLASH		S*5 /////////SFERSVD/////////

Hexadecimal Displacement	Field Name			Field Description, Contents, Meaning
0	SFBPNT	DS	1F	Pointer to next SFBLOCK
4	SFBSTART	DS	1F	EASE location (CCPD) of first page buffer
8	SFBUSER	DS	CL8	VMUSER identification of file owner
10	SFBORIG	DS	CL8	VMUSER identification of file origin
18	SFBRECNO	DS	1F	Number of data records in file
1C	SFBRECSZ	DS	1H	Logical record size - excluding CCWs
1E	SFBFILID	DS	1H	Binary system file number
20	SFBFLAG	DS	1X	S*1 SFBLOCK control bits

Bits defined in SFBFLAG

SFBINUSE	EQU	X'80'	File being processed
SFBRECOK	EQU	X'40'	Allocation records complete
SFBUHOLD	EQU	X'20'	File in user hold status
SFBDUMP	EQU	X'10'	File is a CP system dump
SFBOPEN	EQU	X'08'	Input file has been opened
SFBSHOLD	EQU	X'04'	File in system hold status
SFBEOF	EQU	X'02'	Input file has reached ECF
SFBRECER	EQU	X'01'	SFEREC chain incomplete

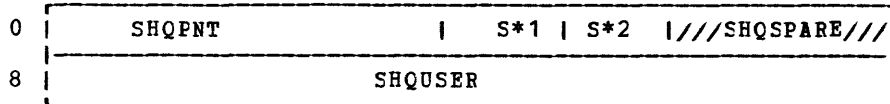
21	SFBFTYPE	DS	1X	S*2 Device type for spool output
22	SFBMISC1	DS	1H	Use varies according to caller
24	SFBRECS	DS	1F	Pointer to RECBLOCKS for active file
28	SFBFNAME	DS	CL12	Filename

Hexadecimal Displacement	Field Name			Field Description, Contents, Meaning
34	SFBFTYPE	DS	CL12	Filetype
40	SFBDATE	DS	CL8	Creation date of spool file
48	SFBTIME	DS	CL8	Creation time of spool file
50	SFBLAST	DS	1F	DASD location (CCPD) of last page buffer
54	SFBCOPY	DS	1H	Number of copies requested
56	SFBCLAS	DS	1X	S*3 Spool file class character
57	SFBFLAG2	DS	1X	S*4 SFBLOCK flag (byte 2)
<u>Bits defined in SFBFLAG2</u>				
	SFBHOLD	EQU	X'80'	Save input file, or hold output file
	SFBNOHLD	EQU	X'40'	Delete input file, or do not hold output file
	SFBFLNMT	EQU	X'20'	File not empty if on
<u>Note:</u> SFBHOLD and SFBNOHLD override options in VDEVBLCK.				
	SFBREQUE	EQU	X'20'	Requeue spool file
	SFBRSTRT	EQU	X'10'	Restart in progress
	SFBTICER	EQU	X'08'	Buffer TIC error
	SFBPURGE	EQU	X'04'	Purge open spool file
	SFBFIRST	EQU	X'02'	Indicate first page written
	SFBMON	EQU	X'01'	Monitor spool file
58	SFBDIST	DS	CL8	Distribution code
60	SFBFLASH	DS	CL4	Overlay name for 3800 is flashing
64	SFBSTCPY	DS	1X	S*5 Current starting copy number
65	SFBRSVD	DS	XL3	Reserved for IBM use
	SFBFSIZE	EQU	(* - SFBLOCK) / 8	Size in doublewords (X'0E')

SHQBLOK

SHQBLOK: SPOOL HOLD QUEUE BLOCK

SHQBLOK provides a holding function for identified spool files to prevent output to another user or to a real output device.



Hexadecimal Displacement	Field Name				Field Description, Contents, Meaning
0	SHQPNT DS	1F			Address of next SHQBLOK
4	SHQFLAGS DS	0CL4			Length
4	SHQUHOLD DS	1X	S*1		User USER HOLD flag byte
5	SHQSHOLD DS	1X	S*2		User SYSTEM HOLD flag byte
<u>Bits defined in SHQUHOLD and SHQSHOLD</u>					
	TYPprt EQU	Printer			See "Appendix A. CP and RSCS Equate Symbols"
	TYPpUN EQU	Punch			See "Appendix A. CP and RSCS Equate Symbols"
6	SHQSPARE DS	2X			Reserved for IBM use
8	SHQUSER DS	CL8			Virtual machine user identification of file owner
	SHQBSIZE EQU	(*-SHQBLOK)/8			Size in doublewords (X'02')

SHRTABLE: NAMED-SHARED SEGMENT SYSTEMS TABLE

SHRTABLE contains pointers to the segment locations of named systems for both the shared and nonshared user. This block is used in paging, IPL, and virtual machine assist operations. PAGTABLE-8 points to SHRTABLE.

0	S*1	SHRFPNT	SHRFPNT
8	SHRNAME		
10	SHRTSIZE	SHRUSECT	SHRSEGCT
18	SHRSEGNM		SHRPAGE

Hexadecimal Displacement	Field Name				Field Description, Contents, Meaning
0	SHRFPNT DS	1F			Pointer to next SHRTABLE
0	SHRFLAG DS	1X	ORG SHRFPNT	S*1	Shared segment flag byte
1	SHRFPNT DS	3X			Pointer to next SHRTABLE
	<u>Bits defined in SHRFLAG</u>				
	SHRNOPT EQU	X'40'			Nonprotected shared segments
4	SHRBPNT DS	1F			Pointer to previous SHRTABLE
8	SHRNAME DS	CL8			Name of saved system
10	SHRTSIZE DS	1H			Size of SHRTABLE in doublewords
12	SHRUSECT DS	1H			Number of users using this segment name
14	SHRSEGCT DS	1F			Number of shared segments
18	SHRSEGNM DS	1F			Contains shared segment numbers; up to four segment numbers per word
1C	SHRPAGE DS	1F			Pointers to each of the shared page tables. There is one word for each shared segment. The entry is the same as S*1 SEGPAGE in "SEGTABLE: Segment Table."

Note: In attached processor (AP) mode, there will be two sets of page tables (PAGTABLE) for each shared segment; one for the main processor, one for the attached processor. SHRPAGE will point to the page tables for the main processor. The page tables for the attached processor will be at a fixed displacement (PAGBMP) from those of the main processor.

SPLINK

SPLINK: SPOOL PAGE BUFFER LINKAGE BLOCK

SPLINK resides in auxiliary storage and contains one page (4096 bytes) of unit record spool information consisting of data and all required CCWs. The RSPVPAGE field and the RSPRPAGE field in RSPLCTL point to SPLINK.

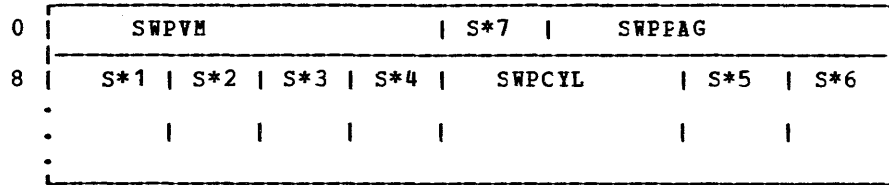
0	SPNXTPAG		SPPREPAG
8	SPRMISC		SPRECNUM
10	Spool Buffer Data Area (4032 bytes) ¹		
FD0	SPCHAR		
FD8	SPRSVD1		
FE0	SPFCBS		SPCMOD
	SPRSVD2		
FF8		S*1 S*2	SPRSVD3

Hexadecimal Displacement	Field Name	Field Description, Contents, Meaning		
0	SPNXTPAG DS 1F	DASD location (DCHR) of next page buffer		
4	SPPREPAG DS 1F	DASD location (DCHR) of previous page buffer		
8	SPRMISC DS 1F	Use varies according to caller		
C	SPRECNUM DS 1F	Number of data records in buffer		
	SPSIZE EQU (*-SPLINK)	Size in bytes (X'10')		
	ORG SPLINK+4096-48	End of first DASD buffer		
FD0	SPCHAR DS CL4	3800 character arrangement table for file		
FD4	SPRSVD1 DS CL12	Reserved for IBM use		
FE0	SPFCB DS CL4	3800 function control block for file		
FE4	SPCMOD DS CL4	3800 copy module for file		
FE8	SPRSVD2 DS CL12	Reserved for IBM use		
FF4	SPFLSHC DS 1X S*1	3800 flash count		
FF5	SPFLAG1 DS 1X S*2	3800 flag byte		
	<u>Bits defined in SPFLAG1</u>			
	SPCOPYFG EQU X'80'	If flag is set, multiple copies can be done in one transmission		
FF6	SPRSVD3 DS 2X	Reserved for IBM use		
FF8	SPFILID DS 1H	File identifier used for verification		
FFA	SPTIME DS CL6	SFETIME -- used for verification		
	SPENDSIZ EQU *-SPCHAR	End-of-buffer size in bytes		

¹First DASD buffer only. All others comprise 4080 bytes.

SWPTABLE: SWAP TABLE FOR VIRTUAL MACHINE PAGING

SWPTABLE is used in conjunction with the page table (PAGTABLE) and the segment table (SEGTABLE) by the CP page management routines for relating the virtual storage to DASD slots and real storage. The PAGSWP field of the PAGTABLE points to SWPTABLE.



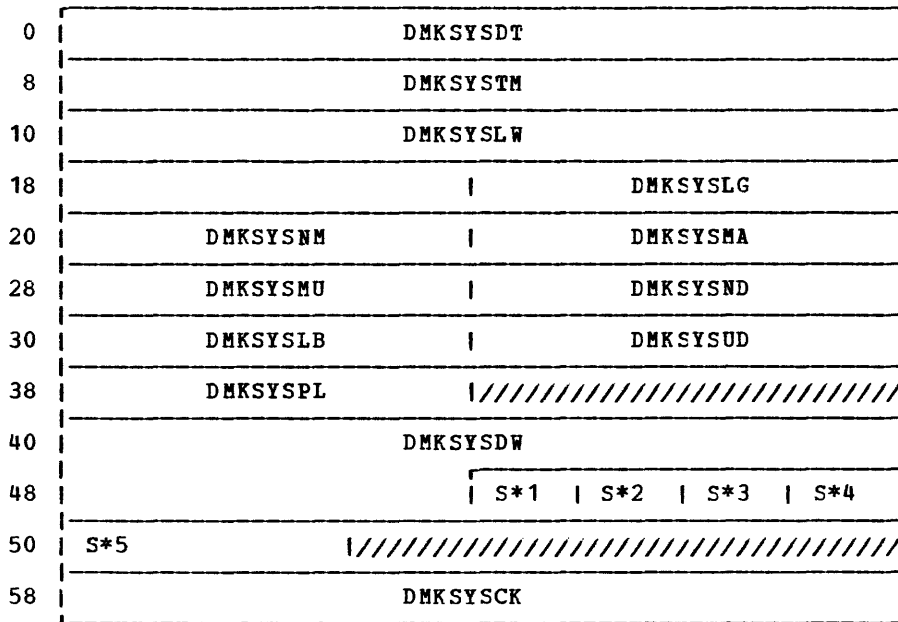
Hexadecimal Displacement	Field Name					Field Description, Contents, Meaning
0	SWPVM	DS	1F		Pointer to VMBLCK	
4	SWPFLAG2	DS	1X	S*7	Swap table FLAG2 bits	
	<u>Bits defined in SWPFLAG2</u>					
	SWPAPP	EQU	X'80'		Attached processor's shared page table and swap table	
4	SWPPAG	DS	1F		Pointer to page table	
8	SWPFLAG	DS	1X	S*1	Swap table flag bits	
	<u>Bits defined in SWPFLAG</u>					
	SWPTRANS	EQU	X'80'		Page in transit	
	SWPRECMP	EQU	X'40'		Page permanently assigned	
	SWPALLOC	EQU	X'20'		Page enqueued for allocation	
	SWPSHR	EQU	X'10'		Page shared	
	SWPREF1	EQU	X'08'		First half-page referenced	
	SWPCHG1	EQU	X'04'		First half-page changed	
	SWPREF2	EQU	X'02'		Second half-page referenced	
	SWPCHG2	EQU	X'01'		Second half-page changed	
9	SWVPAGE	DS	1X	S*2	Virtual page number within the segment	
A	SWPKEY1	DS	1X	S*3	Virtual storage key, first 2048 bytes	
B	SWPKEY2	DS	1X	S*4	Virtual storage key, second 2048 bytes	
C	SWPCYL	DS	1H		DASD cylinder address	
E	SWDPAGE	DS	1X	S*5	DASD page number on cylinder	
F	SWPCODE	DS	1X	S*6	RDEVELOK device code (The device code is used as an index into the list of CP-owned paging volumes pointed to by DMKSYSOW)	

Note: For each SWPTABLE there is only one doubleword that consists of SWPVM and SWPPAG followed by 16 entries (one for each PAGTABLE entry) that consist of S*1, S*2, S*3, S*4, SWPCYL, S*5, and S*6. Thus, the total size of the SWPTABLE is 17 doublewords.

SYSLOCS

SYSLOCS: SYSTEM LOW STORAGE INFORMATION BLOCK

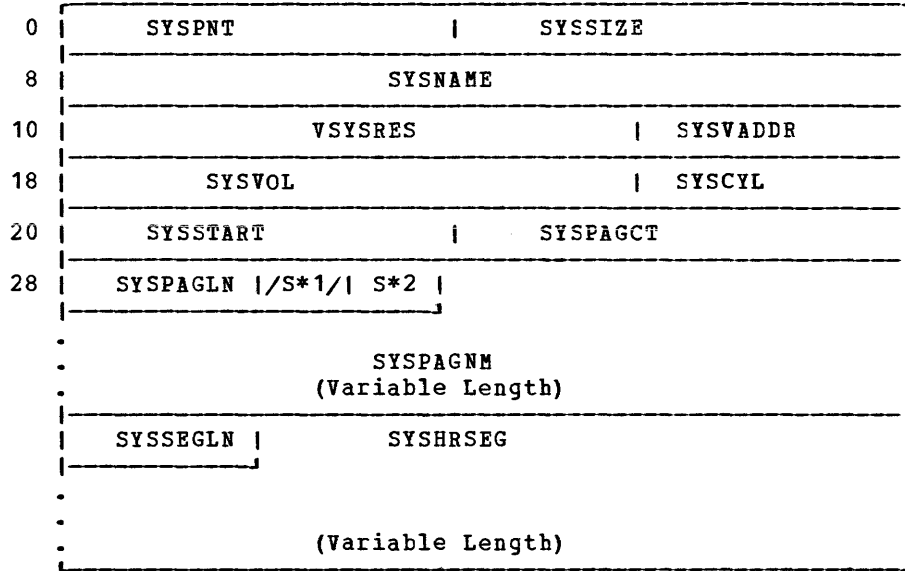
SYSLOCS contains user logon and dial statistics, time/date and log message data, TOD values, and line edit values.



Hexadecimal Displacement	Field Name	Field Description, Contents, Meaning
0	DMKSYSDT DC	CL8'MM/DD/YY' Date of system log message
8	DMKSYSTEM DC	CL8'HH:MM:SS' Time of system log message
10	DMKSYSLW DC	X'00',X'00',CL10' ' Weekday of system log messages
1C	DMKSYSLG DC	A(0) Pointer to first log message block
20	DMKSYSNM DC	F'0' Current number of users on the system
24	DMKSYSMA DC	F'0' Maximum number of users allowed on the system
28	DMKYSYMU DC	F'0' Maximum number of users on the system
2C	DMKSYSND DC	F'0' Number of dialed users on the system
30	DMKSYSLB DC	A(0) Pointer to user directory lock block
34	DMKSYSUD DC	A(0) Pointer to start of user directory on SYSRES
38	DMKSYSPL DC	A(0) Pointer to a list of virtual page buffers
3C	DC	A(0) Reserved for IBM use
40	DMKSYSDW DC	X'00',X'00', CL10' Day of week in hexadecimal and EBCDIC
4C	DMKSYSLE DC	X'7B' S*1 Terminal line-end symbol
4D	DMKSYSLD DC	X'4A' S*2 Terminal line-delete symbol
4E	DMKSYSCD DC	X'7C' S*3 Terminal character-delete symbol
4F	DMKSYSES DC	X'7F' S*4 Terminal escape symbol
50	DMKSYSLL DC	AL1(130,129,72,80) S*5 Default line lengths for 3210 and 3215 - 2741 and 1050 - 3270 and 3066 terminals
53	DC	XL5'0' Reserved for IBM use
58	DMKSYSCK DC	D'0' TOD clock value last stored by accounting, DUMP, or machine check

SYSTEL: NAMED SYSTEM TABLE

SYSTEL contains the system and DASD information required to load a saved system by name. SYSTEL is built during system generation in DMKSNT using the NAMESYS macro.

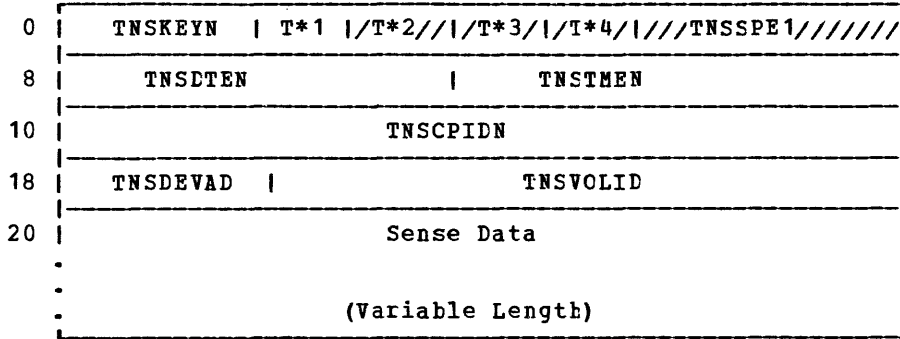


Hexadecimal Displacement	Field Name	Field Description, Contents, Meaning		
0	SYSPNT DS 1F	Chain pointer to next entry		
4	SYSSIZE DS 1F	Minimum storage size needed to run system		
8	SYSNAME DS CL8	System name		
10	VSYSRES DS CL6	Volume identification number of DASD containing user's system		
16	YSVADDR DS 1H	Virtual address of VSYSRES		
18	SYSVOL DS CL6	Volume identification number of DASD containing saved pages		
1E	SYSCYL DS 1H	Cylinder on VSYSRES of user's system. Same as VDEVRELN		
20	SYSSTART DS 1F	CCPD of first page on SYSVOL		
24	SYSPAGCT DS 1F	Total number of pages saved		
28	SYSPAGLN DS 1H	Number of entries in SYSPAGNM		
2A	DS 1X	S*1	Reserved for IBM use	
2B	SYSFLAG DS 1X	S*2	Named system flag	
	<u>Bits defined in SYSFLAG</u>			
	SYSPROT EQU	X'80'	Unprotected shared segments	
	SYSPAGNM DS 1F	One fullword entry for each range of pages to be saved		
	SYSSEGLN DS 1H	Numbers of entries in SYSHRSEG		
	SYSHRSEG DS 1X	One byte for each segment to be shared		

TNSREC

TNSREC: "T" TYPE RECORD FORMAT (ENVIRONMENTAL RECORDING)

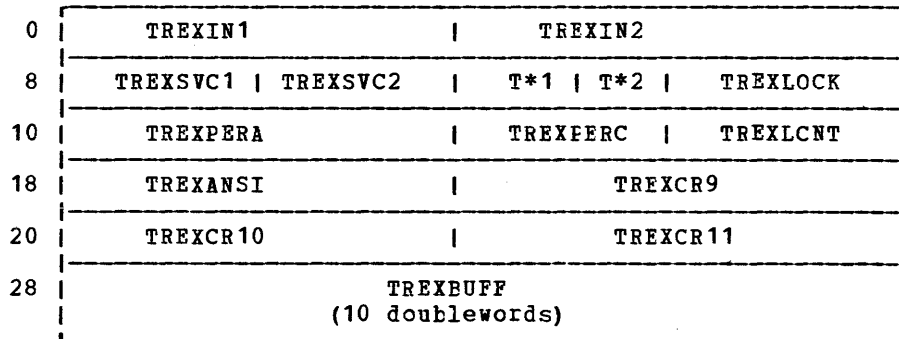
TNSREC is used by DMKIOE to record miscellaneous data records on CP's I/O error recording cylinders. The record contains sense data applicable to a specific I/C device.



Hexadecimal Displacement	Field Name			Field Description, Contents, Meaning
<u>24-Byte Header Record</u>				
0	TNSKEYN	DS	1H	Class source 90=T type N/S NCN TPER
2	TNSSWS1	DS	1X	T*1 Switch byte 0
3	TNSSWS2	DS	1X	T*2 Reserved for IBM use
4	TNSSWS3	DS	1X	T*3 Reserved for IBM use
5	TNSRECNT	DS	1X	T*4 Reserved for IBM use
6	TNSSPE1	DS	1H	Reserved for IBM use
8	TNSDTEN	DS	1F	Date
C	TNSTMEN	DS	1F	Time
10	TNSCPIDN	DS	2F	Processor identification and model number
<u>End of 24-Byte Header</u>				
<u>Device Dependent Data</u>				
18	TNSDEVAD	DS	1H	Device address request is pending
1A	TNSVOLID	DS	6X	Volume identification number
20	TNSSNS1	DS	24X	24 Sense bytes
38	TNSSNS2	DS	24X	24 Additional sense bytes
50	TNSSNS3	DS	24X	24 Additional sense bytes
68	TNSSNS4	DS	24X	24 Additional sense bytes
80	TNSSNS5	DS	24X	24 Additional sense bytes
98	TNSSNS6	DS	24X	24 Additional sense bytes
E0	TNSSNS7	DS	24X	Last 24 sense bytes

TREXT: VIRTUAL MACHINE TRACING EXTENSION TO VMBLCK

TREXT facilitates the tracing of virtual machine program instructions and interrupts. This block is used whenever the CP TRACE command is invoked. The VMTREXT field of the VMBLCK points to TREXT



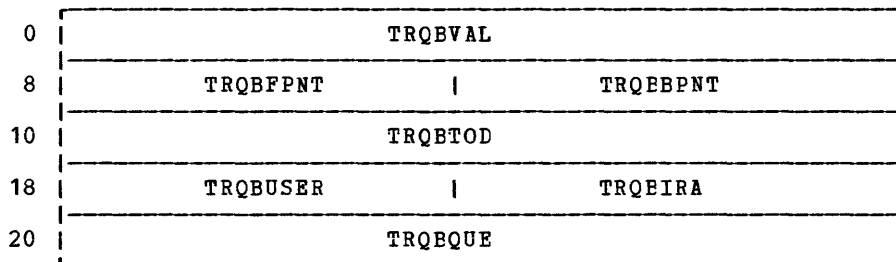
Hexadecimal Displacement	Field Name			Field Description, Contents, Meaning
0	TREXIN1	DS	1F	First address - replaced instruction
4	TREXIN2	DS	1F	Secnd address - replaced instruction
8	TREXSVC1	DS	1H	Displaced halfword - instruction 1
A	TREXSVC2	DS	1H	Displaced halfword - instruction 2
		ORG	TREXIN1	
0	TREXPSW	DS	1D	Old PSW for pending SVC interrupt
8	TREXINTL	DS	1H	Instruction length code
A	TREXINTC	DS	1H	Interrupt code for pending interrupt
C	TREXFLAG	DS	1X	T*1 Tracing control flags
	<u>Bits defined in TREXFLAG</u>			
	TREXRUN	EQU	X'80'	Prevent CFWAIT between events
	TREXVAT	EQU	X'40'	Call DMKVATR to put back virtual instruction
	TREXNDSP	EQU	X'20'	DSP should not call TRCIT
D	TREXOUT	DS	1X	T*2 Trace output controls
	<u>Bits defined in TREXOUT</u>			
	TREXPRT	EQU	X'80'	Output to the virtual spool printer
	TREXCON	EQU	X'40'	Output to user virtual machine terminal
E	TREXLOCK	DS	1H	Indicates tracing when set
10	TREXPERA	DS	1F	Address of instruction causing PER interrupt
		ORG	TREXPERA	Redefinition for TRACE use
10	TREXNSI	DS	6X	Actual next (or last) sequential instruction
14	TREXPERC	DS	1H	PER code to be reflected
16	TREXLCNT	DS	1H	Printed output line count
18	TREXANSI	DS	1A	Address of next (or last) sequential instruction

TREXT

Hexadecimal Displacement	Field Name			Field Description, Contents, Meaning
1C	TREXCR9	DS	0F	Shadow control registers for PER trace
		ORG	TREXCR9	Redefinition for TRACE use
1C	TREXCTL	DS	0H	Halfword holding tracing control bits:
	TREXCTL1	DS	1X	First byte = same as VMTRCTL in VMBLOK
1D	TREXCTL2	DS	1X	Second byte = remaining control bits
<u>Bits defined in TREXCTL2</u>				
	TREXCCW	EQU	X'80'	Trace virtual and real CCWs
	TREXCSW	EQU	X'40'	Trace virtual and real CSWs
	TREXBRAN	EQU	X'20'	Trace successful branches
	TREXINST	EQU	X'10'	Trace all instructions
1C	TREXPER	DS	XL2	PER control field
1E	TREXPREG	DS	1H	PER register mask field
20	TREXCR10	DS	1F	Address range starting value
1E	TREXPRNT	DS	1H	Printer flag bits corresponding to TREXCTL
20	TREXTERM	DS	1H	Terminal flag bits corresponding to TREXCTL
22	TREXRUNF	DS	1H	Run/ncrun flag bits corresponding to TREXCTL
24	TREXPNTR	DS	1F	Pointer to first stacked TRACE request, if any
24	TREXCR11	DS	1F	Address range ending value
28	TREXBUFF	DS	10D	Console/printer output buffer (80 bytes)
	TREXSIZE	EQU	(*--TREXT)/8	TREXT size in doublewords (X'0F')

TRQBLOK: TOD CLOCK COMPARATOR REQUEST

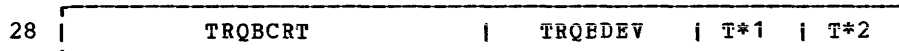
TRQBLOK manages the timing facilities of VM/370.



Hexadecimal Displacement	Field Name			Field Description, Contents, Meaning
0	TRQBVAL	DS	1D	TOD clock comparator value for interrupt
8	TRQBFNT	DS	1F	Pointer to next TRQBLOK
C	TRQBBPNT	DS	1F	Pointer to previous TRQBLOK
10	TRQBTOD	DS	1D	TOD clock value when TRQBLOK is queued
18	TRQBUSER	DS	1F	Address of VMBLOK for user
1C	TRQEIRA	DS	1F	Interrupt return address
20	TRQBQUE	DS	1D	Time left in queue; tracking virtual processor timer
	TRQBSIZE	EQU	(* - TRQBLOK) / 8	Size in doublewords (X'05')

| • Local Graphic Device Support

The following continuation of the TRQBLOK DSECT for local graphic device support is built, referenced, and released by DMKGRF.

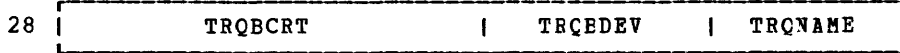


Hexadecimal Displacement	Field Name			Field Description, Contents, Meaning
28	TRQBCRT	DS	1F	Graphic device return IRA
2C	TRQBDEV	DS	1H	Graphic device device address
2E	TRQBFLAG	DS	1X	T*1 Graphic device flags
	<u>Bits defined in TRQBFLAG</u>			
	CRTFMT	EQU	X'80'	Screen formatted VM/370 online
	CRTDIAG	EQU	X'40'	Screen written with Diagnose
	CRTALRM	EQU	X'20'	Screen has alarm message
	CRTWNG	EQU	X'10'	Screen has MORE... warning
	CRTCARD	EQU	X'08'	Data from card reader
	CRTTAB2	EQU	X'04'	Second pass flag for tab function
	CRTAPL	EQU	X'02'	APL read buffer allocated
	CRTSIO	EQU	X'01'	User issue Diagnose to input area
	CRTAIO	EQU	CRTSIO	Timer interrupt pending after I/O completes
2F	TRQBLINE	DS	1X	T*2 Line coordinate for input area

TRQBLOK, UDEFBLOK

| • Remote Graphic Device Support

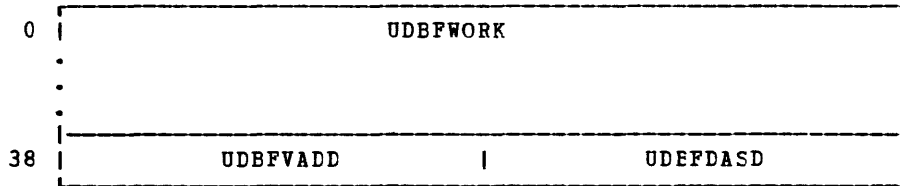
The following continuation of the TRQBLOK DSECT for remote graphic device support is built, referenced, and released by DMKRG and DMKRG.



Hexadecimal Displacement	Field Name	Field Description, Contents, Meaning		
28	TRQBCRT DS 1F	Graphic device return IRA		
2C	TRQBDEV DS 1H	Graphic device line address		
2E	TRQNAME DS 1H	Resource identification		
	TRQBpoll EQU X'FF'	Timer interrupt for general poll		

UDEFBLOK: USER DIRECTORY BUFFER BLOCK

UDEFBLOK is used as a buffer for user device block data in user directory access operations.



Hexadecimal Displacement	Field Name	Field Description, Contents, Meaning		
0	UDEFWORK DS 7D	Buffer work space used by the caller		
38	UDEFVADD DS 1F	Virtual address of the last directory page		
3C	UDEFDASD DS 1F	DASD address of the last directory page		
	UDEFsize EQU (*-UDEFBLOK)/8	UDEFBLOK size in doublewords (X'08')		

UDEVLOK: USER DEVICE BLOCK

UDEVBLOCK supplies the information about the virtual machine's virtual devices, the operational parameters for its use, such as DASD access passwords, read/write link mode, spool device, T-disk space versus dedicated device space, as well as other parameters.

0	UDEVADD	UDEVDISP	UDEVDAASD
8	U*1	U*2	U*3 U*4 U*5 U*6 UDEVNCYL
10	UDEVRELN	UDEVVSR	
18	UDEVPASR		
20	UDEVPASW		
28	UDEVPASM		

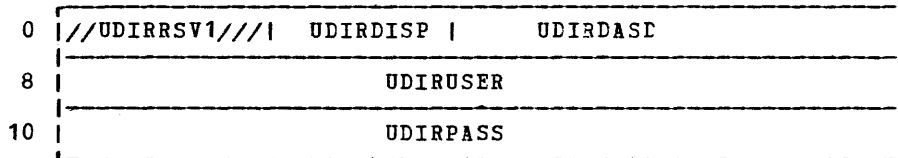
Hexadecimal Displacement	Field Name			Field Description, Contents, Meaning
0	UDEVADD	DS	1H	Virtual device address
2	UDEVDISP	DS	1H	Displacement of the next block
4	UDEVDAASD	DS	1F	DASD address of the next block
8	UDEVSTAT	DS	1X	U*1 Device status information
<u>Bits defined in UDEVSTAT</u>				
	UDEVDED	EQU	X'80'	Device to be dedicated to this user
	UDEVTDISK	EQU	X'40'	T-disk to be allocated
	UDEVLONG	EQU	X'20'	Device block is full length (6 doublewords)
	UDEVLKDV	EQU	X'10'	Device is to be linked (at logon)
	UDEVSPOO	EQU	X'08'	Device is a spool device
	UDEV3158	EQU	X'04'	Device is a 3158 console
	UDEVVRR	EQU	X'02'	Virtual reserve/release requested
9	UDEVMODE	DS	1X	U*2 Access mode information
<u>Bits defined in UDEVMODE</u>				
	UDEVLR	EQU	X'80'	Read links allowed
	UDEVLW	EQU	X'40'	Write links allowed
	UDEVLM	EQU	X'20'	Multiple write links allowed
	UDEVRR	EQU	00	Device to be in R link mode for owner
	UDEVRRR	EQU	04	Device to be in RR link mode for owner
	UDEVW	EQU	08	Device to be in W link mode for owner
	UDEVWR	EQU	12	Device to be in WR link mode for owner
	UDEVMM	EQU	16	Device to be in M link mode for owner
	UDEVMR	EQU	20	Device to be in MR link mode for owner
	UDEVMMW	EQU	24	Device to be in MW link mode for owner
A	UDEVTPC	DS	1C	U*3 Virtual device class
B	UDEVTYPE	DS	1C	U*4 Virtual device type
C	UDEVFTR	DS	1C	U*5 Device feature codes
D	UDEVMDL	DS	1C	U*6 Device model number
		ORG	UDEVMDL	User device block (short)
D	UDEVCLAS	DS	1C	Spool device output class
E	UDEVLINK	DS	1H	User link to disk
10	UDEVLKID	DS	1D	User link to userid
E	UDEVNCYL	DS	1H	Virtual DASD size
10	UDEVRELN	DS	1H	Virtual DASD cylinder relocation
12	UDEVVSR	DS	6C	Volume identification number

UDEVELOK, UDIRBLOK

Hexadecimal Displacement	Field Name	Field Description, Contents, Meaning
18	UDEVPASR DS 1D	Password for read access
20	UDEVPASW DS 1D	Password for write access
28	UDEVPASM DS 1D	Password for multiple access
	UDEVSIZ EQU (*-UDEVBLOK)/8	UDEVELOK size in doublewords (X'06')

UDIRELOK: USER DIRECTORY BLOCK

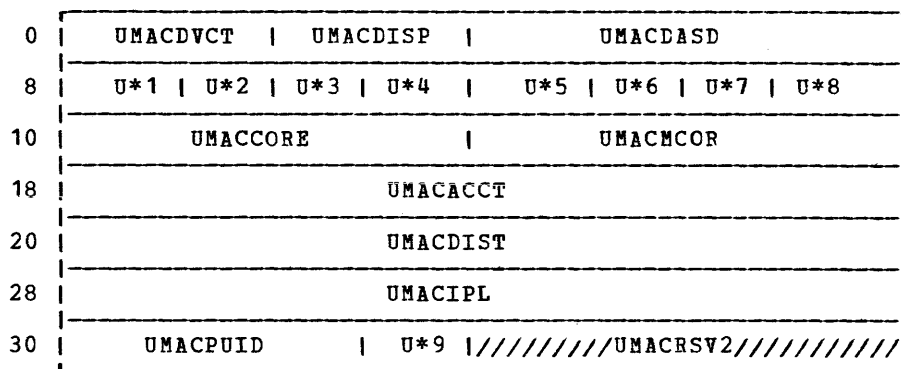
UDIRELOK contains data describing the user's command privilege classes, special virtual machine options, terminal line edit values, and other values.



Hexadecimal Displacement	Field Name	Field Description, Contents, Meaning
0	UDIRRSV1 DS 1H	Reserved for IBM use
2	UDIRDISP DS 1H	Displacement of the user's UMACBLOK
4	UDIRDASD DS 1F	DASD address of the user's UMACBLOK
8	UDIRUSER DS 1D	Userid
10	UDIRPASS DS 1D	User password
	UDIRSIZE EQU (*-UDIRBLOK)/8	UDIRELOK size in doublewords (X'03')

UMACBLCK: USER MACHINE BLOCK

UMACBLCK contains the logon parameters for one virtual machine user. This block provides, in addition to the linkage to the user's defined virtual machine device UDEVBLCK, the command privilege class, assigned line edit values, as well as other virtual machine options.



Hexadecimal Displacement	Field Name				Field Description, Contents, Meaning
0	UMACDVCT	DS	1H		Number of devices
2	UMACDISP	DS	1H		Displacement of the next block
4	UMACDASD	DS	1F		DASD address of the next block
8	UMACCLEV	DS	1C	U*1	Command level flags
	<u>Bits defined in UMACCLEV</u>				
	UMACCLA	EQU	X'80'		Privilege Class A functions
	UMACCLB	EQU	X'40'		Privilege Class B functions
	UMACCLC	EQU	X'20'		Privilege Class C functions
	UMACCLD	EQU	X'10'		Privilege Class D functions
	UMACCLE	EQU	X'08'		Privilege Class E functions
	UMACCLF	EQU	X'04'		Privilege Class F functions
	UMACCLG	EQU	X'02'		Privilege Class G functions
	UMACCLH	EQU	X'01'		Privilege Class H functions
9	UMACPRIR	DS	1X	U*2	Virtual machine priority
A	UMACOPT	DS	1X	U*3	Virtual machine option flags
	<u>Bits defined in UMACOPT</u>				
	UMACISAM	EQU	X'80'		ISAM CCW checking option
	UMACECOP	EQU	X'40'		Extended control mode option
	UMACRT	EQU	X'20'		Real timer option
	UMACVROP	EQU	X'10'		Virtual = Real storage option
	UMACACC	EQU	X'08'		Accounting card option
	UMACRSV4	EQU	X'04'		Reserved for IBM use
	UMACNSVC	EQU	X'02'		SVCs not handled by virtual machine assist feature
	UMACBMX	EQU	X'01'		Virtual block multiplexer channel

UMACBLOK

Hexadecimal Displacement	Field Name			Field Description, Contents, Meaning
B	UMACOPT2 DS	1C	U*4	Virtual machine option flags
	<u>Bits defined in UMACOPT2</u>			
	UMACCPU EQU	X'80'		Processor identification number on option statement
	UMACVMSV EQU	X'10'		VMSAVE on option statement
C	UMACLEND DS	1C	U*5	Terminal line end symbol
D	UMACLDEL DS	1C	U*6	Terminal line delete symbol
E	UMACCDEL DS	1C	U*7	Terminal character delete symbol
F	UMACES DS	1C	U*8	Edit escape symbol
10	UMACCORE DS	1F		Virtual storage size in bytes
14	UMACMCOR DS	1F		Maximum virtual storage size in bytes
18	UMACACCT DS	1D		Accounting information
20	UMACDIST DS	1D		User machine distribution information
28	UMACIPL DS	1D		Name of system to be IPLed at lcgcn
30	UMACPUID DS	XL3		Processor identification number in binary
33	UMACAFF DS	1X	U*9	Affinity and processor address
	<u>Bits defined in UMACAFF</u>			
	UMACFFON EQU	X'40'		Affinity specified
	UMACFFAD DS	OBL6		Processor address for affinity
34	UMACRSVR DS	1F		Reserved for IBM use
	UMACSIZE EQU	(*-UMACBLOK)/8		UMACBLOK size in doublewords (X'06')

Virtual I/O Control Blocks

The base for locating the I/O block structure is the user's Virtual Machine Control Block (VMBLOCK). The VMBLOCK contains a pointer to the start of three control block tables, and a table of 16 channel indexes. The control block tables contain one block for each of the virtual channels, control units, and devices that are defined for the user's virtual machine. The entries in the channel index table (VMCHTBL) contain the pointers to each channel defined for the user in the table of Virtual Channel Blocks (VCHBLOCKS). Each VCHBLOCK contains a table of pointers that point to the Virtual Control Unit Blocks (VCUBLOCKS) for the control units attached to that virtual channel. Each VCUBLOCK contains pointers to the Virtual Device Blocks (VDEVBLOCK) attached to the control unit.

Thus, if given the unit address of any component in the form `cuu`, the appropriate control blocks representing each component in the subchannel path to the given unit is located via the indexing scheme.

VIRTUAL CHANNEL BLOCKS

There is one Virtual Channel Block (VCHBLOCK) for each virtual channel connected to the user's virtual processor. Each VCHBLOCK contains the channel address and flag indicating the channel type (selector, byte multiplexer, or block multiplexer). The status of the channel and its attached units are represented by several status and mask bytes.

- A status byte (VCHSTAT) indicates whether the channel is busy or has a channel class interrupt pending.
- A halfword unit address identifies the unit causing the channel-class interrupt (if it is present).
- A halfword mask (VCHCUINT) contains a bit map of the attached control units that have interrupt status pending.

Following these status flags and masks is the table of indexes pointing to the attached VCUBLOCKS; index entries representing addresses at which no control unit is attached have a value of -1.

VIRTUAL CONTROL UNIT BLOCKS

There is one Virtual Control Unit Block (VCUBLOCK) for each control unit in the virtual configuration. These blocks are arranged in a table, each contains: in addition to its base address, status flags similar to those in the VCHBLOCK and a table of indexes to attached VDEVBLOCKS. The status flags defined for the VCUBLOCK differ from those for the VCHBLOCK in that they can contain status for the control unit and also for a subchannel.

For example, if the VCUBLOCK representing a 2803 tape control unit is attached to a virtual selector channel, both the VCHBLOCK and the VCUBLOCK are marked busy. However, if the VCUBLOCK is attached to a virtual byte multiplexer channel and is for a control unit on a selector subchannel of the multiplexer, the busy status of the channel is reflected in the VCUBLOCK only. Thus, the virtual byte multiplexer appears nonbusy to operations on other, nonshared subchannels.

VIRTUAL DEVICE BLOCKS

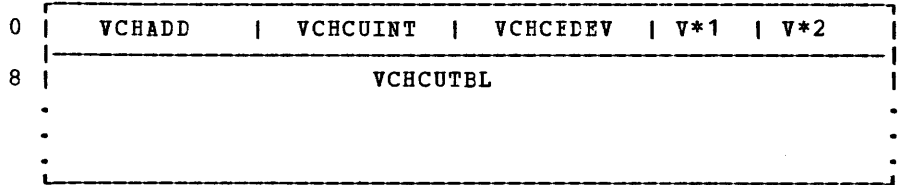
There is one Virtual Device Block (VDEVBLOCK) in the configuration for each virtual device defined by the user. Each VDEVBLOCK contains the device portion of the unit address, device status, and the virtual CSW for the last interrupt taken by the device. In addition, the VDEVBLOCK contains device type specific information that allows the I/O translation and simulation routines to interpret the channel programs presented by the user.

Note: The VCHBLOCK, VCUBLOCK, VDEVBLOCK, VFCEBLOCK, and VSPXBLOCK DSECTS are all contained in the VELOKS COPY file.

VCHBLOK

VCHBLOK: VIRTUAL CHANNEL BLOCK

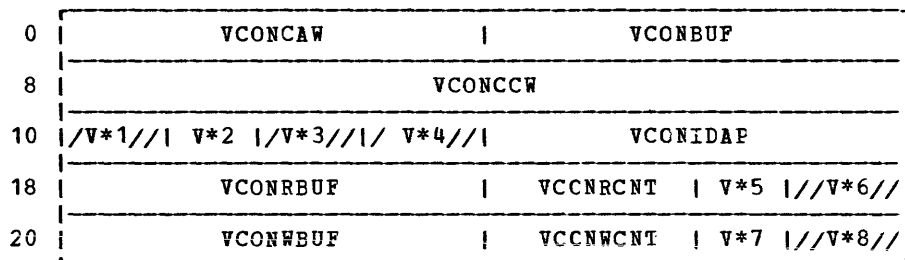
VCHBLOK contains information providing linkage between the virtual machine and one of its virtual channels. Supplied in this block, in addition to channel status and type information, are the reflected interrupts from attached virtual control units. The VMCHSTRT field of the VMBLOK points to the first VCHBLOK.



Hexadecimal Displacement	Field Name				Field Description, Contents, Meaning
0	VCHADD	DS	1H		Virtual channel address
2	VCHCUINT	DS	1H		VCUBLOK with interrupt-bit map
4	VCHCEDEV	DS	1H		Virtual device address with channel class interrupt
6	VCHSTAT	DS	1X	V*1	Virtual channel status
	<u>Bits defined in VCHSTAT</u>				
	VCHBUSY	EQU	X'80'		Virtual channel busy
	VCHCEPND	EQU	X'40'		Virtual channel class interrupt pending
	VCHDED	EQU	X'01'		Virtual channel dedicated
7	VCHTYPE	DS	1X	V*2	Virtual channel type
	<u>Bits defined in VCHTYPE</u>				
	VCHSEL	EQU	X'80'		Virtual selector channel
	VCHBMX	EQU	X'40'		Virtual block multiplexer
8	VCHCUTBL	DS	16H		Control units attached - VMCHSTRT index
	VCHSIZE	EQU	(*-VCHBLOK)/8		VCHBLOK size in doublewords (X'05')

VCONCTL: VIRTUAL CONSOLE CONTROL BLOCK

VCONCTL contains CCW and data buffer information for the communications of the virtual console. The VDEVCON field of the VDEVBLK points to VCONCTL.

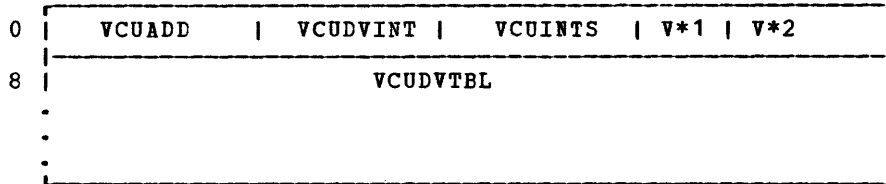


Hexadecimal Displacement	Field Name	Field Description, Contents, Meaning		
0	VCONCAW DS 1F	Virtual address of user CCW		
4	VCONBUF DS 1F	Pointer to data buffer		
8	VCONCCW DS 1D	Current user CCW		
		ORG	VCONCCW	
8	VCONADDR DS 1F	CCW data address		
C	VCONFLAG DS 1X	CCW flag bits		
D	VCONDWC DS 1X	Diagnose write control		
E	VCONCNT DS 1H	CCW byte count		
		ORG	VCONADDR	
8	VCONCMD DS 1X	CCW command code		
10	VCONRSV1 DS 1X	V*1	Reserved for IBM use	
11	VCONBFSZ DS 1X	V*2	Data buffer size in doublewords	
12	VCONRSV2 DS 1X	V*3	Reserved for IBM use	
13	VCONRSV3 DS 1X	V*4	Reserved for IBM use	
14	VCONIDAP DS 1F	For indirect data addressing pointer to current IDAW		
18	VCONRBUF DS 1F	Address of read data buffer		
1C	VCONRCNT DS 1H	Data count in read buffer		
1E	VCONRBSZ DS 1X	V*5	Read buffer size in doublewords	
1F	VCONRSV6 DS 1X	V*6	Reserved for IBM use	
20	VCONWBUF DS 1F	Address of write data buffer		
24	VCONWCNT DS 1H	Data count in write buffer		
26	VCONWSZ DS 1X	V*7	Write buffer size in doublewords	
27	VCONRSV8 DS 1X	V*8	Reserved for IBM use	
	VCONSIZE EQU	(*-VCONCTL)/8	VCONCTL size in doublewords (X'05')	

VCUBLOK

VCUBLOK: VIRTUAL CONTROL UNIT BLOCK

VCUBLOK contains status information relating to the virtual channel, and the status and features of the virtual control unit. The VMCUSTRT field of the VMELCK points to the first VCUBLOK.



Hexadecimal Displacement	Field Name					Field Description, Contents, Meaning
0	VCUADD	DS	1H		Virtual control unit address	
2	VCUDVINT	DS	1H		VDEVELOK with interrupt - bit map	
4	VCUINTS	DS	1H		Virtual control unit interrupt status	
6	VCUSTAT	DS	1X	V*1	Virtual control unit status	
	<u>Bits defined in VCUSTAT</u>					
	VCUCHBSY	EQU	X'80'		Virtual subchannel busy	
	VCUCEPND	EQU	X'40'		Interrupt pending in subchannel	
	VCUBUSY	EQU	X'20'		Virtual control unit busy	
	VCUPEND	EQU	X'10'		Virtual control unit interrupt pending	
	VCUCUEPN	EQU	X'08'		Virtual control unit end pending	
	VCUACTV	EQU	X'04'		Virtual control unit active	
7	VCUTYPE	DS	1X	V*2	Virtual control unit type	
	<u>Bits defined in VCUTYPE</u>					
	VCUSHRD	EQU	X'80'		Virtual control unit on shared subchannel	
	VCUCTCA	EQU	X'40'		Virtual control unit is a channel-to-channel adapter	
8	VCUDVTBL	DS	16H		Devices attached - VMDVSTRT index	
	VCUSIZE	EQU	(*VCUBLOK)/8		VCUBLOK size in doublewords (X'05')	

VDEVBLCK: VIRTUAL DEVICE BLOCK

VDEVBLCK maintains status and interrupt conditions for one virtual device. The VMDVSTRT field of the VMBLCK points to the first VDEVBLCK.

0	VDEVADD	VDEVINTS	V*1	V*2	V*3	V*4
8	VDEVCSW					
10	VDEVRELN	VDEVBND	VDEVPCSN			
18	VDEVQUED		VDEVOPER			
20	VDEVLINK		VDEVREAL			
28	VDEVIOCT		VDEVUSER			
30	VDEVIOER		VDEVICB			
38	V*5	/////VDEVRES1/////	VDEVRRB			

Hexadecimal Displacement	Field Name	Field Description, Contents, Meaning			
0	VDEVADD DS 1H	Virtual device address			
2	VDEVINTS DS 1H	Virtual device interrupt status			
4	VDEVTYPE DS 1X	V*1	Virtual device type class		
5	VDEVTYPE DS 1X	V*2	Virtual device type		
6	VDEVSTAT DS 1X	V*3	Virtual device status		
	<u>Bits defined in VDEVSTAT</u>				
	VDEVCHBS EQU X'80'	Virtual subchannel busy			
	VDEVCHAN EQU X'40'	Virtual channel interrupt pending			
	VDEVBUSY EQU X'20'	Virtual device busy			
	VDEVPEND EQU X'10'	Virtual device interrupt pending			
	VDEV CUE EQU X'08'	Virtual control unit end			
	VDEVNRDY EQU X'04'	Virtual device not ready			
	VDEV CATT EQU X'02'	Virtual device attached by console function			
	VDEVDED EQU X'01'	VDEVREAL is dedicated device RDEVBLCK			
7	VDEVFLAG DS 1X	V*4	Virtual device flags		
	<u>Bits defined in VDEVFLAG</u>				
	VDEV RDO EQU X'80'	DASD - read-only			
	VDEV ENAB EQU X'80'	Virtual 270x - line enabled			
	VDEV TDSK EQU X'40'	DASD - T-disk space allocated by CP			
	VDEV DIAL EQU X'40'	Virtual 270x - line connected			
	VDEV CSPL EQU X'40'	Conscle - activity spooled			
	VDEV 231T EQU X'20'	DASD - 2311 simulated on top half of 2314			
	VDEV 231B EQU X'10'	DASD - 2311 simulated on bottom half of 2314			
	VDEV CCW1 EQU X'10'	Conscle and spooling - processing first CCW			
	VDEV SAS EQU X'08'	DASD - Executing standalone seek			
	VDEV DLY EQU X'08'	Conscle - delay spooling			
	VDEV DET EQU X'04'	Virtual device is being detached			
	VDEV POST EQU X'02'	Present attention with a single interrupt			
	VDEV SRSL EQU X'02'	Reserve/release are valid CCW operation codes			
	VDEV UC EQU X'01'	Virtual device sense bytes present			

VDEVBLK

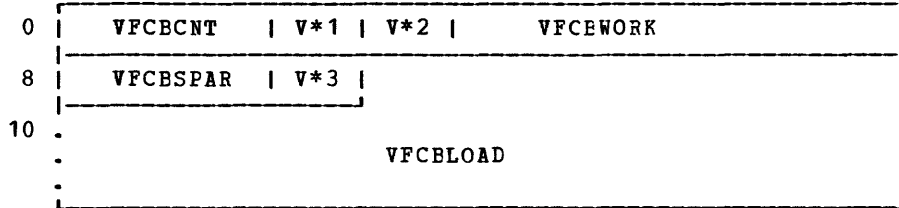
Hexadecimal Displacement	Field Name			Field Description, Contents, Meaning
8	VDEVCSW	DS	1D	Virtual channel status word
10	VDEVRELN	DS	1H	Virtual DASD cylinder relocation
12	VDEVBNDS	DS	1H	Virtual DASD size (in cylinders)
14	VDEVPOSN	DS	1F	Virtual DASD seek position
18	VDEVQUED	DS	1F	Virtual SIO to real SIO queued time
1C	VDEVOPER	DS	1F	Device operational time
20	VDEVLINK	DS	1F	Link to virtual shared devices
		ORG	VDEVLINK	
20	VDEVTMAT	DS	1F	T-disk attached time (TOD clock word 0)
24	VDEVREAL	DS	1F	Pointer to real device RDEVLOCK
28	VDEVIOCT	DS	1F	Virtual device I/O count
2C	VDEVUSER	DS	1F	Pointer to VMBLOCK of VDEVBLK owner
30	VDEVIOER	DS	1F	Pointer to IOERBLOK for last error
		ORG	VDEVIOER	
30	VDEVSNSE	DS	1F	Sense bytes for spool device
34	VDEVFCBK	DS	1F	Address of forms control block (VFCBBLOK)
34	VDEVIOB	DS	1F	Pointer to active IOBLOK
38	VDEVFLG2	DS	1X	V*5 Virtual device flag byte 2
	<u>Bits defined in VDEVFLG2</u>			
	VDEVRRF	EQU	X'80'	Process virtual RESERVE/RELEASE commands
	VDEVRES	EQU	X'40'	Minidisk reserved by VDEVUSER
	VDEVODE	EQU	X'20'	VDEVLOCK to get device when minidisk is released
	VDEVCPX	EQU	X'10'	Virtual I/O waiting for release of minidisk
	VIRTUAL	EQU	X'01'	Virtual device is known by the virtual machine as a 3330V
39	VDEVRES1	DS	3X	Reserved for IBM use
3C	VDEVRRB	DS	1F	Address of VRRBLOK for RESERVE/RELEASE
	VDEVSIZE	EQU	(*-VDEVBLK)/8	VDEVLOCK size in doublewords (X'07')
	<u>For Spooling/Console Devices</u>			
		ORG	VDEVRELN	
10	VDEVEXTN	DS	1F	Pointer to spool extension block
14	VDEVSPAR	DS	1F	Spare pointer to spool extension block
18	VDEVCON	DS	1F	Pointer to VCONCTL console control
1C	VDEVSPPL	DS	1F	Pointer to VSPLCTL spool control
20	VDEVCLAS	DS	1C	Spool output class
21	VDEVKEY	DS	1X	Storage key in user's CAW
22	VDEVUNIT	DS	1H	Spool output directed device address
24	VDEVCPY	DS	1H	Number of copies requested
26	VDEVCLFG	DS	1X	Console - virtual console flags
	<u>Bits defined in VDEVCLFG</u>			
	VDEVATTN	EQU	X'80'	User pressed Attention key two or more times
	VDEVTTIC	EQU	X'40'	Last CCW processed was a TIC
	VDEVTRAN	EQU	X'20'	Data transfer occurred during this channel program
	VDEVVCF	EQU	X'10'	Virtual console function in progress
	VDEVAUCR	EQU	X'08'	Automatic carriage return on first read

Hexadecimal Displacement	Field Name		Field Description, Contents, Meaning
27	VDEVSFLG	DS 1X	Spool - virtual spool flags
	<u>Bits defined in VDEVSFLG</u>		
	VDEVFEED	EQU X'80'	Spool reader - last command was a feed
	VDEVXFER	EQU X'80'	Spool output - transferred to VDEVXUSR
	VDEVCONT	EQU X'40'	Spool device - continuous operation
	VDEVHOLD	EQU X'20'	Hold output - save input
	VDEVFOR	EQU X'10'	Spool output - for user and distribution
	VDEVEOF	EQU X'08'	Spool input - set unit exception at EOF
	VDEVTERM	EQU X'08'	Terminal output required for spooled console
	VDEVCFCL	EQU X'04'	Device closed by console function
	VDEVPURG	EQU X'02'	Spool output - purge file at close
	VDEVDIAG	EQU X'02'	Spool input - device opened by DIAGNOSE
	VDEVSVL	EQU X'01'	Spool device busy by CP

VFCBBLOK, VMABLOK

VFCBBLOK: VIRTUAL FORMS CONTROL BUFFER BLOCK

VFCBBLOK is exclusively used for virtual 3211 printer carriage control forms activity.



Hexadecimal Displacement	Field Name	Field Description, Contents, Meaning				
0	VFCBCNT DS 1H	Current pointer to carriage column				
2	VFCBFLAG DS 1X	V*1	Working flag byte			
<u>Bits defined in VFCBFLAG</u>						
	VFCBEOF EQU X'80'	End-of-forms passed once				
	VFCBCMD EQU X'40'	Forms control given				
3	VFCBCHL DS 1X	V*2	Channel number or space count			
4	VFCBWORK DS 1F	Work area				
8	VFCBSPAR DS 2X	Spare				
A	VCFBNDEX DS 1X	V*3	Index byte value			
B	VFCBLOAD DS CL181	Forms control buffer area				
	VFCBSIZE EQU (*-VFCBBLOK)/8	Size in doublewords (X'18')				

VMABLOK: SHARED SYSTEMS CONTROL ADDITION TO VMELCK

VMABLOK contains the name of a shared system, a pointer to the share table (SHRTABLE), and a pointer to the next VMABLOK. The VMASIST field in the VMELCK points to the VMABLOK.



Hexadecimal Displacement	Field Name	Field Description, Contents, Meaning			
0	VMAFPNT DS 1F	Forward pointer to next VMABLOK			
4	VMASHRBK DS 1F	Address of share table (SHRTABLE)			
8	VMANAME DS CL8	Identification of named shared system			
	VMASIZE EQU (*-VMABLOK)/8	VMABLOK size in doublewords (X'02')			

VMBLOCK: VIRTUAL MACHINE CONTROL BLOCK

VMBLOCK is used as the primary control block for almost all activities related to a single virtual machine. This block contains the following information: the dispatch and priority level of the virtual machine, the virtual machine's processor registers, preferred virtual machine option values, and other values significant to virtual machine operations. The ASYSVM field of the PSA points to the system VMBLOCK.

