Systems

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

Release 3 PLC 1

This publication, along with the *IBM Virtual Machine Facility/370: System Logic and Problem Determination Guide*, is intended for 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).

Prerequisite Publications

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.

IBM

This publication contains reference information about control blocks and data areas associated with three major components of VM/370: CP, CMS, and RSCS.

This publication contains three sections and five appendixes:

- "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, which define constants and variables used during execution.

- "Appendix C: RSCS Request Elements" contains RSCS request elements, which are 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 modules that reference data areas and control blocks.

OTHER VM/370 DATA AREAS AND CONTROL BLOCKS

Some data areas and control blocks relating to 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 manual, Order No. SY20-0882.

RELATED PUBLICATIONS

This publication is intended to be used in conjunction with IBM Virtual Machine Facility/370: System Logic and Problem Determination Guide, Order No. SY20-0885. Users of the publication IBM Virtual Machine Facility/370: System Programmer's Guide, Order No. GC20-1807 will also find this publication useful.

For a glossary of VM/370 terms, see the IBM Virtual Machine Facility/370: Glossary and Master Index, Order No. GC20-1813.

The hardware and software support personnel or installation system programmer should use the IBM Virtual Machine Facility/370: Interactive Problem Control System (IPCS) User's Guide, Order No. GC20-1823, for information on how to use the facilities of IPCS.

HOW TO USE THIS PUBLICATION

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

Control blocks and data areas are generally 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.

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.

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.

In the case where 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 formatted block is followed by listing-related information: 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.

FIGURES

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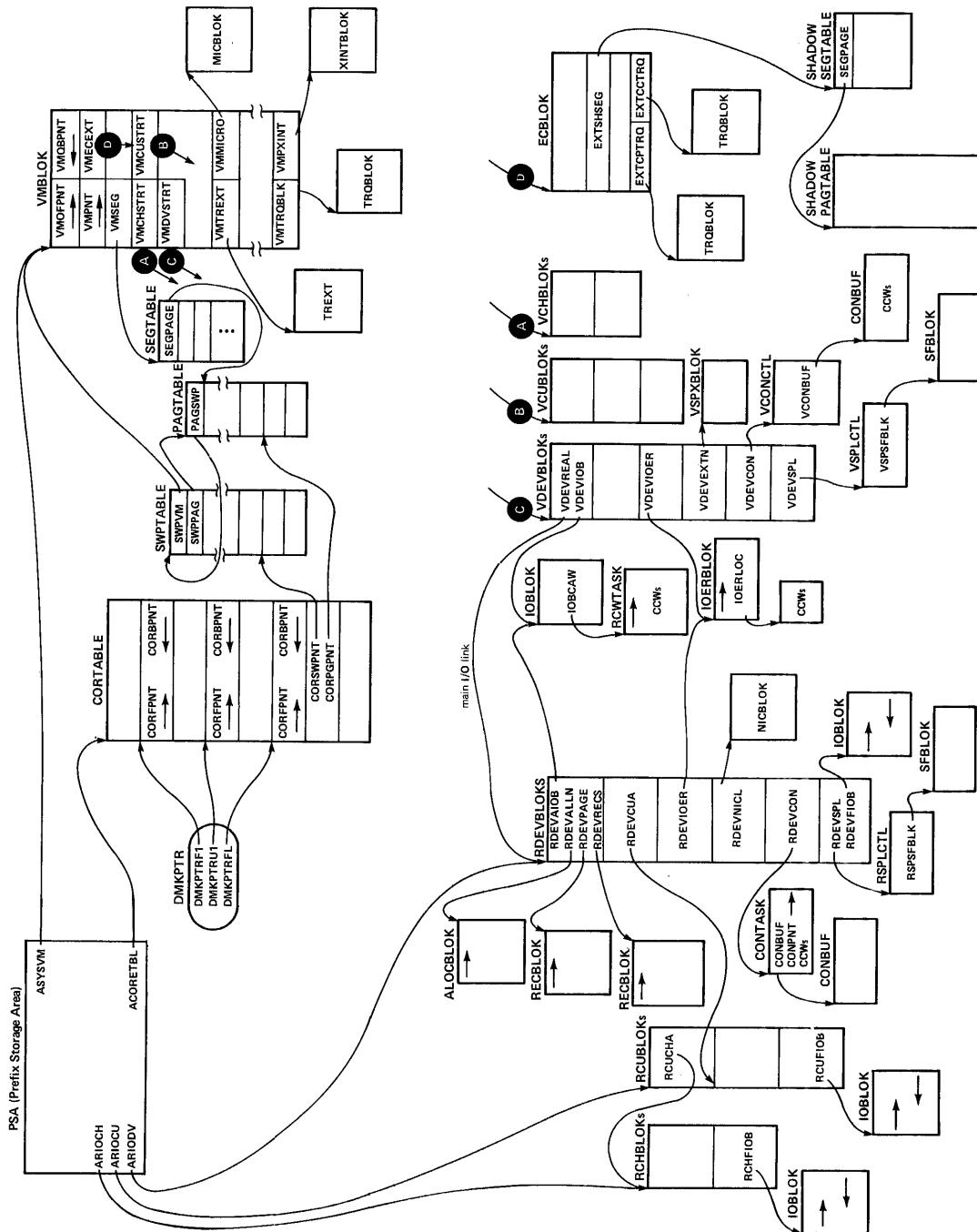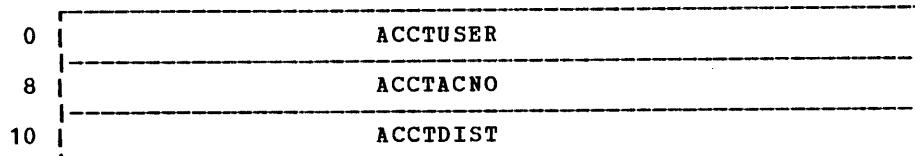
Figure 1. CP Control Block Relationships

ACCTBLOK: USER ACCOUNTING BLOCK
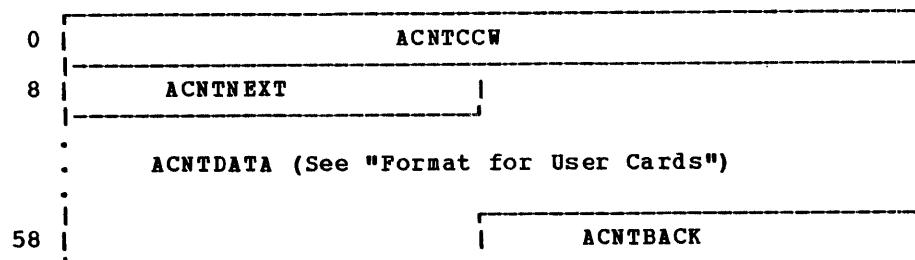
ACCTBLOK provides header information for spool files. The VMACOUNT field (hex 168) in the VMBLOK points to ACCTBLOK.

```
      r---------------------------------------------------------------1
    0 |                          ACCTUSER                             |
      |---------------------------------------------------------------|
    8 |                          ACCTACNO                             |
      |---------------------------------------------------------------|
   10 |                          ACCTDIST                             |
      L---------------------------------------------------------------J
```

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')

ACNTBLCK: ACCOUNTING CARD BUFFER BLOCK
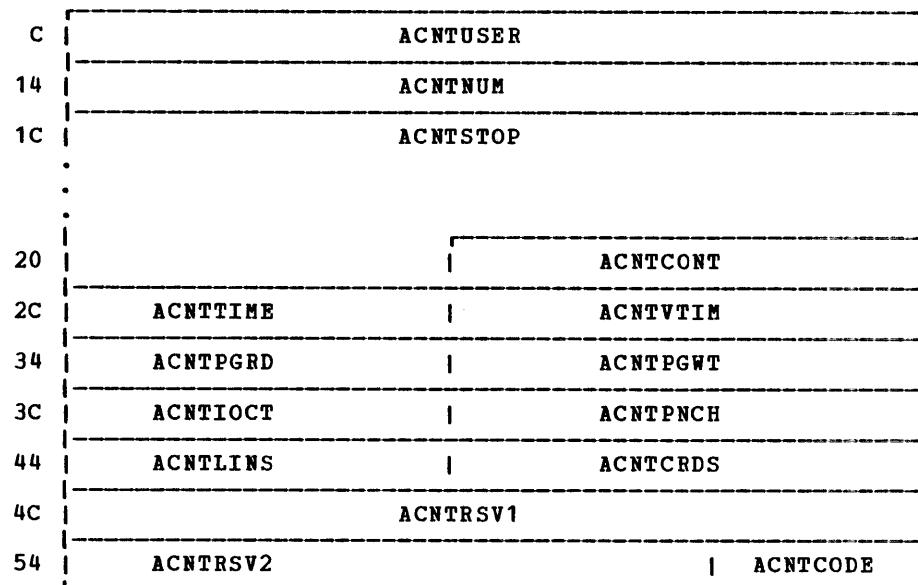
ACNTBLOK provides accounting and statistical information on each user that has used VM/370 facilities. The ARSPAC field (hex 39C) in the Prefix Storage Area (PSA) points to the start of the chain of ACNTBLOKs.

```
      r---------------------------------------------------------------1
    0 |                          ACNTCCW                              |
      |----------------------------------------|                      |
    8 |          ACNTNEXT                      |                      |
      |----------------------------------------J                      |
      .                                                               .
      .        ACNTDATA (See "Format for User Cards")                 .
      .                                                               .
      |                             r---------------------------------|
   58 |                             |          ACNTBACK               |
      L-----------------------------L---------------------------------J
```

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 preceding ACNTBLOK data area.
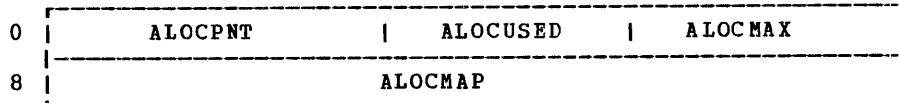
```
      r-----------------------------------------------------------------1
   C  | I                        ACNTUSER                             I |
      I---------------------------------------------------------------I
   14 | I                        ACNTNUM                              I |
      I---------------------------------------------------------------I
   1C | I                        ACNTSTOP                             I |
      .                                                               .
      .                                                               .
      .                          r-------------------------------------I
   20 | I                        I          ACNTCONT                  I |
      I---------------------------------------------------------------I
   2C | I       ACNTTIME         I          ACNTVTIM                  I |
      I---------------------------------------------------------------I
   34 | I       ACNTPGRD         I          ACNTPGWT                  I |
      I---------------------------------------------------------------I
   3C | I       ACNTIOCT         I          ACNTPNCH                  I |
      I---------------------------------------------------------------I
   44 | I       ACNTLINS         I          ACNTCRDS                  I |
      I---------------------------------------------------------------I
   4C | I                        ACNTRSV1                             I |
      I---------------------------------------------------------------I
   54 | I     ACNTRSV2                          I    ACNTCODE         I |
      L---------------------------------------------------------------J
```

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 CPU time used
30	ACNTVTIM	DS	1F	Milliseconds of virtual CPU time used
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	ACNTCRDS	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

	Card Codes for ACNTCODE			
		DC	C'C0'	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.

		ORG	ACNTTIME	
2C	ACNTDEVC	DS	XL4	Device code (CTFM). See DEVTYPE copy file
30	ANCTNCYL	DS	1H	Number of cylinders of T-disk space

ALOCBLOK: DASD CYLINDER ALLOCATION BLOCK

ALOCBLOK provides information on temporary disk space available to a virtual machine. The RDEVALLN field (hex 28) in the RDEVBLOK points to the ALOCBLOK.

```
    ┌──────────────────────────────────────────────────────────┐
  0 │       ALOCPNT        │    ALOCUSED    │    ALOCMAX         │
    ├──────────────────────────────────────────────────────────┤
  8 │                          ALOCMAP                          │
    └──────────────────────────────────────────────────────────┘
```

Hexadecimal Displacement	Field Name			Field Description, Contents, Meaning
0	ALOCPNT	DS	1F	Pointer to next ALOCBLOK on chain
4	ALOCUSED	DS	1H	Number of cylinders currently in use
6	ALOCMAX	DS	1H	Maximum number of cylinders available
8	ALOCMAP	DS	0F	Cylinder allocation bit map

Bits defined in ALOCMAP
0 = Cylinder is available
1 = Cylinder has been allocated

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

$$\text{ALOCSIZE(doublewords)} = ((((\text{ALOCMAX}+7)/8)+7)/8)+1$$

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

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	ALOCCYL1	DS	1H	First cylinder of T-disk area
6	ALOCCYL2	DS	1H	Last cylinder of T-disk area

Bytes defined in ALOCMAP
X'00' = Cylinder is available
X'AA' = Cylinder has been allocated

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

$$\text{ALOCSIZE(doublewords)} = ((\text{ALOCCYL2}-\text{ALOCCYL1}+8)/8)+1$$

Bytes for cylinders that are not available are marked allocated.

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 RDEVBSC field (hex 30) in the RDEVBLOK points to the BSCBLOK.
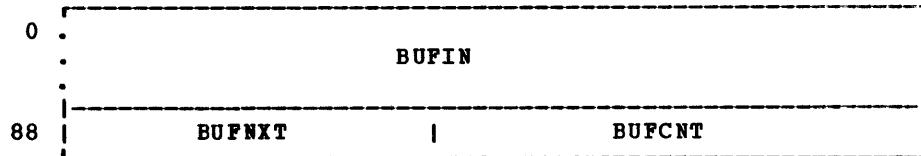
```
      r------------------------------------------------------------------1
   0  |                          BSCSCCW1                                 |
      |------------------------------------------------------------------|
   8  |                          BSCSCCW2                                 |
      |------------------------------------------------------------------|
  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  |      BSCTMRQ             |  BSCRESP       |                        |
      |----------------------------------------------|                    |
  88  |                          BSCREAD                                  |
      .                                                                   .
      .                                                                   .
      .                                                                   .
 128  |                                                                   |
      L------------------------------------------------------------------J
```

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	BSCECCW1	DS	1D	CCW for write error response
40	BSCECCW2	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	BSCFLAG bits

Bits defined in BSCFLAG

	BSCRVI	EQU	X'80'	Sending RVI response
	BSCENQ	EQU	X'40'	ENQ 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
	BSCTSTRQ	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 | BSCFLAG1 bits |

Bits defined in BSCFLAG1

	BSCETB	EQU	X'80'	Station transmitted block record
	BSCIGN	EQU	X'40'	Ignore block record

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; 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	BSCTMRQ	DS	1F	Pointer to TRQBLOK for poll delay
84	BSCRESP	DS	1H	Response buffer for selection
86	BSCREAD	DS	CL257	Head buffer for polling
	BSCSIZE1	EQU	*-(BSCREAD+1)	Read buffer size in bytes
	BSCSIZE2	EQU	(BSCREAD-BSCBLOK)	BSC Header size in bytes
	BSCSIZE	EQU	(*-BSCBLOK+7)	18 BSC blocksize in doublewords

BUFFER

BUFFER is a buffer area that contains console input to be used by CP.

```
          ┌──────────────────────────────────────────────────────────┐
    0  .   │                                                          │  .
       .   │                         BUFIN                            │  .
       .   │                                                          │  .
       .   ├─────────────────────────────────┬────────────────────────┤
   88  │   │           BUFNXT                 │        BUFCNT          │
       │   └─────────────────────────────────┴────────────────────────┘
```

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
	Bits defined in BUFCNT			
	BUFINLTH EQU		L'BUFIN	Size of input line in bytes (136)
	BUFSIZE EQU		(*-BUFFER)/8	Size of input line 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 | B*1   | B*2   | B*3   | B*4   |   CCSW2REV   | B*5   | B*6   |
     |----------------------------------------------------------------|
   8 |                           CCDATE                               |
     |----------------------------------------------------------------|
  10 |                           CCCPUID                              |
     |----------------------------------------------------------------|
  18 |                           CCPROGID                             |
     |----------------------------------------------------------------|
  20 |                           FAILADD                              |
     |                                                                |
  28 |                                                                |
     |----------------------------------------------------------------|
  30 |                           FAILCCW                              |
     |----------------------------------------------------------------|
  38 |                           FAILCSW                              |
     |----------------------------------------------------------------|
  40 | B*7   | B*8   | B*9   | B*10  |          CCDEVTYP             |
     |----------------------------------------------------------------|
  48 |   CCHANID     |   CCHCUA      |          CCHMP                |
     |----------------------------------------------------------------|
  50 |                           CCHLOG                               |
```

Hexadecimal Displacement	Field Name				Field Description, Contents, Meaning
0	CCRECTYP	DS	1X	B*1	Record type
1	CCOPSYS	DS	1X	B*2	Operating system
2	CCSW1	DS	1X	B*3	Switch 1
3	CCSW2	DS	1X	B*4	Switch 2
4	CCSW2REV	DS	2X		Unused
6	CCRECNT	DS	1X	B*5	Record count
7	CCRECNT1	DS	1X	B*6	Unused
8	CCDATE	DS	1D		Date and time
10	CCCPUID	DS	1D		CPU ID
18	CCPROGID	DS	1D		Userid
20	FAILADD	DS	8H		Active I/O units
30	FAILCCW	DS	1D		Failing CCW
38	FAILCSW	DS	1D		Failing CSW
40	FAILECSW	DS	0F		Failing ECSW
40	IGPRGFLG	DS	CL1	B*7	Program flag bits

Bits defined in IGPRGFLG
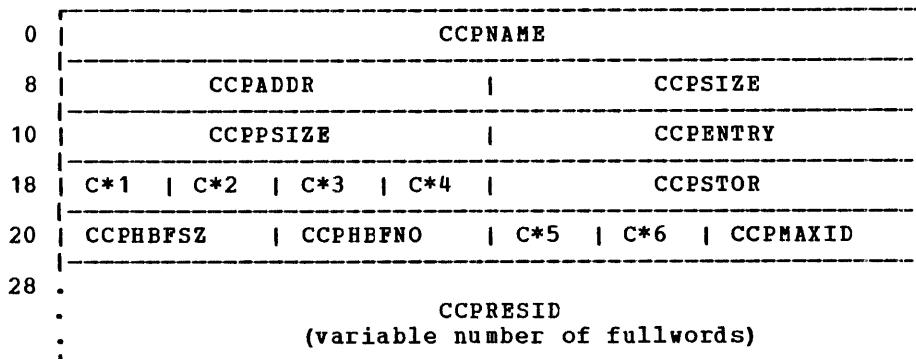
CCHSIOB	EQU	X'80'	Start I/O bit
CCHINTB	EQU	X'40'	Interrupt bit
CCHTIO	EQU	X'20'	Test I/O bit
CCHHIO	EQU	X'10'	Halt I/O bit
CCHSNSB	EQU	X'04'	Sense data stored bit
CCHCNTB	EQU	X'02'	Count valid bit
CCHNRYB	EQU	X'01'	No retry bit

Hexadecimal Displacement	Field Name				Field Description, Contents, Meaning
41	IGBLAME	DS	CL1	B*8	Probable source of error

Bits defined in IGBLAME

CCHCPU	EQU	X'80'		CPU 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'		I/O interface is source of error

42	IGVALIDB	DS	CL1	B*9	Validity indicator bits

Bits defined in IGVALIDB

CCHINTFV	EQU	X'80'		Interface address valid
CCHRCV	EQU	X'10'		Retry code valid
CCHUSV	EQU	X'08'		Selective reset 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	B*10	Termination/sequence code bits

Bits defined in IGTERMSQ

COMPSYS	EQU	X'C0'		System reset
COMPSEL	EQU	X'80'		Selective reset
COMPFES	EQU	X'40'		Forced ending sequence
COMPID	EQU	X'00'		Interface disconnect
CCHDI	EQU	X'08'		Disconnect in sequence code bits

Sequence Code Bits

RTCODE0	EQU	X'00'		Retry code values
RTCODE1	EQU	X'01'		for the constructed ECSW
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	Channel ID
49	CCHCUA	DS	CL3	Actual failing device address
4C	CCHMP	DS	1F	Not used
50	CCHLOG80	DS	0CL112	2880 channel − 112 bytes
50	CCHLOG70	DS	0CL24	2870 channel − 24 bytes
50	CCHLOG60	DS	0CL24	2860 channel − 24 bytes

Bit defined in CCHLOG60

CCHSIZE1	EQU	(*-CCHREC)/8	Size in doublewords (X'0A')

50	CUHADDR	DS	1F	Unit address stored by integrated channel

Bit defined in CUHADDR

CCHSIZE	EQU	(*-CCHREC)/8	Size in doublewords

54	CCHLOG45	DS	0CL96	Model 145 integrated channel (96 bytes)
54	CCHLOG35	DS	0CL24	Model 135 integrated channel (24 bytes)

Section 1. CP Data Areas and Control Blocks   15

CCPARM: COMMUNICATIONS CONTROLLER PARAMETER LIST

CCPARM provides control information used for loading and controlling the 3704/3705 Communication Controller NCP, EP, and PEP programs and their attached resources.

```
       r-----------------------------------------------------------------,
   0   |                          CCPNAME                                |
       |------------------------------------------------------------------|
   8   |          CCPADDR                |          CCPSIZE               |
       |------------------------------------------------------------------|
  10   |          CCPPSIZE               |          CCPENTRY              |
       |------------------------------------------------------------------|
  18   | C*1  | C*2  | C*3  | C*4  |          CCPSTOR                     |
       |------------------------------------------------------------------|
  20   | CCPHBFSZ    | CCPHBFNO   | C*5  | C*6  | CCPMAXID                |
       |------------------------------------------------------------------|
  28   .                                                                  .
       .                          CCPRESID                                .
       .                  (variable number of fullwords)                  .
       L_____J
```

Hexadecimal Displacement	Field Name				Field Description, Contents, Meaning
0	CCPNAME	DS	CL8		NCPNAME specified in NAMECP 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
	Bits defined in CCPTYPE				
	CCPTNCP	EQU	X'01'		Network Control Program
	CCPTEP	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
	Bits defined in CCPAONE and CCPATWO				
	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		3704/3705 storage size specified (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 (bytes)
25	CCPPAD1	DS	1X	C*6	Subsequent buffer pad count
	HOST Values Required by VM/370 Support for 3704/3705				
	CCPVPAD0	EQU	34		34-byte pad in first BTU 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 EP mode

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).

```
       _____
    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  |  CHXYADD |
      |-------------------------------------------------------------------|
   28 |          CHXIDAW              |             CHYIDAW              |
      |-------------------------------------------------------------------|
   30 |          CHXCNCT              |             CHYCNCT              |
      |-------------------------------------------------------------------|
   38 |          CHXDATN              |             CHYDATN              |
      |_____|
```

Note: As indicated in the illustrated block, the CHXBLOK and CHYBLOK are interleaved with a 4-byte displacement. The X-side VDEVBLOK points to the +0 slot, the Y-side VDEVBLOK 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 which is at +0 or at +4.
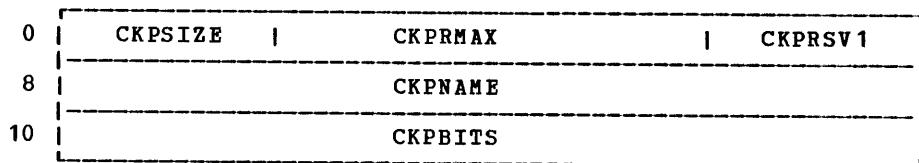
Hexadecimal Displacement	Field Name				Field Description, Contents, Meaning
0	CHXOTHR	DS	2F		VMBLOK address of Y-side adapter user
8	CHXFLAG	DS	1X	X*1	Internal processing flags
	Bits defined in CHXFLAG				
	CHBMNOP	EQU	X'80'		Modified NOP command issued (also in CMDT)
	CHBM370	EQU	X'40'		CTCA operating 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'		HIO or HDV issued
	CHBWAIT	EQU	X'02'		CPEXBLOK 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.)
	Bits defined in CHXCMDT				
	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	X*4	Virtual CAW protection key
C		DS	4X		
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		
28	CHXIDAW	DS	2F		Active indirect data list word
30	CHXCNCT	DS	2F		CPEXBLOK for channel reconnect
38	CHXDATN	DS	2F		IOBLOK address for deferred attention interruption
	CHBSIZE	EQU	(*—CHXBLOK)/8		Total block size in doublewords (X'08')
	CHYBLOK				Y—side channel adapter block
0	CHYOTHR	DS	2F		VMBLOK address of X—side adapter user
8	CHYFLAG	DS	1X	Y*1	Internal processing flags

Bits <u>defined</u> <u>in</u> <u>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 Y—side
CHBREST	EQU	X'10'		CTCA has been reset X—side and Y—side
CHBEOFL	EQU	X'08'		Force EOF to next READ
CHBHIO	EQU	X'04'		Halt I/O or halt Device issued
CHBWAIT	EQU	X'02'		CPEXBLOK available for channel reconnect
CHBEENT	EQU	X'01'		Channel end has been presented on SIO

9	CHYCMDB	DS	1X	Y*2	Active CCW command byte buffer
A	CHYCMDT	DS	1X	Y*3	Active CCW command byte

Bits <u>defined</u> <u>in</u> <u>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
CHBWRIT	EQU	X'01'		Write

B	CHYPKEY	DS	1X	Y*4	Virtual CAW protection key
C		DS	4X		
10	CHYNCCW	DS	2F		Next CCW fetch address
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		
28	CHYIDAW	DS	2F		Active indirect data list word
30	CHYCNCT	DS	2F		CPEXBLOK for channel reconnect
38	CHYDATN	DS	2F		IOBLOK address for deferred I/O interrupt

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.

```
      r------------------------------------------------------------------n
  0  |   CKPSIZE    |      CKPRMAX           |    CKPRSV1        |
     |------------------------------------------------------------------|
  8  |                       CKPNAME                                    |
     |------------------------------------------------------------------|
 10  |                       CKPBITS                                    |
     L------------------------------------------------------------------J
```
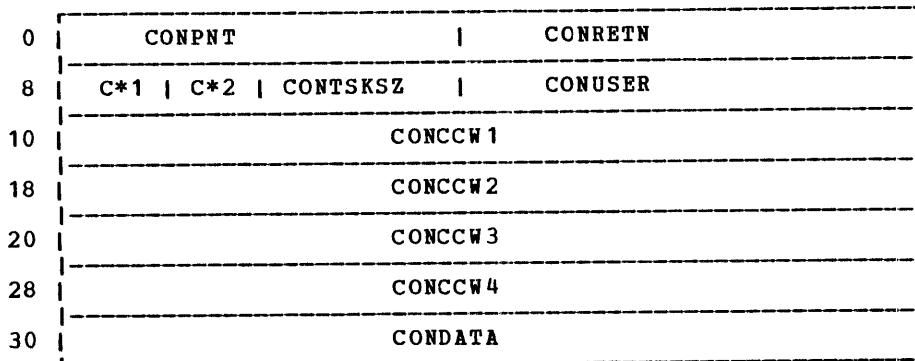
Hexadecimal Displacement	Field Name			Field Description, Contents, Meaning
0	CKPSIZE	DS	1H	Size of CKPBLOK in doublewords
2	CKPRMAX	DS	1H	Number of resources checkpointed
4	CKPRSV1	DS	1F	Reserved for IBM use
8	CKPNAME	DS	CL8	370X control program reference name
A	CKPBITS	DS	0D	Bit map of enabled lines or resources
	CKPBKSZ	EQU	(CKPBITS-CKPBLOK)/8	Header size in doublewords (X'02')

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.

```
      r--------------------------------------------------------------¬
   0  |        CONPNT              |         CONRETN                  |
      |--------------------------------------------------------------|
   8  |  C*1 | C*2 | CONTSKSZ      |         CONUSER                  |
      |--------------------------------------------------------------|
  10  |                        CONCCW1                                |
      |--------------------------------------------------------------|
  18  |                        CONCCW2                                |
      |--------------------------------------------------------------|
  20  |                        CONCCW3                                |
      |--------------------------------------------------------------|
  28  |                        CONCCW4                                |
      |--------------------------------------------------------------|
  30  |                        CONDATA                                |
      L--------------------------------------------------------------
```

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	1F	C*1	CONTASK status control flags

Bits defined in CONSTAT

CONOUTPT	EQU	X'80'		Output CONTASK
CONRESP	EQU	X'40'		Response expected from this CONTASK
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 RDEVLLEN
CONSYNC	EQU	X'01'		CONTASK for synchronization only

Hexadecimal Displacement	Field Name				Field Description, Contents, Meaning
9	CONPARM	DS	1X	C*2	DMKQCN parameter flags (see Appendix A)
A	CONTSKSZ	DS	1H		CONTASK size in doublewords
C	CONUSER	DS	1F		Address of VMBLOK for destination user
10	CONCCW1	DS	1D		First console I/O CCW
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
		ORG	CONCCW1		
10	CONADDR	DS	1F		CCW data address
14	CONFLAG	DS	1X		CCW flag bits
15	CONRSV3	DS	1X		Reserved for IBM use
16	CONCNT	DS	1H		CCW byte count

Hexadecimal Displacement	Field Name			Field Description, Contents, Meaning
		ORG	CONADDR	
10	CONCOMND	DS	1X	CCW command code

Bits redefined in CONCCW for 3704/3705 Network Control Program

		ORG	CONCCW3+2	
22	CONSRID	DS	1H	Source identifier
24	CONDEST	DS	1H	Destination resource ID
26	CONRTAG	DS	1H	Request for this CONTASK
28	CONSYSR	DS	1X	3704/3705 system response byte
29	CONEXTR	DS	1X	3704/3705 extended response byte
2A	CONTCMD	DS	1H	BTU command modifier
2C	CONFUNC	DS	1X	BDU function control flags
2D	CONDFLG	DS	1X	BDU data control flags
2E	CONDCNT	DS	1H	Text data length

Bits Redefined for 3270 Remote Support

		ORG	CONCCW4	
28	CONLABEL	DS	1X	Return index value
29	CONSTX	DS	1X	Start text character
2A	CONESC	DS	1X	Escape character
2B	CONCMD	DS	1X	Command code for remote station
2C	CONWCC	DS	1X	Write control character
2D	CONSBA	DS	1X	Start buffer address
2E		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.

```
        ┌───────────────────────────────────┬───────────────────────────────────┐
0  │            CORFPNT                 │            CORBPNT                 │
        ├───────────────────────────────────┼───────────────────────────────────┤
8  │            CORSWPNT                │            CORPGPNT                │
        └───────────────────────────────────┴───────────────────────────────────┘
```

Hexadecimal Displacement	Field Name			Field Description, Contents, Meaning
0	CORFPNT	DS	1F	Pointer to next CORTABLE entry in queue
4	CORBPNT	DS	1F	Pointer to previous CORTABLE entry in queue
8	CORSWPNT	DS	1F	Pointer to SWPTABLE for page
C	CORPGPNT	DS	1F	Pointer to PAGTABLE for page
		ORG	CORSWPNT	
8	CORFLAG	DS	1X	CORTABLE entry status flags

Bits defined in CORFLAG

CORIOLCK	EQU	X'80'		Page locked for I/O, CORLCNT 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 CP
CORDISA	EQU	X'01'		Page disabled, not available

Entry Definition if Page Is Locked
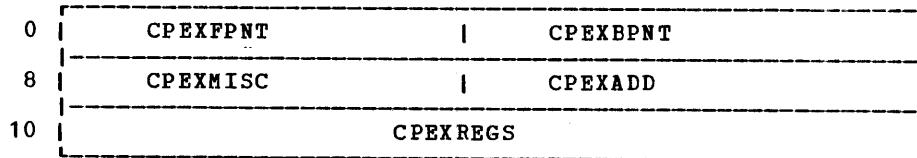
		ORG	CORBPNT	
4	CORLCNT	DS	1F	Page lock count for CORIOLCK

Entry Definition if Page Is In Transit

		ORG	CORFLAG		
8	CORCODE	DS	1X	C*1	DASD op code for DMKPAGIO

CPEXBLOK: CP EXECUTE BLOCK

CPEXBLOK maintains register values and a module address to handle a CP function that is
queued for the dispatcher.

```
      ┌─────────────────────────────────────────────────────────────┐
   0  │           CPEXFPNT            │          CPEXBPNT            │
      ├─────────────────────────────────────────────────────────────┤
   8  │           CPEXMISC            │          CPEXADD            │
      ├─────────────────────────────────────────────────────────────┤
  10  │                          CPEXREGS                           │
      └─────────────────────────────────────────────────────────────┘
```

Hexadecimal Displacement	Field Name			Field Description, Contents, Meaning
0	CPEXFPNT	DS	1F	Pointer to next CPEXBLOK
4	CPEXBPNT	DS	1F	Pointer to previous CPEXBLOK
8	CPEXMISC	DS	1F	Use varies with stacker
C	CPEXADD	DS	1F	Return address
10	CPEXREGS	DS	16F	Execute registers
	CPEXSIZE	EQU	(*-CPEXBLOK)/8	Size in doublewords (X'0A')
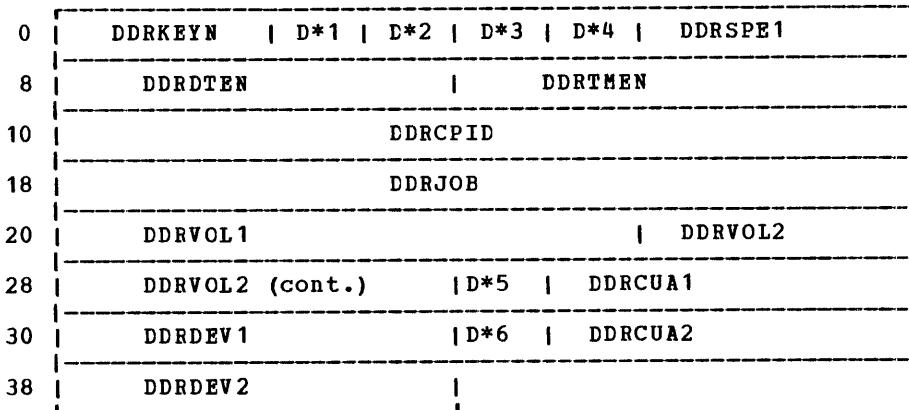
For CPEXREGS Area

		ORG	CPEXREGS
10	CPEXR0	DS	1F
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

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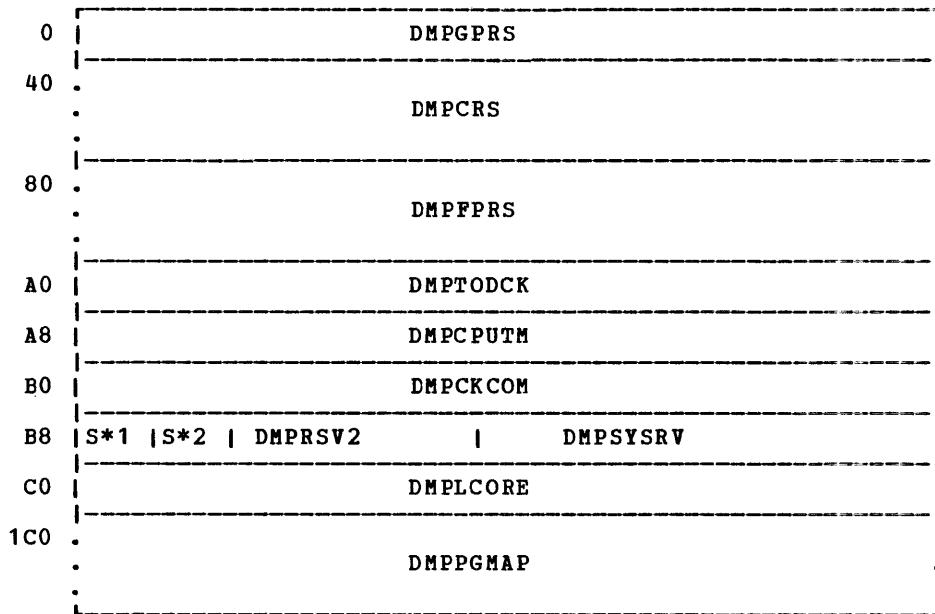
DDRREC is used in the SVC 76-initiated error recording process for type 60 DASD dump restore (DDR) dynamic device reallocation records. The reallocation records contain the replacement of the virtual "FROM" and "TO" control unit addresses (CUA) by the real addresses of the real DASD devices.

```
   ┌────────────────────────────────────────────────────────────────┐
 0 │    DDRKEYN    | D*1 | D*2 | D*3 | D*4 |    DDRSPE1              │
   ├────────────────────────────────────────────────────────────────┤
 8 │       DDRDTEN              |       DDRTMEN                      │
   ├────────────────────────────────────────────────────────────────┤
10 │                         DDRCPID                                │
   ├────────────────────────────────────────────────────────────────┤
18 │                         DDRJOB                                 │
   ├────────────────────────────────────────────────────────────────┤
20 │      DDRVOL1                            |    DDRVOL2           │
   ├────────────────────────────────────────────────────────────────┤
28 │      DDRVOL2 (cont.)        |D*5 |    DDRCUA1                  │
   ├────────────────────────────────────────────────────────────────┤
30 │      DDRDEV1                |D*6 |    DDRCUA2                  │
   ├────────────────────────────────────────────────────────────────┤
38 │      DDRDEV2                │
   └────────────────────────────┘
```

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	DDRDTEN	DS	1F		Date
C	DDRTMEN	DS	1F		Time
10	DDRCPID	DS	2F		CPU ID and model

Device Dependent Data

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 ID 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 ID TO DASD
35	DDRCUA2	DS	3X		Primary CUA of TO device
38	DDRDEV2	DS	4X		Device type of TO device

DMPINREC: DUMP FILE INFORMATION RECORD
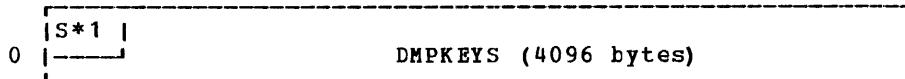
DMPINREC retains vital system register and storage location values necessary for the CPDUMP file. See also DMPKYREC.

```
      ┌──────────────────────────────────────────────────────────┐
   0  │                        DMPGPRS                            │
      │──────────────────────────────────────────────────────────│
  40  .                                                           .
      .                        DMPCRS                             .
      .                                                           .
      │──────────────────────────────────────────────────────────│
  80  .                                                           .
      .                        DMPFPRS                            .
      .                                                           .
      │──────────────────────────────────────────────────────────│
  A0  │                        DMPTODCK                           │
      │──────────────────────────────────────────────────────────│
  A8  │                        DMPCPUTM                           │
      │──────────────────────────────────────────────────────────│
  B0  │                        DMPCKCOM                           │
      │──────────────────────────────────────────────────────────│
  B8  │S*1 │S*2 │  DMPRSV2       │         DMPSYSRV               │
      │──────────────────────────────────────────────────────────│
  C0  │                        DMPLCORE                           │
      │──────────────────────────────────────────────────────────│
 1C0  .                                                           .
      .                        DMPPGMAP                           .
      .                                                           .
      └──────────────────────────────────────────────────────────┘
```

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		4 floating-point registers (if floating-point feature is installed.)
A0	DMPTODCK	DS	1D		Time-of-day clock
A8	DMPCPUTM	DS	1D		CPU timer
B0	DMPCKCOM	DS	1D		Time-of-day clock comparator
B8	DMPFLAG	DS	1X	S*1	Flag byte
	Bits defined in DMPFLAG				
	HALFPAGE	EQU	X'80'		Last record in DUMP file is 2K
B9	DMPRSV1	DS	1X	S*2	Reserved for IBM use
BA	DMPRSV2	DS	1H		Reserved for IBM use
BC	DMPSYSRV	DS	1F		System generated storage size
C0	DMPLCORE	DS	256X		Storage locations 0-256
1C0	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.

```
        r-------------------------------------------------------------------1
        |S*1 |                                                              |
  0     |----'              DMPKEYS (4096 bytes)                            |
        L_____J
```

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

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.

```
       r-----------------------------------------------------------------------1
   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                         |
       L-----------------------------------------------------------------------J
```
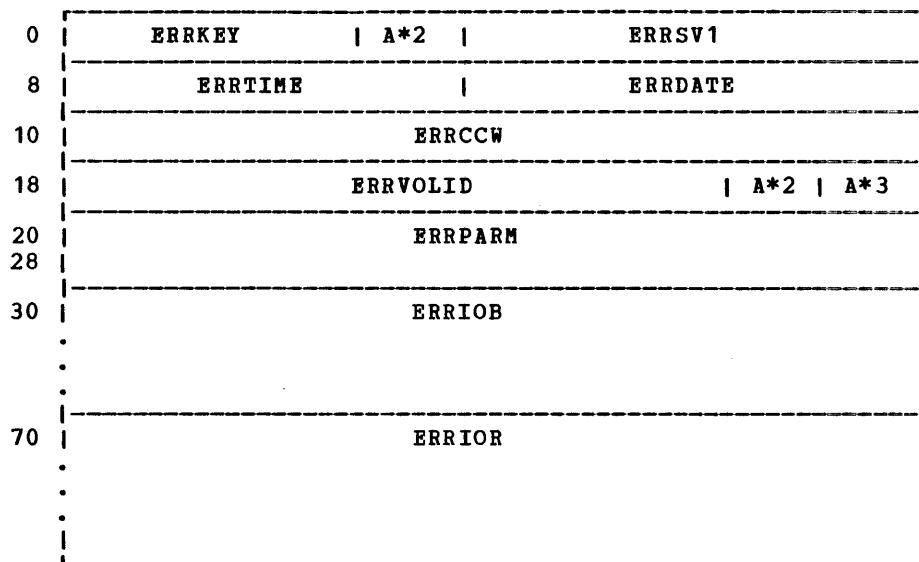
Hexadecimal Displacement	Field Name			Field Description, Contents, Meaning
0	EXTCR0	DS	1F	Virtual control register 0; architecture controls
4	EXTCR1	DS	1F	Virtual control register 1; segment table pointer
8	EXTCR2	DS	1F	Virtual control registers 2 through 15
C	EXTCR3	DS	1F	
10	EXTCR4	DS	1F	
14	EXTCR5	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	

ECBLOK

Hexadecimal Displacement	Field Name			Field Description, Contents, Meaning
40	EXTSHCR0	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	PER interrupt address
5C	EXTPERCD	DS	1H	PER interrupt code to be reflected
5E	EXTCOPY	DS	1H	Length code from active SEGTABLE entry
60	EXTCPTMR	DS	1D	Virtual CPU timer
68	EXTCPTRQ	DS	1F	Address of TRQBLOK for CPU timer
6C	EXTCCTRQ	DS	1F	Address of TRQBLOK for clock comparator
	EXTSIZE	EQU	(*-ECBLOK)/8	ECBLOK size in doublewords (X'0E')

ERRBLOK: ERROR BLOCK USED TO BUILD OBR/MDR

ERRBLOK contains data describing an error condition such as a channel failure or a device
failure.