0	VMQFPNT		VMQBPNT	120	VMDIST			
8	VMPNT		VMECEXT	128	VMPGREAD		VMPGWRT	
10	VMSEG		VMSIZE	130	VMWCNT	VMSEGDSP	VMSTOR	
18	VMCHSTRT		VMCUSTRT	138	VHIOCNT		VMPNCH	
20	VMDVSTRT		VMTERM	140	VMLINS		VMCRDS	
28	VMVTERM	VMTRMID	V*1 V*2 V*3 V*4	148	VMCOMND			
30	VMCHCNT	VMCUCNT	VMDVCNT	150	/VMPDRUM/	/VMPDISK/	VMPAGES	VMPRGIL
38	VMCHTBL			158	VMDDELCH	VMQPRIOR	VMWSPROJ	VMSTEALS
.	.	.	.	160	VMTIMEON		VMTRQBLK	
58	V*5	V*6 V*7	V*8 V*9 V*10 V*11 V*12	168	VMACCUNT		VMRDINQ	
60	V*13 V*14 V*15 V*16		VMLOCKER	170	VMPGRINQ		VMEPRIOR	
68	V*19 V*20	VMIOINT	VMTIMER	178	VMSTKO		VMMICRO	
70	VMVTIME			180	VMPFUNC		VMPXINT	
78	VMTMOUTQ			188	VMDELAY		VMRPRIOR	
80	VMTTIME			190	VMPGPNT	VMNDCNT	VMSHRSYS	
88	VMTMINQ			198	V*21 V*22 V*26 V*27		VMASSIST	
90	VMTODINQ			1A0	VMCPNT	VMCPUID	/V*25/	
98	VMINST		V*17 V*18	1A8	VMLOCK		VMDFTPNT	
A0	VMTREXT		VMADSTOP	1B0	VMUSER1		VMUSER2	
A8	VMPSW			1B8	VMUSER3		VMUSER4	
E0	VMGPRS			1C0	VMUHS		VMPCKP	
.	.	.	.	1C8	VMXPG	C*1 C*2	VMSTKCNT	VMPRRCT
F0	VMFPRS			1D0	VMSPMIG	C*3 C*4 V*28	V*29	
.	.	.	.	1D8	VMCPTIME			
.	.	.	.	1E0	VMAPTIME			
110	VMUSER			1E8	VMACTDEV	VMFLPAG	V*30 RSVD	VMCONLN
118	VMACNT			1FC	VMCONBUF	V*31	///RESERVED///	

VMBLOK

Hexadecimal Displacement	Field Name			Field Description, Contents, Meaning
0	VMQFPNT	DS	1F	Pointer to next VMELCK in queue
4	VMQBNT	DS	1F	Pointer to previous VMELCK in queue
8	VMFNT	DS	1F	Pointer (CYCLIC) to next VMELCK
C	VMECEXT	DS	1F	VMELCK extended control pointer - ECELOK
	VMVCR0	EQU	VMECEXT	Virtual control register 0 for non-EC mode virtual machine
10	VMSEG	DS	1F	Pointer to VMSEGtbl
14	VMSIZE	DS	1F	Virtual temporary storage size in bytes
18	VMCHSTRT	DS	1F	Pointer to VCHBLOK table
1C	VMCUSTRT	DS	1F	Pointer to VCUBLOK table
20	VMDEVSTRT	DS	1F	Pointer to VDEVBLOK table
24	VMTERM	DS	1F	Pointer to RDEVBLOK for user terminal
28	VMVTERM	DS	1H	Displacement to virtual console VDEVBLOK
2A	VMTRMID	DS	1H	Resource ID of real terminal if 370x
2C	VMTLEND	DS	1C	V*1 Terminal line end symbol
2D	VMTLDL	DS	1C	V*2 Terminal line delete symbol
2E	VMTCDEL	DS	1C	V*3 Terminal character delete symbol
2F	VMTESCP	DS	1C	V*4 Terminal escape symbol
30	VMCHCNT	DS	1H	Virtual channel count
32	VMCUCNT	DS	1H	Virtual control unit count
34	VMDEVCNT	DS	1H	Virtual device count
36	VMIOACTV	DS	1H	Active channel mask
38	VMCHTBL	DS	16H	Channels attached - VMCHSTRT index
58	VMRSTAT	DS	1X	V*5 Virtual machine running status

Bits defined in VMRSTAT

VMCFWAIT	EQU	X'80'	Waiting - Executing console function
VMPGWAIT	EQU	X'40'	Waiting - Paging operation(s)
VMIOWAIT	EQU	X'20'	Waiting - Scheduled IOBLOK start
VMPSWAIT	EQU	X'10'	Waiting - Virtual PSW wait state
VMEXWAIT	EQU	X'08'	Waiting - Instruction simulation
VMLOGON	EQU	X'04'	User not logged on
VMLOGOFF	EQU	X'02'	User logging off
VMIDLE	EQU	X'01'	Virtual machine in idle wait state
VMCPWAIT	EQU	VMCFWAIT+VMPGWAIT+VMIOWAIT+VMEXWAIT+VMLOGOFF+VMLOGON	
VMNORUN	EQU	VMCPWAIT+VMPSWAIT	
VMLONGWT	EQU	VMCFWAIT+VMLOGON+VMLOGOFF+VMIDLE	

59 VMDSTAT DS 1X V*6 Virtual machine dispatching status

Bits defined in VMDSTAT

VMDSP	EQU	X'80'	Virtual machine is dispatched run user
VMTSEND	EQU	X'40'	Virtual machine is compute bound
VMQSEND	EQU	X'20'	Virtual machine in-queue time slice end
VMTIO	EQU	X'10'	Virtual machine is in TIO busy loop
VMRUN	EQU	X'08'	Virtual machine runnable
VMINQ	EQU	X'04'	Virtual machine in a queue
VMELIG	EQU	X'02'	Virtual machine in eligible list
VMPAZAPL	EQU	X'01'	Use dispatch path DMKDSPCH

Hexadecimal Displacement	Field Name				Field Description, Contents, Meaning
5A	VMOSTAT	DS	1X	V*7	Virtual machine operating status
	<u>Bits defined in VMOSTAT</u>				
	VMSYSOP	EQU	X'80'		Virtual machine is system operator
	VMSHR	EQU	X'40'		Virtual machine running shared system
	VMSLEEP	EQU	X'20'		Virtual machine is in SLEEP state
	VMDISC	EQU	X'10'		Virtual machine console disconnected
	VMCFRUN	EQU	X'08'		Virtual machine running in console function mode
	VMVIRCF	EQU	X'04'		Virtual machine executing virtual console function
	VMCF	EQU	X'02'		Virtual machine executing console function
	VMKILL	EQU	X'01'		Virtual machine is to be logged off
5B	VMQSTAT	DS	1X	V*8	Virtual machine queueing status
	<u>Bits defined in VMQSTAT</u>				
	VMPRIDSP	EQU	X'80'		Virtual machine eligible for queue 1
	VMAUTLOG	EQU	X'40'		Autologged user is in disconnect mode
	VMWSERNG	EQU	X'20'		Last working set error was negative
	VMDLDRP	EQU	X'10'		Delayed queue drop flag
	VMWSCHG	EQU	X'08'		Force a new working set on queue entry
	VMINHMIG	EQU	X'04'		Page migration switch
	VMCFREAD	EQU	X'02'		Virtual machine with console function read on
	VMPA2APL	EQU	X'01'		Reflects external interrupts to machine
5C	VMPSTAT	DS	1X	V*9	Virtual machine processing status
	<u>Bits defined in VMPSTAT</u>				
	VMISAM	EQU	X'80'		Virtual machine has ISAM CCW checking
	VMV37OR	EQU	X'40'		Virtual machine can use extended format
	VMPAGE	EQU	X'20'		Virtual machine can reserve pages
	VMREAL	EQU	X'10'		Virtual machine has virtual=real option
	VMNOTRAN	EQU	X'08'		No CCW translation for virtual=real user
	VMNSHR	EQU	X'04'		Running nonshared named saved system
	VMACOUN	EQU	X'02'		Virtual machine may punch account cards
	VMPAGEX	EQU	X'01'		Virtual machine receiving pseudo page faults
5D	VMESTAT	DS	1X	V*10	Virtual machine control status
	<u>Bits defined in VMESTAT</u>				
	VMSHADT	EQU	X'80'		Shadow tables are present
	VMPERCM	EQU	X'40'		Virtual CP PER active
	VMBADCR0	EQU	X'20'		Virtual control register 0 is invalid
	VMMICSVC	EQU	X'10'		User wants SVCs done by virtual machine assist feature
	VMEXTCM	EQU	X'08'		Virtual machine in extended control mode
	VMNEWCRO	EQU	X'04'		Virtual control register 0 has changed
	VMINVSEG	EQU	X'02'		All shadow tables invalid
	VMINVPAG	EQU	X'01'		Shadow page tables invalid
	VMECZAP	EQU	255-VMMICSVC		All bits except VMMICSVC
5E	VMTRCTL	DS	1X	V*11	Virtual machine tracing control
	<u>Bits defined in VMTRCTL</u>				
	VMTRPER	EQU	X'80'		Virtual PER tracing active
	VMTRSVC	EQU	X'40'		Trace user SVC instructions
	VMTRPRG	EQU	X'20'		Trace virtual program interrupts
	VMTRIO	EQU	X'10'		Trace virtual I/O interrupts
	VMTRES	EQU	X'08'		Trace external interrupts
	VMTRPRV	EQU	X'04'		Trace user privileged instructions
	VMTRSIO	EQU	X'02'		Trace virtual I/O instructions
	VMTRBRIN	EQU	X'01'		Trace successful branches or all instructions
	VMTRINT	EQU	VMTRSVC+VMTRPRG+VMTRIO+VMTRES		Trace all user interrupts

VMBLOK

Hexadecimal Displacement	Field Name			Field Description, Contents, Meaning
5F	VMMLEVEL DS	1X	V*12	Message level
	<u>Bits defined in VMMLEVEL</u>			
	VMSGON EQU	X'80'		Receiving messages
	VMWNGON EQU	X'40'		Receiving warnings
	VMMCODE EQU	X'20'		Receiving error message codes
	VMTEXT EQU	X'10'		Receiving texts of error messages
	VMMLINED EQU	X'08'		Line editing on
	VMMACCON EQU	X'04'		Receiving accounting information
	VMMCPENV EQU	X'02'		Terminal in CP mode
	VMMSTMP EQU	X'01'		Time stamp desired on console output
60	VMQLEVEL DS	1X	V*13	Queue level
	<u>Bits defined in VMQLEVEL</u>			
	VMQ1 EQU	X'80'		Virtual machine is interactive
	VMCOMP EQU	X'40'		Virtual machine is compute bound
	VMHIPRI EQU	X'20'		Virtual machine is highest priority
	VMLOPRI EQU	X'10'		Virtual machine is lowest priority
	VMAEX EQU	X'08'		Virtual machine is assured execution
	VMAEXP EQU	X'04'		Virtual machine is assured percentage
	VMQ3 EQU	X'02'		Virtual machine using multiple Q2. Note that this bit is used only by the fair share scheduler.
	VMDROP1 EQU	X'02'		Virtual machine dropped from Q1. This bit is used only by the standard scheduler.
	VMFS EQU	X'01'		Using more than fair share flag
61	VMCLEVEL DS	1X	V*14	Command level
	<u>Bits defined in VMCLEVEL</u>			
	VMCLASSA EQU	X'80'		Privilege Class A functions
	VMCLASSB EQU	X'40'		Privilege Class B functions
	VMCLASSC EQU	X'20'		Privilege Class C functions
	VMCLASSD EQU	X'10'		Privilege Class D functions
	VMCLASSE EQU	X'08'		Privilege Class E functions
	VMCLASSF EQU	X'04'		Privilege Class F functions
	VMCLASSG EQU	X'02'		Privilege Class G functions
	VMCLASSH EQU	X'01'		Reserved for IBM use
62	VMTLEVEL DS	1X	V*15	Timer level
	<u>Bits defined in VMTLEVEL</u>			
	VMTON EQU	X'80'		Virtual timer running
	VMRON EQU	X'40'		Virtual real timer running
	VMCPUTMR EQU	X'20'		Virtual processor timer in real processor timer
	VMSTMPI EQU	X'08'		Virtual interval timer request queued
	VMSTMPT EQU	X'04'		Virtual processor timer request queued
	VMTMRINT EQU	X'01'		Interrupt from processor real timer pending
	<u>Bits redefined for System VMBLOK</u>			
	VMTIDLE EQU	VMTON		Processor timer contains idle wait state time
	VMTPAGE EQU	VMRON		Processor timer contains system page wait time
	VMTIONT EQU	VMSTMPI		Processor timer contains I/O wait state time

Hexadecimal Displacement	Field Name				Field Description, Contents, Meaning
63	VMPEND	DS	1X	V*16	Interrupt pending summary flag
	<u>Bits defined in VMPEND</u>				
	VMDEFSTK	EQU	X'80'		Deferred task wait for system lock
	VMPERPND	EQU	X'40'		Virtual PER interrupt pending
	VMPRGPND	EQU	X'20'		Virtual program interrupt deferred
	VMSVCPND	EQU	X'10'		Virtual SVC interrupt deferred
	VMPGPND	EQU	X'08'		Virtual pseudo page fault pending
	VMIOPND	EQU	X'02'		Virtual I/O interrupt pending
	VMEXTPND	EQU	X'01'		Virtual external interrupt pending
64	VMLOCKER	DS	1F		Base address of holder of VMLCK
68	VMFSTAT	DS	1X	V*19	Virtual machine feature status
	<u>Bits defined in VMFSTAT</u>				
	VMFBMX	EQU	X'80'		Virtual block multiplexer channels
	VMFAUTO	EQU	X'40'		Autocall handshake option in use
	VMFVTR	EQU	X'20'		User requested virtual timer assist enabled
	VMNPWDCL	EQU	X'04'		Virtual machine requests password suppression
69	VMMLVL2	DS	1X	V*20	Additional message handling information
	<u>Bits defined in VMMLVL2</u>				
	VMIMSG	EQU	X'80'		Receiving all informational messages
6A	VMIOINT	DS	1H		I/O interrupt pending flags
6C	VMTIMER	DS	1F		Virtual timer value - X'50'
70	VMVTIME	DS	1D		Virtual processor time used
78	VMTMOUTQ	DS	1D		Time remaining in queue 1 and/or queue 2
80	VMTTIME	DS	1D		Total time while in supervisor state
88	VMTMINQ	DS	1D		VMTTIME value at entry to queue
	VMTSOUTQ	EQU	VMTMINQ		Supervisor time allowed (redefine label)
90	VMTODINQ	DS	1D		TOD clock time stamp at queue entry
98	VMINST	DS	3H		Virtual machine privileged or tracing instruction
9E	VMUPRIOR	DS	1H	V*17	User priority from directory
9F	VMPSWDCT	DS	1X	V*18	Invalid LINK password count
A0	VMTRXT	DS	1F		Address of extended trace control block
A4	VMADSTOP	DS	1F		Address of address stop control block
A8	VMPSW	DS	1D		Virtual machine PSW
B0	VMGPRS	DS	16F		Virtual machine general purpose registers
F0	VMFPRS	DS	4D		Virtual machine floating-point registers
110	VMUSER	DS	CL8		Virtual machine identification
118	VMACNT	DS	CL8		Virtual machine accounting number
120	VMDIST	DS	CL8		Virtual machine distribution code
128	VMPGREAD	DS	1F		Total page reads
12C	VMPGWRT	DS	1F		Total page writes
130	VMWCNT	DS	1H		Page wait count
132	VMSEGDSP	DS	1H		Displacement of virtual machine SEGTABLE from start of block
134	VMSTOR	DS	1F		Permanent storage size (in bytes)
138	VMIOCNT	DS	1F		Virtual SIO count for nonspooled I/O
13C	VMPNCH	DS	1F		Virtual card count - spooled punch
140	VMLINS	DS	1F		Virtual line count - spooled printer
144	VMCRDS	DS	1F		Virtual card count - spooled reader
148	VMCOMND	DS	CL8		Last CP command executed
150	VMPDRUM	DS	1H		Reserved for IBM use
152	VMPDISK	DS	1H		Reserved for IBM use
154	VMPAGES	DS	1H		Number of real pages currently resident
156	VMPRGIL	DS	1H		ILC for latest program interrupt
158	VMDEDCH	DS	1H		Mask for dedicated channel
15A	VMQPRIOR	DS	1H		Priority in dispatching queue
15C	VMWSPROJ	DS	1H		Projected working set size

VMBLOK

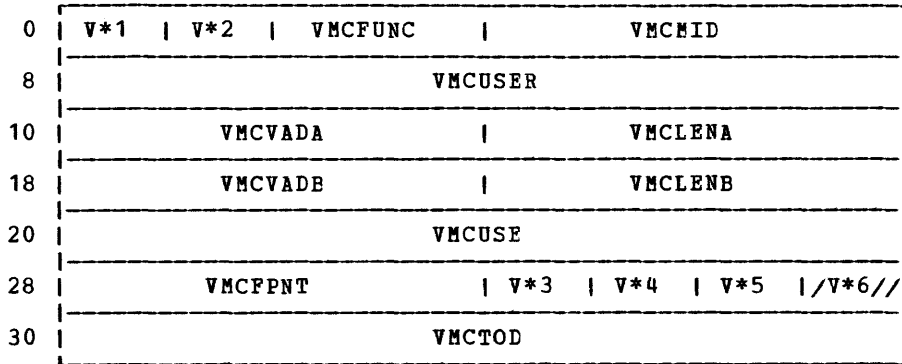
Hexadecimal Displacement	Field Name			Field Description, Contents, Meaning
15E	VMSTEALS	DS	1H	Number of waits for stolen pages
160	VMTIMEON	DS	1F	Logon time -- TOD clock word 0
164	VMTRQBLK	DS	1F	Address of TRQBLOK for real timer
168	VMACOUNT	DS	1F	Address of user ACCTBLOK
16C	VMRDINQ	DS	1F	Page read total (VMPGREAD) at queue entry
170	VMPGRINQ	DS	1F	Sum of virtual machine pages count at each page read
174	VMEPRIOR	DS	1F	Eligible list priority
178	VMSTKO	DS	1F	Console function output stack pointer
17C	VMMICRO	DS	1F	Virtual machine assist - real control R6
17C	VMMCR6	ORG DS	VMMICRO 1X	Control register 6 - hardware flag byte
	<u>Bits defined in VMMCR6</u>			
	VMMFE	EQU	X'80'	Virtual machine assist feature enabled
	VMMPROB	EQU	X'40'	Virtual machine in problem state
	VMMNOSK	EQU	X'20'	Virtual machine assist does not handle SSK, or ISK
	VMM360	EQU	X'10'	S/360 operations only; no EC mode operations allowed
	VMMSVC	EQU	X'08'	Virtual machine assist does not handle SVCs
	VMMSHADT	EQU	X'04'	Shadow tables present (EC mode and translate)
	VMMCPAST	EQU	X'02'	CP assist feature enabled
	VMMVTMR	EQU	X'01'	Virtual interval timer assist feature enabled
17D	VMMADDR	DS	3X	Control register 6 - address of virtual machine's pointer list (MICELOK)
180	VMPFUNC	DS	1F	PFnn function table
184	VMPXINT	DS	1F	Extended external interrupt stack pointer
188	VMDELAY	DS	1F	TRQBLOK for delayed SLEEP or LOGOFF
18C	VMPRIOR	DS	1F	Run list dispatching priority
190	VMPGPNT	DS	1F	Pointer to list of pages in PGBLOK
194	VMNDCNT	DS	1H	Nondeferred page read count
196	VMSHRSYS	DS	1H	Number of shared named systems
198	VMRBSC	DS	1X	V*21 Remote display line count
199	VMCXSTAT	DS	1X	V*22 VMCF status byte
	<u>Bits defined in VMCXSTAT</u>			
	VMBCAUTH	EQU	X'80'	VMCF active
	VMIOLOG	EQU	X'20'	I/O logout mask bit from control register 14. Referenced through VMVCR14.
	VMVCR14	EQU	VMCXSTAT V*22	Contains I/O logout mask bit from control register 14 (for both EC and EC mode). During EC mode, control register 14 data is also kept in the ECBLOK.
19A	VMAFF	DS	1X	V*26 Affinity request field
	<u>Bits defined in VMAFF</u>			
	VMAFFON	EQU	X'40'	Affinity set on
	VMAFFAD	DS	OBL6	Processor address
19B	VMLSTPRC	DS	1X	V*27 Last processor executed in problem state
19C	VMASST	DS	1F	Pointer to list of VMABLOKS
1A0	VMCPNT	DS	1F	VMCELOK anchor
1A4	VMCPUID	DS	3X	Processor identification number in binary
1A7	VMNOECPS	DS	1X	V*25 Reserved for IBM use
1A8	VMLOCK	DS	1F	Lock word for compare and swap locking
1AC	VMDFTPNT	DS	1F	Deferred task pointer
1B0	VMUSER1	DS	1F	Reserved for installation use
1B4	VMUSER2	DS	1F	Reserved for installation use

Hexadecimal Displacement	Field Name			Field Description, Contents, Meaning
1B8	VMUSER3	DS	1F	Reserved for installation use
1BC	VMUSER4	DS	1F	Reserved for installation use
1C0	VMUHS	DS	1F	Recent history of user processor utilization
1C4	VMPCCKP	DS	1F	User page read checkpoint
1C8	VMXPG	DS	1H	Maximum virtual machine pages count in-queue
1CA	VMQ2CNT	DS	1X	C*1 Consecutive queue 2 count
1CB	VMQ3CNT	DS	1X	C*2 Count of consecutive Q2s allowed
1CC	VMSTKCNT	DS	1H	Count of stacked ICB plus CPEXBLOCKS
1CE	VMRRCT	DS	1H	Processor related stacked CPEXBLOCKS
1D0	VMSWPMIG	DS	1F	Pointer to pseudo page table
1D4	VMVRF	DS	1X	C*3 SET FAVORED percentage for user
1D5	VMCRTO	DS	1X	C*4 COMPUTE/ELAPSED ratio
1D6	VMHRPRC	DS	1X	V*28 Processor (main or attached) whose shared segments were last used
1D7	VMGRFTAB	DS	1X	V*29 GRAF virtual console logical tab
1D8	VMCPTIME	DS	1D	Main processor supervisor time
1E0	VMAPTIME	DS	1D	Attached processor supervisor time
1E8	VMACTDEV	DS	1H	Virtual device address for last virtual SIO
1EA	VMFLPAG	DS	1H	Count of nonshared flushed pages
1EC	VMSPMFLG	DS	1X	V*30 VMCF special message flag
	<u>Bits defined in VMSPMFLG</u>			
	VMSPMON	EQU	X'40'	Receiving special messages
	VMSMSGON	EQU	X'20'	Processing special messages
1ED	RESERVED	DS	1X	Reserved for IBM use
1EE	VMCONLN	DS	1H	Bytes left in response buffer
1F0	VMCONBUF	DS	1F	Virtual address of response buffer
1F4	VMPSWDCA	DS	1X	V*31 Invalid AUTCLOG password ccunt
1F5	RESERVED	DS	3X	Reserved for IBM use
	VMBSIZE	EQU	(*-VMBLCK)/8	VMBLCK size in doublewords (X'38')

VMCBLOK

VMCBLOK: VIRTUAL MACHINE COMMUNICATION BLOCK

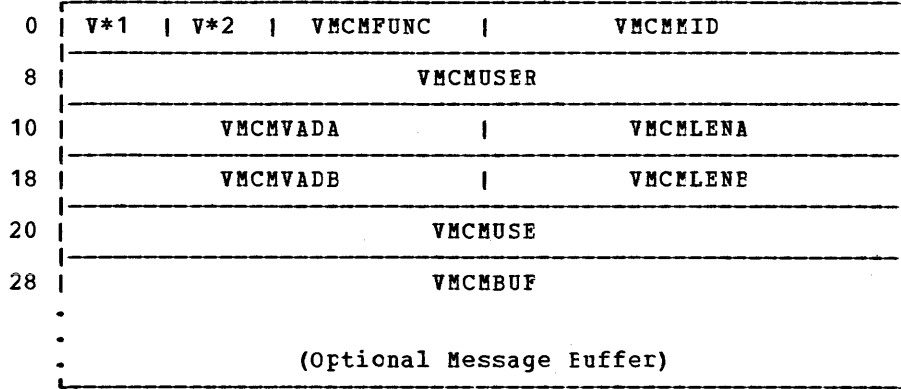
VMCBLOK contains data transfer and status information used by the Virtual Machine Communication Facility (VMCF). The VMCPNT field of the VMBLOK points to VMCBLOK.



Hexadecimal Displacement	Field Name				Field Description, Contents, Meaning
0	VMCSTAT	DS	1X	V*1	VMCBLOK user status
	<u>Bits defined in VMCSTAT</u>				
	VMCRESP	EQU	X'80'		Final response interrupt
	VMCRJCT	EQU	X'40'		Message rejected
	VMCPRTY	EQU	X'20'		Priority message
1	VMCEFLG	DS	1X	V*2	Data transfer return code
2	VMCFUNC	DS	1H		Subfunction code
4	VMCMID	DS	1F		Message identifier
8	VMCUSER	DS	1D		Source and/or sink userid (VMUSER)
10	VMCVADA	DS	1F		Vaddr of message buffer
14	VMCLENB	DS	1F		Length of message
18	VMCVADB	DS	1F		Vaddr of reply buffer (SEND/RECV only)
1C	VMCLENB	DS	1F		Length of reply buffer (SEND/RECV only)
20	VMCUSE	DS	1D		User-supplied doubleword
28	VMCFPNT	DS	1F		Address of next VMCBLOK
2C	VMCKEY	DS	1X	V*3	User PSW key
2D	VMCCSTAT	DS	1X	V*4	VMCBLOK control status
	<u>Bits defined in VMCCSTAT</u>				
	VMCCXINT	EQU	X'80'		External interrupt VMCBLOK
	VMCCRECP	EQU	X'40'		Transaction processed
	VMCCBUSY	EQU	X'20'		VMCBLOK busy
2E	VMCASTAT	DS	1X	V*5	VMCBLOK authorization status
	<u>Bits defined in VMCASTAT</u>				
	VMCAAUTS	EQU	X'80'		Authorized specific
	VMCAPRTY	EQU	X'40'		Authorized priority
	VMCAQIES	EQU	X'20'		User is quiescent
2F	VMCRSB1	DS	1X	V*6	Reserved for IBM use
30	VMCTOD	DS	1D		TOD at authorization and/or build operation
	VMCBSIZE	EQU	(*-VMCBLOK)/8		VMCBLOK size in doublewords
	<u>Redefinition for Master VMCBLOK</u>				
	VMCACNT	EQU	VMCFUNC		Active message count

VMCMHDR: VMCF MESSAGE HEADER

VMCMHDR provides information to identify the special VMCF external interrupts.



Hexadecimal Displacement	Field Name			Field Description, Contents, Meaning
0	VMCMSTAT	DS	1X	V*1 Message status byte
	<u>Bits defined in VMCMSTAT</u>			
	VMCMRESP	EQU	X'80'	Final response interrupt
	VMCMRJCT	EQU	X'40'	Message rejected
	VMCMPRTY	EQU	X'20'	Priority message
1	VMCMFLG	DS	1X	V*2 Data transfer return code
2	VMCMFUNC	DS	1H	Subfunction code (original request)
4	VMCMID	DS	1F	Message identifier
8	VMCMUSER	DS	1D	Source and/or sink userid (VMUSER)
10	VMCMVADA	DS	1F	Virtual buffer address
14	VMCMLENA	DS	1F	Message length
18	VMCMVADB	DS	1F	Virtual reply buffer address
1C	VMCMLENE	DS	1F	Reply buffer length
20	VMCMUSE	DS	1D	User supplied doubleword
28	VMCMBUF	DS	0X	Optional SENDX data buffer
	VMCMLEN	EQU	(* - VMCMHDR)	VMCMHDR size in bytes

VMCPARM

VMCPARM: VMCF PARAMETER LIST

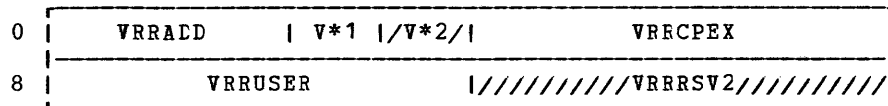
VMCPARM contains the user-supplied parameters when a VMCF subfunction is executed. Register 1 in the Diagnose instruction points to VMCPARM.

0	V*1	/V*2//	VMCPFUNC		VMCPMID
8	VMCPUSER				
10	VMCPVADA			VMCFLENA	
18	VMCPVADE			VMCFLENE	
20	VMCPUSE				

Hexadecimal Displacement	Field Name	Field Description, Contents, Meaning			
0	VMCPFLG1 DS 1X	V*1	VMCPARM flag byte		
	<u>Bits defined in VMCPFLG1</u>				
	VMCPAUTS EQU X'80'	Authcrize specific request			
	VMCPPTY EQU X'40'	Pricity message			
	VMCPMSG EQU X'20'	Receiving special messages			
1	VMCPFLG2 DS 1X	V*2	Reserved for IBM use		
2	VMCPFUNC DS 1H	Subfunction code			
	<u>Bits defined in VMCPFUNC</u>				
	VMCPRJCT EQU X'000B'	REJECT			
	VMCPIDEN EQU X'000A'	IDENTIFY			
	VMCPRESM EQU X'0009'	RESUMF			
	VMCPQUIE EQU X'0008'	QUIESCE			
	VMCPREPL EQU X'0007'	REPLY			
	VMCPCANC EQU X'0006'	CANCEL			
	VMCPRECV EQU X'0005'	RECEIVE			
	VMCPSENX EQU X'0004'	SENDX			
	VMCPSENR EQU X'0003'	SEND/RECEIVE			
	VMCPSEND EQU X'0002'	SEND			
	VMCPAUT EQU X'0001'	UNAUTHORIZE			
	VMCPAUTH EQU X'0000'	AUTHCRIZE			
4	VMCPMID DS 1F	Message identifier			
8	VMCPUSER DS 1D	Target userid			
10	VMCPVADA DS 1F	Virtual address of message buffer			
14	VMCFLENA DS 1F	Length of message			
18	VMCPVADE DS 1F	Virtual address of reply buffer (SEND/RECEIVE only)			
1C	VMCFLENE DS 1F	Length of reply buffer (SEND/RECEIVE only)			
20	VMCPUSE DS 1D	User supplied doubleword			
	VMCPLEN EQU (*-VMCPARM)	Length of VMCPARM (bytes)			

VRRBLOK: VIRTUAL RESERVE/RELEASE BLOCK

VRRBLOK is linked to the VDEVBLOK and contains information about minidisk usage. The VDEVRRB field of the VDEVBLOK points to VRRBLOK.



Hexadecimal Displacement	Field Name	Field Description, Contents, Meaning
0	VRRADD DS 1H	Reserved virtual device address
2	VRRSTAT DS 1X	V*1 Virtual RESERVE/RELEASE status flags
<u>Bits defined in VRRSTAT</u>		
2	VRRRES EQU X'80'	Minidisk is reserved
3	VRRRSV1 DS 1X	V*2 Reserved for IBM use
4	VRCPEX DS 1F	Queued request for the minidisk
8	VRRUSER DS 1F	VMPLCK of the user who has the minidisk
	reserved	
C	VRRRSV2 DS 1F	Reserved for IBM use
	VRRSIZE EQU (*-VRRBLOK)/8	Size in doublewords (X'02')

VSPLCTL

VSPLCTL: VIRTUAL SPOOLING CONTROL BLOCK

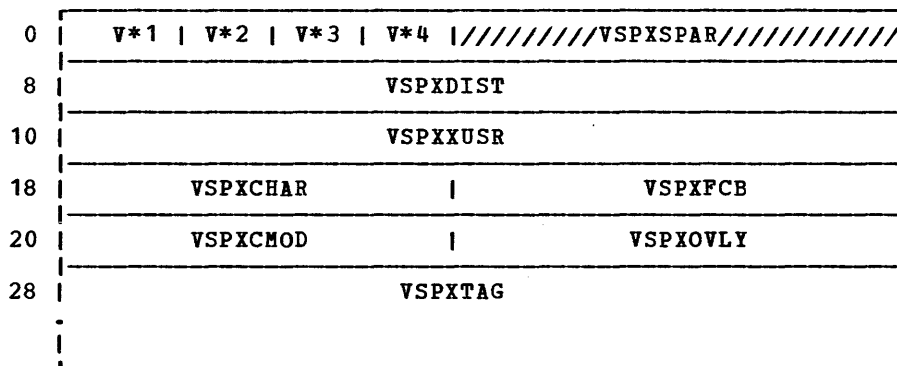
VSPLCTL is linked to the VDEVBLK and contains information for opened spool files. The VDEVSP field of the VDEVBLK points to VSPLCTL.

0	VSPCAW		VSPDPAGE
8	VSPVPAGE		VSPRECNO
10	VSPNEXT		VSPIDACT
			VSPSFBLK
18	VSPCCW		
20	VSPBUFBK		VSPMISC
28	V*1		VSPIDAL
			VSPIDAW2

Hexadecimal Displacement	Field Name			Field Description, Contents, Meaning
0	VSPCAW	DS	1F	Virtual address of user CCW
4	VSPDPAGE	DS	1F	DASD location (DCHR) of current page buffer
8	VSPVPAGE	DS	1F	Virtual address of page buffer
C	VSPRECNO	DS	1F	Records remaining in current buffer
10	VSPNEXT	DS	1H	Displacement in buffer of next record start
12	VSPIDACT	DS	1H	Data byte count of indirect data address CCW
14	VSPSFBLK	DS	1F	Pointer to SFBLK for file
18	VSPCCW	DS	1D	Current user CCW
20	VSPBUFBK	DS	1F	Address of a buffer area
24	VSPMISC	DS	1F	Use varies according to caller
28	VSPIDASW	DS	1X	V*1 Indirect data address work flag
29	VSPIDAL	DS	3X	Address of indirect data list
2C	VSPIDAW2	DS	1F	Contains IDAW2
	VSPSIZE	EQU	(*-VSPLCTL)/8	Size in doublewords (X'06')
	VSPBUFSZ	EQU	(200)/8	Size in doublewords (X'19')

VSPXBLOK: VIRTUAL SPOOL EXTENSION BLOCK

VSPXBLOK serves as an extension to the virtual spool control block (VSPCTL). It contains the user named destination of the file as well as RSCS tag information used by the Remote Spooling Communications Subsystem.

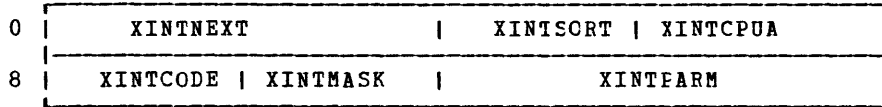


Hexadecimal Displacement	Field Name				Field Description, Contents, Meaning
0	VSPXLEN DS	1X	V*1	VSPXBLOK length (doublewords)	
1	VSPXTGLN DS	1X	V*2	VSPXTAG data length (bytes)	
2	VSPXFLSH DS	1X	V*3	3800 flash count	
3	VSPXFLG1 DS	1X	V*4	3800 flag byte	
	<u>Bits defined in VSPXFLG1</u>				
	VSPXCPYF EQU	X'80'		If this bit is set, multiple copies can be sent in one transmission	
4	VSPXSPAR DS	XL4		Reserved for IBM use	
8	VSPXDIST DS	CL8		Virtual device distribution code	
10	VSPXXUSR DS	CL8		Virtual machine user to whom the file is transferred	
18	VSPXCHAR DS	CL4		3800 character arrangement table	
1C	VSPXFCB DS	CL4		3800 forms control buffer	
20	VSPXCMOD DS	CL4		3800 copy modification	
24	VSPXOVLY DS	CL4		Name of flash overlay to be used	
	VSPXSIZ1 EQU	(*-VSPXBLOK)/8		VSPXBLOK header in doublewords	
28	VSPXTAG DS	CL136		Tag application data area	
	VSPXSIZ2 EQU	(*-VSPXBLOK)/8		VSPXBLOK size in doublewords	

XINTELOK

XINTELOK: EXTERNAL INTERRUPT BLOCK

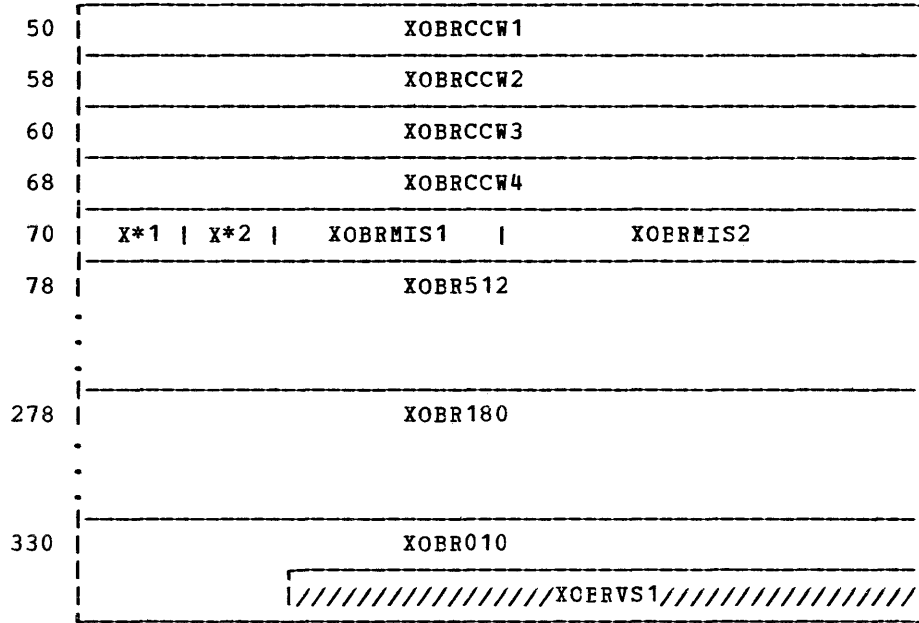
XINTELOK saves the various types of external interrupts that are presented to the virtual machine. If multiple external interrupt conditions are simultaneously presented, as indicated by values presented in the block, code exists for handling the interrupts in their defined hierarchical order. The VMPXINT field of the VMBLCK points to XINTELOK.



Hexadecimal Displacement	Field Name	Field Description, Contents, Meaning
0	XINTNEXT DS 1F	Address of next external interrupt block
4	XINTSORT DS 1H	Left half of interrupt collating key
6	XINTCPUA DS 1H	Right half of interrupt collating key
8	XINTCODE DS 1H	External interrupt code
A	XINTMASK DS 1H	Control register 0 mask (bits 16-31)
C	XINTPARM DS 1F	External interrupt parameter word
	XINTSIZE EQU (*-XINTBLOK)/8	XINTELOK size in doublewords (X'02')

XOBR3211: EXTENDED OUTBOARD RECORDING BLOCK

XOBR3211 is appended to the IOERBLOK to contain sense data and other data associated with I/O errors and error recovery for devices that generate more than 24 bytes of sense information.



Hexadecimal Displacement	Field Name			Field Description, Contents, Meaning
50	XOBRCCW1	DS	1D	CCW used to read OBR information
58	XOBRCCW2	DS	1D	CCW used to read OBR information
60	XOBRCCW3	DS	1D	CCW used to read OBR information
68	XOBRCCW4	DS	1D	CCW used to read OBR information
70	XOBRFLAG	DS	1X	X*1 Outboard recording flag byte
<u>Bits defined in XOBRFLAG</u>				
	XOBRRT1	EQU	X'80'	T1 Buffer type information present
	XOBRRT2	EQU	X'40'	T2 Buffer type information present
	XOBRRT3	EQU	X'20'	T3 Buffer type information present
71	XOBRSTAT	DS	1X	X*2 Outboard recording status byte
<u>Bits defined in XOBRSTAT</u>				
	XOBRRT1	EQU	X'80'	Perform routine 1 in error module
	XOBRRT2	EQU	X'40'	Perform routine 2 in error module
	XOBRRT3	EQU	X'20'	Perform routine 3 in error module
	XOBRRT4	EQU	X'10'	Perform routine 4 in error module
	XOBRRT5	EQU	X'08'	Perform routine 5 in error module
	XOBRRT6	EQU	X'04'	Perform routine 6 in error module
	XOBRRT7	EQU	X'02'	Perform routine 7 in error module
	XOBRRT8	EQU	X'01'	Perform routine 8 in error module

XOBR3211

Hexadecimal Displacement	Field Name			Field Description, Contents, Meaning
72	XOBRMIS1	DS	1H	Used by the error routine
74	XOBRMIS2	DS	1F	Used by the error routine
78	XOBR512	DS	CL512	Space for USCB data
278	XOBR180	DS	CL184	Space for FCB data
330	XOBR010	DS	CL10	Space for first ten error characters
33A	XOERSV1	DS	CL6	Reserved for IBM use
		ORG	XOBR180	Redefinition for PLB check
278	XOBR150	DS	CL150	Space for PLB check data
	XOBRSIZE	EQU	(*-IOERBLOK)/8	Size of IOER and XOBR in dcubewords (X'67')
	XOBRXT	EQU	(*-XOBRCCW1)/8	Size of XOBR3211 in doublewrds (X'5E')

Section 2. CMS Data Areas and Control Blocks

This section contains CMS data areas and control blocks. Figure 2 shows the relationships between the control blocks and data areas of CMS.

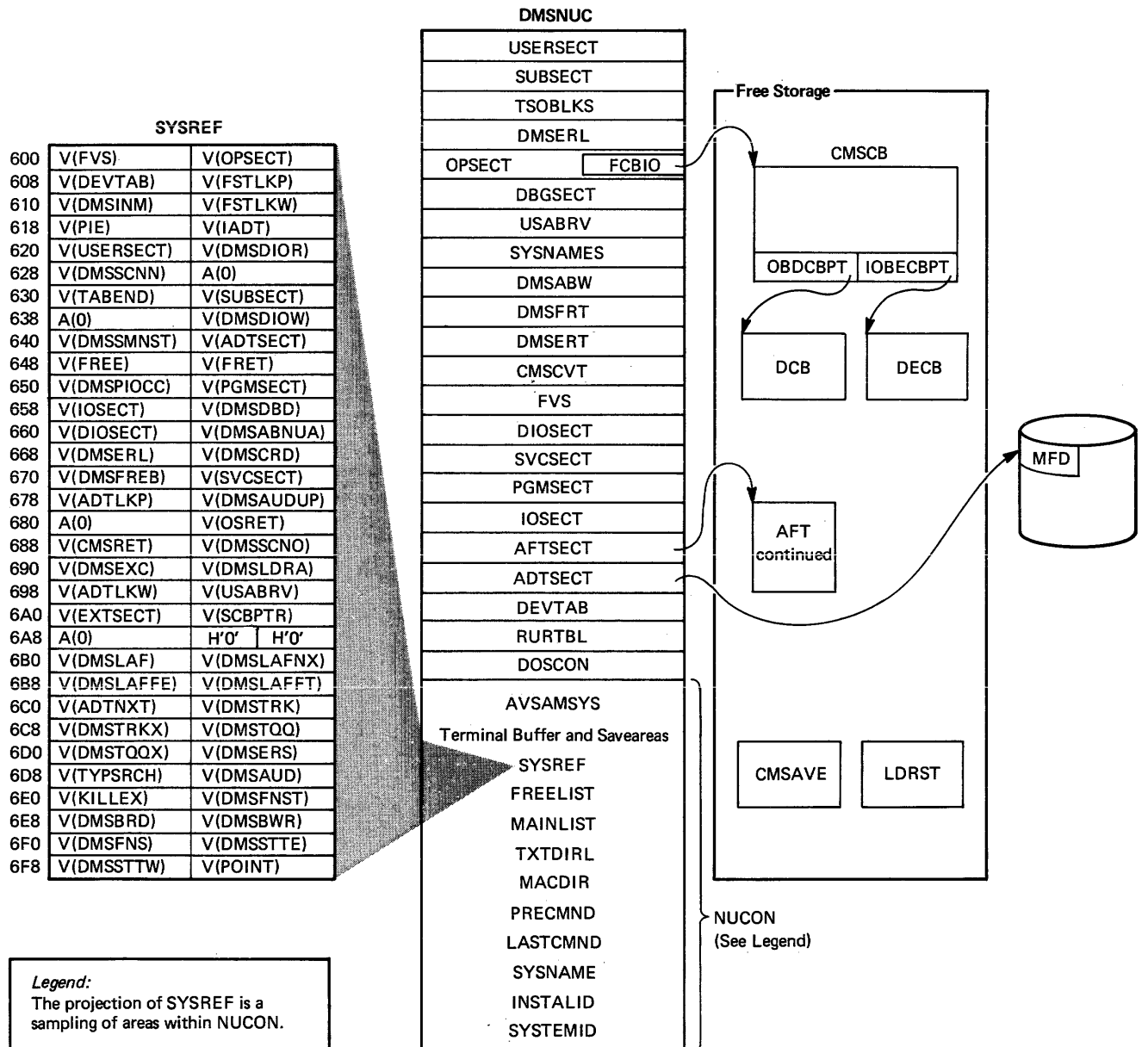
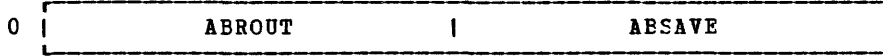


Figure 2. CMS Control Block Relationships

AETAB

AETAB: ABEND TERMINATION OPTION TABLE

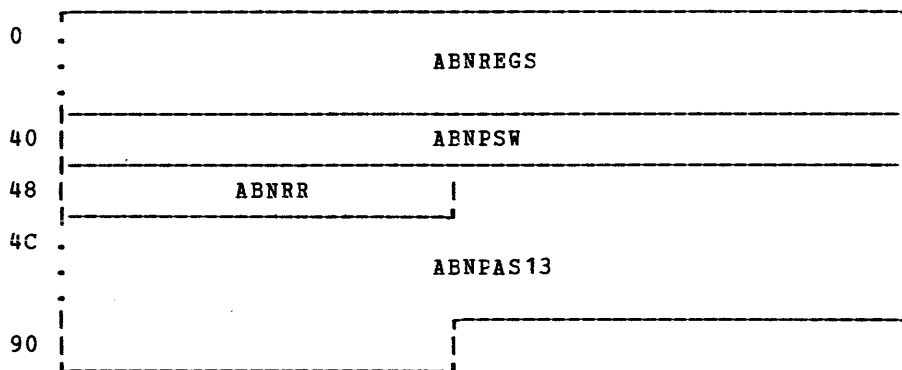
AETAB contains one 8-byte entry for the background partition. Bytes 0-3 contain the address of the entry point of the user's abnormal termination routine. Bytes 4-7 contain the address of a 72-byte save area used by the supervisor to store the interrupt status information and the contents of the general registers. The IJBABTAB field in the SYSCOM block in the DOSCON CSECT of NUCON points to the AETAB block.



Hexadecimal Displacement	Field Name			Field Description, Contents, Meaning
0	ABROUT	DC	F'0'	Address of user termination routine
4	ABSAVE	DC	F'0'	Address of supervisor save area
	ABTAB	EQU	*-8	Abend option table

ABWSECT: ABEND RECOVERY WORKSPACE

ABWSECT describes the fields used for saving registers and other data during abend recovery. V-constants in DMSABN, DMSDEG, DMSFRE, DMSITI, DMSITP, and DMSITS point to the ABWSECT block. ABWSECT is defined in module DMSAPW.



Hexadecimal Displacement	Field Name			Field Description, Contents, Meaning
0	ABNREGS	DS	16F	Registers at time of abend
40	ABNPSW	DS	D	PSW at time of abend
48	ABNRR	DS	F	Temporary save area
4C	ABNPAS13	DS	18F	Area passed to nucleus routines
<u>Space for DMSERR PLIST</u>				
94		ORG	ABNPAS13	
4C	ABNERLST	DS	47X	

ADTSECT

ADTSECT: ACTIVE DISK TABLE

ADTSECT describes the attributes of virtual disks (A-G, S, Y, Z) accessed by a virtual machine via the ACCESS command. Space is allocated for the ADT when DMSNUC is assembled. In the ADT, certain fields are defined for use by both CMS and OS. For example, ADTHECT field at displacement 1C (hexadecimal) into ADTSECT is also defined as OSACTVTA for use by OS simulation routines. ADTSECT is invoked by the ACT macro.

0	ADTID	A*1	A*2
8	ADTPTR	ADTDTA	
10	ADTFDA	ADTMFDN	
18	ADTMFDA	ADTHECT	
20	ADTFSTC	ADTCHBA	
28	ADTCFST	ADT1ST	
30	ADTNUM	ADTUSEL	
38	ADTLEFT	ADTLAST	
40	ADTCYL	A*3	A*4 A*5 A*6
48	ADTMSK	ADTQQM	
50	ADTPQM1	ADTPQM2	
58	ADTPQM3	ADTLHBA	
60	ADTLFST	ADTNACW	ADTRES
68	ADTXNREC	ADTXAREC	

Hexadecimal Displacement	Field Name	Field Description, Contents, Meaning			
<u>Needed for Read-Only Disks and Read/Write Disks</u>					
0	ADTID DS CL6	Disk identifier (label)			
6	ADTFGL3 DS 1X A*1	Third flag byte			
<u>Bits defined in ADTFGL3</u>					
	ADTFUPD1 EQU X'80'	First half of DMSAUD has been called			
	ADTFXCHN EQU X'40'	Extra chain link(s) to be returned			
	ADTFRWOS EQU X'20'	Read/write CS or DCS disk			
	ADTFSORT EQU X'10'	All FST hyperblocks and FST entries sorted			
	ADTFORCE EQU X'08'	CMS/DCS/OS disk forced to a read-only			
	ADTFNOAB EQU X'04'	For DMSAUD routine: Do not abend if it is a disk error			
7	ADTFITYP DS 1X A*2	Filetype flag byte			
8	ADTPTR DS 1A	Pointer to next ADT block in chain			
C	ADTDTA DS 1A	Device table address in NUCCN			
10	ADTFDA DS 1A	File directory (PSTAT) address			
14	ADTN DS 1F	Number of doublewords in master file directory			
18	ADTMFDA DS 1A	Master file directory address			
1C	OSACTVTA DS 0F	VTOC address of OS pack			
1C	ADTHBCT DS 1F	FST hyperblock count			

Hexadecimal Displacement	Field Name			Field Description, Contents, Meaning
20	ADTFSTC	DS	1F	Number of FST 40-byte entries (files)
24	OSADTFST	DS	0F	Address of first OS FST
24	ADTCHBA	DS	1A	Pointer to current FST hyperblock
28	OSADTVTB	DS	0F	Upper VTOC address of OS pack
28	ADTCFST	DS	1F	Displacement of current FST entry
2C	ADT1ST	DS	1F	Displacement of first word in bit mask with "hole"
30	OSADTDSK	DS	0F	OS disk address (cuu)
30	ADTNUM	DS	1F	Number of records (NUMTRKS)
34	OSADTSV1	DS	0F	Save area
34	ADTUSED	DS	1F	Number of records in use (QTUSEDP)
38	ADTLEFT	DS	1F	Number of records left (QTLLEFTP)
3C	ADTLAST	DS	1F	Displacement of last nonzero byte in bit mask
40	ADTCYL	DS	1F	Number of cylinders on disk (NUMCYLP)
44	ADTM	DS	1C	A*3 Mode letter (A,B,C,...,S,Y,Z)
45	ADTMX	DS	1C	A*4 Extension-of-mode letter (A,B,C,...,S,Y,Z)
46	ADTFLG1	DS	1X	A*5 First flag byte

Bits defined in ADTFLG1

ADTFSP	EQU	X'80'	ADT block in free storage
ADTFRO	EQU	X'40'	CMS read-only disk (attached and ready)
ADTFRW	EQU	X'20'	CMS read/write disk (attached and ready)
ADTFFSTF	EQU	X'10'	First FST hyperblock is in free storage
ADTFFSTV	EQU	X'08'	FST hyperblocks are of varying lengths
ADTFQQF	EQU	X'04'	200-byte QQMSK is in free storage
ADTROX	EQU	X'02'	This disk has read-only extension(s)
ADTFMIN	EQU	X'01'	ADT block is minimum size

47 ADTFLG2 DS 1X A*6 Second flag byte

Bits defined in ADTFLG2

ADTFMFD	EQU	X'80'	MFD is in storage
ADTFALNM	EQU	X'40'	All filenames are in storage
ADTFALTY	EQU	X'20'	All filetypes are in storage
ADTFMDRO	EQU	X'10'	Modes 1 through 5 are in storage
ADTFALMD	EQU	ADTFMDRO+X'08'	All modes (0 through 5) are in storage
ADTFALUF	EQU	ADTFMFD+ADTFALNM+ADTFALTY+ADTFALMD	All UFD is in storage
ADTPSTM	EQU	X'02'	ADT FSTAT chain modified
ADTFROS	EQU	X'04'	Indicates this is an OS disk
ADTFDOS	EQU	X'01'	Indicates this is a DOS disk

48 ADT2ND DS 0D

Needed for Read/Write Disks

48	ADTMSK	DS	1A	800-byte (PQMSK) bit-mask address
4C	ADTQQM	DS	1A	200-byte (PQQMSK) bit-mask address
50	ADTPQM1	DS	1F	PQMSIZ equals number of bytes in PQMSK is greater than 215
54	ADTPQM2	DS	1F	PQMNUM equals number of 800-byte records for PQMSK
58	ADTPQM3	DS	1F	RONUM equals number of doublewords in PQMSK
5C	ADTLHBA	DS	1A	Pointer to last FST hyperblock (see Note)
60	ADTLFST	DS	1F	Displacement of last FST in last hyperblock (see Note)
64	ADTNACW	DS	1H	Number of active write files -- halfword
66	ADTRES	DS	1H	Reserve count (RESRVCNT) -- halfword

Note: Applies to all CMS disks except for the S-disk.

68 ADTXNREC DS 1F Number of doublewords of extra chain link records

ADTSECT

Hexadecimal Displacement	Field Name		Field Description, Contents, Meaning
6C	ADTXAREC	DS 1F	Address of block of extra chain link records
	ADTLBM	EQU ADT2ND-ADTSECT	Length of minimum ADT block (in bytes)
	ADTLDM	EQU ADTLBM/8	Length of minimum ADT block in doublewords
	ADTLB	EQU *-ADTSECT	Length of full ADT block (in bytes)
	ADTLD	EQU (ADTLB+7)/8	Length of full ADT block in doublewords
<u>Other Parameters</u>			
	ADTRL	EQU 800	Logical record length
	ADTMXBML	EQU 10	Maximum bit map length (number of records) for 3330
<u>NUCON Device Table Displacements</u>			
	DTAD	EQU 0	Device number
	DTADT	EQU 3	Device type byte
	DTAS	EQU 4	Symbolic device name

AFTSECT: ACTIVE FILE TABLE

AFTSECT is used to describe a file currently open for a read or write. The AFT is created when a file is opened. Space for up to five AFTs is available in DMSNUC; any others must reside in free storage. AFTSECT is invoked via the AFT macro.

0	AFTCLD		AFTCLN		AFTCLA
8	AFTDBD		AFTDBN		AFTDEA
10	AFTCLB				
60	A*1		AFTPFST		AFTIN AFTID
68	AFTFCLA				AFTFCLX AFTCLDX
70	A*2		//////		AFTOCLDX
78	AFTN				
80	AFTT				
88	AFTD			AFTWP	AFTRP
90	AFTM		AFTIC		AFTFCL A*3 A*4
98	AFTIL			AFTIEC	AFTYR
A0	AFTADT			AFTPTR	

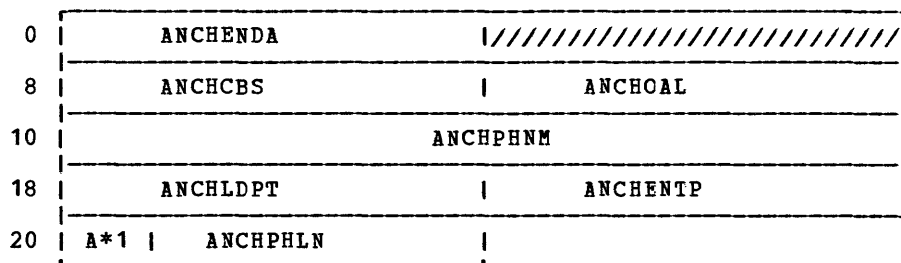
Hexadecimal Displacement	Field Name			Field Description, Contents, Meaning
<u>Active File Table Block</u>				
0	AFTSTART	DS	0D	
0		DC	41F'0'	First AFT block
A4		DC	A(**4)	
A8		DC	41F'0'	Second AFT block
14C		DC	A(**4)	
150		DC	41F'0'	Third AFT block
1F4		DC	A(**4)	
1F8		DC	41F'0'	Fourth AFT block
29C		DC	A(**4)	
2A0		DC	41F'0'	Fifth AFT block
344		DC	A(0)	Address of next AFT block (in free storage)
348		DC	2F'0'	Reserved for IBM use
<u>Active File Table</u>				
0	AFTCLD	DS	H	Disk address of current chain link
2	AFTCLN	DS	H	Number of current chain link
4	AFTCLA	DS	F	Address of chain link buffer
8	AFTDBD	DS	H	Disk address of current data block
A	AFTDBN	DS	H	Number of current data block
C	AFTDBA	DS	F	Address of current data block
10	AFTCLB	DS	XL80	Chain link buffer from first chain link

AFTSECT

Hexadecimal Displacement	Field Name			Field Description, Contents, Meaning
60	AFTFLG	DS	X	A*1 Flag byte
<u>Bits defined in AFTFLG</u>				
	AFTUSED	EQU	X'80'	Active file table block in use
			X'40'	Reserved for IMB use
	AFTICF	EQU	X'20'	First chain link in storage
	AFTFBA	EQU	X'10'	Full buffer assigned
	AFTDBF	EQU	X'08'	Data block in storage
	AFTWRT	EQU	X'04'	Active write
	AFTRD	EQU	X'02'	Active read
	AFTFULD	EQU	X'01'	Full disk special case
61	AFTPFST	DS	3X	Pointer to (static) FST entry
64	AFTIN	DS	H	Current item number
66	AFTID	DS	H	Displacement of current item in data block
68	AFTFCLA	DS	F	Address of first chain link
6C	AFTFCLX	DS	H	Disk address of swapped FCL
6E	AFTCLDX	DS	H	Disk address of swapped chain link
70	AFTFLG2	DS	X	A*2 Secnd flag byte
<u>Bits defined in AFTFLG2</u>				
	AFTNEW	EQU	X'80'	New file
	AFTOLDCL	EQU	X'40'	Current chain link existed previously
	AFTCLX	EQU	X'20'	Alternate chain link assigned/implied
	SAMLEN	EQU	X'01'	Force same length update
71		DS	1X	Reserved for IBM use
72	AFTOCLDX	DS	1H	Old value (if any) of AFTCLDX
74		DS	F	Reserved for IBM use
<u>Copy of FST Block Embedded in AFT Block</u>				
78	AFTFST	DS	OD	
78	AFTN	DS	D	Filename
80	AFTT	DS	D	Filetype
88	AFTD	DS	F	Date/time last written
8C	AFTWP	DS	H	Write pointer (item no.)
8E	AFTRP	DS	H	Read pointer (item no.)
90	AFTM	DS	H	Filemode
92	AFTIC	DS	H	Item count
94	AFTFCL	DS	H	First chain link
96	AFTFV	DS	C	A*3 Fixed(F)/variable(V) flag
97	AFTFB	DS	X	A*4 FST flag byte
98	AFTIL	DS	F	(Maximum) item length
9C	AFTDBC	DS	H	800-byte data block count
9E	AFTYR	DS	H	Year
A0	AFTADT	DS	F	Pointer to active disk table
A4	AFTPTR	DS	F	Pointer to next AFT block in chain
<u>Bit defined in AFTPTR</u>				
	AFTFSF	EQU	X'40'	Indicates in free storage
A8		DS	OD	End of DSECT
	AFTLB	EQU	*-AFTSECT	Length of AFT block in bytes
	AFTLD	EQU	AFTLB/8	Length of AFT block in doublewords

ANCHSECT: ANCHOR TABLE

ANCHSECT defines the DOS/VS anchor table. This DSECT is used by DMSDCS when a CDLOAD (SVC 65) is issued, and the phase is not found in either the CMSVSAM or CMSAMS segment. In this case, the specified phase is loaded either from a CMS DOSLIB or a DOS core image library, and the name, load point, entry point, and the length in bytes, of the phase are saved in an available slot in the anchor table. ANCHSECT is invoked by the ANCHTAP macro.



Hexadecimal Displacement	Field Name	Field Description, Contents, Meaning		
0	ANCHENDA DC	A(0)	End address of anchor table	
4	DC	F'0'	Reserved for IBM use	
8	ANCHCBS DC	A(0)	Pointer to VSAM AMCB table	
C	ANCHOAL DC	A(0)	Pointer to VSAM OAL (OPEN ACE) table, which is followed by one or more Anchor Table entries	
<u>Anchor Table Entries and Their Format</u>				
10	ANCHPHNM DC	CL8'	Phase name	
18	ANCHLDPT DC	A(0)	Load point	
1C	ANCHENTP DC	A(0)	Entry point	
20	ANCHSTSW DC	X'00'	A*1	Status switch
<u>Bits defined in ANCHSTSW</u>				
	ANCHMLOD EQU	X'00'	Phase must be loaded	
	ANCHINST EQU	X'7F'	Phase is already in storage	
	ANCHRPJL EQU	X'FF'	Requested phase just loaded by another task (only if AP=YES)	
	ANCHLENG EQU	20	Length of one anchor table entry	
	ANCHSIZ EQU	1024	Default size of anchor table (in bytes)	
21	ANCHPLN DC	AL3(0)	Length of phase in bytes	

BATLSECT 146 BBOX

BATLSECT: CMS BATCH USER JOB LIMITS

BATLSECT describes the fields in the user job limits table for CMS batch jobs. The BEATLIMT field in NUCON points to BATLSECT.

0	BATCPUL	BATCPUC	BATPRTL	BATPRTC
8	BATPUNL	BATPUNC		

Hexadecimal Displacement	Field Name	Field Description, Contents, Meaning		
0	BATCPUL DC H'32767'	Virtual processor limit (in seconds); can be reset		
2	BATCPUC DC H'0'	Current processor count; dc nct reset		
4	BATPRTL DC H'32767'	Number printed lines limit; can be reset		
6	BATPRTC DC H'0'	Current line count; do not reset		
8	BATPUNL DC H'32767'	Number punched cards limit; can be reset		
A	BATPUNC DC H'0'	Current card count; do not reset		

EPOX: BOUNDARY BOX

EEOX contains the begin and end addresses of the virtual and real partitions, respectively. The IJBBOX field in the SYSCOM block in the DOSCCN CSECT of NUCON points to the BBOX block.

0	REALORP	REALNDP
8	VIRTORP	VIRTENDP

Hexadecimal Displacement	Field Name	Field Description, Contents, Meaning		
0	REALORP DS XL4	Origin of real partition		
4	REALNDP DS XL4	End of real partition +1		
8	VIRTORP DS XL4	Origin of virtual partition		
C	VIRTENDP DS XL4	End of virtual partition +1		
<u>Equates for Fields in BBOX</u>				
	ALTS DAR EQU REALORP	Address of alternate area		
	MINREALP EQU REALNDP	No. page frames in minimum real partition		
	MPGEPOOL EQU REALNDP+2	No. page frames in main page pool		
	ORIGVIRT EQU VIRTORP	Origin of virtual storage		
	FINVIRT EQU VIRTENDP	End of virtual storage +1		

BGCOM: DOS/VS PARTITION COMMUNICATION REGION

BGCOM simulates the DOS/VS Partition Communication Region (BGCOM). The ABGCOM field in NUCON points to the BGCOM block.

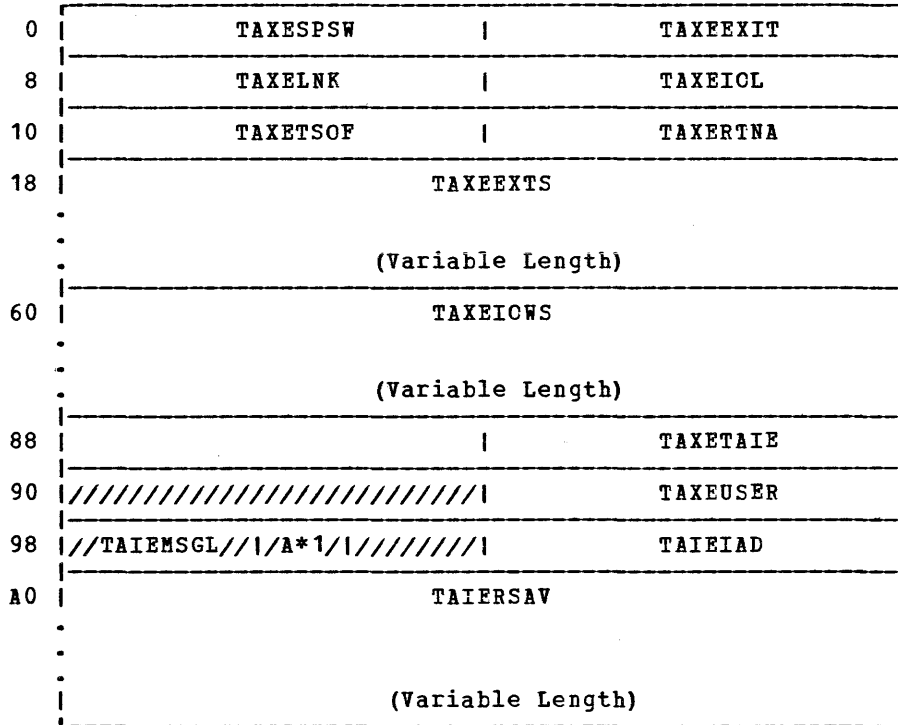
0	JOBDATE					
8	PPBEG		EOSSP			
10						UPSI
18	COMNAME					
20	PPEND		HIPHAS			
28	HIPROG		LAELEN		PIK	
30	EOCADR		A*1		A*2	A*3 A*4
38	A*5		A*6		A*7	A*8 DALC FCCLPT
40	PUBPT		FAVPT		JIBPT	TEBPT
48	FICLPT		NICLPT		LUEFT	A*9
50	MMDD		YYDD			
58	LIOSCOM		PIBPT		CHKPTID	JCBZCN
60	DIBPT		//////////		PCPTR	ITPTR
68	OCPT		PWTIMS		//////////	LTK
70	SYSPAR		JAPART			
78	TODCOM		PIB2PTR		PDTAEB	
80	IJLQTTAD		BGCCMPT		A*10	A*11
88	COMEX		A*12		A*13	A*14
90	PROCNAM (cont. from 8F)					A*15
98	POVNAM					A*16

Hexadecimal Displacement	Field Name	Field Description, Contents, Meaning
0	JOBDATE DC C'00/00/00'	Job date
8	PPBEG DC S(0)	Supervisor end
A	EOSSP DC S(0)	End of storage protection
C	DC 11X'00'	User scratch area
17	UPSI DC X'00'	UPSI byte
18	COMNAME DC CL8'CMS/DOS'	Job name
20	PPEND DC A(0)	Highest storage address of partition

Hexadecimal Displacement	Field Name			Field Description, Contents, Meaning
24	HIPHAS	DC	A(0)	End address of last phase loaded
28	HIPROG	DC	A(0)	End address of longest phase loaded
2C	LABLEN	DC	H'0'	Length of problem program label area
2E	PIK	DC	X'0010'	Program interrupt key
30	EOCADR	DC	A(0)	End of virtual storage address
34	CONFIG	DC	B'11101000'	A*1 Machine configuration byte
35	LTACT	DC	B'00010000'	A*2 System configuration byte
36	SOB1	DC	B'11000100'	A*3 Standard language translator options
37	SOB2	DC	B'11011010'	A*4 Standard supervisor options
38	JCSW1	DC	B'11010000'	A*5 Job control byte
39	JCSW2	DC	B'00000000'	A*6 Linkage control byte
3A	JCSW3	DC	B'11000100'	A*7 Nonstandard language translator options
3B	JCSW4	DC	B'10000000'	A*8 Job duration indicator byte
3C	DALC	DC	H'0'	Disk address of volume label
3E	FOCLPT	DC	S(0)	Address of FOCL
40	PUBPT	DC	S(0)	Address of PUB
42	FAVPT	DC	S(0)	Address of FAVP
44	JIBPT	DC	S(0)	Address of JIB
46	TEBPT	DC	S(0)	Address of TEB
48	FICLPT	DC	S(0)	Address of FICL
4A	NICLPT	DC	S(0)	Address of NICL
4C	LUBPT	DC	S(0)	Address of LUB
4E	SYSLINE	DC	AL1(56)	A*9 SYSLSLST line count
4F	SYSDATE	DS	0CL9	System date
4F	MMDD	DC	XL4'00'	MMDD or DDMM
53	YYDD	DC	XL5'00'	YYDD portion of date
58	LIOCSCOM	DC	2X'00'	LIOCS communication bytes
5A	PIBPT	DC	S(0)	Address of PIB
5C	CHKPTID	DC	H'0'	Last checkpoint number
5E	JOBZON	DC	S(0)	Job zone in minutes
60	DIBPT	DC	S(0)	Background DIB pointer
62		DC	H'0'	Reserved for IBM use
64	PCPTR	DC	S(0)	PC option table
66	ITPTR	DC	S(0)	IT option table
68	OCPT	DC	S(0)	OC option table
6A	PWTIMS	DC	X'0000'	Key of program with IT support
6C		DC	H'0'	Reserved for IBM use
6E	LTK	DC	S(0)	Logical transient key
70	SYSPAR	DC	F'0'	Address of SYSPARM
74	JAPART	DC	F'0'	Address of job accounting table
78	TODCOM	DC	A(0)	Address of TOD communications area
7C	PIB2PTR	DC	S(0)	Address of PIB extension
7E	PDTABB	DC	S(0)	Address of MICR DTF table
80	IJLQTTAD	DC	A(0)	Address of QTAM vector table
84	BGCOMPT	DC	S(0)	Address of background COMREG
86	OPTNBYTE	DC	X'00'	A*10 Option indicator byte
87	RMSROPEN	DC	B'00000000'	A*11 System configuration byte 2
88	COMEX	DC	A(0)	Pointer to SYSCOM option table
8C	STDOPT	DC	B'01000000'	A*12 Standard job control option byte
8D	TEMOPT	DC	B'01000000'	A*13 Temporary job control option byte
8E	DISKCONF	DC	X'00'	A*14 Disk configuration byte
8F	PROCNAM	DC	CL8' '	Procedure name
97	PSWTCH	DC	X'0'	A*15 Interface byte for catalog procedure
98	POVNAM	DC	CL7' '	Save area for statement name
9F	INSIZE	DC	X'0'	A*16 81-byte SYSIN indicator

CMSTAXE: TERMINAL ATTENTION EXIT ELEMENT

CMSTAXE defines the fields used in a Terminal Attention Exit Element (TAXE). The TAXE is used mainly by DMSCIT for processing attention interrupts. CMSTAXE is invoked via the TSOBLKS macro. The TAXEADDR field in NUCON points to CMSTAXE.

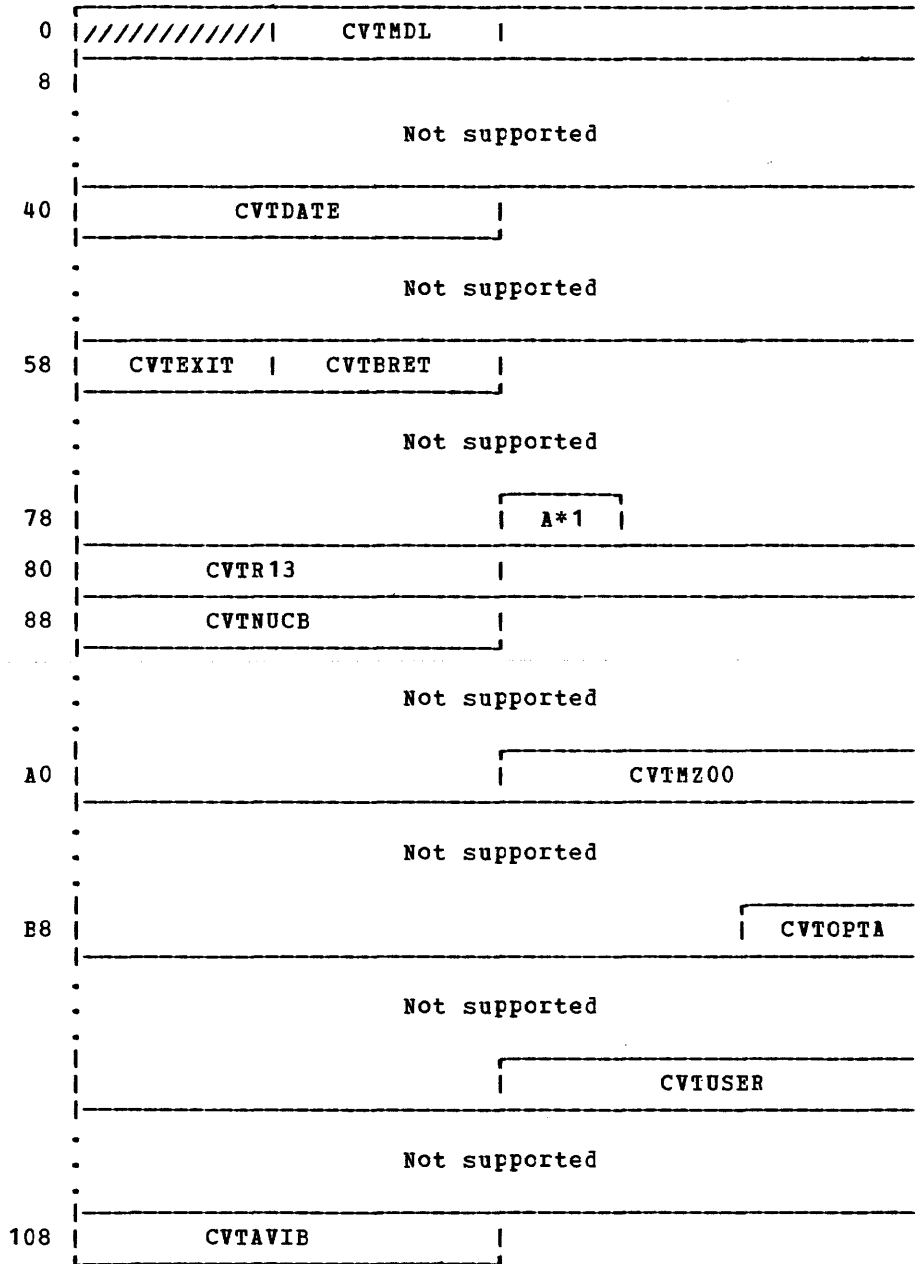


Hexadecimal Displacement	Field Name			Field Description, Contents, Meaning
0	TAXESPSW	DS	1F	Left half PSW for ATTN return
4	TAXEEXIT	DS	A	ATTN exit address
8	TAXESTAT	DS	0X	Status of exit return
	TAXEFREQ	EQU	X'80'	ATTN exit taken
8	TAXELNK	DS	A	Next TAXE on queue
C	TAXEIOL	DS	1F	Left half of I/C old PSW
10	TAXE TSOF	DS	1F	TSO flags saved here
14	TAXERTNA	DS	A	Return address
18	TAXEEXTS	DS	18F	ATTN exit routine save area
60	TAXEIEWS	DS	11F	DMSICW save area
<u>Also Attention Exit Parameter List</u>				
8C	TAXETAIE	DS	A	Address of TAIE
90		DS	1F	Reserved for IBM use
94	TAXEDEF	DS	0X	Defer indicator
94	TAXEUSER	DS	A	User PLIST address
<u>Terminal Attention Interrupt Element (TAIE)</u>				
98	TAIEMSGL	DS	2X	Reserved for IBM use
9A	TAIETGET	DS	1X	A*1 Reserved for IBM use
9B		DS	1X	Reserved for IBM use
9C	TAIEIAD	DS	A	Right half of I/O old PSW
A0	TAIERSAV	DS	16F	Registers 0-15 of interrupted program

CVTSECT

CVTSECT: COMMUNICATION VECTOR TABLE AS SUPPORTED BY CMS

CVTSECT simulates the OS Communication Vector Table. CVTSECT is invoked via the CMSCVT macro. The ACMSCVT field in NUCON points to CMSCVT.



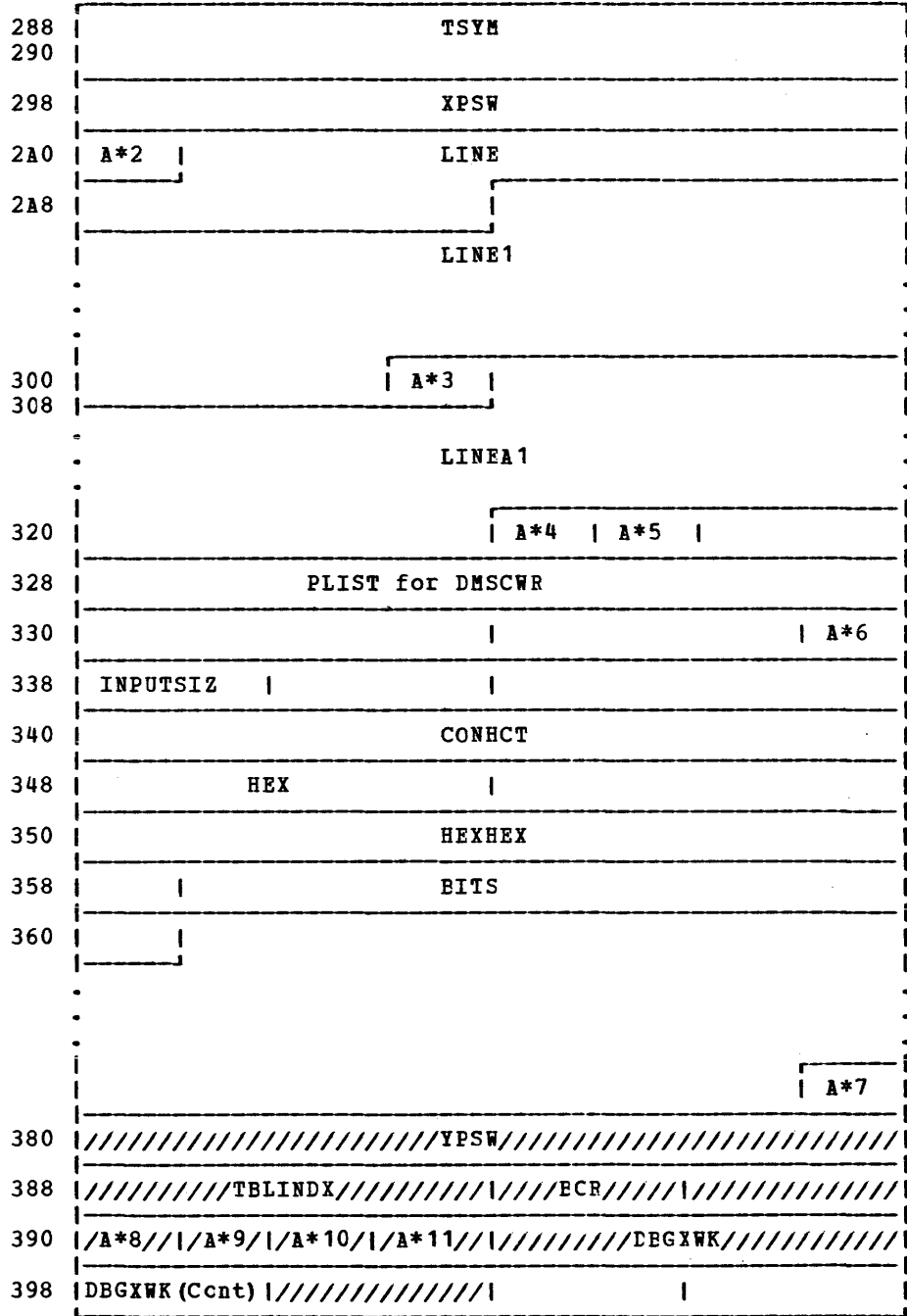
Hexadecimal Displacement	Field Name			Field Description, Contents, Meaning
0		DC	H'0'	Reserved for IBM use
2	CVTMDL	DC	H'0'	Processor identification and model number
4		DC	CL4'Cx.x'	CMS release number (latest release level)
8	CMSCVT	DS	0D	CVT start
8		DC	14F'-1'	Not supported
40	CVTDATE	DC	PL4'0'	Current date in packed decimal
44		DC	3F'-1'	Not supported
50		DC	A(0)	Not supported
54		DC	F'-1'	Not supported
58	CVTEXIT	DC	XL2'0A03'	A SVC 3 instruction (exit)
5A	CVTBRET	DC	XL2'07FE'	A BCR 15,14 instruction
5C		EC	8F'-1'	Not supported
7C	CVTDCB	DC	XL1'40'	A*1 System configuration = PCP
7D		DC	FL3'-1'	Not supported
80	CVTR13	DC	F'0'	R13 saved during OPEN
84		DC	F'-1'	Not supported
88	CVTNUCB	DC	A(0)	Lowest storage address not in nucleus
8C		EC	8F'-1'	Not supported
AC	CVTMZ00	DC	A(0)	Highest storage address in machine
B0		DC	3F'-1'	Not supported
BC		EC	XL2'00'	Not supported
BE	CVTOPTA	DC	XL2'00'	Bit 7: extended precision floating point hardware in processor
C0		DC	2F'-1'	Not supported
C8		EC	3A(0)	Not supported
D4	CVTUSER	DC	F'0'	Field available to user
D8		DC	12F'-1'	Not supported
108	CVTAVIB	DC	V(DMSVIB)	Address of VSAM interface bootstrap

DEGSECT

DEGSECT: DEBUG WORK AREA

DEGSECT contains the files used by DEBUG for saving registers, breakpoints, PSWs, and other data. V-constants in DMSDBD, DMSDBG, and LMSITE point to the DEBUG work area.

0	CCWPRINT	
8	SYMTABLE	
108	SYMTBG	
		BRKPNBTL
188		TELEND
190	ARGS	
1B8	JFLAGS	A*1
1C0	WAITRD	
1C8		WTRDCNT
1D0	ORG	
1D8	DEC	
1E0	DECDEC	
1E8	ARGSAV	RETSAV
1F0	EXAMLC	EXAMLG
1F8	BEGAT	STOPAT
200	SAVE1	SAVE2
208	DEGSAV1	
248	DEGSAV2	



tr

DEGSECT

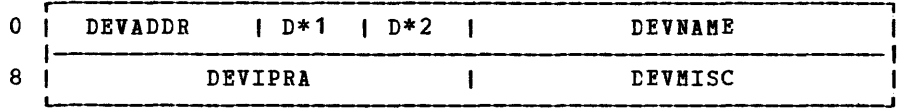
Hexadecimal Displacement	Field Name	Field Description, Contents, Meaning		
0	CCWPRINT	CCW	X'09',0,SILI,133	Printer CCW for DEBDUMP
8	SYMTABLE	DS	32D	User-defined symbol table
108	SYMTBG	DC	F'0'	Symbcl table entries
10C	BRKPNTBL	DC	16F'0,-1'	Breakpoint table
18C	TBLEND	DC	A(TBLEND)	End address of breakpoint table
<u>Storage and Constants for NEWLIN and Control</u>				
190		DS	0D	
190	ARGS	DS	5D	Arguments stored here
	MVCNT1	EQU	*-ARGS	Number of bytes in arguments
	MVCNT	EQU	MVCNT1-8	Needed for SET GPR command
	ARGMAX	EQU	*	End of argument area
1B8	JFLAGS	DS	6X	Flags corresponding to arguments. There is one flag for each parameter, as follows: 00 = Numeric (0 - 9) F0 = Hexadecimal (A - F, 0 - 9) FF = Alphabetic (A - F)
1BE	ARGSCT	DS	1X	A*1 Number of arguments in command line
	MVCNT2	EQU	*-ARGS	For initializing to zero
1C0		DS	0D	
1C0	WAITRD	DC	CL8'WAITRD'	Parameter list to get input line
1C8		DC	A(INPUT)	A(input buffer)
1CC		DC	C'U'	Clean up and logical carriage return
1CD		DC	X'00'	
1CE	WTRDCNT	DC	H'0'	Byte count filled in here
1D0	ORG	DC	F'0'	Origin of routine being examined
<u>The Following Variables Are Used by DEBUG and DEBDUMP</u>				
1D8	DEC	DS	1D	Binary word
1E0	DECDEC	DS	1D	Decimal word
1E8	ARGSAV	DS	1F	Storage for argument location
1EC	RETSAV	DS	1F	Storage for return address
1F0	EXAMLC	DS	1F	First location to be examined
1F4	EXAMLG	DS	1F	Length of field to be examined
1F8	BEGAT	DS	1F	Beginning parameter being processed
1FC	STOPAT	DS	1F	Last parameter location
	LASTLINE	EQU	DECDEC	32 bytes for last line dumped
200	SAVE1	DS	1F	DEBDUMP uses this area for line count
204	SAVE2	DS	1F	
208	DBGSAV1	DS	16F	DEBUG BALR-call save
248	DBGSAV2	DS	16F	Save area for CCNWAIT/CONREAD

Hexadecimal Displacement	Field Name				Field Description, Contents, Meaning
288	TSYM	DS	4F		Symbcl entry
298	XPSW	DS	1D		Execution PSW
2A0	OUTPT1	DC	1C' '	A*2	Byte count
2A1	LINE	DC	CL11' '		I/O buffer
2AC	LINE1	DC	87X'40'		Filler bytes
303	LINE1B	DC	C*' '	A*3	Filler bytes
304	LINE1A	DC	32X'40'		Filler bytes
324	LINE1C	DC	C*' '	A*4	Filler bytes
325	LINE1D	DC	X'40'	A*5	Filler bytes
	DBGOUT	EQU	LINE		Output buffer
	INPUT	EQU	LINE		Input buffer
	INPUT1	EQU	LINE+60		Hexadecimal pack area
328	CONWR	DS	0D		PLIST for DMSCWR to type output line
328		DC	CL8'TYPLIN'		PLIST continued
330		DC	AL1(1),AL3(DBGOUT)		PLIST continued
334		DC	CL1'B',AL2(0)		PLIST continued
337	CONWRL	DC	AL1(0)	A*6	PLIST continued
338	INPUTSIZ	DS	1H		Size of typed-in input line
33A		DS	1H		Reserved for IBM use
33C		DC	F'0'		
340	CONHCT	DC	X'FAFBFCFDFF0000'		Translate table
348	HEX	DS	1F		Binary word
34C		DC	X'FFFFFFFF'		Fence
350	HEXHEX	DS	2F		Printer graphic word
358		DC	X'FF'		Extra translate byte
359	BITS	DC	X'C0C0C0C0C0C0C0'		Scratch word
361		ORG	**14		Translate table
36F		DC	C'0123456789'		Translate table
379		DC	C'ABCDEF'		Translate table
	CONHXT	EQU	CONHCT-C'A'		
37F	DBGSWTCH	DC	X'00'	A*7	Internal DEBUG status flags
	<u>Bits defined in DBGSWTCH</u>				
			X'80'		Reserved for IBM use
			X'40'		Reserved for IBM use
	DBDEXIT	EQU	X'20'		Exit from DEBDUMP
	DBDDMSG	EQU	X'10'		Duplicate message in DEBDUMP
	DBGSET	EQU	X'08'		SET command
	DBGPERM	EQU	X'04'		Reserved for IBM use
	DBGCOND	EQU	X'02'		Reserved for IBM use
	<u>The Following are Reserved for IBM Use</u>				
380	YPSW	DS	D		PSW containing NSI
388	TBLINDX	DS	F		Current BRKPT table index
38C	BCR	NOPR	0		NOPR to pad DBGXWK when needed
38E		NOPR	0		Additional NOPR (if needed)
390	ILC	DS	1C	A*8	ILC of instruction in DBGXWK
391	ILC11	DC	X'06'	A*9	3 halfword instructions (6 bytes)
392	ILC0110	DC	X'04'	A*10	2 halfword instructions (4 bytes)
393	ILC00	DC	X'02'	A*11	1 halfword instructions (2 bytes)
	BAL	EQU	X'45'		BAL operation code
	BALR	EQU	X'05'		BALR operation code
394	DBGXWK	DS	3H		Re-create instruction at BRKPT address
39A		LPSW	48		Give control to NSI

DEVSECT

DEVSECT: DEVICE TABLE DSECT

DEVSECT describes the device information required for input/output routines. DEVSECT is a DSECT corresponding to the data in a DEVTAB entry.



Hexadecimal Displacement	Field Name	Field Description, Contents, Meaning			
0	DEVADDR	DS	1H		Virtual device address
2	DEVFLAG	DS	1X	D*1	Device flags
3	DEVTYPE	DS	1X	D*2	Device type
4	DEVNAME	DS	1F		Synbolic device name
8	DEVIPRA	DS	1F		Interrupt processing routine address
C	DEVMISC	DS	1F		Miscellaneous -- device dependent
	DEVSIZE	EQU	*-DEVSECT		Device table size (in bytes)

DEVTAB: DEVICE TABLE

DEVTAB contains the entries for the various devices handled by CMS (disks, tapes, reader, punch, printer, and console). DEVTAB is pointed to by V-constants in EMSICW and DMSITI, and is also referenced indirectly by the ADEVTAB field in NUCON.

0	CONSOLE	E0	READER1
10	ADISK	C0	PUNCH1
20	BDISK	D0	PRINTER1
30	CDISK	E0	READER2
40	DDISK	F0	PUNCH2
50	EDISK	100	PRINTER2
60	FDISK	110	TAPE1
70	GDISK	120	TAPE2
80	SDISK	130	TAPE3
90	YDISK	140	TAPE4
A0	ZDISK	150	DUMMY

Hexadecimal Displacement	Field Name			Field Description, Contents, Meaning
0	CONSOLE	DS	0D	Device table entry for console
0		DC	XL2'009'	
2		DC	XL2'0'	
4		DC	CL4'CON1'	
8		EC	VL4 (CONSI)	
C		DC	XL4'0'	Device table entry for A-disk
10	ADISK	DS	0D	
10		DC	XL2'191'	
12		DC	XL2'0'	
14		DC	CL4'DSK1'	
18		DC	AL4 (0)	Device table entry for B-disk
1C		DC	XL4'0'	
20	BDISK	DS	0D	
20		DC	XL2'000'	
22		DC	XL2'0'	
24		DC	CL4'DSK2'	Device table entry for C-disk
28		DC	AL4 (0)	
2C		DC	XL4'0'	
30	CDISK	DS	0D	
30		DC	XL2'000'	
32		DC	XL2'0'	

DEVTAB

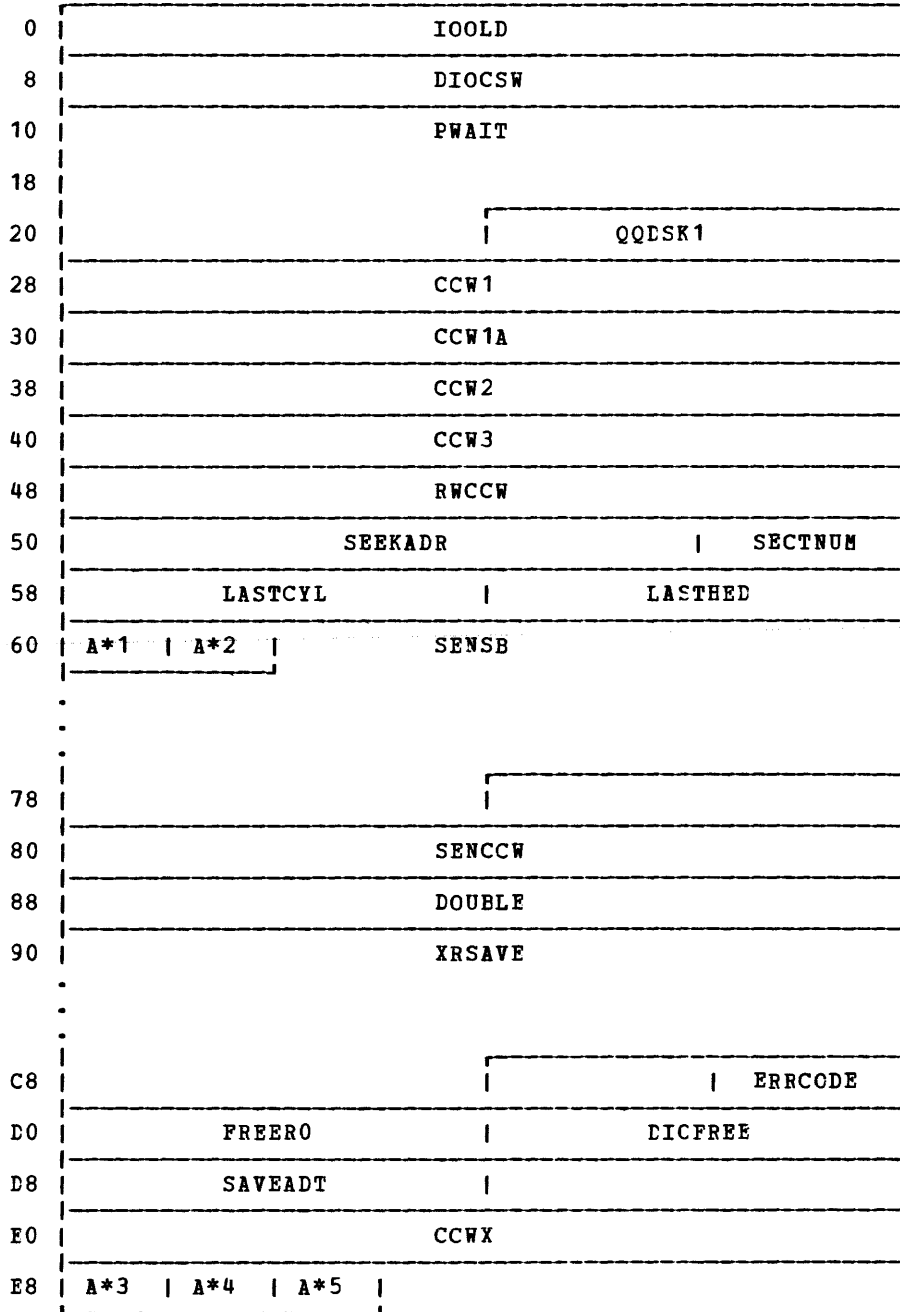
Hexadecimal Displacement	Field Name		Field Description, Contents, Meaning	
34		DC	CL4'DSK3'	
38		DC	AL4(0)	
3C		DC	XL4'0'	
40	DDISK	DS	0D	Device table entry for D-disk
40		DC	XL2'192'	
42		DC	XL2'0'	
44		DC	CL4'DSK4'	
48		DC	AL4(0)	
4C		DC	XL4'0'	
50	EDISK	DS	0D	Device table entry for E-disk
50		DC	XL2'000'	
52		DC	XL2'0'	
54		DC	CL4'DSK5'	
58		DC	AL4(0)	
5C		DC	XL4'0'	
60	FDISK	DS	0D	Device table entry for F-disk
60		DC	XL2'000'	
62		DC	XL2'0'	
64		DC	CL4'DSK6'	
68		DC	AL4(0)	
6C		DC	XL4'0'	
70	GDISK	DS	0D	Device table entry for G-disk
70		DC	XL2'000'	
72		DC	XL2'0'	
74		DC	CL4'DSK7'	
78		DC	AL4(0)	
7C		DC	XL4'0'	
80	SDISK	DS	0D	Device table entry for S-disk
80		DC	XL2'190'	
82		DC	XL2'0'	
84		DC	CL4'DSK8'	
88		DC	AL4(0)	
8C		DC	XL4'0'	
90	YDISK	DS	0D	Device table entry for Y-disk
90		DC	XL2'19E'	
92		DC	XL2'0'	
94		DC	CL4'DSK9'	
98		DC	AL4(0)	
9C		DC	XL4'0'	
A0	ZDISK	DS	0D	Device table entry for Z-disk
A0		DC	XL2'000'	
A2		DC	XL2'0'	
A4		DC	CL4'DSK0'	
A8		DC	AL4(0)	
AC		DC	XL4'0'	
B0	READER1	DS	0D	Device table entry for READER1
B0		DC	XL2'00C'	
B2		DC	XL2'0'	
B4		DC	CL4'RDR1'	
B8		DC	AL4(0)	
BC		DC	XL4'0'	
C0	PUNCH1	DS	0D	Device table entry for PUNCH1
C0		DC	XL2'00D'	
C2		DC	XL2'0'	
C4		DC	CL4'PCH1'	
C8		DC	AL4(0)	
CC		DC	XL4'0'	
D0	PRINTER1	DS	0D	Device table entry for PRINTER1
D0		DC	XL2'00E'	
D2		DC	XL2'0'	
D4		DC	CL4'PRN1'	
D8		DC	AL4(0)	
DC		DC	XL4'0'	

Hexadecimal Displacement	Field Name			Field Description, Contents, Meaning
E0	READER2	DS	0D	Device table entry for READER2
E0		DC	XL2'012'	
E2		DC	XL2'0'	
E4		DC	CL4'RDR2'	
E8		DC	AL4(0)	
EC		DC	XL4'0'	Device table entry for PUNCH2
F0	PUNCH2	DS	0D	
F0		DC	XL2'013'	
F2		DC	XL2'0'	
F4		DC	CL4'PCH2'	
F8		DC	AL4(0)	Device table entry for PRINTER2
FC		DC	XL4'0'	
100	PRINTER2	DS	0D	
100		DC	XL2'010'	
102		DC	XL2'0'	
104		DC	CL4'PRN2'	Device table entry for TAPE1
108		DC	AL4(0)	
10C		DC	XL4'0'	
110	TAPE1	DS	0D	
110		DC	XL2'181'	
112		DC	XL2'0'	Device table entry for TAPE2
114		DC	CL4'TAP1'	
118		DC	AL4(0)	
11C		DC	XL4'0'	
120	TAPE2	DS	0D	
120		DC	XL2'182'	Device table entry for TAPE3
122		DC	XL2'0'	
124		DC	CL4'TAP2'	
128		DC	AL4(0)	
12C		DC	XL4'0'	
130	TAPE3	DS	0D	Device table entry for TAPE4
130		DC	XL2'183'	
132		DC	XL2'0'	
134		DC	CL4'TAP3'	
138		DC	AL4(0)	
13C		DC	XL4'0'	Device table entry for DUMMY
140	TAPE4	DS	0D	
140		DC	XL2'184'	
142		DC	XL2'0'	
144		DC	CL4'TAP4'	
148		DC	AL4(0)	Device table entry for TABEND
14C		DC	XL4'0'	
150	DUMMY	DS	0D	
150		DC	XL2'000'	
152		DC	XL2'0'	
154		DC	CL4'XXXX'	Device table entry for TABEND
158		DC	AL4(0)	
15C		DC	XL4'0'	
160	TABEND	DS	0D	

DIOSECT

DIOSECT: DISK I/O WORK AREA

DIOSECT describes the fields used by DMSDIO as a work area when reading and writing actual blocks of data on CMS disks. DIOSECT is pointed to by a V-constant in DMSNUC, and referenced indirectly by ADIOSECT in NUCON.

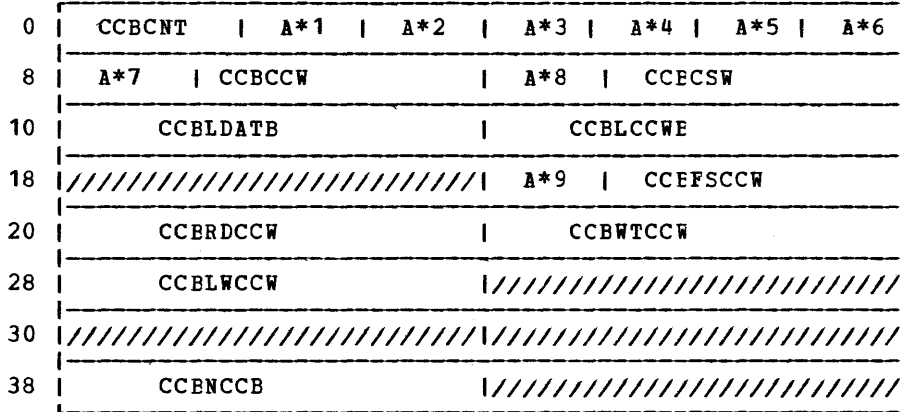


Hexadecimal Displacement	Field Name	Field Description, Contents, Meaning		
0	DIOSECT	DSECT		
0	IOOLD	DC	1D'0'	I/O cld PSW (from interrupt routine)
8	DIOCSW	DC	1D'0'	CSW (from interrupt routine)
<u>PLIST To Call DMSIOW</u>				
10		DS	0F	
10	PWAIT	DC	CL8'WAIT'	
18		DC	C'DSK-'	Filled in to correct symbolic disk number
1C		DC	F'0'	
20		DC	F'0'	
24	QQDSK1	DC	F'0'	First two bytes are always 0
	QQDSK2	EQU	QQDSK1+2	Halfword copy of 16th track disk address
<u>CCW Chain</u>				
28	CCW1	CCW	X'07',SEEKADR,X'40',6	Seek
30	CCW1A	CCW	X'03',0,X'40',1	Seek or set sector
38	CCW2	CCW	X'31',SEEKADR+2,X'40',5	Search
40	CCW3	CCW	X'08',*-8,0,1	TIC back to search
48	RWCCW	CCW	X'00',*-,X'20',*-*	Read or write data
50	SEEKADR	DC	XL7'00'	Seek/search information (first 3 bytes are 0)
57	SECTNUM	DC	X'00'	Sector number
<u>I/O Information</u>				
58	LASTCYL	DC	F'0'	Becomes last cylinder number used
5C	LASTHED	DC	F'0'	Becomes last head number used
60	DEVTYP	DC	X'00'	A*1 01=2311, 08=2314, 09=3330
61	DIOFLAG	DC	X'00'	A*2 RDTK/WRTK flag:
<u>Bits defined in DIOFLAG</u>				
	TOOBIG	EQU	X'04'	Byte count is greater than 800
	WRTKF	EQU	X'02'	Writing first chain link
	QQTRK	EQU	X'01'	Handling first chain link
	DIAGNUM	EQU	24	Number assigned by CP for DIAGNOSE I/O
62	SENSB	DC	XL24'00'	Sense information
7C		DS	0F	
80	SENCCW	CCW	X'04',SENSB,X'20',24	READ 24 BYTES SILI
<u>Miscellaneous Storage</u>				
88	DOUBLE	DC	1D'0'	(Scratch area, for CVD use, etc.)
<u>Keep the Following Three in Order</u>				
90	XRSAVE	DS	15F	Registers 0-14 saved here for RDTK/WRTK
CC		DC	AL3(0)	First 3 bytes of R15 error code
CF	ERRCODE	DC	AL1(*-*)	Error code (in R15 at exit)
<u>Keep the Following Two in Order</u>				
D0	FREER0	DC	F'0'	No. of doublewords of free storage (if any)
D4	DIOFREE	DC	F'0'	Address of free storage for buffer or CCWs
D8	SAVEADT	DC	F'0'	Handy place for an ADT address
E0	CCWX	CCW	X'23',SECTNUM,X'40',1	Set sector
E8	DIAGRET	DC	X'00'	A*3 CP's DIAGNOSE return code if nonzero
E9	IOCOMM	DC	X'00'	A*4 Set to read (06) or write (05)
EA	LASTREC	DC	X'00'	A*5 Number (1-14) of the last recrd processed

DMSCCB

DMSCCB: COMMAND CONTROL BLOCK

DMSCCB describes all fields of a DOS Command Control Block (CCB). This DSECT is used by DMSXCP to map the CCB specified by a user for an SVC 0 (EXCP) and passes the address of CCB to DMSXCP.



Hexadecimal Displacement	Field Name			Field Description, Contents, Meaning
	CCBST	EQU	*	Start CCB
	CCBD	EQU	*	Command control block
0	CCBLEN	DS	0XL16	Map of the DOS CCB
0	CCBCNT	DS	XL2	Residual count
2	CCBERMAP	DS	0XL4	Four bytes used to check errors
2	CCBCOM1	DS	XL1	A*1 Communications byte 1
	<u>Bits defined in CCBCOM1</u>			
	CCBWAIT	EQU	X'80'	Traffic bit (set at CE)
	CCBEOF	EQU	X'40'	End of file
	CCBIOERR	EQU	X'20'	Unrecoverable I/O error
	CCBERROK	EQU	X'10'	Accept unrecoverable error
	CCBRDC	EQU	X'08'	Return data checks
	CCBPDE	EQU	X'04'	Post at device end
	CCBDCV	EQU	X'02'	Return data check RD/CHK
	CCBUERR	EQU	X'01'	User error routine
3	CCBCOM2	DS	XL1	A*2 Communications byte 2
	<u>Bits defined in CCBCOM2</u>			
	CCBDCCNT	EQU	X'80'	Data check in count area
	CCBTRKOV	EQU	X'40'	Track overrun
	CCBEOC	EQU	X'20'	End of cylinder
	CCBDC	EQU	X'10'	Data check
	CCBNOREC	EQU	X'08'	No record found
	CCBRETRY	EQU	X'04'	Retry no record found
	CCBVER	EQU	X'02'	Verify error
	CCBCC	EQU	X'01'	Command chain (retry)

Hexadecimal Displacement	Field Name				Field Description, Contents, Meaning
4	CCBCSW1	DS	XL1	A*3	CSW status bit 1
	<u>Bits defined in CCECSW1</u>				
	CCBATTN	EQU	X'80'		Attention
	CCBSTMOD	EQU	X'40'		Status modifier
	CCBCUE	EQU	X'20'		Control unit end
	CCBBUSY	EQU	X'10'		Busy
	CCBCE	EQU	X'08'		Channel end
	CCBDE	EQU	X'04'		Device end
	CCBUC	EQU	X'02'		Unit check
	CCBUE	EQU	X'01'		Unit exception
5	CCBCSW2	DS	XL1	A*4	CSW status bit 2
	<u>Bits defined in CCBCSW2</u>				
	CCBPCI	EQU	X'80'		Program-controlled interrupt
	CCBILEN	EQU	X'40'		Incorrect length
	CCBPROGM	EQU	X'20'		Program check
	CCBPROT	EQU	X'10'		Protection check
	CCBCHAND	EQU	X'08'		Channel data check
	CCBCHANC	EQU	X'04'		Channel control check
	CCBICTRL	EQU	X'02'		Interface control check
	CCBCHAIN	EQU	X'01'		Chaining check
6	CCBSYMU	DS	0XL2		Symblic unit (SYSUN)
6	CCBSUCLS	DS	XL1	A*5	U - IUB class
7	CCBSUNUM	DS	XL1	A*6	N - IUB number within class
8	CCBLIOBS	DS	XL1	A*7	Reserved for LICBS
9	CCBCCW	DS	XL3		Pointer to start of channel program
C	CCBCOM3	DS	XL1	A*8	Communication byte 3
	CCBAPEND	EQU	X'40'		Appendage exit at interrupt
D	CCBCSW	DS	XL3		Pointer to CSW or to appendage routine
10	CCBLDATB	DS	A		Address of last data block
14	CCBLCCWB	DS	A		Address of last CCW block
18		DS	F		Reserved for IBM use
1C	CCBUFLGS	DS	X	A*9	I/O manager CCB flags
	<u>Bits defined in CCBUFLGS</u>				
	CCBUEAIC	EQU	X'80'		Error analysis in control
	CCBUEAC	EQU	X'40'		Error analysis complete
	CCBURDCW	EQU	X'20'		Read CCW active
	CCBRPS	EQU	X'10'		RPS channel program candidate
1D	CCBFSCCW	DS	XL3		Save area for first CCW address
20	CCBRDCCW	DS	F		Address of first read CCW
24	CCBWTCCW	DS	F		Address of first write CCW
28	CCBLWCCW	DS	F		Address of the last write CCW
2C		DS	3F		Reserved for IBM use
 <u>Note:</u> CCBLWCCW chain field must have the same displacement as does FCCHAIN in FCDB and also BKPFSTBK in BKPRD					
38	CCBNCCB	DS	A		Address of next CCB block
3C		DS	F		Reserved for IBM use

DOSSECT

DOSSECT: DOS SIMULATION CONTROL BLOCK

DOSSECT simulates the CMS File Control Block (FCB) in the CMS/DOS environment. DOSSECT is invoked by the DOSCB macro.

The DOS Simulation Control Blocks are chained together. The DOSFIRST field in NUCCN points to the first DOSCB in the chain, or if no chain exists, contains zero.

0	DOSNEXT			DOSCBID	
8	DOSDD				
10	DOSOP				
18	DOSDSNAM				
20	DOSDSTYP				
28	DOSDSMD		DOSITEM		DOSBUFF
30	DOSBYTE			DOSFORM	DOSCOUT
38	DOSREAD		A*1	A*2	DOSBLKSZ
40	DOSWORK				
48	A*3	A*4	A*5	A*6	DOSCSFST
50	DOSOSDSN			DOSVOLTE	
58	DOSEXTEB			DOSSENSE	A*7 A*8
60	DOSBUFSP			DOSUCNAM	
68	DOSUCNAM (cont.)				
	DOSSAVE				
80				A*9	A*10 ///////////////

Hexadecimal Displacement	Field Name			Field Description, Contents, Meaning
0	DOSINIT	DS	0X	DOSCE flag byte
	<u>Bits defined in DOSINIT</u>			
	DOSOS	EQU	X'40'	OS user-created bit
	DOSDOS	EQU	X'20'	Defined for non-CMS disk
	DOSCMS	EQU	X'10'	Defined for CMS disk
	DOSDDCAT	EQU	X'08'	User catalog data set
	DOSPERM	EQU	X'04'	Permanent control block
	DOSJCAT	EQU	X'02'	Search VSAM job catalog
	DOSUCAT	EQU	X'01'	Search VSAM user catalog
0	DOSNEXT	DS	A	AL3(next DOSCB)
4	DOSCBID	DS	CL4	DLBL to distinguish from CMSCE
8	DOSDD	DS	CL8	Data definition name
10	DOSOP	DS	CL8	CMS operation
18	DOSTAPID	DS	0X	Tape identification
18	DOSDSNAM	DS	CL8	Data set name
20	DOSDSTYP	DS	CL8	Data set type
28	DOSDSMD	DS	CL2	Data set mode
2A	DOSITEM	DS	H	Item (record) number
2C	DOSBUFF	DS	F	A(input/output buffer)
30	DOSBYTE	DS	F	Size of buffer (data count)
34	DOSFORM	DS	CL2	File format: fixed/variable
36	DOSCOUT	DS	H	Recrdrds per CMS physical block
38	DOSREAD	DS	F	Number of bytes actually read
3C	DOSDEV	DS	X	A*1 Device type code
	<u>Bits defined in DOSDEV</u>			
	DOSDUM	EQU	0	Dummy device
	DOSDSK	EQU	20	Disk
3D	DOSTAPMD	DS	X	A*2 Tape mode set to save
3E	DOSBLKSZ	DS	H	Block size
40	DOSWORK	DS	D	Work area
48	DOSYSXXX	DS	0H	Logical unit for CMS/DOS
48	DOSSYS	DS	1X	A*3 SYS/PROG unit: X'00'=SYS, X'01'=PROG
49	DOSXXX	DS	1X	A*4 Number from 000-255 associated with the unit
4A	DOSEXT	DS	1X	A*5 Number of DCS extents left to process
4B	DOSEXTCT	DS	1X	A*6 Current DCS extent
4C	DOSOSPST	DS	F	Pointer to CS FST
50	DOSOSDSN	DS	F	Pointer to CS dsname block
54	DOSVOLTB	DS	F	A(volume ID table)--VSAM multivolume data set
58	DOSEXTTB	DS	F	A(extent table) for VSAM data space
5C	DOSSENSE	DS	H	I/O sense data
5E	DOSVOLNO	DS	X	A*7 No. of volumes (entries in DOSVOLTB)
5F	DOSEXTNO	DS	X	A*8 No. of extents (entries in DOSEXTTB)
60	DOSBUFSP	DS	F	Size of VSAM I/C buffer(s)
64	DOSUCNAM	DS	CL8	VSAM user catalog ddname
6C	DOSSAVE	DS	6F	Temporary save area for re-entrant code
84	DOSEXTCX	DS	1X	A*9 Current extent (used by DMSXCF)
85	DOSTYPE	DS	1C	A*10 Data set type (SAM=S, VSAM=A)
86		DS	H	Reserved for IBM use
88	DOSEND	DS	0D	End address of this block
	DOSSENSIZ	EQU	(*-DOSSECT)/8	Size of block in doublewords

EDCB

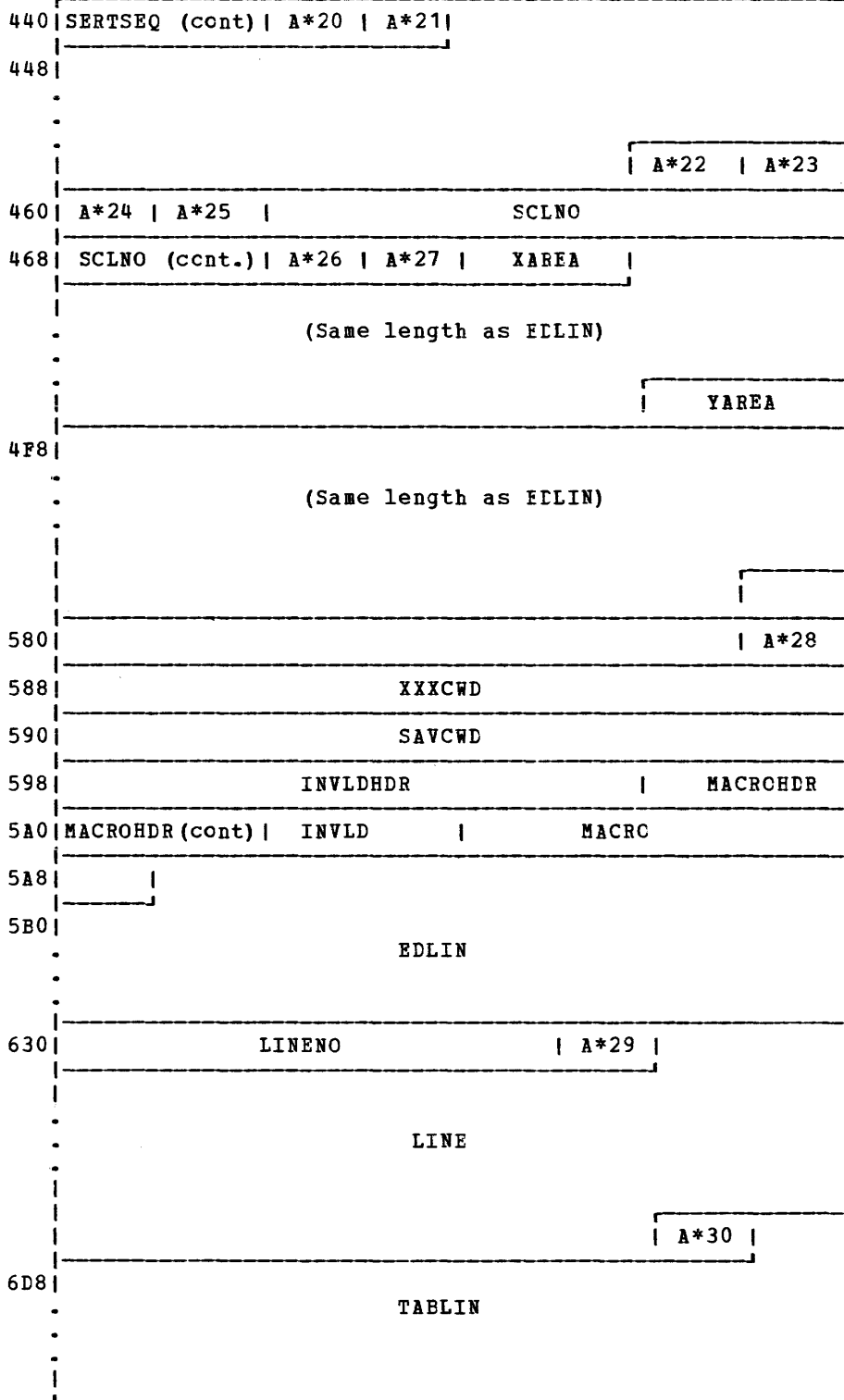
EDCB: EDIT CONTROL BLOCK

EDCB is used by all CMS EDIT modules to define common free storage control blocks. It is initialized by DMSEDX, the EDIT bootstrap routine, and built dynamically from user free storage each time a user issues the EDIT command.