```
        r---------------------------------------------------------------------¬
    0   |      ERRKEY       | A*2 |              ERRSV1                        |
        |---------------------------------------------------------------------|
    8   |       ERRTIME             |              ERRDATE                     |
        |---------------------------------------------------------------------|
   10   |                         ERRCCW                                       |
        |---------------------------------------------------------------------|
   18   |                   ERRVOLID                    | A*2 | A*3            |
        |---------------------------------------------------------------------|
   20   |                         ERRPARM                                      |
   28   |                                                                      |
        |---------------------------------------------------------------------|
   30   |                         ERRIOB                                       |
        .                                                                      .
        .                                                                      .
        .                                                                      .
        |---------------------------------------------------------------------|
   70   |                         ERRIOR                                       |
        .                                                                      .
        .                                                                      .
        .                                                                      .
        |                                                                      |
        L---------------------------------------------------------------------⌋
```

Hexadecimal Displacement	Field Name				Field Description, Contents, Meaning
0	ERRKEY	DS	3X		Key used to determine OBR/MDR processing
3	ERRSV1	DS	1X	A*1	Reserved for IBM use
4	ERRSV2	DS	1F		Reserved for IBM use
8	ERRTIME	DS	1F		Time record was built
C	ERRDATE	DS	1F		Date record was built
	ERRHEADR	EQU	(*-ERRBLOK)		Size of header in bytes
10	ERRCCW	DS	1D		Failing CCW
18	ERRVOLID	DS	6X		Volid of failing device
1E	ERRSDR	DS	1X	A*2	SDRFLAGS from SDRBLOK
1F	ERRCORR	DS	1X	A*3	Correlation count for MDR record
20	ERRPARM	DS	2D		Device dependent parameter string
30	ERRIOB	DS	(IOBSIZE)D		Copied IOBLOK
70	ERRIOER	DS	(IOERSIZE)D		Copied IOERBLOK
	ERRSIZE	EQU	(*-ERRSIZE)/8		Size of ERRBLOK in doublewords
		ORG	ERRCCW		
10	ERRMIOB	DS	(IOBSIZE)D		Copied IOBLOK
18	ERRMIOER	DS	(IOBRSIZE)D		Copied IOERBLOK
		ORG	ERRCCW		
10	ERRCCNT	DS	2X		Size of CONTASK data buffer
10	ERRCONT	DS	OC		CONTASK data buffer (variable length)

IOBLOK: I/O CONTROL BLOCK

IOBLOK 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.
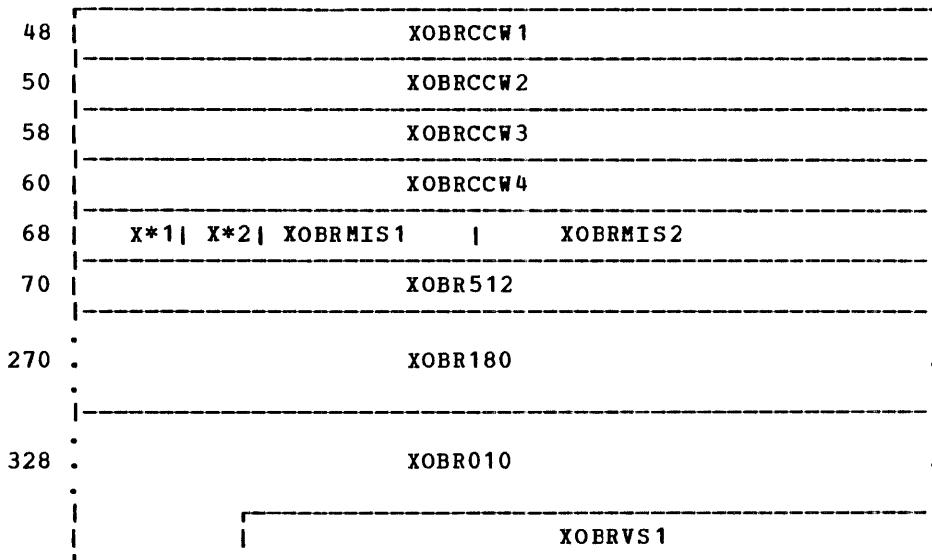
```
     ┌──────────────────────────────────────────────────────────────────────┐
  0  │      IOBRADD    |I*1 |I*2 |          IOBLINK                          │
     │──────────────────────────────────────────────────────────────────────│
  8  │      IOBFPNT              |          IOBBPNT                          │
     │──────────────────────────────────────────────────────────────────────│
 10  │    IOBCYL    |IOBVADD     |          IOBMISC                          │
     │──────────────────────────────────────────────────────────────────────│
 18  │      IOBUSER             |          IOBIRA                           │
     │──────────────────────────────────────────────────────────────────────│
 20  │      IOBCAW             |          IOBRCAW                           │
     │──────────────────────────────────────────────────────────────────────│
 28  │                        IOBCSW                                        │
     │──────────────────────────────────────────────────────────────────────│
 30  │      IOBIOER            |          IOBMISC2                          │
     │──────────────────────────────────────────────────────────────────────│
 38  │  I*3 |I*4 |   IOBRSV2    |          IOBRSV3                          │
     └──────────────────────────────────────────────────────────────────────┘
```

Hexadecimal Displacement	Field Name				Field Description, Contents, Meaning
0	IOBRADD	DS	1H		Real device address for SIO
2	IOBFLAG	DS	1X	I*1	IOBLOK flags

Bits defined in IOBFLAG

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'		IOBLOK 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	IOBLOK status

Bits defined in IOBSTAT

IOBFATAL	EQU	X'80'		Uncorrectable error in this I/O operation
IOBUC	EQU	X'40'		Unit check status
IOBSNSIO	EQU	X'20'		Sense operation (IOBSNSE)
IOBREQUE	EQU	X'10'		Restarted operation (IOBCAW)
IOBWRAP	EQU	X'08'		I/O task for autopoll wrap list
IOBCC0	EQU	X'00'		Processing I/O interrupt
IOBCC1	EQU	X'01'		Processing CC 1, CSW stored
IOBCC2	EQU	X'02'		Processing CC 2, channel busy
IOBCC3	EQU	X'03'		Processing CC 3, not available

Hexadecimal Displacement	Field Name				Field Description, Contents, Meaning
4	IOBLINK	DS	1F		Reserved for IBM use
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 doublewords (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 VMBLOK of user
1C	IOBIRA	DS	1F		IOBLOK 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
34	IOBMISC2	DS	1F		Use varies according to caller
38	IOBSPEC	DS	1X	I*3	IOBLOK special requests
	Bits defined in IOBSPEC				
	IOBTIO	EQU	X'80'		IOBLOK request for a TIO
	IOBHIO	EQU	X'40'		IOBLOK request for a HIO
	IOBSIOF	EQU	X'20'		Virtual SIO fast release
	IOBUNSL	EQU	X'08'		IOBLOK resulting from unsolicited interrupt
	IOBCOPY	EQU	X'04'		I/O block associated with a COPY request
39	IOBRSV1	DS	1X	I*4	Reserved for IBM use
3A	IOBRSV2	DS	1H		Reserved for IBM use
3C	IOBRSV3	DS	1F		Reserved for IBM use
	IOBSIZE	EQU	(*-IOBLOK)/8		IOBLOK size in doublewords (X'08')
	For CP IOBLOKs				
		ORG	IOBVADD		
12	IOBRCNT	DS	1H		Retry count

IOBR3211: EXTENDED OUTBOARD RECORDING BLOCK

IOBR3211 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.

```
      r----------------------------------------------------------------1
  48  |                        XOBRCCW1                                 |
      |----------------------------------------------------------------|
  50  |                        XOBRCCW2                                 |
      |----------------------------------------------------------------|
  58  |                        XOBRCCW3                                 |
      |----------------------------------------------------------------|
  60  |                        XOBRCCW4                                 |
      |----------------------------------------------------------------|
  68  |  X*1| X*2| XOBRMIS1       |        XOBRMIS2                     |
      |----------------------------------------------------------------|
  70  |                        XOBR512                                  |
      |----------------------------------------------------------------|
      .                                                                .
 270  .                        XOBR180                                 .
      .                                                                .
      |----------------------------------------------------------------|
      .                                                                .
 328  .                        XOBR010                                 .
      .                                                                .
      |                  r---------------------------------------------|
      |                  |                XOBRVS1                      |
      L------------------L---------------------------------------------J
```

Hexadecimal Displacement	Field Name			Field Description, Contents, Meaning
48	XOBRCCW1 DS	1D		CCW used to read OBR information
50	XOBRCCW2 DS	1D		CCW used to read OBR information
58	XOBRCCW3 DS	1D		CCW used to read OBR information
60	XOBRCCW4 DS	1D		CCW used to read OBR information
68	XOBRFLAG DS	1X	X*1	XOBRFLAG field

Bits defined in XOBRFLAG

XOBRT1	EQU	X'80'	T1 Buffer type information present
XOBRT2	EQU	X'40'	T2 Buffer type information present
XOBRT3	EQU	X'20'	T3 Buffer type information present

69	XOBRSTAT DS	1X	X*2	XOBRSTAT field

Bits defined in XOBRSTAT

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

Hexadecimal Displacement	Field Name			Field Description, Contents, Meaning
6A	XOBRMIS1	DS	1H	Used by the error routine
6C	XOBRMIS2	DS	1F	Used by the error routine
70	XOBR512	DS	CL512	Space for USCB data
270	XOBR180	DS	CL184	Space for FCB data
328	XOBR010	DS	CL10	Space for first ten error characters
332	XOBRSV1	DS	CL6	Reserved for IBM use
		ORG	XOBR180	
270	XOBR150	DS	CL150	Space for PLB check data
	.	ORG		
	XOBRSIZE	EQU	(*-IOERBLOK)/8	Size of IOER and XOBR in doublewords (X'67')
	XOBREXT	EQU	(*-XOBRCCW1)/8	Size of XOBR3211 in doublewords (X'5E')

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.

```
      ----------------------------------------------------------------
   0  |            IOERPNT              |          IOERLOC            |
      |--------------------------------------------------------------|
   8  |    IOERDW     |      IOERMSG         | I*1  | I*2  | I*3      |
      |--------------------------------------------------------------|
  10  |                           IOERADR                            |
      |--------------------------------------------------------------|
  18  |                           IOERCSW                            |
      |--------------------------------------------------------------|
  20  |                           IOERCCW                            |
      |--------------------------------------------------------------|
  28  |    IOEREXT    |    IOERCCH   | I*4 |        IOERSV1           |
      |--------------------------------------------------------------|
  30  |                           IOERDATA                           |
      .                                                              .
  48  .                                                              .
      |--------------------------------------------------------------|
      | Additional sense data area for devices that return           |
      | more than 24 sense bytes. See "IOBR3211:  Extended           |
      | Outboard Recording Block."                                   |
      .                                                              .
      .                                                              .
      .                                                              .
      ----------------------------------------------------------------
```

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 ERP and message writer
		ORG	IOERMSG	
A	IOERNUM	DS	1X	Message number for message writer
B	IOERIND3	DS	1X	Indicators for message writer
	Bits defined in IOERIND3			
	IOERIGN	EQU	X'80'	Allow IGNORE response
	IOERETRY	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

Hexadecimal Displacement	Field Name			Field Description, Contents, Meaning
C	IOERIND4 DS	1X		Indicators for message writer
	Bits defined in IOERIND4			
	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	IOERFLG1 field
	Bits defined in IOERFLG1			
	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	IOERFLG2 field
	Bits defined in IOERFLG2			
	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 written is active
	IOERCEMD EQU	X'02'		Intensive recording mode
	IOERVOL1 EQU	X'01'		DASD volid being read
F	IOERWRK DS	1X	I*3	Miscellaneous work area
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		
20	IOERVSER DS	CL6		Volume serial number for statistical data
26	IOERLEN DS	1H		Number of sense bytes present
28	IOEREXT DS	1H		Size of extended sense area in doublewords
2A	IOERCCH DS	1H		Size of I/O extended logout
2C	IOERFLG3 DS	1X		Flag field
	Bits defined in IOERFLG3			
	IOERread EQU	X'80'		SDR READ operation flag
2D	IOERSV1 DS	3X		Reserved for IBM use
30	IOERDATA DS	3D		Sense bytes associated with error
	IOERSIZE EQU	(*-IOERBLOK)/8		IOERBLOK size in doublewords (X'09')
	ORG	IOERDATA		Breakdown for channel check handler
30	IOERECSW DS	1F		ECSW information from channel logout
34	IOERCHAN DS	1X		Channel type flag

Note: Additional sense bytes may be appended to ERRBLOK; their format is described in "IOBR3211: Extended Outboard Recording Block" in this section.

IRMBLOK: INTENSIVE ERROR RECORDING MODE BLOCK
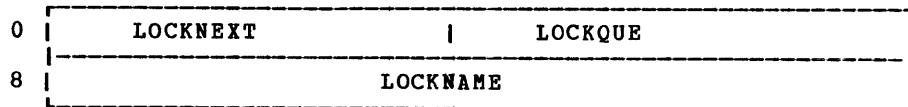
IRMBLOK provides the information necessary for the implementation of intensive recording mode via CP SET RECORD command. Intensive recording mode allows the recording of unit check errors from a specified device whose SENSE data matches the values selected.

```
     r-------------------------------------------------------------------¬
 0   |      IRMFWPTR            |    IRMRLADD  |    IRMLMT                 |
     |-----------------------------------------------------------------|
 8   |  I*1 | I*2 | I*3 | I*4 |    IRMLMTCT   |    I*5 | I*6             |
     L-------------------------------------------------------------------
```

Hexadecimal Displacement	Field Name			Field Description, Contents, Meaning
0	IRMFWPTR	DS	1F	Reserved for IBM use
4	IRMRLADD	DS	1H	Device address
6	IRMLMT	DS	1H	Limit count — every 'nth' record is requested.
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	Temporary 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

Bits defined in IRMFLG
IRMAND   EQU   X'80'   AND condition specified
IRMOR   EQU   X'40'   OR condition specified

IRMSIZE   EQU   (*-IRMBLOK)/8   IRMBLOK size in doublewords (X'02')

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.

```
   ┌──────────────────────────────────────────────────────────────┐
 0 │        LOCKNEXT             │          LOCKQUE                 │
   ├──────────────────────────────────────────────────────────────┤
 8 │                         LOCKNAME                               │
   └──────────────────────────────────────────────────────────────┘
```

Hexadecimal Displacement	Field Name			Field Description, Contents, Meaning
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: MACHINE CHECK SAVE AREA

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

```
     r--------------------------------------------------------------1
   0 |   MCDAMLEN    |            MCHRESEV                           |
     |-------------------------------------------------------------|
   8 | M*1  | M*2   | M*3   | M*4   | M*5   | M*6   | M*7   | M*8   |
     |-------------------------------------------------------------|
  10 |                       MCHLSUM                                |
     .                                                             .
     .                                                             .
     .                                                             .
     |-------------------------------------------------------------|
  38 | N*1  | N*2   | N*3   | N*4   | N*5   | N*6   | N*7   | N*8   |
     |-------------------------------------------------------------|
  40 |        MCHFSAR            |           MCHFSAV                |
     |-------------------------------------------------------------|
  48 |        MCHFSEAV           |           MCHPDARI               |
     |-------------------------------------------------------------|
  50 | L*1  | L*2   | L*3   | L*4   |  CPULIMIT    |    MCHRES1     |
     |-------------------------------------------------------------|
  58 |                       BUFDIA55                               |
     |-------------------------------------------------------------|
  60 |                       BUFENA55                               |
     |-------------------------------------------------------------|
  68 |                       ECCDIS55                               |
     |-------------------------------------------------------------|
  70 |                       ECCENA55                               |
     |-------------------------------------------------------------|
  78 |                       BUFDIA65                               |
     |-------------------------------------------------------------|
  80 |                       BUFENA65                               |
     |-------------------------------------------------------------|
  88 |                       ECCDIS65                               |
     |-------------------------------------------------------------|
  90 |                       ECCENA65                               |
     L--------------------------------------------------------------J
```

Hexadecimal Displacement	Field Name				Field Description, Contents, Meaning
0	MCDAMASS	DS	0D		Damage assessment
0	MCDAMLEN	DS	1H		Length of the damage assessment field
2	MCHRESEV	DS	XL6		Reserved for IBM use
8	MCHDAMFL	DS	0BL8		Damage assessment data
8	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 aborted
	MCHOTERM	EQU	X'08'	Operating system termination
	MCHOQUIT	EQU	X'04'	Quiet mode in effect

Hexadecimal Displacement	Field Name			Field Description, Contents, Meaning
9	MCHFLAG1 DS	1X	M*2	Damage area
	Bits defined in MCHFLAG1			
	MCH1MAIN EQU	X'80'		Main storage
	MCH1BUFF EQU	X'40'		Buffer
	MCH1COST EQU	X'20'		Control storage
	MCH1PROC EQU	X'08'		Processor
	MCH1TODC EQU	X'02'		Time-of-day clock
	MCH1SYSD EQU	X'01'		System damage
A	MCHFLAG2 DS	1X	M*3	Damage area (continued)
B	MCHFLAG3 DS	1X	M*4	Error type
	Bits defined in MCHFLAG3			
	MCH3INTE EQU	X'80'		Intermittent error
	MCH3SOLD EQU	X'40'		Solid error
	MCH3DATA EQU	X'20'		Data error
	MCH3PROT EQU	X'10'		Protect error
C	MCHFLAG4 DS	1X	M*5	RMS Action data
	Bits defined in MCHFLAG4			
	MCH4TOLO EQU	X'80'		Time-out loop
	MCH4REPA EQU	X'40'		Repair
	MCH4STRE EQU	X'20'		Storage reconfiguration
	MCH4BURE EQU	X'10'		Buffer reconfiguration
D	MCHFLAG5 DS	1X	M*6	RMS information status
	Bits defined in MCHFLAG5			
	MCH5INLG EQU	X'80'		Invalid logout
	MCH5INMC EQU	X'40'		Invalid machine check interrupt code
	MCH5IFSA EQU	X'20'		Invalid failing storage address
E	MCHFLAG6 DS	1X	M*7	RMS wait state suffix
F	MCHFLAG7 DS	1X	M*8	Reserved for IBM use
	Bits defined in MCHFLAG7			
	MCH7SMCR EQU	X'80'		Second machine check recursion
	MCH7VRTM EQU	X'40'		Terminate the virtual user
	MCH7OPSW EQU	X'10'		Machine check old PSW in problem state
	MCH7VEQR EQU	X'08'		Terminate the virtual=real user
10	MCHLSUM DS	1X		Summary
38	MCHPDAR DS	0BL8		
38	MCHPDAR0 DS	1X	N*1	Action taken
39	MCHPDAR1 DS	1X	N 2	Failure type

Machine check communication area must be cleared above this line

Bits defined in MCHPDAR1			
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
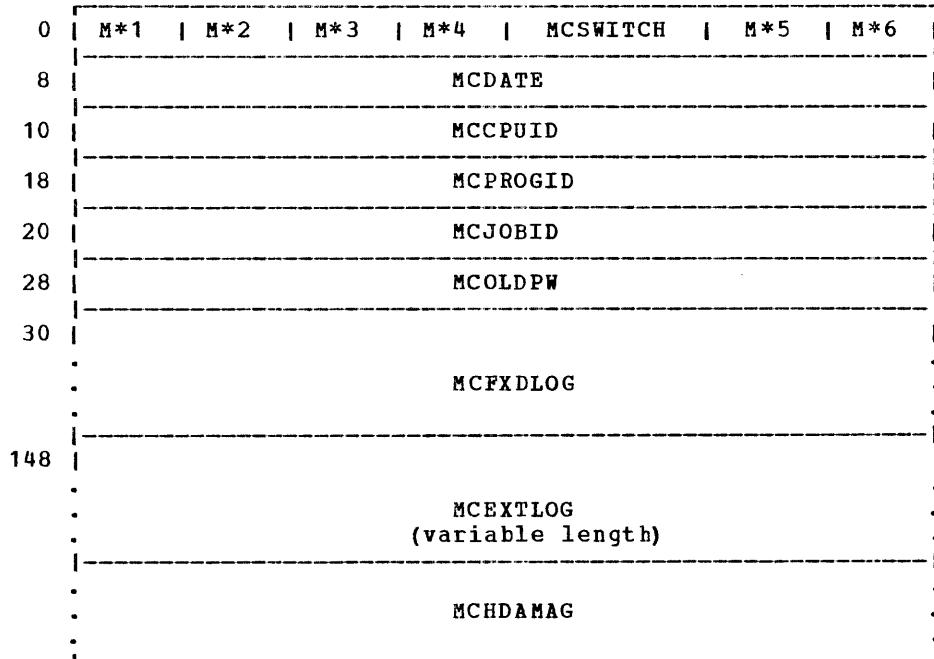
Hexadecimal Displacement	Field Name			Field Description, Contents, Meaning
3A	MCHPDAR2 DS	1X	N*3	Operating system status
3B	MCHPDAR3 DS	1X	N*4	Location of failure
3C	MCHPDAR4 DS	1X	N*5	Location of failure
3D	MCHPDAR5 DS	1X	N*6	Requested operator awareness
3E	MCHPDAR6 DS	1X	N*7	flag byte
	Bits defined in MCHPDAR6			
	MCHP6CBA EQU	X'80'		Change bit active
3F	MCHPDAR7 DS	1X	N*8	flag byte
	Bits defined in MCHPDAR7			
	MCH7STCK EQU	X'80'		Interfaces for DMSSTIKCP
	MCH7GSTR EQU	X'40'		Interfaces for DMKFRE
	MCH7PURG EQU	X'20'		Interfaces for DMKPTRFT
	MCH7LOGO EQU	X'10'		Reserved for IBM
	MCH7EXIT EQU	X'08'		Interfaces for exit to CP
	MCH7RSRE EQU	X'04'		Interfaces for DMKCFMBK and for DMKCFPRR and DMKPGSPO
	MCH7IOEM EQU	X'02'		Interfaces for the recorder
40	MCHFSAR DS	1F		Failing location real address
44	MCHFSAV DS	1F		Instruction address at failure
48	MCHFSEAV DS	1F		End of the failing location
4C	MCHPDARI DS	1F		End of failing virtual storage address
	MCHLEN1 EQU	*-MCDAMASS		Length of damage assessment area
	MCHLEN EQU	*-MCHRESEV		Length of area to be cleared
50	MCHMODEL DS	1X	L*1	The model number for the machine
	Bits defined in MCHMODEL			
	NOMODEL EQU	X'00'		No support for machine
	MODEL135 EQU	X'04'		ID number for the Model 135
	MODEL145 EQU	X'08'		ID number for the Model 145
	MODEL155 EQU	X'0C'		ID number for the Model 155
	MODEL158 EQU	X'0C'		ID number for the Model 158
	MODEL165 EQU	X'10'		ID number for the Model 165
	MODEL168 EQU	X'10'		ID number for the Model 168
51	SWITCH DS	1X	L*2	Main storage exercise switch

Mode Command Communication Area

52	MODEFLAG DS	1X	L*3	Flag field for MODE command
	Bits defined in MODEFLAG			
	MODEQUIT EQU	X'80'		ECC is in quiet mode
53	MODFLAG1 DS	1X	L*4	Flag field for message indicator in MODE command
	Bits defined in MODFLAG1			
	MOD1RETY EQU	X'80'		Message indicator for retry message
	MOD1QUIT EQU	X'40'		Message indicator for quiet message

Hexadecimal Displacement	Field Name			Field Description, Contents, Meaning
54	CPULIMIT	DS	1H	The count field for soft error
56	MCHRES1	DS	1H	Reserved for IBM use

DIAGNOSE Support for all Machine Models

158		DS	0D	
158	BUFDIA55	DC	X'0100D100'	Disable buffer for Model 155
160		DS	0D	
160	BUFENA55	DC	X'0200D100'	Enable buffer for Model 155
168		DS	0D	
168	ECCDIS55	DC	X'0300D100'	Disable ECC for Model 155
170		DS	0D	
170	ECCENA55	DC	X'0400D100'	Enable ECC for Model 155
178		DS	0D	
178	BUFDIA65	DC	X'0300000000000000'	Disable buffer for Model 165
180	BUFENA65	DC	X'0300002000000000'	Enable buffer for Model 165
188	ECCDIS65	DC	X'0200000003000000'	Disable ECC for Model 165
190	ECCENA65	DC	X'0200000000000000'	Enable ECC for Model 165
1	MCHFIX	EQU	280+48	Length of the fixed logout and header
1				record for machine check handler
1	MCHLEN2	EQU	*-MCDAMASS	Communication area length

MCRECORD: MACHINE CHECK HANDLER RECORD

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

```
      0 | M*1  | M*2  | M*3  | M*4  | MCSWITCH   | M*5  | M*6  |
        |------------------------------------------------------|
      8 |                     MCDATE                           |
        |------------------------------------------------------|
     10 |                     MCCPUID                          |
        |------------------------------------------------------|
     18 |                     MCPROGID                         |
        |------------------------------------------------------|
     20 |                     MCJOBID                          |
        |------------------------------------------------------|
     28 |                     MCOLDPW                          |
        |------------------------------------------------------|
     30 |                                                      |
        .                                                      .
        .                    MCFXDLOG                          .
        |------------------------------------------------------|
    148 |                                                      |
        .                                                      .
        .                    MCEXTLOG                          .
        .                  (variable length)                   .
        |------------------------------------------------------|
        .                                                      .
        .                    MCHDAMAG                          .
        .                                                      .
        L_____J
```

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		CPU ID
18	MCPROGID	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)		
148	MCEXTLOG	EQU	*		Machine check extended logout (the extended logout length is variable length and machine dependent)
	MCHDAMAG	EQU	*		The damage assessment area (80 bytes)

MDRREC: MISCELLANEOUS DATA RECORDING RECORD

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

```
     r--------------------------------------------------------------------------1
 0 |     MDRKEYN     | M*1 | M*2  | M*3 | M*4 |      MDRSPE1       |
    |------------------------------------------------------------------|
 8 |           MDRDTEN            |           MDRTMEN               |
    |------------------------------------------------------------------|
10 |                          MDRCPID                               |
    |------------------------------------------------------------------|
18 |    MDRCUA1     |           MDRVOL                              |
    |------------------------------------------------------------------|
20 |                          MDRSENS                               |
    .                                                                  .
    .                                                                  .
38 .                                                                  .
    L_____J
```

Hexadecimal Displacement	Field Name				Field Description, Contents, Meaning
	24-Byte Header				
0	MDRKEYN	DS	1H		Type and operating system
2	MDRSWS1	DD	1C	M*1	Switch byte
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	Bits 0 through 3 indicate the sequence number; bits 4 through 7 indicate the record total
6	MDRSPE1	DS	1H		
8	MDRDTEN	DS	1F		Date
C	MDRTMEN	DS	1F		Time
10	MDRCPID	DS	2F		CPU ID and model
	Device Dependent Data				
18	MDRCUA1	DS	2X		Primary CUA of device
1A	MDRVOL	DS	6X		Volume serial of device
20	MDRSENS	DS	24X		Sense byte data
	For 3270 Remote Support				
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

MICBLOK: VIRTUAL MACHINE POINTER LIST FOR VIRTUAL MACHINE ASSIST FEATURE
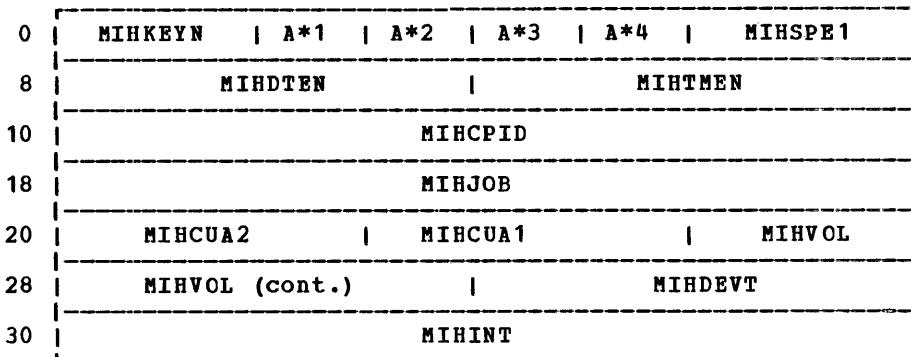
MICBLOK contains pointers to control registers, the segment table, and other values required by the virtual machine assist feature. This information is needed for the handling of certain instructions and privileged operations requested by the virtual machine.

```
    r-----------------------------------------------------------------1
0 |          MICRSEG           |          MICCREG             |
  |----------------------------------------------------------------|
8 |          MICVPSW           |          MICWORK             |
    L-----------------------------------------------------------------J
```

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
C	MICWORK	DS	1F	Workspace pointer
	MICSIZE	EQU	(*-MICBLOK)/8	Size of DSECT in doublewords (X'02')
		ORG	MICVPSW	
8	MICVIP	DS	1X	Virtual interrupt pending bit
	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

MIHREC: MISSING INTERRRUPT HANDLER ERROR RECORD
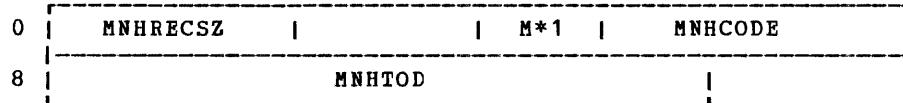
MIHREC block is used in the SVC 76-initiated error recording process of type 70 MIH (Missing Interrupt Handler) records.

```
    ┌────────────────────────────────────────────────────────────────┐
  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
	24-Byte Header				
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		CPU ID and model
	Device Dependent Data				
18	MIHJOB	DS	8X		Job whose I/O request pending
20	MIHCUA2	DS	3X		CUA used to address the device
23	MIHCUA1	DS	3X		Primary device address
26	MIHVOL	DS	6X		Volume serial
2C	MIHDEVT	DS	4X		Device type
30	MIHINT	DS	8X		Time interval used to check pending interrupt

MNHDR: VM MONITOR RECORD HEADER

MNHDR provides header information for following monitor records.

```
     r-----------------------------------------------------------1
0  |      MNHRECSZ       |          | M*1 |    MNHCODE        |
   |-----------------------------------------------------------|
8  |                     MNHTOD                    |
   L_____J
```

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		DS	1H		Reserved for IBM use
4	MNHCLASS	DS	1X	M*1	Monitor class
5	MNHCODE	DS	XL2		Monitor code
8	MNHTOD	DS	XL5		Current TOD value
	MNHDRLEN	EQU	*-MNHDR		Length of header record

MN000: VM MONITOR PERFORM CLASS RECORD

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

0	MN000WID		A0	MN000SI	MN000SF
8	MN000WPG		A8	MN000TI	MN000CI
10	MN000WIO		B0	MN000HI	MN000HD
18	MN000PRB		B8	MN000TC	MN000MN
20	MN000PSI	MN000CPA	C0	MN000MO	MN000LR
28	MN000NFL	MN000PSN	C8	MN000CP	MN000CH
30	MN000PRC	MN000RPC	D0	MN000TE	MN000CE
38	MN000SPC	MN000FLF	D8	MN000CT	MN000PE
40	MN000CPT	MN000SS	E0	MN000PT	MN000EP
48	MN000PFF	MN000PRF	E8	MN000IP	MN000PB
50	MN000PCS	MN000NXR	F0	MN000RR	MN000TCL
58	MN000CPR	MN000CVI	F8	MN000LCL	MN000CS
60	MN000CCW	MN000ITI	100	MN000CD	MN000HDI
68	MN000PTI	MN000CKI	108	MN000NDU	MN000NAU
70	MN000CSV	MN000CPG	110	MN000PRD	MN000PWR
78	MN000CIO	MN000CDS	118	MN000NPP	MN000SWS
80	MN000CDA	MN000CDB	120	MN000Q1N	MN000Q2N
88	MN000CSC	MN00EK	128	MN000Q1E | MN000Q2E | MN000INT | MN000PPA
90	MN000IK	MN000MS	130	MN000PPC	
98	MN000LP	MN000DI			

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

Section 1. CP Data Areas and Control Blocks 47

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 CPU 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 interruption handled
78	MN000CIO	DS	1F	No. of I/O interruption 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 DMKSCHDL
8C	MN000BK	DS	1F	Inst. count for X'08' SSK
90	MN000IK	DS	1F	Inst. count for X'09' ISK
94	MN000MS	DS	1F	Inst. count for X'80' SSM
98	MN000LP	DS	1F	Inst. count for X'82' LPSW
9C	MN000DI	DS	1F	Inst. count for X'83' DIAG
A0	MN000SI	DS	1F	Inst. count for X'9CX0' SIO
A4	MN000SF	DS	1F	Inst. count for X'9CX1' SIOF
A8	MN000TI	DS	1F	Inst. count for X'9DX0' TIO
AC	MN000CI	DS	1F	Inst. count for X'9DX1' CLRIO
B0	MN000HI	DS	1F	Inst. count for X'9EX0' HIO
B4	MN000HD	DS	1F	Inst. count for X'9EX1' HDV
B8	MN000TC	DS	1F	Inst. count for X'9F' TCH
BC	MN000MN	DS	1F	Inst. count for X'AC' STNSM
C0	MN000MO	DS	1F	Inst. count for X'AD' STOSM
C4	MN000LR	DS	1F	Inst. count for X'B1' LRA
C8	MN000CP	DS	1F	Inst. count for X'B202' STIDP
CC	MN000CH	DS	1F	Inst. count for X'B203' STIDC
D0	MN000TE	DS	1F	Inst. count for X'B204' SCK
D4	MN000CE	DS	1F	Inst. count for X'B206' SCKC
D8	MN000CT	DS	1F	Inst. count for X'B207' STCKC
DC	MN000PE	DS	1F	Inst. count for X'B208' SPT
E0	MN000PT	DS	1F	Inst. count for X'B209' STPT
E4	MN000EP	DS	1F	Inst. count for X'B20A' SPKA
E8	MN000IP	DS	1F	Inst. count for X'B20B' IPK
EC	MN000PB	DS	1F	Inst. count for X'B20D' PTLB
F0	MN000RR	DS	1F	Inst. count for X'B213' RRB
F4	MN000TCL	DS	1F	Inst. count for X'B6' STCTL
F8	MN000LCL	DS	1F	Inst. count for X'B7' LCTL
FC	MN000CS	DS	1F	Inst. count for X'BA' CS
100	MN000CD	DS	1F	Inst. count for X'BB' 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 (secs)
12E	MN000PPA	DS	1H	Pseudo—cylinders[1] of allocated temporary space
130	MN000PPC	DS	1H	Pseudo—cylinders of system temporary space

---

[1] A pseudo-cylinder comprises 100 available page slots for all system-owned volumes. The total of available pseudo-cylinders per device is determined by the formula:
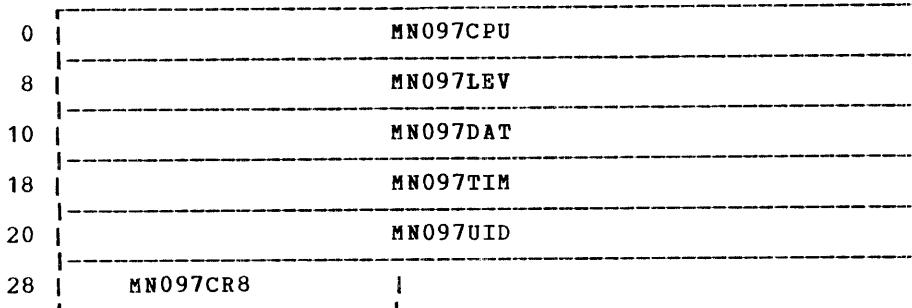
$$\frac{(\text{number of cylinders} \times \text{number of records per cylinder for a device})}{100}$$
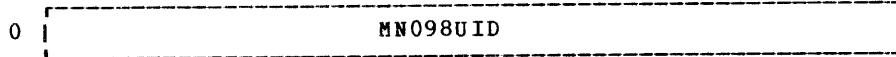
MN097 AND MN098: VM MONITOR TAPE HEADER AND TRAILER RECORDS

MN097 provides tape header information for data accumulated by VM monitor.

```
       r-----------------------------------------------------------1
   0   |                        MN097CPU                           |
       |-----------------------------------------------------------|
   8   |                        MN097LEV                           |
       |-----------------------------------------------------------|
  10   |                        MN097DAT                           |
       |-----------------------------------------------------------|
  18   |                        MN097TIM                           |
       |-----------------------------------------------------------|
  20   |                        MN097UID                           |
       |-----------------------------------------------------------|
  28   |      MN097CR8          |
       L------------------------J
```

Hexadecimal Displacement	Field Name			Field Description, Contents, Meaning
0	MN097CPU	DS	XL8	CPU serial/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
	MN097LEN	EQU	*-MN097	Length of the tape header record
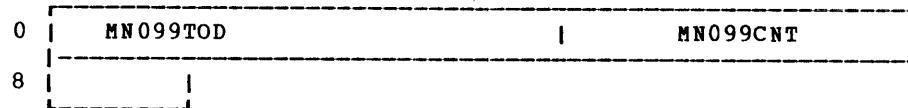
MN098 contains the userid of the user who has terminated current VM monitor activity.

```
       r-----------------------------------------------------------1
   0   |                        MN098UID                           |
       L-----------------------------------------------------------J
```

Hexadecimal Displacement	Field Name			Field Description, Contents, Meaning
0	MN098UID	DS	CL8	User stopping the VM monitor
	MN098LEN	EQU	*-MN098	Length of the tape trailer record

MN099: VM MONITOR SUSPENSION RECORD
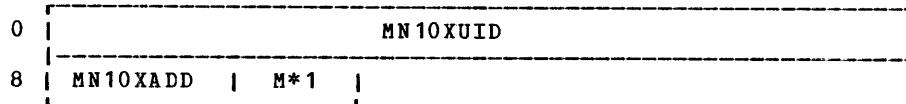
MN099 contains information recorded when monitor activity is suspended because of a tape busy condition.

```
     r----------------------------------------------------------1
0  |    MN099TOD                         |      MN099CNT        |
   |----------------------------------------------------------|
8  |            |
   L_____J
```

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

MN10X: VM MONITOR RESPONSE CLASS RECORDS

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

```
     r----------------------------------------------------------1
0  |                    MN10XUID                              |
   |----------------------------------------------------------|
8  | MN10XADD    |  M*1  |
   L_____J
```

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
	MN10YLEN	EQU	*-MN10X		Long record, header length
B	MN10YIO	EQU	*		Input/output line starts here

MN20X: VM MONITOR SCHEDULE CLASS RECORDS

MN20X contains VM 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   │   MN202PRI   │    M202PRG       │   MN202APR    │  MN202REF  │
      ├──────────────┼──────────────────┼────────────────────────────┤
 38   │   MN202RES   │    M202PST       │         MN202IOC           │
      ├──────────────┴──────────────────┼────────────────────────────┤
 40   │          MN202PNC               │         MN202LIN           │
      ├─────────────────────────────────┼────────────────────────────┤
 48   │          MN202CRD               │                            │
      └─────────────────────────────────┴────────────────────────────┘
```

Hexadecimal Displacement	Field Name			Field Description, Contents, Meaning
0	MN20XUID	DS	CL8	Userid being added/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 being added/dropped to/from
1F	MN2RSV1	DS	1X M*2	Reserved for IBM use
	MN203LEN	EQU	*-MN20X	Length of class 2 code 3 Record (Add queue)
20	MN20YTTI	DS	XL8	Current VMTTIME (CP simulation time)
28	MN20YVTI	DS	XL8	Current VMVTIME (user virtual time)
30	MN204PRI	DS	1F	Eligible list priority
	MN204LEN	EQU	*-MN20X	Length of class 2 code 4 Record (Drop queue)
		ORG	MN204PRI	Back up to priority field
30	MN202PRI	DS	1H	Dispatch priority
32	MN202PGR	DS	1H	Pages read while in queue
34	MN202APR	DS	1H	Av. no. of pages resident at each paging operation
36	MN202REF	DS	1H	No. of pages referenced while in queue
38	MN202RES	DS	1H	Current no. 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
	MN202LEN	EQU	*-MN20X	Length of class 2 code 2 Record (add to the eligible list)

MN400: VM MONITOR USER CLASS RECORD

MN400 provides user virtual machine statistics.

```
     ┌────────────────────────────────────────────────────────────────┐
  0  │                          MN400UID                              │
     │────────────────────────────────────────────────────────────────│
  8  │                          MN400TTI                              │
     │────────────────────────────────────────────────────────────────│
 10  │                          MN400VTI                              │
     │────────────────────────────────────────────────────────────────│
 18  │                          MN400PGR                              │
     │────────────────────────────────────────────────────────────────│
 1C  │                          MN400PGW                              │
     │────────────────────────────────────────────────────────────────│
 20  │                          MN400IOC                              │
     │────────────────────────────────────────────────────────────────│
 24  │                          MN400PNC                              │
     │────────────────────────────────────────────────────────────────│
 28  │                          MN400LIN                              │
     │────────────────────────────────────────────────────────────────│
 2C  │                          MN400CRD                              │
     │────────────────────────────────────────────────────────────────│
 30  │  MN400RST    │  MN400DST  │  MN400OST    │    MN400QST          │
     │────────────────────────────────────────────────────────────────│
 34  │  MN400PST    │  MN400EST  │  MN400TST    │    MN400MLV          │
     │────────────────────────────────────────────────────────────────│
 38  │  MN400QLV    │  MN400CLV  │  MN400TLV    │    MN400PND          │
     │────────────────────────────────────────────────────────────────│
 3C  │  MN400UPR    │ MN4RSV1    │          MN400RES                  │
     │────────────────────────────────────────────────────────────────│
 40  │     MN400WSS           │          MN400PDR                     │
     │────────────────────────────────────────────────────────────────│
 44  │     MN400PDK           │          MN400INT                     │
     └────────────────────────────────────────────────────────────────┘
```

Hexadecimal Displacement	Field Name			Field Description, Contents, Meaning
0	MN400UID	DS	CL8	Userid
8	MN400TTI	DS	XL8	Current VMTTIME (in VMBLOK); CP simulation time
10	MN400VTI	DS	XL8	Current VMVTIME (in VMBLOK); user simulation time
18	MN400PGR	DS	1F	Total page reads – this user
1C	MN400PGW	DS	1F	Total page writes – this user
20	MN400IOC	DS	1F	Virtual nonspooled 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	User running status
31	MN400DST	DS	1X	User dispatch status
32	MN400OST	DS	1X	User operating status
33	MN400QST	DS	1X	User queuing status
34	MN400PST	DS	1X	User processing status
35	MN400EST	DS	1X	User execution status
36	MN400TST	DS	1X	User tracing control status
37	MN400MLV	DS	1X	User message level
38	MN400QLV	DS	1X	User queue level

Hexadecimal Displacement	Field Name			Field Description, Contents, Meaning
39	MN400CLV	DS	1X	User command level
3A	MN400TLV	DS	1X	User timer level
3B	MN400PND	DS	1X	Interrupt pending status
3C	MN400UPR	DS	1X	Directory or SET priority
3D	MN4RSV1	DS	1X	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	MN400LEN	EQU	*-NM400	Length of CL4 code 0 record
48	MN400INT	DS	1H	Monitor sampling interval (in seconds)

MN500: VM MONITOR INSTRUCTION SIMULATION CLASS RECORD

MN500 provides data on instructions simulated by CP.

```
     r------------------------------------------------------------,
  0  |                        MN500UID                            |
     |------------------------------------------------------------|
  8  |                        MN500INS                            |
     |------------------------------------------------------------|
  C  |                        MN500VAD                            |
     |------------------------------------------------------------|
 10  |                        MN500OVH                            |
     L------------------------------------------------------------
```

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 addr of the instruction
10	MN500OVH	DS	XL8	Current total of CP simulation time
	MN500LEN	EQU	*-MN500	Length of CL4 code 0 record

MN600DEV: VM MONITOR DASTAP CLASS DEVICE PORTION

MN600DEV provides I/O activity count for DASD and tape devices as invoked by CP monitor function.

```
     r------------------------------------------------------------,
  0  |  MN600ADD   |  MN600TY    |         MN600SER               |
     |------------------------------------------------------------|
  8  |             |                    MN600CNT                  |
     L------------------------------------------------------------
```

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

MN600HDR: VM MONITOR HEADER RECORD FOR DEVICE PACKAGES
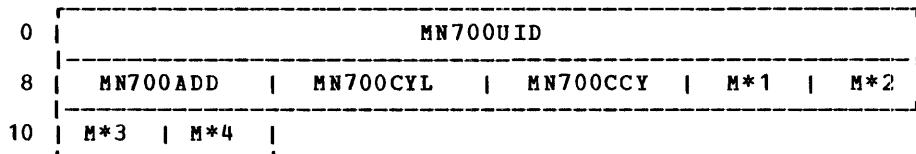
MN600HDR provides the number of device data packages.

```
   ┌───────────────────────────────────────────────────────────┐
 0 │   MN600NUM   │                    │                        │
   └───────────────────────────────────────────────────────────┘
```

Hexadecimal Displacement	Field Name			Field Description, Contents, Meaning
0	MN600NUM	DS	H	Number of device data packages

MN700: VM MONITOR SEEKS CLASS RECORD

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

```
   ┌───────────────────────────────────────────────────────────┐
 0 │                         MN700UID                           │
   │───────────────────────────────────────────────────────────│
 8 │   MN700ADD   │   MN700CYL   │   MN700CCY   │  M*1  │  M*2  │
   │───────────────────────────────────────────────────────────│
10 │  M*3  │  M*4  │
   └───────────────┘
```

Hexadecimal Displacement	Field Name				Field Description, Contents, Meaning
0	MN700UID	DS	CL8		Userid
8	MN700ADD	DS	1H		Device address
A	MN700CYL	DS	1H		Cylinder seeking to
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

MN802CTR: VM MONITOR SYSPROF CLASS RECORD

MN802CTR provides, via CP monitor function, 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, plus total system I/O, page wait, and problem state time.

```
       r--------------------------------------------------------------------¬
   0 |         MN802NAU         |         MN802PGR                  |
       |--------------------------------------------------------------------|
   8 |         MN802PGW         |         MN802NPP                  |
       |--------------------------------------------------------------------|
  10 |                      MN802WID                              |
       |--------------------------------------------------------------------|
  18 |                      MN802WPG                              |
       |--------------------------------------------------------------------|
  20 |                      MN802WIO                              |
       |--------------------------------------------------------------------|
  28 |                      MN802PRB                              |
       L--------------------------------------------------------------------
```

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

MONCOM: VM MONITOR COMMUNICATIONS AREA
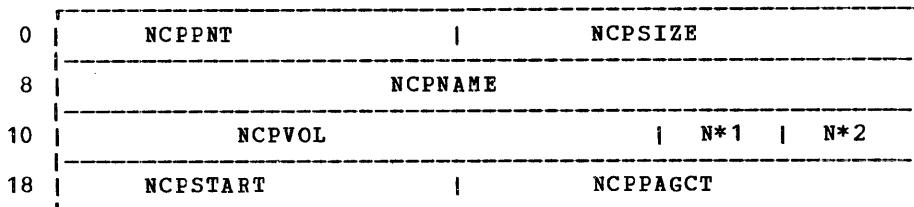
MONCOM provides the control link for CP's monitor activity, the user, and the tape drive.

```
      r------------------------------------------------------------------
   0  |         MONCPEX         |  M*1  |  M*2  |    MONDVNUM           |
      |------------------------------------------------------------------
   8  |        MONDVLST         |              MCNARDB                  |
      |------------------------------------------------------------------
  10  |        MONAIOB          |              MONATRB                  |
      |------------------------------------------------------------------
  18  |                        MONCLOCK                                 |
      |------------------------------------------------------------------
  20  |                        MONSUSCK                                 |
      |------------------------------------------------------------------
  28  |       MONSUSCT          |             MONRSVD1                  |
      |------------------------------------------------------------------
  30  |       MONCTEB1          |             MONCTEB2                  |
      |------------------------------------------------------------------
  38  |                        MONSAVE                                  |
      .                                                                 .
      .                                                                 .
      .                                                                 .
      |------------------------------------------------------------------
  78  |                        MONUSER                                  |
      L------------------------------------------------------------------
```

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
	Bits defined in MONFLAG1				
	TBUSY	EQU	X'80'		Tape is busy
	SUSPEND	EQU	X'40'		Monitoring suspended
	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 - stop VM monitor
	MONTIINT	EQU	X'04'		Handling timer interruption
5	MONFLAG2	DS	1X	M*2	Work byte
6	MONDVNUM	DS	1H		Number of entries in real device list
8	MONDVLST	DS	1F		Address of the real device list
C	MCNRSV1	DS	1F		Reserved for IBM use
10	MONAIOB	DS	1F		Address of monitor tape I/O block
14	MCNATRB	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	MCNRSVD1	DS	1F		Reserved for IBM use
30	MONCTEB1	DS	1F		CORTABLE entry for buffer page 1
34	MONCTEB2	DS	1F		CORTABLE entry for buffer page 2
38	MONSAVE	DS	16F		Monitor internal save area
78	MCNUSER	DS	8C		User starting/stopping the VM monitor

NCPTBL: NAMED 3704/3705 CONTROL PROGRAM TABLE

NCPTBL retains the information requirements for loading the saved image 3704/3705 NCP, EP, or PEP program into a 3704 or 3705 communications controller.

```
      ┌─────────────────────────────────┬─────────────────────────────────┐
  0   │            NCPPNT               │            NCPSIZE              │
      ├─────────────────────────────────┴─────────────────────────────────┤
  8   │                            NCPNAME                                 │
      ├─────────────────────────────────────────────┬────────┬────────────┤
 10   │            NCPVOL                            │  N*1   │   N*2      │
      ├─────────────────────────────────────────────┴────────┴────────────┤
 18   │            NCPSTART             │            NCPPAGCT             │
      └─────────────────────────────────┴─────────────────────────────────┘
```

Hexadecimal Displacement	Field Name				Field Description, Contents, Meaning
0	NCPPNT	DS	1F		Displacement to next entry
4	NCPSIZE	DS	1F		3704/3705 storage size required for load
8	NCPNAME	DS	CL8		Control program reference name
10	NCPVOL	DS	CL6		Volid of DASD containing saved image
16	NCPFLAG	DS	1X	N*1	CPTYPE flag byte
	Bits defined in NCPFLAG				
	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		CCPD of first page on NCPVOL
1C	NCPPAGCT	DS	1F		Total number of pages saved

NICBLOK: NETWORK INTERFACE CONTROL BLOCK

NICBLOK contains control information related to 3704/3705 resources, teleprocessing lines, and display screen status information.

```
      r------------------------------------------------------------1
   0  | NICNAME   |  NICEPAD   | N*1 | N*2 | N*3 | N*4 |
      |------------------------------------------------------------|
   8  | NICRCNT   |  NICVRID   |       NICTMAT            |
      |------------------------------------------------------------|
  10  |      NICUSER          |       NICQPNT            |
      L_____J
```

Hexadecimal Displacement	Field Name				Field Description, Contents, Meaning
0	NICNAME	DS	1H		3704/3705 NCP resource name
2	NICEPAD	DS	1H		Subchannel address when in EP mode
4	NICSTAT	DS	1X	N*1	Resource status flags

Bits defined in NICSTAT

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
NICHOLD	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

Bits defined in NICFLAG

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 now

Hexadecimal Displacement	Field Name				Field Description, Contents, Meaning
6	NICLLEN	DS	1X	N*3	Terminal output line length
7	NICTYPE	DS	1X	N*4	Resource type/features

Bits defined in NICTYPE

NICCTLR	EQU	X'00'		Resource is the 3704/3705
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
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 control unit
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	VMBLOK address of associated user
14	NICQPNT	DS	1F	Pointer to input BTU chain

Device Dependent Data - 3271 and 3275 on Binary Synchronous Lines

		ORG	NICEPAD	
2	NICCORD	DS	1X	Current line coordinates
3	NICTMCD	DS	1X	Terminal mode

Bits defined in NICTMCD

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 on for 3270 remote

		ORG	NICRCNT	
8	NICSELT	DS	1H	Remote station selection character
A	NICPOLL	DS	1H	Remote station polling characters
C	NICATRB	DS	1F	Timer request block address

NICSIZE	EQU	(*-NICBLOK)/8	Size of block in doublewords (X'03')

Equate Symbols for VM/370 Support of the 3704/3705

WRITBRK	EQU	X'09'	Write break CCW op code
RDBUFLN	EQU	96	Length of host read buffers
RDBUFNO	EQU	6	Number of host read buffers

Sense Bits (sense byte 0) Peculiar to the 3704/3705

IPLREQ	EQU	X'02'	IPL required--3705 inactive
ABORT	EQU	X'01'	Buffer depletion--transfer terminated

OBRRECN: UNIT CHECK ERROR RECORD (LONG OBR)

OBRRECN provides error, sense, and other statistical data needed for error recording on a specified channel-attached I/O device.
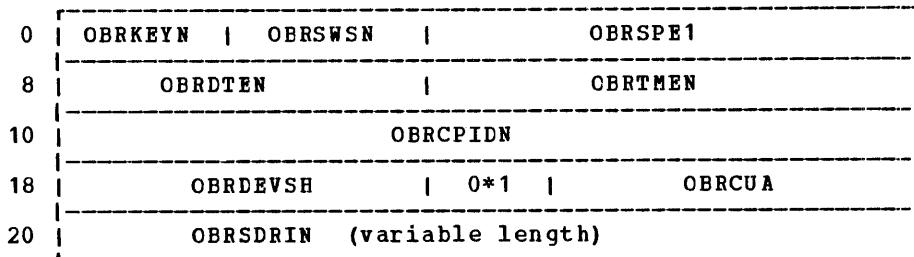
```
      ┌──────────────────────────────────────────────────────────────┐
   0  | OBRKEYN  |  OBRSWSN  |            OBRSPE1                      |
      |──────────────────────────────────────────────────────────────|
   8  |       OBRDTEN         |            OBRTMEN                     |
      |──────────────────────────────────────────────────────────────|
  10  |                       OBRCPIDN                                |
      |──────────────────────────────────────────────────────────────|
  18  |                       OBRPGMN                                 |
      |──────────────────────────────────────────────────────────────|
  20  |                       OBRFCCWN                                |
      |──────────────────────────────────────────────────────────────|
  28  |                       OBRCSWN                                 |
      |──────────────────────────────────────────────────────────────|
  30  |  S*1  |  OBRCUAIN    |              OBRDEVTN                   |
      |──────────────────────────────────────────────────────────────|
  38  |  S*2  |  OBRCUAPR   |  OBRIORTY  |    OBRSCNCT                 |
      |──────────────────────────────────────────────────────────────|
  40  |                  DEVICE DEPENDENT DATA                        |
      └──────────────────────────────────────────────────────────────┘
```


Hexadecimal Displacement	Field Name			Field Description, Contents, Meaning

24-Byte Header Record

| 0 | OBRKEYN | DS | 1H | Record type |
| 2 | OBRSWSN | DS | 1H | Switches |

Bits defined in OBRSWSN and Byte 0

	OBRMORE	EQU	X'80'	More records to follow
	OBRTOD	EQU	X'40'	Time-of-day clock instruction issued
	OBRTIME	EQU	X'08'	TIME macro used

Bits defined in OBRSWSN Byte 1

	OBREOD	EQU	X'80'	SDR counters dumped at EOD
	OBRTEMP	EQU	X'40'	Temporary error
	OBRSHOBR	EQU	X'20'	Short record
	OBRDEMNT	EQU	X'04'	Volume demount
4	OBRSPE1	DS	1F	Reserved for IBM use
8	OBRDTEN	DS	1F	Date
C	OBRTMEN	DS	1F	Time
10	OBRCPIDN	DS	2F	CPU ID and serial number

End of 24-Byte Header Record

18	OBRPGMN	DS	2F		Job ID
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
34	OBRDEVTN	DS	1F		Device type
38	OBRSDRCT	DS	1X	S*2	Number of SDR work area bytes
39	OBRCUAPR	DS	3C		Primary unit address

Hexadecimal Displacement	Field Name		Field Description, Contents, Meaning
3C	OBRICRTY DS	2X	Number of retries
3E	OBRSNSCT DS	2X	Number of sense bytes

Format of Device Dependent Data

All DASD Devices Format
40	OBRVOLN DS	8C	Volume ID
48	OBRLSKN DS	8X	Last seek address
50	OBRHAN DS	8X	Home address

2314/2319 Format
58	OBRSDRWK DS	10X	SDR work area
62	OBRSENSN DS	24C	Sense data
7C	OBRSIOCT DS	1F	Count of SIO's since last recording

3350/3340/3330/2305 Format
	ORG	OBRSKN	
58	OBR33SNS DS	24C	3350/3340/3330/2305 sense data

Unit Record Format
	ORG	OBRVOLN	
40	OBRURST DS	10X	SDR work area
4A	OBRURSNS DS	1C	Unit record sense data

3505/3525 Format
	ORG	OBRVOLN	
40	OBR3505S DS	1C	3505/3525 sense data

3211 Format
	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

2400 Tape Format
	ORG	OBRSKN	
48	OBRTAPST DS	10X	SDR work area
52	OBRTAPSN DS	24C	Tape sense data

3420/3410 Tape Format
	ORG	OBRLSKN	
48	OBRDVDEP DS	16C	Device dependent data
58	OBR342ST DS	20X	SDR work area
6C	OBR3420S DS	24C	3420 sense data

Short Outboard Record Format
	ORG	OBRSDRWK	
18	OBRDEVSH DS	1F	Device type
1C	OBRSDRSH DS	1X	No. of SDR work area bytes
1D	OBRCUA DS	3X	Channel and unit address
20	OBRSDRIN DS	20C	SDR work area

OBRRECN: UNIT CHECK ERROR RECORD (SHORT OBR)

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

```
        r---------------------------------------------------------------1
     0  | OBRKEYN   |  OBRSWSN    |           OBRSPE1                    |
        |--------------------------------------------------------------|
     8  |      OBRDTEN           |           OBRTMEN                     |
        |--------------------------------------------------------------|
    10  |                      OBRCPIDN                                 |
        |--------------------------------------------------------------|
    18  |      OBRDEVSH          |  0*1  |        OBRCUA                 |
        |--------------------------------------------------------------|
    20  |      OBRSDRIN   (variable length)                            |
        L_____J
```

Hexadecimal Displacement	Field Name				Field Description, Contents, Meaning
0	OBRKEYN	DS	1H		Record type
2	OBRSWSN	DS	1H		Switches
	Bits defined in OBRSWSN				
Byte 0					
	OBRMORE	EQU	X'80'		More records follow
	OBRTOD	EQU	X'40'		Time-of-day clock instruction issued
	OBRTIME	EQU	X'08'		TIME macro used
Byte 1					
	OBREOD	EQU	X'80'		SDR counters dumped at EOD
	OBRTEMP	EQU	X'40'		Temporary error
	OBRSHOBR	EQU	X'20'		Short record
	OBRDEMNT	EQU	X'04'		Volume demount
4	OBRSPE1	DS	1F		
8	OBRDTEN	DS	1F		Date
C	OBRTMEN	DS	1F		Time
10	OBRCPIDN	DS	2F		CPU ID and serial number
	End of 24-Byte Header Record				
18	OBRDEVSH	DS	1F		Device type
1C	OBRSDRSH	DS	1X	0*1	Number of SDR work area bytes
1D	OBRCUA	DS	3X		Channel and unit address
20	OBRSDRIN	DS	20C		SDR work area

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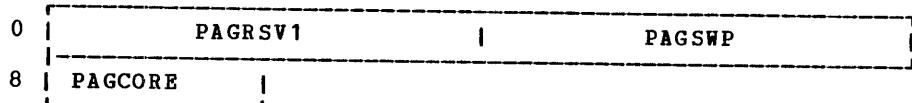
OWNDLIST contains a list of all the system owned DASD volumes that are used for paging, spooling, and temporary disk storage activity. Each entry specifies the volume identity and its preferred use (that is, paging/spooling/T-disk space). This block is generated by the SYSOWN macro at system generation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

```
0 |----------------------------------------|-------------------|
  |              OWNDVSER                  |     OWNDRDEV      |
  |_____|_____|
```

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

PAGTABLE: 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.

```
0 |------------------------------------|-------------------------|
  |            PAGRSV1                 |         PAGSWP          |
  |------------------------------------|-------------------------|
8 | PAGCORE            |
  |_____|
```
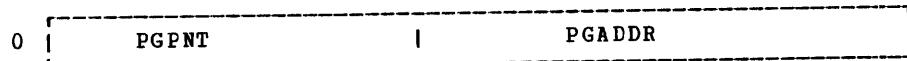
Hexadecimal Displacement	Field Name			Field Description, Contents, Meaning
	8-Byte Header			
0	PAGRSV1	DS	1F	Reserved to align PAGCORE on a doubleword
4	PAGSWP	DS	1F	Pointer to SWPTABLE
8	PAGCORE	DS	1H	Real page address; start of page table
	Bits defined in PAGCORE+1			
	PAGINVAL	EQU	X'08'	PAGTABLE entry invalid
	PAGREF	EQU	X'01'	Page has been referenced

Section 1. CP Data Areas and Control Blocks 65

PGBLOK: PSEUDO PAGE FAULT STACK BLOCK

PGBLOK 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.

```
      r------------------------------------------------------------------1
   0  |         PGPNT              |           PGADDR                     |
      L_____J
```

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

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 power and features of the processor, save areas used by BALR and FREER. This block also contains monitor and trace data and the necessary linkages to virtual machines, real devices, and spool files.

Page 0, Machine Usage

Addr			
0	IPLPSW		IPLCCW1
10	IPLCCW2		EXOPSW
20	SVCOPSW		PROPSW
30	MCOPSW		IOOPSW
40	CSW		CAW QUANTUMR
50	TIMER QUANTUM		EXNPSW
60	SVCNPSW		PRNPSW
70	MCNPSW		IONPSW
80	CPULOG		
100	FXDLOG		
160	FPRLOG		
180	GPRLOG		
1C0	CRLOG		
200	TEMPSAVE		
240	BALRSAVE		
280	FREESAVE		
2C0	FREEWORK		
2F0	DATE		TODATE
300	STARTIME		CPUID
310	IDLEWAIT		PAGEWAIT

Addr				
320	IONTWAIT		PROBTIME	
330	RUNPSW		RUNUSER	DSPLPSW
340	RUNCR0	RUNCR1	CPSTAT	CPRESTRT
350	PGREAD	PGWRITE	PGWAITIM	
360	PGWAITPG		PSASVCCT	P*1 P*2
370	CPID	CPABEND	P*3 P*4	ASYSVM
380	ARSPPR	ARSPPU	ARSPRD	ARIOPU
390	ARIOPR	ARIORD	P*5 P*6	ARSPAC
3A0	AVMREAL	ASYSABND	ASYSLC	ASYSOP
3B0	ARIOCT	ARIOCH	ARIOCU	ARIODV
3C0	ARIOCC	ARIOUC	ARIODC	ACORETBL
3D0	APAGCP	CPCREG0	CPCREG8	PSARSV9
3E0	PSARSV10	PSARSV11	ADMKFVR	XVRINST
3F0	PAGECUR	MONNEXT	PAGEND	PAGENXT
400	TRACEFLG	PSARSV12		
	PSARSV15			
430	INSTWRD1	INSTWRD2	INSTWRD3	INSTWRD4
440	Constants Pool			
4D0	APTRLK	NOADD	X4OFFS	XRIGHT24
4E0	XPAGNUM	XRIGHT16	AFREE	AFRET
4F0	AQCNWT	ADSPCH	APTRAN	X2048BND

Hexadecimal Displacement	Field Name			Field Description, Contents, Meaning

Machine Usage

0	IPLPSW	DS	1D	IPL start 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
10	TRACEND	DS	1F	Address of end of trace table
14	TRACCURR	DS	1F	Address of next available trace table entry
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	MCNPSW	DS	1D	Machine check new PSW
78	IONPSW	DS	1D	I/O new PSW
80	CPULOG	DS	16D	CPU and storage logout area

Definitions for the CPULOG

		ORG	CPULOG	
80		DS	1F	Reserved for IBM use
84	INTEXF	DS	1F	External interrupt code (fullword)
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 interrupt code
98	PERADD	DS	1F	PER interrupt address
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	INTKFLIN	DS	1F	I/O interrupt key, flags, interface address
BA	INTTIO	EQU	INTKFLIN+2	I/O interrupt device address (halfword)
BC		DS	11F	Reserved for IBM use
E8	INTMC	DS	1D	Machine check interrupt code
F0		DS	1D	Reserved for IBM use

Hexadecimal Displacement	Field Name			Field Description, Contents, Meaning
F8	FAILSTAD	DS	1F	Failing storage address
FC	REGNCODE	DS	1F	Region code
10	FXDLOG	DS	12D	Fixed logout area
16	FPRLOG	DS	4D	Floating-point register logout area
18	GPRLOG	DS	16F	General register logout area
1C	CRLOG	DS	16F	Control register logout area
20	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	
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	DMKFRE 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	

Hexadecimal Displacement	Field Name			Field Description, Contents, Meaning
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	FREEWORK	DS	12F	DMKFRE work area
2F0	DATE	DS	CL8	Date – mm/dd/yy – edited EBCDIC
2F8	TODATE	DS	1D	TOD clock at 00.00.00 today – local time
300	STARTIME	DS	1D	Date and time started – TOD clock value
308	CPUID	DS	1D	CPU ID
		ORG	CPUID	
308	CPUVERSN	DS	1X	Version code
309	CPUSER	DS	3X	CPU serial number – packed unsigned
30C	CPUMODEL	DS	2X	CPU model number
30E	CPUMCELL	DS	1H	Maximum length in bytes of MCEL
310	IDLEWAIT	DC	X'7FFFFFFFFFFFF000'	Total system idle wait time
318	PAGEWAIT	DC	X'7FFFFFFFFFFFF000'	Total system page wait time
320	IONTWAIT	DC	X'7FFFFFFFFFFFF000'	Total system I/O wait time
328	PROBTIME	DC	X'7FFFFFFFFFFFF000'	Total system problem state time
330	RUNPSW	DS	1D	PSW last loaded by dispatcher
338	RUNUSER	DS	1F	Address of dispatched VMBLOK
33C	DSPLPSW	DS	1F	Load PSW instruction used to dispatch
340	RUNCR0	DS	1F	Control register 0 at dispatch
344	RUNCR1	DS	1F	Control register 1 at dispatch
348	CPSTAT	DS	1F	CP running status
		ORG	CPSTAT	
348	CPSTATUS	DS	1X	CP running status

Bits defined in CPSTATUS

	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
349	XTNDLOCK	DC	X'00'	System extending free storage if = X'FF'
34A	CPSTAT2	DC	1X	Flag byte

Bits defined in CPSTAT2

	CPMICAVL	EQU	X'80'	Virtual machine assist available on CPU
	CPMICON	EQU	X'40'	Virtual machine assist is on for system
	CPSHRLK	EQU	X'20'	CP processing shared named system page

		ORG	TRACEFLAG	
34C	CPRESTRT	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 (TOD units)
360	PGWAITPG	DS	1D	Time spent in page wait, x pages waiting
368	PSASVCCT	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
370	PSENDCLR	DS	0F	End of area cleared by DMKCPINT
370	PID	DS	1F	CP running identifier
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 IPL device

Hexadecimal Displacement	Field Name			Field Description, Contents, Meaning
37A	PGSRATIO	DC	H'0' P*4	Page steals/total replenished
37C	ASYSVM	DC	V(DMKSYSVM)	Address of system VMBLOK
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	PSARSV6	DS	1H P*6	Reserved for IBM use
39C	ARSPAC	DC	V(DMKRSPAC)	Address of system accounting chain
3A0	AVMREAL	DC	A(0)	VMBLOK address of Virtual=Real user
3A4	ASYSABND	DC	A(0)	Address of system ABEND printer
3A8	ASYSLC	DC	V(DMKSYSLC)	Address of SYSLOCS information
3AC	ASYSOP	DC	V(DMKSYSOP)	Address of system operator VMBLOK
3B0	ARIOCT	DC	V(DMKRIOCT)	Address of real channel index table
3B4	ARIOCH	DC	V(DMKRIOCH)	Address of first RCHBLOK
3B8	ARIOCU	DC	V(DMKRIOCU)	Address of first RCUBLOK
3BC	ARIODV	DC	V(DMKRIODV)	Address of first RDEVBLOK
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(X'FFFFFF')	Address of first pageable program
3D4	CPCREG0	DC	X'808008C0'	CP architecture control and external mask
3D8	CPCREG8	DC	F'0'	Monitor call enable mask
3DC	LASTUSER	DS	V(DMKSYSVM)	Last user to be dispatched
3E0	PSARSV10	DS	1F	Reserved for IBM use
3E4	PSARSV11	DS	1F	Reserved for IBM use
3E8	ADMKFVR	DC	F'0'	Reserved for IBM use
3EC	XVRINST	DC	F'0'	Reserved for IBM use
3F0	PAGECUR	DS	1F	Reserved for IBM use
3F4	MONNEXT	DS	1F	Reserved for IBM use
3F8	PAGEND	DS	1F	Reserved for IBM use
3FC	PAGENXT	DS	1F	Reserved for IBM use
400	TRACEFLG	DS	1F	Trace table flags
400	TRACFLG1	DS	1X	Trace table flag

	Bits defined in TRACFLG1			
	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

	Bits defined in TRACFLG2			
	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, HDV tracing on
	TRAC10	EQU	X'08'	Unstack IOBLOK or TRQBLOK tracing on
	TRAC11	EQU	X'04'	Trace BTU activity for 370X NCP
402	TRACFLG3	DS	1H	Reserved for IBM use
404	PSARSV12	DS	1F	Reserved for IBM use

Hexadecimal Displacement	Field Name			Field Description, Contents, Meaning
408	PSARSV15	DS	5D	Reserved for IBM use
430	INSTWRD1	DC	F'0'	Reserved for installation use
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

List of Frequently Used Constants

440	ZEROES	DC	6D'0'	
470	BLANKS	DC	8X'40'	
478	FFS	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'0000000F'
4AC	F16	DC	F'16'	
4B0	F20	DC	F'20'	
4B4	F24	DC	F'24'	
4B8	F60	DC	F'60'	Also = X'0000003C'
4BC	F240	DC	F'240'	Also = X'000000F0' = C'0'
4C0	F255	DC	F'255'	Also = X'000000FF'
4C4	F256	DC	F'256'	Also = X'00000100'
4C8	F4095	DC	F'4095'	Also = X'00000FFF'
4CC	F4096	DC	F'4096'	Also = X'00001000'
4D0	APTRLK	DC	V(DMKPTRLK	Address of DMKPTR
4D4	NOADD	DC	X'FF000000'	Frequently-used work value
4D8	X40FFS	DC	X'40FFFFFF'	Frequently-used work value
4DC	XRIGHT24	DC	X'00FFFFFF'	Isolate right 24 bits
4E0	XPAGNUM	DC	X'00FFF0006'	Isolate the page number
4E4	XRIGHT16	DC	X'0000FFFF'	Isolate the right 16 bits
4E8	AFREE	DC	V(DMKFREE)	Address of DMKFREE
4EC	AFRET	DC	V(DMKFRET)	Address of DMKFRET
4F0	AFRET	DC	V(DMKQCNWT)	Address of DMKQCNWT
4F4	ADSPCH	DC	V(DMKDSPCH)	Address of DMKDSPCH
4F8	APTRAN	DC	V(DMKPTRAN)	Address of DMKPTRAN
4FC	X2048BND	DC	X'00FFF800'	Locate a half-page boundary
500	PSAEND	DS	0D	End of page 0 usage.

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.

RCHBLOK: 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 half-word 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.

RCUBLOK: The Real Control Unit Table is composed of Real Control Unit Blocks (RCUBLCK), 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 displacment into the Real Device Table of its attached devices. This table is referenced in the same manner as the table in the RCHBLOK.

RDEVBLOK: Each device and 3270 remote communications line in the system is represented by a Real Device Control Block (RDEVBLOK), 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 RDEVBLOK and not in the other blocks include a pointer to the I/O request that is currently active on the device, SIO counts, and a pointer to error and sense information. The RDEVBLOK 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 RDEVBLOK is associated with 3270 remote communications line, then the RDEVBLOK contains a pointer to NICBLOKs that represents the resources on that line.

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

• The identity of the requestor

• 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/O control stacks the IOBLOK for the
given task, it attempts to restart all of
the components that have been freed by the
completion of the operation.
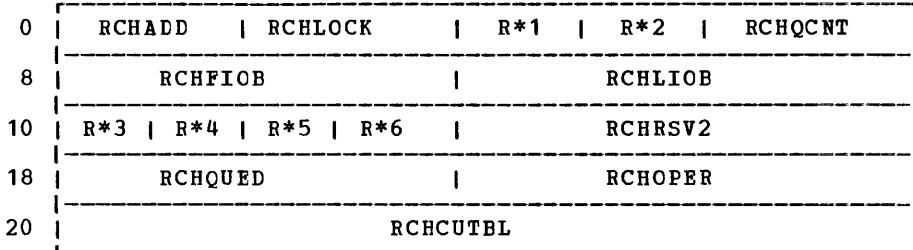
NICBLOK: There is one Network Interface
Control Block for each defined 3704 or 3705
and each resource attached to a 3270 bisync
line. The NICBLOK provides the
correspondence between the line or device
address and the physical resource connected
to that line. This block not only defines
the identity of 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 co-ordinates, polling and
selection characters information as well.

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

RCHBLOK: REAL CHANNEL BLOCK

RCHBLOK 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.

```
     r----------------------------------------------------------------,
  0  | RCHADD    | RCHLOCK     | R*1  | R*2  | RCHQCNT       |
     |----------------------------------------------------------------|
  8  |      RCHFIOB           |        RCHLIOB               |
     |----------------------------------------------------------------|
 10  | R*3 | R*4 | R*5 | R*6  |        RCHRSV2               |
     |----------------------------------------------------------------|
 18  |      RCHQUED           |        RCHOPER               |
     |----------------------------------------------------------------|
 20  |                     RCHCUTBL                          |
     L----------------------------------------------------------------J
```
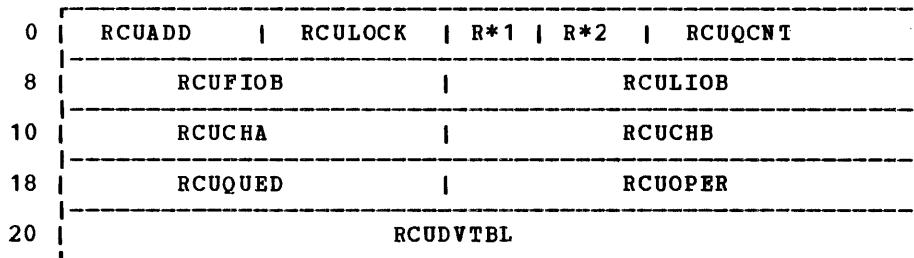
Hexadecimal Displacement	Field Name				Field Description, Contents, Meaning
0	RCHADD	DS	1H		Channel address
2	RCHLOCK	DS	1H		Channel lock
4	RCHSTAT	DS	1X	R*1	channel status
	Bits defined in RCHSTAT				
	RCHBUSY	EQU	X'80'		Channel busy
	RCHSCED	EQU	X'40'		IOB scheduled on channel
	RCHDISA	EQU	X'20'		Channel disabled
	RCHDED	EQU	X'01'		Channel dedicated
5	RCHTYPE	DS	1X	R*2	Channel type
	Bits defined in RCHTYPE				
	RCHSEL	EQU	X'80'		Selector channel
	RCHBMX	EQU	X'40'		Block multiplexer channel
	RCHMPX	EQU	X'20'		Byte multiplexer channel
	RCH370	EQU	X'01'		S/370 type channel (S/370 instruction support)
6	RCHQCNT	DS	1H		Number of IOBLOKs queued off channel
8	RCHFIOB	DS	1F		Pointer to first IOBLOK queued
C	RCHLIOB	DS	1F		Pointer to last IOBLOK 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	RCHRSV2	DS	1F		Reserved for IBM use
18	RCHQUED	DS	1F		IOBLOK queued on channel time
1C	RCHOPER	DS	1F		IOBLOK operational on channel time
20	RCHCUTBL	DS	32H		Control units attached — RCUSTART index
	RCHSIZE	EQU	(*-RCHBLOK)/8		RCHBLOK size in doublewords (X'0C')

RCUBLOK: REAL CONTROL UNIT BLOCK
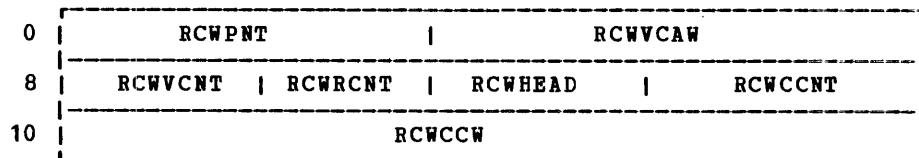
RCUBLOK provides control and status information on a defined real control unit. Linkages are provided to IOBLOKs queued.

```
      r---------------------------------------------------------------------1
  0   |   RCUADD     |   RCULOCK   | R*1 | R*2 |     RCUQCNT              |
      |--------------------------------------------------------------------|
  8   |        RCUFIOB            |              RCULIOB                   |
      |--------------------------------------------------------------------|
 10   |        RCUCHA            |              RCUCHB                    |
      |--------------------------------------------------------------------|
 18   |        RCUQUED           |              RCUOPER                   |
      |--------------------------------------------------------------------|
 20   |                         RCUDVTBL                                   |
      L_____J
```

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
	Bits defined in RCUSTAT				
	RCUBUSY	EQU	X'80'		Control unit busy
	RCUSCED	EQU	X'40'		IOB scheduled on control unit
	RCUDISA	EQU	X'20'		Control unit disabled
	RCUDED	EQU	X'01'		Control unit dedicated
5	RCUTYPE	DS	1X	R*2	Control unit type
	Bits defined in RCUTYPE				
	RCUSHRD	EQU	X'80'		This control unit can attach to only 1 subchannel
	RCUSUB	EQU	X'40'		This is a subordinate control unit
	RCU2701	EQU	X'01'		TCU is a 2701
	RCU2702	EQU	X'02'		TCU is a 2702
	RCU2703	EQU	X'03'		TCU is a 2703
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 – interface A
14	RCUCHB	DS	1F		Pointer to RCHBLOK – interface B
18	RCUQUED	DS	1F		IOBLOK queued on control unit time
1C	RCUOPER	DS	1F		IOBLOK operational on control unit time
20	RCUDVTBL	DS	16H		Devices attached – RDVSTART index
	RCUSIZE	EQU	(*-RCUBLOK)/8		RCUBLOK size in doublewords (X'08')
		ORG	RCUCHA		
10	RCUPRIME	DS	1F		Pointer to the primary control unit

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.

```
        ┌─────────────────────────────────────────────────────────────┐
    0   │         RCWPNT           │           RCWVCAW                  │
        ├─────────────────────────────────────────────────────────────┤
    8   │   RCWVCNT  │  RCWRCNT  │  RCWHEAD   │        RCWCCNT          │
        ├─────────────────────────────────────────────────────────────┤
    10  │                          RCWCCW                              │
        └─────────────────────────────────────────────────────────────┘
```

Hexadecimal Displacement	Field Name			Field Description, Contents, Meaning
0	RCWPNT	DS	1F	Pointer to next RCWTASK
4	RCWVCAW	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 size in doublewords
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
	Bits defined in RCWCTL			
	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-B 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

RDEVBLOK: REAL DEVICE BLOCK

RDEVBLCK is generated by the RDEV macro at system generation time. There is one RDEVBLOK for each real device and one for each binary synchronous line. The block contains status and device parameters applicable to I/O instruction processing.

```
     r--------------------------------------------------------------------------¬
 0   | RDEVADD   | RDEVLOCK     |  R*1  |  R*2  |  R*3  |  R*4  |
     |--------------------------------------------------------------------------|
 8   |    RDEVFIOB            |              RDEVLIOB            |
     |--------------------------------------------------------------------------|
10   |    RDEVCUA            |              RDEVCUB             |
     |--------------------------------------------------------------------------|
18   |                       RDEVQUED                          |
     |--------------------------------------------------------------------------|
20   |    RDEVIOCT          |              RDEVAIOB             |
     |--------------------------------------------------------------------------|
28   |    RDEVUSER          |   RDEVATT   |     RDEVCYL         |
     |--------------------------------------------------------------------------|
30   |              RDEVSER           |      RDEVLNKS           |
     |--------------------------------------------------------------------------|
38   |              RDEVTCTL                                   |
     |--------------------------------------------------------------------------|
40   |    RDEVTMAT          |  R*5  |  R*6  |  R*7  |  R*8  |
     |--------------------------------------------------------------------------|
48   |    RDEVIOER          |              RDEVCTRS            |
     |--------------------------------------------------------------------------|
50   | RDEVNAME | RDEVRSV1   |              RDEVRSV2           |
     L--------------------------------------------------------------------------
```

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

Bits defined in RDEVSTAT

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
RDEVWAII	EQU	X'02'	GRAF — IOBLOK pending, queue requests
RDEVDED	EQU	X'01'	Dedicated device (attached to a user)

Hexadecimal Displacement	Field Name			Field Description, Contents, Meaning
5	RDEVFLAG DS	1X	R*2	Device flags, device dependent
	Bits defined in RDEVFLAG			
	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
	RDEVPSUP EQU	X'80'		Console - terminal has print suppress
	RDEVPREP EQU	X'40'		Console - terminal executing PREPARE command
	RDEVACTV EQU	X'20'		Console - IOBLOK 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
	RDEVEPMD 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 - IOBLOK pending; queue requests
	RDEVEPLN EQU	X'04'		Special - Emulator lines in use by system
	RDEVRCVY EQU	X'02'		Special - Auto dump/load process active
	RDEVTBTU EQU	X'01'		Special - BTU trace requested
6	RDEVTYPC DS	1X	R*3	Device type class (See Appendix A)
7	RDEVTYPE DS	1X	R*4	Device type (See Appendix A)
8	RDEVFIOB DS	1F		Pointer to first IOBLOK queued
C	RDEVLIOB DS	1F		Pointer to last IOBLOK queued
10	RDEVCUA DS	1F		Pointer to RCUBLOK - interface A
14	RDEVCUB DS	1F		Pointer to RCUBLOK - interface B
18	RDEVQUED DS	1D		IOBLOK queued time - TOD clock units
20	RDEVIOCT DS	1F		Device I/O count
24	RDEVAIOB DS	1F		Active IOBLOK
28	RDEVUSER DS	1F		Pointer to VMBLOK 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	RDEVTCTL DS	8X		Terminal control bytes
40	RDEVTMAT DS	1F		Device attached time - TOD clock word 0
44	RDEVQCNT DS	1X	R*5	Number of queued IOBLOKs
45	RDEVSTA2 DS	1X	R*6	Device status (byte 2)
	Bits defined in RDEVSTA2			
	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
46	RDEVMDL DS	1X	R*7	Device model number
47	RDEVFTR DS	1X	R*8	Device feature code
48	RDEVIOER DS	1F		Pointer to IOERBLOK for last CP error
4C	RDEVCTRS DS	1F		Pointer to statistical data counter control block
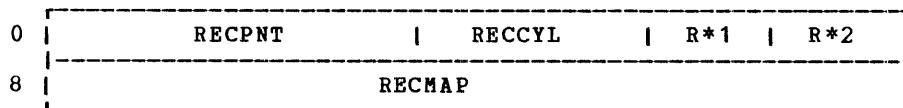
Hexadecimal Displacement	Field Name			Field Description, Contents, Meaning
50	RDEVNAME	DS	1H	Real device name
52	RDEVRSV1	DS	1H	Reserved for IBM use
54	RDEVRSV2	DS	1F	Reserved for IBM use
	RDEVSIZE	EQU	(*-RDEVBLOK)/8	RDEVBLOK size in doublewords (X'0A')

For CP-owned Devices

		ORG	RDEVUSER	
28	RDEVALLN	DS	1F	Anchor for ALOCBLOK chain for this device
2C	RDEVCODE	DS	1H	Device code - SYSOWNED index
		ORG	RDEVTCTL	
38	RDEVPAGE	DS	1F	Anchor for RECBLOK chain for paging
3C	RDEVRECS	DS	1F	Anchor for RECBLOK chain for spooling
40	RDEVPNT	DS	1F	Pointer to next RDEVBLOK for allocation

For Slotted 2301 Paging Devices

		ORG	RDEVRECS	
3C	RDEVDCTL	DS	1F	Pointer to DRUMTABL control block

For Graphic Devices

		ORG	RDEVCYL	
2E	RDEVCORD	DS	1H	Current line coordinates

For Spooling Unit Record Devices

		ORG	RDEVQUED	
18	RDEVSPL	DS	1F	Pointer to active RSPLCTL block
1C	RDEVCLAS	DS	4C	Device class(es)

For Terminal Devices

		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
3A	RDEVTFLG	DS	1X	Additional terminal flags
3B	RDEVRSV3	DS	1X	Reserved for IBM use
3C	RDEVLLEN	DS	1X	Device line length
3D	RDEVATNC	DS	1X	Device attention count
3E	RDEVBASE	DS	1H	370X base address for emulator line
		ORG	RDEVMDL	
46	RDEVTMCD	DS	1X	Terminal code
47	RDEVSADN	DS	1X	Terminal set-address number

Bits defined in RDEVTFLG

	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
	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
3F	RDEVRSV3	DS	1X	Reserved for IBM use

Hexadecimal Displacement	Field Name			Field Description, Contents, Meaning
		ORG	RDEVMDL	
46	RDEVTMCD	DS	1X	Terminal code
	Bits defined in RDEVTMCD			
	RDEVPTTC	EQU	X'00'	PTTC/EBCD keyboard
	RDEVCORR	EQU	X'04'	Correspondence keyboard
	RDEVAPLP	EQU	X'08'	APL PTTC/EBCD keyboard
	RDEVAPLC	EQU	X'0C'	APL Correspondence keyboard
	RDEVUSC8	EQU	X'10'	UASCII-8 level keyboard
47	RDEVSADN	DS	1X	Terminal set-address number

For Real 3704/3705 Communications Controller

		ORG	RDEVAIRA	
1C	RDEVEPDV	DS	1F	Start of free RDEVBLOK 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

For 3270 Remote Support

		ORG	RDEVNCP	
30	RDEVBSC	DS	1F	Pointer to BSCBLOK
34	RDEVDLY	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.

```
    r------------------------------------------------------------------1
0 |              RECPNT          |   RECCYL      | R*1 |  R*2      |
    |------------------------------------------------------------------|
8 |                         RECMAP                                   |
    L------------------------------------------------------------------J
```

Hexadecimal Displacement	Field Name				Field Description, Contents, Meaning
0	RECPNT	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	RECMAX	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
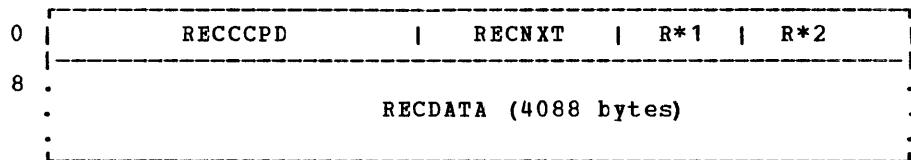
RECSIZE   EQU   (*-RECBLOK)/8   RECBLOK size in doublewords (X'03')

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.

RECPAG: ERROR RECORDING PAGE RECORD

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

```
    r-----------------------------------------------------------------------,
0   |           RECCCPD           |   RECNXT   |  R*1  |  R*2   |
    |-----------------------------------------------------------------------|
8   .                                                                       .
    .                     RECDATA (4088 bytes)                              .
    .                                                                       .
    L-----------------------------------------------------------------------J
```
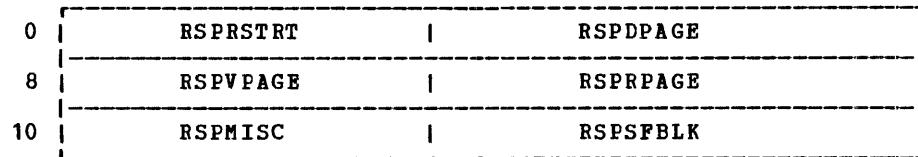
Hexadecimal Displacement	Field Name			Field Description, Contents, Meaning
0	RECCCPD	DS	4X	CCPD of the record
4	RECNXT	DS	2X	Displacement to next error record
6	RECFLAG1	DS	1X R*1	Record usage flags
	Bits defined in RECFLAG1			
	RECPAGIU	EQU	X'80'	Page contains valid data
	RECPAGFR	EQU	X'40'	Page is cleared
	RECPAGFL	EQU	X'20'	Page is full of error records
	RECPAGER	EQU	X'10'	Next page is unreadable (I/O error)
7	RECFLAG2	DS	1X R*2	Record format flag
	Bits defined in RECFLAG2			
	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')

RSPLCTL: REAL SPOOL CONTROL BLOCK

RSPLCTL is used in conjunction with SFBLOK for processing closed spool files.