0	FNAME				
8	FTYPE				
10	FMODE	A*1	A*2	TRUNCOL	ZONE1
18	ZONE2	VERCOL1	VERCOL2	VERLEN	
20	SCRBUFAD		CARDINCR		
28	LMSTART	LMINCR	A*3	A*4	
.	TABS				.
48	SEQNAME	A*5	PADBUF		
50	PADBUF (cont.)				
58	PTR1				
60	PTR2			PTR3	
68	AEXTEND			CORITEM	
70	SPARES			FPTR	
78	ITEM			AFSTFNRD	
80	FREELEN			FREEAD	
88	EDRET			EDMSK	
90	MAINAD				
.	AUTOREG				.
C8	CARDNO			COUNT	
D0	LMCURR				
D8	BUFFL			BUFFA	
E0	CANSV				
.					.
100					DUALNOS

DUALNOS (cont.)	
1F8	DECIMAL HALF
200	REGSAVE
210	
218	REGSAVX
220	REPCNT
	SAVEAR
260	XYCNT CHNGNUM
268	TIN
270	AEDLIN A*6
278	TOUT
280	A*7
288	IOLIST
290	IOID
298	
2A0	IOMODE IOAD
2A8	RECS
2B0	ALTLIST
2B8	ALTLIST (cont.) EDWORK
2C0	EDWORK (cont.)
2C8	ALTMODE
2D0	ALTMODE (cont.)
2F0	STACKAT
2F8	STACKAT (cont.)
300	STACKATL ATIN

308	ATTN (cont.)				
310	ATTNLEN		RENLIST		
318	RENLIST (cont.)		RPLIST		
320	RPLIST				
328	STRTNO		INCRNO		
330	AINCORE		FSIZE		
338	DECLTH				
340	RANGE				
348	////////////////////////////////////RESVD1////////////////////////////////////				
350	A*8	BUFAD		A*9	A*10 WRCOUNT
358	BUFFLOC		ALINELOC		
360	ANUMLOC		AFLAGLOC		
368	TRNCNUM		AUTOCNT	AUTOCURR	
370	CHNGCNT	DITCNT	ELCT		LINELOC
378	NUMLOC	SAVCNT	TVERCOL1	TVERCOL2	
380	A*11	A*12	AREA		
388	AREA (cont.)	A*13	A*14	CHNGMSG	
3A0			CHGTRUNC		
3A8					
3B0			(unidentified)		
3B8	CMODE				
3C0			FILEMS		
					A*15 A*16
3D8	A*17		JAR		
					NEWNAME
428	NEWNAME (cont.)		NEWTTYPE		
430	NEWTTYPE (cont.)		NEWMODE	A*18	
438	SERSAV (cont.)				A*19



EDCB

Hexadecimal Displacement	Field Name			Field Description, Contents, Meaning
		DS	0F	
	BLOC	EQU	*	
0	FNAME	DS	CL8	Filename
8	FTYPE	DS	CL8	Filetype
10	FMODE	DS	CL2	Filemode
12	FV	DS	CL1	A*1 Recrd format
13	CASESW	DS	CL1	A*2 Case setting
14	TRUNCOL	DS	H	Truncation column
16	ZONE1	DS	H	Beginning zone initialized to first column
18	ZONE2	DS	H	End zone
1A	VERCOL1	DS	H	Verify column 1
1C	VERCOL2	DS	H	Verify column 2
1E	VERLEN	DS	H	Verify length
20	SCRBUFAD	DS	F	Address of GETMAIN buffer
24	CARDINCR	DS	F	Increment for serialization
28	LMSTART	DS	H	Where line numbers start
2A	LMINCR	DS	H	Automatic line numbers using default increment
2C	FLAG	DS	CL1	A*3 Flags for line monitoring
2D	FLAG2	DS	CL1	A*4 Miscellaneous flags
2E	TABS	DS	26AL1	Maximum of 25 tabs is allowed
	ENDTABS	EQU	*	End of tabs
48	SEQNAME	DS	CL3	Name, if any, for serialization
4B	PADCHAR	DS	CL1	A*5 '0' on right, ' ' on left
	ENDBLOC	EQU	*	End of blocks
<u>Note:</u> PADBUF must remain directly behind PADCHAR				
4C	PADBUF	DS	9C	Pad characters
	PTRCONS	EQU	*	DMSEDI line pointers
58	PTR1	DS	2F	Pointer to top of file (for dummy top line)
60	PTR2	DS	F	Current line pointer
64	PTR3	DS	F	Pointer to bottom line
68	AEXTEND	DS	F	Pointer to end of used area of storage
6C	CORTEM	DS	F	Number of bytes for one line in storage
70	SPARES	DS	F	Number of spare lines
74	FPTR	DS	F	Free list pointer
78	ITEM	DS	F	Item length
7C	AFSTFNRD	DS	F	Anchor for stacked lines upon entry
80	FREELN	DS	F	Length of free storage
84	FREED	DS	F	Address of free storage
88	EDRET	DS	F	CMS return address
8C	EDMSK	DS	F	DMSSCR edit mask
90	MAINAD	DS	F	LOADSYS address; 0 if LOADMCD
	EPTRCONS	EQU	*	DMSEDI save areas and buffer pointers
94	AUTOREG	DS	13F	Autocheck save area
C8	CARDNO	DS	F	Save area for sequence number
CC	COUNT	DS	F	Number of characters in EDLIN
D0	LMCURR	DS	2F	Prompter current line number
D8	BUFL	DS	F	Length of string (EDC)
DC	BUFFA	DS	F	Address of string (EDC)
E0	CANSAV	DS	9F	Register save (EDC)
104	DUALNOS	DS	CL240	Temporary string buffer (EDC)
1F8		DS	0D	
1F8	DECIMAL	DS	F	Used by DECBIN and BINDEC

Hexadecimal Displacement	Field Name			Field Description, Contents, Meaning
1FC	HALF	DS	F	BINDEC only edits four characters
200	REGSAV	DS	5F	Register save area
214	REGSAVX	DS	3F	Register save area
220	REPCNT	DS	F	FOR count
224	SAVEAR	DS	15F	DMSSCR save area
260	XYCNT	DS	F	X or Y execution count
264	CHNGNUM	DS	F	Number of lines to change
268	TIN	DS	0F	WAITRE PLIST
268		DS	CL8	WAITRE
270		DS	X	'1' for console No. 1
271	AEDLIN	DS	3X	A(EDLIN)
274	CASEREA	DS	C	A*6 Default to uppercase
275		DS	3X	Length put here
278	TOUT	DS	0F	TYPLIN PLIST
278		DS	CL8	TYPLIN
280		DS	X	'1' for console No. 1
281		DS	3X	Address goes here
284		DS	C	'B' for black ribbon
285	TYPFLG	DS	X	A*7 X'20' maximum length override
286		DS	H	Length goes here
	CRBIT	EQU	X'80'	Suppress carriage return
288	IOLIST	DS	0F	Initialization for state of source
288		DS	CL8	STATE
290	IOID	DS	CL8	EDIT
298		DS	CL8	CMSUT1
2A0	IOMODE	DS	CL2	A1
2A2		DS	H	Item number for RDBUF
2A4	IOAD	DS	CL4	Do not allow asterisks
2A8		DS	F	133 (XINSCRIPT use LINE)
2AC		DS	CL2	F/V and null record indicator
2AE	RECS	DS	H	1 (XINSCRIPT one line at a time)
2B0		DS	F	Number of bytes read from RDBUF
2B4	ALLLIST	DS	0F	
2B4		DS	CL8	RENAME
2BC	EDWORK	DS	CL8	EDIT
2C4		DS	CL8	CMSUT1
2CC	ALTMODE	DS	CL8	A1
2D4		DS	CL8	New filename
2DC		DS	CL8	New filetype
2E4		DS	CL2	'*'
2E6		DS	CL6	New filemode
2EC		DS	8X	FF fence
2F4	STACKAT	DS	0F	PLIST to stack first in first out
2F4		DS	CL8	ATTN
2FC		DS	CL4	First in first out
300	STACKATL	DS	F	Length and address of line to stack
304	ATTN	DS	0F	
304		DS	CL8	ATTN
30C		DS	CL4	Last in first out
310	ATTNLEN	DS	F	Length and address of line to stack
314	RENLIST	DS	0F	RENUM PLIST
314		DS	CL8	RENUM
31C	RPLIST	DS	CL12	Fileid
328	STRTNO	DS	F	Starting number
32C	INCRNO	DS	F	Increment number
330	AINCORE	DS	F	In-storage copy address
334	FSIZE	DS	F	Record length
338	DECLTH	DS	D	DMSSCR work area
340	RANGE	DS	D	Message data areas
348	RESVD1	DS	D	Reserved for IBM use
350	CMBLOK	DS	X	A*8 X'19'
351	BUFAD	DS	3X	Buffer address
354	FLG	DS	X	A*9 CCW flag

EDCB

Hexadecimal Displacement	Field Name			Field Description, Contents, Meaning
355	CTL	DS	X	A*10 Control byte
356	WRCOUNT	DS	H	Write count
358	GIOPLIST	DS	OF	DMSSCR PLIST for DMSGIO
358	BUFFLOC	DS	F	Buffer location
35C	ALINELOC	DS	F	Address of line location
360	ANUMLOC	DS	F	Address of number location
364	AFLAGLOC	DS	F	Address of flag location
368	TRNCNUM	DS	F	Number of lines truncated
36C	AUTOCNT	DS	H	Autosave parameter
36E	AUTOCURR	DS	H	Current modification count
370	CHNGCNT	DS	H	Temporary area for change
372	DITCNT	DS	H	Count of lines stacked by the REUSE (=) subroutine
374	EDCT	DS	H	Next char in EDLIN
376	LINELOC	DS	H	Display line number
378	NUMLOC	DS	H	Display count
37A	SAVCNT	DS	H	Length of last saved request not beginning with either a ? or =
37C	TVERCOL1	DS	H	Temporary area for verifying column 1
37E	TVERCOL2	DS	H	Temporary area for verifying column 2
380	ALCHAR1	DS	C	A*11 Temporary byte used by ALTER
381	ALCHAR2	DS	C	A*12 Temporary byte used by ALTER
382	AREA	DS	CL8	EDIT instruction work
38A	BYTE	DS	X	A*13 Temporary byte (used by GET)
38B	CHNGFLAG	DS	X	A*14 Flag for change
38C	CHNGMSG	DS	CL20	Lines changed message
3A0		DS	2X	C','
3A2	CHGTRUNC	DS	13X	C'...LINE(S)'
3AF		DS	9X	C'TRUNCATED'
3B8	CMODE	DS	CL4	Filemode for MODECHK routine
3BC	FILEMS	DS	CL26	Retry message
3D6	FLAGLOC	DS	X	A*15 Flag for DMSGIO
3D7	GETFLAG	DS	X	A*16 Flag for GETFILE
3D8	HOLDFLAG	DS	X	A*17 DMSSCR SCRFLGS
3D9	JAR	DS	(ENDBLOC-BLOC)	AL1 Save area for preserve
425	NEWNAME	DS	CL8	Name area for FILE and SAVE commands
42D	NEWTYP	DS	CL8	Type area for FILE and SAVE commands
435	NEWMODE	DS	CL2	Mode area for FILE and SAVE commands
437	SERSAV	DS	CL8	A*18 Identification number for save area
43F	SERTSEQ	DS	CL3	A*19 Temporary byte identification number area
442	SERTSW	DS	X	A*20 Temporary byte used by identification number
443	SIGNAL	DS	X	A*21 Signal between routines
444	TEMPTAB	DS	(ENDTABS-TABS)	AL1 Temporary spot for new tabs
45E	UTILFLAG	DS	X	A*22 DMSSCR utility flags
45F	XYFLAG	DS	X	A*23 X/Y active flag
460	SCRFLGS	DS	X	A*24 Screen function flags
461	SCRFLG2	DS	X	A*25 More screen function flags
462	SCLNO	DS	8C	Save LINEMODE sequence number
46A	TWITCH	DS	X	A*26 Location flags
46B	TYPSCR	DS	X	A*27 Display unit size index value
46C	XAREA	DS	H	X length and request buffer
46E		DS	CL135	(Same length as EDLIN)
4F6	YAREA	DS	H	Y length and request buffer
4F8		DS	CL135	(Same length as EDLIN)
57F				
580		CNOP	6,8	Alignment for XXXCWD
586		DS	X	Alignment for XXXCWD
587	BLANK1	DS	X	A*28 Blank for clearing EDIT XXXCWD
588	XXXCWD	DS	CL8	EDIT token buffer
590	SAVCWD	DS	CL8	Location at which contents of XXXCWD are saved
598	INVLHDR	DS	CL6	?EDIT: (for invalid request message)

Hexadecimal Displacement	Field Name			Field Description, Contents, Meaning
59E	MACROHDR	DS	CL4	EXEC for EDIT macro stacking
5A2	INVLD	DS	OCL6'?EDIT:'	Location to store invalid header
		ORG	INVLD+2	
5A4	MACRO	DS	OCL4'EXEC'	Location to hold macro header (if invalid)
		ORG	MACRO+4	
5A8		DS	C	Blank used for clearing EDLIN
5A9	EDLIN	DS	CL135	Terminal input buffer
630	LINENO	DS	CL5	Line number for typeout
635	BLANK2	DS	X	A*29 Blank for clearing line
636	LINE	DS	CL160	Current line is held here
6D6	BLANK3	DS	X	A*30 Blank for clearing TABLIN
6D7	TABLIN	DS	CL160	Output from spread
778	EDCBEND	DS	OD	
	EDCBLTH	EQU	(EDCBEND-EDCB)	Length of EDCB in doublewords

ERDSECT

ERDSECT: ERROR HANDLING ROUTINE DSECT

ERDSECT describes the fields in a work area used for giving responses and error messages via the DMSERR or LINEDIT macros. A V-constant in DMSERR points to the DMSERT CSECT in DMSNUC.

0	ERT1		
8	ERT2		
10			
18	ERSAVE		
.			
.			
58	ERPAS13		
.			
.			
A0	A*1 A*2		ERPTXA
A8	ERPNUM A*3	ERPCS	
B0	ERPBFA		ERPSEA
B8	ERSBD	A*4 A*5	
C0	ERSSZ		
C8	(Doubleword preceding text)		
D0	ERMESS	ERSECT	ERNUM
D8	A*6 A*7		
E0	ERTEXT		
.			
.			
160	ERTPL		
168	ERTPLA		ERTPLL

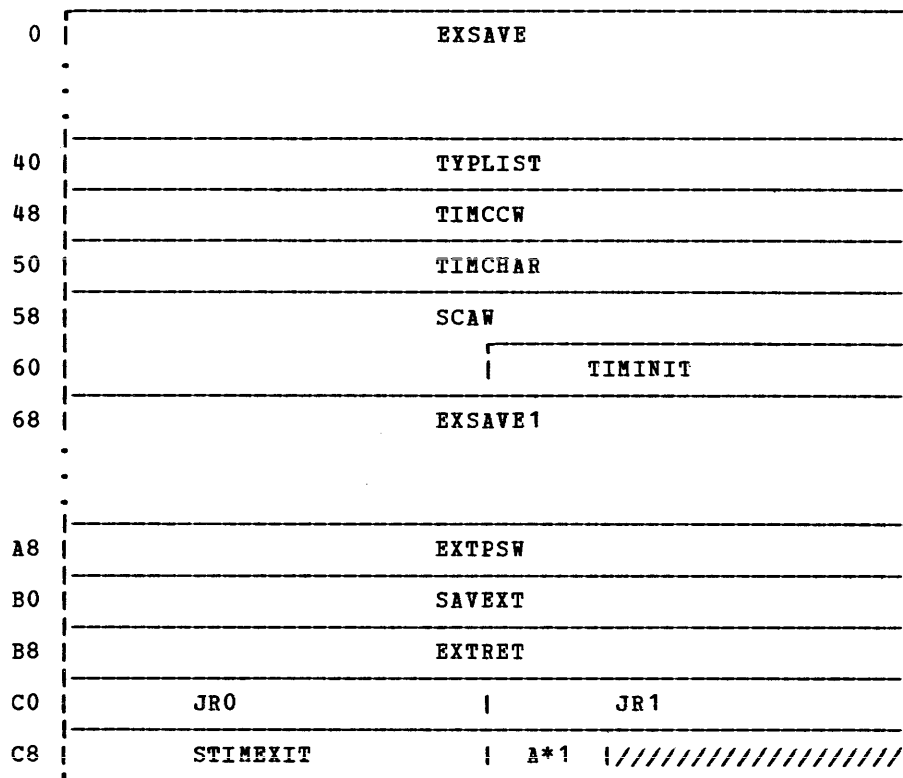
Hexadecimal Displacement	Field Name	Field Description, Contents, Meaning		
<u>Work Area for DMSERR Error Handling Routine</u>				
0	ERT1 DS D	Doubleword workspace		
8	ERT2 DS 2D	Two doublewords workspace		
<u>Save Area</u>				
18	ERSAVE DS 16F	Pass this save area in R13 to BALR-ed to-routines		
58	ERPAS13 DS 18F			
<u>Reconstructed PLIST Area</u>				
A0	ERPF1 DS B	A*1	First flag byte	
<u>Bits defined in ERPF1</u>				
	ERF1TX EQU X'80'	Text address in PLIST		
	ERF1HD EQU X'40'	Header in PLIST		
	ERF1BF EQU X'20'	Buffer address in PLIST		
	ERF1SB1 EQU X'10'	One substitution		
	ERF1SBN EQU X'08'	More than one substitution		
A1	ERPF2 DS B	A*2	Second flag byte	
<u>Bits defined in ERPF2</u>				
	ERF2CM EQU X'80'	Blank compression wanted		
	ERF2DT EQU X'40'	Dot at end of line wanted		
	ERF2DI EQU X'20'	HALT=YES wanted		
<u>Last 3 Bits Indicate DISP Field</u>				
	ERF2ER EQU 0	Error message		
	ERF2TY EQU 1	Type		
	ERF2SI EQU 2	SIO		
	ERF2NO EQU 3	None		
	ERF2PR EQU 4	Print		
	ERF2CP EQU 5	CPCOMM		
A4	ERPTXA DS A	Text address		
A8	ERPHDR DS 0CL6	Error message header		
A8	ERPNUM DS H	Message number		
AA	ERPLET DS C	A*3	Message letter	
AB	ERPCS DS CL3	CSECT name		
B0	ERPBFA DS A	Buffer address (for BUFFA)		
<u>Fields for Substitutions</u>				
B4	ERPSBA DS A	Pointer to first (next) group of substitution parameter in original PLIST		
B8	ERSBD DS A	Data address and/or value of current substitution parameter		
BC	ERSBF DS B	A*4	Flag byte for current substitution parameter	
<u>Bits defined in ERSBF</u>				
	ERSFLST EQU X'80'	The last substitution parameter		
	ERSFA EQU X'40'	A-type option		
	ERSFL EQU X'20'	Length specified		

ERDSECT

Hexadecimal Displacement	Field Name				Field Description, Contents, Meaning
	<u>Last 3 Bits Give Option Type</u>				
	ERSFH	EQU	0		HEX or HEXA
	ERSFD	EQU	1		DEC or DECA
	ERSFC	EQU	2		CHARA
	ERSFH4	EQU	3		HEX4A
	ERSFC8	EQU	4		CHAR8A
BD	ERSBL	DS	X	A*5	Byte length for the current substitution parameter
C0	ERSSZ	DS	A		Size of substitution field is number of dots minus 1
	<u>Message Construction Area</u>				
C8		DS	D		Need doubleword before text
D0	ERMESS	DC	C'DMS'		First letters of header
D3	ERSECT	DC	C'MMM'		DSECT name
D6	ERNUM	DC	C'NNN'		Message number
D9	ERLET	DC	C'L'	A*6	Message level letter
DA	ERBL	DC	C' '	A*7	Blank
	ERTSIZE	EQU	130		Maximum text size
DB	ERTEXT	DS	(ERTSIZE+1)C		Message text area
	<u>TYPLIN/PRINTER PLIST Construction Area</u>				
160		DS	0F		
160	ERTPL	DC	CL8'TYPLIN '		
168	ERTPLA	DS	AL1(1),AL3(ERMESS)		Message text address
16C	ERTPLL	DS	C'R',AL3		Message length

EXTSECT: EXTERNAL INTERRUPT WORK AREA

EXTSECT describes the fields in the External Interrupt work area referenced by DMSITE. EXTSECT is pointed to by the AEXTSECT field in NUCON.



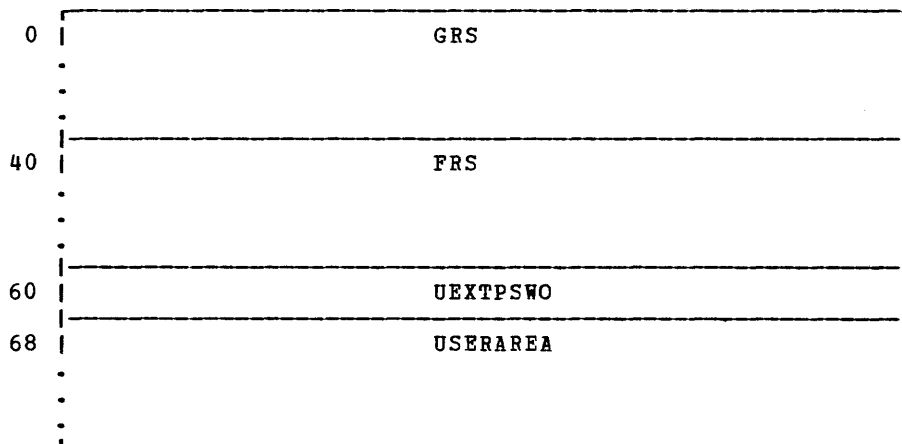
Hexadecimal Displacement	Field Name	Field Description, Contents, Meaning		
<u>Storage for Timer Interrupt</u>				
0	EXSAVE	DS	16F	Saved external old PSW and/or contents of registers
40	TYPLIST	DC	CL8'TYPLIN'	PLIST to type BLIP character
48	TIMCCW	DC	A(TIMCHAR)	
4C		DC	C'B',X'81',AL2(1)	
50	TIMCHAR	DC	X'FF',XL7'00'	BLIP character(s)
58	SCAW	DC	XL12'00'	Saved CSW/CAW
64	TIMINIT	DC	A(2000000/13)	Value that sets timer equal to 2 secnds
<u>Storage for External (Other than Timer) Interrupt</u>				
68	EXSAVE1	DS	16F	Saved registers
A8	EXTPSW	DC	X'80000000'	Filled-in PSW
AC		DC	A(0)	
B0	SAVEXT	DC	F'0'	Transfer address for external interrupt
B4		DC	V(DMSDBG)	Address in LEBU for external interrupt

EXTSECT

Hexadecimal Displacement	Field Name				Field Description, Contents, Meaning
		<u>Storage for External Interrupt Set Up by TRAP</u>			
B8	EXTRET	DS	D		Saved external old PSW
C0	JR0	DC	F'22'		22 doublewords for floating-point registers and user save area
C4	JR1	DC	A(0)		Address of free storage
C8	STIMEXIT	DC	A(0)		Address of STIMER exit routine
CC	EXTFLAG	DC	X'00'	A*1	External flag
	REALTIMR	EQU	X'80'		Real timer indicator
CD		DC	AL3(0)		Reserved for IBM use

EXTUAREA: EXTERNAL USER AREA

EXTUAREA is a 96-byte user area generated by the CMSAVE macro. The pointer to the user area is passed to the user via register 13. The USAVEPTR field in CMSAVE also points to the user area.



Hexadecimal Displacement	Field Name			Field Description, Contents, Meaning
		DS	0D	
0	GRS	DS	16F	Registers at time of interrupt
40	FRS	DS	4D	Floating-point registers at interrupt
60	UEXTPSWO	DS	1D	External old PSW at interrupt
68	USERAREA	DS	18F	User save area
B0	USEREAND	DS	0F	End user area

FCBSECT

FCBSECT: SIMULATED OS CONTROL BLOCKS

FCBSECT consists of the CMS File Control Block (FCB) (used for file management under CMS), the simulated OS Job File Control Block (JFCB), Input/Output Block (IOB), and Data Extent Block (DEB). FCBSECT is invoked via the CMSCE macro. FCBSECT is dynamically allocated from CMS free storage each time the FILEDEF command is issued.

0	FCBNEXT		FCBPROC
8	FCBDD		
10	FCBOP		
18	FCBDSNAM		
20	FCBDSTYP		
28	FCBDSMD		FCBITEM FCEBUFF
30	FCBBYTE		FCEFORM FCBCOUT
38	FCBREAD		A*1 A*2 FCBYTENT
40	FCBRECL		A*3 A*4 FCBMEMBR
48	FCBMEMBR (cont.)		FCBOSFST
50	FCBOSDSN		FCER13
58	FCBKEYS		FCBPDS
60	JFCBMASK		
68	JFCBCRDT		JFCBXPDT A*5 A*6
70	A*7 A*8		JFCBUFL A*9 A*10 A*11
78	JFCLIMCT (cont.)		JFCDSORG A*12 A*13 JFCBLKSI
80	JFCLRECL		A*14 A*15 A*16 A*17 A*18 A*19 A*20 A*21 A*22 A*23 A*24
88	DEBTCBAD		SEBSAV
90	DEBOFLGS		DEBCPATB
98	IOBNXTAD		IOBICB
A0	DEBDCBAD		IOBFCBPT
A8	IOBCSW		
B0	IOBSTART		IOBLCBPT

- Format of Location X'24' for Console Device

20	FCBIOOUT		
28	FCBIOOUT (cont.)		FCBIOBUF
30	A*16 A*17		FCBIOCNT

Hexadecimal Displacement	Field Name			Field Description, Contents, Meaning
0	FCBINIT	DS	0X	Initialization flag bytes
	<u>Bits defined in FCBINIT</u>			
	FCBDOSL	EQU	X'20'	Concatenated DOSLIB data set
	FCBOS	EQU	X'10'	FCB for OS formatted disk
	FCBOPCB	EQU	X'08'	OPEN acquired this CMS block
	FCBPERM	EQU	X'04'	Permanent control block
	FCBBATCH	EQU	X'02'	Special batch data set
	FCBCATML	EQU	X'01'	Concatenated MACLIB data set
0	FCBNEXT	DS	A	AL3(next CMSCB)
4	FCBPROC	DS	A	A(special processing routine)
8	FCBDD	DS	CL8	Data definition name
10	FCBOP	DS	CL8	CMS operation
18	IHAJFCB	DS	0D	Job file control block
18	JFCBDSNM	DS	0X	44 bytes, data set name
18	FCBTAPID	DS	0X	Tape identification
18	FCBDSNAM	DS	CL8	Data set name
20	FCBDSTYP	DS	CL8	Data set type
	FCBPRPU	EQU	FCBDSTYP+4	Printer/punch command list
		ORG	FCBDSTYP+4	
24	FCBIOOUT	DS	CL8	Special I/O command list
2C	FCBIOBUF	DS	A	A(data buffer)
30	FCBCONCR	DS	C	A*16 Console color code
31	FCBCONMS	DS	X	A*17 Console miscellaneous information
32	FCBIOCNT	DS	H	Length of data buffer
28	FCBTBSP	DS	0X	Two bytes for tape backspace count
28	FCBDSMD	DS	CL2	Data set mode
2A	FCBITEM	DS	H	Item identification number
2C	FCBBUFF	DS	F	A(input/output buffer)
30	FCBBYTE	DS	F	Data count
34	FCBFORM	DS	CL2	File format: fixed/variable records
36	FCBCOUT	DS	H	Records per CMS physical block
38	FCBREAD	DS	F	Number of bytes actually read
3C	FCBDEV	DS	X	A*1 Device type code
	<u>Bits defined in FCBDEV</u>			
	FCBCRT	EQU	28	CRT
	FCBPCH	EQU	24	Punch
	FCBDSK	EQU	20	Disk
	FCBTAP	EQU	16	Tape
	FCBCON	EQU	12	Console terminal
	FCBRDR	EQU	8	Reader
	FCBPTR	EQU	4	Printer
	FCBDUM	EQU	0	Dummy device
3D	FCBMODE	DS	X	A*2 Mode: 1, 2, 3, 4, and 5
3E	FCBXTENT	DS	H	Number of items in extent
40	FCBRECL	DS	H	DCB LRECL at open time
42	IOBIOFLG	DS	X	A*3 I/O flags
43	FCBDCBCT	DS	X	A*4 No. of DCEs using this FCB
44	FCBMEMBR	DS	2F	OS PLS member name
4C	FCBOSFST	DS	F	Pointer to OS FST
50	FCBOSDSN	DS	F	Pointer to OS dsname block
54	FCBR13	DS	F	Save area vector R13
58	FCBKEYS	DS	A	A(DDS in-storage key table)
5C	FCBPDS	DS	A	A(PLS in-storage directory)
60	JFCBMASK	DS	8X	Various mask bits
68	JFCBCRDT	DS	3C	Data set creation date (YDD)
6B	JFCBXPDT	DS	3C	Data set expiration date (YED)

FCBSECT

Hexadecimal Displacement	Field Name			Field Description, Contents, Meaning
6E	JFCBIND1	DS	X	A*5 Indicator one
6F	JFCBIND2	DS	X	A*6 Indicator two
70	JFCBUFNO	DS	X	A*7 Number of buffers
71	JFCBFTEK	DS	0X	Buffering technique
71	JFCBFALN	DS	X	A*8 Buffer alignment
72	JFCBUFL	DS	H	Buffer length
74	JFCEROPT	DS	X	A*9 Error option
75	JFCKEYLE	DS	X	A*10 Key length
76		DS	X	Reserved for IBM use
77	JFCLIMCT	DS	3X	A*11 EDAM search limit
7A	FCBDSORG	DS	0X	Data set organization
7A	JFCDSORG	DS	2X	Data set organization
7C	FCBRECFM	DS	0X	Record format
7C	JFCRECFM	DS	X	A*12 Record format
7D	JFCOPTCD	DS	X	A*13 Option codes
7E	FCBBLKSZ	DS	0H	Block size
7E	JFCBLKSI	DS	H	Block size
80	FCBLRECL	DS	0H	Logical record length
80	JFCLRECL	DS	H	Logical record length
82	FCBIOSW	DS	X	A*14 I/O operation indicator
<u>Bits defined in FCBIOSW</u>				
	FCBCLOSE	EQU	X'80'	Switch turned on during CLCSE operation
	FCBCLEAV	EQU	X'40'	DISP=LEAVE during CLOSE operation
	FCBPROCC	EQU	X'20'	GOTO FCBPROC during CLOSE operation
	FCBPROCO	EQU	X'10'	GOTO FCBPROC during OPEN operation
	FCBCASE	EQU	X'08'	ON=LOWER CASE console I/C
	FCBPVMB	EQU	X'04'	PUT-MOVE-VAR-BLK
	FCBIOWR	EQU	X'02'	WRITE/PUT
	FCBIORD	EQU	X'01'	REAL/GET
83	FCBIOSW2	DS	1X	A*15 I/O operation indicators
<u>Bits defined in FCBIOSW2</u>				
	FCBMVFIL	EQU	X'08'	Move file is active
	FCBMMV	EQU	X'02'	Move PDS switch for FIND
	FCBMVPDS	EQU	X'01'	Switch for MOVEFILE with PDS option
84	DEBLNGTH	DS	0X	Length of DEB in doublewords
84		DS	F	Reserved for IBM use
88	IHADEB	DS	0D	Data extent block
88	DEBTCBAD	DS	A	A(move-mode user buffer)
8C	SEBSAV	DS	F	Dynamic save for SEB return address (OS input/output simulation)
90	DEBOFLGS	DS	4X	Data set status flags
94	DEBOPATB	DS	4X	OPEN/CLOSE option byte
98	IOBFLG	DS	0X	Start of IOBPREFIX for normal scheduling
<u>Bits defined in IOBFLG</u>				
	IOBBFLG	EQU	0	Displacement of IOB flag in ICB
	IOBOUT	EQU	X'40'	WRITE,PUT in progress
	IOBIN	EQU	X'20'	READ,GET in progress
	IOBUPD	EQU	X'10'	QSAM PUTX in progress
98	IOBNXTAD	DS	A	A(next buffer to be used)
9C	IOBECB	DS	F	ECB for QSAM normal scheduling
A0	IHAIOB	DS	0F	Input/output block
A0	DEBDEBID	DS	0X	DEB identification
A0	DEBDCBAD	DS	A	A(data control block)
A4	IOBECBCC	DS	0X	ECB completion code
<u>Bits defined in IOBECBCC</u>				
	IOBBECBC	EQU	12	Displacement of ECB code in ICB
	IOBBECBP	EQU	12	Displacement of ECB pointer in IOB

Hexadecimal Displacement	Field Name			Field Description, Contents, Meaning
A4	IOBECBPT	DS	A	A(event control block) - see IHADECB DSECT
A8	IOBFLAG3	DS	0X	I/O error flag
	IOBBCSW	EQU	16	Displacement of CSW in IOB
A8	IOBCSW	DS	8X	Last CCW stored (that is, residual count)
B0	IOBSTART	DS	A	X'ID-NEXT BUFFER',AL3(INITIAL BUFFER)
B4	IOBDCBPT	DS	A	A(data control block)
B8	IOBEND	DS	0X	End of input/output block
B8	FCBEND	DS	0D	End of FCB, JFCB, DEB, and IOB blocks
	FCBENSIZ	EQU	(*-FCBSECT)/8	Size of FCB entry in doublewords

FCHTAB

FCHTAB: FETCH TABLE

FCHTAB contains a fetch/load parameter list that points to a 34-byte directory list. The fetch table is used when a DOS program issues a LOAD or FETCH request without the LIST= parameter. The IJBFTTAB field in the SYSCOM block in the DOSCON CSECT of NUCCON points to the fetch table.

0	FCHAPHNM	A*1	FCHALSNM
8	DIRNAME		
10	DIRTTR	A*2	DIRTT DIRLL
18	A*3 /A*4//	DIRPPP	DIREEE
20	DIRRR	A*5	DIRAAA /A*6//
28	DIRVEE		

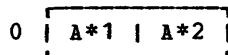
Hexadecimal Displacement	Field Name	Field Description, Contents, Meaning		
<u>8-Byte Parameter List Pointing to Directory List</u>				
0	FCHAPHNM DC	A (DIRNAME)		Address of phase name
4	FCHOPT DC	X'00'	A*1	Options
5	FCHALSNM DS	AL3		Address of listname
<u>34-Byte Directory List</u>				
8	DIRNAME DS	CL8		Phase name
10	DIRTTR DS	XL3		Phase TTR
13	DIRN DS	XL1	A*2	No. of halfwords in directory
14	DIRTT DS	XL2		No. of text blocks in phase
16	DIRLL DS	XL2		Length last text block
18	DIRC DS	XL1	A*3	Flag byte
<u>Bits defined in DIRC</u>				
	SELFREL EQU	X'80'		Phase self-relocatable
	RELPHSE EQU	X'40'		Phase to be relocated
	SVAELIG EQU	X'20'		Phase SVA eligible
	SVAPHSE EQU	X'10'		Phase in SVA
	PCLPHSE EQU	X'08'		Phase in private core image library
	PNOTFND EQU	X'04'		Phase not found
	DACTIVE EQU	X'02'		Phase directory active
	NOTEXT EQU	X'01'		TEXT=NO specified
19	DIRT DS	XL1	A*4	Reserved for IBM use
1A	DIRPPP DS	XL3		Phase load point
1D	DIREEE DS	XL3		Phase entry point
20	DIRRR DS	XL2		No. of RLD items in phase
22	DIRR DS	XL1	A*5	No. of additional RLD blocks
23	DIRAAA DS	XL3		Partition start address
26	DIRK DS	XL1	A*6	Reserved for IBM use
27	DIRVEE DS	XL3		Phase entry point in SVA
	FCHLENG EQU	*-FCHTAB		Total length in bytes (X'2A')
	FCHLENDW EQU	(FCHLENG+7)/8		Total length in doublewords (X'06')

FICL: FIRST IN CLASS BLOCK

FICL is a 2-byte table used in CMS/DOS to address system and programmer logical unit blocks.

Byte 0 of FICL points to the first system class logical unit in the LUB table. This is always the first entry in the LUB table. The second byte points to the first programmer class logical unit in the LUB table partition area.

The FICLPT field in the BGC0M block points to the FICL block.

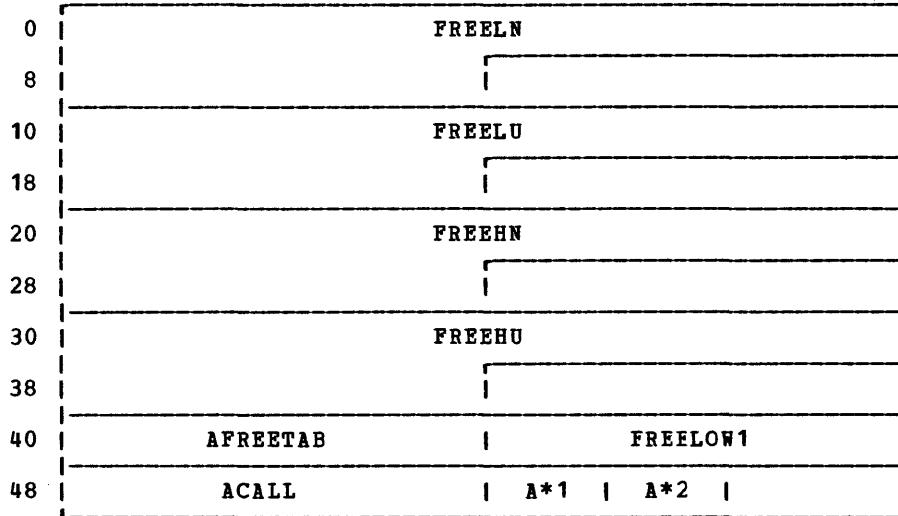


Hexadecimal Displacement	Field Name	Field Description, Contents, Meaning		
0	SYSINDX DC	AL1(0)	A*1	First BG system LUB index
1	PROGINDX DC	AL1(14)	A*2	Programmer LUBs index

FRDSECT

FRDSECT: FREE CHAIN ELEMENT HEADER BLOCKS

FRDSECT describes the fields used by DMSFRE to reference the four free chain element header blocks. FRDSECT is invoked by the macro DMSFRT. The DMSFRT DSECT is pointed to by a V-constant in DMSFREE, and also by the ADMSFRT field in NUCCN.



Hexadecimal Displacement	Field Name	Field Description, Contents, Meaning
<u>The Following Are the Four Free Chain Element Header Blocks</u>		
0	FREELN DC	3F'0',AL1(FLNU+FLPA,NUCKEY,NUCCODE,0) Low storage nucleus
10	FREELU DC	3F'0',AL1(0,USERKEY,USERCODE,0) Low storage user chain
20	FREEHN DC	3F'0',AL1(FLNU+FLHC,NUCKEY,NUCCODE,0) High storage nucleus
30	FREEHU DC	3F'0',AL1(FLHC,USERKEY,USERCODE,0) High storage user chain

The Following Symbolic Equates Describe the Format of Each of the Four Free Chain Element Header Blocks

POINTER	EQU	0	Pointer to first free element
NUM	EQU	4	Number of elements in chain
MAX	EQU	8	Maximum size of an element
FLAGS	EQU	12	Flag byte

Bits defined in FLAGS

FLCLN	EQU	X'80'	Cleanup flag
FLCLB	EQU	X'40'	Overlaid chain flag
FLHC	EQU	X'20'	High storage flag
FLNU	EQU	X'10'	Nucleus flag
FLPA	EQU	X'08'	Page available on chain
SKEY	EQU	13	Storage key for this chain

Bits defined in SKEY

USERKEY	EQU	X'E0'	User storage key
NUCKEY	EQU	X'F0'	Nucleus storage key

Hexadecimal Displacement	Field Name		Field Description, Contents, Meaning
	TCODE	EQU 14	FREEFAB table code
	<u>Bits defined in TCODE</u>		
	USERCODE	EQU 1	User free storage page
	NUCCODE	EQU 2	Nucleus free storage page
	TRNCODE	EQU 3	Transient area page
	USARCODE	EQU 4	User area page
	SYSCODE	EQU 5	System page
	MAXCODE	EQU 5	Maximum possible code value
	*UNUSED	EQU 15	
	BLOCKLEN	EQU 16	Symbolic length of block
40	AFREETAB	DC A(0)	Address of FREEFAB table
44	FREELOW1	DS F	Original value of FREELOWE (set by INIT2)
48	ACALL	DS A	Address of caller (for errors)

Flags Set by Examining SVC 203 Halfword Code

4C FREEFLG1 DC BL1'0' A*1

Bits defined in FREEFLG1

FRF1C	EQU	X'80'	Conditional request
FRF1V	EQU	X'40'	Variable request
FRF1N	EQU	X'20'	Nucleus request
FRF1E	EQU	X'10'	FREE (vs FRET) request
FRF1L	EQU	X'08'	Low storage is OK
FRF1H	EQU	X'04'	High storage is OK
FRF1M	EQU	X'02'	Messages wanted on error
FRF1B	EQU	X'01'	TYPICAL equals BALR in macro

The Following Byte Holds Flags Internal to the DMSFRE Routine

4D FREEFLG2 DC BL1'0' A*2

Bits defined in FREEFLG2

FRF2CL	EQU	X'80'	Cleanup flag
FRF2SVP	EQU	X'40'	Variable pages request flag (SCHVPGE)
FRF2NOI	EQU	X'20'	Second initialization routine has not yet been called by DMSINS
FRF2CKE	EQU	X'10'	Do a check each time FREE or FRET is called
FRF2CKT	EQU	X'08'	Do a check this time
FRF2CKX	EQU	X'04'	Executing CHECK routine now

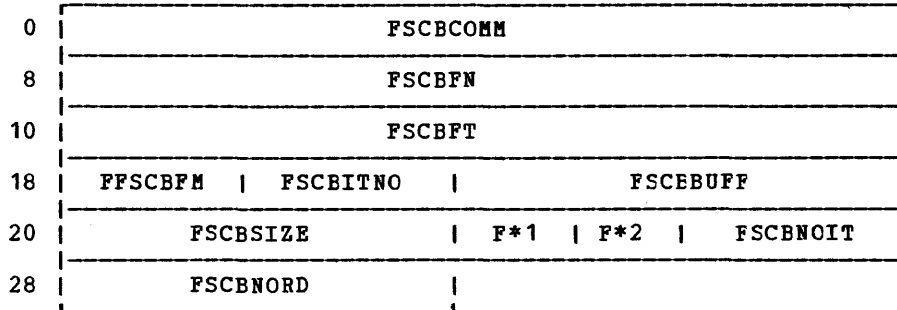
Free Chain Element Description

POINTER	EQU	0	Pointer to next FREE element
SIZE	EQU	4	Size of this element in bytes

FSCBD

FSCBD: FILE SYSTEM CONTROL BLOCK

FSCBD is a PLIST defined for general use by routines that use the CMS file system. FSCBD is generated when the user invokes the FSCBD macro.



Hexadecimal Displacement	Field Name	Field Description, Contents, Meaning		
0	FSCBCOMM DS CL8	File system command (RDBUF, WRBUF, etc.)		
8	FSCBFN DS CL8	Filename		
10	FSCBFT DS CL8	Filetype		
18	FSCBFM DS CL2	Filemode		
1A	FSCBITNO DS H	Relative record number to be read/written		
1C	FSCBBUFF DS F	Address of read/write buffer or of STATEPST		
20	FSCBSIZE DS F	Length of buffer		
24	FSCBFV DS CL2	F*1	RECFM -- C'F' or C'V'	
25	FSCBFLG EQU FSCBFV+1	F*2	Flag byte	
26	FSCBNOIT DS H	Number of records to be read/written		
28	FSCBNORD DS A	Number of bytes actually read		

FSTD: FILE STATUS TABLE ENTRY DSECT

FSTD describes the fields in a 40-byte file status table entry as found by STATE, STATEW, DMSLFS or DMSLFSW. FSTD is functionally equivalent to the FSTSECT DSECT.

0	FSTFNAME				
8	FSTFTYPE				
10	FSTDATEW	FSTTIMEW	FSTWRPNT	FSTRDPNT	
18	FSTFMODE	FSTRECCT	FSTFCLPT	F*1	F*2
20	FSTLRECL		FSTELKCT	FSTYEARW	

Hexadecimal Displacement	Field Name			Field Description, Contents, Meaning
0	FSTFNAME	DS	1D	Filename
8	FSTFTYPE	DS	1D	Filetype
10	FSTDATEW	DS	1H	Date last written - mddy
12	FSTTIMEW	DS	1H	Time last written - hhmmss
14	FSTWRPNT	DS	1H	Write pointer - item number
16	FSTRDPNT	DS	1H	Read pointer - item number
18	FSTFMODE	DS	1H	Filemode - letter and number
1A	FSTRECCT	DS	1H	Number of logical records
1C	FSTFCLPT	DS	1H	First chain link pointer
1E	FSTRECFM	DS	1C	F*1 Record format (F or V)
1F	FSTFLAGS	DS	1X	F*2 FST flag byte

Bits defined in FSTFLAGS

FSTXWDSK	EQU	X'C0'	Extension of read/write disk
FSTRWDSK	EQU	X'80'	Read/write disk
FSTXRDSK	EQU	X'40'	Extension of read-only disk
FSTFILEA	EQU	X'07'	File is active (one of the following)
FSTACTRD	EQU	X'04'	File active for reading
FSTACTWR	EQU	X'02'	File active for writing
FSTACTPT	EQU	X'01'	File active from a point
FSTRODSK	EQU	X'00'	Read-only disk

Bits redefined for use in RDBUF

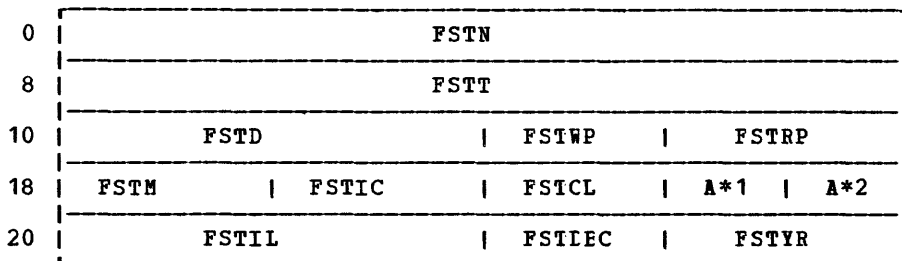
FSTDIA	EQU	X'40'	Item available	
FSTDRA	EQU	X'01'	Previous record null	
FSTDNI	EQU	X'00'	Null record	
20	FSTLRECL	DS	1F	Logical record length
24	FSTBLKCT	DS	1H	Number of 800-byte blocks
26	FSTYEARW	DS	1H	Year last written
	FSTDSIZE	EQU	(*-FSTD)	FST size in bytes

FSTSECT

FSTSECT: FILE STATUS TABLE

FSTSECT defines the file status table (FST) which describes the attributes of a file on a CMS virtual disk. FSTSECT is invoked by the macro FSTB.

The file status tables for all files on the disk are grouped into 800-byte disk records referred to as file status table blocks (FSTEs). Each file status table block can accommodate up to 20 file status tables.



Hexadecimal Displacement	Field Name			Field Description, Contents, Meaning
0	FSTN	DS	1D	Filename
8	FSTT	DS	1D	Filetype
10	FSTD	DS	1F	Date/time last written
14	FSTWP	DS	1H	Write pointer (item number)
16	FSTRP	DS	1H	Read pointer (item number)
18	FSTM	DS	1H	Filemode
1A	FSTIC	DS	1H	Item count
1C	FSTFCL	DS	1H	First chain link
1E	FSTFV	DS	1C	A*1 Fixed(F)/variable(V) flag
1F	FSTFB	DS	1C	A*2 Flag byte (if used)

Bits defined in FSTFB (Applicable only to STATEFST copy of FST-entry after successful STATE or STATEW call)

FSTFRWX	EQU	X'CO'		Read-only extension of read/write disk
FSTFRW	EQU	X'80'		Read/write disk
FSTFROX	EQU	X'40'		Read-only extension of read-only disk
FSTFACT	EQU	X'07'		File is active (one of the following)
FSTFAR	EQU	X'04'		File active for reading
FSTFAW	EQU	X'02'		File active for writing
FSTFAP	EQU	X'01'		File active from a designated point
FSTFRO	EQU	X'00'		Read-only disk

Bits redefined for use in RDEUF

FSTITAV	EQU	X'40'		Item available
FSTRECAV	EQU	X'01'		Previous record null
FSTNOIT	EQU	X'00'		Null record

20	FSTIL	DS	1F	Maximum item length
24	FSTDBC	DS	1H	800-byte data block count
26	FSTYR	DS	1H	Year

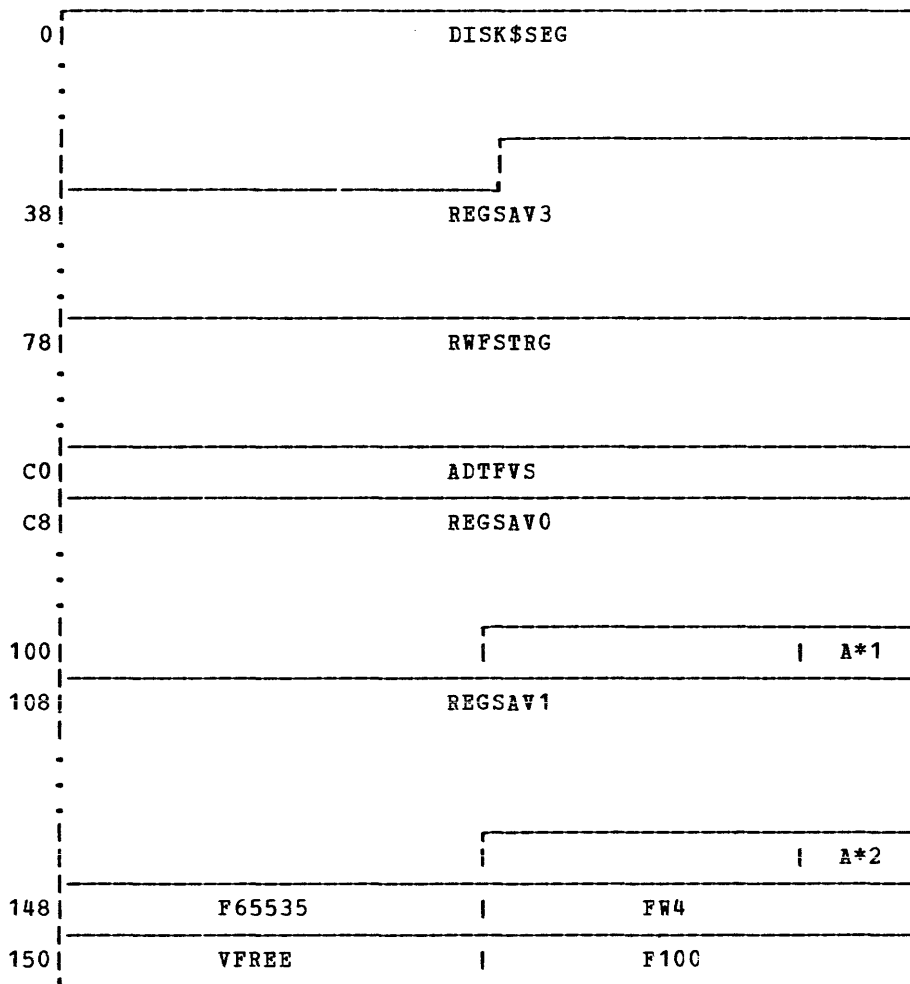
FSTL EQU *-FSTSECT Size of FST in bytes (X'28')

FST Hyperblock Parameters

FSTFWDP	EQU	800		Forward pointer to next hyperblock in storage
FSTBKWD	EQU	804		Backward pointer to previous hyperblock in storage

FVSECT: FIXED VARIABLE STORAGE WORK AREA FOR CMS FILE SYSTEM

FVSECT is used mainly by file management and I/O routines. FVS contains save areas, work areas, and commonly used constants. A typical use of FVS is when a reentrant I/O routine requires a work area or save area, since the routine cannot modify itself. FVSECT is invoked by the FVS macro.



FVSECT

158	VFRET		JSRC		
160	JSR1		RWMFD		
168	F800				
170	FVSDSKA		DSKLOC		
178	RWCNT		LSKADR		
180	ADTADD				
188			FINISLST		
.					
.					
198				FFF	
1A0	FFE		FFD		SIGNAL A*3 A*4
1A8	/////		A*5 A*6 A*7		FVSERAS0
1B0	FVSERAS1		FVSERAS2		
1B8	READCNT		////////////////////////////////////		
1C0			FVSFSTN		
1C8			FVSFSTT		
1D0	FVSFSTDT		FVSFSTWP		FVSFSTRP
1D8	FVSFSTM		FVSFSTIC		FVSFSTCL A*8 A*9
1E0	FVSFSTIL		FVSFSTDB		FVSFSTYR
1E8	FVSFSTAD		FVSFSTAC		

Hexadecimal Displacement	Field Name			Field Description, Contents, Meaning
0	DISK\$SEG	DS	15F	For FSTLKP, FSTLKW, ACTLKP, TRKLP, QQTRK
3C	REGSAV3	DS	15F	For RDBUF, WRBUF, FINIS, STATE, POINT
78	RWFSTRG	DS	18F	Remaining storage for RDBUF, WRBUF, FINIS
C0	ADTFVS	DC	2F'0'	ADTLKP
<u>Save Area for Lowest Level Routines</u> (For example, <u>READMFD</u> , <u>RELUFD</u> , <u>UPLISK</u> , <u>TYPSRCH</u> , and <u>ADTLKW</u>)				
C8	REGSAV0	DS	15F	Saved R0-R15
104		DC	AL3(00)	First 3 bytes of return code
107	ERRCOD0	DC	AL1(*-*) A*1	Error code
	TRKLSAVE	EQU	REGSAV0	For TRKLP/X only when called by QQTRK/X
<u>Save Area for Next-to-Lowest Level Routines</u> (For example, <u>READFST</u> , <u>ERASE</u> , <u>ALTER</u> , and <u>INTSVC-LOADMOD</u>)				
108	REGSAV1	DS	15F	Register save area
144		DC	AL3(00)	First 3 bytes of return code
147	ERRCOD1	DC	AL1(*-*) A*2	Error code
148	F65535	DC	F'65535'	= X'C000FFFF'
14C	FW4	DC	F'4'	Constant value
	HW4	EQU	FW4+2	Constant value
150	VFREE	DC	V(FREE)	Constant value
154	F100	DC	F'100'	Constant value
158	VFRET	DC	V(FRET)	Address of FRET (into R15)
15C	JSR0	DC	F'0'	R0 saved here for FRET calls
160	JSR1	DC	F'0'	R1 saved here for FRET calls
<u>PLIST to Read/Write MFD</u>				
164	RWMFD	DC	A(*-*)	Address of MFD
168	F800	DC	F'800'	800 bytes
16C		DC	A(HW4)	
170	FVSDSKA	DC	A(*-*)	Address of the active disk table
174	DSKLST	DS	0F	All-purpose RDTK/WRTK PLIST
174	DSKLOC	DS	A(*-*)	Address of item to be read or written
178	RWCNT	DC	A(*-*)	Byte count (usually 800)
17C	DSKADR	DC	A(*-*)	Disk address of item to be read or written
180	ADTADD	DC	A(*-*)	Address of active disk table now in use
184	FINISLST	DC	CL8'FINIS'	PLIST to close all files
18C		DC	CL8'**'	
194		DC	CL8'**'	
19C		DC	CL2'**'	
19E		DS	0H	Halfword constants
19E	FFF	DC	X'FFFF'	Means no significant data past 215th byte
1A0	FFE	DC	X'FFFE'	1968-era MFD still supported on input only
1A2	FFD	DC	X'FFFD'	Newest signal for 2314 handling

FVSECT

Hexadecimal Displacement	Field Name			Field Description, Contents, Meaning
1A4	SIGNAL	DC	H'0'	SIGNAL = Scratch halfword used by READMFD or ERASE = C0C0, X'FFFF', X'FFFE', or X'FFFD'
<u>Bits defined in SIGNAL</u>				
	SWTCH	EQU	SIGNAL+1	00, FF, FE, or FD
1A6	UFDBUSY	DC	X'00'	A*3 Nonzero means UFD is being updated
<u>Bits defined in UFDBUSY</u>				
	WRBIT	EQU	X'80'	WREUF
	UPBIT	EQU	X'40'	UPDISK - READMFD
	PNBIT	EQU	X'20'	FINIS
	ERBIT	EQU	X'10'	ERASE - ALTER - READFST
	DIOBIT	EQU	X'08'	RDTK/WRTK
<u>Bits for Routines That Do Not Update the Disk, but That Cannot Be Interrupted by a HX Command</u>				
	ABNBIT	EQU	X'02'	DMSAEN (abend recovery routine)
	ITSBIT	EQU	X'01'	DMSITS (SVC handling routine)
1A7	KXFLAG	DC	X'00'	A*4 HX flags
<u>Bits defined in KXFLAG</u>				
	KXWANT	EQU	X'80'	HX wanted as soon as possible
	KXWSVC	EQU	X'01'	Hold HX until any SVC activity
1A8		DC	X'00'	Reserved for IBM use
1A9	PLGSAVE	DC	X'00'	A*5 For scratch use (for example, by RELUFD)
1AA	FVSFLAG	DC	X'00'	A*6 For general use (as needed)
<u>Miscellaneous Storage Used by ERASE (or RENAME)</u>				
1AB	ERSFLAG	DC	X'00'	A*7 Flag for use by ERASE or RENAME
1AC	FVSERAS0	DC	F'0'	R0 to/from FSTLKW (for ERASE)
1B0	FVSERAS1	DC	F'0'	R1 to ACTLKE or FSTLKW (for ERASE)
1B4	FVSERAS2	DC	F'0'	Address of free storage used by ERASE
1B8	READCNT	DC	F'0'	Current read count (DMSBRD)
1BC		DC	F'0'	Reserved for IBM use
<u>File Status Table (FST) Copy from STATE</u>				
1C0	STATEFST	DS	0D	Full FST of file (STATE)
1C0	FVSFSTN	DC	D'0'	Filename
1C8	FVSFSTT	DC	D'0'	Filetype
1D0	FVSFSTD	DC	2H'0'	Date/time last written
1D4	FVSFSTWP	DC	H'0'	Write pointer (item ID)
1D6	FVSFSTRP	DC	H'0'	Read pointer (item ID)
1D8	FVSFSTM	DC	H'0'	Filemode
1DA	FVSFSTIC	DC	H'0'	Number of items in file
1DC	FVSFSTCL	DC	H'0'	Disk address (first chain link)
1DE	FVSFSTFV	DC	C' '	A*8 Fixed(F)/variable(V) indicator
1DF	FVSFSTFB	DC	X'00'	A*9 Flag byte
1E0	FVSFSTIL	DC	F'0'	Length of largest item in file
1E4	FVSFSTDB	DC	H'0'	Number of data blocks
1E6	FVSFSTYR	DC	2C' '	Year last written
1E8	FVSFSTAD	DC	A(0)	A(active disk table for this file)
	STATER0	EQU	FVSFSTAD	
1EC	FVSFSTAC	DC	A(0)	A(real FST entry for this file)
	STATER1	EQU	FVSFSTAC	

IHADECB: DATA EVENT CONTROL BLOCK

IHADECB, which is invoked via the CMSCB macro, is the simulated data event control block used for CMS processing of OS macros and OS access methods. The ICBECP field in FCBSECT points to IHADECB.

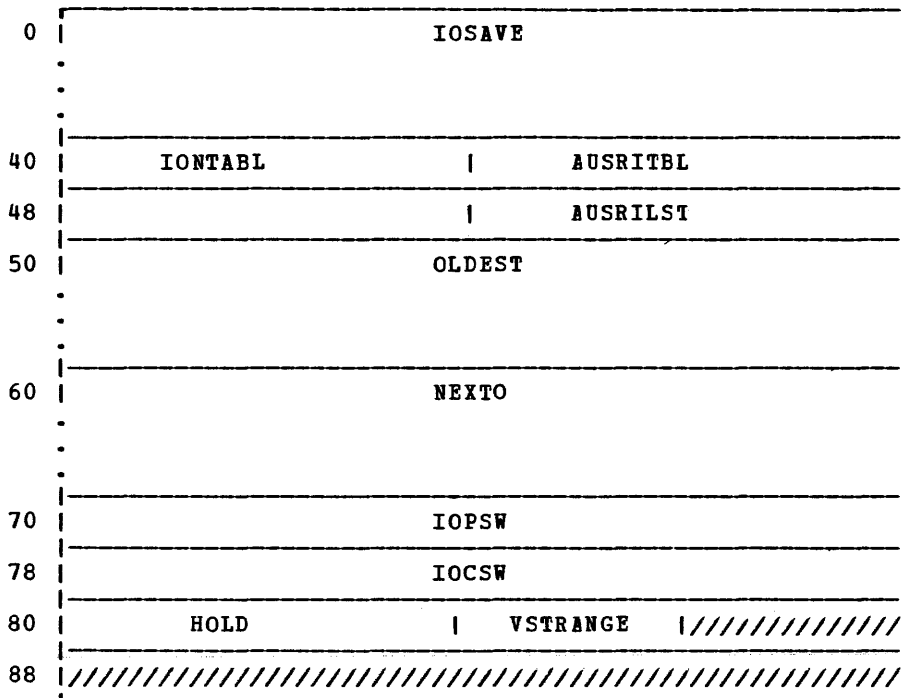
0	DECSDECB	DECTYPE	DECLNGTH
8	DECDCBAD	DECAREA	
10	DECIOBPT	DECKYADR	
18	DECRCPT		

Hexadecimal Displacement	Field Name			Field Description, Contents, Meaning
0	DECSDECB	DS	F	Event control block
4	DECTYPE	DS	H	Type of I/O request
	<u>Bits defined in DECTYPE</u>			
	DECBRD	EQU	X'80'	Read SF
	DECBWR	EQU	X'20'	Write SF
6	DECLNGTH	DS	H	Length of key and data
8	DECDCBAD	DS	A	V(data control block)
C	DECAREA	DS	A	V(key and data, buffer)
10	DECIOBPT	DS	A	V(IOE)
	<u>BDAM Extension</u>			
14	DECKYADR	DS	A	V(key)
18	DECRCPT	DS	A	V(block reference field)
	<u>Frequently Used Equates</u>			
	DDNAM	EQU	FCBDSTYP	Filetype = data set name
	BLK	EQU	X'10'	RECFM=blocked records
	BS	EQU	X'20'	MACRF=BSAM
	DA	EQU	X'20'	DSORG=direct access
	FXD	EQU	X'80'	RECFM=fixed-length records
	IS	EQU	X'80'	DSORG=indexed sequential
	LOC	EQU	X'08'	MACRF=locate mode
	MOV	EQU	X'10'	MACRF=move mode
	PS	EQU	X'40'	DSORG=physical sequential
	PO	EQU	X'02'	DSORG=partitioned organization
	PREVIOUS	EQU	X'80'	OFLGS=previous I/O operation
	QS	EQU	X'40'	MACRF=QSAM
	UND	EQU	X'C0'	RECFM=undefined format records
	VAR	EQU	X'40'	RECFM=variable-length records

IOSECT

IOSECT: I/O INTERRUPT SAVE AREA

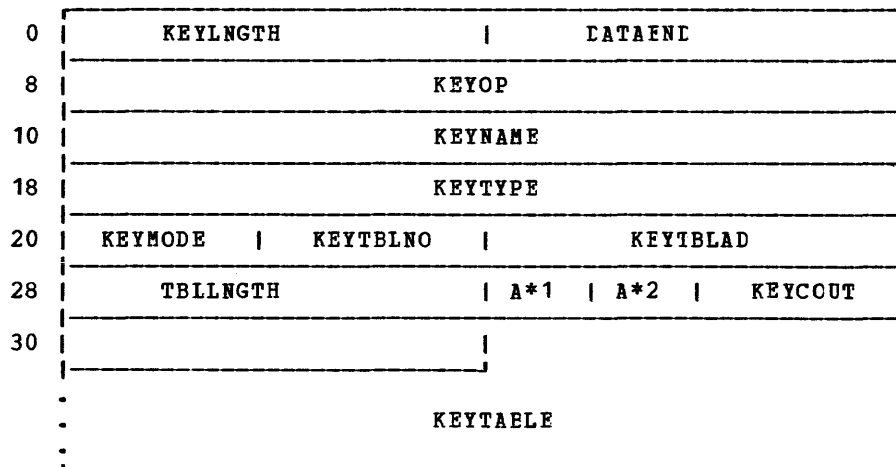
IOSECT describes the fields used by DMSITI for save registers, I/O old PSW, and other data when handling I/O interrupts. IOSECT is pointed to by the AIOSECT field in NUCON.



Hexadecimal Displacement	Field Name	Field Description, Contents, Meaning
0	IOSAVE DS 16F	Register save area
40	IONTABL DC F'0'	Size of user interrupt table in doublewords
44	AUSRITBL DC A(0)	Address of user interrupt table
48	DC F'28'	Length of each entry
4C	AUSRILST DC A(0)	Address of last entry in table
50	OLDEST DS 4F	Oldest I/O old PSW and CSW
60	NEXTO DS 4F	Next oldest I/O old PSW and CSW
70	IOPSW DS 2F	Newest I/O old PSW
78	IOCSW DS 2F	Newest CSW
80	HOLD DC F'0'	Holds entry pointer for device
84	VSTRANGE DC H'0'	Unknown device address saved here
86	DC 1H'0'	Reserved for IBM use
88	DC 2F'0'	Reserved for IBM use

KEYSECT: DISK KEY TABLE DSECT FOR BDAM SIMULATION

KEYSECT defines the key table used in OS simulation of BDAM files for I/O by key. KEYSECT is built dynamically from CMS free storage.