```
      ┌─────────────────────────────┬─────────────────────────────┐
  0   │         RSPRSTRT            │         RSPDPAGE             │
      ├─────────────────────────────┼─────────────────────────────┤
  8   │         RSPVPAGE            │         RSPRPAGE             │
      ├─────────────────────────────┼─────────────────────────────┤
 10   │         RSPMISC             │         RSPSFBLK             │
      └─────────────────────────────┴─────────────────────────────┘
```

Hexadecimal Displacement	Field Name			Field Description, Contents, Meaning
0	RSPRSTRT	DS	1F	Restart CAW – CCW address
4	RSPDPAGE	DS	1F	DASD 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 SFBLOK for file
	RSPSIZE	EQU	(*-RSPLCTL)/8	Size in doublewords (X'03')

SAVEAREA

SAVEAREA is used to save the registers of a module when that module has called another module.
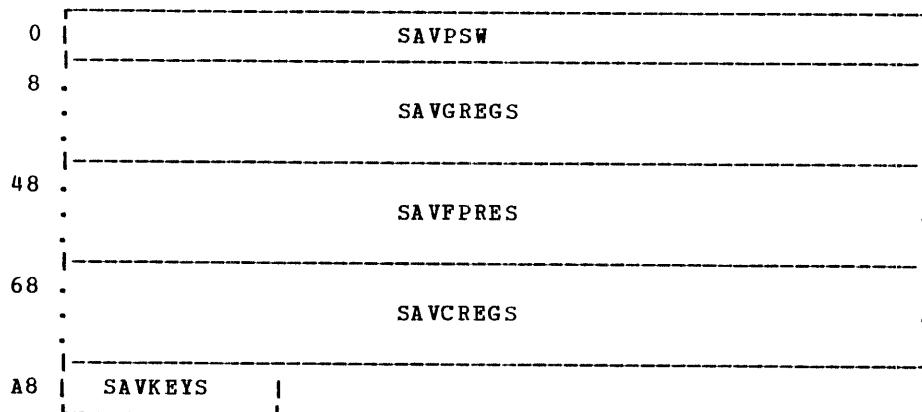
```
     r------------------------------------------------------------------¬
   0 |           SAVERETN         |           SAVER12                    |
     |----------------------------+-------------------------------------|
   8 |           SAVER13          |           SAVEWRK1                   |
     |----------------------------+-------------------------------------|
  10 .                                                                  .
     .                            SAVEREGS                              .
     .                                                                  .
     |----------------------------+-------------------------------------|
  40 |           SAVEWRK2         |           SAVEWRK3                   |
     |----------------------------+-------------------------------------|
  48 |           SAVEWRK4         |           SAVEWRK5                   |
     |----------------------------+-------------------------------------|
  50 |           SAVEWRK6         |           SAVEWRK7                   |
     |----------------------------+-------------------------------------|
  58 |           SAVEWRK8         |           SAVEWRK9                   |
     L------------------------------------------------------------------
```

Hexadecimal Displacement	Field Name			Field Description, Contents, Meaning
0	SAVERETN	DS	1F	Active SAVEAREA (caller's 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
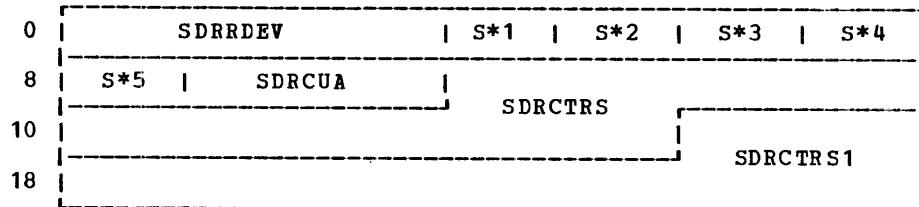
SAVTABLE is used in the initial program loading of saved virtual machine named systems. It is created by the name system generation process (SAVESYS macro/SAVESYS command).

```
         r---------------------------------------------------------1
    0   |                        SAVPSW                             |
        |---------------------------------------------------------|
    8   .                                                          .
        .                        SAVGREGS                          .
        .                                                          .
        |---------------------------------------------------------|
   48   .                                                          .
        .                        SAVFPRES                          .
        .                                                          .
        |---------------------------------------------------------|
   68   .                                                          .
        .                        SAVCREGS                          .
        .                                                          .
        |---------------------------------------------------------J
   A8   |   SAVKEYS      |
        L_____J
```

Hexadecimal Displacement	Field Name			Field Description, Contents, Meaning
0	SAVPSW	DS	1D	PSW of virtual machine at SAVSYS table
8	SAVGREGS	DS	16F	General registers
48	SAVFPRES	DS	4D	Floating-point registers
68	SAVCREGS	DS	16F	Control registers
A8	SAVKEYS	DS	1H	2-byte entry for each saved page containing storage keys for each page

SDRBLOK: STATISTICAL DATA RECORDING BLOCK

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

```
       r----------------------------------------------------------------------
 0  |          SDRRDEV          | S*1  |  S*2  |  S*3  |  S*4  |
    |----------------------------------------------------------------------
 8  |  S*5  |     SDRCUA        |                                         |
    |-------------------------------------|  SDRCTRS   r-----------------|
10  |                                              |                     |
    |----------------------------------------------J  SDRCTRS1           |
18  |                                                                    |
    L----------------------------------------------------------------------
```
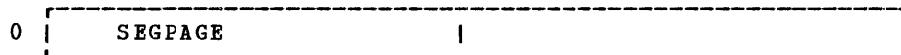
Hexadecimal Displacement	Field Name				Field Description, Contents, Meaning
0	SDRRDEV	DS	1F		Address of associated RDEVBLOK
4	SDRFLAGS	DS	1X	S*1	SDRBLOK flags
	Bits defined in SDRFLAGS				
	SDRSHRT	EQU	X'80'		Short OBR to be written
	SDRFLCT	EQU	X'40'		Full byte counter
		EQU	X'20'		
		EQU	X'10'		
		EQU	X'08'		
		EQU	X'04'		
		EQU	X'02'		
		EQU	X'01'		
5	SDRPRMCT	DS	1X	S*1	Parameter list counter
6	SDRRSV1	DS	1H		Reserved for IBM use
8	SDRLNGTH	DS	1X	S*3	Length, in bytes, of SDR counters
9	SDRCUA	DS	3X		
C	SDRCNTS	DS	5F		SDR error counters

SEGTABLE: SEGMENT TABLE
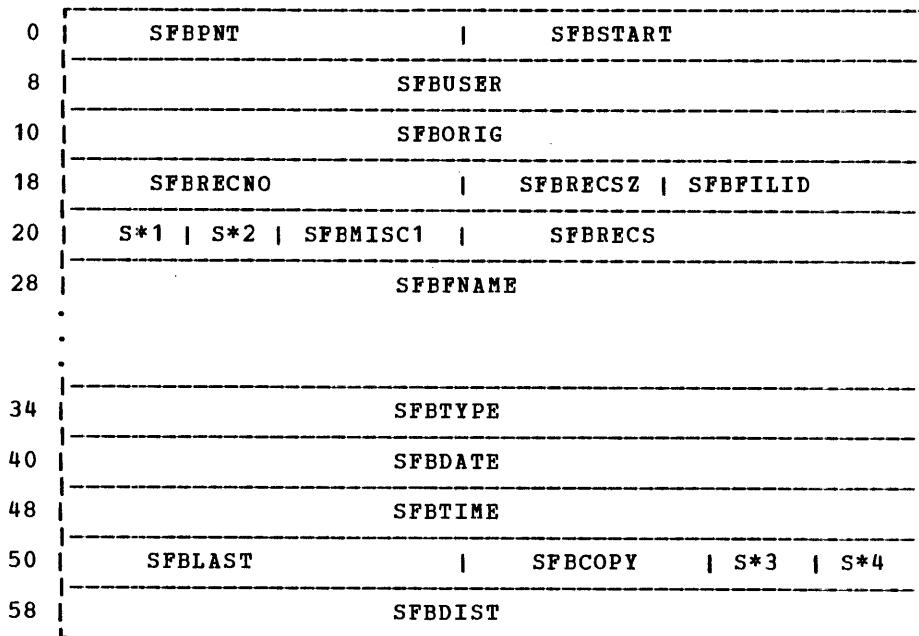
SEGTABLE is used in conjunction with the page table (PAGTABLE) and swap table (SWPTABLE) by the page management routines.

```
     ┌──────────────────────────────────────────────────────────┐
0    │   SEGPAGE                    │                            │
     └──────────────────────────────────────────────────────────┘
```

Hexadecimal Displacement	Field Name			Field Description, Contents, Meaning
0	SEGPAGE	DS	1F	Pointer to page table
	Page Table Length			
		ORG	SEGPAGE	
0	SEGPLEN	DS	1X	Page table length (Total pages − 1) (in left half of byte)

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.

```
      r-----------------------------------------------------------------1
   0  |       SFBPNT              |        SFBSTART                      |
      |---------------------------------------------------------------- |
   8  |                        SFBUSER                                   |
      |-----------------------------------------------------------------|
  10  |                        SFBORIG                                   |
      |-----------------------------------------------------------------|
  18  |       SFBRECNO            |   SFBRECSZ  |  SFBFILID              |
      |-----------------------------------------------------------------|
  20  |  S*1  |  S*2  | SFBMISC1  |        SFBRECS                       |
      |-----------------------------------------------------------------|
  28  |                        SFBFNAME                                  |
      .                                                                  .
      .                                                                  .
      .                                                                  .
      |-----------------------------------------------------------------|
  34  |                        SFBTYPE                                   |
      |-----------------------------------------------------------------|
  40  |                        SFBDATE                                   |
      |-----------------------------------------------------------------|
  48  |                        SFBTIME                                   |
      |-----------------------------------------------------------------|
  50  |     SFBLAST              |     SFBCOPY     |  S*3  |  S*4        |
      |-----------------------------------------------------------------|
  58  |                        SFBDIST                                   |
      L-----------------------------------------------------------------J
```
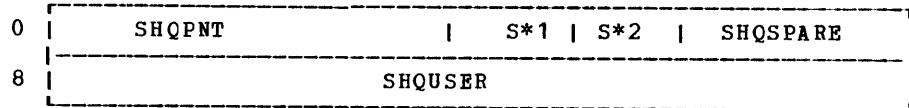
Hexadecimal Displacement	Field Name				Field Description, Contents, Meaning
0	SFBPNT	DS	1F		Pointer to next SFBLOK
4	SFBSTART	DS	1F		DASD location (DCHR) of last page buffer
8	SFBUSER	DS	CL8		VMUSER identification of file owner
10	SFBORIG	DS	CL8		VMUSER identification of file origin
18	SFBRECNO	DS	1F		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	SFBLOK 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 EOF
SFBRECER	EQU	X'01'	SFBREC chain incomplete

21	SFBTYPE	DS	1X	S*2	Device type for output
22	SFBMISC1	DS	1H		Use varies according to caller
24	SFBRECS	DS	1F		Pointer to RECBLOKS 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 (DCHR) of last page buffer
54	SFBCOPY DS	1H		Number of copies requested
56	SFBCLAS DS	1C	S*3	Spool output class
57	SFBFLAG2 DS	1X	S*4	SFBLOK 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 ouput file

Note: SFBHOLD and SFBNOHLD override options in VDEVBLOK.

SFBREQUE	EQU	X'20'	Re-queue 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

58	SFBDIST DS	CL8	Distribution code
	SFBSIZE EQU	(*-SFBLOK)/8	Size in doublewords (X'0C')

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.

```
     r---------------------------------------------------------1
0  |         SHQPNT            |  S*1 | S*2 |   SHQSPARE    |
   |---------------------------------------------------------|
8  |                       SHQUSER                           |
     L_____J
```
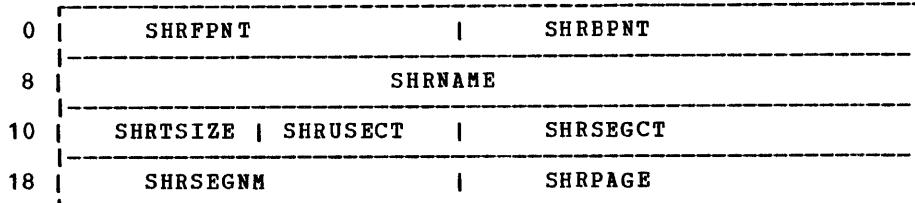
Hexadecimal Displacement	Field Name			Field Description, Contents, Meaning
0	SHQPNT	DS	1F	Address of next SHQBLOK
4	SHQFLAGS	DS	0CL4	Length
4	SHQUHOLD	DS	1X S*1	User USER HOLD flag byte
5	SHQSHOLD	DS	1X S*2	User SYSTEM HOLD flag byte
	Bits defined in SHQUHOLD and SHQSHOLD			
	TYPPRT			Used for printer type
	TYPPUN			Used for punch type (see Appendix A for DEVTYPES for both TYPPRT and TYPPUN)
6	SHQSPARE	DS	2X	Reserved for IBM use
8	SHQUSER	DS	CL8	VMUSER identification of file owner
	SHQBSIZE	EQU	(*-SHQBLOK)/8	Size in doublewords (X'02')

SHRTABLE: NAMED-SHARED SEGMENT SYSTEMS TABLE
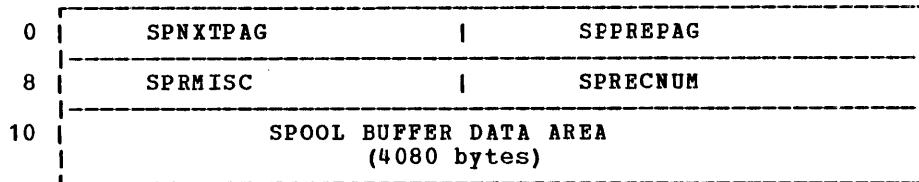
SHRTABLE contains pointers to the segment locations of named systems for both the shared and nonshared user. This block is used in paging, IPL, and VMA operations.

```
      r-----------------------------------------------------------------------¬
   0  |         SHRFPNT               |         SHRBPNT                       |
      |-----------------------------------------------------------------------|
   8  |                           SHRNAME                                     |
      |-----------------------------------------------------------------------|
  10  |    SHRTSIZE | SHRUSECT        |         SHRSEGCT                      |
      |-----------------------------------------------------------------------|
  18  |         SHRSEGNM             |         SHRPAGE                        |
      L-----------------------------------------------------------------------┘
```

Hexadecimal Displacement	Field Name			Field Description, Contents, Meaning
0	SHRFPNT	DS	1F	Pointer to next SHRTABLE
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 to 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 SEGTABLEs. There is one word for each shared segment. The entry is the same as S*1 SEGPAGE in "SEGTABLE: Segment Table."

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.

```
     r--------------------------------------------------------------1
 0 |        SPNXTPAG              |        SPPREPAG               |
   |------------------------------|------------------------------|
 8 |        SPRMISC               |        SPRECNUM              |
   |------------------------------|------------------------------|
10 |                 SPOOL BUFFER DATA AREA                      |
   |                      (4080 bytes)                           |
   L-------------------------------------------------------------J
```

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')

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.
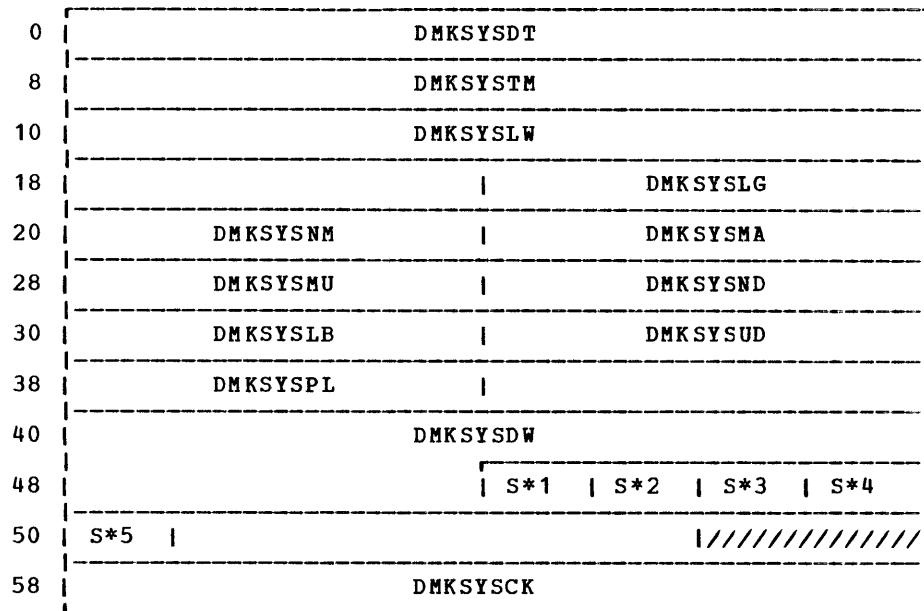
```
    r----------------------------------------------------------------------1
0 |            SWPVM              |            SWPPAG                        |
    |----------------------------------------------------------------------|
8 |   S*1  | S*2  | S*3  | S*4  |    SWPCYL         | S*5  | S*6  |
    L_____J
```

Hexadecimal Displacement	Field Name				Field Description, Contents, Meaning
0	SWPVM	DS	1F		Pointer to VMBLOK
4	SWPPAG	DS	1F		Pointer to PAGTABLE
8	SWPFLAG	DS	1X	S*1	SWPTABLE flag bits
	Bits defined in SWPFLAG				
	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	SWPVPAGE	DS	1X	S*2	Virtual page number
A	SWPKEY1	DS	1X	S*3	Virtual storage key
B	SWPKEY2	DS	1X	S*4	Virtual storage key
C	SWPCYL	DS	1H		DASD cylinder address
E	SWPDPAGE	DS	1X	S*5	Page number on cylinder
F	SWPCODE	DS	1X	S*6	RDEVBLOK 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: SYSTEM LOW STORAGE INFORMATION BLOCK
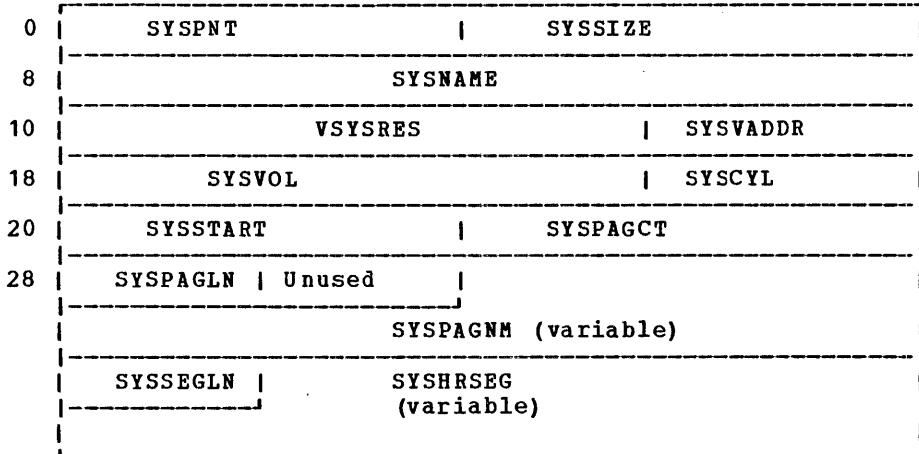
SYSLOCS contains user logon and dial statistics, time/date and log message data, TOD values, and line edit values.

```
     r----------------------------------------------------------------------¬
   0 |                            DMKSYSDT                                   |
     |----------------------------------------------------------------------|
   8 |                            DMKSYSTM                                   |
     |----------------------------------------------------------------------|
  10 |                            DMKSYSLW                                   |
     |----------------------------------------------------------------------|
  18 |                              |              DMKSYSLG                  |
     |----------------------------------------------------------------------|
  20 |        DMKSYSNM              |              DMKSYSMA                  |
     |----------------------------------------------------------------------|
  28 |        DMKSYSMU              |              DMKSYSND                  |
     |----------------------------------------------------------------------|
  30 |        DMKSYSLB              |              DMKSYSUD                  |
     |----------------------------------------------------------------------|
  38 |        DMKSYSPL              |                                       |
     |----------------------------------------------------------------------|
  40 |                            DMKSYSDW                                   |
     |                              r----------------------------------------|
  48 |                              | S*1  | S*2  | S*3  | S*4   |
     |--------------------------------------------------------------|
  50 | S*5   |                                       |////////////////|
     |----------------------------------------------------------------------|
  58 |                            DMKSYSCK                                   |
     L----------------------------------------------------------------------
```

Hexadecimal Displacement	Field Name		Field Description, Contents, Meaning
0	DMKSYSDT DC	CL8'MM/DD/YY'	Date of system log message
8	DMKSYSTM 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	DMKSYSMU 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	# default line-end (pound-sign)
4D	DMKSYSLD DC	X'4A' S*2	¢ default line-delete (cent-sign)
4E	DMKSYSCD DC	X'7C' S*3	@ default character-delete (at-sign)
4F	DMKSYSES DC	X'7F' S*4	" default edit escape (quotation mark)
50	DMKSYSLL DC	AL1(130,129,72) S*5	Default line lengths for 3210 and 3215 – 2741 and 1050 – TTY terminals
53	DC	XL5'0'	Reserved for IBM use
58	DMKSYSCK DC	D'0'	Time-of-day clock value last stored by accounting, DUMP or machine check

SYSTBL: NAMED SYSTEM TABLE
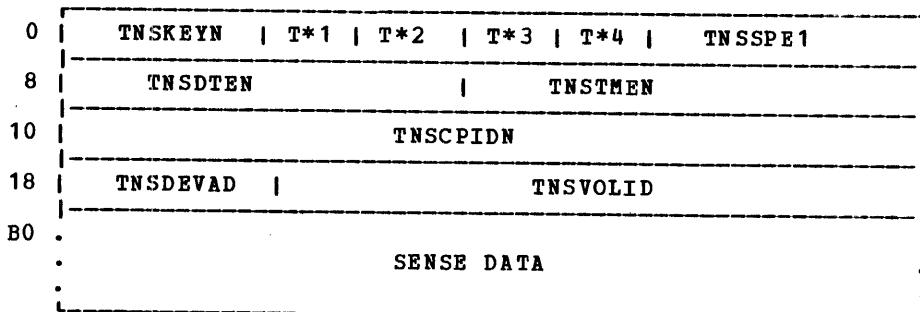
SYSTBL contains the system and DASD information required to load a saved system by name.

```
        r-----------------------------------------------------------------1
     0  | SYSPNT                   |        SYSSIZE                        |
        |-----------------------------------------------------------------|
     8  |                        SYSNAME                                   |
        |-----------------------------------------------------------------|
    10  |             VSYSRES                    |   SYSVADDR              |
        |-----------------------------------------------------------------|
    18  |      SYSVOL                         |    SYSCYL                  |
        |-----------------------------------------------------------------|
    20  |     SYSSTART             |        SYSPAGCT                       |
        |-----------------------------------------------------------------|
    28  | SYSPAGLN  | Unused       |                                      |
        |--------------------------'                                      |
        |                  SYSPAGNM  (variable)                           |
        |-----------------------------------------------------------------|
        | SYSSEGLN   |      SYSHRSEG                                       |
        |------------'         (variable)                                 |
        |                                                                 |
        L_____J
```

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 serial of DASD containing user's system
16	SYSVADDR	DS	1H	Virtual address of VSYSRES
18	SYSVOL	DS	CL6	Volume serial 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
2C	SYSPAGNM	DS	1F	One fullword entry for each range of pages to be saved
30	SYSSEGLN	DS	1H	Numbers of entries in SYSHRSEG
32	SYSHRSEG	DS	1X	One byte for each segment to be shared

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/O device.
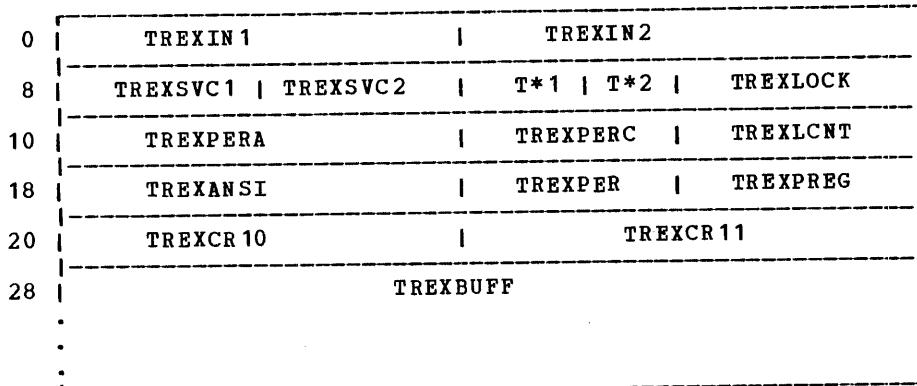
```
      r----------------------------------------------------------------------1
  0  | TNSKEYN  | T*1 | T*2  | T*3 | T*4 |   TNSSPE1                |
     |------------------------------------------------------------------------|
  8  |       TNSDTEN              |       TNSTMEN                   |
     |------------------------------------------------------------------------|
 10  |                   TNSCPIDN                                   |
     |------------------------------------------------------------------------|
 18  |   TNSDEVAD   |              TNSVOLID                         |
     |------------------------------------------------------------------------|
 B0  .                                                             .
     .                   SENSE DATA                                .
     .                                                             .
     L_____J
```

Hexadecimal Displacement	Field Name				Field Description, Contents, Meaning
	24-Byte Header Record				
0	TNSKEYN	DS	1H		Class source 90=T type N/S NON TPER
2	TNSSWS1	DS	1C	T*1	Switch byte
3	TNSSWS2	DS	1C	T*2	Reserved for IBM use
4	TNSSWS3	DS	1C	T*3	Reserved for IBM use
5	TNSRECNT	DS	1C	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		CPU ID and serial
	Device Dependent Data				
18	TNSDEVAD	DS	1H		Device address request is pending
1A	TNSVOLID	DS	6C		Volume serial
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
B0	TNSSNS7	DS	24X		Last 24 sense bytes

TREXT: VIRTUAL MACHINE TRACING EXTENSION TO VMBLOK

TREXT facilitates the tracing of virtual machine program instructions and interrupts. This block is used whenever the CP TRACE command is invoked.

```
        ┌─────────────────────────────────┬─────────────────────────────────┐
    0   │         TREXIN1                 │         TREXIN2                 │
        ├─────────────────┬───────────────┼───────┬───────┬─────────────────┤
    8   │    TREXSVC1     │   TREXSVC2    │  T*1  │  T*2  │    TREXLOCK     │
        ├─────────────────┴───────────────┼───────┴───────┼─────────────────┤
   10   │         TREXPERA                 │    TREXPERC   │    TREXLCNT     │
        ├─────────────────────────────────┼───────────────┼─────────────────┤
   18   │         TREXANSI                 │    TREXPER    │    TREXPREG     │
        ├─────────────────────────────────┴───────────────┴─────────────────┤
   20   │         TREXCR10                 │         TREXCR11               │
        ├───────────────────────────────────────────────────────────────────┤
   28   │                         TREXBUFF                                   │
        │                                                                   .
        .                                                                   .
        .                                                                   .
        .                                                                   .
        └───────────────────────────────────────────────────────────────────┘
```
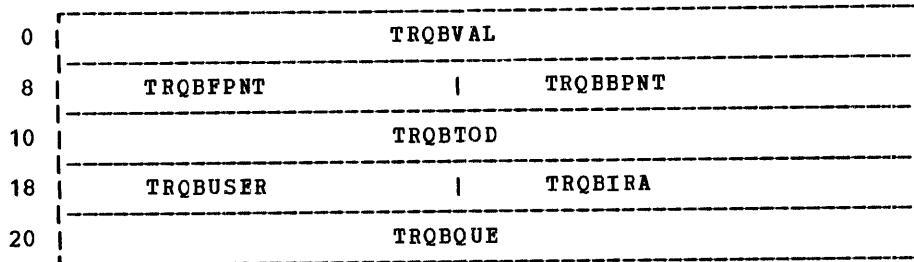
Hexadecimal Displacement	Field Name			Field Description, Contents, Meaning
0	TREXIN1	DS	1F	First address – replaced instruction
4	TREXIN2	DS	1F	Second address – replaced instruction
8	TREXSVC1	DS	1H	Displaced halfword – instruction 1
A	TREXSVC2	DS	1H	Displaced halfword – instruction 2
		ORG	TREXIN1	
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
	Bits defined in TREXFLAG			
	TREXRUN	EQU	X'80'	Prevent CPWAIT between events
	TREXVAT	EQU	X'40'	Call DMKVATRN to put back virtual instruction
D	TREXOUT	DS	1X T*2	Trace output controls
	Bits defined in TREXOUT			
	TREXPRT	EQU	X'80'	Output to the virtual printer
	TREXCON	EQU	X'40'	Output to user terminal
E	TREXLOCK	DS	1H	Indicates tracing when set
10	TREXPERA	DS	1F	PER event address on interrupt
14	TREXPERC	DS	1H	PER code bits from hardware event
16	TREXLCNT	DS	1H	Printed output line count
18	TREXANSI	DS	1A	Address of next (or last) sequential instruction
1C	TREXCR9	DS	0F	Shadow control registers for PER trace
1C	TREXPER	DS	XL2	PER control field
1E	TREXPREG	DS	1H	PER register mask field
20	TREXCR10	DS	1F	Address range starting value

Hexadecimal Displacement	Field Name			Field Description, Contents, Meaning
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')
		ORG	TREXPERA	Redefinition for TRACE use
10	TREXNSI	DS	6X	Actual next (or last) sequential instruction
		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
	Bits defined in TREXCTL2			
	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
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/norun flagbits corresponding to TREXCTL
24	TREXPNTR	DS	1F	Pointer to first stacked TRACE request, if any

TRQBLOK: TIMER REQUEST BLOCK

TRQBLOK manages the timing facilities of VM/370.

```
      r------------------------------------------------------------------,
   0  |                          TRQBVAL                                 |
      |--------------------------------------------------------------------|
   8  |        TRQBFPNT              |         TRQBBPNT                    |
      |--------------------------------------------------------------------|
  10  |                          TRQBTOD                                 |
      |--------------------------------------------------------------------|
  18  |        TRQBUSER             |          TRQBIRA                    |
      |--------------------------------------------------------------------|
  20  |                          TRQBQUE                                 |
      L--------------------------------------------------------------------
```
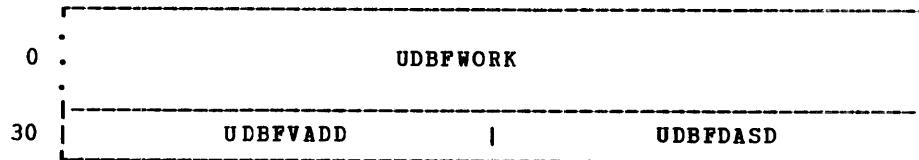
Hexadecimal Displacement	Field Name			Field Description, Contents, Meaning
0	TRQBVAL	DS	1D	TOD clock comparator value for interrupt
8	TRQBFPNT	DS	1F	Pointer to next TRQBLOK
C	TRQBBPNT	DS	1F	Pointer to previous TRQBLOK
10	TRQBTOD	DS	1D	TOD clock value when TRQBLOK is queued
18	TRQBUSER	DS	1F	Address of VMBLOK for user
1C	TRQBIRA	DS	1F	Interrupt return address
20	TRQBQUE	DS	1D	Time left in queue; tracking virtual CPU timer

TRQBSIZE EQU (*-TRQBLOK)/8 Size in doublewords

UDBFBLOK: USER DIRECTORY BUFFER BLOCK

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

```
        r---------------------------------------------------------1
        .                                                         .
  0     .                        UDBFWORK                         .
        .                                                         .
        |---------------------------------------------------------|
  30    |          UDBFVADD            |          UDBFDASD         |
        L_____J
```

Hexadecimal Displacement	Field Name			Field Description, Contents, Meaning
0	UDBFWORK	DS	6D	Buffer work space used by the caller
30	UDBFVADD	DS	1F	Virtual address of the last directory page
34	UDBFDASD	DS	1F	DASD address of the last directory page
	UDBFSIZE	EQU	(*-UDBFBLOK)/8	UDBFBLOK size in doublewords (X'07')

UDEVBLCK: USER DEVICE BLOCK

UDEVBLOK 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.
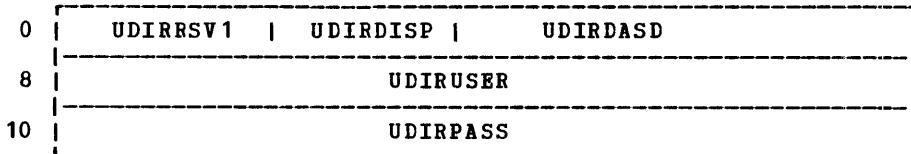
```
    ┌──────────────────────────────────────────────────────────────┐
 0  │  UDEVADD   | UDEVDISP  |        UDEVDASD                      │
    ├──────────────────────────────────────────────────────────────┤
 8  │  U*1 | U*2 | U*3 | U*4 | U*5 | U*6 |      UDEVNCYL            │
    ├──────────────────────────────────────────────────────────────┤
10  │  UDEVRELN  |                 UDEVVSER                         │
    ├──────────────────────────────────────────────────────────────┤
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 UDEVBLOK
4	UDEVDASD	DS	1F		DASD address of the next UDEVBLOK
8	UDEVSTAT	DS	1X	U*1	Status information
	Bits defined in UDEVSTAT				
	UDEVDED	EQU	X'80'		Device to be dedicated to this user
	UDEVTDSK	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 time)
	UDEVSPOO	EQU	X'08'		Device is a spool device
	UDEV3158	EQU	X'04'		Device is a 3158 console
9	UDEVMODE	DS	1X	U*2	Access mode information
	Bits defined in UDEVMODE				
	UDEVLR	EQU	X'80'		Read links allowed
	UDEVLW	EQU	X'40'		Write links allowed
	UDEVLM	EQU	X'20'		Multiple write links allowed
	UDEVR	EQU	0		Device to be in R link mode for owner
	UDEVRR	EQU	4		Device to be in RR link mode for owner
	UDEVW	EQU	8		Device to be in W link mode for owner
	UDEVWR	EQU	12		Device to be in WR link mode for owner
	UDEVM	EQU	16		Device to be in M link mode for owner
	UDEVMR	EQU	20		Device to be in MR link mode for owner
	UDEVMW	EQU	24		Device to be in MW link mode for owner
A	UDEVTYPC	DS	1C	U*3	Device class
B	UDEVTYPE	DS	1C	U*4	Device type
C	UDEVFTR	DS	1C	U*5	Device feature mode
D	UDEVMDL	DS	1C	U*6	Device model number
E	UDEVNCYL	DS	1H		Virtual DASD size
10	UDEVRELN	DS	1H		Virtual DASD cylinder relocation
12	UDEVVSER	DS	6C		Volume serial number
18	UDEVPASR	DS	1D		Password for read access
20	UDEVPASW	DS	1D		Password for write access

Hexadecimal Displacement	Field Name			Field Description, Contents, Meaning
28	UDEVPASM DS	1D		Password for multiple access
	UDEVSIZE EQU	(*—UDEVBLOK)/8		UDEVBLOK size in doublewords
	ORG	UDEVMDL		User device block (short)
D	UDEVCLAS DS	1C	C*6	Unit spool output class
E	UDEVLINK DS	1H		User link to disk
10	UDEVLKID DS	1D		User link to userid

UDIRBLOK: USER DIRECTORY BLOCK

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

```
        r---------------------------------------------------------------------1
    0   |   UDIRRSV1   |   UDIRDISP  |      UDIRDASD                          |
        |---------------------------------------------------------------------|
    8   |                           UDIRUSER                                   |
        |---------------------------------------------------------------------|
   10   |                           UDIRPASS                                   |
        L_____J
```

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	UDIRBLOK size in doublewords (X'03')

UMACBLOK: USER MACHINE BLOCK

UMACBLOK 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 UDEVBLOK, the command privilege class, assigned line edit values, as well as other virtual machine options.

```
      r-------------------------------------------------------------------1
  0   |    UMACDVCT    |   UMACDISP   |         UMACDASD                  |
      |----------------------------------------------------------------------|
  8   |  U*1  |  U*2  |  U*3  |  U*4  |   U*5   |  U*6  |  U*7  |  U*8      |
      |----------------------------------------------------------------------|
 10   |        UMACCORE           |          UMACMCOR                     |
      |----------------------------------------------------------------------|
 18   |                        UMACACCT                                  |
      |----------------------------------------------------------------------|
 20   |                        UMACDIST                                  |
      |----------------------------------------------------------------------|
 28   |                        UMACIPL                                   |
      L-------------------------------------------------------------------J
```

Hexadecimal Displacement	Field Name				Field Description, Contents, Meaning
0	UMACDVCT	DS	1H		Number of devices
2	UMACDISP	DS	1H		Displacement of the user's first UDEVBLOK
4	UMACDASD	DS	1F		DASD address of the user's first UDEVBLOK
8	UMACCLEV	DS	1C	U*1	Command level
	Bits defined in UMACCLEV				
	UMACCLA	EQU	X'80'		Class A functions
	UMACCLB	EQU	X'40'		Class B functions
	UMACCLC	EQU	X'20'		Class C functions
	UMACCLD	EQU	X'10'		Class D functions
	UMACCLE	EQU	X'08'		Class E functions
	UMACCLF	EQU	X'04'		Class F functions
	UMACCLG	EQU	X'02'		Class G functions
	UMACCLH	EQU	X'01'		Class H functions
9	UMACPRIR	DS	1X	U*2	Priority
A	UMACOPT	DS	1X	U*3	Virtual machine options
	Bits defined in UMACOPT				
	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

UMACBLCK

```
Hexadecimal   Field
Displacement  Name                                    Field Description, Contents, Meaning
_____  _____                                   _____
        B     UMACRSV1 DS    1C      U*4      Reserved for IBM use
        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 logon time

              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 Block (VMBLOK). The VMBLOK 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 (VCHBLOKs). Each VCHBLOK contains a table of pointers that point to the Virtual Control Unit Blocks (VCUBLOKs) for the control units attached to that virtual channel. Each VCUBLOK contains pointers to the Virtual Device Blocks (VDEVBLOKs) attached to the control unit.

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

VCHBLOK: There is one VCHBLOK for each virtual channel connected to the user's virtual CPU. Each VCHBLOK 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, as follows:

1. A status byte (VCHSTAT) indicates whether the channel is busy or has a channel class interrupt pending.

2. A halfword unit address identifies the unit causing the channel—class interrupt (if it is present).

3. 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 VCUBLOKs; index entries representing addresses at which no control unit is attached have a value of −1.
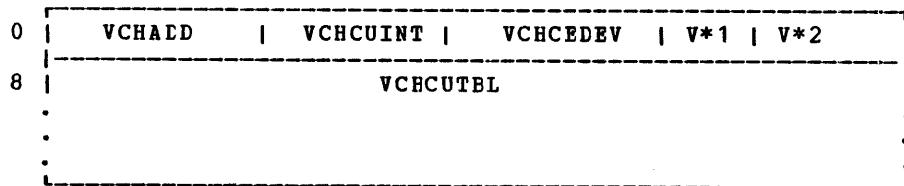
VCUBLOK: There is one VCUBLOK for each control unit in the virtual configuration. These blocks are arranged in a table, and each contains, in addition to its base address, status flags similar to those in the VCHBLOK and a table of indexes to attached VDEVBLOKs. The status flags defined for the VCUBLOK differ from those for the VCHBLOK in that they can contain status for the control unit and also for a subchannel.

   For example, if the VCUBLOK representing a 2803 Tape Control Unit is attached to a virtual selector channel, both the VCHBLOK and the VCUBLOK are marked busy. However, if the VCUBLOK 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 VCUBLOK only. Thus the virtual multiplexer appears nonbusy to operations on other, nonshared subchannels.

VDEVBLOK: There is one VDEVBLOK in the configuration for each virtual device defined by the user. Each VDEVBLOK 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 VDEVBLOK contains device type specific information that allows the I/O translation and simulation routines to interpret the channel programs presented by the user.

VCHBLOK: VIRTUAL CHANNEL BLOCK
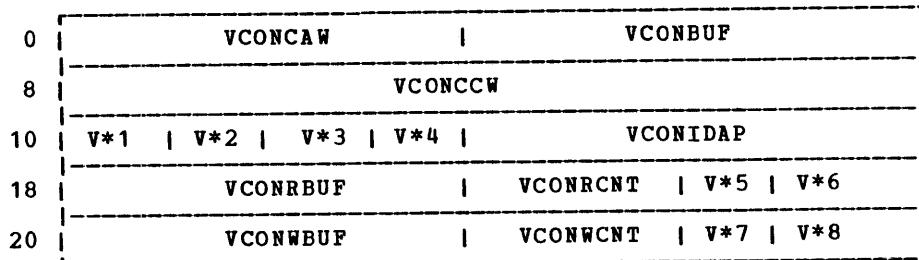
VCHBLOK contains information providing linkage between the virtual machine and one of its virtual channels. Supplied in this block, in addition to channel status and type information, are the reflected interrupts from attached virtual control units.

```
        r-------------------------------------------------------------1
   0    |    VCHADD    |  VCHCUINT |    VCHCEDEV  | V*1 | V*2 |        |
        |-------------------------------------------------------------|
   8    |                        VCHCUTBL                             |
        .                                                             .
        .                                                             .
        .                                                             .
        .                                                             .
        L_____J
```

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
	Bits defined in VCHSTAT				
	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
	Bits defined in VCHTYPE				
	VCHSEL	EQU	X'80'		Virtual selector channel
	VCHBMX	EQU	X'40'		Virtual block multiplexer
8	VCHCUTBL	DS	16H		Control units attached - VMCUSTRT index
	VCHSIZE	EQU	(*-VCHBLOK)/8		VCHBLOK size in doublewords (X'05')

VCONCTL: VIRTUAL CONSOLE CONTROL BLOCK
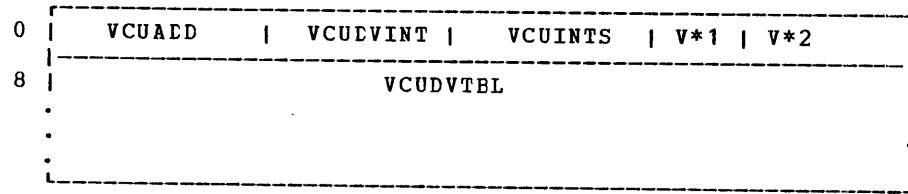
VCONCTL contains CCW and data buffer information for the communications of the virtual console.

```
      r----------------------------------------------------------------1
   0  | VCONCAW                    |              VCONBUF              |
      |------------------------------------------------------------------|
   8  |                         VCONCCW                                  |
      |------------------------------------------------------------------|
  10  | V*1  | V*2  |  V*3  | V*4  |            VCONIDAP                 |
      |------------------------------------------------------------------|
  18  |        VCONRBUF            |   VCONRCNT  | V*5 |  V*6            |
      |------------------------------------------------------------------|
  20  |        VCONWBUF            |   VCONWCNT  | V*7 |  V*8            |
      L----------------------------------------------------------------J
```

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
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 IDA 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	VCONWBSZ	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')
		ORG	VCONCCW		
8	VCONADDR	DS	1F		CCW data address
C	VCONFLAG	DS	1X		CCW flag bits
D	VCONRSV4	DS	1X		Reserved for IBM use
E	VCONCNT	DS	1H		CCW byte count
		ORG	VCONADDR		
8	VCONCOMD	DS	1X		CCW command code

VCUBLOK: VIRTUAL CONTROL UNIT BLOCK

VCUBLOK contains status information relating to the virtual channel, the status and features of the virtual control unit.

```
     r--------------------------------------------------------------------
0  |    VCUADD     |   VCUDVINT  |    VCUINTS   | V*1 | V*2    |
     |-------------------------------------------------------------------|
8  |                          VCUDVTBL                                  |
     .                                                                    .
     .                                                                    .
     .                                                                    .
     L_____J
```

Hexadecimal Displacement	Field Name				Field Description, Contents, Meaning
0	VCUADD	DS	1H		Virtual control unit address
2	VCUDVINT	DS	1H		VDEVBLOK with interrupt-bit map
4	VCUINTS	DS	1H		Virtual control unit interrupt status
6	VCUSTAT	DS	1X	V*1	Virtual control unit status
	Bits defined in VCUSTAT				
	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
	Bits defined in VCUTYPE				
	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		VCUBLCK size in doublewords (X'05')

VDEVBLOK: VIRTUAL DEVICE BLOCK

VDEVBLOK maintains status and interrupt conditions applicable to one virtual device.

```
     ┌──────────────────────────────────────────────────────────────────┐
  0  │   VDEVADD   │  VDEVINTS  │ V*1 │ V*2 │ V*3 │ V*4 │
     │──────────────────────────────────────────────────────────────────│
  8  │                        VDEVCSW                                     │
     │──────────────────────────────────────────────────────────────────│
 10  │   VDEVRELN  │  VDEVBND   │        VDEVPOSN                         │
     │──────────────────────────────────────────────────────────────────│
 18  │      VDEVQUED          │        VDEVOPER                           │
     │──────────────────────────────────────────────────────────────────│
 20  │      VDEVLINK          │        VDEVREAL                           │
     │──────────────────────────────────────────────────────────────────│
 28  │      VDEVIOCT          │        VDEVUSER                           │
     │──────────────────────────────────────────────────────────────────│
 30  │      VDEVIOER          │        VDEVIOB                            │
     └──────────────────────────────────────────────────────────────────┘
```
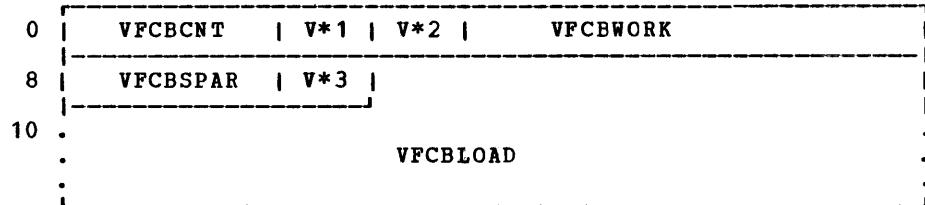
Hexadecimal Displacement	Field Name			Field Description, Contents, Meaning
0	VDEVADD	DS	1H	Virtual device address
2	VDEVINTS	DS	1H	Virtual device interrupt status
4	VDEVTYPC	DS	1X	V*1	Virtual device type class
5	VDEVTYPE	DS	1X	V*2	Virtual device type
6	VDEVSTAT	DS	1X	V*3	Virtual device status

Bits defined in VDEVSTAT

			Field Description, Contents, Meaning
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
VDEVCUE	EQU	X'08'	Virtual control unit end
VDEVNRDY	EQU	X'04'	Virtual device not ready
VDEVCATT	EQU	X'02'	Virtual device attached by console function
VDEVDED	EQU	X'01'	VDEVREAL is dedicated device RDEVBLOK

Hexadecimal Displacement	Field Name			Field Description, Contents, Meaning
7	VDEVFLAG	DS	1X	V*4	Virtual device flags

Bits defined in VDEVFLAG

			Field Description, Contents, Meaning
VDEVRDO	EQU	X'80'	DASD - read-only
VDEVENAB	EQU	X'80'	Virtual 270X - line enabled
VDEVTDSK	EQU	X'40'	DASD - T-disk space allocated by CP
VDEVDIAL	EQU	X'40'	Virtual 270x - line connected
VDEVCSPL	EQU	X'40'	Console - activity spooled
VDEV231T	EQU	X'20'	DASD - 2311 simulated on top half of 2314
VDEV231B	EQU	X'10'	DASD - 2311 simulated on bottom half of 2314
VDEVCCW1	EQU	X'10'	Console and spooling - processing first CCW
VDEVSAS	EQU	X'08'	DASD - Executing standalone seek
VDEVPOST	EQU	X'04'	Present attention with a single interrupt

Hexadecimal Displacement	Field Name			Field Description, Contents, Meaning
	VDEVRSRL	EQU	X'02'	Reserve/release are valid CCW operation codes
	VDEVUC	EQU	X'01'	Virtual device sense bytes present
8	VDEVCSW	DS	1D	Virtual channel status word
10	VDEVRELN	DS	1H	Virtual DASD cylinder relocation
12	VDEVBND	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
24	VDEVREAL	DS	1F	Pointer to real device RDEVBLOK
28	VDEVIOCT	DS	1F	Virtual device I/O count
2C	VDEVUSER	DS	1F	Pointer to VMBLOK of VDEVBLOK owner
30	VDEVIOER	DS	1F	Pointer to IOERBLOK for last error
34	VDEVIOB	DS	1F	Pointer to active IOBLOK
	VDEVSIZE	EQU	(*-VDEVBLOK)/8	VDEVBLOK size in doublewords (X'07')

For Spooling/Console Devices

		ORG	VDEVRELN	
10	VDEVEXTN	DS	1F	Pointer to VSPXBLOK
14	VDEVSPAR	DS	1F	Reserved for IBM use
18	VDEVCON	DS	1F	Pointer to VCONCTL console control
1C	VDEVSPL	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	VDEVCOPY	DS	1H	Number of copies requested
26	VDEVCFLG	DS	1X	Console – virtual console flags

Bits defined in VDEVCFLG

	VDEVATTN	EQU	X'80'	User pressed Attention key more than once
	VDEVTIC	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'	Auto carriage return on first read
27	VDEVSFLG	DS	1X	Spool – virtual spool flags

Bits defined in VDEVSFLG

	VDEVFEED	EQU	X'80'	Spool reader – last command was a feed
	VDEVXFER	EQU	X'80'	Spool output – transfered to VSPXXUSR
	VDEVCONT	EQU	X'40'	Spool input – continuous reading
	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
	VDEVSVC	EQU	X'01'	Spool device busy by CP
		ORG	VDEVIOER	
30	VDEVSNSE	DS	1F	Sense bytes for spool device
34	VDEVFCBK	DS	1F	Address of forms control block (VFCBBLOK)
		ORG	VDEVLINK	
20	VDEVTMAT	DS	1F	T-disk attached time (TOD clock word 0)

VFCBBLOK: VIRTUAL FORM CONTROL BUFFER BLOCK

VFCBBLOK is exclusively used for printer carriage control forms activity.

```
        r----------------------------------------------------------------1
     0  |    VFCBCNT    |  V*1 |  V*2 |      VFCBWORK              |
        |-----------------------------------------------------------------|
     8  |    VFCBSPAR   |  V*3 |                                   |
        |----------------------J                                   |
    10  .                                                          .
        .                      VFCBLOAD                            .
        .                                                          .
        L----------------------------------------------------------J
```

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
	Bits defined in VCFCBFLAG			
	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')

VMBLOK: VIRTUAL MACHINE CONTROL BLOCK

VMBLOK 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_ CPU registers, preferred virtual machine option values, other values significant to virtual machine operations.

Off							
0	VMQFPNT			VMQBPNT			
8	VMPNT			VMECEXT			
10	VMSEG			VMSIZE			
18	VMCHSTRT			VMCUSTRT			
20	VMDVSTRT			VMTERM			
28	VMVTERM	VMTRMID	V*1	V*2	V*3	V*4	
30	VMCHCNT	VMCUCNT	VMDVCNT	VMIOACTV			
38	VMCHTBL						
58	V*5	V*6	V*7	V*8	V*9	V*10	V*11	V*12
60	V*13	V*14	V*15	V*16	VMSLOCK	VMLLOCK	
68	V*19	V*20	VMIOINT		VMTIMER		
70	VMVTIME						
78	VMTMOUTQ						
80	VMTTIME						
88	VMTMINQ						
90	VMTODINQ						
98	VMINST		VMACTDEV	V*17	V*18		
A0	VMTREXT		VMADSTOP				
A8	VMPSW						
B0	VMGPRS						
F0	VMFPRS						

Off			
110	VMUSER		
118	VMACNT		
120	VMDIST		
128	VMPGREAD		VMPGWRIT
130	VMWCNT	VMSEGDSP	VMSTOR
138	VMIOCNT		VMPNCH
140	VMLINS		VMCRDS
148	VMCOMND		
150	VMPDRUM	VMPDISK	VMPAGES	VMPRGIL
158	VMDEDCH	VMQPRIOR	VMWSPROJ	VMSTEALS
160	VMTIMEON		VMTRQBLK
168	VMACOUNT		VMRDINQ
170	VMPRGINQ		VMEPRIOR
178	VMSTKO		VMMICRO
180	VMPFUNC		VMPXINT
188	VMDELAY		VMRPRIOR
190	VMPGNT	VMNDCNT	VMSHRSYS
198	V*21	VMRSVW4	VMASSIST
1A0	VMRSVW1		VMRSVW2
1A8	VMRSVW3		VMRSVW5
1B0	VMUSER1		VMUSER2
1B8	VMUSER1		VMUSER4

Hexadecimal Displacement	Field Name				Field Description, Contents, Meaning
0	VMQFPNT	DS	1F		Pointer to next VMBLOK in queue
4	VMQBPNT	DS	1F		Pointer to previous VMBLOK in queue
8	VMPNT	DS	1F		Pointer (CYCLIC) to next VMBLOK
C	VMECEXT	DS	1F		VMBLOK extended control pointer — ECBLOK
C	VMVCR0	EQU	VMECEXT		Virtual control register 0 for non—EC mode machine
10	VMSEG	DS	1F		Pointer to VMSEGTBL
14	VMSIZE	DS	1F		Virtual storage size in bytes
18	VMCHSTRT	DS	1F		Pointer to VCHBLOK table
1C	VMCUSTRT	DS	1F		Pointer to VCUBLOK table
20	VMDVSTRT	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	VMTLDEL	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	VMDVCNT	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 runuser
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/SIO busy loop
VMRUN	EQU	X'08'	Virtual machine runnable
VMINQ	EQU	X'04'	Virtual machine in a queue
VMELIG	EQU	X'02'	Virtual machine is in an eligible list
VMPAZAPL	EQU	X'01'	Reflect an external interrupt to a virtual machine

Hexadecimal Displacement	Field Name				Field Description, Contents, Meaning
5A	VMOSTAT	DS	1X	V*7	Virtual machine operating status
	Bits defined in VMOSTAT				
	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 CF mode
	VMVIRCF	EQU	X'04'		Virtual machine executing virtual CF
	VMCF	EQU	X'02'		Virtual machine executing CF
	VMKILL	EQU	X'01'		Virtual machine is to be logged off
5B	VMQSTAT	DS	1X	V*8	Virtual machine queueing status
	Bits defined in VMQSTAT				
	VMPRIDSP	EQU	X'80'		Eligible for queue 1
	VMAUTLOG	EQU	X'40'		Autologged user is in disconnect mode
	VMWSERNG	EQU	X'20'		Last working set error was negative; keep in queue 1 for terminal output
	VMWSCHG	EQU	X'08'		Force a new working set on queue entry
	VMCFREAD	EQU	X'02'		Virtual machine with console function read up
	VMPA2APL	EQU	X'01'		Reflects external interrupts to machine
5C	VMPSTAT	DS	1X	V*9	Virtual machine processing status
	Bits defined in VMPSTAT				
	VMISAM	EQU	X'80'		Virtual machine has ISAM CCW checking
	VMV370R	EQU	X'40'		Virtual machine can use extended control
	VMRPAGE	EQU	X'20'		Virtual machine can reserve pages
	VMREAL	EQU	X'10'		Virtual machine has V=R option
	VMNOTRAN	EQU	X'08'		No CCW translation for V=R user
	VMPNMCS	EQU	X'04'		Reserved for IBM use
	VMACCOUN	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
	Bits defined in VMESTAT				
	VMSHADT	EQU	X'80'		Shadow tables are present
	VMPERCM	EQU	X'40'		Virtual CP PER active
	VMBADCRO	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
	Bits defined in VMTRCTL				
	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
	VMTREX	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+VMTREX		Trace all user interrupts

Hexadecimal Displacement	Field Name			Field Description, Contents, Meaning
5F	VMMLEVEL DS	1X	V*12	Message level

Bits defined in VMMLEVEL

VMMSGON	EQU	X'80'	Receiving messages
VMWNGON	EQU	X'40'	Receiving warnings
VMMCODE	EQU	X'20'	Receiving error message codes
VMMTEXT	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

Bits defined in VMQLEVEL

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
VMDROP1	EQU	X'02'	Virtual machine just dropped from Q1

61	VMCLEVEL DS	1X	V*14	Command level

Bits defined in VMCLEVEL

VMCLASSA	EQU	X'80'	Class A functions
VMCLASSB	EQU	X'40'	Class B functions
VMCLASSC	EQU	X'20'	Class C functions
VMCLASSD	EQU	X'10'	Class D functions
VMCLASSE	EQU	X'08'	Calss E functions
VMCLASSF	EQU	X'04'	Class F functions
VMCLASSG	EQU	X'02'	Class G functions
VMCLASSH	EQU	X'01'	Class H functions

62	VMTLEVEL DS	1X	V*15	Timer level

Bits defined in VMTLEVEL

VMTON	EQU	X'80'	Virtual timer running
VMRON	EQU	X'40'	Virtual real timer running
VMCPUTMR	EQU	X'20'	Virtual CPU timer in real CPU timer
VMSTMPI	EQU	X'08'	Virtual interval timer request queued
VMSTMPT	EQU	X'04'	Virtual CPU timer request queued
VMTMRINT	EQU	X'01'	Interrupt from CPU real timer pending

Bits redefined for System VMBLOK

VMTIDLE	EQU	VMTON	CPU timer contains idle wait state time
VMTPAGE	EQU	VMRON	CPU timer contains system page wait time
VMTIONT	EQU	VMSTMPI	CPU timer contains I/O wait state time

63	VMPEND	DS	1X	V*16	Interrupt pending summary flag

Bits defined in VMPEND

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
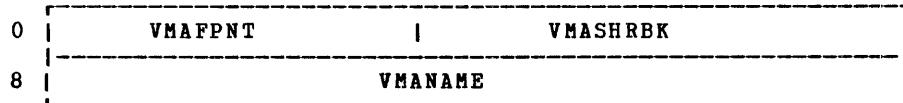
Hexadecimal Displacement	Field Name				Field Description, Contents, Meaning
64	VMSLOCK	DS	1H		Short lock – reserved for IBM use
66	VMLLOCK	DS	1H		Long lock – reserved for IBM use
68	VMFSTAT	DS	1X	V*19	Virtual machine feature status
	Bits defined in VMFSTAT				
	VMFBMX	EQU	X'80'		Virtual block multiplexer channels
	VMMLVL2	DS	1X	V*20	Additional message handling information
	Bits defined in VMMLVL2				
	VMMIMSG	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 CPU time used
78	VMTMOUTQ	DS	1D		Time remaining in queue
80	VMTTIME	DS	1D		Total time while in supervisor state
88	VMTMINQ	DS	1D		VMTTIME value at entry to queue
90	VMTODINQ	DS	1D		TOD clock time stamp at queue entry
98	VMINST	DS	1F		Virtual machine privileged or tracing instruction
9C	VMACTDEV	DS	1H		Virtual device issuing last virtual SIO
9E	VMUPRIOR	DS	1H	V*17	User priority from directory
9F	VMPSWDCT	DS	1X	V*18	Invalid LINK password count
A0	VMTREXT	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 registers
F0	VMFPRS	DS	4D		Virtual machine floating pcint 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	VMPGWRIT	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 non-spooled 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 pages currently resident
156	VMPRGIL	DS	1H		ILC for pending program interrupt
158	VMDEDCH	DS	1H		Dedicated channel mask
15A	VMQPRIOR	DS	1H		Priority in dispatching queue
15C	VMWSPROJ	DS	1H		Projected working set size
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 VMPAGES 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 register 6

Hexadecimal Displacement	Field Name			Field Description, Contents, Meaning
		ORG	VMMICRO	
17C	VMMCR6	DS	1X	Control register 6 – hardware flag byte

Bits defined in VMMCR6

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, 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)

(Note: The last two bits in this byte must always be zeros.)

Hexadecimal Displacement	Field Name			Field Description, Contents, Meaning
17D	VMMADDR	DS	3X	Control register 6 – address of virtual machine's pointer list (MICBLOK)
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	VMRPRIOR	DS	1F	Run list dispatching priority
190	VMPGPNT	DS	1F	Pointer to list of PGBLOKs
194	VMNDCNT	DS	1F	Non—deferred page read count
196	VMSHRSYS	DS	1H	Number of shared named systems
198	VMRBSC	DS	1X V*21	Remote display line count
199	VMRSVW4	DS	3X	Reserved for IBM use
19C	VMASSIST	DS	1F	Pointer to list of VMABLOKs
1A0	VMRSVW1	DS	1F	Reserved for IBM use
1A4	VMRSVW2	DS	1F	Reserved for IBM use
1A8	VMRSVW3	DS	1F	Reserved for IBM use
1AC	VMRSVW5	DS	1F	Reserved for IBM use
1B0	VMUSER1	DS	1F	Reserved for installation use
1B4	VMUSER2	DS	1F	Reserved for installation use
1B8	VMUSER3	DS	1F	Reserved for installation use
1BC	VMUSER4	DS	1F	Reserved for installation use
	VMBSIZE	EQU	(*—VMBLOK)/8	VMBLOK size in doublewords (X'32')

VMABLOK: SHARED SYSTEMS RUNNING WITH VMABLOK

VMABLOK contains the name of a shared system, a pointer to the share table (SHRTABLE), and a pointer to the next VMABLOK.

```
   ┌──────────────────────────────┬──────────────────────────────┐
0  │           VMAFPNT            │           VMASHRBK           │
   ├──────────────────────────────┴──────────────────────────────┤
8  │                            VMANAME                           │
   └──────────────────────────────────────────────────────────────┘
```

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

VSPLCTL: VIRTUAL SPOOL CONTROL BLOCK

VSPLCTL is linked to the VDEVBLOK and contains information for opened spool files.
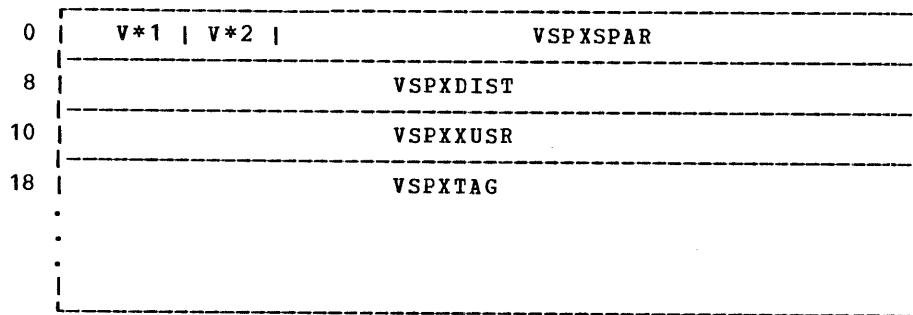
```
    ┌─────────────────────────────────┬─────────────────────────────┐
 0  │        VSPCAW                   │        VSPDPAGE              │
    ├─────────────────────────────────┼─────────────────────────────┤
 8  │        VSPVPAGE                 │        VSPRECNO              │
    ├──────────────────┬──────────────┼─────────────────────────────┤
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 IDA CCW
14	VSPSFBLK	DS	1F		Pointer to SFBLOK 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	IDA 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

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

```
      r----------------------------------------------------------------1
   0 |  V*1  | V*2  |              VSPXSPAR                           |
      |--------------------------------------------------------------|
   8 |                        VSPXDIST                                |
      |--------------------------------------------------------------|
  10 |                        VSPXXUSR                                |
      |--------------------------------------------------------------|
  18 |                        VSPXTAG                                 |
      .                                                               .
      .                                                               .
      .                                                               .
      |                                                               |
      L--------------------------------------------------------------J
```

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 (doublewords)
2	VSPXSPAR	DS	XL6		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	VSPXTAG	DS	CL136		Tag information data area
	VSPXSIZE	EQU	(*-VSPXBLOK)/8		VSPXBLOK size in doublewords (X'14')

XINTBLOK: EXTERNAL INTERRUPT BLOCK

XINTBLOK 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.

```
   r------------------------------------------------------------1
0  |      XINTNEXT            |    XINTSORT | XINTCPUA          |
   |------------------------------------------------------------|
8  |   XINTCODE | XINTMASK    |       XINTPARM                  |
   L------------------------------------------------------------J
```

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	XINTBLOK size in doublewords (X'02')

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

SYSREF

600	A(FVS)	A(OPSECT)
608	A(DEVTAB)	V(FSTLKP)
610	V(DMSINM)	V(FSTLKW)
618	A(PIE)	A(IADT)
620	A(USERSECT)	V(DMSDIOR)
628	V(DMSSCNN)	A(0)
630	A(TABEND)	A(SUBSECT)
638	V(DMSSBDFR)	V(DMSDIOW)
640	V(DMSSMNST)	A(ADTSECT)
648	V(FREE)	V(FRET)
650	V(DMSPIOCC)	A(PGMSECT)
658	A(IOSECT)	V(DMSDBD)
660	A(DIOSECT)	V(OSTABLE)
668	A(DMSERL)	A(DMSCRD)
670	V(DMSFREB)	A(SVCSECT)
678	A(ADTLKP)	V(DMSAUDUR)
680	A(0)	V(OSRET)
688	V(CMSRET)	V(DMSSCNO)
690	V(DMSEXC)	V(DMSLDRA)
698	V(ADTLKW)	V(USABRV)
6A0	A(EXTSECT)	A(SCBPTR)
6A8	A(0)	A(0)
6B0	V(DMSLAF)	V(DMSLAFNX)
6B8	V(DMSLAFFE)	V(DMSLAFFT)
6C0	V(ADTNXT)	V(DMSTRK)
6C8	V(DMSTRKX)	V(DMSTQQ)
6D0	V(DMSTQQX)	V(DMSERS)
6D8	V(TYPSRCH)	V(DMSAUD)
6E0	V(KILLEX)	V(DMSFNST)
6E8	V(DMSBRD)	V(DMSBWR)
6F0	V(DMSFNS)	V(DMSSTTE)
6F8	V(DMSSTTW)	V(POINT)

DMSNUC

USERSECT
SUBSECT
TSOBLKS
OPSECT — FCBIO
DMSABW
DMSFRT
DMSERT
DBGSECT (Debug work area)
CVTSECT(Some fields are filled in at IPL.)
FVS
DIOSECT
SVCSECT
PGMSECT
IOSECT
EXTSECT
AFTSECT (Create when the file is opened. There is room for 5 AFTs in DMSNUC, others are in free storage.)
ADTSECT (Space is allocated when DMSNUC is assembled, fields are filled in when ACCESS command is issued. There is one ADT entry for each of the 10 possible disks.)
DEVTAB
Terminal Buffers and Saveareas
SYSREF
MACDIRC and TXTDIRC
NUCON

Free Storage

CMSCB
OBDCBPT   IOBECBPT
DCB   DECB
AFT continued
CMSAVE   LDRST
MFD

Figure 2. CMS Control Block Relationships

ABTAB: ABEND TERMINATION OPTION TABLE

ABTAB 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 (hex 54) in the SYSCOM block points to the ABTAB block.
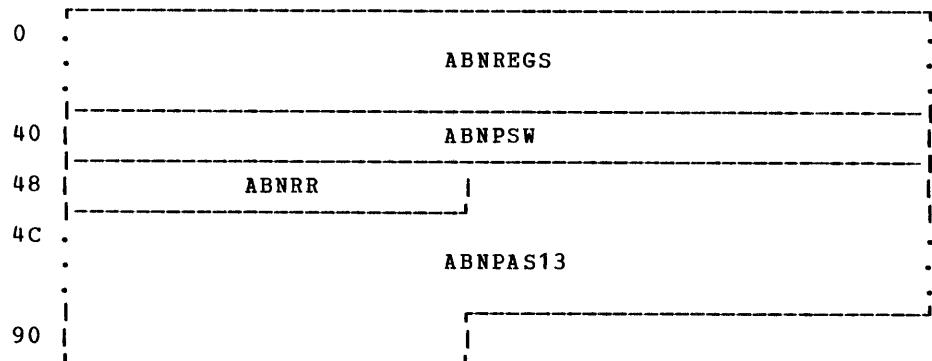
```
     r----------------------------------------------------1
0 |         ABROUT              |         ABSAVE          |
     L----------------------------------------------------J
```

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

ABWSECT: ABEND RECOVERY WORKSPACE

ABWSECT describes the fields used for saving registers and other data during ABEND recovery.

V-constants in DMSABN, DMSDBG, DMSFRE, DMSITI, DMSITP, and DMSITS point to the ABWSECT block. ABWSECT is defined in module DMSABW.

```
       r-----------------------------------------------------------------1
  0  .  .                                                               .
     .  .                          ABNREGS                              .
     .  .                                                               .
     .  |---------------------------------------------------------------|
 40  |  |                          ABNPSW                               |
     .  |---------------------------------------------------------------|
 48  |  |          ABNRR               |                                |
     .  |-----------------------------_|                                |
 4C  .  .                                                               .
     .  .                          ABNPAS13                             .
     .  .                                             r-----------------J
     .  |                                             |
 90  |  |                                             |
        L---------------------------------------------J
```

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

Space for DMSERR PLIST

94		ORG	ABNPAS13	
4C	ABNERLST	DS	47X	

ADTSECT: ACTIVE DISK TABLE

ADTSECT describes the attributes of virtual disks (A-Y) 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, field ADTHBCT at displacement 1C (hexadecimal) into ADTSECT is also defined as OSADTVTA for use by OS simulation routines. ADTSECT is invoked by the ADT macro.

Offset						
0	ADTID			A*1	A*2	
8	ADTPTR		ADTDTA			
10	ADTFDA		ADTMFDN			
18	ADTMFDA		ADTHBCT			
20	ADTFSTC		ADTCHBA			
28	ADTCFST		ADT1ST			
30	ADTNUM		ADTUSED			
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			
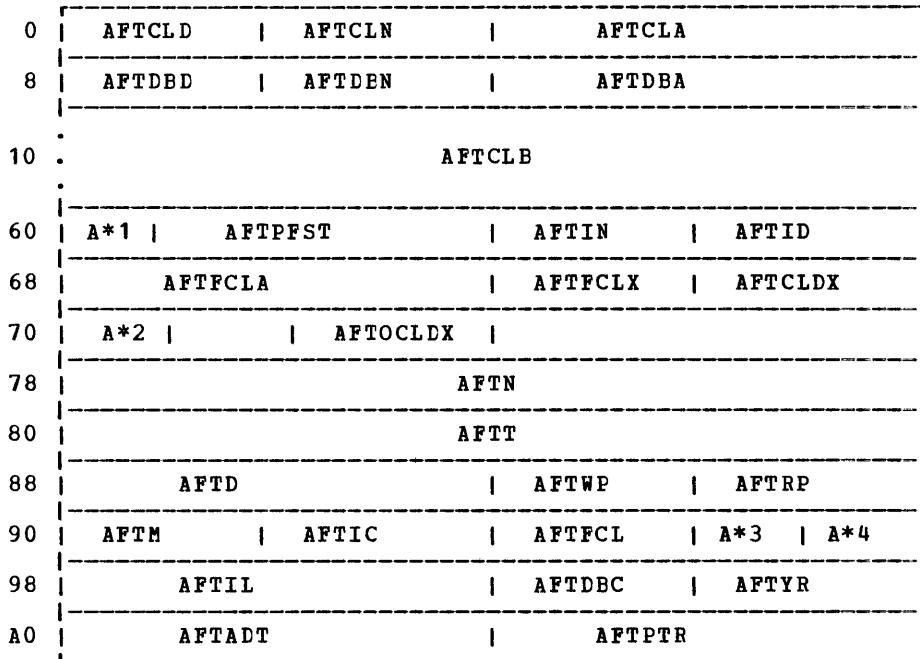
Hexadecimal Displacement	Field Name			Field Description, Contents, Meaning

Needed for Read-Only Disks and Read/Write Disks

0	ADTID	DS	CL6	Disk identifier (label)
6	ADTFLG3	DS	1X	A*1 Third flag byte

Bits defined in ADTFLG3

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 OS or DOS disk
ADTFSORT	EQU	X'10'	All FST hyperblocks and FST entries sorted
ADTFORCE	EQU	X'08'	

7	ADTFTYP	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 NUCON
10	ADTFDA	DS	1A	File directory (SSTAT) address
14	ADTMFDN	DS	1F	Number doublewords in MFD
18	ADTMFDA	DS	1A	Master file directory address
1C	OSADTVTA	DS	0F	VTOC address of OS pack
1C	ADTHBCT	DS	1F	FST hyperblock count
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 (QTLEFTP)
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, etc.)
45	ADTMX	DS	1C	A*4 Extension-of-mode letter (A,B,C, etc.)
46	ADTFLG1	DS	1X	A*5 First flag byte

Bits defined in ADTFLG1

ADTFSF	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 length
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	AFTFLG2	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 1-5 are in storage
ADTFALMD	EQU	ADTFMDRO+X'08'	All modes (0-5) are in storage
ADTFALUF	EQU	ADTFMFD+ADTFALNM+ADTFALTY+ADTFALMD	
			All UFD is in storage
ADTSSTM	EQU	X'02'	ADT SSTAT 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

Hexadecimal Displacement	Field Name			Field Description, Contents, Meaning

Needed for Read/Write Disks

Hexadecimal Displacement	Field Name			Field Description, Contents, Meaning
48	ADTMSK	DS	1A	800-byte (QMSK) bit-mask address
4C	ADTQQM	DS	1A	200-byte (QQMSK) bit-mask address
50	ADTPQM1	DS	1F	PQMSIZ = number of bytes in QMSK > 215
54	ADTPQM2	DS	1F	PQMNUM = number of 800-byte records for QMSK
58	ADTPQM3	DS	1F	RONUM = number of doublewords in QMSK
5C	ADTLHBA	DS	1A	Pointer to last FST hyperblock
60	ADTLFST	DS	1F	Displacment of FST in last hyperblock
64	ADTNACW	DS	1H	Number of active write files - halfword
66	ADTRES	DS	1H	Reserve count (RESRVCNT) - halfword
68	ADTXNREC	DS	1F	Number doublewords of extra chain link records
6C	ADTXAREC	DS	1F	Address of block of extra chain link records
	ADTLBM	EQU	ADT2ND-ADTSECT	Length of minimum ADT block (bytes)
	ADTLDM	EQU	ADTLBM/8	Length of minimum ADT block in doublewords
	ADTLB	EQU	*-ADTSECT	Length of full ADT block (bytes)
	ADTLD	EQU	(ADTLB+7)/8	Length of full ADT block in doublewords

Other Parameters

	ADTRL	EQU	800	Logical record length
	ADTMXBML	EQU	10	Maximum bit map length (number of records) for 3330

NUCON Device Table Offsets

	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.

```
      r--------------------------------------------------------------------¬
   0  | AFTCLD     |   AFTCLN      |          AFTCLA                        |
      |--------------------------------------------------------------------|
   8  | AFTDBD     |   AFTDBN      |          AFTDBA                        |
      |--------------------------------------------------------------------|
      .                                                                    .
  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                |   AFTDBC     |   AFTYR                 |
      |--------------------------------------------------------------------|
  A0  |       AFTADT               |       AFTPTR                          |
      L--------------------------------------------------------------------┘
```
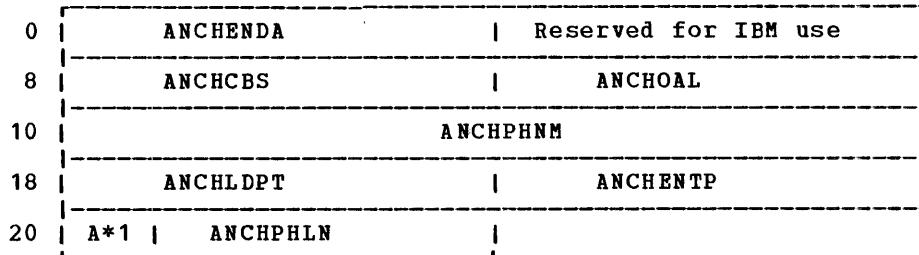
Hexadecimal Displacement	Field Name				Field Description, Contents, Meaning
0	AFTCLD	DS	H		Disk address of current chain link
2	AFTCLN	DS	H		Number of current chain link
4	AFTCLA	DS	F		Address of chain link buffer
8	AFTDBD	DS	H		Disk address of current data block
A	AFTDBN	DS	H		Number of current data block
C	AFTDBA	DS	F		Address of current data block
10	AFTCLB	DS	XL80		Chain link buffer from 1st chain link
60	AFTFLG	DS	X	A*1	Flag byte

	Bits defined in AFTFLG				
	AFTUSED	EQU	X'80'		Active file table block in use
	***	EQU	X'40'		
	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

Hexadecimal Displacement	Field Name				Field Description, Contents, Meaning
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	Second flag byte

Bits defined in AFTFLG2

	AFTNEW	EQU	X'80'		New file
	AFTOLDCL	EQU	X'40'		Current chain link existed previously
	AFTCLX	EQU	X'20'		Alternate chain link assigned/implied

71		DS	1X		Reserved for IBM use
72	AFTOCLDX	DS	1H		Old value (if any) of AFTCLDX

Copy of FST Block Imbedded in AFT Block

78	AFTFST	DS	0D		
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

Bit defined in AFTPTR

	AFTFSF	EQU	X'40'		Indicates in free storage

A8		DS	0D		End of DSECT
	AFTLB	EQU	*-AFTSECT		Length of AFT block in bytes
	AFTLD	EQU	AFTLB/8		Length of AFT block in doublewords
	AFTSECT				
0	AFTSTART	DS	0D		
	ENTRY	AFTSTART			

Active File Table

0		DC	4F'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) —		Addr of next AFT block (in free storage)
348		DC	2F'0' —		Reserved for IBM use

ANCHSECT: ANCHOR TABLE

ANCHSECT defines the DOS/VS anchor table. This DSECT is used by DMSDOS 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 ANCHTAB macro.

```
     r---------------------------------------------------------------1
  0  |        ANCHENDA              | Reserved for IBM use           |
     |------------------------------|--------------------------------|
  8  |        ANCHCBS               |        ANCHOAL                  |
     |------------------------------|--------------------------------|
 10  |                          ANCHPHNM                             |
     |---------------------------------------------------------------|
 18  |        ANCHLDPT              |        ANCHENTP                 |
     |------------------------------|--------------------------------|
 20  | A*1 |    ANCHPHLN            |
     L-------------------------------J
```

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 ACB) table

Followed by one or more Anchor Table Entries of the following format:

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

Bits defined in ANCHSTSW

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	ANCHPHLN	DC	AL3(0)	Length of phase in bytes

BATLSECT: CMS BATCH USER JOB LIMITS

BATLSECT describes the fields in the user job limits table for CMS batch jobs.
The ABATLIMT field (hex 45C) in NUCON points to the job limits table.
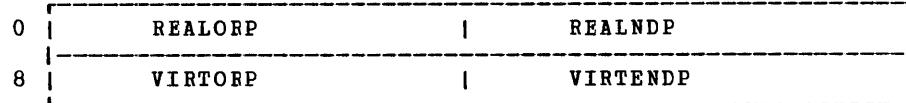
```
    ┌─────────────────────────────────────────────────────────────────────┐
  0 │  BATCPUL   │  BATCPUC    │  BATPRTL    │  BATPRTC            │
    ├─────────────────────────────────────────────────────────────────────┤
  8 │  BATPUNL   │  BATPUNC    │                                   │
    └─────────────────────────────────────────────────────────────────────┘
```

Hexadecimal Displacement	Field Name			Field Description, Contents, Meaning
0	BATCPUL	DC	H'32767'	Virtual CPU limit (second); can be reset
2	BATCPUC	DC	H'0'	Current CPU count; do not 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

BBOX: BOUNDARY BOX

BBOX contains the begin and end addresses of the virtual and real partitions, respectively.

The IJBBOX field (hex DC) in the SYSCOM block 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

Equates for First Entry in BBOX

ALTSDAR	EQU	REALORP	Address of alternate area
MINREALP	EQU	REALNDP	No. page frames in min. 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 (hex 4E0) in NUCON points to the BGCOM block.

```
        .---------------------------------------------------------------.
     0  |                         JOBDATE                                |
        |---------------------------------------------------------------|
     8  |  PPBEG         |  EOSSP          |                             |
        |---------------------------------------------------------------|
    10  |                                                     | UPSI    |
        |---------------------------------------------------------------|
    18  |                         COMNAME                               |
        |---------------------------------------------------------------|
    20  |       PPEND              |          HIPHAS                     |
        |---------------------------------------------------------------|
    28  |       HIPROG             |   LABLEN    |   PIK                 |
        |---------------------------------------------------------------|
    30  |       EOCADR             | A*1  | A*2 | A*3  | A*4             |
        |---------------------------------------------------------------|
    38  | A*5  | A*6 | A*7  | A*8  |    DALC     |   FOCLPT              |
        |---------------------------------------------------------------|
    40  |  PUBPT       |  FAVPT    |   JIBPT     |   TEBPT              |
        |---------------------------------------------------------------|
    48  |  FICLPT      |  NICLPT   |   LUBPT     |  A*9  |              |
        |---------------------------------------------------------------|
    50  |    MMDD          |              YYDDD                         |
        |---------------------------------------------------------------|
    58  | LIOCSCOM  |  PIBPT      |   CHKPTID   |   JOBZON             |
        |---------------------------------------------------------------|
    60  | DIBPT        | Reserved   |   PCPTR     |   ITPTR            |
        |---------------------------------------------------------------|
    68  | OCPT         |  PWTIMS    |  Reserved  |  LTK               |
        |---------------------------------------------------------------|
    70  |    SYSPAR         |          JAPART                           |
        |---------------------------------------------------------------|
    78  |    TODCOM         |  PIB2PTR    |  PDTABB                     |
        |---------------------------------------------------------------|
    80  |    IJLQTTAD       |  BGCOMPT    | A*10 | A*11               |
        |---------------------------------------------------------------|
    88  |    COMEX          | A*12 | A*13| A*14 |                      |
        |---------------------------------------------------------------|
    90  |                   PROCNAM               | A*15             |
        |---------------------------------------------------------------|
    98  |                   POVNAM                | A*16             |
        '---------------------------------------------------------------'
```
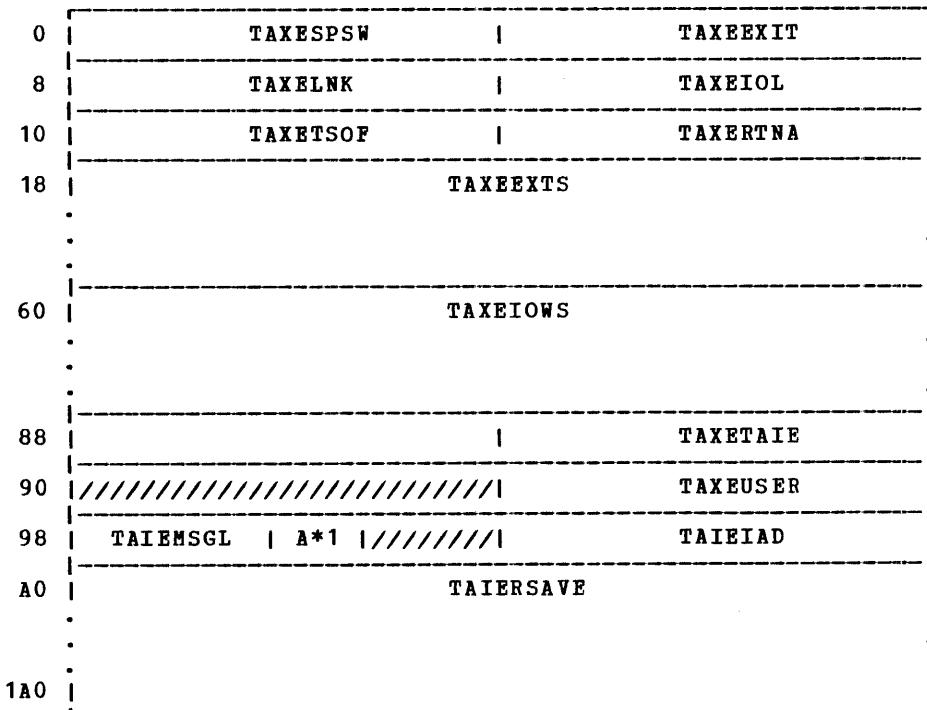
Hexadecimal Displacement	Field Name			Field Description, Contents, Meaning
0	JOBDATE	DC	C'00/00/00'	Job date
8	PPBEG	DC	S(0)	Supervisor end
A	EOSSP	DC	S(0)	End of storage protection
C		DC	11X'00'	User scratch area
17	UPSI	DC	X'00'	UPSI byte
18	COMNAME	DC	CL8'CMS/DOS'	Job name
20	PPEND	DC	A(0)	Highest 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(55)	A*9	SYSLST line count
4F	SYSDATE	DS	OCL9		System date
4F	MMDD	DC	XL4'00'		MMDD or DDMM
53	YYDDD	DC	XL5'00'		YYDDD 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 cat. 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 TAXE macro.

The TAXEADDR field (hex 5D4) in NUCON points to CMSTAXE.

```
      r------------------------------------------------------------------¬
  0   |           TAXESPSW            |           TAXEEXIT               |
      |------------------------------------------------------------------|
  8   |           TAXELNK             |           TAXEIOL                |
      |------------------------------------------------------------------|
 10   |           TAXETSOF            |           TAXERTNA               |
      |------------------------------------------------------------------|
 18   |                          TAXEEXTS                                |
      .                                                                  .
      .                                                                  .
      .                                                                  .
      |------------------------------------------------------------------|
 60   |                          TAXEIOWS                                |
      .                                                                  .
      .                                                                  .
      .                                                                  .
      |------------------------------------------------------------------|
 88   |                               |           TAXETAIE               |
      |------------------------------------------------------------------|
 90   |///////////////////////////////|           TAXEUSER              |
      |------------------------------------------------------------------|
 98   | TAIEMSGL | A*1 |////////|                TAIEIAD                 |
      |------------------------------------------------------------------|
 A0   |                          TAIERSAVE                               |
      .                                                                  .
      .                                                                  .
1A0   |                                                                  |
      L------------------------------------------------------------------J
```

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 I/O old PSW
10	TAXETSOF	DS	1F	TSOFLAGS saved here
14	TAXERTNA	DS	A	Return address
18	TAXEEXTS	DS	18F	ATTN exit return save area
60	TAXEIOWS	DS	11F	DMSIOW save area

Also Attention Exit Parameter List

8C	TAXETAIE	DS	A	Address of TAIE
90		DS	1F	Reserved
94	TAXEDEF	DS	0X	Defer indicator
94	TAXEUSER	DS	A	User PLIST address
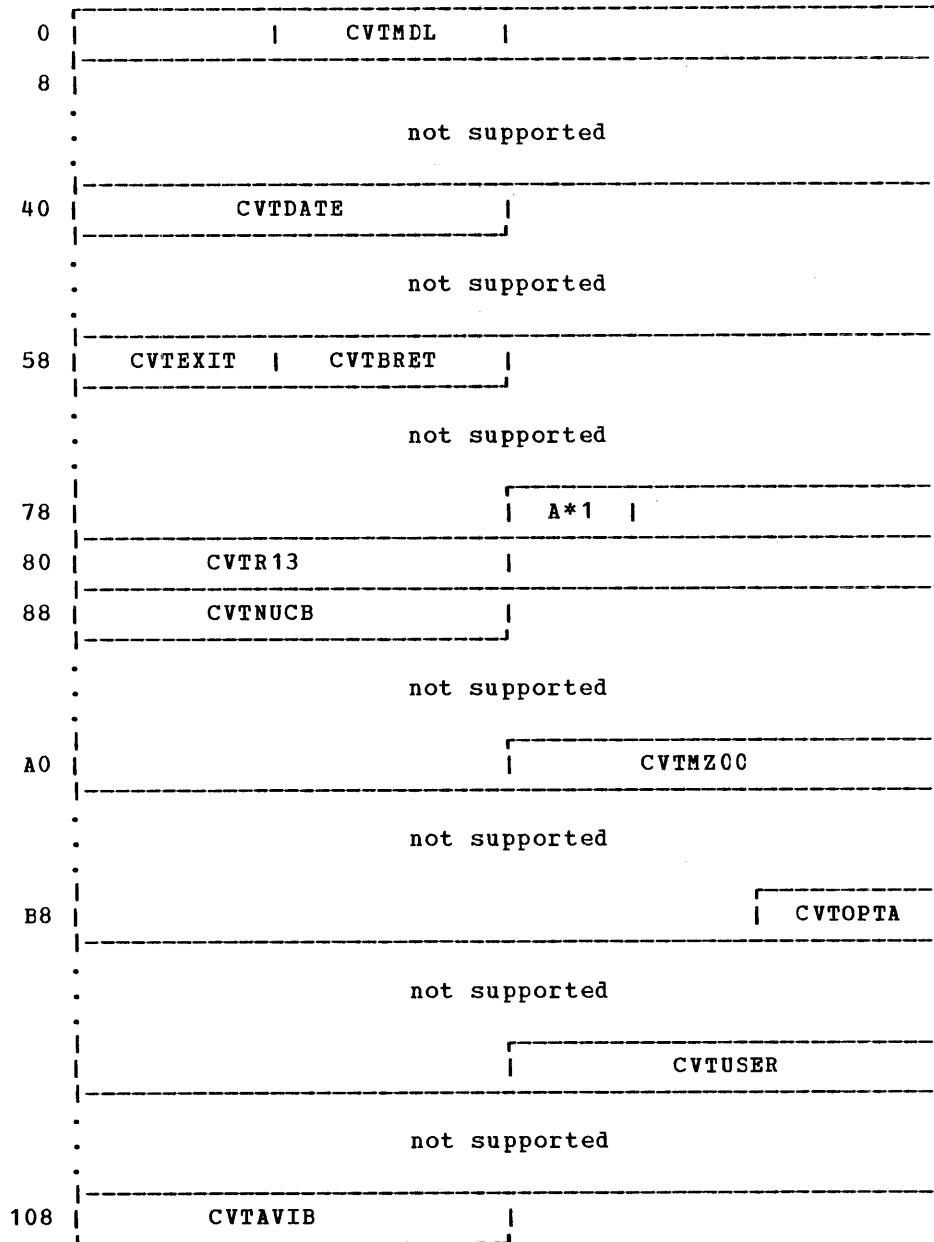
Terminal Attention Interrupt Element (TAIE)

98	TAIEMSGL	DS	2X	Reserved for IBM use
9A	TAIETGET	DS	1X	Reserved for IBM use
9B		DS	1X	Reserved for IBM use
9C	TAIEIAD	DS	A	Right half I/O old PSW
A0	TAIERSAV	DS	64F	Register 0-15 of interrupted program

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 (hex 10) in NUCON points to CMSCVT.

```
        r-------------------------------------------------------------------1
    0   |                   |     CVTMDL      |                             |
        |-------------------------------------------------------------------|
    8   |                                                                   |
        .                                                                   .
        .                        not supported                             .
        |-------------------------------------------------------------------|
   40   |         CVTDATE              |                                     |
        |-----------------------------J                                     |
        .                        not supported                             .
        .                                                                   .
        |-------------------------------------------------------------------|
   58   |  CVTEXIT   |   CVTBRET       |                                     |
        |-----------------------------J                                     |
        .                        not supported                             .
        .                                             r---------------------1
   78   |                                             |   A*1   |           |
        |-------------------------------------------------------------------|
   80   |      CVTR13                  |                                     |
        |------------------------------------------------------------------|
   88   |      CVTNUCB                 |                                     |
        |-----------------------------J                                     |
        .                        not supported                             .
        .                                     r-----------------------------1
   A0   |                                     |        CVTMZOC               |
        |-------------------------------------------------------------------|
        .                        not supported                             .
        .                                             r---------------------1
   B8   |                                             |     CVTOPTA         |
        |-------------------------------------------------------------------|
        .                        not supported                             .
        .                                     r-----------------------------1
        |                                     |        CVTUSER               |
        |-------------------------------------------------------------------|
        .                        not supported                             .
        .                                                                   J
        |-------------------------------------------------------------------
  108   |      CVTAVIB                 |
        L-----------------------------J
```

Hexadecimal Displacement	Field Name			Field Description, Contents, Meaning

Communication Vector Table as Supported by CMS

0		DC	H'0'		Reserved for IBM use
2	CVTMDL	DC	H'0'		CPU model ID
4		DC	CL4'C3.0'		CMS release number
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'		AN SVC 3 instruction (exit)
5A	CVTBRET	DC	XL2'07FE'		A BCR 15,14 instruction
5C		DC	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		DC	8F'-1'		Not supported
AC	CVTMZ00	DC	A(0)		Highest storage address in machine
B0		DC	3F'-1'		Not supported
BC		DC	XL2'00'		Not supported
BE	CVTOPTA	DC	XL2'00'		Bit 7: extended precision floating point hardware
C0		DC	2F'-1'		Not supported
C8		DC	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

DBGSECT: DEBUG WORK AREA

DBGSECT contains the files used by DEBUG for saving registers, breakpoints, PSWs, and other data.