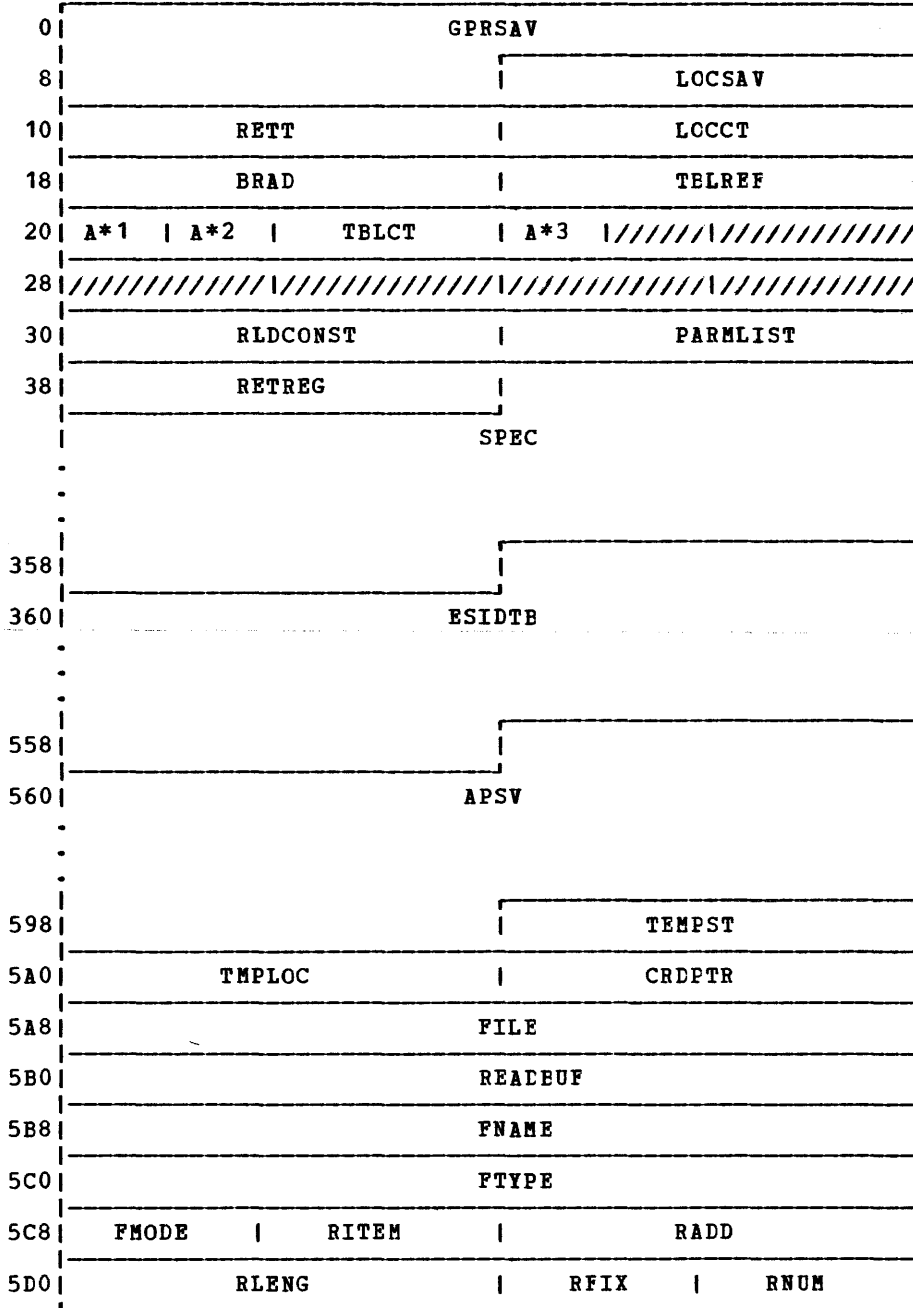


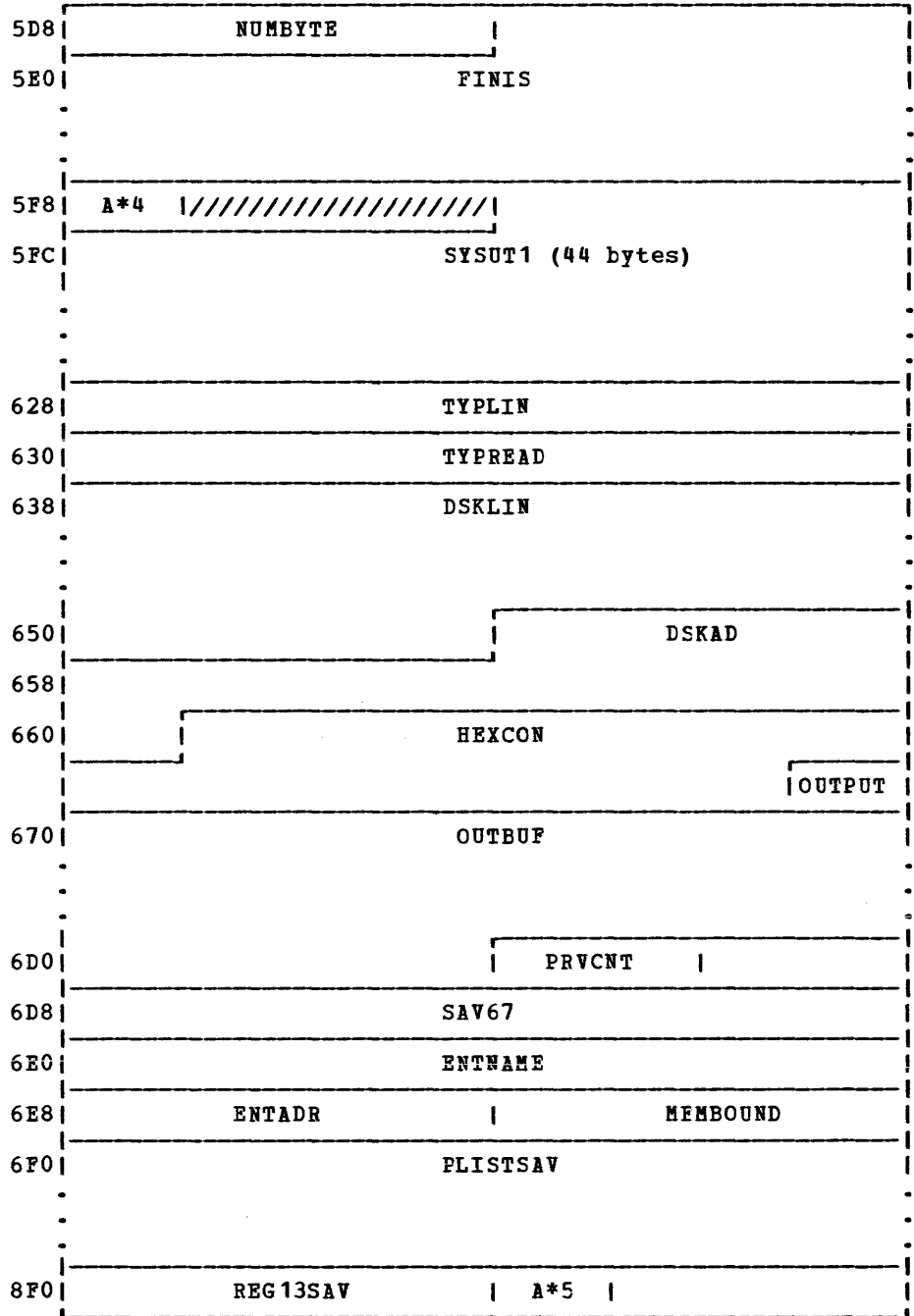
Hexadecimal Displacement	Field Name			Field Description, Contents, Meaning
0	KEYLNPTH	DS	1F	Key length
4	DATAEND	DS	1F	Pointer to last data item in file
8	KEYOP	DS	2F	Start of PLIST for keys file
10	KEYNAME	DS	2F	Filename of keys file
18	KEYTYPE	DS	2F	Filetype of keys file
20	KEYMODE	DS	1H	Filemode of keys file
22	KEYTBLNO	DS	1H	Item number of key table
24	KEYTBLAD	DS	1F	Address of key table
28	TBLLNPTH	DS	1F	Byte size of key table
2C	KEYFORM	DS	1X	A*1 Format of keys file
2D	KEYCHNG	DS	1X	A*2 Indicates change in key table
2E	KEYCOUT	DS	1H	Blocking factor of key table
30		DS	1F	Number of bytes read
34	KEYTABLE	DS	0F	Start of keys table (item number)

LDRST

LDRST: LOADER STORAGE AREA

LDRST describes the fields of the work area used by the loader. The work area is obtained and built by DMSLDR. LDRST is built dynamically by DMSLDR from CMS free storage.





LDRST

Hexadecimal Displacement	Field Name			Field Description, Contents, Meaning
0	GPRSAV	DS	3F	R9 through R12
C	LOCSAV	DS	F	Base register contains A (DMSLDRA)
10	RETT	DS	F	Return register for DMSLSB
14	LOCCT	DS	F	(LOC CNT) next load location
18	BRAD	DS	F	(STR TADDR) start execution address
1C	TBLREF	DS	F	(ALDR TBL) top of loader table
20	FLAG1	DS	X	A*1 Loader switches (permanent)
<u>Bits defined in FLAG1</u>				
	ABSOLUTE	EQU	X'80'	Absolute loading
	FSTXTADR	EQU	X'40'	First text address saved
	COMMONEX	EQU	X'20'	Common entries exist in loader table
	PREXIST	EQU	X'10'	PR entries exist in loader table
	ENDCDADR	EQU	X'08'	Allow end card address
	NOERASE	EQU	X'04'	Do not erase the load map
	WORKFILE	EQU	X'02'	Work file (SYSUT1) exists
	NODUP	EQU	X'01'	Do not type message DMSLI0202W
21	FLAG2	DS	X	A*2 Loader switches (permanent)
<u>Bits defined in FLAG2</u>				
	STRINITC	EQU	X'80'	Call STRINIT in LOADMOD
	NOMAP	EQU	X'40'	Do not create a load map
	APRILB	EQU	X'20'	REP card processing control
	NOAUTO	EQU	X'10'	No automatic text deck checking
	TYPE	EQU	X'08'	Type load map at terminal
	NOREP	EQU	X'04'	No REP card printing
	NOINV	EQU	X'02'	No invalid card typeout
	NOLIBE	EQU	X'01'	No automatic TXT library searching
22	TBLCT	DS	H	Number of entries in loader table
24	FLAG3	DS	X	A*3 More flags
<u>Bits defined in FLAG3</u>				
	CMD	EQU	X'80'	Processing names from command list
25		DS	X	Reserved for IBM use
26		DS	5H	Reserved for IBM use
30	RLDCONST	DS	F	Relocation constant
34	PARMLIST	DS	F	Updated parameter list pointer
38	RETREG	DS	F	Return register
3C	SPEC	DS	200F	10-card input buffer
35C	ESIDTB	DS	256H	256 ESD entries; object deck
55C	APSV	DS	16F	Register save area for subroutine calls
59C	TEMPST	DS	F	Temporary RLD routine storage
5A0	TMPLOC	DS	F	Temporary storage
5A4	CRDPTR	DS	F	Input card pointer
5A8	FILE	DS	D	Save location for DMSLIB
5B0	READBUF	DS	2F	Input read parameter list
5B8	FNAME	DS	2F	Filename
5C0	FTYPE	DS	2F	Filetype
5C8	FMODE	DS	H	Filemode
5CA	RITEM	DS	H	Number of items
5CC	RADD	DS	F	Buffer address
5D0	RLENG	DS	F	Buffer length
5D4	RFIX	DS	H	Fixed/variable flag byte
5D6	RNUM	DS	H	Number of items

Hexadecimal Displacement	Field Name	Field Description, Contents, Meaning		
5D8	NUMBYTE DS F			Number of bytes actually read
5DC	FINIS DS 7F			FINIS parameter list
5F8	FLAGS DS X	A*4		Loader switches (nonpermanent)
	START EQU X'80'			Start execution requested
	ONEDYNA EQU X'40'			One call to dynamic loading per text file
	ESD1ST EQU X'20'			First ESD data item this card
	NOSLCADR EQU X'10'			No address field in SLC card
	SETLIB EQU X'08'			Set up for library searching
	CLOSELIB EQU X'04'			Clear TXTLIB searching
	LUNDEF EQU X'02'			Undefined entries exist in loader table
	RESET EQU X'01'			Reset "entry" specified
5F9		DS	3X	Library search work area pointer
5FC	SYSUT1 DS 11F			RLD work file PLISTs
628	TYPLIN DS 2F			TYPLIN parameter list
630	TYPEAD DS 2F			TYPLIN buffer address
638	DSKLIN DS 7F			Disk parameter list for load map
654	DSKAD DS 13X			
661	HEXCON DS 14X			Hexadecimal constant
	PACK EQU HEXCON			Hexadecimal constant
	UNPACK EQU HEXCON+5			Hexadecimal constant
66F	OUTPUT DS X			
670	OUTBUF DS 100X			Output buffer for load map and terminal printing
6D4	PRVCNT DS H			Address of next PR load address
6D8	SAV67 DS 2F			Temporary save area of R6 and R7
6E0	ENTNAME DS CL8			Entry name (reset ENTRY or entry control card)
6E8	ENTADR DS F			Entry name's loader table location
6EC	MEMBOUND DS F			Low extend of free storage (FREELOWE)
6F0	PLISTSAV DS 64D			LOAD (INCLUDE) PLIST saved
8F0	REG13SAV DS F			Address of LDRST
8F4	FRSTSDID DS X	A*5		First section definition identification
8F8	ENDFREE DS 0D			
	NEED EQU (ENDFREE-LDRST)/8			
Note: The following equates refer to displacements and flags in the REFTABLE entry usually pointed to by register 12				
	REFNAME EQU 0			Displacement of 8-byte name field
	REFLG1 EQU 8			Displacement of flag byte 1
	REFPRB EQU X'7C'			PR - byte alignment
	REFPRH EQU X'7D'			PR - halfword alignment
	REFPRF EQU X'7E'			PR - fullword alignment
	REFPRD EQU X'7F'			PR - doubleword alignment
	REFUND EQU X'80'			Undefined symbol
	REFCXD EQU X'81'			Resolve CXD
	REFCOM EQU X'82'			Define common area
	REFWEX EQU X'83'			Weak external reference
	REFNOB EQU X'90'			LIEF card - nonobligatory
	REFLIB EQU X'10'			Single bit for nonobligatory LIBE card
	REFINFO EQU 9			Displacement of relocation factor or maximum address
	REFVAL EQU 13			Displacement of absolute or assigned value
	REFLG2 EQU 16			Displacement of flag byte 2
	REFCMD EQU X'80'			Command line name - must resolve

LUBTAB, LUBPR

LUBTAB AND LUBPR: LOGICAL UNIT BLOCK TABLE

LUBTAB is a device table that has a 2-byte entry for each symbolic name used by CMS/DOS. The simulated LUB has 255 entries: 14 entries for the system logical units and 241 entries for programmer logical units. System devices (SYSRDR, SYSIPT, SYSPCH, SYSLST, and SYSLOG) can be assigned to alternate devices. The system and programmer tables are defined with separate DSECTS: LUBTAB and LUBPR. LUBTAB is pointed to by the LUBPT field in BGC0M. The address of the first LUB entry is in the first byte of the FICL control block.

System (LUBTAB)

0	LUBRDR	LUBIPT	LUBPCH	LUBLST
8	LUBLOG	/// <u>LUBLNK</u> /////	LUBRES	LUBSLB
10	LUBRLB	/// <u>LUBUSE</u> /////	/// <u>LUBREC</u> /////	LUBCLB
18	/// <u>LUBVIS</u> /////	LUBCAT		

Programmer (LUBPR)

0	LUB000	LUB001	LUB002	LUB003
8	LUB004 through LUB239			
.
.
1E0	LUB240	LUB241		

Hexadecimal Displacement	Field Name			Field Description, Contents, Meaning
	<u>System LUBs</u>			
0	LUBRDR	DS	XL2	System virtual reader
2	LUBIPT	DS	XL2	System virtual input device
4	LUBPCH	DS	XL2	System virtual punch
6	LUBLST	DS	XL2	System virtual printer
8	LUBLOG	DS	XL2	Terminal
A	LUBLNK	DS	XL2	Reserved for IBM use
C	LUBRES	DS	XL2	System residence volume
E	LUBSLB	DS	XL2	Private source statement library
10	LUBRLB	DS	XL2	Private relocatable library
12	LUBUSE	DS	XL2	Reserved for IBM use
14	LUBREC	DS	XL2	Reserved for IBM use
16	LUBCLB	DS	XL2	Private core image library
18	LUBVIS	DS	XL2	Reserved for IBM use
1A	LUBCAT	DS	XL2	VSAM catalog
	<u>Programmer LUBs</u>			
0	LUB000	DS	XL2	Programmer logical unit block
2	LUB001	DS	XL2	Programmer logical unit block
4	LUB002	DS	XL2	Programmer logical unit block
6	LUB003	DS	XL2	Programmer logical unit block
8	.	.	.	LUB004 through LUB239 are defined with DS and XL2. Each is a programmer logical unit block.
.	.	.	.	
.	.	.	.	
1E0	LUB240	DS	XL2	Programmer logical unit block

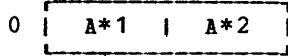
Hexadecimal Displacement	Field Name			Field Description, Contents, Meaning
1E2	LUB241	DS	XL2	Programmer logical unit block
	LUBP	EQU	0	Displacement to PUB pointer
	LUBJ	EQU	1	Displacement to JIB pointer
	LUBL	EQU	*-LUB241	LUB length

NICL

NICL: NUMBER IN CLASS

Byte 0 of the Number In Class block (NICL) contains the number of system class logical units. The second byte contains the number of programmer class logical units for the partition.

The NICLPT field in the BGCOP block points to the NICL block.



<u>Hexadecimal Displacement</u>	<u>Field Name</u>	<u>Field Description, Contents, Meaning</u>			
0	NOSYS	DC	AL1(14)	A*1	Total number of system LUBs
1	NOPROG	DC	AL1(242)	A*2	Total number of programmer LUEs

NUCON: NUCLEUS CONSTANT AREA

NUCON is the nucleus constant area of CMS.

0	IPLPSW
8	IPLCCW1
10	IPLCCW2
18	EXTOPSW
20	SVCOPSW
28	PGHOPSW
30	MCKOPSW
38	IOOPSW
40	CSW
48	CAW /////////////////NUCRSV1/////////////////
50	TIMER /////////////////NUCRSV2/////////////////
58	EXTNPSW
60	SVCNPSW
68	PGMNPSW
70	MCKNPSW
78	IONPSW
80	CPULOG
88	
90	/////////////////NUCRSV4///////////////// MONCLASS PERCODE
98	PERADDR MCNCODE
A0	/////////////////NUCRSV5/////////////////
	/////////////////
	/////////////////
C0	LOWSAVE
.	
.	
160	FPRLOG
.	
.	
180	GPRLOG
.	
.	

1C0	ECRLOG		
200	SYSTEMID		
220	INSTALID		
260	SYSNAME		
268	IPLADDR	SYSADDR	DEVICE
270	/////////////////NUCRSV6/////////////////		
278	FEIBM		
280	CURRDATE		
288	CURRTIME		
290	CURRVIRT		CURRCPUT
298	LASTVIRT		LASTCPUT
2A0	LASTCMND		
2A8	PREVCMND		
2B0	LASTEXEC		
2B8	PREVEXEC		
2C0	LASTLMOD		
2C8	LASTTMOD		
2D0	DATIPCMS		
2D8	CLKVALMD		
2E0	MACDIRC		
300	MACLIBL		
348	TXLIBSV		MACLBSV
350	TOTLIBS		TXDIRC

358	TXTLIES	
3A0	GRS015	LOC0176
3A8	FIRSTDMP	LASTDMP
3B0	FRS06	DMPTIT
3B8	////////////////////////////////////	
	DMPTITLE	
440	////////////////////////////////////	SVC\$202
448	ERR\$202	////////////////////////////////////
450	A*1 A*2 ////////////////////////////////////	AEATPROC
458	ABATABND	AEATLMT
460	AUSERST	////////////////////////////////////
468	////////////////////////////////////	DOSLBSV
470	DOSDIRC	
490	DOSLIEL	
4D8	A*3 A*4 ////////////////////////////////////	ALTASAVE
4E0	ABGCOM	ASYSOM
4E8	ADOSDCSS	SVC12SAV
4F0	DOSFIRST	DOSNUM ////////////////////////////////////
4F8	APPSAVE	DOSTRANS
500	MAINLIST	MAINSTR
508	FREELIST	FREENUM
510	MAINHIGH	FRELOWE
518	FREELWR	FREEUPPR
520	ANUCEND	AUSRAREA

528	CURRSAVE		CODE203		PCTCHSFS
530	ADMSFRT				VCADTLKP
538	VCADTNXT				VCADTLKW
540	CURRIOOP				PENDREAD
548	PENDWRIT				FSTFINRD
550	LSTFINRD				AINTRTBL
558	AOUTRTBL		NUMFINRD		NUMPNDWR
560	VMSIZE				ALIRTBLS
568	STRTADDR				FRSTLOC
570	LASTLOC				LOCCNT
578	LDRADDR				LDRRTCD
580			PSW		
588	LDRFLAGS				PRHOLD
590	TBENT		A*5		A*6
					GET1
598			DSYM		
5A0	JSYM		A*7		
5A8					ALIASNT
5B0	DYNAEND		////////////////////		
5B8	////////////////////		////////////////////		////////////////////
5C0	FCBFIRST		FCENUM		///////// A*8
5C8	////////////////////				LINKLAST
5D0	LINKSTRT				TAXEADDR
5D8	ATSOCPL				LCBSAV
5E0	A*9		A*10		A*11
	A*12		////////////////////		A*13
					A*14
5E8	A*15		A*16		A*17
	/////////				ASYSNAMS
5F0	ACMSSEG				ADMSLIC
5F8	VCFSTLKP				VCFSTLKW
600	AFVS				AOPSECT
608	ADEVTAB				AFSTLKP
610	AGETCLK				AFSTLKW

618	APIE		AIADT
620	AUSER		ARLTK
628	ASCANN		ASSTAT
630	ATABEND		ASUBSECT
638	AOSMODL		AWRTK
640	ASTRINIT		IALT
648	AFREE		AFRET
650	ADMSPIOC		APGMSECT
658	AIOSECT		ADMPEXEC
660	ADIOSECT		AAENSVC
668	ADMSERL		ADMSCRD
670	ADMSFREB		ASVCSECT
678	AADTLKP		AUPUFD
680	ASTATEXT		AOSRET
688	ACMSRET		ASCANO
690	AEXEC		ASTART
698	AADTLKW		AUSABRV
6A0	AEXTSECT		ASCBPTR
6A8	ADMSROS		LMSROS CDMROS
6B0	AACTLKP		AACTNXT
6B8	AACTFREE		AACTFRET
6C0	AADTNXT		ATRKLKP
6C8	ATRKLKPX		AQQTRK
6D0	AQQTRKX		AERASE
6D8	ATYPSRCH		AUPDISK
6E0	AKILLEX		ATFINIS
6E8	ARDBUF		AWRBUF
6F0	AFINIS		ASTATE
6F8	ASTATEW		APCINT

700	CONCCWS	
708		
710	CONINBLK	
718	CONINBUF	
:		
7A0	CMNDLINE	
:		
848	CMNDLIST	
:		
A60	CONSTACK	
:		
BA0	FREESAVE	
:		
BE0	BALRSAVE	
:		
C20	WAITSAVE	
:		
C60	PCTVSAM	
C68	ADIKQLAB	NDIKQLAB
C70	ARURTBL	ADMSVIB
C78	AVIPWORK	A*18
C80	AVSAMSYS	AAHSSYS
C88	AVSREOJ	AVSRWORK
C90	ACBLIST	
C98		

Hexadecimal Displacement	Field Name			Field Description, Contents, Meaning
	<u>Machine Usage</u>			
0	IPLPSW	DS	1D	Initial program load of PSW
8	IPLCCW1	DS	1D	Initial program load of CCW1
10	IPLCCW2	DS	1D	Initial program load of CCW2
		ORG	IPLPSW	
0	RSTNPSW	DS	1D	PSW restart new PSW
8	RSTOPSW	DS	1D	PSW restart old PSW
10	ACMSCVT	DS	1F	Address of simulated OS CVT
14	ASYSREF	DS	1F	Address of nucleus address table
18	EXTOPSW	DS	1D	External old PSW
20	SVCOPSW	DS	1D	Supervisor call old PSW
28	PGMOPSW	DS	1D	Program old PSW
30	MCKOPSW	DS	1D	Machine-check old PSW
38	IOOPSW	DS	1D	Input/output old PSW
40	CSW	DS	1D	Channel status word
48	CAW	DS	1F	Channel address word
4C	NUCRSV1	DS	1F	Reserved for IBM use
50	TIMER	DS	1F	Interval timer
54	NUCRSV2	DS	1F	Reserved for IBM use
58	EXTNPSW	DS	1D	External new PSW
60	SVCNPSW	DS	1D	Supervisor call new PSW
68	PGMNPSW	DS	1D	Program new PSW
70	MCKNPSW	DS	1D	Machine-check new PSW
78	IONPSW	DS	1D	Input/output new PSW
80	CPULOG	DS	48D	Processor logout area
		ORG	CPULOG	
80	NUCRSV3	DS	2D	Reserved for IBM use
90	NUCRSV4	DS	1F	Reserved for IBM use
94	MONCLASS	DS	1H	Monitor call class number
96	PERCODE	DS	1H	Program event recorder code
98	PERADDR	DS	1F	Program event recorder address
9C	MONCODE	DS	1F	MONITOR CALL code
A0	NUCRSV5	DS	4D	Reserved for IBM use
C0	LOWSAVE	DS	XL160	Save area for first 160 bytes of storage
160	FPRLOG	DS	4D	Floating-point register logout area
180	GPRLOG	DS	16F	General-purpose register logout area
1C0	ECRLOG	DS	16F	Extended control register logout area
	<u>System Usage</u>			
200	SYSTEMID	DS	CL32	System name and date
220	INSTALID	DS	CL64	Installation identification
260	SYSNAME	DS	CL8	Name of saved system loaded (via IPL)
268	IPLADDR	DS	1H	Address of device loaded (via IPL)
26A	SYSADDR	DS	1H	Address of system disk
26C	DEVICE	DS	1F	Name of device causing last I/O interrupt
270	NUCRSV6	DS	1F	Reserved for IBM use
274	FEIBM	DC	CL12'FEIBM154067'	FE service number
280	DIAGTIME	DS	CL24	Buffer for DIAGNOSE timer
		ORG	DIAGTIME	
280	CURRDATE	DS	CL8	Current date - mm/dd/yy
288	CURRTIME	DS	CL8	Current time - hh.mm.ss
290	CURRVIRT	DS	1F	Current elapsed virtual time used
294	CURRCPUT	DS	1F	Current elapsed processor time used
298	LASTVIRT	DS	1F	Previous elapsed virtual time used
29C	LASTCPUT	DS	1F	Previous elapsed processor time used
2A0	LASTCMND	DC	CL8' '	Last command issued
2A8	PREVCMD	DC	CL8' '	Next to last command
2B0	LASTEXEC	DC	CL8' '	Last EXEC procedure
2B8	PREVEXEC	DC	CL8' '	Next to last EXEC procedure

Hexadecimal Displacement	Field Name		Field Description, Contents, Meaning	
2C0	LASTLMOD	DC	CL8' '	Last module LOADMOD into main storage
2C8	LASTTMOD	DC	CL8'ACCESS'	Last module LOADMOD into transient area
2D0	DATIPCMS	DC	D'0'	Date (mm/dd/yy) at last IPL CMS
2D8	CLKVALMD	DC	D'0'	Time (STCK form) at midnight (0000 hours)

Macro and Text Library Pointers

2E0	MACDIRC	DC	8A(0)	Address of macro library directories
300	MACLIBL	DC	18F'-1'	Current macro library names
348	TXLIBSV	DC	F'0'	Library save area for TXTLIBS
34C	MACLBSV	DC	F'0'	Library save area for MACLIBS
350	TOTLIBS	DC	F'0'	Total global chains (in bytes)
354	TXTDIRC	DC	A(0)	Address of TEXT library directories
358	TXTLIBS	DC	18F'-1'	Current TEXT library names

Debug Dump Parameters

3A0	DUMPLIST	DS	0D	DEBUG DUMP PLIST
3A0	GRS015	DC	A(GPRLOG)	Address of GPR save area
3A4	LOC0176	DC	A(LWSAVE)	Address of low storage save area
3A8	FIRSTDMP	DC	A(0)	Address of first location to dump
3AC	LASTDMP	DC	A(0)	Address of last location to dump
3B0	FRS06	DC	A(FPRLOG)	Address of FPR save area
3B4	DMPTIT	DC	A(DMPTITLE)	Address of dump title line
3B8		DC	4X'FF'	Reserved for IBM use
3BC	DMPTITLE	DC	CL132' '	Dump title line
440	GLBLTABL	DC	F'0'	Reserved for IBM use
444		DC	H'0'	Used for alignment
446	SVC\$202	SVC	202	Common SVC for reentrant ccde
448	ERR\$202	DC	A(**4)	User will fill if necessary
44C		BR	14	Return to caller
44E		DC	H'0'	Reserved for IBM use

Batch Monitor Information

450	BATFLAGS	DC	1X'00'	A*1	Batch flags
-----	----------	----	--------	-----	-------------

Bits defined in BATFLAGS

BATRUN	EQU	X'80'	Batch monitor running
BATLOAD	EQU	X'40'	Loading batch processor
BATNOEX	EQU	X'20'	Suppress user job execution
BATRERR	EQU	X'10'	Batch reader error
BATCPEX	EQU	X'08'	CP command executing
BATUSEX	EQU	X'04'	User job executing
BATMOVE	EQU	X'02'	MOVEFILE executing from terminal
BATTERM	EQU	X'01'	User job being flushed

451	BATFLAG2	DC	1X'00'	A*2	More batch flags
-----	----------	----	--------	-----	------------------

Bits defined in BATFLAG2

BATXLIM	EQU	X'80'	User job limit exceeded
BATXCPU	EQU	X'40'	Processor time exceeded
BATXPRT	EQU	X'20'	No. of printed lines exceeded
BATXPUN	EQU	X'10'	No. of punched cards exceeded
BATDCMS	EQU	X'08'	Disabled CMS command called
BATIPLSS	EQU	X'04'	Batch loading (via IPL) saved system
BATSTOP	EQU	X'02'	Batch stopping after current job
BATSYSAB	EQU	X'01'	System abnormal termination in process

452		DC	2X'00'	Reserved for IBM use
-----	--	----	--------	----------------------

Batch Processor Entry Points

454	ABATPROC	DC	A(0)	Main entry
458	ABATABND	DC	A(0)	User job abend entry
45C	ABATLIMT	DC	A(0)	User job limits table
460	AUSERST	DC	A(0)	Virtual machine restart entry point
464		DC	2F'0'	Reserved for IBM use

Hexadecimal Displacement	Field Name			Field Description, Contents, Meaning
<u>DOS Library Pointers</u>				
46C	DOSLBSV	DC	F'0'	Library save area for DOSLIBS
470	DOSDIRC	DC	8A(0)	Address of DOS library directories
490	DOSLIBL	DC	18F'-1'	Current DOS library names
4D8	DOSFLAGS	DC	X'00'	A*3 DOS simulation flags
<u>Bits defined in DOSFLAGS</u>				
	DOSMODE	EQU	X'80'	DOS environment flag
	DOSSVC	EQU	X'40'	DOS SVC simulation flag
	DOSVSAM	EQU	X'20'	DOS VSAM running flag
	DOSCOMP	EQU	X'10'	DOS compiler running flag
	DOSPIO	EQU	X'08'	DOS printer indicator
	VSMINSTL	EQU	X'04'	VSAM installation flag to relocate DCSS table
4D9	DOSRC	DC	X'00'	A*4 DOS return code to user
4DA		DC	2X'00'	Reserved for IBM use
4DC	ALTASAVE	DC	V(LTASAVE)	Address of LTA save area
4E0	ABGCOM	DC	V(BGCOM)	Address of partition communication region
4E4	ASYSCOM	DC	V(SYSCOM)	Address of system communication region
4E8	ADOSDCSS	DC	A(0)	Address of IOS DCSS
4EC	SVC12SAV	DC	F'0'	Work area for SVC 12
4F0	DOSFIRST	DC	A(0)	Address of first DOSCB in chain
4F0	DOSNUM	DC	H'0'	Number of DOSCBs in chain
4F6	DOSKPART	DS	H'0'	Number of K-bytes in DOS partition
4F8	APPSAVE	DC	V(PPSAVE)	Address of problem program save area
4FC	DOSTRANS	DC	A(0)	Address of IOS transient area
<u>Free Storage Pointers</u>				
500	MAINLIST	DC	A(0)	Address of first block of user free storage
504	MAINSTRT	DC	V(USERAREA)	Address of the start of user free storage
508	FREELIST	DC	V(NUCEND)	Address of first block of system storage
50C	FREENUM	DC	F'1'	Number of blocks of system storage
510	MAINHIGH	DC	V(USERAREA)	High extend of user free storage
514	FREELOWE	DC	V(NUCEND)	Low extend of system free storage
518	FREELWR	DC	V(TRANSAR)	Lower limit of system free storage
51C	FREUPPR	DC	A(0)	Upper limit of system free storage
520	ANUCEND	DC	V(NUCEND)	Address of end of nucleus storage area
524	AUSRAREA	DC	V(INITSUB)	Address of beginning of user area
528	CURRSAVE	DC	A(0)	Address of current save area
52C	CODE203	DC	H'0'	Code number of last SVC 203
52E	FRERESPG	DS	H'2'	Amount of user storage to reserve for CMS free storage (pages: >=2)
530	ADMSFRT	DC	V(DMSFRT)	DMSFRE work area
534	VCADTLKP	DS	A(DMSLAD)	BALR equivalent of ADTLKP
538	VCADTNXT	DC	A(DMSLADN)	BALR equivalent of ADTNXT
53C	VCADTLKW	DC	A(DMSLADW)	BALR equivalent of ADTLKW
<u>Console I/O Pointers</u>				
540	CURRIOOP	DC	A(0)	Address of current I/O buffer
544	PENDREAD	DC	A(0)	Address of pending read operation
548	PENDWRIT	DC	A(CONSTACK)	Address of pending write operation
54C	PSTFINRD	DC	A(0)	Address of finished read buffer
550	LSTFINRD	DC	A(0)	Address of last finished read buffer
554	AINTRTBL	DC	A(0)	Address of user input translate table
558	AOUTRTBL	DC	A(0)	Address of user output translate table
55C	NUMFINRD	DC	H'0'	Number of finished read buffers
55E	NUMPNDWR	DC	H'0'	Number of pending write operations

NUCON

Hexadecimal Displacement	Field Name			Field Description, Contents, Meaning
<u>Loader Information</u>				
560	VMSIZE	DS	1F	Virtual storage size
564	ALDRTBLS	DC	1F'0'	Address of loader tables
568	STRTADDR	DC	1F'0'	Module starting address
56C	FRSTLOC	DC	1F'0'	Module beginning address
570	LASTLOC	DC	1F'0'	Module ending address
574	LOCCNT	DC	1F'0'	Loader location counter
578	LDRADDR	DC	1F'0'	Loader return address
57C	LDRRTCD	DC	1F'0'	Loader return code
580	PSW	DC	1D'0'	User's starting PSW
588	LDRFLAGS	DC	1F'0'	Loader flags
58C	PRHOLD	DC	1F'0'	Pseudo register counter
590	TBENT	DC	H'0'	Initialize table entries to zeros
592	UNRES	DC	X'00'	A*5 Unresolved reference bit for CMS loader
593	MODFLGS	DC	1X'00'	A*6 Flags
<u>Bits defined in MODFLGS</u>				
	NOMAPFLG	EQU	X'80'	NOMAP flag
	CLEAROP	EQU	X'40'	CLEAR option flag
	MODGNDOS	EQU	X'20'	Module generated with DOS option
	MODGNALL	EQU	X'10'	Module generated with ALL option
	SYSLOAD	EQU	X'08'	Allow load greater than FREELCWE or less than transient
	MDCALL	EQU	X'04'	Indicate module called by DMSMDF
	MOD6	EQU	X'02'	Reserved for IBM use
	MOD7	EQU	X'01'	Reserved for IBM use
594	GET1	DC	1F'0'	DMSLSY R1 save location
598	DSYM	DC	2F'0'	DMSLSY work space
5A0	JSYM	DC	F'0'	DMSLSY unique identifier base
5A4	NXTSYM	DC	C'Z'	A*7 First character of unique identifier
5A5		DC	XL7'0'	Rest of unique identifier
5AC	ALIASENT	DC	1F'0'	Alias entry point (dynamic load)
5B0	DYNAEND	DC	1F'0'	Maximum load location (dynamic load)
5B4		DS	3F	Reserved for IBM use
<u>OS Simulation Pointers</u>				
5C0	FCBTAB	DS	0D	FCB chain anchor
5C0	FCBFIRST	DC	A(0)	Address of first FCB
5C4	FCBNUM	DC	H'0'	Number of FCBs in chain
5C6		DC	X'00'	Reserved for IBM use
5C7	OSSFLAGS	DC	X'00'	A*8 OS simulation flags
<u>Bits defined in OSSFLAGS</u>				
	COMPSWT	EQU	X'80'	Compiler switch
	OSSMNU	EQU	X'40'	DMSMNU unconditional flag
	OSRESET	EQU	X'20'	Reset for OS
	OSWAIT	EQU	X'10'	Wait for OS
	DYLD	EQU	X'08'	Dynamic loading in process
	DYLIBO	EQU	X'04'	Omit dynamic library scan
	DYLIBNOW	EQU	X'02'	Dynamic library scan
	DYMBRNM	EQU	X'01'	Linked via member name
5C8		DC	A(0)	Reserved for IBM use
5CC	LINKLAST	DC	A(0)	Address of last OS linkage block
5D0	LINKSTRT	DC	A(0)	Address of entry point of last module
5D4	TAXEADDR	DC	A(0)	Terminal attention exit element address
5D8	ATSOCPL	DC	V(CPP)	Address of TMP PLIST for TSO programs
5DC	DCBSAV	DC	1F'0'	DCE restoration address

Hexadecimal Displacement	Field Name			Field Description, Contents, Meaning
5E0	OPTFLAGS DC	1X'00'	A*9	Option flags
	<u>Bits defined in OPTFLAGS</u>			
	NOIMPEX EQU	X'80'		No implied EXEC commands
	NOIMPCP EQU	X'40'		No implied CP commands
	NOSTDSYN EQU	X'20'		No standard synonyms
	NOABBREV EQU	X'10'		No command abbreviations
	NOPAGREL EQU	X'08'		No automatic page release
	NOREAD EQU	X'04'		No automatic VM/370 console read
5E1	MISFLAGS DC	1X'00'	A*10	Miscellaneous flags
	<u>Bits defined in MISFLAGS</u>			
	KXSWITCH EQU	X'80'		Halt execution switch
	KOSWITCH EQU	X'40'		Halt tracing switch
	REL PAGES EQU	X'20'		Release pages switch
	GRAFDEV EQU	X'10'		Graphics console
	QSWITCH EQU	X'08'		Quiet switch for console read
	NODDSK EQU	X'04'		Do not access D-disk
	NEGITS EQU	X'02'		Negative return code from DMSITS
	ATTNHIT EQU	X'01'		Attention posted
5E2	MSGFLAGS DC	1X'00'	A*11	Message flags
	<u>Bits defined in MSGFLAGS</u>			
	NOTYPOUT EQU	X'80'		No typing - set by EXEC
	NOTYPING EQU	X'40'		No typing - set by HT
	NORDYMSG EQU	X'20'		No ready message to be typed
	NORDYTIM EQU	X'10'		No time on ready message
	REDERRID EQU	X'08'		Error code to be typed in red
	NOERRMSG EQU	X'04'		No error messages to be typed
	NOERRTXT EQU	X'02'		No text on error messages
	SPECLF EQU	X'01'		Linefeed for typewriter CCW
5E3	DBGFLAGS DC	1X'00'	A*12	DEBUG flags
	<u>Bits defined in DBGFLAGS</u>			
	DBGEXEC EQU	X'80'		DEBUG routine executing
	DBGPGMCK EQU	X'40'		DEBUG entered by a program check
	DBGEXINT EQU	X'20'		DEBUG entered by an external interrupt
	DBGABN EQU	X'10'		DEBUG entered from DMSABN
	DBGNSHR EQU	X'08'		No shared segment present
	DBGSHR EQU	X'04'		Shared segment present
	DBGRECUR EQU	X'02'		Recursion flag
5E4		DC	2X'00'	Reserved for IBM use
5E6	EXECFLAG DC	1X'00'	A*13	EXEC flags
	<u>Bit defined in EXECFLAG</u>			
	EXECCRUN EQU	X'80'		EXEC command running
5E7	PROTFLAG DC	1X'00'	A*14	Storage protection flags
	<u>Bits defined by PROTFLAG</u>			
	PRFPOFF EQU	X'80'		Storage protection is shut off
	PRFTSYS EQU	X'40'		System routine in transient area
	PRFUSYS EQU	X'20'		System routine in user area
5E8	TSOFLAGS DC	1X'00'	A*15	TSO flag byte
	<u>Bit defined in TSOFLAGS</u>			
	TSOATCNL EQU	X'80'		Read canceled by attention

NUCON

Hexadecimal Displacement	Field Name			Field Description, Contents, Meaning
5E9	SUBFLAG	DC	1X'00' A*16	CMS subset flag byte
	<u>Bits defined in SUBFLAG</u>			
	SUBREJ	EQU	X'08'	Subset command reject
	SUBRTN	EQU	X'04'	Subset return
	SUBINIT	EQU	X'02'	Subset initialization
	SUBACT	EQU	X'01'	Subset active
5EA	DCSSFLAG	DC	X'00' A*17	DCSS indicators
	<u>Bits defined in DCSSFLAG</u>			
	DCSSAVAL	EQU	X'80'	CMSSEG segment exists
	DCSSLDED	EQU	X'40'	CMSSEG loaded
	DCSSCPNV	EQU	X'20'	CP or invalid command issued
	DCSSLDS	EQU	X'10'	Loading S-disk
	DCSSVTNA	EQU	X'08'	DMSSVT not available
	DCSSVTL	EQU	X'04'	DMSSVT is loaded
	DCSSOVL	EQU	X'02'	Virtual machine storage overlaid by DCSS
	DCSSJLNS	EQU	X'01'	CMSSEG just loaded nonshared
5EB		DC	X'00'	Reserved for IBM use
5EC	ASYSNAMS	DC	V(SYSNAMES)	
5F0	ACMSSEG	DC	F'0'	Address of CMS saved segment
5F4	ADMSLIO	DC	V(DMSLIO)	
5F8	VCFSTLKP	DC	V(DMSLFS)	EALR equivalent of FSTLKP
5FC	VCFSTLKW	DC	V(DMSLFSW)	EALR equivalent of FSTLKW
	<u>Nucleus Address Table</u>			
600	SYSREF	DS	0D	
600	AFVS	DC	V(FVS)	
604	AOPSECT	DC	V(OPSECT)	
608	ADEVTAB	DC	V(DEVTAB)	
60C	AFSTLKP	DC	V(FSTLKP)	
610	AGETCLK	DC	V(DMSINM)	
614	AFSTLKW	DC	V(FSTLKW)	
618	APIE	DC	V(PIE)	
61C	AIADT	DC	V(IADT)	
620	AUSER	DC	V(USERSECT)	
624	ARDTK	DC	V(DMSDIOR)	
628	ASCANN	DC	V(DMSSCNN)	
62C	ASSTAT	DC	A(0)	
630	ATABEND	DC	V(TABEND)	
634	ASUBSECT	DC	V(SUBSECT)	
638	AOSMODL	DC	A(0)	
63C	AWRTK	DC	V(DMSDIOW)	
640	ASTRINIT	DC	V(DMSSTGST)	
644	IADT	DC	V(ADTSECT)	
648	AFREE	DC	V(FREE)	
64C	AFRET	DC	V(FRET)	
650	ADMSPIOC	DC	V(DMSPIOCC)	
654	APGMSECT	DC	V(PGMSECT)	
658	AIOSECT	DC	V(IOSECT)	
65C	ADMPEXEC	DC	V(DMSDBD)	
660	ADIOSECT	DC	V(DIOSECT)	
664	AABNSVC	DC	V(DMSABNUA)	
668	ADMSERL	DC	V(DMSERL)	
66C	ADMSCRD	DC	V(DMSCRD)	
670	ADMSFREQ	DC	V(DMSFREQ)	
674	ASVCSECT	DC	V(SVCSECT)	
678	AADTLKP	DC	V(ADTLKP)	
67C	AUPUFD	DC	V(DMSAUDUP)	
680	ASTATEXT	DC	A(0)	
684	AOSRET	DC	V(OSRET)	
688	ACMSRET	DC	V(CMSRET)	

Hexadecimal Displacement	Field Name		Field Description, Contents, Meaning
68C	ASCANO	DC	V (DMSSCNO)
690	AEXEC	DC	V (DMSEXC)
694	ASTART	DC	V (DMSLDRA)
698	AADTLKW	DC	V (ADTLKW)
69C	AUSABRV	DC	V (USABRV)
6A0	AEXTSECT	DC	V (EXTSECT)
6A4	ASCBPTR	DC	V (SCBPTR)
6A8	ADMSROS	DC	A (0)
6AC	LDMSROS	DC	H'0'
6AE	CDMSROS	DC	H'0'
6B0	AACTLKP	DC	V (DMSLAF)
6B4	AACTNXT	DC	V (DMSLAFNX)
6B8	AACTFREE	DC	V (DMSLAFPE)
6BC	AACTFRET	DC	V (DMSLAFPT)
6C0	AADTNXT	DC	V (ADTNXT)
6C4	ATRCLKP	DC	V (DMSTRK)
6C8	ATRCLKPX	DC	V (DMSTRKX)
6CC	AQQTRK	DC	V (DMSTCQ)
6D0	AQQTRKX	DC	V (DMSTCQX)
6D4	AERASE	DC	V (DMSERS)
6D8	ATYPSRCH	DC	V (TYPSCRCH)
6DC	AUPDISK	DC	V (DMSAUD)
6E0	AKILLEX	DC	V (KILLEX)
6E4	ATFINIS	DC	V (DMSFNST)
6E8	ARDBUF	DC	V (DMSERD)
6EC	AWRBUF	DC	V (DMSBWR)
6F0	AFINIS	DC	V (DMSFNST)
6F4	ASTATE	DC	V (DMSSTTE)
6F8	ASTATEW	DC	V (DMSSTTW)
6FC	APOINT	DC	V (POINT)
<u>Terminal Buffers</u>			
700	DS	0D	
700	CONCCWS	CCW	0,0,X'60',0 Console read and write CCW
708	CCW	3,0,X'20',1	NOP to get CE and DE together
710	CONINBLK	DC	A (0)
714	DC	XL1'0A'	
715	DC	AL1(134)	
716	CONINBUF	DS	CL134
7A0	DS	0D	
7A0	CMNDLINE	DS	CL160
840	DS	0D	
840	DC	CL8'EXEC'	
848	CMNDLIST	DS	CL536
A60	DS	0D	
A60	CONSTACK	DS	CL320
<u>Save Areas</u>			
BA0	FREESAVE	DS	16F
BE0	BALRSAVE	DS	16F
C20	WAITSAVE	DS	16F
<u>VSAM and AMSERV Control Words</u>			
C60	DS	0D	
<u>Percent of Available User Storage To Reserve for GETVIS/FREEVIS Use When Running VSAM</u>			
C60	PCTVSAM	DC	H'50' 50 percent for CMS/VSAM use
C62	DS	1H	Reserved for IBM use
C64	DS	1F	Reserved for IBM use

NUCON

Hexadecimal Displacement	Field Name	Field Description, Contents, Meaning		
<u>Beginning and End of IKQLAB (when in storage)</u>				
C68	ADIKQLAB DC	A(X'FFFFFF')		Set to A(IKQLAB) when it is in storage
C6C	NDIKQLAB DC	A(0)		Set to end of IKQLAB when in storage
C70	ARURTBL DC	V(RURTBL)		VSAM resource table address
C74	ADMSVIB DC	V(DMSVIB)		Address of VSAM interface bootstrap
C78	AVIPWORK DC	A(0)		Address of IMSVIP work area
C7C	VSAMFLG1 DC	X'00'	A*18	VSAM information flag
<u>Bits defined in VSAMFLG1</u>				
	VSAMRUN EQU	X'80'		VSAM system loaded
	VSJOB CAT EQU	X'40'		VSAM job catalog active
	VIPINIT EQU	X'20'		DMSVIP has been initialized
	VSAMSERV EQU	X'10'		CMSAMS system loaded (AMSERV running)
	VIPSOP EQU	X'08'		OS interface SVC 2 call
	VIPTCLOS EQU	X'04'		OS TCLOSE call
	VSAMSOS EQU	X'02'		OS AMSERV running
C7D		DS	3X	Reserved for IBM use
C80	AVSAMSYS DC	A(0)		Address of VSAM saved system
C84	AAMSSYS DC	A(0)		Address of CMSAMS saved system
C88	AVSREOJ DC	V(\$\$BEOJ4)		DMSVSR entry point from VSAM \$\$EACLOS
C8C	AVSRWORK DC	A(0)		Address of IMSVSR work area
C90	ACBLIST DC	A(0)		ACE list built by OPEN/CLOSE
C94		DS	3F	Reserved for IBM use
CA0		DS	0D	

OPSECT: MAJOR CSECT FOR ALL I/O OPERATION LISTS

OPSECT describes the fields used by several programs as parameter lists for reading and writing on disks and other devices. The OPSECT CSECT is pointed to by the AOPSECT field in NUCON.

0	CMSOP		
8	FILENAME		
10	FILETYPE		
18	FILEMODE	FILEITEM	FILEBUFF
20	FILEBYTE	FILEFORM	FILECOUT
28	FILEREAD	SAVER14	
30	SAVER15	SAVER0	
38	SAVER1	CMSNAME	
40	CMSNAME (cont.)	CONREAD	
48	CONREAD (cont.)	CONRDBUF	
50	A*1	CONRDCNT	////////////////////////////////////
58	WAITLIST		
60	CONWRITE		
68	CONWRBUF	A*2	CONWRCNT
70	WAITLST		
78	WAITDEV		
80		READLST	
88	READLST (cont.)	RIBUFF	
90	RDCCW	RDCOUNT	PUNCHLST
98	PUNCHLST (cont.)	PUNEBUFF	
A0	PUNCOUNT	PRINTLST	
A8	PRINTLST (cont.)	PRBUF	
B0	PRCNT	TAPELIST	
B8	TAPELIST (cont.)	TAPEOPER	
C0	TAPEOPER (cont.)	TAPEDEV	
C8	A*3	TAPEBUFF	TAPESIZE

D0	TAPECOUT		CLOSIO
D8	CLOSIO (cont.)		CLOSIODV
E0	CLOSIODV (cont.)		
E8	////////////////////////////////////		////////////////////////////////////
	////////////////////////////////////		////////////////////////////////////
	////////////////////////////////////		////////////////////////////////////
	////////////////////////////////////		////////////////////////////////////
118	EXLEVEL		EXF1
120	EXNUM		EXADD
128	////////////////////////////////////		////////////////////////////////////
130	FCBIO		A*4

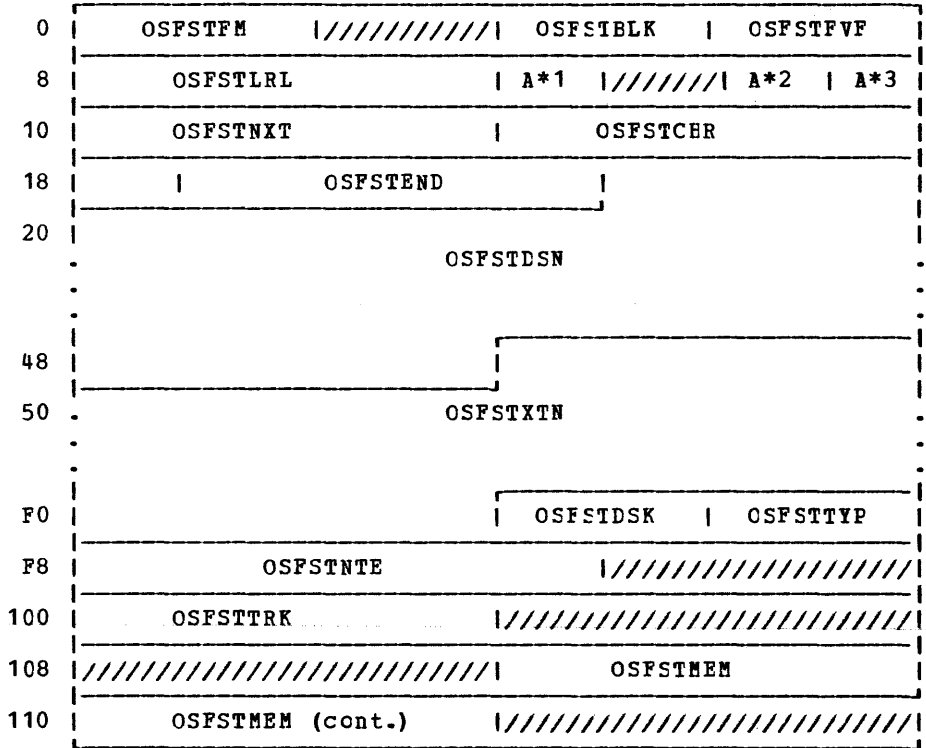
Hexadecimal Displacement	Field Name			Field Description, Contents, Meaning
<u>Main I/O Operation List</u>				
0	PLIST	DS	0D	
0	CMSOP	DS	CL8	I/O operation command word
8	FILENAME	DS	CL8	Filename
10	FILETYPE	DS	CL8	Filetype
18	FILEMODE	DS	CL2	Filemode
1A	FILEITEM	DS	H	Item identification number
1C	FILEBUFF	DS	F	Input/output buffer
20	FILEBYTE	DS	F	Data count
24	FILEFORM	DS	CL2	File format: fixed/variable records
26	FILECOUT	DS	H	Records per block
28	FILEREAD	DS	F	Read data count
	POINTERS	EQU	FILEITEM	
	AFST	EQU	FILEBUFF	
	IOAREA	EQU	FILEBUFF	Buffer area location
	IOLENGTH	EQU	FILEBYTE	Buffer length
<u>Immediate Register Save Area</u>				
2C	SAVER14	DC	F'0'	Temporary R14 save
30	SAVER15	DC	F'0'	Temporary R15 save
34	SAVER0	DC	F'0'	Temporary R0 save
38	SAVER1	DC	F'0'	Temporary R1 save
3C	CMSNAME	DC	CL8'FILE'	Default filename
<u>Console Parameter Lists</u>				
44		DS	0F	
<u>Read Console</u>				
44	CONREAD	DC	CL8'WAITRD'	Terminal read
4C	CONRDBUF	DC	V(CMNDLINE)	Address of input buffer
50	CONRDCOD	DC	C'U' A*1	Translate code
51		DC	X'0'	
52	CONRDCNT	DC	AL2(0)	Data byte count
54		DC	F'0'	Reserved for IBM use
<u>Console Wait List</u>				
58	WAITLIST	DS	0F	
58		DC	CL8'CONWAIT'	

Hexadecimal Displacement	Field Name			Field Description, Contents, Meaning
<u>Write Console</u>				
60	CONWRITE	DS	0F	
60		DC	CL8'TYPLIN'	
68	CONWRBUF	DC	A(0)	Location of message text
6C	CONWRCOD	DC	C'B' A*2	Color code
6D		DC	X'00'	
6E	CONWRCNT	DC	AL2(0)	Length of message text
<u>Wait Parameter List</u>				
70	WAITLST	DS	0F	
70		DC	CL8'WAIT'	Address of IMSCWT
78	WAITDEV	DC	CL4'CON1'	Symbolic address of console
7C		DC	F'0'	
80		DC	F'0'	
<u>Reader Parameter List</u>				
84		DS	0F	
84	READLST	DC	CL8'CARDRD'	
8C	RDBUFF	DC	A(0)	Buffer address
90	RDCCW	DC	H'0'	CCW byte count
92	RDCOUNT	DC	H'0'	Eytes actually read
<u>Card Punch Parameter List</u>				
94	PUNCHLST	DS	0F	
94		DC	CL8'CARDPH'	
9C	PUNBUFF	DC	A(0)	Punch buffer address
A0	PUNCOUNT	DC	A(0)	Punch CCW count
<u>Printer Parameter List</u>				
A4	PRINTLST	DS	0F	
A4		DC	CL8'PRINTR'	
AC	PRBUF	DC	A(0)	Printer buffer address
B0	PRCNT	DC	A(0)	Printer data count
<u>Tape Parameter List</u>				
B4	TAPELIST	DS	0F	
B4		DC	CL8'TAPEIO'	
BC	TAPEOPER	DC	CL8' '	Tape operation command
C4	TAPEDEV	DC	CL4'TAP1'	Tape symbolic device
C8	TAPEMASK	DC	X'00' A*3	Set mode
C9	TAPEBUFF	DC	AL3(0)	Buffer location
CC	TAPESIZE	DC	F'0'	
D0	TAPECOUT	DC	F'0'	Tape counter
<u>Close Out Device Dependent Data Set on Unit Record Equipment</u>				
D4	CLOSIO	DS	0F	
D4		DC	CL8'CLOSIO'	Operation
DC	CLOSIODV	DC	CL8' '	Device type
E4		DC	4X'FF'	
E8		DC	6D'0'	Reserved for IBM use
<u>Storage for EXEC Bootstrap</u>				
118	EXLEVEL	DC	F'0'	EXEC level
11C	EXP1	DC	F'1'	Follows EXEC level
120	EXNUM	DC	F'0'	Number of doublewords of free storage
124	EXADD	DC	F'0'	Address of EXECTOR in storage
128		DC	2F'0'	Reserved for IBM use
<u>Storage for OS Macro Simulation Routines</u>				
130	FCBIO	DC	A(0)	Address of last FCB used during I/O
134	OSIOTYPE	DC	X'DD' A*4	OS access method type

OSFST

OSFST: OS FILE STATUS TABLE

OSFST describes the fields of an OS file status table. When an OS disk is accessed, DMSROS obtains storage from CMS free storage, builds and fills in an OSFST block, which is comparable to a CMS FST block. This block is released by DMSALU.



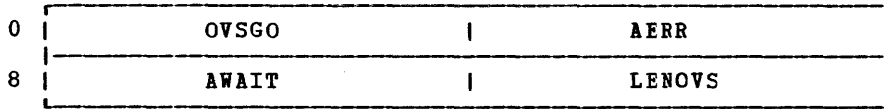
Hexadecimal Displacement	Field Name	Field Description, Contents, Meaning
0	OSFSTFM DS 1H	Disk mode
2	DS 1H	Reserved for IBM use
4	OSFSTBLK DS 1H	Block size
6	OSFSTFVF DS 1H	Fixed/variable flag
8	OSFSTLRL DS 1F	Logical record size
C	OSFSTRFM DS 1X A*1	OS record format
D	DS 1X	Reserved for IBM use
E	OSFSTFLG DS 1X A*2	Flag byte
<u>Bits defined in OSFSTFLG</u>		
	OSFSTALT EQU X'80'	Alternate track indicator
	OSFSTDBK EQU X'40'	Block size not specified in DSCB
	OSFSTMVL EQU X'08'	Multiple volume data set
	OSFSTUMV EQU X'02'	Unmoveable data set
	OSFSTRSW EQU X'01'	Indicates point+1 just issued
F	OSFSTXNO DS 1X A*3	Number of data extents on disk
10	OSFSTNXT DS 1F	Next OS FST
14	OSFSTCHR DS 5X	CCHRR of last I/O operation

Hexadecimal Displacement	Field Name		Field Description, Contents, Meaning
19	OSFSTEND DS	5X	Current extent end
1E	OSFSTDSN DS	44C	Data set name
4A	OSFSTXTN DS	170X	Data extents description
	<u>Bits defined in OSFSTXTN</u>		
	OSFSTEX4 EQU	OSFSTXTN+30	Location of fourth extent from DSCB3
F4	OSFSTDSK DS	1H	Disk address (0cuu)
F6	OSFSTTYP DS	1H	Disk device type (see OSADT for type flags)
F8	OSFSTNTE DS	5X	Used to save CCEHR for NCTE macro
FD	DS	3X	Reserved for IBM use
100	OSFSTTRK DS	1F	No. of tracks per cylinder
104	DS	2F	Reserved for IBM use
10C	OSFSTMEM DS	CL8	Partitioned data set member name
114	DS	1F	Reserved for IBM use
118	DS	0D	Reserved for IBM use
	OSFSTLTH EQU	(*-OSFST)/8	OS FST length in doublewords (X'35')

OVSECT, PCTAB

OVSECT: DESCRIBES THE FIRST FEW LOCATIONS OF DMSOVS

OVSECT is used by module DMSOVS to provide trace information requested by SVCTRACE.