V-constants in DMSDBD, DMSDBG, and DMSITE point to the DEBUG work area.

```
       r--------------------------------------------------------------¬
  0  |       CCWPRINT          |////////////////////////////////|
     |--------------------------------------------------------------|
  8  |                        SYMTABLE                             |
     .                                                            .
     .                                                            .
     .                                                            .
     |--------------------------------------------------------------|
 108 |       SYMTBG            |                                   |
     |------------------------J                                   |
     .                        BRKPNTBL                            .
     .                                                            .
     .            r---------------------------------------------- |
 188 |            |                      TBLEND                   |
     |------------|-------------------------------------------------|
 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 |                        DBGSAV1                              |
     .                                                            .
     .                                                            .
     |--------------------------------------------------------------|
 248 |                        DBGSAV2                              |
     .                                                            .
     .                                                            .
     L--------------------------------------------------------------
```

```
288  |----------------------------------------------------------------|
290  |                              TSYM                              |
     |----------------------------------------------------------------|
298  |                              XPSW                              |
     |----------------------------------------------------------------|
2A0  | A*2  |                       LINE                              |
     |------------|                                                   |
2A8  |                                      |-------------------------|
     |--------------------------------------|                         |
     |                              LINE1                             |
     .                                                                .
     .                                                                .
     |                        |---------------------------------------|
300  |                        | A*3  |                                |
     |----------------------------------------------------------------|
     .                                                                .
     .                                                                .
     .                                                                .
     |----------------------------------------------------------------|
320  |                              | A*4  | A*5  |                    |
     |----------------------------------------------------------------|
328  |                        PLIST for DMSCWR                        |
     |----------------------------------------------------------------|
330  |                                |                       | A*6   |
     |----------------------------------------------------------------|
338  | INPUTSIZ      |                |                               |
     |----------------------------------------------------------------|
340  |                              CONHCT                            |
     |----------------------------------------------------------------|
348  |         HEX            |               |                       |
     |----------------------------------------------------------------|
350  |                              HEXHEX                            |
     |----------------------------------------------------------------|
358  |       |                      BITS                             |
     |----------------------------------------------------------------|
360  |       |                                                       |
     |-------|                                                       |
     .                                                                .
     .                                                                .
     .                                                                .
     |                                             |------------------|
     |                                             | A*7  |           |
     |----------------------------------------------------------------|
380  |                              YPSW                              |
     |----------------------------------------------------------------|
388  |          TBLINDX              |        BCR      |              |
     |----------------------------------------------------------------|
390  | A*8  | A*9  | A*10 | A*11 |            DBGXWK                  |
     |----------------------------------------------------------------|
398  |DBGXWK(Cont) |               |               |                 |
     |----------------------------------------------------------------|
```
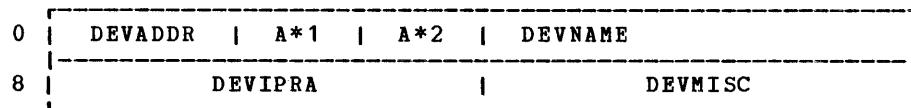
Hexadecimal Displacement	Field Name			Field Description, Contents, Meaning
		ENTRY	DBGSECT	
0	CCWPRINT	CCW	X'09',0,SILI,133	Printer CCW for DEBDUMP
8	SYMTABLE	DS	32D	User-defined symbol table
108	SYMTBG	DC	F'0'	Symbol table entries
10C	BRKPNTBL	DC	16F'0,-1'	Breakpoint table
18C	TBLEND	DC	A(TBLEND)	End address of breakpoint table

Storage and Constants for NEWLIN and Control

190		DS	0D	
190	ARGS	DS	5D	Arguments stored here
	MVCNT1	EQU	*-ARGS	Number of bytes in args
	MVCNT	EQU	MVCNT1-8	Needed for set GPR command
	ARGMAX	EQU	*	End of argument area
1B8	JFLAGS	DS	6X	Flags corresponding to ARGS

One Flag for Each Parameter

```
                                       00 = Numeric (0 - 9)
                                       F0 = Hex (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

The Following Variables Are Used by DEBUG and DEBDUMP

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 for line count
204	SAVE2	DS	1F	
208	DBGSAV1	DS	16F	DEBUG BALR call save
248	DBGSAV2	DS	16F	Save area for CONWAIT/CONREAD
288	TSYM	DS	4F	Symbol entry

Hexadecimal Displacement	Field Name				Field Description, Contents, Meaning
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		Hex 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
33C		DC	F'0'		
340	CONHCT	DC	X'FAFBFCFDFEFF0000'		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'C0C0C0C0C0C0C0C0'		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

Bits defined in DBGSWTCH

		EQU	X'80'		Reserved for IBM use
		EQU	X'40'		Reserved for IBM use
	DBDEXIT	EQU	X'20'		Signals exit from DEBDUMP
	DBDDMSG	EQU	X'10'		Signals duplicate msg in DEBDUMP
	DBGSET	EQU	X'08'		Signals SET command
	DBGPERM	EQU	X'04'		Reserved for IBM use
	DBGCOND	EQU	X'02'		Reserved for IBM use

The Following are Reserved for IBM Use

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 op code
	BALR	EQU	X'05'		BALR op code
394	DBGXWK	DS	3H		Recreate instruction at BRKPT addr
39A		LPSW	48		Give control to NSI

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.

```
    r------------------------------------------------------------1
0 | DEVADDR | A*1 | A*2 | DEVNAME                          |
  |------------------------------------------------------------|
8 |              DEVIPRA              |         DEVMISC       |
    L_____J
```

Hexadecimal Displacement	Field Name				Field Description, Contents, Meaning
0	DEVADDR	DS	1H		Virtual device·address
2	DEVFLAG	DS	1X	A*1	Device flags
3	DEVTYPE	DS	1X	A*2	Device type
4	DEVNAME	DS	1F		Symbolic 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).

The device table is pointed to by V-constants in DMSIOW and DMSITI, and is also referenced indirectly by the ADEVTAB field (hex 608) in NUCON.

offset			offset	
0	CONSOLE		A0	ZDISK
8			A8	
10	ADISK		B0	READER1
18			B8	
20	BDISK		C0	PUNCH1
28			C8	
30	CDISK		D0	PRINTER1
38			E0	READER2
40	DDISK		E8	
48			F0	PUNCH2
50	EDISK		F8	
58			100	PRINTER2
60	FDISK		108	
68			110	TAPE1
70	GDISK		118	
78			120	TAPE2
80	SDISK		128	
88			130	TAPE4
90	YDISK		138	
98			140	DUMMY
			148	

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		DC	VL4(CONSI)	
C		DC	XL4'0'	
10	ADISK	DS	0D	Device table entry for A-disk
10		DC	XL2'191'	
12		DC	XL2'0'	
14		DC	CL4'DSK1'	
18		DC	AL4(0)	
1C		DC	XL4'0'	
20	BDISK	DS	0D	Device table entry for B-disk
20		DC	XL2'000'	
22		DC	XL2'0'	
24		DC	CL4'DSK2'	
28		DC	AL4(0)	
2C		DC	XL4'0'	
30	CDISK	DS	0D	Device table entry for C-disk
30		DC	XL2'000'	
32		DC	XL2'0'	
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'	

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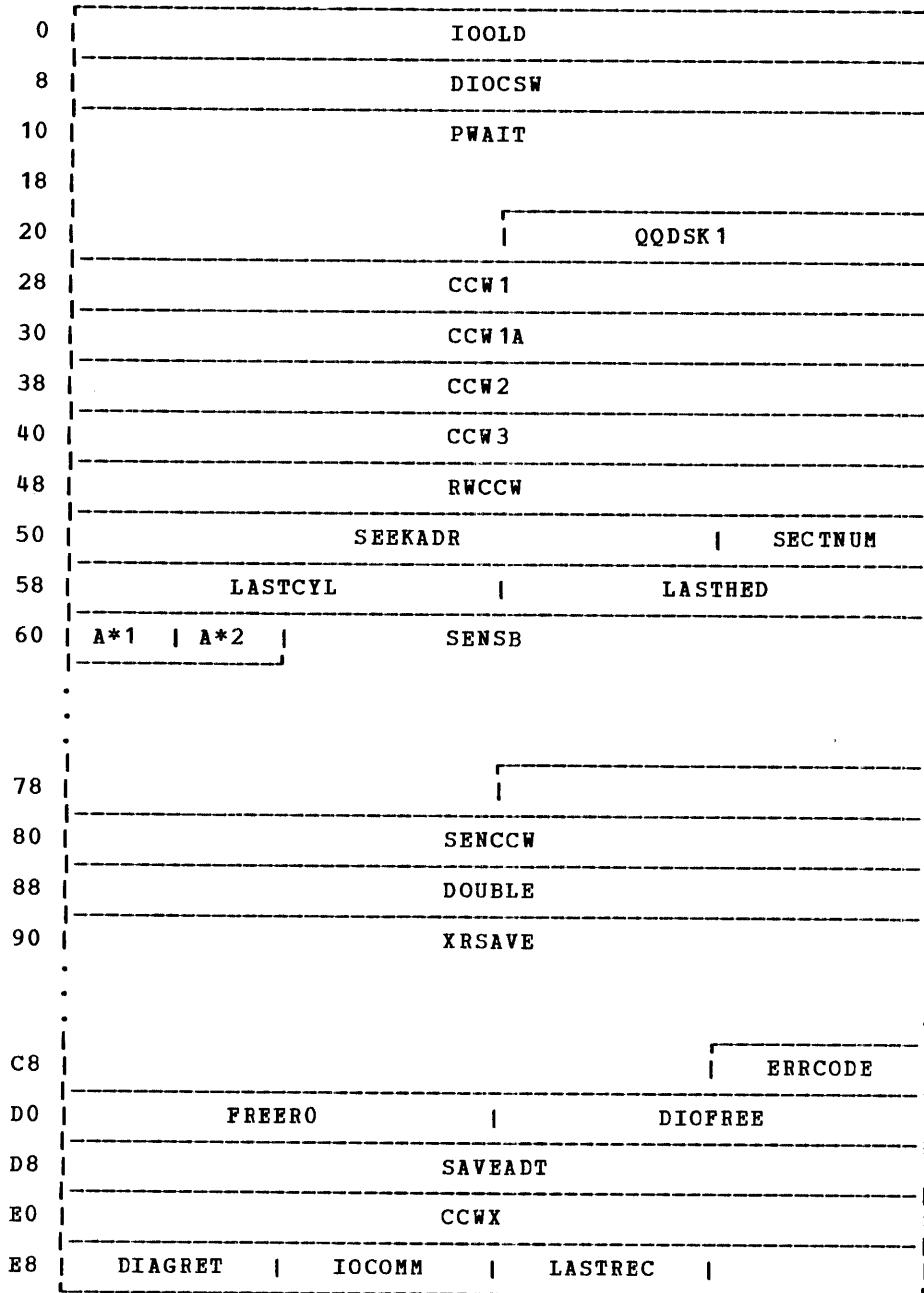
Hexadecimal Displacement	Field Name			Field Description, Contents, Meaning
130	TAPE3	DS	0D	Device table entry for TAPE3
130		DC	XL2'183'	
132		DC	XL2'0'	
134		DC	CL4'TAP3'	
138		DC	AL4(0)	
13C		DC	XL4'0'	
140	TAPE4	DS	0D	Device table entry for TAPE4
140		DC	XL2'184'	
142		DC	XL2'0'	
144		DC	CL4'TAP4'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

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.

The Disk I/O Work Area is pointed to by a V-constant in DMSNUC, and referenced indirectly by ADIOSECT (hex 660) in NUCON.

```
 0  |-------------------------------------------------------|
    |                        IOOLD                          |
    |-------------------------------------------------------|
 8  |                        DIOCSW                          |
    |-------------------------------------------------------|
10  |                        PWAIT                           |
    |                                                       |
18  |                                                       |
    |                             r-------------------------|
20  |                             |          QQDSK1         |
    |-------------------------------------------------------|
28  |                        CCW1                            |
    |-------------------------------------------------------|
30  |                        CCW1A                           |
    |-------------------------------------------------------|
38  |                        CCW2                            |
    |-------------------------------------------------------|
40  |                        CCW3                            |
    |-------------------------------------------------------|
48  |                        RWCCW                           |
    |-------------------------------------------------------|
50  |            SEEKADR               |     SECTNUM         |
    |-------------------------------------------------------|
58  |       LASTCYL           |           LASTHED           |
    |-------------------------------------------------------|
60  | A*1  | A*2  |           SENSB                          |
    |------------------------'                              |
    .                                                       .
    .                                                       .
    .                                                       .
78  |                             r-------------------------|
    |                             |                         |
    |-------------------------------------------------------|
80  |                        SENCCW                          |
    |-------------------------------------------------------|
88  |                        DOUBLE                          |
    |-------------------------------------------------------|
90  |                        XRSAVE                          |
    .                                                       .
    .                                                       .
    .                                                       .
C8  |                             r-------------------------|
    |                             |       ERRCODE           |
    |-------------------------------------------------------|
D0  |           FREERO            |         DIOFREE          |
    |-------------------------------------------------------|
D8  |                        SAVEADT                         |
    |-------------------------------------------------------|
E0  |                        CCWX                            |
    |-------------------------------------------------------|
E8  |  DIAGRET    |  IOCOMM    |  LASTREC   |                |
    |-------------------------------------------------------|
```

```
Hexadecimal      Field
Displacement     Name                          Field Description, Contents, Meaning
-----------      -----                         ------------------------------------------
        0        DIOSECT  DSECT
        0        IOOLD    DC    1D'0'           I/O old PSW (from interrupt routine)

        8        DIOCSW   DC    1D'0'           CSW (from interrupt routine)

                                                PLIST to call DMSIOW

       10                 DS    0F
       10        PWAIT    DC    CL8'WAIT'
       18                 DC    C'DSK-'         Filled in to correct symbolic disk no.
       1C                 DC    F'0'
       20                 DC    F'0'

       24        QQDSK1   DC    F'0'            1st two bytes are always 0
                 QQDSK2   EQU   QQDSK1+2        Halfword copy of 16th track disk-address

                 CCW Chain

       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 info (1st 3 bytes are 0)

       57        SECTNUM  DC    X'00'           Sector number

                 I/O Information

       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:

                 Bits defined in DIOFLAG
                 TOOBIG   EQU   X'04'           Byte count > 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

       7C
       80        SENCCW   CCW   X'04',SENSB,X'20',24 READ 24 BYTES SILI

                 Miscellaneous Storage

       88        DOUBLE   DC    1D'0'           (Scratch area, for CVD use etc.)

                 Keep the following three in order

       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)
```

Hexadecimal Displacement	Field Name			Field Description, Contents, Meaning

Keep the following two in order

D0	FREERO	DC	F'0'	No. doublewords of free storage (if any)
D4	DIOFREE	DC	F'0'	Add. of free storage for buffer or CCW'S
D8	SAVEADT	DC	F'0'	Handy place for an ADT address
DC				
E0	CCWX	CCW	X'23',SECTNUM,X'40',1 Set sector	
E8	DIAGRET	DC	X'00'	CPs DIAGNOSE return code if nonzero
E9	IOCOMM	DC	X'00'	Set to read (06) or write (05)
EA	LASTREC	DC	X'00'	Number (1-14) of the last record processed

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).

```
     .------------------------------------------------------------------.
  0  | CCBCNT    |  A*1  |  A*2  |  A*3  |  A*4  | A*5  | A*6 |
     |------------------------------------------------------------------|
  8  |  A*7   | CCBCCW           |  A*8  | CCBCSW                        |
     |------------------------------------------------------------------|
 10  |     CCBLDATB              |       CCBLCCWB                        |
     |------------------------------------------------------------------|
 18  |                          | A*9   | CCBFSCCW                       |
     |------------------------------------------------------------------|
 20  |     CCBRDCCW             |        CCBWTCCW                        |
     |------------------------------------------------------------------|
 28  |     CCBLWCCW            |                                         |
     |------------------------------------------------------------------|
 30  |                        |                                         |
     |------------------------------------------------------------------|
 38  |     CCBNCCB            |                                         |
     '------------------------------------------------------------------'
```

Hexadecimal Displacement	Field Name			Field Description, Contents, Meaning
	CCBST	EQU	*	Start CCB
	CCBD	EQU	*	Command control block
0	CCBLEN	DS	0CL16	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

Bits defined in CCBCOM1

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

Bits defined in CCBCOM2

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
	Bits defined in CCBCSW1:				
	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
	Bits defined in CCBCSW2:				
	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		Symbolic unit (SYSUN)
6	CCBSUCLS	DS	XL1	A*5	U - LUB class
7	CCBSUNUM	DS	XL1	A*6	N - LUB number within class
8	CCBLIOBS	DS	XL1	A*7	Reserved for LIOBS
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
	Bits defined in CCBUFLGS:				
	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

CCBLWCCW chain field must have same displacement as FCBCHAIN in FCDB and BKPFSTBK in BKPHD.

| 38 | CCBNCCB | DS | A | | Address of next CCB block |
| 3C | | DS | F | | Reserved for IBM use |

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 (hex 4F0) points to the first DOSCB in the chain or if no chain exists contains zero.

```
     r-----------------------------------------------------------------1
  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  |          DOSOSFST                    |
     |-----------------------------------------------------------------|
 50  |      DOSOSDSN          |         DOSVOLTB                        |
     |-----------------------------------------------------------------|
 58  |      DOSEXTTB          |  DOSSENSE     | A*7  | A*8              |
     |-----------------------------------------------------------------|
 60  |      DOSBUFSP          |         DOSUCNAM                        |
     |-----------------------------------------------------------------|
 68  | DOSUCNAM (cont.)                      |                         |
     |---------------------------------------J                         |
     |                      DOSSAVE                                    |
     .                                                                 .
     .                                                                 .
     |                             r-----------------------------------|
 80  |                             | A*9  | A*10 |                      |
     L-----------------------------------------------------------------J
```

Hexadecimal Displacement	Field Name				Field Description, Contents, Meaning
0	DOSINIT	DS	0X		DOSCB flag byte
	Bits defined in DOSINIT				
	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 distinguich from CMSCB
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		Records per CMS physical block
38	DOSREAD	DS	F		Number of bytes actually read
3C	DOSDEV	DS	X	A*1	Device type code
	DOSDUM	EQU	0		Dummy device
	DOSDSK	EQU	2		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 assoc with unit
4A	DOSEXT	DS	1X	A*5	Number DOS extents left to process
4B	DOSEXTCT	DS	1X	A*6	Current DOS extent
4C	DOSOSFST	DS	F		Pointer to OS FST
50	DOSOSDSN	DS	F		Pointer to OS 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. volumes (entries in DOSVOLTB)
5F	DOSEXTNO	DS	X	A*8	No. extents (entries in DOSEXTTB)
60	DOSBUFSP	DS	F		Size of VSAM I/O buffer (s)
64	DOSUCNAM	DS	CL8		VSAM user catalog ddname
6C	DOSSAVE	DS	6F		Temp save for re-entrant code
84	DOSEXTCX	DS	1X		Current extent (used by DMSXCP)
85	DOSTYPE	DS	1C		Data set type (SAM=S, VSAM=A)
86		DS	H		Reserved for IBM use
88	DOSEND	DS	0D		End address of this block
	DOSENSIZ	EQU	(* DOSSECT)/8		Size of block in doublewords

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.

```
     r------------------------------------------------------------------------1
   0 |                             FNAME                                      |
     |-----------------------------------------------------------------------|
   8 |                             FTYPE                                      |
     |-----------------------------------------------------------------------|
  10 |    FMODE     | A*1  | A*2  |    TRUNCOL     |      ZONE1              |
     |-----------------------------------------------------------------------|
  18 |    ZONE2     |   VERCOL1   |    VERCOL2     |      VERLEN             |
     |-----------------------------------------------------------------------|
  20 |     SCRBUFAD              |             CARDINCR                      |
     |-----------------------------------------------------------------------|
  28 |  LMSTART    |   LIMINCR   | A*3  | A*4  |                             |
     |-----------------------------------------------------------------J      |
     .                                                                        .
     .                             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                 |                                           |
     |---------------------------J                                           |
     |                             AUTOREG                                   |
     .                                                                        .
     .                                                                        .
     .                                                                        .
     |-----------------------------------------------------------------------|
  C8 |    CARDNO                 |             COUNT                         |
     |-----------------------------------------------------------------------|
  D0 |                             LMCURR                                    |
     |-----------------------------------------------------------------------|
  D8 |    LMTEMP                 |             BUFFL                         |
     |-----------------------------------------------------------------------|
  E0 |    BUFFA                  |                                           |
     |---------------------------J                                           |
     |                             CANSAV                                    |
     .                                                                        .
     .                                                                        .
     L------------------------------------------------------------------------J
```

```
      r---------------------------------------------------------------¬
108|  |                           DUALNOS                             |
   .  |                                                               .
   .  |                                                               .
   .  |                                                               .
   .  |---------------------------------+-----------------------------|
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               |          ATTN              |
      L---------------------------------------------------------------
```

```
308 | ATTN (cont.)              |
    |--------------------------------------------------------------
310 |        ATTNLEN            |            RENLIST
    |--------------------------------------------------------------
318 | RENLIST (cont.)           |            RPLIST
    |--------------------------------------------------------------
320 |                        RPLIST
    |--------------------------------------------------------------
328 |        STRTNO             |            INCRNO
    |--------------------------------------------------------------
330 |        AINCORE            |            FSIZE
    |--------------------------------------------------------------
338 |                        DECLTH
    |--------------------------------------------------------------
340 |                        RANGE
    |--------------------------------------------------------------
348 |                        CANCCW
    |--------------------------------------------------------------
350 | A*8 |               | A*9 | A*10 |    WRCOUNT
    |--------------------------------------------------------------
358 |        BUFFLOC            |            ALINELOC
    |--------------------------------------------------------------
360 |        ANUMLOC            |            AFLAGLOC
    |--------------------------------------------------------------
368 | AUTOCNT  | AUTOCURR  | CHNGCNT  | DITCNT
    |--------------------------------------------------------------
370 | EDCT     | LINELOC   | NUMLOC   | SAVCNT
    |--------------------------------------------------------------
378 | TVERCOL1 | TVERCOL2  | A*11 | A*12 |    AREA
    |--------------------------------------------------------------
380 | AREA (cont.)                    | A*13 | A*14
    |--------------------------------------------------------------
388 |                        CHNGMSG
    |
390 |
    |
398 |                            CMODE
    |--------------------------------------------------------------
3A0 |                        FILEMS
    .                                            .
    .                                            .
    .                                            .
3B8 |          | A*15 | A*16 | A*17 |
    |--------------------------------------------------------------
    |                        JAR
    .                                            .
    .                                            .
    .                                            .
    |          |--------------------------------------------------
408 |          |             NEWNAME
    |--------------------------------------------------------------
410 |          |             NEWTYPE
    |--------------------------------------------------------------
418 |    NEWMODE    |             SERSAVE
    |--------------------------------------------------------------
420 | SERSAVE (cont.)   |     SERTSEQ     | A*18 | A*19 |
```

```
      ┌──────────────────────────────────────────────────────────────┐
428│                            TEMPTAB                                │
   .                                                                  .
   .                                                                  .
   .                                                                  .
   │        ┌─────────────────────────────────────────────────────────┤
440│        │ A*20 │ A*21 │ A*22 │ A*23 │   SCLNO                     │
   │        ├─────────────────────────────────────────────────────────┤
448│         SCLNO (ccnt.)                     │ A*24 │//////│
   │        ├───────────────────────────┐                             │
450│    XAREA              │                                           │
   │    ├───────────────────────────────┘                             │
   .                                                                  .
   .                                                                  .
   │       ┌──────────────────────────────────────────────────────────┤
4D8│       │/////////│    YAREA         │                             │
   │       └──────────────────────────────┘                           │
   .                                                                  .
   .                                                                  .
   │                                              ┌──────────┐
560│                                              │ A*25 │
   │        ──────────────────────────────────────────────────────────┤
568│                            XXXCWD                                 │
   │        ──────────────────────────────────────────────────────────┤
570│                            SAVCWD                                 │
   │        ──────────────────────────────────────────────────────────┤
578│                            INVLD                                  │
   │        ──────────────────────────────────────────────────────────┤
   │                            EDLIN                                  │
   .                                                                  .
   .                                                                  .
   │        ──────────────────────────────────────────────────────────┤
600│                                         │   LINENO              │
   │        ──────────────────────────────────────────────────────────┤
608│  LINENO (cont.)           │ A*26│                                │
   │        ────────────────────────────────┘                         │
   │                            LINE                                   │
   .                                                                  .
   .                                                                  .
   │                           ┌──────────────────────────────────────┤
6A8│                           │ A*27 │                               │
   │        ───────────────────────────┘                              │
   │                            TABLIN                                 │
   .                                                                  .
   .                                                                  .
   └──────────────────────────────────────────────────────────────────┘
```

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	CL' '		Filemode
12	FV	DS	CL1	A*1	Record format
13	CASESW	DS	CL1	A*2	Case setting
14	TRUNCOL	DS	H		Truncation column
16	ZONE1	DS	H		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		Addr of GETMAIN buffer
24	CARDINCR	DS	F		Increment for serialization
28	LMSTART	DS	H		Where line nums start
2A	LMINCR	DS	H		Auto. line nums default incr.
2C	FLAG	DS	CL1	A*3	Flags for line monitoring
2D	FLAG2	DS	CL1	A*4	Misc. 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 BLOCX

Note: PADBUF must remain directly behind PADCHAR

Hexadecimal Displacement	Field Name			Field Description, Contents, Meaning
4C	PADBUF	DS	9C	Pad characters
	PTRCONS	EQU	*	Same as PTRCONS in DMSEDI
58	PTR1	DS	2F	'TOP' pointer (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	CORITEM	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	FREELEN	DS	F	Length of free storage
84	FREEAD	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 adr; 0 if LOADMOD
	EPTRCONS	EQU	*	
94	AUTOREG	DS	13F	Autocheck save area
C8	CARDNO	DS	F	Save area for sequence number
CC	COUNT	DS	F	Number of chars in EDLIN
D0	LMCURR	DS	2F	Prompter current line no.
D8	LMTEMP	DS	F	WRTYPE/LINEMODE save area
DC	BUFFL	DS	F	Length of string (EDC)
E0	BUFFA	DS	F	Address of string (EDC)
E4	CANSAV	DS	9F	Register save (EDC)
108	DUALNOS	DS	CL240	Temp string buffer (EDC)
1F8		DS	0D	
1F8	DECIMAL	DS	F	Used by DECBIN & BINDEC
1FC	HALF	DS	F	BINDEC only edits 4 chars
200	REGSAV	DS	5F	Register save area
214	REGSAVX	DS	3F	Register save area

Hexadecimal Displacement	Field Name				Field Description, Contents, Meaning
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		No. of lines to change
268	TIN	DS	0F		WAITRD PLIST
268		DS	CL8		'WAITRD'
270		DS	X		'1' for console no. 1
271	AEDLIN	DS	3X		A(EDLIN)
274	CASEREAD	DS	C	A*6	Default to uppercase
275		DS	3X		Length put here
278	TOUT	DS	0F		PLIST for call to DMSCWR
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' max length override
286		DS	H		Length goes here
	CRBIT	EQU	X'80'		Suppress carriage return
288	IOLIST	DS	0F		Initial 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		'****' Don't allow '*'s
2A8		DS	F		133 (XINSCRIPT use LINE)
2AC		DS	CL2		F/V and null record indicator
2AE	RECS	DS	H		1 (XINSCRIPT 1 line at a time)
2B0		DS	F		Number of bytes read from RDBUF
2B4	ALTLIST	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		New filemode
2E6		DS	CL6		New filemode
2EC		DS	8X		FF fence
2F4	STACKAT	DS	0F		PLIST to stack FIFO
2F4		DS	CL8		ATTN
2FC		DS	CL4		FIFO
300	STACKATL	DS	F		Length and address of line to stack
304	ATTN	DS	0F		
304		DS	CL8		ATTN
30C		DS	CL4		LIFO
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	CANCCW	DS	D		Cancel screen CCW
350	CMDBLOK	DS	X	A*8	X19
351	BUFAD	DS	3X		Buffer address
354	FLG	DS	X	A*9	CCW flag
355	CTL	DS	X	A*10	Control byte
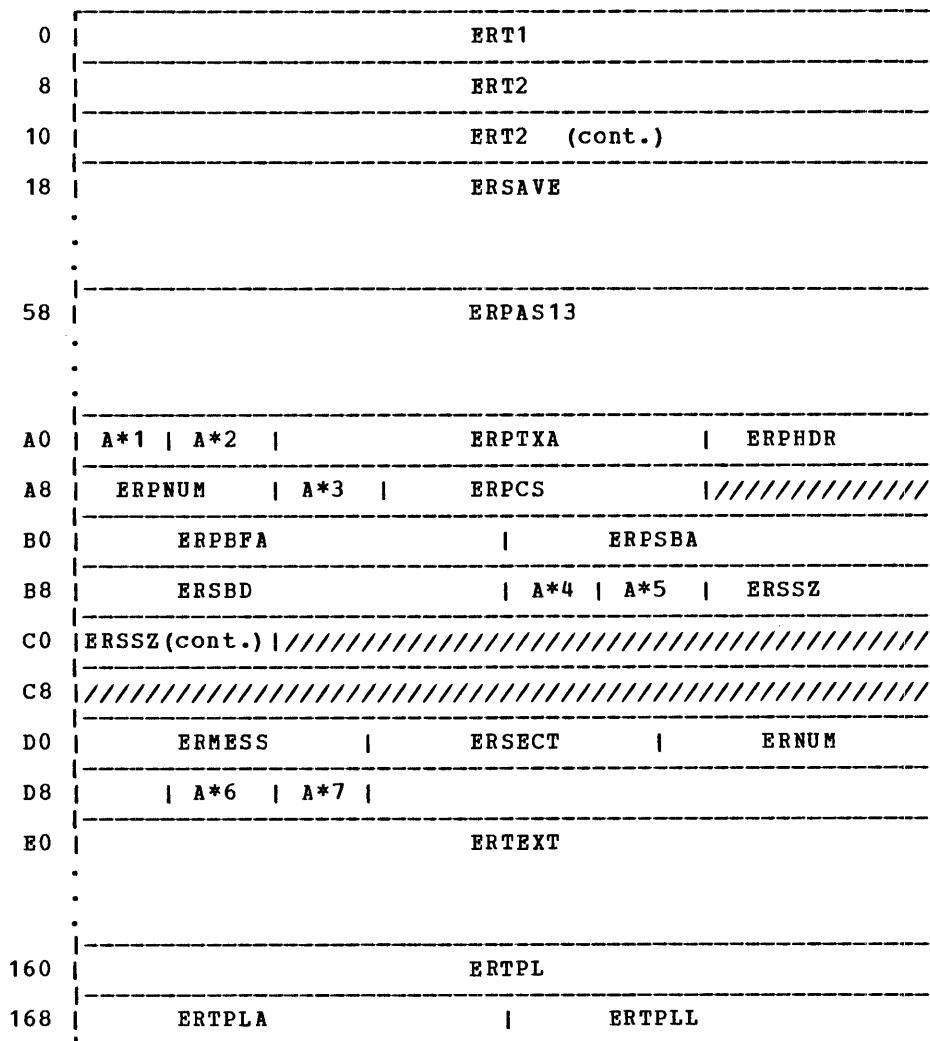
Hexadecimal Displacement	Field Name				Field Description, Contents, Meaning
356	WRCOUNT	DS	H		Write count
358	GIOPLIST	DS	0F		DMSSCR PLIST for DMSGIO
358	BUFFLOC	DS	F		Buffer location
35C	ALINELOC	DS	F		A(LINELOC)
360	ANUMLOC	DS	F		A(NUMLOC)
364	AFLAGLOC	DS	F		A(FLAGLOC)
368	AUTOCNT	DS	H		Autosave parameter
36A	AUTOCURR	DS	H		Current modification count
36C	CHNGCNT	DS	H		Temp area for change
36E	DITCNT	DS	H		No. lines stacked by ditto
370	EDCT	DS	H		Next char in EDLIN
372	LINELOC	DS	H		Display line number
374	NUMLOC	DS	H		Display count
376	SAVCNT	DS	H		Spot to save count for ditto
	TVERCOL1	DS	H		Temporary area for verify col 1
37A	TVERCOL2	DS	H		Temporary area for verify col 2
37C	ALCHAR1	DS	C	A*11	Temporary byte used by ALTER
37D	ALCHAR2	DS	C	A*12	Temporary byte used by ALTER
37E	AREA	DS	CL8		EDIT instruction work
386	BYTE	DS	X	A*13	Temporary byte (used by GET)
387	CHNGFLAG	DS	X	A*14	Flag for change
388	CHNGMSG	DS	CL20		Lines changed msg
39C	CMODE	DS	CL4		Filemode for MODECHK routine
3A0	FILEMS	DS	CL26		Retry message
3BA	FLAGLOC	DS	X	A*15	Flag for DMSGIO
3BB	GETFLAG	DS	X	A*16	Flag for GETFILE
3BC	HOLDFLAG	DS	X	A*17	DMSSCR SCRFLGS
3BD	JAR	DS	(ENDBLOC-BLOC)		AL1 Save area for preserve
409	NEWNAME	DS	CL8		Name area for FILE & SAVE
411	NEWTYPE	DS	CL8		Type area for FILE & SAVE
419	NEWMODE	DS	CL2		Mode area for FILE & SAVE
41B	SERSAV	DS	CL8		Serial no. save area
423	SERTSEQ	DS	CL3		Temporary byte serial area
426	SERTSW	DS	X	A*18	Temporary byte used by serial
427	SIGNAL	DS	X	A*19	Signal between routines
428	TEMPTAB	DS	(ENDTABS-TABS)		AL1 Temporary spot for new tabs
442	UTILFLAG	DS	X	A*20	DMSSCR utility flags
443	XYFLAG	DS	X	A*21	X/Y active flag
444	SCRFLGS	DS	X	A*22	Screen function flags
445	SCRFLG2	DS	X	A*23	More screen function flags
446	SCLNO	DS	8C		Save LINEMODE seq no
44E	TWITCH	DS	X	A*24	Location flags
450	XAREA	DS	H		X length and request buffer
452		DS	CL135		(Same length as EDLIN)
4DA	YAREA	DS	H		Y length and request buffer
4DC		DS	CL135		(Same length as EDLIN)
563					
564		CNOP	6,8		Alignment for XXXCWD
566		DS	X		More alignment
567	BLANK1	DS	X	A*25	Blank for clearing XXXCWD
568	XXXCWD	DS	CL8		EDIT token buffer
570	SAVCWD	DS	CL8		Spot to save XXXCWD
578	INVLD	DS	CL7		?EDIT:
57F	EDLIN	DS	CL135		Terminal input buffer
606	LINENO	DS	CL5		Line no. for typeout
60B	BLANK2	DS	X	A*26	Blank for clearing line
60C	LINE	DS	CL160		Current line is held here
6AC	BLANK3	DS	X	A*27	Blank for clearing TABLIN
6AD	TABLIN	DS	CL160		Output from spread
750	EDCBEND	DS	0D		
	EDCBLTH	EQU	(EDCBEND-EDCB)		Length of EDCB

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 DSECT.

```
     r------------------------------------------------------------------1
  0  |                            ERT1                                   |
     |------------------------------------------------------------------|
  8  |                            ERT2                                   |
     |------------------------------------------------------------------|
 10  |                            ERT2    (cont.)                        |
     |------------------------------------------------------------------|
 18  |                            ERSAVE                                 |
     .                                                                   .
     .                                                                   .
     .                                                                   .
     |------------------------------------------------------------------|
 58  |                            ERPAS13                                |
     .                                                                   .
     .                                                                   .
     |------------------------------------------------------------------|
 A0  | A*1 | A*2 |              ERPTXA            |    ERPHDR            |
     |------------------------------------------------------------------|
 A8  |   ERPNUM    | A*3 |       ERPCS         |///////////////|
     |------------------------------------------------------------------|
 B0  |     ERPBFA          |          ERPSBA                            |
     |------------------------------------------------------------------|
 B8  |     ERSBD          | A*4 | A*5 |      ERSSZ                      |
     |------------------------------------------------------------------|
 C0  |ERSSZ(cont.)|/////////////////////////////////////////////////|
     |------------------------------------------------------------------|
 C8  |////////////////////////////////////////////////////////////////|
     |------------------------------------------------------------------|
 D0  |     ERMESS      |      ERSECT       |       ERNUM               |
     |------------------------------------------------------------------|
 D8  |    | A*6  | A*7 |                                               |
     |------------------------------------------------------------------|
 E0  |                            ERTEXT                                |
     .                                                                   .
     .                                                                   .
     |------------------------------------------------------------------|
160  |                            ERTPL                                 |
     |------------------------------------------------------------------|
168  |      ERTPLA          |          ERTPLL                          |
     L------------------------------------------------------------------J
```
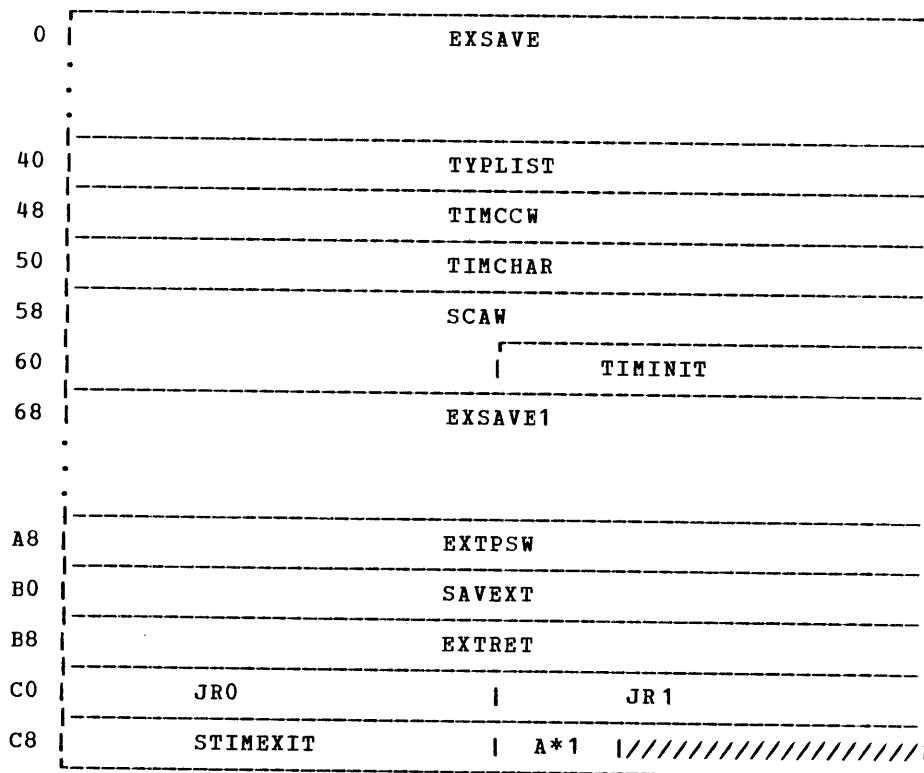
Hexadecimal Displacement	Field Name				Field Description, Contents, Meaning

Work Area for DMSERR Error Handling Routine

Hexadecimal Displacement	Field Name				Field Description, Contents, Meaning
0	ERT1	DS	D		Doubleword workspace
8	ERT2	DS	2D		Two doublewords workspace

Save Area

18	ERSAVE	DS	16F		

Reconstructed PLIST Area

58	ERPAS13	DS	18F		Pass this save area in reg 13
A0	ERPF1	DS	B	A*1	First flag byte

Bits defined in ERPF1

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'		Multiple substitutions (>1)

A1	ERPF2	DS	B	A*2	Second flag byte

Bits defined in ERPF2

ERF2CM	EQU	X'80'		Blank compression wanted
ERF2DT	EQU	X'40'		Dot at end of line wanted
ERF2DI	EQU	X'20'		HALT=YES wanted

Last 3 Bits Indicate DISP Field

ERF2ER	EQU	0		Errmsg
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)

Fields for Substitutions

B4	ERPSBA	DS	A		Pointer to first (next) group of substitution parameter in original PLIST
B8	ERSBD	DS	A		Data addr/value or current substitution parameter
BC	ERSBF	DS	B	A*4	Flag byte for current substitution parameter

Bits defined in ERSBF

ERSFLST	EQU	X'80'		The last substitution parameter
ERSFA	EQU	X'40'		A-type option
ERSFL	EQU	X'20'		Length specified

Hexadecimal Displacement	Field Name				Field Description, Contents, Meaning
	Last 3 Bits Give Option Type				
	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 current substitution parameter
C0	ERSSZ	DS	A		Size of substitution field (# dots − 1)
	Message Construction Area				
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	13		Maximum text size
DB	ERTEXT	DS	(ERTSIZE+1)C		Message text area
	TYPLIN PRINTER PLIST Construction Area				
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 (hex 6A0) in NUCON.

```
    ┌───────────────────────────────────────────────────────────┐
 0  │                          EXSAVE                            │
    .                                                            .
    .                                                            .
    .                                                            .
    │───────────────────────────────────────────────────────────│
40  │                          TYPLIST                           │
    │───────────────────────────────────────────────────────────│
48  │                          TIMCCW                            │
    │───────────────────────────────────────────────────────────│
50  │                          TIMCHAR                           │
    │───────────────────────────────────────────────────────────│
58  │                          SCAW                              │
    │                            ┌──────────────────────────────│
60  │                            │        TIMINIT                │
    │───────────────────────────────────────────────────────────│
68  │                          EXSAVE1                           │
    .                                                            .
    .                                                            .
    │───────────────────────────────────────────────────────────│
A8  │                          EXTPSW                            │
    │───────────────────────────────────────────────────────────│
B0  │                          SAVEXT                            │
    │───────────────────────────────────────────────────────────│
B8  │                          EXTRET                            │
    │───────────────────────────────────────────────────────────│
C0  │         JR0               │          JR1                   │
    │───────────────────────────────────────────────────────────│
C8  │       STIMEXIT            │   A*1  │////////////////////// │
    └───────────────────────────────────────────────────────────┘
```
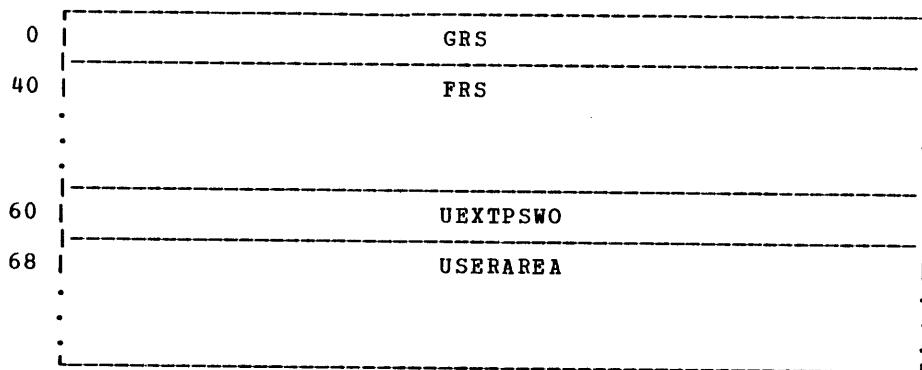
Hexadecimal Displacement	Field Name			Field Description, Contents, Meaning
	Storage for Timer Interrupt			
0	EXSAVE	DS	16F	Saved external old PSW
40	TYPLIST	DC	CL8'TYPLIN'	PLIST to type BLIP character
48	TIMCCW	DC	A(TIMCHAR)	
4C		DC	C'B',X'81',AL2(1)	
50	TIMCHAR	DC	X'FF',XL7'00'	BLIP character(s)
58	SCAW	DC	XL12'00'	Saved CSW/CAW
64	TIMINIT	DC	A(2000000/13)	Value to set timer = 2 seconds
	Storage for External (Other Than Timer) Interrupt			
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 DEBUG for external interrupt

Hexadecimal Displacement	Field Name			Field Description, Contents, Meaning

Storage for External Interrupt Set Up By Trap

Hexadecimal Displacement	Field Name				Field Description, Contents, Meaning
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	
	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 field USAVEPTR (hex 8C) in CMSAVE also points to the user area.

```
      ┌──────────────────────────────────────────────────┐
  0   │                      GRS                         │
      ├──────────────────────────────────────────────────┤
 40   │                      FRS                         │
      .                                                  .
      .                                                  .
      .                                                  .
      ├──────────────────────────────────────────────────┤
 60   │                    UEXTPSWO                      │
      ├──────────────────────────────────────────────────┤
 68   │                    USERAREA                      │
      .                                                  .
      .                                                  .
      .                                                  .
      └──────────────────────────────────────────────────┘
```

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: 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 CMSCB macro.

0	FCBNEXT	FCBPROC
8	FCBDD	
10	FCBOP	
18	FCBDSNAM	
20	FCBDSTYP	
28	FCBDSMD | FCBITEM	FCBBUFF
30	FCBBYTE	FCBFORM | FCBCOUT
38	FCBREAD | A*1 | A*2 | FCBXTENT	
40	FCBRECL | A*3 | A*4 | FCBMEMBR	
48	FCBMEMBR (cont.)	FCBOSFST
50	FCBOSDSN	FCBR13
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 |	
88	DEBTCBAD	SEBSAV
90	DEBOFLGS	DEBOPATB
98	IOBNXTAD	IOBECB
A0	DEBDCBAD	IOBECBPT
A8	IOBCSW	
B0	IOBSTART	IOBDCBPT

Format of location X'24' for console device:

20		FCBIOOUT
28	FCBIOOUT (Cont.)	FCBIOBUF
30	A*16 | A*17 | A*18 | FCBIOCNT	

Hexadecimal Displacement	Field Name				Field Description, Contents, Meaning
0	FCBINIT	DS	0X		Initialization flag bytes
	Bits defined in FCBINIT				
	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
	FCBOS	EQU	X'10'		FCB for OS formatted disk
	FCBDOSL	EQU	X'20'		Concatenated DOSLIB 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
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
	Bits defined in FCBDEV				
	FCBDUM	EQU	0		Dummy device
	FCBPTR	EQU	4		Printer
	FCBRDR	EQU	8		Reader
	FCBCON	EQU	12		Console terminal
	FCBTAP	EQU	16		Tape
	FCBDSK	EQU	20		Disk
	FCBPCH	EQU	24		Punch
	FCBCRT	EQU	28		CRT
3D	FCBMODE	DS	X	A*2	Mode: 1,2,3,4,5
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 DCB's using this FCB
44	FCBMEMBR	DS	2F		OS PDS member name
4C	FCBOSFST	DS	F		Pointer to OS dsname
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(PDS 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 (YDD)
6E	JFCBIND1	DS	X	A*5	Indicator 1
6F	JFCBIND2	DS	X	A*6	Indicator 2
70	JFCBUFNO	DS	X	A*7	Number of buffers
71	JFCBFTEK	DS	X		Buffering technique
71	JFCBFALN	DS	X	A*8	Buffer alignment
72	JFCBUFL	DS	H		Buffer length

Hexadecimal Displacement	Field Name			Field Description, Contents, Meaning
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 BDAM search limit
7A	FCBDSORG	DS	0X	Data set organization
7A	JFCDSORG	DS	2X	
7C	FCBRECFM	DS	0X	
7C	JFCRECFM	DS	X	A*12 Record format
7D	JFCOPTCD	DS	X	A*13 Option codes
	JCBBLKSZ	DS	0H	
7E	JFCBLKSI	DS	H	Block size
80	FCBLRECL	DS	0H	
80	JFCLRECL	DS	H	Logical length record
82	FCBIOSW	DS	X	A*14 I/O operation indicator

Bits defined in FCBIOSW

FCBCLOSE	EQU	X'80'		During CLOSE
FCBCLEAV	EQU	X'40'		DISP=LEAVE during CLOSE
FCBPROCC	EQU	X'20'		GOTO FCBPROC during CLOSE
FCBPROCO	EQU	X'10'		GOTO FCBPROC during OPEN
FCBCASE	EQU	X'08'		ON=LOWER CASE console I/O
FCBPVMB	EQU	X'04'		PUT-MOVE-VAR-BLK
FCBIOWR	EQU	X'02'		WRITE/PUT
FCBIORD	EQU	X'01'		READ/GET

83	FCBIOSW2	DS	1X	A*15 I/O operation indicators

Bits defined in FCBIOSW2

FCBMVPDS	EQU	X'01'		Switch for MOVEFILE with PDS option
FCBMMV	EQU	X'02'		Move PDS switch for FIND
84	DEBLNGTH	DS	0X	L'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	SEB return address
90	DEBOFLGS	DS	4X	Data set status flags
	DEBOPATB	DS	4X	OPEN/CLOSE option byte
98	IOBFLG	DS	0X	(Start of IOBPREFIX for normal scheduling

Bits defined in IOBFLG

IOBBFLG	EQU	0X		Displacement of IOB flag in IOB
IOBOUT	EQU	X'40'		WRITE,PUT in process
IOBIN	EQU	X'20'		READ,GET in process
IOBUPD	EQU	X'10'		QSAM PUTX in process

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

Bits defined in IOBECBCC

IOBBECBC	EQU	12		Displacement of ECB code in IOB
IOBBECBP	EQU	12		Displacement of ECB pointer in IOB

A4	IOBECBPT	DS	A	A(event control block)
A8	IOBFLAG3	DS	0X	I/O error flag
	IOBBCSW	EQU	16	Displacement of CSW in IOB
A8	IOBCSW	DS	8X	Last CCW stored(i.e., residual count)

Hexadecimal Displacement	Field Name			Field Description, Contents, Meaning
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,IOB blocks
	FCBENSIZ EQU	(*—FCESECT)/8		Size of FCB entry, doublewords
B8	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	A*18	Length of data buffer

Data Event Control Block (DECB)

0	IHADECB DSECT		
0	DECSDECB DS	F	Event control block
4	DECTYPE DS	H	Type of I/O request

Bits defined in DECTYPE
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 & data, buffer)
10	DECIOBPT DS	A	V(IOB)

BDAM Extension

14	DECKYADR DS	A	V(key)
18	DECRECPT DS	A	V(block reference field)

Some Frequently Used Equates
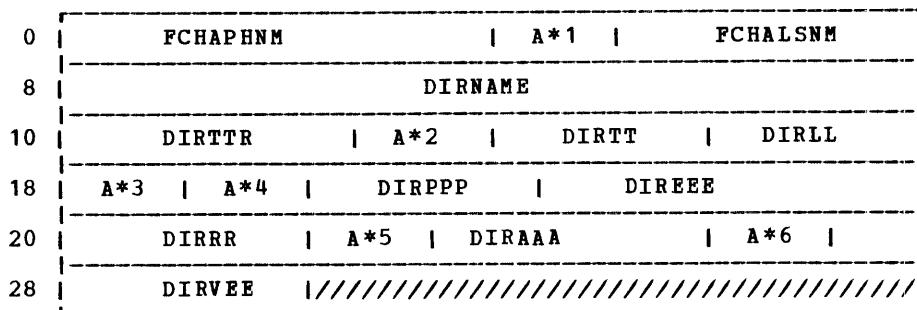
DDNAM	EQU	FCBDSTYP	Filetype = data set name
BLK	EQU	X'10'	RECFM=blocked records
BS	EQU	X'20'	MACRF=BSAM
DA	EQU	X'20'	DSORG=direct access
FXD	EQU	X'80'	RECFM=fixed—length records
IS	EQU	X'80'	DSORG=indexed sequential
LOC	EQU	X'08'	MACRF=locate mode
MOV	EQU	X'10'	MACRF=move mode
PS	EQU	X'40'	DSORG=physical 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

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 (hex F0) in the SYSCOM block points to the fetch table.

```
     r------------------------------------------------------------------1
 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      |//////////////////////////////////////////////|
     L------------------------------------------------------------------J
```
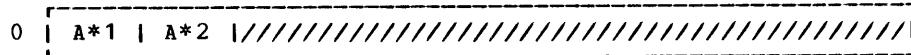
Hexadecimal Displacement	Field Name			Field Description, Contents, Meaning

8-Byte Parameter List Pointing to Directory List

0	FCHAPHNM	DC	A(DIRNAME)	Address of phase name
4	FCHOPT	DC	X'00' A*1	Options
5	FCHALSNM	DS	AL3	Address of listname

34-Byte Directory List

8	DIRNAME	DS	CL8	Phase name
10	DIRTTR	DS	XL3	Phase TTR
13	DIRN	DS	XL1 A*2	No. halfwords in directory
14	DIRTT	DS	XL2	No. text blocks in phase
16	DIRLL	DS	XL2	Length last text block
18	DIRC	DS	XL1 A*3	Flag byte

Bits defined in DIRC

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 priv CIL
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	Number RLD items in phase
22	DIRR	DS	XL1 A*5	No. 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 (42) in bytes
	FCHLENDW	EQU	(FCHLENG+7)/8	Total length in doublewords

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 (hex 48) in the BGCOM block points to the FICL block.

```
      r---------------------------------------------------------------------1
0   | A*1  |  A*2  |//////////////////////////////////////////////////////|
      L---------------------------------------------------------------------J
```
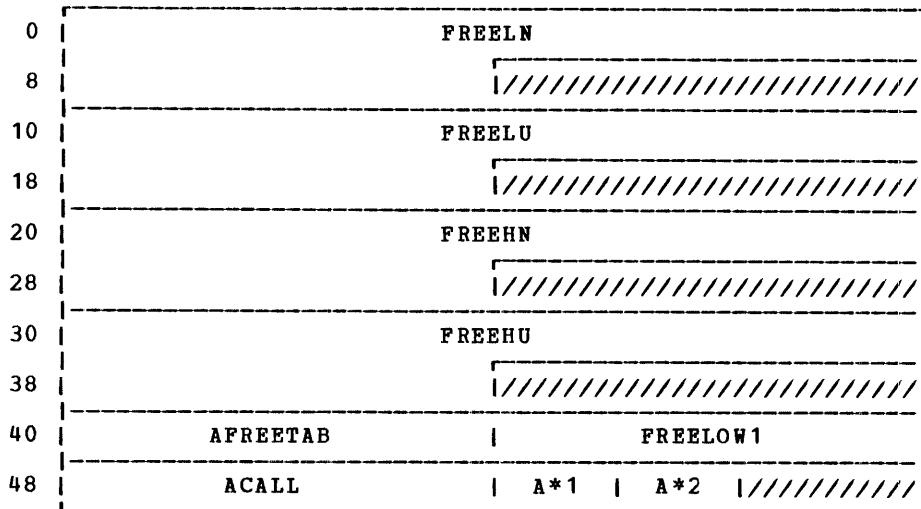
Hexadecimal Displacement	Field Name				Field Description, Contents, Meaning
0	SYSINDX	DC	AL1(0)	A*1	First BG system LUB index
2	PROGINDX	DC	AL1(14)	A*2	Programmer LUB index

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 (hex 530) in NUCON.

```
     r---------------------------------------------------------------------¬
  0  |                              FREELN                                  |
     |                   r-------------------------------------------------|
  8  |                   |//////////////////////////////////////|
     |-------------------------------------------------------------------|
 10  |                              FREELU                                  |
     |                   r-------------------------------------------------|
 18  |                   |//////////////////////////////////////|
     |-------------------------------------------------------------------|
 20  |                              FREEHN                                  |
     |                   r-------------------------------------------------|
 28  |                   |//////////////////////////////////////|
     |-------------------------------------------------------------------|
 30  |                              FREEHU                                  |
     |                   r-------------------------------------------------|
 38  |                   |//////////////////////////////////////|
     |-------------------------------------------------------------------|
 40  |        AFREETAB            |            FREELOW1                     |
     |-------------------------------------------------------------------|
 48  |        ACALL              |  A*1  |  A*2  |///////////|
     L---------------------------------------------------------------------
```

Hexadecimal Displacement	Field Name			Field Description, Contents, Meaning

The Following Are the Four Free Chain Element Header Blocks

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	Point 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'	Clobbered 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
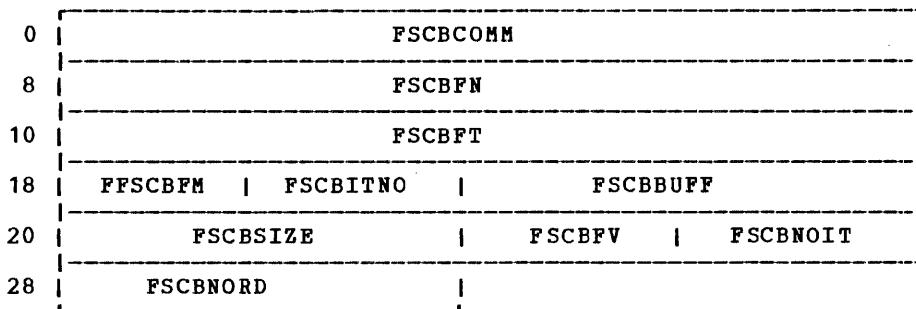
Bits defined in SKEY

USERKEY	EQU	X'E0'	User storage key
NUCKEY	EQU	X'F0'	Nucleus storage key
TCODE	EQU	14	FREETAB table code

Hexadecimal Displacement	Field Name			Field Description, Contents, Meaning
	Bits defined in TCODE			
	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 FREETAB table
	FREELOW1	DC	A(0)	Address of caller (for errors) (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'	TYPCALL equals BALR in macro

The Following Byte Holds Flags Internal to the DMSFRE Routine

04D	FREEFLG2 DC	BL1'0'	A*2

	Bits defined in FREEFLG2			
	FRF2CL	EQU	X'80'	Cleanup flag
	FRF2SVP	EQU	X'40'	SCHVPGE flag
	FRF2NOI	EQU	X'20'	2nd 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: FILE SYSTEM CONTROL BLOCK
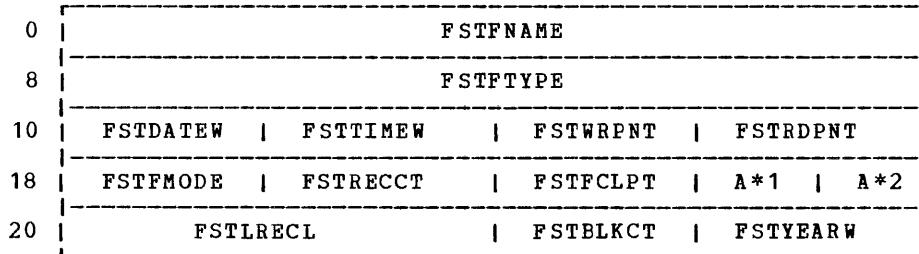
FSCBD is a PLIST defined for general use by routines that use the CMS file system. FSCBD is generated when the user invokes the FSCBD macro.

```
    ┌────────────────────────────────────────────────────────────────┐
 0  │                          FSCBCOMM                               │
    ├────────────────────────────────────────────────────────────────┤
 8  │                          FSCBFN                                 │
    ├────────────────────────────────────────────────────────────────┤
10  │                          FSCBFT                                 │
    ├───────────────────┬───────────────────┬────────────────────────┤
18  │     FFSCBFM       │   FSCBITNO        │        FSCBBUFF         │
    ├───────────────────┴──────────────┬────┴──────────┬─────────────┤
20  │           FSCBSIZE               │    FSCBFV     │   FSCBNOIT   │
    ├──────────────────────────────────┴───────────────┴─────────────┘
28  │      FSCBNORD            │
    └─────────────────────────┘
```

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	AS	H	Address of R/W buffer or of STATEFST
20	FSCBSIZE	DS	F	Length of buffer
22	FSCBFV	DS	CL2	RECFM – C'F' or C'V'
24	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.

```
     r--------------------------------------------------------------¬
  0  |                          FSTFNAME                            |
     |--------------------------------------------------------------|
  8  |                          FSTFTYPE                            |
     |--------------------------------------------------------------|
 10  |  FSTDATEW  |  FSTTIMEW    |  FSTWRPNT  |     FSTRDPNT         |
     |--------------------------------------------------------------|
 18  |  FSTFMODE  |  FSTRECCT    |  FSTFCLPT  |  A*1  |  A*2         |
     |--------------------------------------------------------------|
 20  |     FSTLRECL            |  FSTBLKCT   |  FSTYEARW            |
     L--------------------------------------------------------------J
```
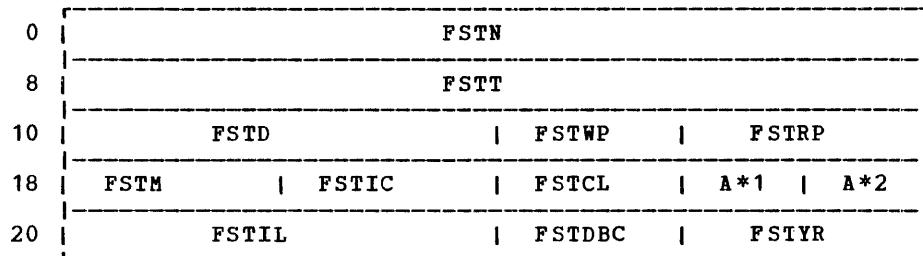
Hexadecimal Displacement	Field Name				Field Description, Contents, Meaning
0	FSTFNAME	DS	1D		Filename
8	FSTFTYPE	DS	1D		Filetype
10	FSTDATEW	DS	1H		Date last written – MMDD
12	FSTTIMEW	DS	1H		Time last written – HHMM
14	FSTWRPNT	DS	1H		Write pointer – item number
16	FSTRDPNT	DS	1H		Read pointer – item 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	A*1	Record format (F or V)
1F	FSTFLAGS	DS	1X	A*2	FST flag byte

Bits defined in FSTFLAGS

FSTRWDSK	EQU	X'80'	Read/write disk
FSTRODSK	EQU	X'00'	Read-only disk
FSTXRDSK	EQU	X'40'	Extension of R-O disk
FSTXWDSK	EQU	X'C0'	Extension of R/W disk
FSTACTRD	EQU	X'04'	Active for reading
FSTACTWR	EQU	X'02'	Active for writing
FSTACTPT	EQU	X'01'	Active from a point
FSTFILEA	EQU	X'07'	File is active

Hexadecimal Displacement	Field Name			Field Description, Contents, Meaning
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: 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 FST.
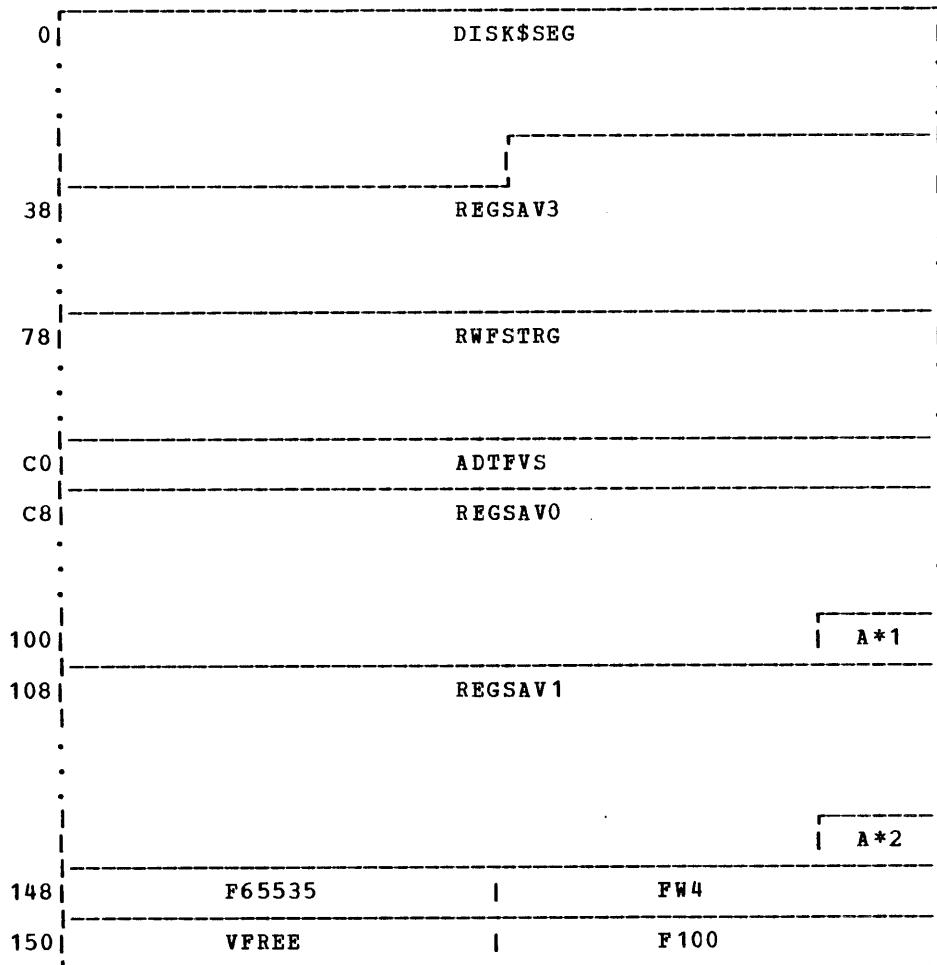
The file status tables for all files on the disk are grouped into 800-byte disk records referred to as file status table blocks (FSTBs). Each file status table block can accommodate up to 20 file status tables.

```
     r---------------------------------------------------------------------1
  0  |                             FSTN                                     |
     |---------------------------------------------------------------------|
  8  |                             FSTT                                     |
     |---------------------------------------------------------------------|
 10  |          FSTD               |    FSTWP      |      FSTRP             |
     |---------------------------------------------------------------------|
 18  |   FSTM        |   FSTIC     |    FSTCL      | A*1  |   A*2           |
     |---------------------------------------------------------------------|
 20  |          FSTIL              |    FSTDBC     |     FSTYR             |
     L---------------------------------------------------------------------J
```

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 no.)
16	FSTRP	DS	1H		Read pointer (item no.)
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 FSTCB (Applicable only to STATEFST of FST-entry after successful STATE or STATEW call)

	FSTFRO	EQU	X'00'	Read-only disk
	FSTFROX	EQU	X'40'	Read-only extension of read-only disk
	FSTFRW	EQU	X'80'	Read/write disk
	FSTFRWX	EQU	X'C0'	Read-only extension of read/write disk
	FSTFACT	EQU	X'07'	File "active" is 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 "point"
20	FSTIL	DS	1F	Maximum item length
24	FSTDBC	DS	1H	800-byte data block count
26	FSTYR	DS	1H	Year
	FSTL	EQU	*-FSTSECT	

FST Hyperblock Parameters

FSTFWDP	EQU	80	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.

```
    ┌─────────────────────────────────────────────────────────────┐
  0 │                          DISK$SEG                            │
    .                                                              .
    .                          ┌───────────────────────────────────┤
    │               ┌──────────┘                                   │
    │               │                                              │
    ├───────────────┘                                              │
 38 │                          REGSAV3                             │
    .                                                              .
    .                                                              .
    ├──────────────────────────────────────────────────────────────┤
 78 │                          RWFSTRG                             │
    .                                                              .
    .                                                              .
    ├──────────────────────────────────────────────────────────────┤
 C0 │                          ADTFVS                             │
    ├──────────────────────────────────────────────────────────────┤
 C8 │                          REGSAV0                            │
    .                                                    ┌─────────┤
    │                                                    │  A*1    │
100 │                                                    └─────────┤
    ├──────────────────────────────────────────────────────────────┤
108 │                          REGSAV1                            │
    │                                                              │
    .                                                              .
    .                                                              .
    │                                                   ┌──────────┤
    │                                                   │  A*2     │
    ├─────────────────────────────┬─────────────────────┴──────────┤
148 │          F65535             │          FW4                   │
    ├─────────────────────────────┼────────────────────────────────┤
150 │          VFREE              │          F100                  │
    └─────────────────────────────┴────────────────────────────────┘
```

```
     ┌──────────────────────────────────────────────────────────────┐
158 │          VFRET            │            JSR0                     │
     ├──────────────────────────────────────────────────────────────┤
160 │          JSR1             │            RWMFD                    │
     ├──────────────────────────────────────────────────────────────┤
168 │          F800             │                                    │
     ├──────────────────────────────────────────────────────────────┤
170 │          FVSDSKA          │            DSKLOC                   │
     ├──────────────────────────────────────────────────────────────┤
178 │          RWCNT            │            DSKADR                   │
     ├──────────────────────────────────────────────────────────────┤
180 │          ADTADD           │                                    │
     ├──────────────────────────┘                                    │
188 │                      FINISLST (cont.)                          │
     │  .                                                          . │
     │  .                                                          . │
     │  .                                                          . │
     │                                          ┌────────────────────┤
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          │  FVFSTDB   │   FVSFSTYR           │
     ├──────────────────────────────────────────────────────────────┤
1E8 │         FVSFSTAD          │            FVSFSTAC                │
     └──────────────────────────────────────────────────────────────┘
```

Hexadecimal Displacement	Field Name			Field Description, Contents, Meaning
0	DISK$SEG	DS	15F	For FSTLKP, FSTLKW, ACTLKP, TRKLKP, 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

Save Area for Lowest-Level Routines
e.g. READMFD, RELUFD, UPDISK, TYPSRCH, ADTLKW

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 goes here
	TRKLSAVE	EQU	REGSAV0	For TRKLKP/X only when called by QQTRK/X

Save Area for Next-to-Lowest Level Routines
e.g. READFST, ERASE, ALTER, INTSVC-LOADMOD

108	REGSAV1	DS	15F	Register save area
144		DC	AL3(00)	
147	ERRCOD1	DC	AL1(*-*) A*2	Error code
148	F65535	DC	F'65535'	= X'0000FFFF'
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
15C	JSR0	DC	F'0'	R0 and
160	JSR1	DC	F'0'	R1 saved here for FRET calls.

PLIST to Read/Write MFD

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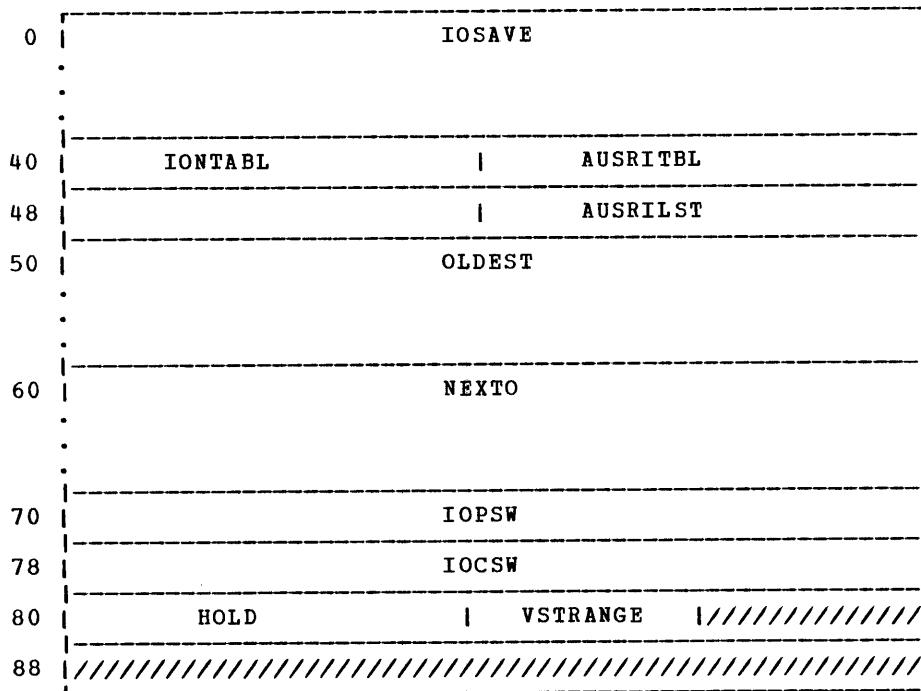
Hexadecimal Displacement	Field Name			Field Description, Contents, Meaning

SIGNAL = Scratch halfword used by READMFD or ERASE

1A4	SIGNAL	DC	H'0'	= 0000, X'FFFF', X'FFFE', or X'FFFD'
	SWTCH	EQU	SIGNAL+1	00, FF, FE, OR FD
1A6	UFDBUSY	DC	X'00'	A*3 Nonzero means UFD is being updated

Bits defined in UFDBUSY

WRBIT	EQU	X'80'	WRBUF
UPBIT	EQU	X'40'	UPDISK - READMFD
FNBIT	EQU	X'20'	FINIS
ERBIT	EQU	X'10'	ERASE - ALTER - READFST
DIOBIT	EQU	X'08'	RDTK/WRTK

The following bits are for routines which do not update the disk, but which cannot be interrupted by an HX command

ABNBIT	EQU	X'02'	DMSABN (ABEND recovery routine)
ITSBIT	EQU	X'01'	DMSITS (SVC handling routine)

1A7	KXFLAG	DC	X'00'	A*4 'HX' flags
	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	FLGSAVE	DC	X'00'	A*5 For scratch use (e.g. by RELUFD)
1AA	FVSFLAG	DC	X'00'	A*6 For general use (as needed)

Miscellaneous Storage Used by ERASE (or RENAME)

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 ACTLKP 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

File Status Table (FST) Copy from STATE

1C0	STATEFST	DS	0D	Full FST of STATED file
1C0	FVSFSTN	DC	D'0'	Filename
1C8	FVSFSTT	DC	D'0'	Filetype
1D0	FVSFSTDT	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(ADT for this file)
	STATER0	EQU	FVSFSTAD	
1EC	FVSFSTAC	DC	A(0)	A(real FST entry for this file)
	STATER1	EQU	FVSFSTAC	

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.
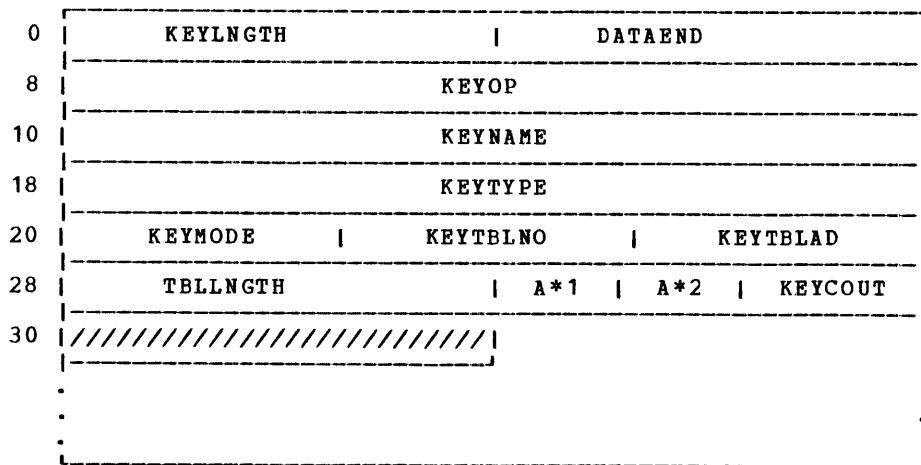
IOSECT is pointed to by the AIOSECT field (HEX 658) in NUCON.

```
     r---------------------------------------------------------------------¬
   0 |                            IOSAVE                                    |
     .                                                                      .
     .                                                                      .
     .                                                                      .
     |------------------------------------------------------------------|
  40 |          IONTABL                    |          AUSRITBL           |
     |------------------------------------------------------------------|
  48 |                                     |          AUSRILST           |
     |------------------------------------------------------------------|
  50 |                            OLDEST                                   |
     .                                                                      .
     .                                                                      .
     .                                                                      .
     |------------------------------------------------------------------|
  60 |                            NEXTO                                    |
     .                                                                      .
     .                                                                      .
     |------------------------------------------------------------------|
  70 |                            IOPSW                                    |
     |------------------------------------------------------------------|
  78 |                            IOCSW                                    |
     |------------------------------------------------------------------|
  80 |          HOLD           |     VSTRANGE   |//////////////|
     |------------------------------------------------------------------|
  88 |////////////////////////////////////////////////////////////////////|
     L---------------------------------------------------------------------
```

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 (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
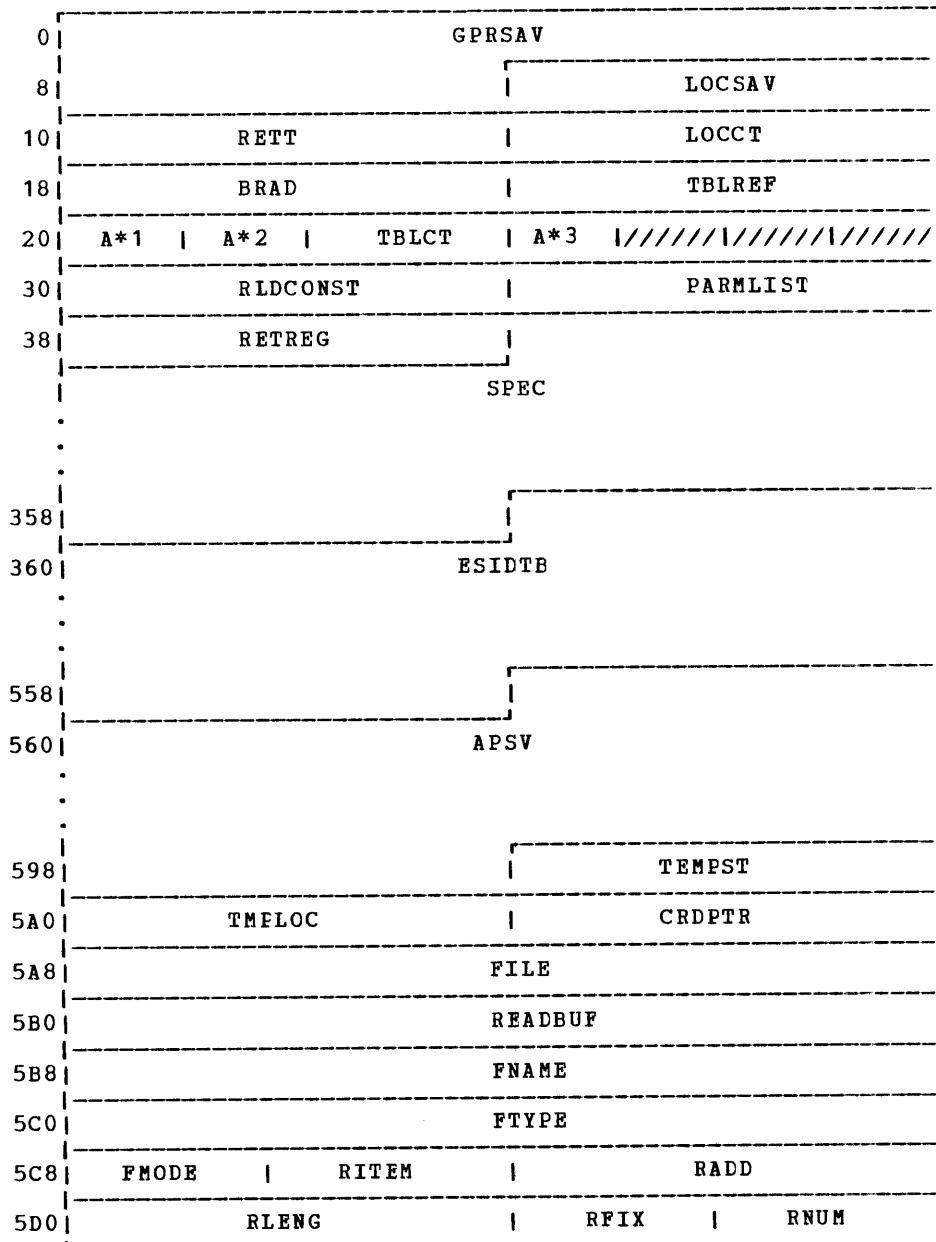
KEYSECT defines the key table used in OS simulation for I/O by key.

```
      r----------------------------------------------------------------------------------¬
  0 |          KEYLNGTH                    |          DATAEND                |
    |--------------------------------------------------------------------------------|
  8 |                              KEYOP                                     |
    |--------------------------------------------------------------------------------|
 10 |                              KEYNAME                                   |
    |--------------------------------------------------------------------------------|
 18 |                              KEYTYPE                                   |
    |--------------------------------------------------------------------------------|
 20 |     KEYMODE       |       KEYTBLNO       |        KEYTBLAD          |
    |--------------------------------------------------------------------------------|
 28 |       TBLLNGTH                  | A*1 |  A*2  |  KEYCOUT  |
    |-------------------------------------------------------------------|
 30 |///////////////////////////////|                          |
    |----------------------------------------------------------¦         |
                                    .                          .
                                    .                          .
                                    .                          .
      L_____J
```
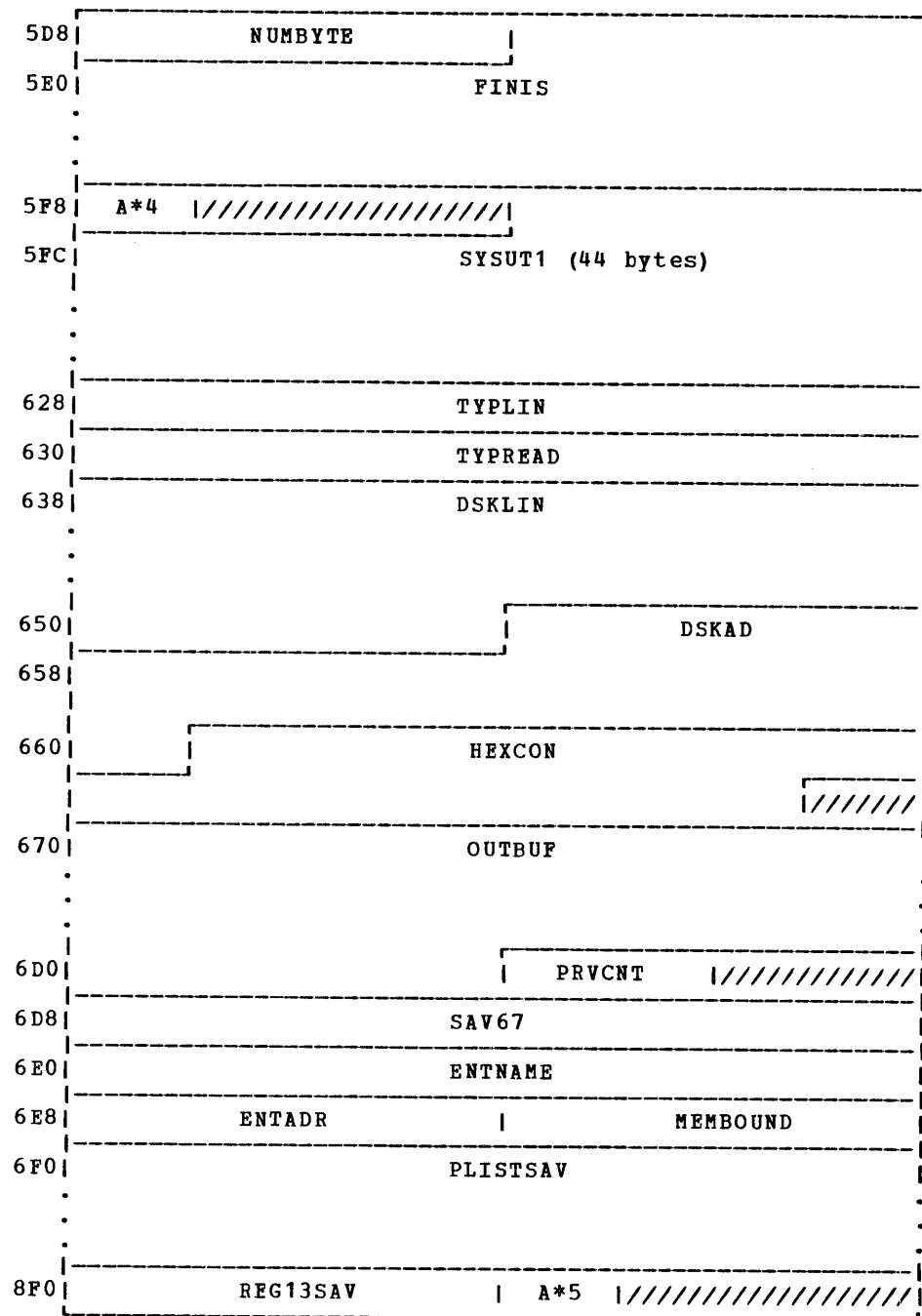
Hexadecimal Displacement	Field Name			Field Description, Contents, Meaning
		KEYSECT PUSH	PRINT	
	KEYSECT	DSECT		
0	KEYLNGTH	DS	1F		Key length
4	DATAEND	DS	1F		Item pointer to last data item in file
8	KEYOP	DS	2F		Start of PLIST for keys file
10	KEYNAME	DS	2F		Filename of keys file
18	KEYTYPE	DS	2F		Filetype of keys file
20	KEYMODE	DS	1H		Filemode of keys file
22	KEYTBLNO	DS	1H		Item no. of key table
24	KEYTBLAD	DS	1F		Address of key table
28	TBLLNGTH	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		No. of bytes read
34	KEYTABLE	DS	0F		Start of keys table (item number)

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.