Hexadecimal Displacement	Field Name			Field Description, Contents, Meaning
0	OVSGO	B	*-*	Branch to this point for TRACE
4	AERR	DC	A(0)	Address of IMSERR
8	AWAIT	DC	A(0)	Address of CONWAIT
C	LENOVS	DC	A(0)	Length of DMSOVS doublewords

PCTAB: PROGRAM CHECK OPTION TABLE

PCTAB is used by DOS/VS routines in the event of a program check. The address of PCTAB is in bytes X'64' and X'65' of the partition communication region (BGCCN).



Hexadecimal Displacement	Field Name			Field Description, Contents, Meaning
	PCTAB	EQU	*-8	Program check option table
0	PCROUT	DC	F'0'	PCROUT = 0 when no STXIT macro was issued = address of user program check routine (indicating that the STXIT macro was issued) = complement of the address of the user program check routine (indicating that specified routine in the user STXIT macro was already in use)
4	PCSAVE	DC	F'0'	PCSAVE = 0 when no STXIT macro was issued = address of user save area (indicating that the STXIT macro was issued)

PDSSECT: DIRECTORY TABLE FOR BPAM SIMULATION

PDSSECT describes the fields of the in-storage directory that is used in CS simulation of BPAM. The in-storage directory is built dynamically by DMSSVT from CMS free storage.

0	DIRNAME				DIRPTR
8	A*1	A*2	CORESIZE	PDSBLKSI	A*3 ////////

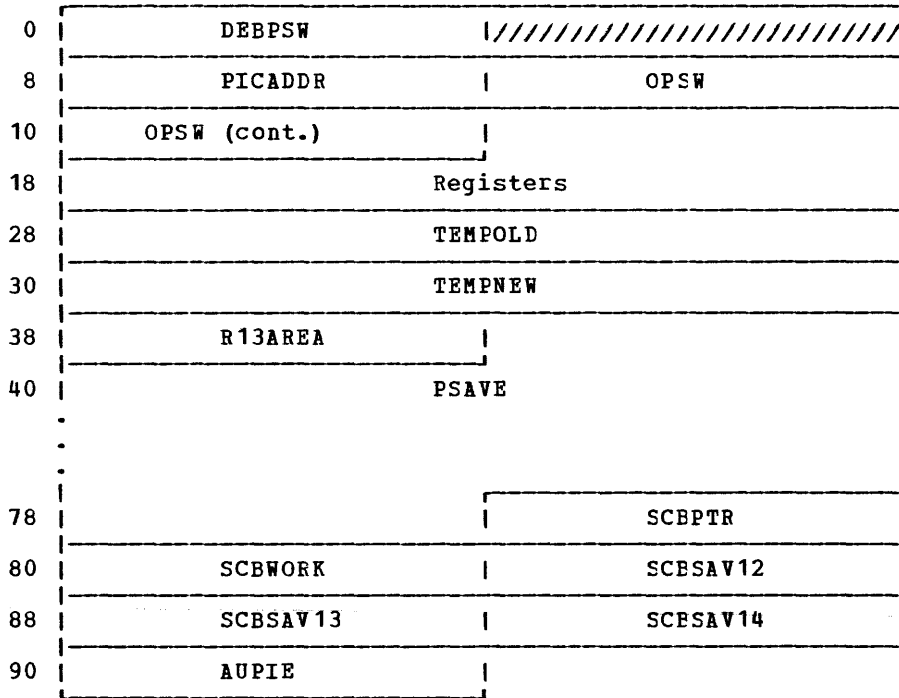
Hexadecimal Displacement	Field Name			Field Description, Contents, Meaning
0	DIRNAME	DS	3H	MACLIP identifier
6	DIRPTR	DS	1H	Item pointer to start of directory
8	TEMPBYTE	DS	1X	A*1 If byte has dollar sign (\$), then PDS is in \$PDSTEMP file
9	NEWBLKS	DS	1X	A*2 No. of new blocks added to EDS by STCW
A	CORESIZE	DS	1H	Size of dictionary in bytes
C	PDSBLKSI	DS	1H	Block size of dictionary
E	CHNGBYTE	DS	1X	A*3 Indicates updates to dictionary
F		DS	1X	Reserved for IBM use
10	PDSDIR	DS	0F	Start of in-storage dictionary

PGMSECT

PGMSECT: PROGRAM INTERRUPT WORK AREA

PGMSECT describes the fields used by DMSITP for saving registers, old PSW, and other data for handling program interrupts.

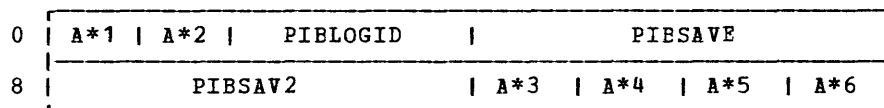
The PGMSECT CSECT is pointed to by the APMSECT field in NUCON.



Hexadecimal Displacement	Field Name	Field Description, Contents, Meaning		
	<u>Storage for Program Interrupt Routine (DMSITP)</u>			
0	DEBPSW DC	F'0',V(DMSDBG)	Point to debug	
4	DC	1F	Reserved for IBM use	
8	PIE DS	0D	Program interrupt element	
8	PICADDR DC	F'0'	PICA address from recent SPIE	
C	OPSW DC	2F'0'	Old PSW after program interrupt	
14	DC	5F'0'	Registers are: R14, R15, R0, R1, and R2	
	*EPIE		End program interrupt element	
28	TEMPOLD DC	8X'00'	Work area	
30	TEMPNEW DC	8X'00'	Work area	
38	R13AREA DC	F'0'	Saved R13	
3C	PSAVE DC	16F'0'	Registers saved at interrupt time	
7C	SCBPTR DC	F'0'	Pointer to first STAE control block	
	<u>Bits defined in SCBPTR</u>			
	STAEBIT EQU	X'80'		
	STAIBIT EQU	X'40'		
	RETRYBIT EQU	X'20'		
80	SCBWORK DC	A(0)	Address of work area for STAE exit routine	
84	SCBSAV12 DC	A(0)	Address of R12 save area for DMSSAB	
88	SCBSAV13 DC	A(0)	Address of R13 save area for DMSSAB	
8C	SCBSAV14 DC	A(0)	Address of R14 save area for DMSSAB	
90	AUPIE DS	A	Address of user's PIE, in SPIE exit	

PIBADR: PROGRAM INFORMATION BLOCK

PIBADR contains a save area address and interrupt information. PIBADR is invoked by the PIBTAB macro and is often referred to by this macro name. The PIBPT field in the BGCCM block points to the PIBTAB CSECT.



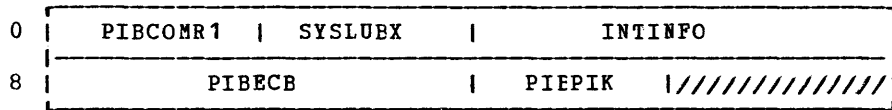
Hexadecimal Displacement	Field Name					Field Description, Contents, Meaning
0	PIBFLG	DS	X	A*1	Flags	
1	PIBCNCL	DS	X	A*2	Cancel code	
2	PIBLOGID	DS	XL2		SYSLOG ID	
4	PIBSAVE	DS	XL4		Address of save area	
<u>Bits defined in PIBSAVE</u>						
	ARFLG	EQU		PIBADR+8	Save area address	
8	PIBSAV2	DS	XL4		Address of system save area	
C	PIBPUBAS	DS	X	A*3	PUP assign flags	
D	PIBLUBID	DS	X	A*4	LUB number of first problem program LUB	
E	PIBLUBNO	DS	C	A*5	Number of LUBs	
F	PIBFLG2	DS	C	A*6	More flags	
<u>ORG PIBADR</u>						
0	PIBCOMRA	DS	XL2		PIB extension DSECT Communications region address	
2	SYSLUBX	DS	XL2		System class LUB address	
4	INTINFO	DS	XL4		Interrupt information	
<u>Bits defined in INTINFO</u>						
	SVCIC	EQU		INTINFO+3	SVC interruption code	
8	PIBECB	DS	XL4		ECB address	
C	PIBPIK	DS	XL2		Program interrupt key	
E		DS	XL2		Reserved for IBM use	

PIB2TAB

PIB2TAB: PROGRAM INFORMATION BLOCK EXTENSION

PIB2TAB describes the entries in the PIB2TAB block, which is an extension of the PIPTAB block. For each PIB table entry, an entry exists in the PIB table extension block (PIB2TAB).

The PIB2PTR field in the BGC0M block points to the PIB2TAB block.



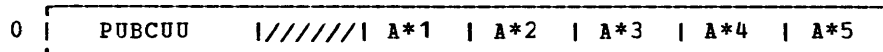
Hexadecimal Displacement	Field Name			Field Description, Contents, Meaning
0	PIBCOMR1	DS	XL2	Address of communications region
2	SYSLUBX	DS	XL2	System LUP index
4	INTINFO	DS	XL4	Used for interruption code
	<u>Bits defined in INTINFO</u>			
	SVCIC	EQU	INTINFO+3	SVC interrupt code
8	PIBECB	DS	XL4	Address of termination ECB, if any
C	PIBPIK	DS	XL2	Program interrupt key
E		DS	XL2	Reserved for IBM use

| PUBADR: PHYSICAL UNIT BLOCK TABLE

| PUBADR defines the fields of a physical unit block table as used by CMS and/or DOS routines. Both DSECTs define the same storage.

| • For Use by CMS Routines (MAPSUB macro)

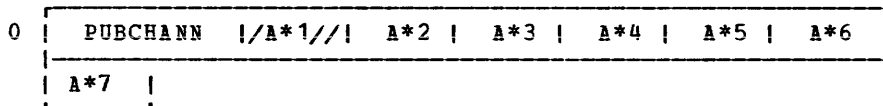
| The simulated PUBADR DSECT has eighteen 8-byte entries, one for each device supported by CMS. The simulated PUBADR DSECT is invoked by the MAPSUB macro.



Hexadecimal Displacement	Field Name	Field Description, Contents, Meaning		
0	PUBCUU DS XL2	Channel and device number		
2	DS X	Reserved for IBM use		
3	PUBDSKM DS X	A*1	Disk mode if assigned DASD	
4	PUBDEVT DS X	A*2	Device type code	
5	PUBTAPM1 DS X	A*3	CMS tape set mode attributes	
6	PUBTAPM2 DS X	A*4	DOS tape set mode attributes	
7	PUBTAP7 DS X	A*5	7-track indicator	

| • For Use by DOS/VS Routines (PUBTAB macro)

| The PUBADR DSECT is invoked by the PUBTAB macro. The address of PUBTAB is at displacement X'40' of BGC0M.

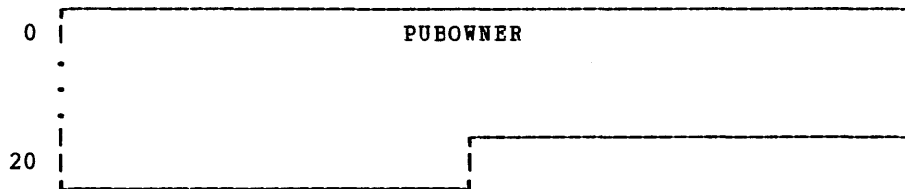


Hexadecimal Displacement	Field Name	Field Description, Contents, Meaning		
0	PUBCHANN DS XL2	Channel and device number		
2	PUBCHQPT DS X	A*1	Reserved for IBM use	
3	PUBERR DS X	A*2	Error retry counter or TEB point	
4	PUBDEVTY DS X	A*3	Device type code	
5	PUBOPTN DS X	A*4	Set mode command or other options	
6	PUBCSFLG DS X	A*5	Channel scheduler flags	
7	PUBJCFLG DS X	A*6	Job control flags	
8	NEXTPUB DS X	A*7	First byte of next PUB entry	
	PUBWIT EQU *-PUBADR	Length of PUB table		
	PUBPTR EQU NEXTPUB	Pointer to original PUB		

PUBOWNER

PUBOWNER: PHYSICAL UNIT BLOCK OWNERSHIP TABLE

PUBOWNER contains a 2-byte entry for each entry in the PUB table. For CMS/DOS, there are eighteen 2-byte entries. The address of the PUBOWNER table is in the SYSCOM block in the DOSCON CSECT of NUCON.



Hexadecimal Displacement	Field Name	Field Description, Contents, Meaning
0	PUBOWNER DS 0H	PUB ownership table
0	DC 18X'0001'	PUB owner

PUBOWNER entries have the following meanings:

Byte	Value	Meaning
0	X'00'	The physical unit is reserved
	X'40'	CMS is waiting for the volume to be mounted
1	X'01'	Background partition owns the physical unit

SSAVE: SYSTEM SAVE AREA

SSAVE is used by DMSITS to save the value of the SVC old PSW, the caller's registers, and other necessary control information required to process the SVC and return to the caller. Since SVC calls can be nested, several of these save areas can exist at one time. The system save area is dynamically allocated in protected free storage. SSAVE is invoked via the CMSAVE macro.

0	A*1 A*2	CODE		CALLER
8		CALLEE		
10		OLDPSW		
18		NRMRET		RRRET
20		EGPR0		EGPR1
28		EGPR2		EGPR3
30		EGPR4		EGPR5
38		EGPR6		EGPR7
40		EGPR8		EGPR9
48		EGPR10		EGPR11
50		EGPR12		EGPR13
58		EGPR14		EGPR15
60		EFPR0		
68		EFPR2		
70		EFPR4		
78		EFPR6		
80		CHKWRD1		SSAVENXT
88		SSAVEPRV		USAVEPTR
90		OSTEMP	A*3	KEYS
98		KEYS (cont.)		XGPR0
A0		XGPR1		XGPR15
A8		XCOUNT		CHKWRD2

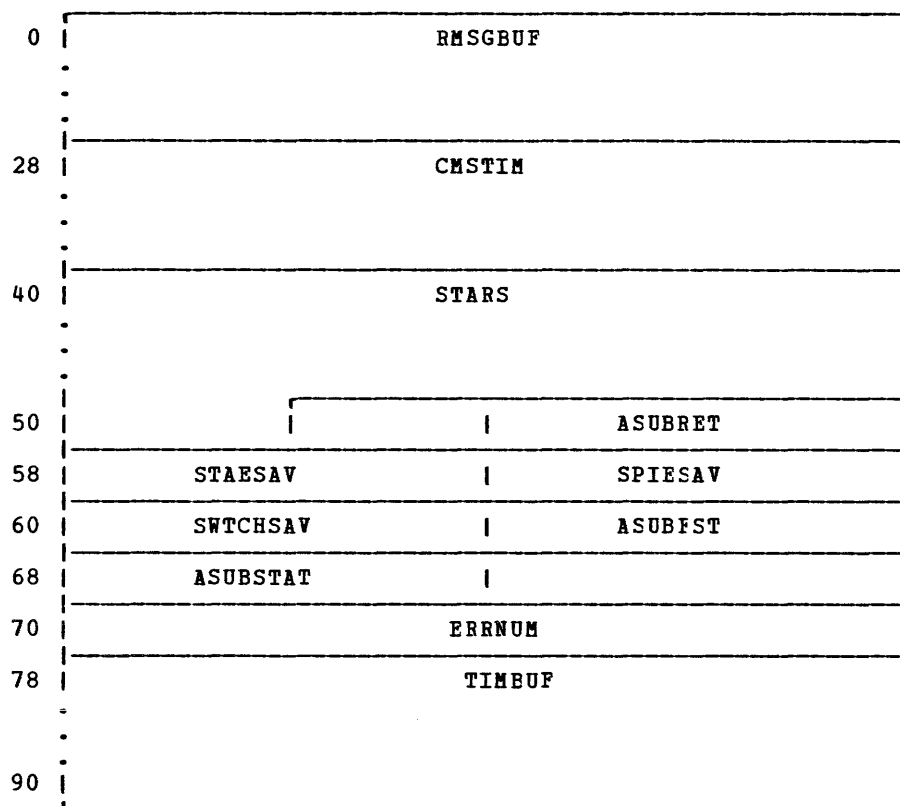
SSAVE

Hexadecimal Displacement	Field Name				Field Description, Contents, Meaning
0	OVIND	DS	X	A*1	Override indicator: 0, 1, 2, and 3
	TYPFLAG	DS	BL1	A*2	SVC-type flag byte
<u>Bits defined in TYPFLAG</u>					
	TPFERT	EQU	X'80'		Error return desired
	TPFNS	EQU	X'40'		No save area allocated
	TPFR01	EQU	X'20'		Return callee's R0-R1 to caller
	TPFUSR	EQU	X'10'		User SVC call
	TPFACB	EQU	X'08'		OS VSAM SVC request
	TPFSV3	EQU	X'02'		SVC 203
	TPFSVO	EQU	X'01'		OS simulation SVC
2	CODE	DS	H		SVC 203 code value
4	CALLER	DS	A		Address of SVC caller
8	CALLEE	DS	D		Name of routine being called
10	OLDPSW	DS	D		SVC old PSW of caller
18	NRMRET	DS	A		Address for normal return
1C	ERRET	DS	A		Address for error return
20	EGPRS	DS	OF		General-purpose registers at entry to SVC
20	EGPR0	DS	F		R0
24	EGPR1	DS	F		R1
28	EGPR2	DS	F		R2
2C	EGPR3	DS	F		R3
30	EGPR4	DS	F		R4
34	EGPR5	DS	F		R5
38	EGPR6	DS	F		R6
3C	EGPR7	DS	F		R7
40	EGPR8	DS	F		R8
44	EGPR9	DS	F		R9
48	EGPR10	DS	F		R10
4C	EGPR11	DS	F		R11
50	EGPR12	DS	F		R12
54	EGPR13	DS	F		R13
58	EGPR14	DS	F		R14
5C	EGPR15	DS	F		R15
60	EFPRS	DS	OD		Floating-point registers at entry
60	EFPR0	DS	D		FPR0
68	EFPR2	DS	D		FPR2
70	EFPR4	DS	D		FPR4
78	EFPR6	DS	D		FPR6
80	CHKWRD1	DC	C'ABCD'		Check word one
84	SSAVENXT	DS	A		Address of next SSAVE area
88	SSAVEPRV	DS	A		Address of previous SSAVE area
8C	USAVEPTR	DS	A		Address of corresponding user save area - see USAVE DSECT
90	OSTEMP	DS	F		Temporary work area for OS simulation routines
<u>DMSKEY Key Stack</u>					
	KEYMAX	EQU	7		Maximum number of keys in stack
94	KEYP	DC	X'00'	A*3	Number of keys in stack
95	KEYS	DS	(KEYMAX)X		Key stack
<u>The Following Fields Are Filled in Only by DMSOVS, the SVCTRACE Subroutine</u>					
9C	XGPR0	DS	F		Extra copy of EGPR0
A0	XGPR1	DS	F		Extra copy of EGPR1
A4	XGPR15	DS	F		Extra copy of EGPR15
A8	XCOUNT	DS	F		Extra copy of SVCOUNT
AC	CHKWRD2	DC	C'EFGH'		Check word two
	SSAVESZ	EQU	(*-SSAVE+7)/8		Size of system save area in doublewords (X'15')

SUBSECT: SUBSET WORK AREA

SUBSECT defines the fields in the SUBSET work area which is used by CMS SUBSET command processing and abend recovery.

The SUBSECT block is pointed to by the ASUBSECT field in NUCON.



Hexadecimal Displacement	Field Name	Field Description, Contents, Meaning		
0	RMSGBUF	DS	10F	Buffer area for formatting time message
28	CMSTIM	DS	0F	PLIST to obtain time from DMSINM
28		DC	CL8'CMSTIME'	
30		DS	F	Virtual machine time used
34		DS	F	Total processor time used (CP+CMS)
38		DS	A	Address of time buffer
3C		DS	F	Message length (filled in by CMSTIME)
40		DS	0F	PLIST for ACTLKP (used by SUBSET)
40	STARS	DC	CL8'**'	
48		DC	CL8'**'	
50		DC	CL2'**'	
54		DS	0F	Subset address for storage areas
54	ASUBRET	DS	F	Return address to caller
58	STAESAV	DS	F	STAF save area
5C	SPIESAV	DS	F	SPIE save area
60	SWTCHSAV	DS	F	
64	ASUBFST	DS	F	
68	ASUBSTAT	DS	F	
70	ERRNUM	DS	D	Work area for error return code
78	TIMBUF	DS	4D	Work area for DMSINM to store date and time

SVCSECT

SVCSECT: SVC INTERRUPT STORAGE

SVCSECT describes the fields used by DMSITS in handling SVC interrupts. An SVCSECT block is built dynamically when an SVC is issued. The first SVCSECT is pointed to by the ASVCSECT field in NUCON; if SVCs are nested, the chain of SVCSECT blocks is processed using the CURRALOC and LASTALOC fields.

0	JNUMB		JFIRST
8	JF4		JLAST
10	A*1 ////////	SVCAB	CURRALOC
18	LASTALOC		DEPTH
20	ADMSOVS		OVEPF OVBTf
28	OVAPF		OVATF A*2 //////////
30	SVCSAVE		
78	NRMSAV		
158	//////////		SVCOUNT
160	SVCSTOP		SVLAD
168	SVLADW		SVLFS
170	LOADLIST		
178	LOADNAME		
180	(Literals are loaded into this area)		
188	(The literals here are from the origin)		
190	LOADSTRT		
198	(Hexadecimals are entered in this area)		
1A0	//////////\//////////		
1A8	MODLIST		
1B0	DUMCOM		
1B8	A*3	ZERO3	TRANSRT
1C0	TRANSRT (cont.)		A*4 ATRANS
1C8	TEMPO2		
1D0	//////////\//////////		
1D8	//////////\//////////		
1E0	RGPRO		RGPR1

1E8	RGPR2		RGPR3
1F0	RGPR4		RGPR5
1F8	RGPR6		RGPR7
200	RGPR8		RGPR9
208	RGPR10		RGPR11
210	RGPR12		RGPR13
218	RGPR14		RGPR15
220	RFPR0		
228	RFPR2		
230	RFPR4		
238	RFPR6		
240	NRMUSAV		
.	.		
.	.		
.	.		

Hexadecimal Displacement	Field Name			Field Description, Contents, Meaning
0	USVCTBL	DS	OF	User SVC table
	<u>Keep Next Four in Order</u>			
0	JNUMB	DC	F'0'	No. of doublewords in SVC number table
4	JFIRST	DC	A(*-*)	Address of first item (if any) in table
8	JP4	DC	F'4'	Loop increment for BXLE
C	JLAST	DC	A(*-*)	Address of last item in table
	<u>Start-up Flags -- Indicate System Parameter Flag for the Called Routine</u>			
10	SFLAG	DC	BL1'0' A*1	Flag byte
	<u>Bits defined in SFLAG</u>			
	SFSYS	EQU	X'80'	System flag -- SVC protect key is zero
	SFTRN	EQU	X'40'	Transient area routine -- system mask is off
	SFNUC	EQU	X'20'	Nucleus routine -- system mask is off
	SPREN	EQU	X'01'	Invalid reentry flag
11		DC	X'00'	Reserved for IBM use
12	SVCAB	DC	H'0'	SVC abend code if any
14	CURRALOC	DC	A(0)	Current allocated save area
18	LASTALOC	DC	A(0)	Last allocated save area
1C	DEPTH	DC	F'0'	Nested SVC depth
	<u>Information for SYCTRACE</u>			
20	ADMSOVS	DC	A(0)	Address of DMSOVS
24	OVBPFF	DC	BL2'0"	"Before print" flags
26	OVBTFF	DC	BL2'0"	"Before type" flags
28	OVAPFF	DC	BL2'0"	"After print" flags
2A	OVATFF	DC	BL2'0"	"After type" flags

SVCSECT

Hexadecimal Displacement	Field Name	Field Description, Contents, Meaning		
<u>The Following Equate Symbols Are Associated with the First Byte of the OVBFPP and OVAFP Flag Fields Defined Above</u>				
	OVF10N	EQU	X'80'	Current option is set on
	OVF1GB	EQU	X'40'	GPRs before call wanted
	OVF1GA	EQU	X'20'	GPRs after call wanted
	OVF1GS	EQU	X'10'	GPRs returned from SVC callee
	OVF1PA	EQU	X'08'	PLISI wanted
	OVF1F	EQU	X'04'	Floating-point registers wanted
	OVF1FS	EQU	X'01'	Floating-point registers returned from SVC callee

The Following Equate Symbols Are Associated with the Second Byte in the OVBFPP and OVAFP Flag Fields

	OVF2ST	EQU	X'80'	STOP wanted
	OVF2CM	EQU	X'40'	CMS SVC TRACE wanted
	OVF2NR	EQU	X'20'	Normal return CMS SVCs wanted
	OVF2OS	EQU	X'10'	OS SVCs wanted
	OVF2WA	EQU	X'08'	WAIT CMS SVCs wanted

2C OVSTAT DC B'0' A*2 Current status of SVCTRACE

Bits defined in OVSTAT

	OVSON	EQU	X'80'	Override options are on
	OVSPREV	EQU	X'40'	SVCTRACE SAME is valid
	OVSAFT	EQU	X'20'	After bit; set by DMSITS
	OVSHO	EQU	X'10'	HALT override flag
	OVSSO	EQU	X'08'	SUSPEND override flag

2D SVCSAVE DC XL3'0' Reserved for IBM use
 30 NRMSAV DC 18F'0' INTSVC work area
 78 DC 28D'0' Normal standard information

158 DC F'0' Reserved for IBM use
 15C SVCOUNT DC F'0' Current SVC count
 160 SVCSTOP DC F'0' For DMSITS debugging
 164 SVLAD DS F Save R14 for DMSLAD
 168 SVLADW DS F Save R14 for DMSLADW
 16C SVLFS DS F Save R14 for DMSLFS
 170 LOADLIST DC CL8'LOAD'
 178 LOADNAME DC CL8' '
 180 DC CL8') '
 188 DC CL8'ORIGIN'
 190 LOADSTRT DC CL8'0'
 198 DC 8X'FF'
 1A0 DC 2F'0' Reserved for IBM use

PLIST for Calling DMSLDR (Keep next five in strict sequence)

1A8 DS OF
 1A8 MODLIST DC CL8'LOADMOD ' Routine name
 1B0 DUMCOM DC CL8' ' Module filename filled in here
 1B8 SSMON DC X'FF' A*3 Fence; allows all interrupts
 1B9 ZERO3 DC AL3(0) Address is 3-bytes long and set to zeros
 1BC TRANSRT DC CL8' ' Holds filename of transient routine

Hexadecimal Displacement	Field Name			Field Description, Contents, Meaning
1C4	TRANMSK	DC	AL1(X'FF') A*4	
1C5	ADTRANS	DC	VL3(TRANSAR)	
1C8	TEMP02	DC	D'0'	For CVD and other scratch use
1D0		DC	4F'0'	Reserved for IBM use
1E0		DS	0D	
1E0	RGPRS	DS	0F	Returned general-purpose registers
1E0	RGPR0	DS	F	R0
1E4	RGPR1	DS	F	R1
1E8	RGPR2	DS	F	R2
1EC	RGPR3	DS	F	R3
1F0	RGPR4	DS	F	R4
1F4	RGPR5	DS	F	R5
1F8	RGPR6	DS	F	R6
1FC	RGPR7	DS	F	R7
200	RGPR8	DS	F	R8
204	RGPR9	DS	F	R9
208	RGPR10	DS	F	R10
20C	RGPR11	DS	F	R11
210	RGPR12	DS	F	R12
214	RGPR13	DS	F	R13
218	RGPR14	DS	F	R14
21C	RGPR15	DS	F	R15
220	RFPRS	DS	0D	Returned floating-point registers
220	RFPR0	DS	D	FPR0
228	RFPR2	DS	D	FPR2
230	RFPR4	DS	D	FPR4
238	RFPR6	DS	D	FPR6
240	NRMSAV	DS	12D	NRMSAV user save area

SVEARA

SVEARA: LTA AND PP SAVE AREA DSECT

SVEARA describes the fields in a DOS/VS Logical Transient Area (LTA) and Problem Program (PP) save area. SVEARA is invoked via the DOSAVE macro. These areas are used by DOS/VS routines to save the value of the PSW and registers for purposes such as linkage to and from transient routines.

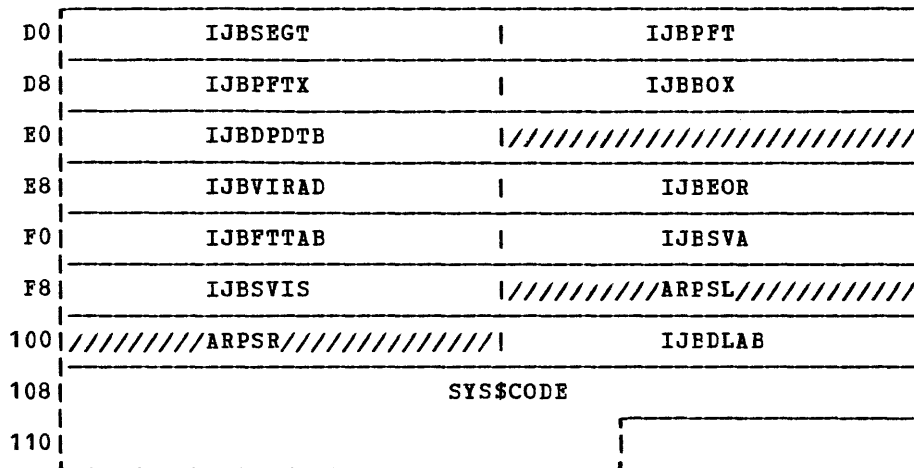
0	////////////////////////////////////\////////////////////////////////////	
8	SVEPSW	SVEPSW2
10	SVER09	SVER0A
18	SVER0B	SVER0C
20	SVER0D	SVER0E
28	SVER0F	SVER00
30	SVER01	SVER02
38	SVER03	SVER04
40	SVER05	SVER06
48	SVER07	SVER08

Hexadecimal Displacement	Field Name			Field Description, Contents, Meaning
0		DS	2F	Reserved for IBM use
8	SVEPSW	DS	F	First half PSW
C	SVEPSW2	DS	F	Second half PSW
10	SVER09	DS	F	Save area for register 9
14	SVER0A	DS	F	Save area for register 10
18	SVER0B	DS	F	Save area for register 11
1C	SVER0C	DS	F	Save area for register 12
20	SVER0D	DS	F	Save area for register 13
24	SVER0E	DS	F	Save area for register 14
28	SVER0F	DS	F	Save area for register 15
2C	SVER00	DS	F	Save area for register 0
30	SVER01	DS	F	Save area for register 1
34	SVER02	DS	F	Save area for register 2
38	SVER03	DS	F	Save area for register 3
3C	SVER04	DS	F	Save area for register 4
40	SVER05	DS	F	Save area for register 5
44	SVER06	DS	F	Save area for register 6
48	SVER07	DS	F	Save area for register 7
4C	SVER08	DS	F	Save area for register 8

SYSCOM: SYSTEM COMMUNICATION REGION

SYSCOM describes the fields in the SYSCOM block which is the CMS/DOS equivalent of the DOS/VS System Communication Region (SYSCOM). The ASYSCOM field in NUCCN points to the SYSCOM block in DMSNUC.

0	IJBERBLC		IJEAREX
8	IJBERR19 IJBERR24		IJEPUBRS
10	IJBFETCH		IJEINTRT
18	IJBEXTRT		IJELTA
20	IJBPPBEG		IJECHANQ
28	IJBQSIZE IJBQLNG		IJENPART ////////////////
30	IJBRSAVE		IJECCNSP
38	IJBSAB		IJECENTB
40	A*1 A*2 A*3 A*4		IJESTID IJBEXIT
48	IJBPDADR		IJETKHLD
50	IJBTIMER		IJEAFETAB
58	IJBBLIK IJBTIK		IJEPWR
60	IJBTCAVT		IJERFTAB
68	IJBEUECB		IJEPOLTEP
70	IJBRASLN		IJETRTAB
78	IJBPBOWN		IJEJATAB
80	IJBPMGR		IJECCWT
88	IJBHAVSD		IJELNSTB
90	IJBAMCOM		IJEAPTA
98	IJBEBLK0		IJESBLKX
A0	A*5 A*6 A*7 A*8 A*9 A*10		////////////////
A8	////////////////		////////////////
B0	////////////////		IJEMVCAD
B8	////////////////		////////////////
C0	IJBMFCEP		
C8	A*11		IJEPUBLN IJBAPNO



Hexadecimal Displacement	Field Name			Field Description, Contents, Meaning
0	IJBERBLC	DC	A(0)	Address of error block
4	IJBAREX	DC	A(0)	Exit address for attention
8	IJBERR19	DC	H'0'	Cancel exit for attention
A	IJBERR24	DC	H'0'	Cancel exit for attention
C	IJBPUBRS	DC	F'0'	SYSRIS PUP address
10	IJBFETCH	DC	A(0)	Address of fetch routine
14	IJBINTRT	DC	A(0)	Address of I/O interrupt routine
18	IJBEXTRT	DC	A(0)	Address of external interrupt routine
1C	IJBLLTA	DC	A(0)	Pointer to logical transient area
20	IJBPPBEG	DC	A(0)	Pointer to problem program area
	IJBFLPTR	EQU	*	Free list pointer
24	IJBCHANQ	DC	F'0'	Pointer to channel queue
28	IJBQSIZE	DC	H'0'	Number of channel queue entries
2A	IJBQLNG	DC	H'0'	Length of one error queue entry
2C	IJBNPART	DC	H'1'	Number of partitions
2E		DC	H'0'	Reserved for IBM use
30	IJBRSAVE	DC	A(0)	Pointer to channel buckets
34	IJBCONSP	DC	A(0)	Address of CRT table
38	IJB SAB	DC	A(0)	Address of SAB table
3C	IJBCHNTB	DC	A(0)	Address of channel control table
40	IJBFLG01	DC	X'00'	A*1 Flags and switches
41	IJBFLG02	DC	X'00'	A*2 Switch byte
42	IJBFLG03	DC	X'00'	A*3 Flags and switches
43	IJBFLG04	DC	X'00'	A*4 Flags and switches
44	IJBSTID	DC	H'0'	System task selection control field
	IJBSELCT	EQU	*-1	System task selection byte
46	IJBEXIT	DC	H'0'	Pointer to task selection
48	IJBPDADR	DC	A(0)	Pointer to PDAREA
	IJBTHPTR	EQU	*-1	Track hold FLPTR
4C	IJBTKHLD	DC	F'0'	Address of track hold table

Hexadecimal Displacement	Field Name			Field Description, Contents, Meaning
50	IJBTIMER	DC	A(0)	Address of timer request table
54	IJBABTAB	DC	A(0)	Address of AB option table
58	IJBBLIK	DC	H'0'	Key of task owning the LTA
5A	IJBTIK	DC	X'0010'	Task interrupt key
5C	IJBPWR	DC	A(0)	Pointer to power table
60	IJBTCVAVT	DC	A(0)	Space for VIAM address
64	IJBRFTAB	DC	A(0)	Pointer to RF table
68	IJBUEUCB	DC	A(0)	Pointer to EU and ECB table
6C	IJBOLTEP	DC	A(0)	Address of CLTEP bucket
70	IJBASLN	DC	A(0)	Pointer to RAS linkage area
74	IJBTRTAB	DC	A(0)	Address of ASCII table
78	IJBPBOWN	DC	A(0)	Address of PUB ownership table
7C	IJBJATAB	DC	A(0)	Address of job accounting common area
80	IJBPMGR	DC	A(0)	Base address of programmer routines
84	IJBCCWT	DC	A(0)	Address of CCW transient work area
88	IJBASVSD	DC	A(0)	Pointer to SDAID common area
8C	IJBLNSTB	DC	A(0)	Address of line mode table
90	IJBAMCOM	DC	A(0)	Address of VSAM common register
94	IJBAPTA	DC	A(0)	Address of PTA
98	IJBSDLKO	DC	A(0)	Pointer to first system task block
9C	IJBSDLKX	DC	F'0'	Pointer to current system task
A0	IJBYSPT	DC	X'00'	A*5 For alignment
A1	IJBASPT	DC	AL1(0)	A*6 Pointer to RAS task block
A2	IJBPMRPT	DC	AL1(0)	A*7 Pointer for PMGR task block
A3	IJBSPPT	DC	AL1(0)	A*8 Pointer to SPVR task block
A4	IJB CRTPT	DC	AL1(0)	A*9 Pointer to CRT task block
A5	IJB ERPPT	DC	AL1(0)	A*10 Pointer to ERP task block
A6		DC	10X'00'	Reserved for IBM use
B0		DC	F'0'	Reserved for IBM use
B4	IJBMV CAD	DC	A(0)	Pointer to MVCFLD
B8		DC	F'0'	Reserved for IBM use
BC		DC	H'0'	Reserved for IBM use
BE		DC	H'0'	Reserved for IBM use
C0	IJB MF CER	DS	11X'00'	Information on MFCM and MFCU ERP
CB	IJBNERQ	DC	AL1(0)	A*11 Number of error queue entries
CC	IJB PUBLN	DC	S(0)	Length of PUB table
CE	IJBAPNO	DC	H'1'	Number of active partitions
D0	IJBSEGT	DC	A(0)	Address of segment table
D4	IJB PFT	DC	A(0)	Address of page frame table
D8	IJB PFTX	DC	A(0)	Pointer to page frame table extension
DC	IJBBOX	DC	A(0)	Pointer to boundary box
E0	IJB DPDTB	DC	A(0)	Pointer to LPD table
E4		DC	F'0'	Reserved for IBM use
E8	IJB VIRAD	DC	A(0)	Address of VIRTAD routine
EC	IJB EOR	DC	F'0'	End of real storage
F0	IJBFTTAB	DC	A(0)	Address of the fetch table
F4	IJB SVA	DC	A(0)	Address of the SVA start
F8	IJB SVIS	DC	A(0)	Address of SVA GETVIS area
FC	ARPSL	DC	A(0)	Reserved for IBM use
100	ARPSR	DC	A(0)	Reserved for IBM use
104	IJB DLAB	DC	A(SYS\$CODE)	Pointer to system code name
108	SYS\$CODE	DC	CL13'CMS/VSAM'	System code name

SYSNAMES

SYSNAMES: SAVED SYSTEMS NAMES

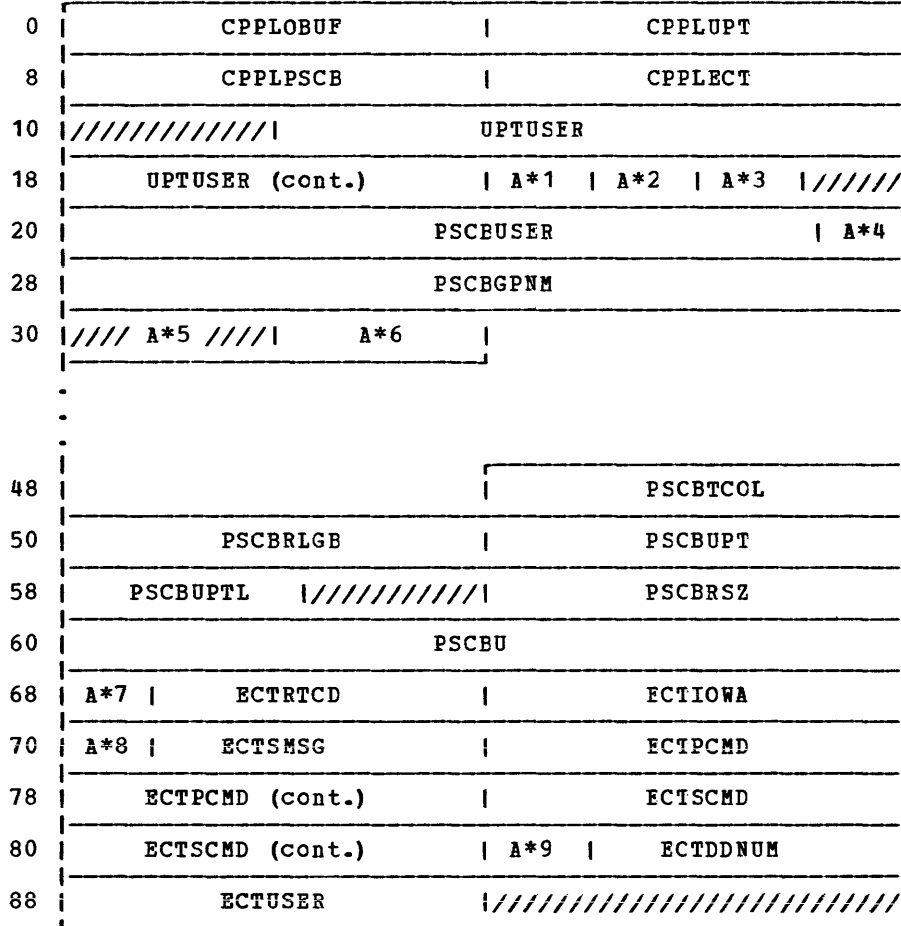
SYSNAMES defines the names of any saved systems which may be loaded by CMS routines. SYSNAMES describes the entries in the SYSNAMES table which is pointed to by the ASYSNAMES field in NUCON.

0	CMSSEG
8	CMSVSAM
10	CMSAMS
18	CMSDOS

Hexadecimal Displacement	Field Name		Field Description, Contents, Meaning
0	CMSSEG	DC	CL8'CMSSEG' CMS shared system name
8	CMSVSAM	DC	CL8'CMSVSAM' VSAM shared system name
10	CMSAMS	DC	CL8'CMSAMS' Access Method Services shared system name
18	CMSDOS	DC	CL8'CMSDOS' DOS shared system name
20	SYSNEND	DS	0D
	SYSNCNT	EQU	(SYSNEND-SYSNAMES)/8 Size in doublewords (X'04')

TSOBLKS: TSO CONTROL BLOCKS

TSOBLKS DSECT describes the entries in the TSOBLKS block which contains OS control information used by CMS, that is, the command program parameters list (CPPL), user profile table (UPT), protected step control block (PSCB), and the environment control table (ECT). The ATSOCPPL field in NUCON points to TSOBLKS block.



Hexadecimal Displacement	Field Name	Field Description, Contents, Meaning		
0	CPPL DS OF	Temporary PLIST to CP programs		
0	CPPLOBUF DC AL4(0)	Address of command line		
4	CPPLUPT DC AL4(UPT)	Address of dummy UPT		
8	CPPLPSCB DC AL4(PSCB)	Address of dummy PSCB		
C	CPPLECT DC AL4(ECT)	Address of dummy ECT		
<u>User Profile Table (UPT)</u>				
10	UPT DS OF	Reserved for IBM use		
10	DS CL2	Reserved for installation use		
12	UPTUSER DS CL10	User's environment switch		
1C	UPTSWS DC X'00'	A*1		
<u>Bits Defined in UPTSWS</u>				
	EQU X'80'	Reserved for IBM use		
	UPTNPRM EQU X'40'	No prompting is to be done		
	UPTMID EQU X'20'	Print message identifiers		
	UPTNCOM EQU X'10'	No user communication allowed via SEND		

TSOBLKS

Hexadecimal Displacement	Field Name	Field Description, Contents, Meaning		
<u>Bits Defined in UPTSWS (cont.)</u>				
	UPTPAUS EQU	X'08'		Pause for "?" when in noninterface mode
	UPTALD EQU	X'04'		ATTN has been specified as line delete
1D	UPTCDEL DC	X'00'	A*2	Character delete character
1E	UPTDEL DC	X'00'	A*3	Line delete character
1F	DS	CL1		Reserved for IBM use
<u>Protected Step Control Block (PSCB)</u>				
20	PSCB DS	0F		
20	PSCBUSER DC	CL7' '		User ID padded with blanks
27	PSCBUSRL DC	X'00'	A*4	Length of user ID
28	PSCBGPNM DS	CL8		Esoteric group name initialized by logon
30	PSCBATR1 DS	X	A*5	16-bit string of user attributes; bits 3-15 reserved for IBM use
	ORG	PSCBATR1		
	PSCBCTRL EQU	X'80'		Operator command user
	PSCBACCT EQU	X'40'		Account command user
	PSCBJCL EQU	X'20'		Submit command user
	DS	X		Reserved for IBM use
32	PSCBATR2 DS	X	A*6	16-bit string reserved for installation use
33	DS	X		
34	DC	6F'0'		Six fullwords used for TSC accounting; initialized to 0
4C	PSCBTCOL DS	1F		
50	PSCBRLGB DS	AL4(0)		
54	PSCBUPT DC	AL4(UPT)		Pointer to the user profile table
58	PSCBUPTL DC	AL2(16)		Length of the user profile table
5A	DS	BL.16		Reserved for IBM use
5C	PSCBRSZ DS	A		Region size requested in 2K units
60	PSCBU DS	CL8		Reserved for installation use
<u>Environment Control Table (ECT)</u>				
68	ECT DS	0F		
68	ECTRCDF DC	AL1(0)	A*7	High-order bit indicates CP abended
69	ECTRTCD DC	AL3(0)		Return code from last control routine
6C	ECTIOWA DC	AL4(0)		Address of I/O service routine wrk area
70	ECTMSGF DC	X'00'	A*8	High-order bit means delete second level messages (Must be initialized by user at start of user's program)
71	ECTSMSG DS	AL3		Address of second level message chain
74	ECTPCMD DC	CL8' '		Primary command name
7C	ECTSCMD DS	CL8		Subcommand name
84	ECTSWS DC	X'00'	A*9	ECT switch
<u>Bits defined in ECTSWS</u>				
	ECTNOPD EQU	X'80'		If 0 bit is on, no operands exist in command buffer
	ECTATRM EQU	X'20'		CP terminated by TMP DETACH with STAE
	* EQU	X'40'		Reserved for IBM use
	ECTLOGF EQU	X'10'		Logon/off requested TMP to log off
	ECTNHML EQU	X'08'		No user messages received at logon
	ECTNNOT EQU	X'04'		No broadcast notices to be received
85	ECTDDNUM DC	AL3(0)		Counter for generated temporary DDNAMS
88	ECTUSER DS	A		Reserved for installation use
8C	DS	A		Reserved for IBM use

| Note: For terminal attention exit element (TAXE), see the CMSTAXE DSECT.

USAVE: USER SAVE AREA

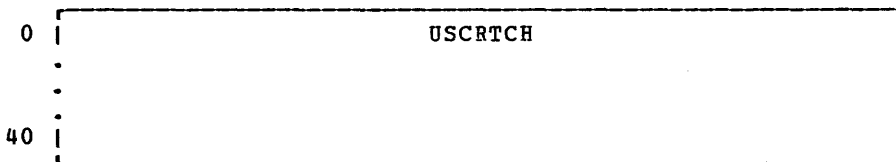
USAVE is used by DMSITS to allocate and free save areas for other routines during SVC processing; it is pointed to by the USAVEPTR field in SSAVE. USAVE is invoked via the SVCSAVE macro.



Hexadecimal Displacement	Field Name	Field Description, Contents, Meaning
0	DS 12D	Scratch area passed to user via a pointer in R13
	USAVESZ EQU (*-USAVE)/8	Size of user save area in doublewords (X'0C')

USERSECT: USER WORK AREA

USERSECT describes the USERSECT block which is a general scratch storage area provided for user-defined purposes. It may be redefined to suit installation requirements. USERSECT is pointed to by the AUSER field in NUCON.



Hexadecimal Displacement	Field Name	Field Description, Contents, Meaning
0	USCRTCH DC 18F'0'	User scratch area; may be redefined per installation requirements

Section 3. RSCS Data Areas and Control Blocks

This section describes in detail each of the data areas used by RSCS. Unlike the CP and CMS format blocks in this publication, the RSCS format blocks are on fullword boundaries.

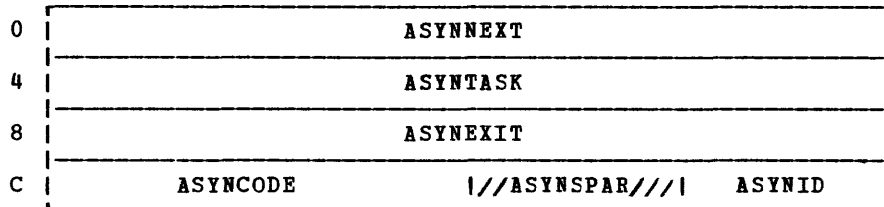
This section of the publication contains only DSECTS. Appendixes B and C contain other control areas used by RSCS.

ASYNE

ASYNE: ASYNCHRONOUS EXIT ELEMENT

ASYNE defines symbolic addresses for elements on an asynchronous exit queue. An asynchronous exit queue element contains information by which a task requests that it handle asynchronous interrupts.

IOEXITQ, EXTQ, and ALERTQ in SVECTORS are the heads of three asynchronous exit queues. Each of these queues is comprised of supervisor elements defined by the ASYNE DSECT. IOEXITQ points to requests for I/O exits, EXTQ points to requests for external exit requests, and ALERTQ points to requests for ALERT exits.



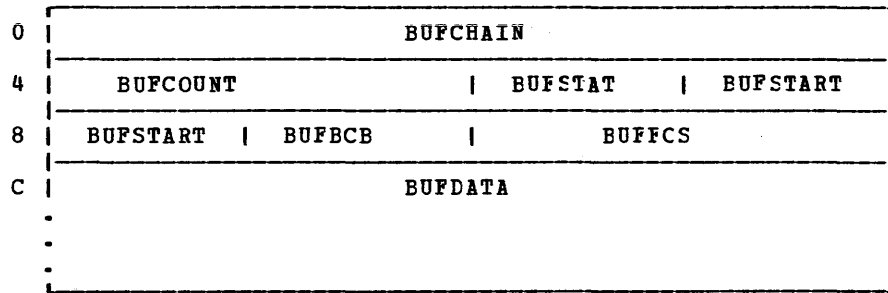
Hexadecimal Displacement	Field Name	Field Description, Contents, Meaning
0	ASYNNEXT DS 1F	Address of the next asynchronous interrupt exit request element
4	ASYNTASK DS 1F	Address of task element describing the task that requested the asynchronous interrupt
8	ASYNEXIT DS 1F	Address of the requested asynchronous exit routine
C	ASYNCODE DS AL2	Address of the device for which asynchronous I/O interrupts are requested or interrupt bit code
E	ASYNSPAR DS 1X	Reserved for IBM use
F	ASYNID DS 1X	1-byte identification of the task owning the asynchronous exit routine

BUFDSECT: SML TELECOMMUNICATIONS BUFFER

BUFDSECT is used to transmit buffer control information and buffer data to and from programmable remote stations.

The buffer sent across the TP line starts at BUFSTART; the first 7 bytes of BUFDSECT are used by the DMTSML line driver but are not transmitted.

\$BUFPOOL in DMTSML points to a queue of available TP buffers; \$INBUF in module DMTSML points to a queue of TP buffers that have been received from a remote station and are waiting to be deblocked. \$OUTBUF in DMTSML points to a queue of TP buffers that are ready for transmission to remote stations.



Hexadecimal Displacement	Field Name	Field Description, Contents, Meaning
0	BUFBEGIN DS 0F	Beginning of the buffer
0	BUFCHAIN DC A(0)	Buffer chain field
4	BUFCOUNT DS 1H	Count of bytes to transmit
6	BUFSTAT DS 1C	Buffer status byte
<u>Bits defined in BUFSTAT</u>		
	BUFFAKE EQU X'01'	Dummy buffer indicator
	BUFRESP EQU X'02'	Response only in buffer
	BUFNACK EQU X'04'	NAK response being sent
	BUFTXT EQU X'08'	Buffer contains text information
	BUFUCHEK EQU X'10'	Unit check expected
7	BUFSTART DS CL2	Bisynchronous transmission control bytes
9	BUFBCB DS 1C	Block control byte
A	BUFFCS DS CL2	Function control sequence
C	BUFDATA DS 0F	Data portion of TP buffer

COMDSECT

COMDSECT: ADDRESS CONSTANTS AS POINTERS

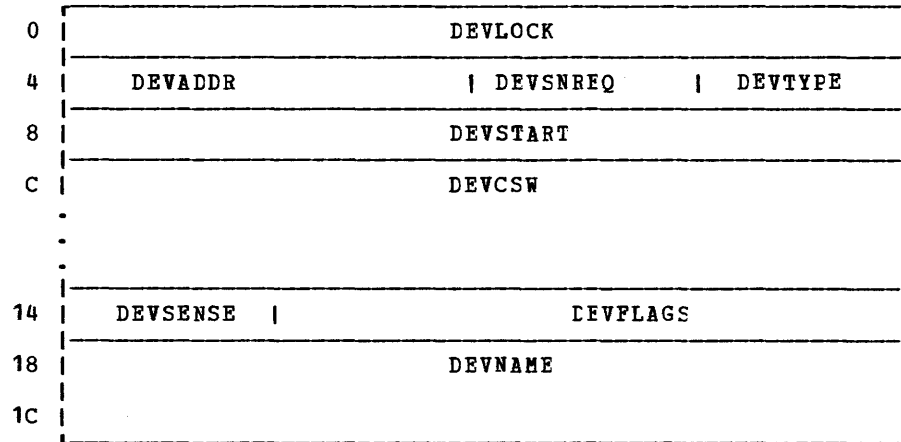
COMDSECT defines address constants used as pointers to subroutines common to all RSCS modules. These subroutines are contained in module DMTCOM. COMDSECT is pointed to by TCOM in SVECTORS.

0	GLINKREQ
4	GPAGEREQ
8	FPAGEREQ
C	PMSGREQ
10	GMSGREQ
14	GTODEBCD

<u>Hexadecimal Displacement</u>	<u>Field Name</u>	<u>Field Description, Contents, Meaning</u>
0	GLINKREQ DS 1A	Get link table entry routine
4	GPAGEREQ DS 1A	Get page of main storage
8	FPAGEREQ DS 1A	Free page of main storage
C	PMSGREQ DS 1A	Put message element into message stack
10	GMSGREQ DS 1A	Remove message element from message stack
14	GTODEBCD DS 1A	Convert S/370 TOD to EBCDIC

DEVTABLE: NPT DEVICE TABLE

DEVTABLE defines virtual device information for an I/O operation. This DSECT corresponds to the IOTABLE DSECT.



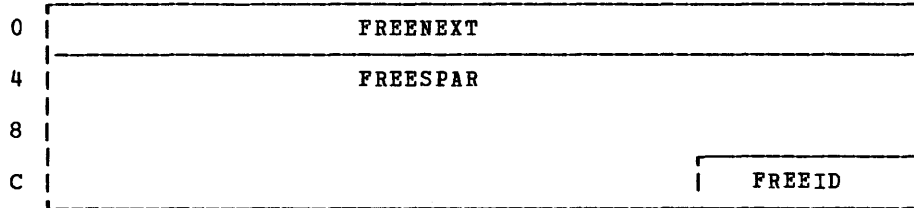
Hexadecimal Displacement	Field Name	Field Description, Contents, Meaning
0	DEVLOCK DC F'0'	I/O request synch lock
4	DEVADDR DC AL2(0)	CUU address of I/O device
6	DEVSREQ DC AL1(0)	Requested sense bytes
7	DEVTYPE DC X'00'	VM/370 device type code
8	DEVSTART DC F'0'	Address start channel program for I/O handling
C	DEVSIOCC DC 0X'00'	Right-justified SIO condition code
C	DEVCSW DC 2F'0'	Ending CSW from last I/O operation
14	DEVSENSE DC X'00'	Sense information on unit check
15	DEVFLAGS DC 3X'00'	Device status flags
18	DEVNAME DC CL8' '	EBCDIC device address and name

FREEE

FREEE: A FREE ELEMENT ON THE SUPERVISOR ELEMENT QUEUE

FREEE defines an element in the chain of elements that comprise the free element queue.

FREEQ in SVECTORS points to the chain of free elements, each of which is defined by the FREEE DSECT.



Hexadecimal Displacement	Field Name	Field Description, Contents, Meaning
0	FREENEXT DS 1F	Address of next element in free queue
4	FREESPAR DS CL11	Spare field
F	FREEID DS 1X	Standard taskid displacement, which is: ID=X'00'=> free element

GIVE REQUEST TABLE

The format of a GIVE Request Table is:

0	Synch Lock
4	Task name or A(GIVE Element)
8	A(GIVE Request Buffer)
C	A(GIVE Response Buffer)

When a task requests the services of another task via a GIVE request, the second field of the table above contains the task name of the task to which the task is to be sent. When DMTGIV builds a GIVE element for the request, it overlays this task name with the address of the GIVE element.

The task performing the requested service builds a table called the TAKE request table, which corresponds to the GIVE request table.

GIVEE: A GIVE ELEMENT

GIVEE defines symbolic addresses for items used in processing a GIVE request.

GIVEQ in SVECTORS points to the queue of GIVE elements used in task-to-task communications.

The GIVEADDR field of this DSECT is the address of a GIVE request table, which, in turn, contains addresses of buffers for elements describing requests and responses to requests. These tables are described below; the elements that fill the buffers are described in "Appendix C: RSCS Request Elements."

0	GIVENEXT
4	GIVEADDR
8	GIVENAME
C	////////GIVESPAR//////// GIVENID GIVERID

Hexadecimal Displacement	Field Name		Field Description, Contents, Meaning
0	GIVENEXT DS	1F	Address of next GIVE element
4	GIVEADDR DS	1F	Address of GIVE request table in sending task's storage
8	GIVENAME DS	CL4	Task name of receiving task
C	GIVESPAR DS	AL2	Reserved for IBM use
E	GIVENID DS	1X	1-byte identification of receiving task after the TAKE request has been processed
F	GIVERID DS	1X	1-byte identification of sending task

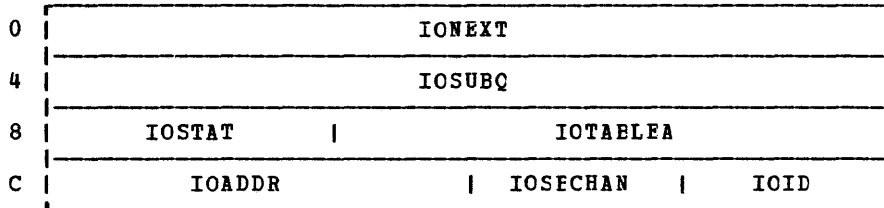
IOE

IOE: AN I/O ELEMENT

IOE defines symbolic addresses of elements and other information associated with an I/O operation requested by a task.

MPXIOQ and SELIOQ in SVECTORS point to queues of I/C elements for the multiplexer and selector channels, respectively.

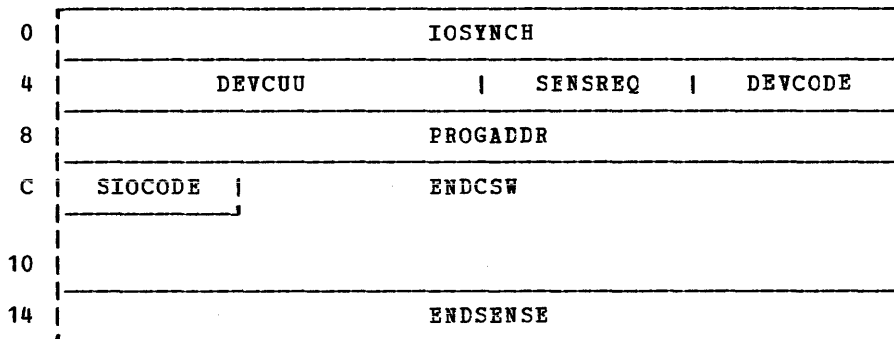
The IOTABLEA field points to the address of an I/O table defined by ICTABLE, which is described in this section.



Hexadecimal Displacement	Field Name		Field Description, Contents, Meaning
0	IONEXT	DS 1F	Address of next active I/O element
4	IOSUBQ	DS 1F	Address of first inactive I/C element for a given subchannel
8	IOSTAT	EQU *	Status flags for current I/C operation (first byte of IOTABLEA)
	<u>Bits defined in IOSTAT</u>		
	SENSING	EQU X'80'	Flag set to 1 while automatic sense is active
	CHANDONE	EQU X'40'	Flag set to 1 when subchannel terminates
8	IOTABLEA	DS 1F	Address of I/O request table in task storage
C	IOADDR	DS AL2	Address (cuu) of the device requesting current I/O operation
E	IOSECHAN	DS 1X	Subchannel address; 1-byte; assigned by MSUP
F	IOID	DS 1X	ID of task associated with this I/O operation is 1-byte and assigned by MSUP

IOTABLE: AN I/O TABLE

IOTABLE defines symbolic addresses for items used in processing an I/O interrupt request. The first five fields are filled in by the task to convey information about the I/O request to the supervisor. The last three fields are filled in by the supervisor to convey status information about the I/O operation to the task.



Hexadecimal Displacement	Field Name			Field Description, Contents, Meaning
0	IOSYNCH	DS	1F	Synchronization lock for I/C operation
4	DEVCUU	DS	AL2	Address (cuu) of device associated with this I/O operation
6	SENSREQ	DS	AL1	Number of sense bytes requested on unit check
7	DEVCODE	DS	AL1	1-byte VM/370 device type code (not used by I/O manager)
8	PROGADDR	DS	1F	Address of channel program for the I/C operation
<u>Bits defined in PROGADDR</u>				
	SIOCOND	EQU	*	1-byte SIO condition code return information
C	SIOCODE			SIO condition code
D	ENDCSW	DS	2F	Ending CSW with composite status return information
14	ENDSENSE	DS	AL1	Requested return sense information on unit check CSW status
<u>Bits defined in ENDSENSE</u>				
	TYP PUN	EQU	X'80'	VM/370 type code for the punch
	TYP PRT	EQU	X'40'	VM/370 type code for the printer

LINKTABL

LINKTABL: LINK TABLE

LINKTABL describes the status of a single link in the RSCS network; collectively, all the links defined for the system are referred to as the link table.

An 8-byte header precedes the first entry in the link table (that is, the first link defined by the LINKTABL DSECT). The TLINKS field in SVECTORS points to this header, which has the following format:

0		4		6	
	total links		maximum links		current links

where:

total links is the total number of links defined for an RSCS installation via the GENLINK macro during system generation. (For information on the GENLINK macro, see the VM/370 System Programmer's Guide.)

maximum links is the maximum number of concurrently active links allowable.

current links is the number of links active in RSCS at a given time.

0	LINKID						
4							
8	LDEFTNME						
C	LACTNME						
10	LDEFDRVR						
14							
18	LACTDRVR						
1C							
20	LDEFLINE		LACTLINE				
24	LDRVRVAR						
28	LDEFCLS1		LDEFCLS2		LDEFCLS3		LDEFCLS4
2C	LACTCLS1		LACTCLS2		LACTCLS3		LACTCLS4
30	LTIMEZON		LFLAG		LRESERVD		
34	LPENDING		LTAKEN				
38	LPOINTER						
3C	LMSGQ						
40	LTRNSCNT		LERRCNT				
44	LTOCNT		LSPARE				
48	LNKCLOCK						
4C							

Hexadecimal Displacement	Field Name		Field Description, Contents, Meaning
0	LINKID	DS CL8	EBCDIC link identification
8	LDEFTNME	DS CL4	Default task name
C	LACTNME	DS CL4	Active task name
10	LDEFDRVR	DS CL8	Default driver identification
18	LACTDRVR	DS CL8	Active driver identification
20	LDEFLINE	DS 2X	Default virtual line address
22	LACTLINE	DS 2X	Active virtual line address
24	LDRVRVAR	DS 1F	Line driver variable information
28	LDEFCLS1	DS CL1	Default spool file Class 1
29	LDEFCLS2	DS CL1	Default spool file Class 2
2A	LDEFCLS3	DS CL1	Default spool file Class 3
2B	LDEFCLS4	DS CL1	Default spool file Class 4
2C	LACTCLS1	DS CL1	Active spool file Class 1
2D	LACTCLS2	DS CL1	Active spool file Class 2
2E	LACTCLS3	DS CL1	Active spool file Class 3
2F	LACTCLS4	DS CL1	Active spool file Class 4
30	LTIMEZON	DS 1X	Displacement of two complete time zones from Greenwich Mean Time
31	LFLAG	DS 1X	Link flag byte
	<u>Bits defined in LFLAG</u>		
	LACTIVE	EQU X'80'	Link active
	LALERT	EQU X'40'	AXS ALERT exit set
	LHOLD	EQU X'20'	Link hold set
	LDRAIN	EQU X'10'	Link drain in progress
	LTRALL	EQU X'08'	Link transaction tracing (all)
	LTRERR	EQU X'04'	Link transaction tracing (error)
	LHALT	EQU X'01'	Link to be forced inactive
	LINKLEN	EQU *-LINKTABL	Length of link table entry
32	LRESERVD	DS 1H	Count of tag elements reserved
34	LPENDING	DS 1H	Count of unaccepted tags
36	LTAKEN	DS 1H	Count of tag slots in use
38	LPOINTER	DS 1F	Address of start of the TAG queue for this RSCS link
3C	LMSGQ	DS 1F	MSG queue pointer
40	LTRNSCNT	DS 1H	Link transaction count
42	LERRCNT	DS 1H	Error count
44	LTOCNT	DS 1H	Timeout count
46	LSPARE	DS 1H	Spare halfword
48	LNKCLOCK	DS 8X	Clock comparator value for this link

REQBLOCK

REQBLOCK: NPT REQUEST BLOCK

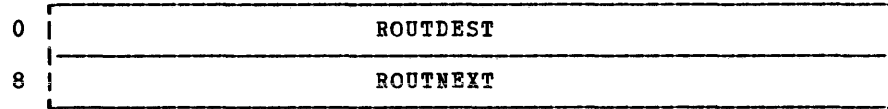
REQBLOCK defines data and information for a request for file processing by the DMTMFT line driver.

The first four fields of this DSECT form a GIVE request table, which is described in "GIVEE: A GIVE Element", in this section.

The next seven fields of this DSECT form a GIVE request buffer in the format of a File Request Element, which is described in "Appendix C: RSCS Request Elements."

0	RDEVSYNC		
4	RDEVREQN		
8	RDEVREQ		
C	RDEVRESP		
10	RDEVRLN	RDEVFUN	/RDEVRESV RDEVSOPT
14	RDEVTAG		
18	RDEVFIOA		
1C	RDEVLINK		
20			
24	RDEVNAME		
28	RDEVSTYL		
2C	RDEVFLAGS		

Hexadecimal Displacement	Field Name	Field Description, Contents, Meaning	
<u>Hexadecimal Locations 0 through F Comprise the NPT GIVE Request Table</u>			
0	RDEVSYNC DC	F'0'	Synchronous lock
4	RDEVREQN DC	CL4'AXS'	File access task
8	RDEVREQ DC	A(0)	Request buffer address
C	RDEVRESP DC	AL1(0),AL3(0)	Response buffer address
<u>Hexadecimal Locations 0 through 1F Comprise the NPI GIVE Request Buffer in the Format of a File Request Element (as shown in Appendix C)</u>			
10	RDEVRLN DC	AL1(0)	Request length
11	RDEVFUN DC	AL1(0)	Request function
12	RDEVRESV DC	AL1(0)	Reserved for IBM use
13	RDEVSOPT DC	AL1(0)	Subordinate option byte
14	RDEVTAG DC	A(0)	Tag address
18	RDEVFIOA DC	A(0)	File I/O area address
1C	RDEVLINK DC	CL8' '	Link name
24	RDEVNAME DC	C' '	Device name
27	RDEVSTYL DC	C' '	Device style
2C	RDEVFLAGS DC	AL1(0,0,0,0)	Device flags

ROUTE: ROUTING TABLE ENTRY

Hexadecimal Displacement	Field Name	Field Description, Contents, Meaning
0	ROUTDEST DS CL8	Final destination ID
8	ROUTNEXT DS CL8	LINKID for indirect routing
	ROUTESIZE EQU *-ROUTDEST	Length of a routing table entry

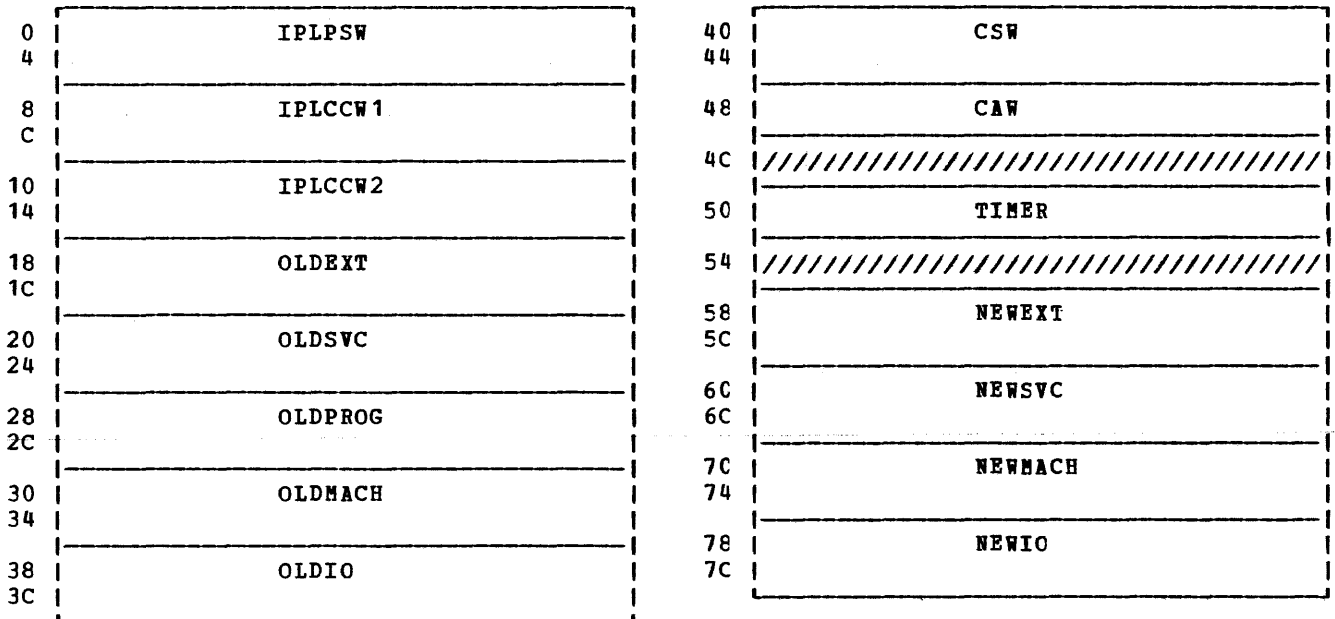
SVECTORS

SVECTORS: LOW STORAGE DEFINITIONS

SVECTORS defines low storage for the RSCS virtual machine. It includes two types of storage: machine-defined and RSCS-defined.

• MACHINE-DEFINED LOW STORAGE

The SVECTORS machine-defined low storage defines machine status data referenced during program execution and required by System/370 architecture.



Hexadecimal Displacement	Field Name	Field Description, Contents, Meaning
0	IPLPSW DS D	X'00040000',V(DMTINI)
8	IPLCCW1 DS D	
10	IPLCCW2 DS D	
18	OLDEXT DS D	External interrupt old PSW
20	OLDSVC DS D	Supervisor call old PSW
28	OLDPROG DS D	Program check old PSW
30	OLDMACH DS D	Machine check old PSW
38	OLDIO DS D	Input/output old PSW
40	CSW DS D	Channel status word
48	CAW DS F	Channel address word
4C	DS F	Reserved for IBM use
50	TIMER DS F	4X' FF' TOD clock
54	DS F	Reserved for IBM use
58	NEWEXT DS D	X'00040000',V(DMIEXT)
60	NEWSVC DS D	X'00040000',V(DMTSVC)
68	NEWPROG DS D	X'00040000',A(REXOUCH)
70	NEWMACH DS D	X'00020000',A(OLDMACH)
78	NEWIO DS D	X'00040000',V(DMTIOMIN)

• RSCS-DEFINED LOW STORAGE

RSCS-defined low storage begins at hexadecimal location 200 and is defined specifically for the RSCS virtual machine. It contains pointers to modules that comprise the supervisor, supervisor control queues, and queues of requests for supervisor services.

200	NEWPSW	248	DISPATCH
204		24C	WAITREQ
208	SSAVE	250	POSTREQ
20C		254	IOREQ
210	ACTIVE	258	TASKREQ
214	MAINMAP	25C	MAINREQ
218	MAINSIZE	260	ASYNREQ
21C	QUEUE	264	ALERTREQ
220	QUEUEEND	268	GIVEREQ
224	FREEQ	26C	TAKEREQ
228	TASKQ	270	TVECTCR0
22C	MPXIOQ	274	TVECTCR1
230	SELIOQ	278	TVECTCR2
234	IOEXITQ	27C	TVECTCR3
238	EXTQ	280	TVECTCR4
23C	ALERTQ	284	TVECTCR5
240	GIVEQ	288	TVECTCR6
244	QREQ	28C	TVECTCR7

Hexadecimal Displacement	Field Name		Field Description, Contents, Meaning
		ORG SVECTORS+X'200'	Leave room for machine extensions
200	NEWPSW	DS D D'0'	Dispatched PSW for last dispatcher
208	SSAVE	DS 2F 2F'0'	General-purpose low storage save area
210	ACTIVE	DS X X'00'	Identifier of currently active task
		DS AL3 AL3(0)	Address of task element for last dispatchee
214	MAINMAP	DS V(DMTMAPMS)	Address of start of main storage allocation map
218	MAINSIZE	DS F F'0'	Total number of pages in main storage

SVECTORS

Hexadecimal Displacement	Field Name			Field Description, Contents, Meaning
21C	QUEUE	DS	V (SQUEUE)	Address of start of supervisor queue
220	QUEUEEND	DS	V (SQUEUEEND)	Address of end of last supervisor queue element
224	FREEQ	DS	A (0)	Address of start of free element queue
228	TASKQ	DS	A (0)	Address of start of task element queue
22C	MPXIOQ	DS	A (0)	Address of start of multiplexer I/O queue
230	SELIOQ	DS	A (0)	Address of start of selectcr I/O queue
234	IOEXITQ	DS	A (0)	Address of start of asynchronous I/O request element queue
238	EXTQ	DS	A (0)	Address of start of external request element queue
23C	ALERTQ	DS	A (0)	Address of start of task asynchronous request element queue
240	GIVEQ	DS	A (0)	Address of start of GIVE request element queue
244	QREQ	DS	V (DMTQRQ)	Supervisor queue allocation request entry address
248	DISPATCH	DS	V (DMTDSP)	Task dispatcher entry address
24C	WAITREQ	DS	V (DMTWAT)	Wait request entry address
250	POSTREQ	DS	V (DMTPST)	Post request entry address
254	IOREQ	DS	V (DMTIOMRQ)	I/O request entry address
258	TASKREQ	DS	V (DMTASK)	Task management request entry address
25C	MAINREQ	DS	V (DMTSTO)	Main allocation request entry address
260	SYNREQ	DS	V (DMTASY)	Asynchronous interrupt request entry address
264	ALERTREQ	DS	A (DMTSIG)	Task asynchronous signal request) A (ALERT) entry address
268	GIVEREQ	DS	V (DMTGIV)	Task request GIVE request entry address
26C	TAKEREQ	DS	V (DMTAKE)	Task request TAKE request entry address
270	TVECTOR0	DS	A (0)	Task defined field
274	TVECTOR1	DS	A (0)	Task defined field
278	TVECTOR2	DS	A (0)	Task defined field
27C	TVECTOR3	DS	A (0)	Task defined field
280	TVECTOR4	DS	A (0)	Task defined field
284	TVECTOR5	DS	A (0)	Task defined field
288	TVECTOR6	DS	A (0)	Task defined field
28C	TVECTOR7	DS	A (0)	Task defined field
	TLINKS	EQU	TVECTOR0	Link table address
	TROUTE	EQU	TVECTOR1	Reserved for IBM use
	TPORTS	EQU	TVECTOR2	Switchable port table address
	TTAGQ	EQU	TVECTOR3	Tag slot queue
	TCOM	EQU	TVECTOR4	Common routine chain

TAG: RSCS FILE DESCRIPTOR

TAG describes a file enqueued for processing by RSCS. The data in this area is built from the TAG record associated with a file via the CP tag command and from the CP spccl file block (SFB) that describes the file.

0	TAGNEXT		
4	TAGBLOCK		
8	TAGINLOC		
C			
10	TAGLINK		
14			
18	TAGINTOD		
1C			
20	TAGINVM		
24			
28	TAGRECNM		
2C	TAGRECLN	TAGINDEV	TAGCLASS
30	TAGID		TAGCOPY
34	TAGFLAG	TAGFLAG2	/////////RESERVED/////////
38	TAGNAME		
3C			
40			
44	TAGTYPE		
48			
4C			
50	TAGDIST		
54			
58	TAGTOLOC		
5C			
60	TAGTOVM		
64			
68	TAGPRIOR		TAGDEV

Hexadecimal Displacement	Field Name		Field Description, Contents, Meaning
0	TAGNEXT	DS 1F	Address of next active queue entry
4	TAGBLOCK	DS 1F	Address of associated I/O area
8	TAGINLOC	DS CL8	Originating location
10	TAGLINK	DS CL8	Next location for transmission
18	TAGINTOD	DS CL8	Time of file origin
20	TAGINVM	DS CL8	Originating virtual machine
28	TAGRECNM	DS 1F	Number of records in file
2C	TAGRECLN	DS 1H	Maximum file data record length
2E	TAGINDEV	DS 1X	Device code of originating device
2F	TAGCLASS	DS CL1	File output class
30	TAGID	DS 1H	File number at origin location
32	TAGCOPY	DS 1H	Number of copies required
34	TAGFLAG	DS 1X	VM/370 SFELCK control flags (SFBFLAG)
35	TAGFLAG2	DS 1X	VM/370 SFELCK control flags (SFEFLAG)
36		DS 1H	Reserved for IBM use
38	TAGNAME	DS CL12	Filename
44	TAGTYPE	DS CL12	Filetype
50	TAGDIST	DS CL8	File distribution code
58	TAGTOLOC	DS CL8	Destination location ID
60	TAGTOVM	DS CL8	Destination virtual machine ID
68	TAGPRIOR	DS CL2	Transmission priority
6A	TAGDEV	DS 2X	Active file's virtual device address
	TAGLEN	EQU *-TAGNEXT	Length (in bytes) of the file TAG (X'0D')

TAGAREA

TAGAREA in DMTAXS module contains tag queue pointers and other tag control information. It is pointed to by TTAGQ in SVECTORS.

0	TAGAFREE	
4	TAGACIN	
8	TAGACOUT	
C	TAGAGOT	TAGAHOLD

Hexadecimal Displacement	Field Name		Field Description, Contents, Meaning
0	TAGAFREE	DC A(0)	Address of queue of free TAG slots (or elements)
4	TAGACIN	DC A(0)	Pointer to queue of active input TAGs
8	TAGACOUT	DC A(0)	Pointer to queue of active output TAGs
C	TAGAGOT	DC H'0'	Number free slots left
E	TAGAHOLD	DC H'0'	Number slots to be held

TAKE REQUEST TABLE

The format of a TAKE request table is:

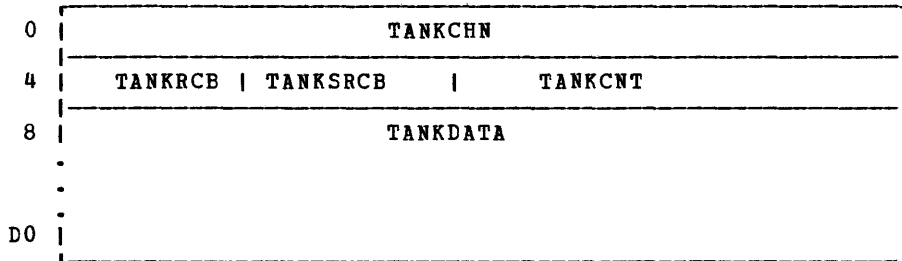
0	Task name of GIVE requestor
4	A(TAKE Request Buffer)
8	A(TAKE Response Buffer)

| The TAKE request table corresponds to the GIVE request table. It is built by the task performing the requested service (via a GIVE request by another task).

TANKDSEC

TANKDSEC: SML UNIT RECORD TANK

TANKDSEC is used to reference buffer data and control information contained in tanks, which are unit buffers used to deblock the larger TP buffers. (TP buffers are defined by the needs of an individual remote station and their size varies from station to station.) \$TANKPOL in module DMTSML points to a queue of available tanks.



Hexadecimal Displacement	Field Name	Field Description, Contents, Meaning
0	TANKCHN DC A(0)	Tank chain field
4	TANKRCB DC 1C	Tank record control byte
5	TANKSRCB DS 1C	Tank subrecord control byte
6	TANKCNT DS 1H	Count of data bytes in tank
8	TANKDATA DS CL200	Data area in tank
D0	TANKEND DS 0F	Force next to word boundary

TAREA: A TASK SAVE AREA

TAREA an area associated with each task. This area is used to save the contents of the task's PSW and general registers and to flag whether or not a task has information ready to pass. TAREA comprises the first 78 bytes of the storage area defined in each task's storage.

0	TPSW
4	
8	TGREG0
C	TGREG1
10	TGREG2
14	TGREG3
18	TGREG4
1C	TGREG5
20	TGREG6
24	TGREG7
28	TGREG8
2C	TGREG9
30	TGREG10
34	TGREG11
38	TGREG12
3C	TGREG13
40	TGREG14
44	TGREG15
48	TREQLOCK

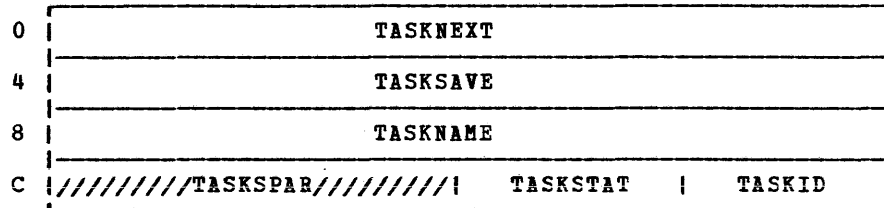
TAREA

Hexadecimal Displacement	Field Name			Field Description, Contents, Meaning
0	TPSW	DS	1D	PSW with which a temporarily interrupted task resumes execution
8	TGREG0	DS	1F	Save area for general register 0
C	TGREG1	DS	1F	Save area for general register 1
10	TGREG2	DS	1F	Save area for general register 2
14	TGREG3	DS	1F	Save area for general register 3
18	TGREG4	DS	1F	Save area for general register 4
1C	TGREG5	DS	1F	Save area for general register 5
20	TGREG6	DS	1F	Save area for general register 6
24	TGREG7	DS	1F	Save area for general register 7
28	TGREG8	DS	1F	Save area for general register 8
2C	TGREG9	DS	1F	Save area for general register 9
30	TGREG10	DS	1F	Save area for general register 10
34	TGREG11	DS	1F	Save area for general register 11
38	TGREG12	DS	1F	Save area for general register 12
3C	TGREG13	DS	1F	Save area for general register 13
40	TGREG14	DS	1F	Save area for general register 14
44	TGREG15	DS	1F	Save area for general register 15
48	TREQLOCK	DS	1F	Synchronization lock used to signal whether or not a task has information

TASKE: A TASK ELEMENT

TASKE defines symbolic names of status information pertaining to an active task.

The TASKQ field of SVECTORS points to a queue of task elements, each of which is defined by this DSECT. The queue consists of one task element (TASKE) for each active task.



Hexadecimal Displacement	Field Name	Field Description, Contents, Meaning
0	TASKNEXT DS 1F	Address of the next element on the task element queue
4	TASKSAVE DS 1F	Address of this task's Task Save Area (TAREA)
8	TASKNAME DS CL4	Task name specified by the task; 4 bytes long
C	TASKSPAR DS AL2	Reserved for IBM use
E	TASKSTAT DS 1X	Status flags associated with the task
	<u>Bits defined in TASKSTAT</u>	
	WAITING EQU X'80'	Flag set to 1 when task is nondispatchable
	LOCKLIST EQU X'40'	Flag set to 1 while task is waiting for the synchronous lock list
	LIMBO EQU X'01'	Flag set to 1 when a task is being terminated.
F	TASKID DS 1X	Number ID for the task; 1 byte is assigned by the supervisor when task is made active

TCTDSECT

TCTDSECT: TASK CONTROL TABLE

TCTDSECT defines the format of six tables used by module DMSML that are at labels \$CCOM1, \$WCOM1, \$PCOM1, \$RCOM1, \$UCOM1, and \$JCOM1. Each table corresponds to a DMSML input/output processor and is used by that processor to perform its I/C function.

The GIVE request table and the GIVE request buffer used by DMSML are embedded in the task control table at locations X'24' through X'30' and X'34' through X'40', respectively.

0	TCTSTRT		TCTENTY
4	TCTRTRN		
8	TCTCCW		TCTDATA
C	TCTFLAG		TCTOPCOD TCTCCWCT
10	TCTECB		TCTSTAT TCTWFE
14	TCTSAV1		
18	TCTNEXT		
1C	TCTFCS		TCTRCER TCTRCPT
20	TCTCOM		
24	TDEVSYNC		
28	TDEVREQN		
2C	TDEVREQ		
30	TDEVRESP		
34	TDEVRLN		TDEVFUN ///TDEVRESV/// TDEVSOPT
38	TDEVTAG		
3C	TDEVFIOA		
40	TDEVLINK		
44			
48	TSW1		TSW2 TSW3 TSW4
4C	TCTTOVM		
50			
54	TCTTANK		
58	TCTBUFER		
5C	TCTTNKLM		TCTTNKCT TCTBUFLM ICTBUFCT

Hexadecimal Displacement	Field Name		Field Description, Contents, Meaning
0	TCTCT	DS 0H	
0	TCTSTRT	DS CL2	Branch to proper processor entry
2	TCTENTY	DS CL2	Address portion modified by processor
4	TCTRTN	DS CL4	Branch to next processor via commutator
8	TCTCCW	DS CL1	CCW for device operation code
9	TCTDATA	DS AL3	Address of data transferred
C	TCTFLAG	DS CL1	Flags on CCW
D	TCTOPCOD	DS CL1	Save area for CCW operation code
E	TCTCCWCT	DS AL2	CCW count of data transferred
10	TCTECB	DS CL1	Event control block
<u>Bits defined in TCTECB</u>			
	TCTBUSY	EQU X'10'	Device busy bit
11	TCTSTAT	DS CL1	Status flags
<u>Bits defined in TCTSTAT</u>			
	TCT1052	EQU X'10'	TCT status flags for 1052
	TCTREL	EQU X'04'	Interlock release request for console
	TCTOPEN	EQU X'80'	TCT open bit
	TCTACT	EQU X'40'	Action required on this TCT
12	TCTWFB	DS AL1	Waiting for buffers
14	TCTSAV1	DS 1F	Save area for processor routine
18	TCTNEXT	DS 1F	Next TCT in chain
1C	TCTFCS	DS AL2	Function control sequence mask
1E	TCTRCBR	DS CL1	RECV record control block
1F	TCTRCBT	DS CL1	Trans record control block
20	TCTCOM	DS 1F	Pointer back to commutator
<u>Hexadecimal Locations 24 through 30 Comprise the SML Give Request Table</u>			
24	TDEVSYNC	DS 1F	Synch lock
28	TDEVREQN	DS CL4	File access name
2C	TDEVREQ	DS 1A	Request buffer address
30	TDEVRESP	DS 1A	Response buffer
<u>Hexadecimal Locations 34 through 4C Comprise the Give Request Buffer in the Form of a File Request Element (as shown in Appendix C)</u>			
34	TDEVRLN	DS AL1	Request length
35	TDEVFUN	DS AL1	Request function
36	TDEVRESV	DS AL1	Reserved for IBM use
37	TDEVSOPT	DS AL1	Subordinate option byte
38	TDEVTAG	DS 1A	Tag address
3C	TDEVFIOA	DS 1A	File I/O area
40	TDEVLINK	DS CL8	Link name
48	TSW1	DS AL1	Device switch 1
49	TSW2	DS AL1	Device switch 2
4A	TSW3	DS AL1	Device switch 3
4B	TSW4	DS AL1	Device switch 4
4C	TCTTOVM	DS CL8	Virtual machine output destination
54	TCTTANK	DS 1F	Next tank to generate output
58	TCTBUFFER	DS 1F	Address of current buffer
5C	TCTTNKLM	DS CL1	Maximum number of tanks assignable
5D	TCTTNKCT	DS CL1	Current number assigned
5E	TCTBUFLM	DS CL1	Maximum number of buffers assignable
5F	TCTBUFCT	DS CL1	Current number assigned

Appendixes

Information in the following appendixes supplements the text in Sections 1 through 3 of this publication:

- "Appendix A. CP and RSCS Equate Symbols" contains assembler language equate symbols used in CP and RSCS to reference data.
- "Appendix B. RSCS Control Areas" shows those constants and variables used during execution of RSCS tasks.
- "Appendix C. RSCS Request Elements" contains information on and formats of tables used during RSCS task-to-task communication.
- "Appendix D. CMS Equate Symbols" contains Assembler language equate symbols used in CMS to reference data.
- "Appendix E. Data Areas and Control Block References" lists the names of CP, CMS, and RSCS control blocks. This appendix (1) shows module references to data areas and/or control blocks and (2) gives information on how certain data areas or control blocks are created and released.

Appendix A. CP and RSCS Equate Symbols

This Appendix contains Assembler language equate symbols used to reference CP and RSCS data for:

- VM/370 Device Classes, Types, Models, and Features
- VM/370 Machine Usage
- VM/370 Extended Control Registers
- VM/370 CP Usage
- VM/370 Registers

VM/370 DEVICE CLASSES, TYPES, MODELS, AND FEATURES

Field Name			Field Description, Contents, Meaning
CLASTERM	EQU	X'80'	Terminal device class
TYP2700	EQU	X'40'	2700 bisynchronous line
TYP2955	EQU	TYP2700	2955 communications line
TYPTELE2	EQU	X'20'	Telegraph terminal control type II
TYPTTY	EQU	X'20'	Teletype terminal
TYPIBM1	EQU	X'10'	IBM terminal control type I
TYP2741	EQU	X'18'	2741 communications terminal
TYP1050	EQU	X'14'	1050 communications terminal
TYPUNDEF	EQU	X'1C'	Terminal device type is undefined
TYPBSC	EQU	X'80'	Bisynchronous line for 3270 remote stations
TYP3210	EQU	X'00'	3210 console
TYP3215	EQU	TYP3210	3215 console
TYP2150	EQU	TYP3210	2150 console
TYP1052	EQU	TYP3210	1052 console
FTRDIAL	EQU	X'01'	Dial feature
CLASGRAF	EQU	X'40'	Graphics device class
TYP2250	EQU	X'80'	2250 display unit
TYP2260	EQU	X'40'	2260 display station
TYP2265	EQU	X'20'	2265 display station
TYP3066	EQU	X'10'	3066 console
TYP1053	EQU	X'08'	1053 printer
TYP3277	EQU	X'04'	3277 display station
TYP3278	EQU	X'01'	3278 Model 2A system console
TYP3284	EQU	X'02'	3284 printer
TYP3286	EQU	TYP3284	3286 printer
TYP3287	EQU	TYP3284	3287 printer
TYP3288	EQU	TYP3284	3288 printer
TYP3138	EQU	TYP3277	3138 system console
TYP3148	EQU	TYP3277	3148 system console
TYP3158	EQU	TYP3277	3158 system console
FTROPRDR	EQU	X'80'	Operator identification card reader
CLASUR1	EQU	X'20'	Unit record input device class
TYPRDR	EQU	X'80'	Card reader device
TYP2501	EQU	X'81'	2501 card reader
TYP2540R	EQU	X'82'	2540 card reader
TYP3505	EQU	X'84'	3505 card reader
TYP1442R	EQU	X'88'	1442 card reader/punch
TYP2520R	EQU	X'90'	2520 card reader/punch
TYPTIMER	EQU	X'40'	Timer device
TYPTR	EQU	X'20'	Tape reader device
TYP2495	EQU	X'21'	2495 magnetic tape cartridge reader
TYP2671	EQU	X'22'	2671 paper tape reader
TYP1017	EQU	X'24'	1017 paper tape reader
CLASURO	EQU	X'10'	Unit record output device class
TYP PUN	EQU	X'80'	Card punch device
TYP2540P	EQU	X'82'	2540 card punch
TYP3525	EQU	X'84'	3525 card punch
TYP1442P	EQU	X'88'	1442 card punch
TYP2520P	EQU	X'90'	2520 card punch
TYP PRT	EQU	X'40'	Printer type device
TYP1403	EQU	X'41'	1403 printer
TYP3211	EQU	X'42'	3211 printer
TYP3203	EQU	X'43'	3203 printer (3211 and 1403)
TYP1443	EQU	X'44'	1443 printer

Field Name			Field Description, Contents, Meaning
TYP3800	EQU	X'45'	3800 Printing subsystem
TYP2TP	EQU	X'20'	Tape punch device
TYP1018	EQU	X'24'	1018 paper tape punch
FTRUCS	EQU	X'01'	UCS feature
FTR4WCGM	EQU	X'80'	3800 has four WCGM available. Note that FTREXTSN (X'40') is also used for a 3800 printer.
CLASTAPE	EQU	X'08'	Magnetic tape device class
TYP2401	EQU	X'80'	2401 tape drive
TYP2415	EQU	X'40'	2415 tape drive
TYP2420	EQU	X'20'	2420 tape drive
TYP3420	EQU	X'10'	3420 tape drive
TYP3410	EQU	X'08'	3410 tape drive
TYP3411	EQU	TYP3410	3411 tape drive
FTR7TRK	EQU	X'80'	7-track feature
FTRDLDNS	EQU	X'40'	Dual density feature
FTRTRANS	EQU	X'20'	Translate feature
FTRDCONV	EQU	X'10'	Data conversion feature
CLASDASD	EQU	X'04'	Direct access storage device class
TYP2311	EQU	X'80'	2311 disk storage drive
TYP2314	EQU	X'40'	2314 disk storage facility
TYP2319	EQU	TYP2314	2319 disk storage facility
TYP2321	EQU	TYP2311	2321 data cell drive
TYP3330	EQU	X'10'	3330 disk storage facility
TYP3333	EQU	TYP3330	3333 disk storage facility
TYP3350	EQU	X'08'	3350 disk storage facility
TYP2301	EQU	TYP2311	2301 parallel drum
TYP2303	EQU	TYP2311	2303 serial drum
TYP2305	EQU	X'02'	2305 fixed head storage device
TYP3340	EQU	X'01'	3340 disk storage facility
FTRRPS	EQU	X'80'	Rotational positional sensing (RPS) installed (3340)
FTREXTSN	EQU	X'40'	Extended sense bytes (24 bytes)
FTR2311T	EQU	X'20'	(= VDEV231T) Top half of 2314 used as 2311
FTR2311B	EQU	X'10'	(= VDEV231B) Bottom half of 2314 used as 2311
FTR35MB	EQU	X'08'	35 multibyte data module mounted (3340)
FTR70MB	EQU	X'04'	70 multibyte data module mounted (3340)
FTRRSRL	EQU	X'02'	Reserve/release are valid CCW operation codes
VIRTUAL	EQU	X'01'	Device is a 3330V virtual machine
SYSVIRT	EQU	X'20'	Device is a 3330V system virtual machine
FTRVIRT	EQU	X'01'	3330 virtual (MSS) volume
CLASSPEC	EQU	X'02'	Special device class
TYPCTCA	EQU	X'80'	Channel-to-channel adapter
TYP3704	EQU	X'40'	3704 programmable communication control unit
TYP3705	EQU	TYP3704	3705 programmable communication control unit
TYP3851	EQU	X'20'	3851 Mass Storage Controller
TYP5SRF	EQU	X'04'	SRF device (#7443)
TYPUNSUP	EQU	X'01'	Device not supported by VM/370
FTRTYP1	EQU	X'10'	Type 1 channel adapter (370x)
FTRTYP2	EQU	X'20'	Type 2 channel adapter (370x)
FTRTYP3	EQU	FTRTYP2	Treat as type 2 channel adapter (370x)
FTRTYP4	EQU	FTRTYP1	Treat as type 1 channel adapter (370x)

VM/370 EQUATE SYMBOLS -- MACHINE USAGE

Field Name	Field Description, Contents, Meaning
------------	--------------------------------------

Bits defined in Standard/Extended PSW

EXTMODE EQU X'08'	Bit 12 - extended mode
MCHEK EQU X'04'	Bit 13 - machine check enabled
WAIT EQU X'02'	Bit 14 - wait state
PROBMODE EQU X'01'	Bit 15 - problem state

Bits defined in Extended PSW

PERMODE EQU X'40'	Bit 01 - PER enabled
TRANMODE EQU X'04'	Bit 05 - translate mode
IOMASK EQU X'02'	Bit 06 - summary I/O mask
EXTMASK EQU X'01'	Bit 07 - summary external mask

Bits defined in Channel Status Word (CSW)

ATTN EQU X'80'	Bit 32 - attention
SM EQU X'40'	Bit 33 - status modifier
CUE EQU X'20'	Bit 34 - control unit end
BUSY EQU X'10'	Bit 35 - busy
CE EQU X'08'	Bit 36 - channel end
DE EQU X'04'	Bit 37 - device end
UC EQU X'02'	Bit 38 - unit check
UE EQU X'01'	Bit 39 - unit exception
PCI EQU X'80'	Bit 40 - program-control interrupt
IL EQU X'40'	Bit 41 - incorrect length
PRGC EQU X'20'	Bit 42 - program check
PRTC EQU X'10'	Bit 43 - protection check
CDC EQU X'08'	Bit 44 - channel data check
CCC EQU X'04'	Bit 45 - channel control check
IFCC EQU X'02'	Bit 46 - interface control check
CHC EQU X'01'	Bit 47 - chaining check

Bits defined in Channel Command Word (CCW)

CD EQU X'80'	Bit 32 - chain data
CC EQU X'40'	Bit 33 - command chain
SILI EQU X'20'	Bit 34 - suppress incorrect length indicator
SKIP EQU X'10'	Bit 35 - suppress data transfer
PCIF EQU X'08'	Bit 36 - program-control interrupt FETCH
IDA EQU X'04'	Bit 37 - indirect data address

Bits defined in Sense Byte 0 (common to most devices)

CMDREJ EQU X'80'	Bit 0 - command reject
INTREQ EQU X'40'	Bit 1 - intervention required
EUSOUT EQU X'20'	Bit 2 - bus out
EQCHK EQU X'10'	Bit 3 - equipment check
DATACHK EQU X'08'	Bit 4 - data check

VM/370 EQUATE SYMBOLS -- EXTENDED CONTROL REGISTERS

Field Name	Field Description, Contents, Meaning
<u>Bits defined in CREG0</u>	
• BYTE 0	
BLKMPX EQU X'80'	Bit 00 - enable block multiplexing
SSMSUPP EQU X'40'	Bit 01 - enable SSM suppression
TODSYNC EQU X'20'	TOD synchronous control
• BYTE 1	
PAGE4K EQU X'80'	Bit 08 - use 4K pages
PAGE2K EQU X'40'	Bit 09 - use 2K pages
SEG1M EQU X'10'	Bit 11 - use 1M segments
• BYTE 2	
MFAMASK EQU X'80'	Bit 16 - malfunction alert mask
EMSMASK EQU X'40'	Bit 17 - emergency signal mask
XCMASK EQU X'20'	Bit 18 - external call mask
SYNCMASK EQU X'10'	Bit 19 - TOD synchronous check mask
CKCMASK EQU X'08'	Bit 20 - mask on clock comparator interrupt
CPTMASK EQU X'04'	Bit 21 - mask on processor timer interrupt
• BYTE 3	
INTMASK EQU X'80'	Bit 24 - mask on interval timer interrupt
KEYMASK EQU X'40'	Bit 25 - mask on operator key interrupt
SIGMASK EQU X'20'	Bit 26 - mask on external signals 2 through 7
<u>Bits defined in CREG8</u>	
• BYTE 3	
PERFCL EQU X'80'	Sample hardware/software utilizations
RESPCL EQU X'40'	Trace response class
SCHEDCL EQU X'20'	Trace scheduler activity class
TIMECL EQU X'10'	Execution timing class
USERCL EQU X'08'	Sample user resource usage class
PRIVCL EQU X'04'	Privileged operands class
DASDCL EQU X'02'	Sample DASDs; utilizations class
SEEKCL EQU X'01'	Trace DASD seek activity
• BYTE 4	
SPROFCL EQU X'80'	Trace system profile class
<u>Bits defined in CREG9</u>	
• BYTE 0	
PERSUBR EQU X'80'	Bit 00 - monitor successful branches
PERIFET EQU X'40'	Bit 01 - monitor instruction fetches
PERSALT EQU X'20'	Bit 02 - monitor storage alteration
PERGPRS EQU X'10'	Bit 03 - monitor register alteration
<u>Bits defined in CREG14</u>	
• BYTE 0	
HARDSTOP EQU X'80'	Bit 00 - check stop control
SYNCLOG EQU X'40'	Bit 01 - synchronous logout control
IOLOG EQU X'20'	Bit 02 - I/O logout control
RECOVERPT EQU X'08'	Bit 04 - recovery report mask
CONFGRPT EQU X'04'	Bit 05 - configuration report mask
DAMAGRPT EQU X'02'	Bit 06 - external damage report mask
WARNGRPT EQU X'01'	Bit 07 - warning condition report mask
• BYTE 1	
ASYNELG EQU X'80'	Bit 08 - asynchronous extended logout control
ASYNFLOG EQU X'40'	Bit 09 - asynchronous fixed logout control

VM/370 EQUATE SYMBOLS -- CP USAGE

Field Name	Field Description, Contents, Meaning
------------	--------------------------------------

Bits defined for TRANS macro

BRING EQU X'80'	Bring requested page
DEFER EQU X'40'	Defer execution until page in core
LOCK EQU X'20'	Lock page for I/O operation
IOERETN EQU X'10'	Return I/O errors to caller
SYSTEM EQU X'08'	Call to DMKPTRAN for system virtual machine space
VFAULT EQU X'04'	DMKPTRAN call for virtual page. Caller will not utilize real address

Equates for Parameter Field for Calls to DMKELDRI/DMKELLRL

DELSEGS EQU X'80'	Release the segment tables
DELPAGES EQU X'40'	Release the page/swap tables
VRALOC EQU X'20'	Attempt allocation of Virtual=Real area
PAGTONLY EQU X'10'	Only one page table and return
NEWPAGES EQU X'08'	Build new page/swap table
NEWSEGS EQU X'04'	Build new segment table
KEEPSEGS EQU X'02'	Retain information in old segment table
OLDVMSEG EQU X'01'	VMSEG pointer in VMELOK valid

Bits defined for Terminal I/O via DMKQCN

NOTRESP EQU X'4000'	Output - Message not a command response
ERRMSG EQU X'0800'	Output - control program error message
NORET EQU X'0400'	Output - return immediately after call
DFRET EQU X'0200'	Output - FRET buffer after queueing
OPERATOR EQU X'0100'	Output - message for system operator
LOGDROP EQU X'80'	Output - logoff and drop line after message
LOGHOLD EQU X'40'	Output - logoff and hold line after message
PRIORITY EQU X'20'	Output - write this message immediately
VMGENIO EQU X'10'	I/O request generated by virtual machine
NOAUTO EQU X'04'	Output - suppress automatic carriage return
ALARM EQU X'02'	Output - sound the audible alarm
NOTIME EQU X'01'	Output - suppress time stamp on message
INHIBIT EQU X'08'	Input - prevent display of this data
EDIT EQU X'04'	Input - edit input data for corrections
UCASE EQU X'02'	Input - translate data to uppercase

Equates for Spool File Recovery Routine - DMKCKS

CHGSHQ EQU X'0200'	Checkpoint a SHQBLCK
CHGRDV EQU X'0100'	Change attributes of real device
ACTSFB EQU X'80'	File being printed or punched
OPNSFB EQU X'40'	An open print-punch file
DELSFB EQU X'20'	Delete SFBLOK from checkpoint
CHGSFB EQU X'10'	Change existing SFBLOK
ADDSFB EQU X'08'	Add new SFBLOK to recovery cylinder
PRTCHN EQU X'04'	SFBLOK goes on print chain
PCHCHN EQU X'02'	SFBLOK goes on punch chain
RDRCHN EQU X'01'	SFBLOK goes on reader chain

Equates for SWTCHVM macro

SVMUNLOK EQU X'04'	Unlock only the current virtual machine
SVMNOUPD EQU X'02'	Lock virtual machine with NOUPDT option
SVMSTAY EQU X'01'	Stack CPEXBLOK for current processor

Field Name	Field Description, Contents, Meaning		
<u>Monitor Class and Code Definitions</u>			
MNCLPERF	EQU	X'00'	Monitor perform class
MNCOSYS	EQU	X'0000'	Perform class, system performance
MNCOTH	EQU	X'0061'	Monitor tape header record
MNCOTT	EQU	X'0062'	Monitor tape trailer record
MNCOSUS	EQU	X'0063'	Monitor collection suspension record
MNCLRESP	EQU	X'01'	Monitor response class
MNCOBRD	EQU	X'0000'	Response class, begin read code
MNCOWRIT	EQU	X'0001'	Response class, write code
MNCOERD	EQU	X'0002'	Response class, end read code
MNCLSCH	EQU	X'02'	Monitor schedule class
MNCODQ	EQU	X'0002'	Schedule class, drop queue code
MNCOAQ	EQU	X'0003'	Schedule class, add to queue code
MNCOAEL	EQU	X'0004'	Schedule class, add to eligible list code
MNCLUSER	EQU	X'04'	Monitor user class
MNCOUSER	EQU	X'0000'	User class, user data
MNCLINST	EQU	X'05'	Monitor instruction simulation class
MNCOSIM	EQU	X'0000'	Instruction class; instruction simulation code
MNCLDAST	EQU	X'06'	Monitor DASD/tape class
MNCODASH	EQU	X'0000'	DASTAP class, first record
MNCODAS	EQU	X'0001'	DASTAP class, data records
MNCLSEEK	EQU	X'07'	Monitor DASD class
MNCOCYL	EQU	X'0000'	DASD class, seeks code
MNCLSYS	EQU	X'08'	Monitor system profile class
MNCODA	EQU	X'0002'	SYS class, DASD data

Field Name	Field Description, Contents, Meaning
<u>Equates for SIGNAL Macro</u>	
SIGSENSE EQU X'01'	Sense order code
SIGXC EQU X'02'	External call order code
SIGEMS EQU X'03'	Emergency signal order code
SIGSTART EQU X'04'	Start order code
SIGSTOP EQU X'05'	Stop order code
SIGREST EQU X'06'	Restart order code
SIGIPR EQU X'07'	Initial program reset order code
SIGPR EQU X'08'	Program reset order code
SIGSSS EQU X'09'	Stop and store status order code
SIGIML EQU X'0A'	Initial microprogram load order code
SIGICR EQU X'0B'	Initial processor reset order code
SIGCR EQU X'0C'	Processor reset order code emergency signals
SIGQUI EQU X'800'	Quiesce emergency signal
SIGEXT EQU X'400'	Extend emergency signal
SIGSYNC EQU X'200'	Clock synchronization emergency signal
SIGSHD EQU X'100'	Shutdown emergency signal
SIGCLK EQU X'080'	Clock check signal (external call signals)
SIGAPR EQU X'800'	Automatic processor recovery (external call signal)
SIGRES EQU X'400'	Resume external call signal
SIGWAKE EQU X'200'	Wakeup external call signal
SIGDISP EQU X'100'	Dispatch external call signal

VM/370 REGISTERS

Field
Name

Field Description, Contents, Meaning

Symbolic Register Equates

R0	EQU	0	}	General register definitions
R1	EQU	1		
R2	EQU	2		
R3	EQU	3		
R4	EQU	4		
R5	EQU	5		
R6	EQU	6		
R7	EQU	7		
R8	EQU	8		
R9	EQU	9		
R10	EQU	10		
R11	EQU	11		
R12	EQU	12		
R13	EQU	13		
R14	EQU	14		
R15	EQU	15		
Y0	EQU	0	}	Floating-point register definitions
Y2	EQU	2		
Y4	EQU	4		
Y6	EQU	6		
C0	EQU	0	}	Control register definitions
C1	EQU	1		
C2	EQU	2		
C3	EQU	3		
C4	EQU	4		
C5	EQU	5		
C6	EQU	6		
C7	EQU	7		
C8	EQU	8		
C9	EQU	9		
C10	EQU	10		
C11	EQU	11		
C12	EQU	12		
C13	EQU	13		
C14	EQU	14		
C15	EQU	15		

Appendix B. RSCS Control Areas

This appendix lists the control areas used during task processing. Information such as lists of synchronous locks, channel programs, TAKE request tables and buffers, and various work constants are included in these areas.

AXS MONITOR CONTROL AREA

The AXS Monitor Control Area is a data area used by DMTAXS to set up synch locks, a TAKE request table, a TAKE request and response buffer, and an input buffer.

<u>Field Name</u>		<u>Field Description, Contents, Meaning</u>
AXSNAME	DC 0F'0',CL4'AXS'	Task name for AXS routine
REXNAME	DC 0F'0',CL4'REX'	Task name for control monitor
AXSLOCKS	DC A (ARRLOCK) DC A (REQLOCK) DC X'80',AL3 (CMDLOCK)	File tag arrival synchronous lock address Request arrival synchronous lock address Command synchronous lock address
ARRLOCK	DC F'0'	File tag arrival synchronous lock
CMDLOCK	DC F'0'	Command synchronous lock
AXSTAKE	DC 0F'0' DC CL4' ' DC AL1(L'AXSREQ) AL3 (AXSREQ) DC A (AXSRESP)	TAKE request table Giver's task name Pointer to request buffer (input) Pointer to response buffer (output)
AXSREQ	DC XL140'00'	TAKE request buffer
AXSRESP	DC XL136'00'	TAKE response buffer
CMDIN	DC CL122' '	Input buffer for command element
CMDINPGS	DC X'00'	Command in progress switch
AXSCSAVE	DC 18F'0'	Common routine save area

REX MONITOR CONTROL AREA

The REX Monitor Control Area is a data area used by DMTREX to initialize the DMTAXS and DMTLAX modules; point to queues of system data (such as the link table chain and the chain of tag elements), set up a series of synch locks for REX processing function, set up a console element, read and write channel programs, a console table buffer, a message buffer, a TAKE request table, and buffer, and various work constants.

<u>Field Name</u>		<u>Field Description, Contents, Meaning</u>
REFREX	DC 0F'0',CL4'REX'	Task name
REXAXS	DC 0F'0',CL8'DMTAXS'	AXSname
REXLAX	DC 0F'0',CL8'DMTLAX'	LAXname
REXTVECT	DC 0F'0'	
	DC V(DMTSYSCLK)	Link table chain
	DC V(DMTSYSRT)	Route table chain
	DC V(DMTSYSPT)	Switchable port chain
	DC V(DMTSYSTQ)	Tag slot queue
	DC V(DMTCOMVC)	Common routine vector
REXEND	DC V(DMTSYSND)	End of REX initial load
REXLOCKS	DC 0F'0'	Main REX wait list
	DC X'00',AL3(REQLOCK)	Request arrival synchronous lock address
	DC X'00',AL3(ATTNLOCK)	Console attention synchronous lock address
	DC X'40',AL3(REXCONSL)	Console I/O synchronous lock address
	DC X'80',AL3(PROGLOCK)	Program check synchronous lock address
ATTNLOCK	DC 0F'0',X'00',AL3(0)	Console attention lock
REXCONSL	DC F'0'	Synchronous lock for I/O operation
	DC AL2(CONSADDR)	CUU console device address
	DC AL1(1)	One byte requested on unit check
	DC AL1(TYP3210)	Console device type code
	DC A(0)	Channel program address to be filled in
	DC 2F'0'	SIO condition code and ending CSW return information
	DC X'00'	Sense return information on unit check
	DC 3X'00'	Reserved for IBM use
DMTREXCN	EQU REXCONSL	External name
CONSADDR	EQU X'009'	Default console cuu
REXREAD	CCW X'0A',REXIN ,SILI,L'REXIN	Reader operator response
REXWRITE	CCW X'09',REXOUT+2,SILI,0	Type a console message
REXIN	DC CL130' '	Console input buffer
REXMSG	DC CL11'M OP'	MSG command start
DMTREXID	EQU REXMSG+2	Transfer user identification
REXOUT	DC CL130' '	Output buffer
REXTAKE	DC 0F'0'	Request TAKE table
	DC CL4' '	Sending task name filled in by TAKE manager
	DC AL1(L'REXREQ),AL3(REXREQ)	Address and length of request buffer
	DC A(REXRESP)	Address of response buffer
REXREQ	DC XL140'00'	TAKE request buffer
REXRESP	DC AL1(0)	Never any response messages

SML MONITOR CONTROL AREA

The SML Monitor Control Area is used to define various constants, save areas, BSC control sequences, channel command words, and bit settings used during SML processing.

<u>Field Name</u>		<u>Field Description, Contents, Meaning</u>
CBUFFER	DC A(0)	Active communications buffer
CFCSTOUT	DC X'8FCF'	Last FCS transmitted to HASP
CFCSTSTD	DC X'88C1'	Standard FCS
FCSCTEMP	DC AL2(0)	FCS compare area
CTEMP	DC H'0'	Temporary storage
CMAXDUP	DC H'3'	Maximum repeated blocks
CECBCNTO	DC AL1(0)	First byte of halfword
	DC AL1(X'80')	Elock check count out
CECBCNTI	DC AL1(0)	Spacer
	DC AL1(X'80')	Elock count character expected
CEUFLAST	DC H'0'	
CREFLAST	DC 10X'00'	Save of start of last buffer
CRESP	DC AL1(0)	Response character received
CREGS	DS 3F	Register save area
CRETREGS	DS 3F	Save area
\$COMEXIT	DC A(\$START)	COMSUD initial entry point
CBCB	DC X'00'	Last ECB sent for reset
CSETBCB	DC X'00'	
	DS 0F	Force fullword alignment
CCSW	DC XL8'00'	Temporary storage for CSW
COLDRCB	DC X'00'	Last RCB sent
CUNITCMD	DC X'00'	Command code storage
CLASTCAW	DC F'0'	CCW address save
BUFSYNSW	DC X'00'	Buffer synchronization switch
	<u>Bits defined in BUFSYNSW</u>	
\$TPPNONE	EQU X'80'	Stop all buffering
OFLSW	EQU X'40'	Flush buffer
GDQBUFS	EQU X'20'	Stop dequeuing buffers
\$COMBUSY	EQU X'10'	Communications inactive
CUNWFAKE	EQU X'08'	Dummy read on for unit exception recovery
CACKSW	EQU X'04'	ACK received
ADAECB	DC F'0'	Synchronous lock
ADACUU	DC X'0000',AL1(1),AL1(TYP2700)	
AIDCWA	DC A(CCTCCW)	Adapter CCW addr
ADASIOCC	EQU *	SIO condition code
ADACSW	DC 2F'0'	Adapter ending CSW
ADASENSE	DC F'0'	Adapter sense byte
ALSAV	DC 8F'0'	\$SIO register save area
	<u>Control Sequences</u>	
XSTXSEQ	DC AL1(XLDR,XSTX)	Start-of-text sequence
XETBSEQ	DC AL1(XTRL,XETB)	End-of-text-block sequence
XACKSEQ	DC AL1(XDLE,XACK0)	Positive acknowledgement sequence
XNAKSEQ	DC AL1(XSYN,XNAK)	Negative acknowledgement sequence
XSYNSEQ	DC AL1(XSYN,XSYN,XSYN,XSYN)	Synchronization sequence

<u>Field Name</u>	<u>Channel Command Words</u>	<u>Field Description, Contents, Meaning</u>
	<u>Normal Data Write with Return Data Read</u>	
CCWS	CCW 1, XSYNSEQ, CD+SILI, 4	Synchronization sequence
CCWA	CCW 1, 0, CC+SILI, 0	Write buffer
CCWB	CCW 1, XETBSEQ, CC+SILI, 2	Write ending sequence
CCWC	CCW 2, 0, SILI, 0	Read return data
	<u>Dummy Read To Turn Off Lost Data Sense</u>	
CCWD	CCW 2, 0, SILI*SKIP, 65000	
	<u>DISABLE command</u>	
CCWOFF	CCW X'2F', 0, SILI, 1	Disable
WRITE	EQU X'01'	Adapter write command code
READ	EQU X'02'	Adapter read command code
NOP	EQU X'03'	Adapter NOP command code
SENSE	EQU X'04'	Adapter sense command code
SENSE	EQU X'04'	Adapter sense command code
DISABLE	EQU X'2F'	Adapter disable command code

Appendix C. RSCS Request Elements

This appendix provides information on the format and use of RSCS request elements. These elements are used by RSCS tasks in task-to-task communication.

The information provided includes:

- The name of the module that builds the element
- The function performed by the element
- A brief description of the element's usage
- The format of the element
- Any operational notes that might be useful in understanding how the element is used

COMMAND ALERT ELEMENT FORMAT A1

BUILT BY: DMTCMX

FUNCTION: Execute an AXS command

DESCRIPTION: This ALERT element is passed via ALERT to the AXS task (DMTAXS) to request second-level processing of ORDER and PURGE commands.

0	Length (n-1)	Function Code: X'10',X'11'	Response Code	Modifiers
4		linkid		
C	spcolid count (n-X'E')/2		spoolid	
10				
		spoolid	spoolid	

OPERATIONAL NOTES

The linkid field specifies the affected link. The spoolid fields are binary halfwords and specify the files enqueued on the specified link which are to be reordered or purged. The spoolid count field is a binary halfword and specifies the total number of spoolid fields present. The meanings of the other fields follow.

ORDER Command

Function Code: X'10'

Response Codes:

- X'00' Element accepted for processing
- X'10' Element rejected, busy

Modifiers:

- X'80' Response messages go to local RSCS operator
- X'00' Response messages go to specified link.

PURGE Command

Function Code: X'11'

Response Codes:

- X'00' Element accepted for processing
- X'10' Element rejected, busy

Modifiers:

- X'80' Response messages go to local RSCS operator
- X'40' Purge all files enqueued on the specified link
- X'00' Purge only specified files, response messages go to specified link

COMMAND ALERT ELEMENT FORMAT A2

BUILT BY: DMTCMX

FUNCTION: Execute AXS command

DESCRIPTION: This ALERT element is passed via ALERT to the AXS task (DMTAXS) to request second-level processing of CHANGE commands.

0	Length (X'33')	Function Code: X'20'	Response Code	Modifiers
4	linkid			
C	spoolid		priority	
10	HOLD	CLASS	COPY	
14	Distribution Code			
1C	filename/filetype, dsname			

OPERATIONAL NOTES

The linkid field specifies the link on which the object inactive file is enqueued. The spoolid field is a binary halfword and specifies the object file's VM/370 RSCS identifier.

The following fields are specified only when the corresponding file attribute is to be changed. If the field is not specified, it is set to all 1 bits (X'FF...').

- Priority halfword contains binary priority 0-99
- HOLD has the following: X'7F' -- set hold status (HCLD)
X'3F' -- reset hold status (NOHOLD)
- CLASS 1-byte EBCDIC class, A-Z, 0-9
- COPY halfword binary copy count, 1-99
- Distribution code 8-byte EBCDIC spool file distribution code
- Filename/filetype, dsname, 24-byte EBCDIC spool file filename or filetype or dsname

The meanings of the other fields follow.

CHANGE Command

Function Code: X'20'

Response Codes:

- X'00' Element accepted for processing
- X'10' Element rejected, busy

Modifiers:

- X'80' Response messages go to local RSCS operator
- X'00' Response messages go to specified link

COMMAND ALERT ELEMENT FORMAT L0

BUILT BY: DMTCMX

FUNCTION: Execute a line driver command

DESCRIPTION: This ALERT element is passed via ALERT to a line driver task (DMTNPT, DMTSML) to request second-level processing of START, DRAIN, FREE, HOLD, and TRACE commands.

0	Length (X'0B')	Function Code: X'80,X'81', X'82',X'83',X'84'	Response Code	Modifiers
4	locid			

OPERATIONAL NOTES

The locid specifies the location that is to receive response messages. The meanings of the other fields follow.

START Command

Function Code: X'80'

Response Codes:

- X'00' Element accepted for processing
- X'10' Element rejected, busy

Modifiers:

- X'80' Start updated classes
- X'00' Reset DRAIN status

DRAIN Command

Function Code: X'81'

Response Codes:

- X'00' Element accepted for processing
- X'10' Element rejected, busy

Modifiers: Unused

FREE Command

Function Code: X'82'

Response Codes:

X'00' Element accepted for processing
X'10' Element rejected, busy

Modifiers: Unused

HOLD Command

Function Code: X'83'

Response Codes:

X'00' Element accepted for processing
X'10' Element rejected, busy

Modifiers:

X'80' HOLD Immediate
X'00' HOLD after file processing

TRACE Command

Function Code: X'84'

Response Codes:

X'00' Element accepted for processing
X'10' Element rejected, busy

Modifiers:

X'C0' TRACE end
X'80' TRACE errors
X'00' TRACE all

COMMAND ALERT ELEMENT FORMAT L1

BUILT BY: DMTCMX

FUNCTION: Execute a line driver command

DESCRIPTION: This ALERT element is passed via ALERT to a line driver task (DMTNPT, DMTSML) to request second-level processing of BACKSPAC and FWDSpace commands.

0	Length (X'0F')	Function Code: X'90', X'91'	Response Code	Modifiers
4		locid		
C		Count		

OPERATIONAL NOTES

The locid specifies the location that is to receive response messages. The count field is a binary fullword, and specifies the number of units to be backspaced or forwardspaced. The meanings of the other fields follow.

BACKSPAC Command

Function Code: X'90'

Response Codes:

- X'00' Element accepted for processing
- X'10' Element rejected, busy

Modifiers:

- X'80' Backspace count
- X'00' Backspace file (restart)

FWDSpace Command

Function Code: X'91'

Response Codes:

- X'00' Element accepted for processing
- X'10' Element rejected, busy

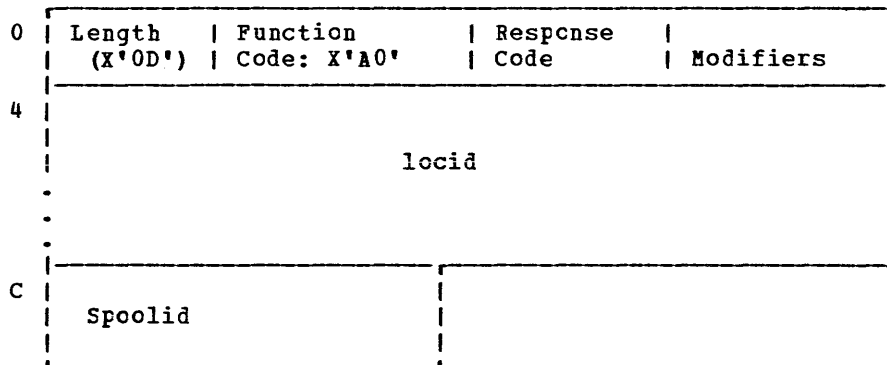
Modifiers: Unused

COMMAND ALERT ELEMENT FORMAT L2

BUILT BY: DMTCMX

FUNCTION: Execute a line driver command

DESCRIPTION: This ALERT element is passed via ALERT to a line driver task (DMTWT, DMTSML) to request second-level processing of FLUSH commands.



OPERATIONAL NOTES

The locid specifies the location that is to receive response messages. The spoolid field is a binary halfword, and specifies the VM/370 RSCS identifier of the active file to be flushed. The meanings of the other fields follow.

FLUSH Command

Function Code: X'A0'

Response Codes:

- X'00' Element accepted for processing
- X'10' Element rejected, busy

Modifiers:

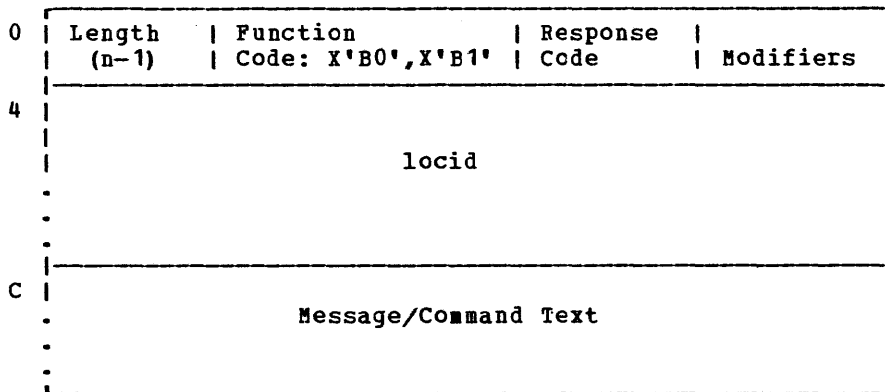
- X'80' Flush all copies, purge file
- X'40' Flush hold, keep file, do not decrement copy count
- X'00' Flush, decrement copy count, purge file if no copy count remains

COMMAND ALERT ELEMENT FORMAT L3 (ALSO MESSAGE ALERT ELEMENT)

BUILT BY: DMTCHX, DMTMGX

FUNCTION: Execute a line driver command

DESCRIPTION: This ALERT element is passed via ALERT to a line driver task (DMTNPT, DMTSML) to forward messages, and to request second-level processing of CMD commands.



OPERATIONAL NOTES

The locid specifies the location that is to receive the message or command text. The meanings of the other fields follow.

CMD Command

Function Code: X'B0'

Response Codes:

- X'00' Element accepted for processing
- X'10' Element rejected, busy

Modifiers: None

MSG Command

Function Code: X'B1'

Response Codes:

- X'00' Element accepted for processing
- X'10' Element rejected, busy

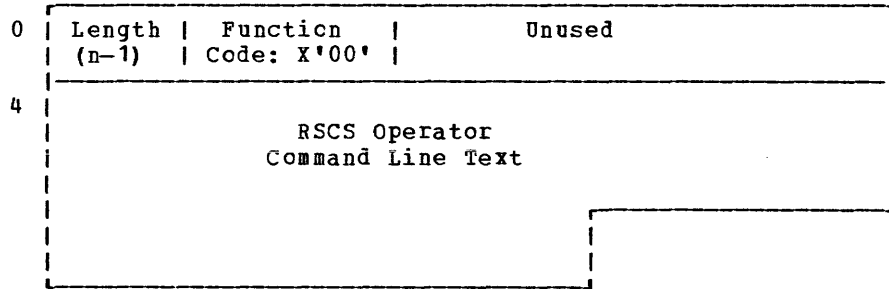
Modifiers: One-byte binary RSCS severity code

COMMAND REQUEST ELEMENT

BUILT BY: DMTNPT, DMTSML

FUNCTION: Execute an RSCS operator command

DESCRIPTION: This request element is passed by a line driver via GIVE/TAKE to the REX task in response to a command entry at a remote station.



OPERATIONAL NOTES

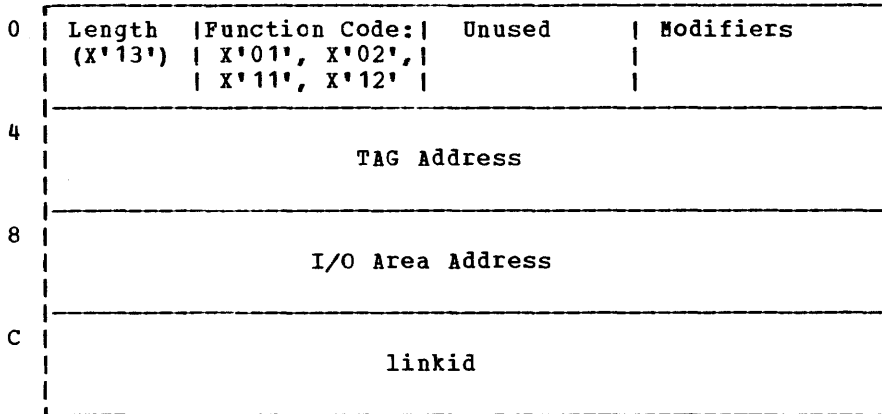
No response text is returned. Command responses are distributed via DMTMGX.

FILE REQUEST ELEMENT

BUILT BY: DMTNPT, DMTSML

FUNCTION: Initiates or terminates processing of an input or output file.

DESCRIPTION: This request element is passed via GIVE/TAKE to the AXS task by line drivers to effect local spool file access during communications with a remote station.



OPERATIONAL NOTES

The use and meaning of the various fields depends on the requested function, as described below. Certain fields may be updated during request processing. The (updated) file request element is returned to the requestor as a GIVE response. The meanings of the other fields follow.

Open Input

Function Code: X'01'

Modifiers: Unused

Tag Address: Response field that points to the opened file's active TAG in DMTSYS

I/O Area Address: Response field that points to a virtual page buffer containing the opened file's first VM/370 spool data buffer.

linkid: Request field that specifies the requesting line driver's linkid.

Response Post Codes:

- X'08' Terminal system error
- X'04' No file available
- X'02' Undefined linkid
- X'01' Previously open file returned

Open Output

Function Code: X'11'

Modifiers: X'80' Do not return possible previously opened file

Tag Address: Request field which points to a prototype file TAG for the output file, constructed by the calling line driver.

I/O Area Address: Response field which points to a virtual page buffer containing an I/O table, a write CCW, and a buffer for processing the output file.

linkid: Request field which specifies the requesting line driver's linkid.

Response Post Codes:

X'04' Error, file not opened

X'02' Undefined linkid

X'01' Previously open file returned

Close Input

Function Code: X'02'

Modifiers:

X'80' Do not purge copy or file

X'40' Purge all copies, and purge file

Tag Address: Request field which points to the file's active TAG in DMTSYS, as supplied by open input.

I/O Area Address: Unused

linkid: Unused

Response Post Codes:

X'04' TAG not found, close failed

Close Output

Function Code: X'12'

Modifiers: Unused

Tag Address: Request field which points to a prototype file TAG for the output file, constructed by the calling line driver. This TAG is used to update the parameters to be set for the output file.

I/O Area Address: Request field which points to the file's I/O area, as supplied by open output.

linkid: Unused

Response Post Codes:

X'04' I/O area not found, close failed

LINE ALERT ELEMENT

BUILT BY: DMTCMX

FUNCTION: Request line port allocation

DESCRIPTION: This ALERT element is passed via ALERT to the LAX task (DMTLAX) to verify and reserve line ports for links being activated in response to a START command.

0	Length (X'0F')	Function Code: X'01'	Response Code	Unused
4	Line Address			Unused
8	linkid			

OPERATIONAL NOTES

Certain fields are updated during processing. The meanings of the fields follow.

Response Codes:

- X'08' Specified line address not attached (CC=3)
- X'04' Specified line address not BSC port device type
- X'02' Line not available

Line Address: Request field specifying requested line address. Zero specification implies request for allocation of a switchable line from the port table. If successful, the port's line address is returned in this field as a response.

linkid: Response field specifying the ID of the link that has reserved the particular requested line address (with response code X'02').

MESSAGE REQUEST ELEMENT

BUILT BY: DMTREX, DMTCMX, DMTAXS, DMTNPT, DMISML

FUNCTION: Issue an RSCS message

DESCRIPTION: This request element is passed via GIVE/TAKE to the REX task, to specify the construction and distribution of an RSCS message (by DMTMGX).

0	Length (n-1)	Function Code: X'02'	Routing Code	Severity Code
4	Receiver locid			
C	Receiver userid			
14	Issuing Module Code			Action Code
18	Binary Message Number		Unused	
1C	8-byte Variable Substitution Values for Message Text			

OPERATIONAL NOTES

The routing code and severity code from the message definition (in DMTMSG) are used when not supplied in the message request element. If the message is not defined in DMTMSG, it is constructed using the specifications in the message request element, and the "variable substitution values" become the message text, unmodified.

Routing codes:

- X'80' Local RSCS console
- X'40' Remote addressee
- X'20' Local user
- X'10' Local VM/370 operator

No response text is returned.

PORT TABLE

BUILT BY: Assembly of DMTSYS at RSCS generation

FUNCTION: Record allocation status of switchable line ports available to RSCS

DESCRIPTION: The first doubleword of the table is reserved for control information. Each following halfword contains the virtual device address of a line port which may be dialed, and which is available to RSCS.

0	Number of Line Port Entries in Table	
4		
8	Virtual Line Address	Virtual Line Address
C	Virtual Line Address	Virtual Line Address
10		
.	.	.
.	.	.
.	.	.
	Virtual Line Address	Virtual Line Address

OPERATIONAL NOTES

The line port entries are marked "in use" by setting the high-order four bits of the entries to 1's.

TERMINATE REQUEST ELEMENT

BUILT BY: DMTNPT, DMTSML

FUNCTION: Terminate line driver task

DESCRIPTION: This request element is passed via GIVE/TAKE to the REX task, to terminate line driver operation in response to a DRAIN command.

0	Length	Function	
	(1)	Code: X'03'	

OPERATIONAL NOTES

There are no error conditions for the terminate function, so no response is made. However, line driver tasks must issue a WAIT request following a call to GIVE for terminate, because REX may not execute the request immediately.

Appendix D. CMS Equate Symbols

This Appendix contains Assembler language equate symbols used in CMS to reference data for:

- CMS usage
- CMS registers

CMS USAGE EQUATES

Field Name	Field Description, Contents, Meaning		
<u>Bits defined in the Program Status Word (PSW)</u>			
CHAN0	EQU	X'80'	Bit 00 - channel 0 mask
CHAN1	EQU	X'40'	Bit 01 - channel 1 mask
CHAN2	EQU	X'20'	Bit 02 - channel 2 mask
CHAN3	EQU	X'10'	Bit 03 - channel 3 mask
CHAN4	EQU	X'08'	Bit 04 - channel 4 mask
CHAN5	EQU	X'04'	Bit 05 - channel 5 mask
CHANM	EQU	X'02'	Bit 06 - input/output mask
EXTM	EQU	X'01'	Bit 07 - external mask
ECMM	EQU	X'08'	Bit 12 - extended control mode mask
MCKM	EQU	X'04'	Bit 13 - machine check mask
WAIT	EQU	X'02'	Bit 14 - wait state mask
PROB	EQU	X'01'	Bit 15 - problem state mask
FOFM	EQU	X'08'	Bit 36 - fixed-point overflow mask
DOPM	EQU	X'04'	Bit 37 - decimal overflow mask
EUFM	EQU	X'02'	Bit 38 - exponent underflow mask
SIGM	EQU	X'01'	Bit 39 - significance mask
<u>Bits defined in the Channel Status Word (CSW)</u>			
ATTN	EQU	X'80'	Bit 32 - attention
SM	EQU	X'40'	Bit 33 - status modifier
CUE	EQU	X'20'	Bit 34 - control unit end
BUSY	EQU	X'10'	Bit 35 - busy
CE	EQU	X'08'	Bit 36 - channel end
DE	EQU	X'04'	Bit 37 - device end
UC	EQU	X'02'	Bit 38 - unit check
UE	EQU	X'01'	Bit 39 - unit exception
PCI	EQU	X'80'	Bit 40 - program-controlled interrupt
ICL	EQU	X'40'	Bit 41 - incorrect length
PGC	EQU	X'20'	Bit 42 - program check
PTC	EQU	X'10'	Bit 43 - protection check
CDC	EQU	X'08'	Bit 44 - channel data check
CCC	EQU	X'04'	Bit 45 - channel control check
ICC	EQU	X'02'	Bit 46 - interface control check
CHC	EQU	X'01'	Bit 47 - chaining check
<u>Common Channel Command Codes</u>			
WRITE	EQU	X'01'	Write
READ	EQU	X'02'	Read
NOP	EQU	X'03'	No operation
SENSE	EQU	X'04'	Sense
WRDATA	EQU	X'05'	Write data
RIDATA	EQU	X'06'	Read data
SPEK	EQU	X'07'	Seek
TIC	EQU	X'08'	Transfer in channel
WRITE1	EQU	X'09'	Write and space 1
RDCONS	EQU	X'0A'	Read from console
SPTSEC	EQU	X'23'	Set sector
SPARCH	EQU	X'31'	Search ID equal
<u>Bits defined in a Channel Command Word (CCW)</u>			
CD	EQU	X'80'	Bit 32 - chain data
CC	EQU	X'40'	Bit 33 - command chain
SILI	EQU	X'20'	Bit 34 - suppress incorrect length
SKIP	EQU	X'10'	Bit 35 - suppress data transfer
PCIF	EQU	X'08'	Bit 36 - cause program control interrupt
IDA	EQU	X'04'	Bit 37 - indirect data address

CMS REGISTER EQUATES

<u>Field</u> <u>Name</u> -----			<u>Field Description, Contents, Meaning</u> -----
<u>General purpose registers</u>			
R0	EQU	0	
R1	EQU	1	
R2	EQU	2	
R3	EQU	3	
R4	EQU	4	
R5	EQU	5	
R6	EQU	6	
R7	EQU	7	
R8	EQU	8	
R9	EQU	9	
R10	EQU	10	
R11	EQU	11	
R12	EQU	12	
R13	EQU	13	
R14	EQU	14	
R15	EQU	15	
<u>Floating-point registers</u>			
F0	EQU	0	
F2	EQU	2	
F4	EQU	4	
F6	EQU	6	
<u>Extended control registers</u>			
C0	EQU	0	
C1	EQU	1	
C2	EQU	2	
C3	EQU	3	
C4	EQU	4	
C5	EQU	5	
C6	EQU	6	
C7	EQU	7	
C8	EQU	8	
C9	EQU	9	
C10	EQU	10	
C11	EQU	11	
C12	EQU	12	
C13	EQU	13	
C14	EQU	14	
C15	EQU	15	

Appendix E. Data Areas and Control Block References

This appendix -- a listing of CP, CMS, and RSCS control blocks -- contains the following:

- Module references to data areas and control blocks.
- Information on how certain data areas or control blocks are created and released.

CP CONTROL BLOCK REFERENCES

ACCTBLOK

Built by: DMKHVD

Released by: DMKHVD, DMKUSO

Referenced by: DMKACO, DMKCKP, DMKHVD, DMKSPL

ACNTBLOK

Built by: DMKACO, DMKHVD, DMKWRM

Released by: DMKACO

Referenced by: DMKACO, DMKCKP, DMKHVD, DMKJRL, DMKRSE, DMKWRM

ALOCBLOK

Built by: DMKCPI, DMKVDC

Released by: DMKCPI, DMKVDC

Referenced by: DMKCPI, DMKMON, DMKPGT, DMKTDK, DMKVDC

BSCBLOK

Built by: DMKRGB

Released by: DMKRG

Referenced by: DMKBSC, DMKRG, DMKRGB

BUFFER

Built by: DMKCFM, DMKCPI, DMKERM, DMKGRF, DMKLNK, DMKLOG, DMKRG, DMKRSP

Released by: DMKCFM, DMKCPI, DMKGRF, DMKLNK, DMKRG, DMKRSP

Referenced by: DMKALG, DMKCDM, DMKCFG, DMKCFM, DMKCFO, DMKCFS, DMKCPI, DMKCPS, DMKCSB, DMKCSO, DMKCSP, DMKCSQ, DMKCST, DMKCSU, DMKCSV, DMKEMA, DMKERM, DMKGRF, DMKGRT, DMKLNK, DMKMSG, DMKNMT, DMKRG, DMKEND, DMKRSP, DMKSCN, DMKUDU, DMKVDC, DMKWRM

CCHREC

Built by: DMKCCH

Released by: DMKCCH, DMKIOE, DMKIOF

Referenced by: DMKCCH, DMKEIG, DMKSEV, DMKSIX

CCPARM

Built by: DMKNLD, DMKSNC

Released by: DMKNLD, DMKSNC

Referenced by: DMKNLD, DMKSNC

CHXBLOK

Built by: DMKDIA

Released by: DMKVCA

Referenced by: DMKCFP, DMKCQG, DMKDIA, DMKVCA, DMKVSI

CHYBLOK

Built by: DMKDIA

Released by: DMKVCA

Referenced by: DMKDIA, DMKVCA

CKPBLOK

Built by: DMKRNH

Released by: DMKRNH

Referenced by: DMKRNH, DMKWRM

CONTASK

Built by: DMKCNS, DMKGRF, DMKQCN, DMKRG, DMKRGB, DMKRNH

Released by: N/A

Referenced by: DMKCNS, DMKGRF, DMKMON, DMKNES, DMKQCN, DMKRG, DMKRGB, DMKRNH

CORTABLE

Assembled in DMKSYS.

Released by: N/A

Referenced by: DMKACO, DMKATS, DMKBLD, DMKCCW, DMKCDS, DMKCFO, DMKCPI, DMKCPU, DMKCPV, DMKDGD, DMKDMP, DMKFRE, DMKMCC, DMKMCH, DMKMNI, DMKPAG, DMKPGS, DMKPSA, DMKPTR, DMKRPA, DMKUDR, DMKUDU, DMKUNT, DMKVMA

CPEXBLOK

Built by:

DMKACO, DMKCDS, DMKCFM, DMKCPV, DMKDIA, DMKGRF, DMKIOE, DMKIOF, DMKIOS, DMKLOC, DMKMCC, DMKMCH, DMKMON, DMKPGT, DMKPTR, DMKQCN, DMKRG, DMKRGB, DMKRNH, DMKRPA, DMKRSP, DMKSPL, DMKSVC, DMKUSO, DMKVCA, DMKVDC, DMKVDE, DMKVMA, DMKVMC

Released by: DMKCPV, DMKDSP, DMKIOF, DMKMON, DMKPTR

Referenced by: DMKACO, DMKALG, DMKCCW, DMKCDS, DMKCFM, DMKCFO, DMKCFP, DMKCNV, DMKCPB, DMKCPV, DMKCPU, DMKCPV, DMKDGD, DMKDIA, DMKDSB, DMKDSP, DMKEXT, DMKFRE, DMKGIO, DMKGRF, DMKIOE, DMKIOF, DMKIOS, DMKLNK, DMKLOC, DMKMCC, DMKMCD, DMKMCH, DMKMCT, DMKMI, DMKMID, DMKMNI, DMKMON, DMKPAG, DMKPGS, DMKPGT, DMKPRG, DMKPRV, DMKPSA, DMKPTR, DMKQCN, DMKRG, DMKRGB, DMKRNH, DMKRPA, DMKRSP, DMKSPL, DMKSSS, DMKSTK, DMKSVC, DMKTAP, DMKTMR, DMKTRD, DMKUNT, DMKUSO, DMKVAT, DMKVCA, DMKVDA, DMKVDC, DMKVDE, DMKVMA, DMKVMC, DMKVSI, DMKVSP

DDRREC

Built by: DMKVER

Released by: DMKVER

Referenced by: DMKVER

DMPINREC

Built by: DMKDMP

Released by: DMKDMP

Referenced by: DMKDMP

DMPKYREC

Built by: DMKDMP

Released by: DMKDMP

Referenced by: DMKDMP

DMPTBREC

Built by: DMKDMP

Released by: DMKDMP

Referenced by: DMKDMP

ECBLOK

Built by: DMKBLD

Released by: DMKCFO, DMKCFV, DMKUSO

Referenced by: DMKBLD, DMKDB, DMKCDM, DMKCDS, DMKCFG, DMKCFH, DMKCFP, DMKCFV, DMKDSP, DMKEXT, DMKPRG, DMKPRV, DMKSCH, DMKSVC, DMKTMR, DMKTRC, DMKTRD, DMKUSO, DMKVAT, DMKVMC

ERRBLOK

Built by: DMKIOE

Released by: DMKIOF

Referenced by: DMKIOE, DMKIOF

IOBLOK

Built by: DMKACO, DMKCCW, DMKCFP, DMKCNV, DMKCPB, DMKCPV, DMKCSO, DMKCSU, DMKDGD, DMKDIA, DMKGIO, DMKGRF, DMKHVC, DMKIOS, DMKNLD, DMKRG, DMKRGB, DMKSPL, DMKTDK, DMKVCA, DMKVDC, DMKVDD, DMKVDE, DMKVDR, DMKVIO

Released by: DMKCFP, DMKCNV, DMKCPB, DMKCPV, DMKCSO, DMKCSU, DMKDAS, DMKDGD, DMKDIA, DMKGIO, DMKGRF, DMKHVC, DMKIOS, DMKMON, DMKNLD, DMKPAG, DMKRG, DMKRGB, DMKRNH, DMKRSP, DMKSEP, DMKTDK, DMKVCA, DMKVDC, DMKVDD, DMKVDE, DMKVIO

Referenced by: DMKACO, DMKBSC, DMKCCW, DMKCCW, DMKCFP, DMKCNV, DMKCPB, DMKCPV, DMKCSO, DMKCSB, DMKCSU, DMKCSU, DMKCSV, DMKDAS, DMKDGD, DMKDIA, DMKDIB, DMKDSB, DMKDSP, DMKGIO, DMKGRF, DMKHVC, DMKIOE, DMKIOG, DMKIOS, DMKISM, DMKLOG, DMKMCC, DMKMNI, DMKMON, DMKMSW, DMKNLD, DMKNLE, DMKPAG, DMKPGT, DMKRG, DMKRGB

DMKRNH, DMKRSE, DMKRSP, DMKSEP, DMKSPL,
DMKSSS, DMKSTK, DMKTAP, DMKTCS, DMKTDK,
DMKTRC, DMKTRD, DMKTRK, DMKUDR, DMKUNT,
DMKUSO, DMKVCA, DMKVDC, DMKVDD, DMKVDE,
DMKVDR, DMKVIO, DMKVSI

MCRECORD

Built by: DMKMCH

Released by: N/A

Referenced by: DMKMCH

IOERBLOK

Built by: DMKBSC, DMKCCH, DMKDAS,
DMKDIA, DMKDIB, DMKIOE, DMKIOS, DMKRSE,
DMKTAP, DMKVCA

MDRREC

Built by: DMKVER

Released by: DMKBSC, DMKCCH, DMKCCW,
DMKCFP, DMKCNS, DMKCPS, DMKDAS, DMKDGD,
DMKDIA, DMKDIB, DMKGIO, DMKGRF, DMKIOE,
DMKIOS, DMKMON, DMKNLD, DMKRG, DMKRGB,
DMKRNH, DMKRSE, DMKRSR, DMKTAP, DMKVIO

Released by: DMKVER

Referenced by: DMKIOF, DMKVER

Referenced by: DMKBSC, DMKCCH, DMKCCW,
DMKCFP, DMKCNS, DMKCPS, DMKDAS, DMKDGD,
DMKDIA, DMKDIB, DMKDSB, DMKEIG, DMKGIO,
DMKGRF, DMKIOE, DMKIOF, DMKIOS, DMKMSW,
DMKNLD, DMKNLE, DMKRG, DMKRGB, DMKRNH,
DMKRSE, DMKRSP, DMKSEV, DMKSIX, DMKTAP,
DMKTRK, DMKUNT, DMKVCA, DMKVDC, DMKVDE,
DMKVIO, DMKVSI

MICBLOK

Built by: DMKCF, DMKLOG

Released by: DMKCF, DMKLOG, DMKUSO

Referenced by: DMKELD, DMKCF, DMKDSP,
DMKLOG, DMKMCH, DMKPTR, DMKRPA, DMKTRA

IRMBLOK

Built by: DMKCFO, DMKCF

Released by: DMKCF, DMKIOE

Referenced by: DMKCFO, DMKIOE

MIHREC

Built by: DMKVER

Released by: DMKVER

Referenced by: DMKVER

JPSCBLOK

Assembled as part of DMKSYS

Referenced by: DMKALG, DMKJRL, DMKLNK,
DMKLOG

MNDEVLST

Built by: DMKENT

Released by: DMKENT

Referenced by: DMKENT

LOCKBLOK

Built by: DMKLOC

Released by: DMKLOC

Referenced by: DMKLOC

MNHDR

Built by: DMKMON

Released by: DMKMON

Referenced by: DMKMON

MCHAREA

Built by: DMKIOG

Released by: N/A

Referenced by: DMKCCH, DMKCFO, DMKCPU,
DMKIOG, DMKMCH, DMKMCT

MN000

Built by: DMKMON

Released by: DMKMON

Referenced by: DMKMON

MN001

Built by: DMKMON
Released by: DMKMON
Referenced by: DMKMON

MN097

Built by: DMKMNI
Released by: DMKMON
Referenced by: DMKMNI

MN098

Built by: DMKMNI
Released by: DMKMON
Referenced by: DMKMNI

MN099

Built by: DMKMON
Released by: DMKMON
Referenced by: DMKMON

MN10X

Built by: DMKMON
Released by: DMKMON
Referenced by: DMKMON

MN20X

Built by: DMKMON
Released by: DMKMON
Referenced by: DMKMON

MN400

Built by: DMKMON
Released by: DMKMON
Referenced by: DMKMON

MN500

Built by: DMKMON
Released by: DMKMON
Referenced by: DMKMON

MN600

Built by: DMKMON, DMKMNI
Released by: DMKMON
Referenced by: DMKMNI, DMKMON

MN602

Built by: DMKENT
Released by: DMKENT
Referenced by: DMKENT

MN700

Built by: DMKMON
Released by: DMKMON
Referenced by: DMKMON

MN802

Built by: DMKMON
Released by: DMKMON
Referenced by: DMKMON

MONCOM

Built by: DMKMCC
Released by: DMKMON
Referenced by: DMKCPS, DMKDMP, DMKENT,
DMKMCC, DMKMCD, DMKMIA, DMKMNI, DMKMON

MSSCOM

Built by: DMKSSS
Released by: DMKLNK, DMKLOG, DMKSSS,
DMKVDA
Referenced by: DMKCPB, DMKDGD, DMKDSB,
DMKLNK, DMKLOG, DMKMSS, DMKSSS, DMKVDA,
DMKVSI

SAVTABLE

Assembled into CP pageable module DMKSNT
Released by: N/A
Referenced by: DMKCFG, DMKCFH,

SDRBLOK

Built by: DMKIOF
Released by: DMKIOE
Referenced by: DMKIOE, DMKIOF

SEGTABLE

Built by: DMKBLD
Released by: DMKBLD
Referenced by: DMKATS, DMKBLD, DMKPGS, DMKVMA

SFBLOK

Built by: DMKCKS, DMKNLD, DMKSPL, DMKWRM
Released by: DMKCKS, DMKRSP, DMKSPL, DMKUSO
Referenced by: DMKCKP, DMKCKS, DMKCPI, DMKCQG, DMKCQR, DMKCSO, DMKCSP, DMKCSQ, DMKCST, DMKCSU, DMKCSV, DMKDMP, DMKDRD, DMKMIA, DMKMNI, DMKNLE, DMKRSE, DMKRSP, DMKSEP, DMKSPL, DMKTCS, DMKUSO, DMKVSP, DMKVSQ, DMKWRM

SHQBLOK

Built by: DMKCSP, DMKWRM
Released by: DMKCSP
Referenced by: DMKCKS, DMKCQR, DMKCSQ, DMKSPL, DMKWRM

SHRTABLE

Built by: DMKCFG
Released by: DMKPGS, DMKVMA
Referenced by: DMKATS, DMKCFG, DMKCFH, DMKCPU, DMKPGS, DMKPTR, DMKVMA

SPLINK

Built By: N/A
Released by: N/A
Referenced by: DMKCKS, DMKCQH, DMKCSU, DMKDRD, DMKMIA, DMKRSP, DMKSPL, DMKTCS, DMKVSP, DMKVSQ

SWPTABLE

Built by: DMKBLD, DMKVMA
Released by: DMKBLD
Referenced by: DMKATS, DMKBLD, DMKCFG, DMKCPU, DMKPGS, DMKPTR, DMKVAT, DMKVMA

SYSLOCS

Assembled into CP nucleus module DMKSYS.
Referenced by: DMKACC, DMKBLD, DMKCFO, DMKCFT, DMKCKP, DMKLOC, DMKLOG, DMKLOH, DMKUDR, DMKUDU, DMKUSO

SYSTBL

Assembled into DMKSNT.
Referenced by: DMKATS, DMKCFG, DMKCFH, DMKCPU

TNSREC

Built by: DMKIOF
Released by: DMKIOF
Referenced by: DMKIOF

TREXT

Built by: DMKTRA
Released by: DMKTRA, DMKTRC, DMKUSO
Referenced by: DMKCFH, DMKDSP, DMKPGS, DMKPRG, DMKPRV, DMKSVC, DMKTMR, DMKTRA, DMKTRC, DMKTRD, DMKVIC

TRQBLOK

Built by: DMKBLD, DMKCFC, DMKCFS,
DMKCPI, DMKGRF, DMKLOG, DMKMCC, DMKQCN,
DMKRG

Released by: DMKCFM, DMKCFS, DMKDIA,
DMKMCC, DMKLOG, DMKMON, DMKQCN, DMKRG,
DMKUSO

Referenced by: DMKBLD, DMKCD, DMKCFC,
DMKCFM, DMKCFP, DMKCFS, DMKCPI, DMKCPU,
DMKDIA, DMKDSP, DMKENT, DMKGRF, DMKLOG,
DMKMCC, DMKMID, DMKMNI, DMKMON, DMKPSA,
DMKQCN, DMKRG, DMKRGB, DMKSCH, DMKSSS,
DMKTMR, DMKUSO

UDBFBLOK

Built by: DMKDEF, DMKHVD, DMKSPL

Released by: DMKDEF, DMKHVD, DMKSPL

Referenced by: DMKCFS, DMKDEF, DMKHVD,
DMKLNK, DMKLOG, DMKSPL, DMKSSS, DMKUDR,
DMKUDU

UDEVBLOK

Built by: DMKCSP, DMKUDR

Released by: DMKCSP, DMKUDR

Referenced by: DMKDEF, DMKDIR, DMKLNK,
DMKLOG, DMKSCN, DMKUDR, DMKVDA, DMKVDS

UDIRBLOK

Built by: DMKCSP

Released by: DMKCSP

Referenced by: DMKCFS, DMKCPI, DMKCSP,
DMKDEF, DMKDIR, DMKHVD, DMKLNK, DMKLOG,
DMKSPL, DMKUDR, DMKUDU

UMACBLOK

Built by: DMKDIR

Released by: DMKDIR

Referenced by: DMKCFS, DMKDEF, DMKDIR,
DMKHVD, DMKLOG, DMKSPL, DMKUDR, DMKUDU

VCHBLOK

Built by: DMKVDS

Released by: DMKUSO

Referenced by: DMKCFM, DMKCFP, DMKCKP,
DMKCPB, DMKCPV, DMKCOG, DMKCSP, DMKCSU,
DMKDEF, DMKDIA, DMKDSP, DMKCSV, DMKLNK,
DMKLOG, DMKPRV, DMKSCN, DMKSPL, DMKSSS,
DMKUSO, DMKVCH, DMKVCN, DMKVDA, DMKVDC,
DMKVDD, DMKVDS, DMKVIC, DMKVI, DMKVSP

VCCNCTL

Built by: DMKVDS

Released by: DMKVDR

Referenced by: DMKALG, DMKCFP, DMKGRF,
DMKRG, DMKVCN, DMKVDR

VCUBLOK

Built by: DMKVDS

Released by: DMKUSO

Referenced by: DMKCFM, DMKCFP, DMKCKP,
DMKCPB, DMKCPV, DMKCOG, DMKCSP, DMKCSU,
DMKDEF, DMKDIA, DMKDSP, DMKLOG,
DMKLNK, DMKPRV, DMKSCN, DMKSPL, DMKSSS,
DMKUSO, DMKVCH, DMKVCN, DMKVDA, DMKVDC,
DMKVDD, DMKVDS, DMKVIC, DMKVI, DMKVSP

VDEVBLOK

Built by: DMKLOG, DMKVDS

Released by: DMKUSO

Referenced by: DMKACO, DMKALG, DMKCCH,
DMKCCW, DMKCFG, DMKCFH, DMKCFM, DMKCFP,
DMKCKP, DMKCPB, DMKCPV, DMKCOG, DMKCSQ,
DMKCSB, DMKCS, DMKCSQ, DMKCS, DMKCS,
DMKCSU, DMKCSV, DMKDS, DMKDEF, DMKDG,
DMKDIA, DMKDIB, DMKDRD, DMKDSP, DMKGIO,
DMKGRF, DMKHVC, DMKHVD, DMKIOS, DMKLNK,
DMKLOG, DMKLNK, DMKPRV, DMKQCN, DMKRG,
DMKSCN, DMKSPL, DMKSSS, DMKTHI, DMKTRC,
DMKTRD, DMKTRK, DMKUNT, DMKUSO, DMKVC,
DMKVCH, DMKVCN, DMKVDA, DMKVDC, DMKVDD,
DMKVDR, DMKVDS, DMKVER, DMKVIO, DMKVI,
DMKVSP, DMKVSQ

VFCBBLOK

Built by: DMKCFG, DMKCSO

Released by: DMKVDR

Referenced by: DMKCSB, DMKVSP

VMABLOK

Built by: DMKBLD, DMKCFG
Released by: DMKBLD, DMKPGS, DMKVMA
Referenced by: DMKATS, DMKCFG, DMKPGS,
DMKVMA

VMBLOK

Built by: DMKBLD
Released by: DMKBLD, DMKDIA, DMKLOG,
DMKUSO
Referenced by: DMKACO, DMKALG, DMKAPI,
DMKATS, DMKBLD, DMKCCH, DMKCCW, DMKADB,
DMKCDM, DMKCD S, DMKCF C, DMKCFD, DMKCFG,
DMKCFH, DMKCFM, DMKCF O, DMKCFP, DMKCF S,
DMKCFT, DMKCKP, DMKCKS, DMKCNS, DMKCPB,
DMKCP I, DMKCP S, DMKCPU, DMKCPV, DMKCOG,
DMKCQ H, DMKCQ P, DMKCQ R, DMKCQ Y, DMKCSB,
DMKCSO, DMKCS P, DMKCS Q, DMKCS T, DMKCS U,
DMKCSV, DMKDA S, DMKDEF, DMKDGD, DMKDIA,
DMKDIB, DMKDRD, DMKDSP, DMKENT, DMKERM,
DMKEXT, DMKFRE, DMKGIO, DMKGRF, DMKGRT,
DMKHVC, DMKHVD, DMKIOE, DMKIOF, DMKIOG,
DMKIOS, DMKISH, DMKJRL, DMKLNK, DMKLOG,
DMKLOH, DMKLOK, DMKMCC, DMKMCD, DMKMCH,
DMKMCT, DMKMIA, DMKMID, DMKMNI, DMKMOM,
DMKMSG, DMKMSW, DMKNES, DMKNET, DMKNLD,
DMKNLE, DMKPAG, DMKPER, DMKPGS, DMKPGT,
DMKPRG, DMKPRV, DMKPSA, DMKPTR, DMKQCN,
DMKRG A, DMKRG B, DMKRNH, DMKRPA, DMKRSE,
DMKRSP, DMKSCH, DMKSCN, DMKSEP, DMKSNC,
DMKSPL, DMKSSS, DMKSTK, DMKSVC, DMKTS C,
DMKTHI, DMKTMR, DMKTRA, DMKTRC, DMKTRD,
DMKTRK, DMKUDR, DMKUDU, DMKUNT, DMKUSO,
DMKVAT, DMKVCA, DMKVCH, DMKVCN, DMKVDA,
DMKVDC, DMKVDD, DMKVER, DMKVDS, DMKVER,
DMKVIO, DMKVMA, DMKVMC, DMKVSI, DMKVSP,
DMKVSQ, DMKWRM

VMCBLOK

Built by: DMKVMC
Released by: DMKVMC
Referenced by: DMKDSP, DMKVMC

VMCMHDR

Built by: N/A
Released by: N/A
Referenced by: DMKMSG

VMCPARM

Built by: Virtual machine user
Released by: Virtual machine user
Referenced by: DMKVMC

VRRBLOK

Built by: DMKVDS
Released by: DMKVDR
Referenced by: DMKCCW, DMKCFP, DMKDGD,
DMKGIO, DMKUNT, DMKVDS, DMKVI

VSPLCTL

Built by: DMKDRD, DMKVSP
Released by: DMKVSP
Referenced by: DMKCKP, DMKCS P, DMKCS Q,
DMKDRD, DMKSPL, DMKVSP, DMKVSQ

VSPXBLOK

Built by: DMKCS T
Released by: DMKCS T
Referenced by: DMKCKP, DMKCOG, DMKCS P,
DMKCS T, DMKSPL, DMKVDR, DMKVDS

XINTBLOK

Built by: DMKCFP, DMKCPB, DMKDSP,
DMKGRF, DMKRG A, DMKSCH, DMKTMR
Released by: DMKCFP, DMKDSP, DMKSCH,
DMKTMR
Referenced by: DMKCFP, DMKCPB, DMKDSP,
DMKGRF, DMKRG A, DMKSCH, DMKTMR, DMKVMC

XOBR3211

Built by: DMKRSE
Released by: DMKIOE
Referenced by: DMKIOF, DMKRSE

CMS CONTROL BLOCK REFERENCES

ABTAB

Assembled as part of DMSNUC.

Referenced by: DMSBAB, DMSDOS, DMSITP

ABWSECT

Assembled as part of DMSNUC

Referenced by: DMSABN, DMSDBG, DMSFRE,
DMSITI, DMSITP, DMSITS

ADTSECT

Assembled as part of DMSNUC.

Referenced by: DMSABN, DMSACC, DMSACF,
DMSACM, DMSALU, DMSAMS, DMSARE, DMSARN,
DMSARX, DMSASM, DMSASN, DMSAUD, DMSBOP,
DMSBWR, DMSCMP, DMSCPY, DMSDIO, DMSDLB,
DMSDLK, DMSDSK, DMSDSL, DMSDEX, DMSERS,
DMSEXC, DMSEXT, DMSFNS, DMSFOR, DMSIFC,
DMSINS, DMSLAD, DMSLAF, DMSLBM, DMSLBT,
DMSLDS, DMSLFS, DMSLKD, DMSLLU, DMSLST,
DMSMVE, DMSPUN, DMSQRY, DMSRNM, DMSROS,
DMSSET, DMSSOP, DMSSTT, DMSTPE, DMSTQQ,
DMSTRK, DMSUPD, DMSXCP

AFTSECT

Assembled as part of DMSNUC; also
created and released dynamically by
DMSLAF.

Referenced by: DMSBRD, DMSBWR, DMSCPY,
DMSERS, DMSFNS, DMSINT, DMSLAF, DMSPNT,
DMSRNM, DMSSOP, DMSSTT, DMSTPE

ANCHSECT

Built by: DMSSTG

Released by: Not released

Referenced by: DMSDOS, DMSSTG

BATLSECT

Assembled as part of DMSBTP.

Referenced by: DMSCIO, DMSITE, DMSPIO

BBOX

Assembled as part of DMSNUC.

Referenced by: No CMS references. This
block is used by the DOS supervisor.

BGCOM

Assembled as part of DMSNUC.

Referenced by: DMSAMS, DMSASN, DMSBAB,
DMSBCP, DMSCLS, DMSDLB, DMSDLK, DMSDMP,
DMSDOS, DMSDSV, DMSFCH, DMSFET, DMSINS,
DMSITP, DMSLLU, DMSOPL, DMSOPT, DMSPRV,
DMSQRY, DMSRRV, DMSSET, DMSMNM, DMSRRV,
DMSSTG, DMSVSR, DMSXCP

CMSTAXE

Built by: DMSSVT

Released by: DMSSVT

Referenced by: DMSKIT, DMSITE, DMSITI,
DMSSVT

CVTSECT

Assembled as part of DMSNUC.

Referenced by: DMSINS

DBGSECT

Assembled as part of DMSNUC.

Referenced by: DMSDBD, DMSDBG, DMSITE.

DEVSECT

Assembled as part of DMSNUC.

Referenced by: DMSTIO, DMSTPE

DEVTAB

Assembled as part of DMSNUC.

Referenced by: DMSASN, DMSDBD, DMSEDI,
DMSDEX, DMSINI, DMSLLU, DMSSVT

DIOSECT

Assembled as part of DMSNUC.

Referenced by: DMSACM, DMSDIO, DMSFNS, DMSITI

DMSCCB

Built by: N/A

Released by: N/A

Referenced by: DMSXCP

DOSSECT

Built by: DMSDLB

Released by: DMSDLB, DMSABN

Referenced by: DMSAMS, DMSBOP, DMSCLS, DMSDLB, DMSDLK, DMSDSV, DMSOPL, DMSQRY, DMSRRV, DMSSRV, DMSSVT, DMSVIP, DMSXCP

EDCB

Built by: DMSEDX

Released by: DMSEDI

Referenced by: DMSEDC, DMSEDI, DMSEDX, DMSGIO, DMSSCR

ERDSECT

Assembled as part of DMSNUC.

Referenced by: DMSERR

EXTSECT

Assembled as part of DMSNUC.

Referenced by: DMSINS, DMSINT, DMSIOW, DMSITE, DMSQRY, DMSSET, DMSSTG, DMSSVN, DMSSVT

EXTUAREA

Assembled as part of DMSNUC.

Released by: N/A

No CMS references.

FCBSECT

Built by: DMSFLD

Released by: DMSFLD, DMSABN

Referenced by: DMSALU, DMSARN, DMSARX, DMSASM, DMSDSL, DMSFCH, DMSFLD, DMSLDS, DMSMVE, DMSQRY, DMSROS, DMSAB, DMSSBD, DMSSBS, DMSSCT, DMSSEP, DMSSOP, DMSSQS, DMSSVN, DMSSVT,

FCHTAB

Assembled as part of DMSNUC.

Referenced by: DMSDOS, DMSFET

FICL

Assembled as part of DMSNUC.

Referenced by: No CMS references. This block is used by the DOS supervisor.

FRDSECT

Assembled as part of DMSNUC.

Referenced by: DMSFRE, DMSSET

FSCBD

Built by: N/A

Released by: N/A

Referenced by: DMSBRD, DMSDLK, DMSIFC, DMSZAP, and user programs that access the CMS file system

FSTD

Built by: N/A

Released by: N/A

Referenced by: DMSCPY, DMSEDX, DMSEXC, DMSFNS, DMSGND, DMSNCP, DMSSOP, DMSTPE

FSTSECT

Built by: DMSACF

Released by: DMSALU

Referenced by: DMSACF, DMSAMS, DMSARN, DMSARX, DMSASM, DMSBOP, DMSBRD, DMSBWR, DMSCPY, DMSDLK, DMSDSK, DMSDSL, DMSERS, DMSFNS, DMSGND, DMSIFC, DMSLAF, DMSLBM, DMSLKD, DMSMVE, DMSRNM, DMSSTT, DMSTPE, DMSUPD, DMSXCP, DMSZAP

FWSECT

Assembled as part of DMSNUC.

Referenced by: DMSABN, DMSACC, DMSACF, DMSACH, DMSALU, DMSAUD, DMSBRD, DMSBTB, DMSBTP, DMSBWR, DMSCIT, DMSCRD, DMSCWR, DMSJWT, DMSDIO, DMSDOS, DMSDSK, DMSERS, DMSFNS, DMSINT, DMSITE, DMSITI, DMSITP, DMSITS, DMSLAD, DMSLFS, DMSMOD, DMSPT, DMSQRY, DMSRNM, DMSSLN, DMSSOP, DMSSTT, DMSTPE, DMSTQQ

IHADECB

Built by: N/A

Released by: N/A

Referenced by: DMSSBD, DMSSBS, DMSSCT, DMSSEB, DMSSVT

IOSECT

Assembled as part of DMSNUC.

Referenced by: DMSABN, DMSHDI, DMSINT, DMSITI

KEYSECT

Built by: DMSSVT

Released by: DMSSVT

Referenced by: DMSSBD, DMSSVT

LDRST

Built by: DMSLDR

Released by: DMSLDR

Referenced by: DMSLDR, DMSLGT, DMSLIB, DMSLIO, DMSLSB, DMSOLD

LUBPR

Assembled as part of DMSNUC

Referenced by: DMSDLK, DMSDSV

LUBTAB

Assembled as part of DMSNUC.

Referenced by: DMSAMS, DMSBOP, DMSCLS, DMSDLB, DMSFCH, DMSLLU, DMSOPL, DMSPRV, DMSRRV, DMSSET, DMSSRV, DMSXCP

NICL

Assembled as part of DMSNUC.

Referenced by: DMSBOP, DMSCLS, DMSDLB, DMSLLU, DMSXCP

NUCON

Assembled as part of DMSNUC.

Referenced by: DMSABN, DMSACC, DMSACF, DMSACH, DMSALU, DMSAMS, DMSARE, DMSARN, DMSARX, DMSASM, DMSASN, DMSAUD, DMSBAB, DMSBOP, DMSBRD, DMSBTB, DMSBTP, DMSBWR, DMSCAT, DMSCIO, DMSCIT, DMSCLS, DMSCMP, DMSCPF, DMSCPY, DMSCRD, DMSCWR, DMSJWT, DMSDDB, DMSDBG, DMSDIO, DMSDLB, DMSDLK, DMSDMP, DMSDOS, DMSDSK, DMSDSL, DMSDSV, DMSEDI, DMSIDX, DMSERR, DMSERS, DMSEXC, DMSEXT, DMSFCH, DMSFET, DMSFLD, DMSFNS, DMSFOR, DMSFRE, DMSGIO, DMSGLB, DMSGND, DMSHDI, DMSHDS, DMSIFC, DMSINA, DMSINI, DMSINM, DMSINS, DMSINT, DMSIOW, DMSITE, DMSITI, DMSITP, DMSITS, DMSLAD, DMSLAF, DMSLBM, DMSLBT, DMSLDR, DMSLDS, DMSLFS, DMSLGT, DMSLIB, DMSLIC, DMSLKD, DMSLLU, DMSLOA, DMSLSB, DMSLST, DMSLSY, DMSMDP, DMSMOD, DMSMVE, DMSNCP, DMSOLD, DMSOPL, DMSOPT, DMSOR1, DMSOVR, DMSOVS, DMSPIO, DMSPT, DMSPT, DMSPRV, DMSPT, DMSQRY, DMSRDC, DMSRNE, DMSRNM, DMSROS, DMSRRV, DMSRAB, DMSSBS, DMSSCN, DMSSET, DMSSEB, DMSSOP, DMSSET, DMSSLN, DMSSMN, DMSSOP, DMSSQS, DMSRRT, DMSRV, DMSSSK, DMSSTG, DMSSTT, DMSSVN, DMSSVT, DMSSTG, DMSSTT, DMSTPE, DMSTQQ, DMSTYP, DMSUPD, DMSVIB, DMSVIP, DMSVSR, DMSXCP, DMSZAP

OPSECT

Assembled as part of DMSNUC.

Referenced by: DMSABN, DMSARX, DMSASM, DMSCPY, DMSCRD, DMSCWR, DMSCWT, DMSDBG, DMSEXC, DMSEXT, DMSINS, DMSINT, DMSROS, DMSSBD, DMSSBS, DMSSCT, DMSSEB, DMSSOP, DMSSQS, DMSVSN, DMSSVT

OSFST

Built by: DMSROS
Released by: DMSALU
Referenced by: DMSABN, DMSALU, DMSBOP, DMSDLK, DMSFCH, DMSMVE, DMSROS, DMSRRV, DMSSOP, DMSSRV, DMSSTT

PUBADR

Assembled as part of DMSNUC.
Referenced by: DMSBOP, DMSCLS, DMSDLK, DMDSV, DMSLLU, DMSPRV, DMSXCP

OVSECT

Built by: N/A
Released by: N/A
Referenced by: DMSITS, DMSOVR

PUBOWNER

Assembled as part of DMSNUC
Referenced by: DMSBOP, DMSCLS, DMSDLK, DMSLLU, DMSXCP

PCTAB

Assembled as part of DMSNUC.
Referenced by: DMSBAB, DMSDOS, DMSITP

SSAVE

Built by: DMSITS
Released by: DMSITS
Referenced by: DMSABN, DMSACC, DMSBAB, DMSDBG, DMSDLB, DMSDOS, DMSERR, DMSFLD, DMSFRE, DMSIFC, DMSITP, DMSITS, DMSLDR, DMSOVS, DMSSAB, DMSSLN, DMSSMN, DMSSOP, DMSSTG, DMSSVN, DMSSVT, DMSVIP, DMSXCP

PDSSECT

Built by: DMSSVT
Released by: DMSSVT
Referenced by: DMSSTG, DMSSVT

SUBSECT

Assembled as part of DMSNUC.
Referenced By: DMSABN, DMSINM, DMSINT

PGMSECT

Assembled as part of DMSNUC.
Referenced by: DMSITP, DMSSAB, DMSSLN, DMSSTG, DMSSVT

SVCSECT

Assembled as part of DMSNUC.
Referenced by: DMSCIT, DMSFRE, DMSHDS, DMSINT, DMSITE, DMSITS, DMSLAD, DMSLFS, DMSOVR, DMSOVS, DMSSLN

PIBADR

Assembled as part of DMSNUC.
Referenced by: DMSBAB, DMSDOS, DMSITP

SVEARA

Assembled as part of DMSNUC.
Referenced by: DMSBAB, DMSDOS, DMSITP

PIB2TAB

Assembled as part of DMSNUC.
Referenced by: DMSBAB, DMSDOS, DMSVSR

SYSKOM

Assembled as part of DMSNUC.
Referenced by: DMSBAB, DMSBOP, DMSDOS, DMSFET, DMSITP, DMSQRY, DMSSTG, DMSSYN

SYSNAMES

Assembled as part of DMSNUC.

Referenced by: DMSAMS, DMSBOP, DMSBPT,
DMSDOS, DMSEDX, DMSEXC, DMSINS, DMSINT,
DMSITS, DMSQRY, DMSSET, DMSVIB, DMSVSR

TSOBLKS

Assembled as part of DMSNUC.

Referenced by: DMSSET

USAVE

Built by: N/A

Released by: N/A

Referenced by: DMSITS

USERSECT

Assembled as part of DMSNUC.

No CMS references.

RSCS CONTROL BLOCK REFERENCES

ASYNE

Built by: DMTASY
Released by: DMTASY, DMTASK
Referenced by: DMTASY, DMTEXT, DMTIOM,
DMTSIG

GIVEE

Built by: DMTGIV
Released by: DMTAKE, DMTASK
Referenced by: DMTAKE, DMTASK, DMTGIV

BUFDSECT

Built by: DMTSML
Released by: DMTSML
Referenced by: DMTSML

IOE

Built by: DMTIOM
Released by: DMTIOM
Referenced by: DMTASK, DMTIOM, DMTREX

COMDSECT

Built by: DMTCOM
Released by: N/A
Referenced by: DMTAXS, DMTCMX, DMTMGX,
DMTNPT, DMTREX, DMTSML

IOTABLE

Built by: DMTIOM, DMTCRE, DMTNPT,
DMTREX, DMTSML
Released by: DMTNPT, DMTSML
Referenced by: DMTAXS, DMTCMX, DMTCRE,
DMTINI, DMTIOM, DMTREX, DMTSML

DEVTABLE

Built by: DMTNPT
Released by: DMTNPT
Referenced by: DMTNPT

LINKTABL

Assembled into DMTSYS at system
generation; also built by DMTCMX.
Released by: DMTCMX
Referenced by: DMTASY, DMTAXS, DMTCMX,
DMTCOM, DMTCRE, DMTEXT, DMTLAX, DMTMGX,
DMTNPT, DMTREX, DMTSML

FREEE

Built by: DMTQRQ
Released by: DMTQRQ
Referenced by: DMTASK, DMTINI, DMTQRQ

REQBLOCK

Built by: DMTNPT
Released by: DMTNPT
Referenced by: DMTNPT

GIVE

Built by: DMTSML, DMTNPT, DMTAXS, DMTREX
Released by: N/A
Referenced by: DMTSML, DMTNPT, DMTAXS,
DMTREX

ROUTE

Assembled in DMTSYS
Released by:
Referenced by: DMTAXS

TAKE

Built by: DMTSML, DMTNPT, DMTAXS, DMTREX
Released by: N/A
Referenced by: DMTSML, DMTNPT, DMTAXS, DMTREX

SVECTORS

Assembled into DMTVEC at system generation; resides in the RSCS nucleus.

Referenced by: DMTAKE, DMTASK, DMTASY, DMTAXS, DMTCHX, DMTCOM, DMTCRE, DMTDSP, DMTEXT, DMTGIV, DMTINI, DMTIOM, DMTLAX, DMTMGX, DMTNPT, DMTQRQ, DMTREX, DMTSIG, DMTSML, DMTSTO, DMTSVC, DMTWAT

TANKDSEC

Built by: DMTSML
Released by: DMTSML
Referenced by: DMTSML

TAG

Built by: DMTAXS
Released by: DMTAXS
Referenced by: DMTAXS, DMTCHX, DMTNPT, DMTSML

TAREA

Assembled into each task module.
Released by: DMTASK
Referenced by: DMTAKE, DMTASK, DMTASY, DMTCOM, DMTCRE, DMTDSP, DMTEXT, DMTGIV, DMTIOM, DMTREX, DMTSIG, DMTSTO, DMTSVC

TAGAREA

Built by: DMTAXS
Released by: N/A
Referenced by: DMTAXS

TASKE

Built by: DMTASK
Released by: DMTASK
Referenced by: DMTAKE, DMTASK, DMTASY, DMTAXS, DMTCOM, DMTDSP, DMTEXT, DMTGIV, DMTINI, DMTIOM, DMTNPT, DMTSTO, DMTREX, DMTSIG, DMTSML, DMTSTC, DMTSVC, DMTWAT

TCTDSECT

Built by: DMTSML
Released by: DMTSML
Referenced by: DMTSML

Title: IBM Virtual Machine
Facility/370:
Data Areas and Control
Block Logic

Order No. SY20-0884-3

Please check or fill in the items; adding explanations/comments in the space provided.

Which of the following terms best describes your job?

- | | | | |
|--|--|---|--|
| <input type="checkbox"/> Customer Engineer | <input type="checkbox"/> Manager | <input type="checkbox"/> Programmer | <input type="checkbox"/> Systems Analyst |
| <input type="checkbox"/> Engineer | <input type="checkbox"/> Mathematician | <input type="checkbox"/> Sales Representative | <input type="checkbox"/> Systems Engineer |
| <input type="checkbox"/> Instructor | <input type="checkbox"/> Operator | <input type="checkbox"/> Student/Trainee | <input type="checkbox"/> Other (explain below) |

How did you use this publication?

- | | | |
|--|---|--|
| <input type="checkbox"/> Introductory text | <input type="checkbox"/> Reference manual | <input type="checkbox"/> Student/ <input type="checkbox"/> Instructor text |
| <input type="checkbox"/> Other (explain) _____ | | |

Did you find the material easy to read and understand? Yes No (explain below)

Did you find the material organized for convenient use? Yes No (explain below)

Specific criticisms (explain below)

Clarifications on pages _____

Additions on pages _____

Deletions on pages _____

Errors on pages _____

Explanations and other comments:

Trim Along This Line

Cut or Fold Along Line

Reader's Comment Form

Fold and tape

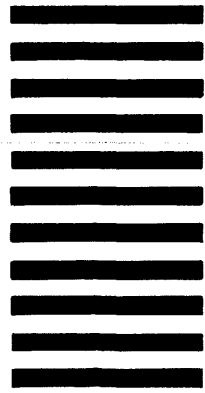
Please Do Not Staple

Fold and tape



NO POSTAGE
NECESSARY
IF MAILED
IN THE
UNITED STATES

BUSINESS REPLY MAIL
FIRST CLASS PERMIT 40 ARMONK, NEW YORK



POSTAGE WILL BE PAID BY ADDRESSEE:

International Business Machines Corporation
Department D58, Building 706-2
PO Box 390
Poughkeepsie, New York 12602

Attn: VM/370 Publications

Fold and tape

Please Do Not Staple

Fold and tape

IBM VM/370 Data Areas and Co Block Logic

Printed in U.S.A.

SY20-0884-3



International Business Machines Corporation
Data Processing Division
1133 Westchester Avenue, White Plains, N.Y. 10604

IBM World Trade Americas/Far East Corporation
Town of Mount Pleasant, Route 9, North Tarrytown, N.Y., U.S.A. 10591

IBM World Trade Europe/Middle East/Africa Corporation
360 Hamilton Avenue, White Plains, N.Y., U.S.A. 10601