```
      ┌──────────────────────────────────────────────────────────┐
   0 |                          GPRSAV                            |
     |                           ┌───────────────────────────────┤
   8 |                           |            LOCSAV              |
     |───────────────────────────┼───────────────────────────────┤
  10 |           RETT            |            LOCCT               |
     |───────────────────────────┼───────────────────────────────┤
  18 |           BRAD            |            TBLREF              |
     |───────────┬───────┬───────┼───────┬───────────────────────┤
  20 |  A*1  |  A*2  |    TBLCT    | A*3  |///////|///////|////////|
     |───────────┴───────┴─────────┴──────┴───────────────────────┤
  30 |         RLDCONST          |           PARMLIST             |
     |───────────────────────────┼───────────────────────────────┤
  38 |           RETREG          |                               |
     |───────────────────────────┘                               |
     |                          SPEC                             |
     .                                                           .
     .                                                           .
     .                           ┌───────────────────────────────┤
 358 |                           |                               |
     |───────────────────────────┘                               |
 360 |                          ESIDTB                           |
     .                                                           .
     .                           ┌───────────────────────────────┤
 558 |                           |                               |
     |───────────────────────────┘                               |
 560 |                          APSV                             |
     .                                                           .
     .                           ┌───────────────────────────────┤
 598 |                           |            TEMPST             |
     |───────────────────────────┼───────────────────────────────┤
 5A0 |          TMPLOC          |            CRDPTR             |
     |───────────────────────────┴───────────────────────────────┤
 5A8 |                          FILE                             |
     |────────────────────────────────────────────────────────────┤
 5B0 |                          READBUF                          |
     |────────────────────────────────────────────────────────────┤
 5B8 |                          FNAME                            |
     |────────────────────────────────────────────────────────────┤
 5C0 |                          FTYPE                            |
     |───────┬───────────┬─────────┬──────────────────────────────┤
 5C8 | FMODE |   RITEM   |              RADD                      |
     |───────────────────┴─────────┬───────┬──────────────────────┤
 5D0 |          RLENG             |  RFIX |      RNUM             |
      └──────────────────────────────────────────────────────────┘
```

```
5D8|  r------------------------------------------------------------¬
     |  |              N UMBYTE                    |                |
     |  |------------------------------------------
5E0|  |                      FINIS                                 |
     |  .                                                          .
     |  .                                                          .
     |  .                                                          .
     |  |------------------------------------------------------------
5F8|  |  A*4   |////////////////////////|                         |
     |  |------------------------------------------
5FC|  |                   SYSUT1 (44 bytes)                       |
     |  |                                                          |
     |  .                                                          .
     |  .                                                          .
     |  .                                                          .
     |  |---------------------------------------------------------|
628|  |                       TYPLIN                              |
     |  |---------------------------------------------------------|
630|  |                       TYPREAD                             |
     |  |---------------------------------------------------------|
638|  |                       DSKLIN                              |
     |  .                                                          .
     |  .                                                          .
     |  .                                          r---------------
650|  |                                            |    DSKAD      |
     |  |------------------------------------------              |
658|  |                                                          |
     |  |                                                          |
     |  .          r-----------------------------------------------
660|  |          |           HEXCON                              |
     |  |----------                            r-----------------|
     |  |                                      |  |///////|       |
     |  |----------------------------------------              |
670|  |                       OUTBUF                            |
     |  .                                                          .
     |  .                                                          .
     |  .                                          r---------------
6D0|  |                                            | PRVCNT   |/////////////|
     |  |------------------------------------------              |
6D8|  |                       SAV67                             |
     |  |---------------------------------------------------------|
6E0|  |                       ENTNAME                           |
     |  |---------------------------------------------------------|
6E8|  |          ENTADR          |          MEMBOUND            |
     |  |---------------------------------------------------------|
6F0|  |                       PLISTSAV                          |
     |  .                                                          .
     |  .                                                          .
     |  |---------------------------------------------------------|
8F0|  |         REG13SAV         | A*5 |/////////////////////////|
     |  L-----------------------------------------------------------
```

Hexadecimal Displacement	Field Name				Field Description, Contents, Meaning
0	GPRSAV	DS	3F		Registers 9-12
C	LOCSAV	DS	F		Base register contains A(DMSLDRA)
10	RETT	DS	F		Return register for DMSLSB
14	LOCCT	DS	F		(LOCCNT) next load location
18	BRAD	DS	F		(STRTADDR) start execution address
1C	TBLREF	DS	F		(ALDRTBL) top of loader table
20	FLAG1	DS	X	A*1	Loader switches (permanent)
	Bits defined in FLAG1				
	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'		Don't erase the load map
	WORKFILE	EQU	X'02'		Work file (SYSUT1) exists
	NODUP	EQU	X'01'		Do not type message DMSLIO202W
21	FLAG2	DS	X	A*2	Loader switches (permanent)
	Bits defined in FLAG2				
	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
	CMD	EQU	X'80'		Processing names from cmd 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

Hexadecimal Displacement	Field Name				Field Description, Contents, Meaning
5D6	RNUM	DS	H		Number of items
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 dynaload 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 PLIST
628	TYPLIN	DS	2F		TYPLIN PLIST
630	TYPEAD	DS	2F		TYPLIN buffer address
638	DSKLIN	DS	7F		Disk PLIST 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 of regs 6 and 7
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 ID
8F8	ENDFREE	DS	0D		
	NEED	EQU	(ENDFREE-LDRST)/8		

The following equates refer to displacements and flags
in the REFTABLE entry usually pointed to by register 12

REFNAME	EQU	0		Offset of 8-byte name field
REFLG1	EQU	8		Offset 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'		LIBE card - nonobligatory
REFLIB	EQU	X'10'		Single bit for nonobligatory LIBE card
REFINFO	EQU	9		Offset of relocation factor or max address
REFVAL	EQU	13		Offset of absolute or assigned value
REFLG2	EQU	16		Offset of flag byte 2
REFCMD	EQU	X'80'		Command line name - must resolve

LUBTAB AND LUBPR: LOGICAL UNIT BLOCK TABLE

LUBTAB is a device table that has a two-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 PRTAB. LUBTAB is pointed to by the field LUBPT in BGCOM. The address of the first LUB entry is in the first byte of the FICL control block.

```
     r--------------------------------------------------------------1
0  |  LUBRDR      |  LUBIPT      |  LUBPCH      |  LUBLST      |
     |--------------------------------------------------------------|
8  |  LUBLOG      |  LUBLNK      |  LUBRES      |  LUBSLB      |
     |--------------------------------------------------------------|
10 |  LUBRLB      |  LUBUSE      |  LUBREC      |  LUBCLB      |
     |--------------------------------------------------------------|
18 |  LUBVIS      |  LUBCAT      |
     L----------------------------------J


     r--------------------------------------------------------------1
0  |  LUB000      |  LUB001      |  LUB002      |  LUB003      |
     |--------------------------------------------------------------|
   .                                                              .
   .              LUB004 through LUB239                           .
   .                                                              .
     |--------------------------------------------------------------J
1E0|  LUB240      |  LUB241      |
     L----------------------------------J
```
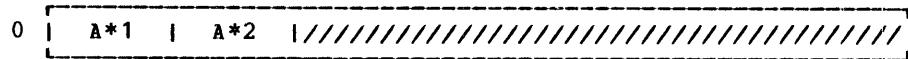
Hexadecimal Displacement	Field Name			Field Description, Contents, Meaning
	System LUBs			
0	LUBRDR	DS	XL2	System virtual reader
2	LUBIPT	DS	XL2	System virtual input device
4	LUBPCH	DS	XL2	System virtual punch
6	LUBLST	DS	XL2	System virtual printer
8	LUBLOG	DS	XL2	Terminal
A	LUBLNK	DS	XL2	Reserved for 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
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
		.		
		.		LUB004 through LUB239 are defined as XL2
		.		
1E0	LUB240	DS	XL2	Programmer logical unit block
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: 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 (hex 4A) in the BGCOM block points to the NICL block.

```
   ┌─────────────────────────────────────────────────────┐
 0 │  A*1  │  A*2  │///////////////////////////////////////│
   └─────────────────────────────────────────────────────┘
```

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

NUCON: NUCLEUS CONSTANT AREA

The nucleus constant area (NUCON) represents the nucleus of CMS.

```
     r-----------------------------------------------------------------1
   0 |                         IPLPSW                                   |
     |-----------------------------------------------------------------|
   8 |                         IPLCCW1                                  |
     |-----------------------------------------------------------------|
  10 |                         IPLCCW2                                  |
     |-----------------------------------------------------------------|
  18 |                         EXTOPSW                                  |
     |-----------------------------------------------------------------|
  20 |                         SVCOPSW                                  |
     |-----------------------------------------------------------------|
  28 |                         PGMOPSW                                  |
     |-----------------------------------------------------------------|
  30 |                         MCKOPSW                                  |
     |-----------------------------------------------------------------|
  38 |                         IOOPSW                                   |
     |-----------------------------------------------------------------|
  40 |                         CSW                                      |
     |-----------------------------------------------------------------|
  48 |        CAWNPSW          |            NUCRSV1                     |
     |-----------------------------------------------------------------|
  50 |        TIMER            |            NUCRSV2                     |
     |-----------------------------------------------------------------|
  58 |                         EXTNPSW                                  |
     |-----------------------------------------------------------------|
  60 |                         SVCNPSW                                  |
     |-----------------------------------------------------------------|
  68 |                         PGMNPSW                                  |
     |-----------------------------------------------------------------|
  70 |                         MCKNPSW                                  |
     |-----------------------------------------------------------------|
  78 |                         IONPSW                                   |
     |-----------------------------------------------------------------|
  80 |                   CPULOG (NUCRSV3)                               |
     .                                                                 .
     .                                                                 .
     .                                                                 .
     L_____J
```

```
       .--------------------------------------------------------.
   90| |            NUCRSV4           |   MONCLASS   |  PERCODE  |
      | |------------------------------------------------------|
   98| |            PERADDR           |            MONCODE       |
      | |------------------------------------------------------|
   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| |                          PREVCMD                        |
      | |------------------------------------------------------|
  2B0| |                          LASTEXEC                       |
      | |------------------------------------------------------|
  2B8| |                          PREVEXEC                       |
       '--------------------------------------------------------'
```

```
        r---------------------------------------------------------1
2C0|                          LASTLMOD                            |
        |---------------------------------------------------------|
2C8|                          LASTTMOD                            |
        |---------------------------------------------------------|
2D0|                          DATIPCMS                            |
        |---------------------------------------------------------|
2D8|                          CLKVALMD                            |
        |---------------------------------------------------------|
2E0|                          MACDIRC                             |
        .                                                         .
        .                                                         .
        .                                                         .
        |---------------------------------------------------------|
300|                          MACLIBL                             |
        .                                                         .
        .                                                         .
        .                                         r---------------.
350|                                              |      TXTDIRC  |
        |----------------------------------------------------------|
358|                          TXTLIBS                             |
        .                                                         .
        .                                                         .
        .                                                         .
        |---------------------------------------------------------|
3A0|         GRS015            |            LOC076                |
        |---------------------------------------------------------|
3A8|         FIRSTDMP          |            LASTDMP               |
        |---------------------------------------------------------|
3B0|         FRS06             |            DMPTIT                |
        |---------------------------------------------------------|
3B8|                            |                                 |
        |----------------------------|                            |
        |                          DMPTITLE                       |
        .                                                         .
        .                                                         .
        |---------------------------------------------------------|
440|         GLBLTABL          |////////////|      SVC$202       |
        |---------------------------------------------------------|
448|                          ERR$202                            |
        |---------------------------------------------------------|
450| A*1  |  A*2  |////////////|          ABATPROC               |
        |---------------------------------------------------------|
458|     ABATABND                        |      ABATLIMT         |
        |---------------------------------------------------------|
460|                                                             |
        |---------------------------------------------------------|
468|                                                             |
        |---------------------------------------------------------|
        |                          DOSDIRC                        |
        .                                                         .
        .                                                         .
        .                                                         .
        |---------------------------------------------------------|
490|                          DOSLIBL                            |
        .                                                         .
        .                                                         .
        .                                                         .
        L---------------------------------------------------------J
```

4D8	A*3	A*4	///////////	ALTASAVE	
4E0	ABGCOM			ASYSCOM	
4E8	ADOSDCSS			SVC12SAV	
4F0	DOSFIRST		DOSNUM	///////////	
4F8	APPSAVE			DOSTRANS	
500	MAINLIST			MAINSTRT	
508	FREELIST			FREENUM	
510	MAINHIGH			FREELOWE	
518	FREELOWR			FREEUPPR	
520	ANUCEND			AUSRAREA	
528	CURRSAVE		CODE203	PCTCMSFS	
530	ADMSFRT			VCADTLKP	
538	VCADTNXT			VCADTLKW	
540	CURRIOOP			PENDREAD	
548	PENDWRIT			FSTFINRD	
550	LSTFINRD			AINTRTBL	
558	AOUTRTBL		NUMFINRD	NUMPNDWR	
560	VMSIZE			ALDRTBLS	
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				ALIASENT	
5B0	DYNAEND		///////////////////////		
5B8	/////////////////////////	///////////////////////////			
5C0	FCBFIRST		FCBNUM	A*8	
5C8	/////////////////////////		LINKLAST		

Offset							
5D0	LINKSTRT					TAXEADDR	
5D8	ATSOCPPL					DCBSAV	
5E0	A*9	A*10	A*11	A*12	///////////	A*13	A*14
5E8	A*15	A*16	A*17	/////////		ASYSNAMS	
5F0	ACMSSEG					ADMSLIO	
5F8	VCFSTLKP					VCFSTLKW	
600	AFVS					AOPSECT	
608	ADEVTAB					AFSTLKP	
610	AGETCLK					AFSTLKW	
618	APIE					AIADT	
620	AUSER					ARDTK	
628	ASCANN					ASSTAT	
630	ATABEND					ASUBSECT	
638	AOSMODL					AWRTK	
640	ASTRINIT					IADT	
648	AFREE					AFRET	
650	ADMSPIOC					APGMSECT	
658	AIOSECT					ADMPEXEC	
660	ADIOSECT					AABNSVC	
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				LDMSROS		CDMSROS
6B0	AACTLKP					AACTNXT	
6B8	AACTFREE					AACTFRET	
6C0	AADTNXT					ATRKLKP	

Section 2. CMS Data Areas and Control Blocks 197

6C8	ATRKLKPX	AQQTRK
6D0	AQQTRKX	AERASE
6D8	ATYPSRCH	AUPDISK
6E0	AKILLEX	ATFINIS
6E8	ARDBUF	AWRBUF
6F0	AFINIS	ASTATE
6F8	ASTATEW	APOINT
7A0	CONCCWS	
708		
710	CONINBLK	
718	CONINBUF	
	.	.
	.	.
	.	.
7A0	CMNDLINE	
	.	.
	.	.
	.	.
848	CMNDLIST	
	.	.
	.	.
	.	.
A60	CONSTACK	
	.	.
	.	.
	.	.
BE0	BALRSAVE	
	.	.
	.	.
	.	.
C20	WAITSAVE	
	.	.
	.	.
	.	.

Hexadecimal Displacement	Field Name			Field Description, Contents, Meaning
	Machine Usage			
000	IPLPSW	DS	1D	Initial program load PSW
008	IPLCCW1	DS	1D	Initial program load CCW1
010	IPLCCW2	DS	1D	Initial program load CCW2
		ORG	IPLPSW	
000	RSTNPSW	DS	1D	PSW restart new PSW
008	RSTOPSW	DS	1D	PSW restart old PSW
010	ACMSCVT	DS	1F	Address of simulated OS CVT
014	ASYSREF	DS	1F	Address of nucleus address table
018	EXTOPSW	DS	1D	External old PSW
020	SVCOPSW	DS	1D	Supervisor call old PSW
028	PGMOPSW	DS	1D	Program old PSW
030	MCKOPSW	DS	1D	Machine-check old PSW
038	IOOPSW	DS	1D	Input/output old PSW
040	CSW	DS	1D	Channel status word
048	CAW	DS	1F	Channel address word
04C	NUCRSV1	DS	1F	Reserved for IBM use
050	TIMER	DS	1F	Interval timer
054	NUCRSV2	DS	1F	Reserved for IBM use
058	EXTNPSW	DS	1D	External new PSW
060	SVCNPSW	DS	1D	Supervisor call new PSW
068	PGMNPSW	DS	1D	Program new PSW
070	MCKNPSW	DS	1D	Machine-check new PSW
078	IONPSW	DS	1D	Input/output new PSW
080	CPULOG	DS	48D	CPU logout area
		ORG	CPULOG	
080	NUCRSV3	DS	2D	Reserved for IBM use
090	NUCRSV4	DS	1F	Reserved for IBM use
094	MONCLASS	DS	1H	Monitor call class number
096	PERCODE	DS	1H	Program event recorder code
098	PERADDR	DS	1F	Program event recorder address
09C	MONCODE	DS	1F	Monitor call code
0A0	NUCRSV5	DS	4D	Reserved for IBM use
0C0	LOWSAVE	DS	XL160	Save area for 1st 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
	System Usage			
200	SYSTEMID	DS	CL32	System name and date
220	INSTALID	DS	CL64	Installation identification
260	SYSNAME	DS	CL8	Name of IPLed saved system
268	IPLADDR	DS	1H	Address of IPLed device
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 CPU time used
298	LASTVIRT	DS	1F	Previous elapsed virtual time used
29C	LASTCPUT	DS	1F	Previous elapsed CPU time used
2A0	LASTCMND	DC	CL8' '	Last command issued
2A8	PREVCMND	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		DS	3F	Reserved for IBM use
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(LOWSAVE)	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 code
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 BATRUN				
BATRUN	EQU	X'80'	Batch monitor running
BATLOAD	EQU	X'40'	Loading batch processor
BATNCEX	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'	CPU time exceeded
BATXPRT	EQU	X'20'	No. printed lines exceeded
BATXPUN	EQU	X'10'	No. punched cards exceeded
BATDCMS	EQU	X'08'	Disabled CMS command called
BATIPLSS	EQU	X'04'	Batch IPLing saved system
BATSTOP	EQU	X'02'	Batch stopping after current job

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		DC	4F'0'	Reserved for IBM use

Hexadecimal Displacement	Field Name			Field Description, Contents, Meaning
	DOS Library Pointers			
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
	Bits defined in DOSFLAGS			
	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 install 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 DOS 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		DS	H	Reserved for IBM use
4F8	APPSAVE	DC	V(PPSAVE)	Address of problem program save area
4FC	DOSTRANS DC		A(0)	Address of DOS transient area
	Free Storage Pointers			
500	MAINLIST DC		A(0)	Address of 1st 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 1st 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	FREELOWR DC		V(TRANSAR)	Lower limit of system free storage
51C	FREEUPPR 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	PCTCMSFS DS		1H'04'	% of available user storage to reserve
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
	Console I/O Pointers			
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	FSTFINRD 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

Hexadecimal Displacement	Field Name			Field Description, Contents, Meaning
	Loader Information			
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
	Bits defined in MODFLGS			
	NOMAPFLG	EQU	X'80'	NOMAP flag
	CLEAROP	EQU	X'40'	CLEAR option flag
	MOD2	EQU	X'20'	Module generated with DOS option
	MOD3	EQU	X'10'	Module generated with ALL option
	MOD4	EQU	X'08'	Reserved for IBM use
	MOD5	EQU	X'04'	Reserved for IBM use
	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'	A*7	DMSLSY unique symbol base
5A4	NXTSYM	DC	C'Z'	1st char of unique symbol
5A5		DC	XL7'0'	Rest of unique symbol
5AC	ALIASENT	DC	1F'0'	Alias entry point (dynamic load)
5B0	DYNAEND	DC	1F'0'	Max. load loc (dynamic load)
5B4		DS	3F	Reserved for IBM
	OS Simulation Pointers			
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
5C7	OSSFLAGS	DC	X'00'	A*8	OS simulation flags
	Bits defined in OSSFLAGS			
	COMPSWT	EQU	X'80'	Compiler switch
	OSSMNU	EQU	X'40'	DMSSMN unconditional flag
	OSRESET	EQU	X'20'	
	OSWAIT	EQU	X'10'	
	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
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	ATSOCPPL	DC	V(CPP)	Address of TMP PLIST for TSO programs
5DC	DCBSAV	DC	1F'0'	DCB restoration address

Hexadecimal Displacement	Field Name			Field Description, Contents, Meaning
5E0	OPTFLAGS DC	1X'00'	A*9	Option flags
	Bits defined in OPTFLAGS			
	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
	NOVMREAD EQU	X'04'		No automatic VM console read
5E1	MISFLAGS DC	1X'00'	A*10	Miscellaneous flags
	Bits defined in MISFLAGS			
	KXSWITCH EQU	X'80'		Halt execution switch
	KOSWITCH EQU	X'40'		Halt tracing switch
	RELPAGES EQU	X'20'		Release pages switch
	GRAFDEV EQU	X'10'		Graphics console
	QSWITCH EQU	X'08'		Quiet switch console read
	NODDSK EQU	X'04'		Don't access d disk
	NEGITS EQU	X'02'		Negative return code from DMSITS
	MISFL01 EQU	X'01'		Reserved for IBM use
5E2	MSGFLAGS DC	1X'00'	A*11	Message flags
	Bits defined in MSGFLAGS			
	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
	Bits defined in DBGFLAGS			
	DBGEXEC EQU	X'80'		DEBUG 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
	Bits defined in EXECFLAG			
	EXECRUN EQU	X'80'		EXEC command running
5E7	PROTFLAG DC	1X'00'	A*14	Storage protection flags
	Bits defined by PROTFLAG			
	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
	Bits defined in TSOFLAGS			
	TSOATCNL EQU	X'80'		Read canceled by attention
5E9	SUBFLAG DC	1X'00'	A*16	CMS subset flag byte

Hexadecimal Displacement	Field Name			Field Description, Contents, Meaning
	Bits defined in SUBFLAG			
	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
	Bits defined in DCSSFLAG			
	DCSSAVAL	EQU	X'80'	CMSSEG segment exists
	DCSSLDED	EQU	X'40'	CMSSEG loaded
	DCSSCPNV	EQU	X'20'	CP or invalid command issued
	DCSSLDSD	EQU	X'10'	Loading S-disk
	DCSSVTNA	EQU	X'08'	DMSSVT not available
	DCSSVTLD	EQU	X'04'	DMSSVT is loaded
	DCSSOVLP	EQU	X'02'	VM storage overlaid by DCSS
	DCSSJLNS DC		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)	BALR equivalent of FSTLKP
5FC	VCFSTLKW DC		V(DMSLFSW)	BALR equivalent of FSTLKW

Nucleus Address Table

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(DMSSMNST)

Hexadecimal Displacement	Field Name			Field Description, Contents, Meaning
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) —	I/O save area used by DMSCIT
664	AABNSVC	DC	V(DMSABNUA)	
668	ADMSERL	DC	V(DMSERL)	
66C	ADMSCRD	DC	V(DMSCRD)	
670	ADMSFREB	DC	V(DMSFREB)	
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)	
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(DMSLAFFE)	
6BC	AACTFRET	DC	V(DMSLAFFT)	
6C0	AADTNXT	DC	V(ADTNXT)	
6C4	ATRKLKP	DC	V(DMSTRK)	
6C8	ATRKLKPX	DC	V(DMSTRKX)	
6CC	AQQTRK	DC	V(DMSTQQ)	
6D0	AQQTRKX	DC	V(DMSTQQX)	
6D4	AERASE	DC	V(DMSERS)	
6D8	ATYPSRCH	DC	V(TYPSRCH)	
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(DMSFNS)	
6F4	ASTATE	DC	V(DMSSTTE)	
6F8	ASTATEW	DC	V(DMSSTTW)	
6FC	APOINT	DC	V(POINT)	

Hexadecimal Displacement	Field Name			Field Description, Contents, Meaning
	Terminal Buffers			
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	
	Save Areas			
BA0	FREESAVE	DS	16F	
BE0	BALRSAVE	DS	16F	
C20	WAITSAVE	DS	16F	
	VSAM and AMSERV Control Words			
C60		DS	0D	
	Percent of available user storage to reserve for GETVIS/FREEVIS use when running VSAM			
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
	Beginning and End of IKQLAB (when in storage)			
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)	AAddress of VSAM interface bootstrap
C78	AVIPWORK	DC	A(0)	Address of DMSVIP work area
C7C	VSAMFLG1	DC	X'00'	VSAM information flag
	VSAMRUN	EQU	X'80'	VSAM system loaded
	VSJOBCAT	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'0'	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($$BEOJ4)	DMSVSR entry point from VSAM $$BACLOS
C8C	AVSRWORK	DC	A(0)	Address of DMSVSR work area
C90	ACBLIST	DC	A(0)	ACB list built by OPEN/CLOSE
C94		DS	3F	Reserved for IBM use
CA0		DS	0D	

OPSECT: MAJOR DSECT 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 DSECT is pointed to by the AOPSECT field (hex 604) 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  │     READLIST (cont.)   │            RDBUFF                       │
     ├────────────────────────────────────────────────────────────────┤
 90  │  RDCCW    │  RDCOUNT   │            PUNCHLST                     │
     ├────────────────────────────────────────────────────────────────┤
 98  │    PUNCHLST (cont.)    │            PUNBUFF                      │
     ├────────────────────────────────────────────────────────────────┤
 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.)                │                                         │
      ├─────────────────────────────────────────┴─────────────────────────────────────────┤
      .                                                                                    .
      .                                                                                    .
      .                                                                                    .
      ├─────────────────────────────────────────┬─────────────────────────────────────────┤
  118 │            EXLEVEL                       │            EXF1                          │
      ├─────────────────────────────────────────┼─────────────────────────────────────────┤
  120 │            EXNUM                         │            EXADD                         │
      ├─────────────────────────────────────────┼─────────────────────────────────────────┤
  128 │                                         │                                         │
      ├─────────────────────────────────────────┴──────────┬──────────────┐
  130 │            FCBIO                                    │   A*4        │
      └────────────────────────────────────────────────────┴──────────────┘
```

Hexadecimal Displacement	Field Name		Field Description, Contents, Meaning
	Main I/O Operation List		
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
	Immediate Register Save Area		
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
	Console Parameter Lists		
44		DS 0F	
	Read Console		
44	CONREAD	DC CL8'WAITRD'	Terminal read
4C	CONRDBUF	DC V(CMNCLINE)	Address of input buffer
50	CONRDCOD	DC C'U' A*1	Translate code
		DC X'0'	
52	CONRDCNT	DC AL2(0)	Data byte count
54		DC F'0'	Reserved for IBM use
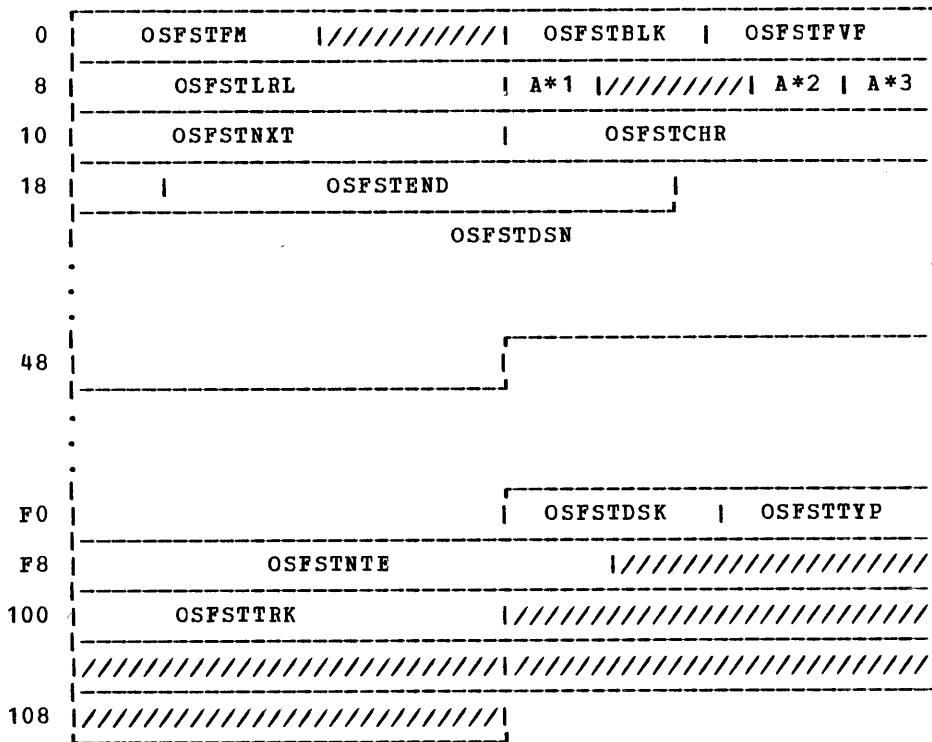
Hexadecimal Displacement	Field Name			Field Description, Contents, Meaning

Console Wait List

| 58 | WAITLIST DS | 0F | | |
| 58 | DC | CL8'CONWAIT' | | |

Write Console

60	CONWRITE DS	0F		
60	DC	CL8'TYPLIN'		
68	CONWRBUF DC	A(0)	A*2	Location of message text
6C	CONWRCOD DC	C'B'		Color code
6D	DC	X'00'		
6E	CONWRCNT DC	AL2(0)		Length of message text

Wait Parameter List

70	WAITLST DS	0F		
70	DC	CL8'WAIT'		Address of DMSCWT
78	WAITDEV DC	CL4'CON1'		Symbolic address of console
7C	DC	F'0'		
80	DC	F'0'		

Reader Parameter List

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'		Bytes actually read

Card Punch Parameter List

94	PUNCHLST DS	0F		
94	DC	CL8'CARDPH'		
9C	PUNBUFF DC	A(0)		Punch buffer address
A0	PUNCOUNT DC	A(0)		Punch CCW count

Printer Parameter List

A4	PRINTLST DS	0F		
A4	DC	CL8'PRINTR'		
AC	PRBUF DC	A(0)		Printer buffer address
B0	PRCNT DC	A(0)		Printer data count

Tape Parameter List

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

Hexadecimal Displacement	Field Name			Field Description, Contents, Meaning

Close Out Device Dependent Data Set On Unit Record Equipment

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

Storage for EXEC Bootstrap

118	EXLEVEL	DC	F'0'	EXEC level
11C	EXF1	DC	F'1'	Follows EXEC level
120	EXNUM	DC	F'0'	Number doublewords free storage
124	EXADD	DC	F'0'	Address of DMSEXT in storage
128		DC	2F'0' —	Reserved for IBM use

Storage for OS Macro Simulation Routines

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: OS FILE STATUS TABLE

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

```
        r---------------------------------------------------------1
   0    |     OSFSTFM     |///////////| OSFSTBLK  |  OSFSTFVF     |
        |---------------------------------------------------------|
   8    |     OSFSTLRL            | A*1 |/////////| A*2 | A*3      |
        |---------------------------------------------------------|
  10    |     OSFSTNXT           |        OSFSTCHR                |
        |---------------------------------------------------------|
  18    |     |        OSFSTEND               |                   |
        |-----------------------------------------|               |
        |                    OSFSTDSN                             |
        .                                                         .
        .                                                         .
        .                                                         .
        |                         r-------------------------------|
  48    |                         |                               |
        |-------------------------|                               |
        .                                                         .
        .                                                         .
        .                                                         .
        |                         r-------------------------------|
  F0    |                         |  OSFSTDSK   |  OSFSTTYP       |
        |---------------------------------------------------------|
  F8    |       OSFSTNTE          |/////////////////////////////  |
        |---------------------------------------------------------|
 100    |  OSFSTTRK         |///////////////////////////////////  |
        |---------------------------------------------------------|
        |////////////////////////////////\//////////////////////  |
        |---------------------------------------------------------|
 108    |/////////////////////////////////|
        L---------------------------------J
```

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

Bits defined in OSFSTFLG

E	OSFSTFLG	DS	1X	A*2	Flag byte
	OSFSTALT	EQU	X'80'		Alternate track indicator
	OSFSTMVL	EQU	X'08'		Multiple volume data set
	OSFSTDBK	EQU	X'40'		Block size not specified in DSCB
	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		CCHHR 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
	OSFSTEX4	EQU	OSFSTXTN+30	Location of 4th 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 CCHHR for NOTE macro
FD		DS	3X	Reserved for IBM use
100	OSFSTTRK	DS	1F	No. tracks per cylinder
104		DS	4F	Reserved for IBM use
118		DS	0D	
	OSFSTLTH	EQU	(*-OSFST)/8	OS FST length in doublewords

OVSECT: DESCRIBES THE FIRST FEW LOCATIONS OF DMSOVS

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

```
     r-----------------------------------------------------------1
  0  |            OVSGO              |            AERR           |
     |-----------------------------------------------------------|
  8  |            AWAIT             |            LENOVS          |
     L-----------------------------------------------------------J
```

Hexadecimal Displacement	Field Name			Field Description, Contents, Meaning

This Macro Describes the First Few Words of the DMSOVS Module

0	OVSGO	B	*-*	Branch to this point for TRACE
4	AERR	DC	A(0)	Address of DMSERR
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 (BGCOM).

```
     r-----------------------------------------------------------1
  0  |            PCROUT            |            PCSAVE          |
     L-----------------------------------------------------------J
```

| Hexadecimal Displacement	Field Name			Field Description, Contents, Meaning
	PCTAB	EQU	*-8	
0	PCROUT	DC	F'0'	Program check routine
4	PCSAVE	DC	F'0'	Program check save area

PCROUT can contain one of three values: zero (indicating that no STXIT macro was issued), the address of a user program check routine (indicating that th STXIT macro was issued), or the complement of the address of a user program check routine (indicating that the routine specified in the user STXIT macro was already in use).

PCSAVE can contain either zero (indicating that no STXIT macro was issued), or the address of the user save area (indicating that the STXIT macro was used).

PDSSECT: DIRECTORY TABLE FOR BPAM SIMULATION

PDSSECT describes the fields of the in-storage directory that is used in OS simulation of BPAM.
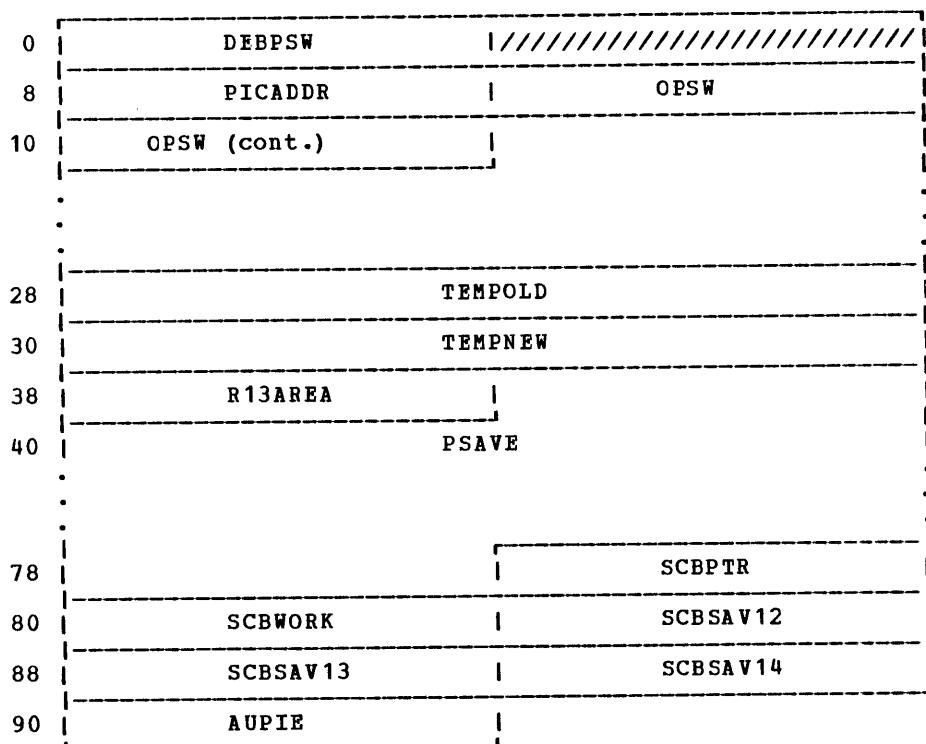
```
   r------------------------------------------------------------1
0 |                 DIRNAME                |     DIRPTR        |
   |------------------------------------------------------------|
8 | A*1 | A*2 |    CORESIZE   | PDSBLKSI | A*3 |///////////|
   L------------------------------------------------------------J
```

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

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 DSECT is pointed to by the APGMSECT field (hex 654) in NUCON.

```
     r----------------------------------------------------------------------¬
  0  |            DEBPSW            |//////////////////////////////////////|
     |----------------------------------------------------------------------|
  8  |           PICADDR           |            OPSW                       |
     |----------------------------------------------------------------------|
 10  |  OPSW (cont.)               |                                       |
     |-----------------------------J                                       |
     .                                                                     .
     .                                                                     .
     |----------------------------------------------------------------------|
 28  |                       TEMPOLD                                        |
     |----------------------------------------------------------------------|
 30  |                       TEMPNEW                                        |
     |----------------------------------------------------------------------|
 38  |       R13AREA               |                                       |
     |-----------------------------J                                       |
 40  |                        PSAVE                                        |
     .                                                                     .
     .                                                                     .
     |                             r----------------------------------------|
 78  |                             |            SCBPTR                      |
     |----------------------------------------------------------------------|
 80  |       SCBWORK               |            SCBSAV12                    |
     |----------------------------------------------------------------------|
 88  |       SCBSAV13              |            SCBSAV14                    |
     |-----------------------------------------------------------------------J
 90  |       AUPIE                 |
     L-----------------------------J
```
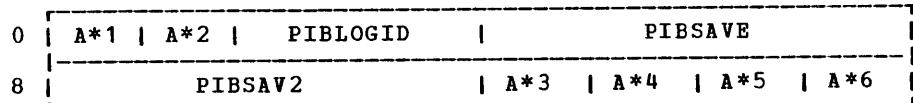
Hexadecimal Displacement	Field Name			Field Description, Contents, Meaning

Storage for Program Interrupt Routine (DMSITP)

0	DEBPSW	DC	F'0',V(DMSDBG)	Point to debug
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'	Regs: R14,R15,R0,R1,R2
				End program interrupt element
28	TEMPOLD	DC	8X'00'	Work area
30	TEMPNEW	DC	8X'00'	
38	R13AREA	DC	F'0'	Saved R13
3C	PSAVE	DC	16F'0'	Regs saved at interrupt time

Hexadecimal Displacement	Field Name			Field Description, Contents, Meaning
7C	SCBPTR	DC	F'0'	Pointer to first STAE control block
	Bits defined in SCBPTR			
	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 reg 12 save area for DMSSAB
88	SCBSAV13	DC	A(0)	Address of reg 13 save area for DMSSAB
8C	SCBSAV14	DC	A(0)	Address of reg 14 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 (hex 5A) in the BGCOM block points to the PIBADR block.

```
   ┌─────────────────────────────────────────────────────────────┐
 0 │ A*1 │ A*2 │    PIBLOGID     │            PIBSAVE             │
   │─────────────────────────────────────────────────────────────│
 8 │            PIBSAV2          │ A*3  │ A*4  │ A*5  │ A*6  │
   └─────────────────────────────────────────────────────────────┘
```
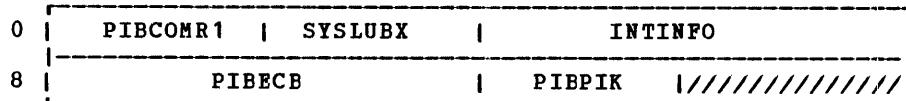
Hexadecimal Displacement	Field Name				Field Description, Contents, Meaning
0	PIBFLG	DS	X	A*1	Flags
1	PIBCNCL	DS	X	A*2	Cancel code
2	PIBLOGID	DS	XL2		SYSLOG ID
4	PIBSAVE	DS	XL4		Address of save area
	ARFLG	EQU	PIBADR+8		Save area address
8	PIBSAV2	DS	XL4		Address of system save area
C	PIBPUBAS	DS	X	A*3	PUB assign flags
D	PIBLUBID	DS	X	A*4	LUB no. of first problem program LUB
E	PIBLUBNO	DS	C	A*5	Number of LUBs
F	PIBFLG2	DS	C	A*6	More flags
		ORG	PIBADR		PIB extension DSECT
0	PIBCOMRA	DS	XL2		Communication region address
2	SYSLUBX	DS	XL2		System class LUB address
4	INTINFO	DS	XL4		Interrupt information
	SVCIC	EQU	INTINFOX3		
8	PIBECB	DS	XL2		Program interrupt key
A		DS	XL2		Reserved for IBM use

PIB2TAB: PROGRAM INFORMATION BLOCK EXTENSION

PIB2TAB is an extension of the PIBTAB block. For each PIB table entry, an entry exists in the PIB table extension block (PIB2TAB).

The PIB2PTR field (hex 7C) in the BGCOM block points to the PIB2TAB block.

```
     r---------------------------------------------------------------------1
 0   |     PIBCOMR1   |   SYSLUBX      |           INTINFO                  |
     |-------------------------------------------------------------------- |
 8   |             PIBECB               |    PIBPIK    |////////////////////|
     L---------------------------------------------------------------------J
```
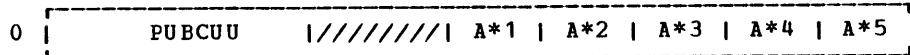
Hexadecimal Displacement	Field Name			Field Description, Contents, Meaning
0	PIBCOMR1	DS	XL2	Address of communications region
2	SYSLUBX	DS	XL2	System LUB index
4	INTINFO	DS	XL4	Used for interruption code
	SVCIC	EQU	INTINFO+3	SVC interrupt code
8	PIBECB	DS	XL4	Address of termination ECB, if any
C	PIBPIK	DS	XL2	Program interrupt key
E		DS	XL2	Reserved for IBM use

PUBADR: PHYSICAL UNIT BLOCK TABLE

PUBADR is a table defining the physical devices being used by CMS/DOS. The simulated PUBADR has eighteen 8-byte entries, one for each device supported by CMS. Also included here is the DSECT used by DOS/VS routines. Both DSECTs define the same storage. The simulated PUBADR is invoked by the MAPPUB macro; the DOS/VS DSECT is invoked by the PUBTAB macro.
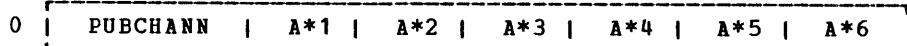
The address of the PUBADR is at displacement X'40' of BGCOM.

```
   r---------------------------------------------------------------------1
0  |        PUBCUU        |////////| A*1 | A*2 | A*3 | A*4 | A*5 |
   L---------------------------------------------------------------------J
```

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

PUBADR defines the fields of a physical unit block table as used in CMS/DOS. PUBADR is invoked by the PUBTAB macro.

```
0  | PUBCHANN | A*1 | A*2 | A*3 | A*4 | A*5 | A*6 |
```

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
	PUBWIT	EQU	*-PUBADR		Length of PUB table
	PUBPTR	EQU	NEXTPUB		Pointer to original PUB

PUBOWNER: PHYSICAL UNIT BLOCK OWNERSHIP TABLE

PUBOWNER is comprised of 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 at displacement X'78' of the system communication table (SYSCOM).

```
     ┌─────────────────────────────────────────────────────────────┐
  0  │                                                               │
     ├─────────────────────────────────────────────────────────────┤
 10  │                                                               │
     └─────────────────────────────────────────────────────────────┘
```

Hexadecimal Displacement	Field Name		Field Description, Contents, Meaning
	PUBOWNER DS	0H	PUB ownership table
	ENTRY PUBOWNER		
	DS	18X'000'	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 allocated in protected free storage. SSAVE is invoked via the CMSAVE macro.

Offset				
0	A*1 | A*2 | CODE		CALLER	
8	CALLEE			
10	OLDPSW			
18	NRMRET		ERRET	
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	

Hexadecimal Displacement	Field Name				Field Description, Contents, Meaning
0	OVIND	DS	X	A*1	Override indicator: 0,1,2,3
	TYPFLAG	DS	BL1	A*2	SVC type byte
	Bits defined in TYPFLAG				
	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
	TPFSV0	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	0F		General-purpose registers at entry
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	0D		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 1
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
90	OSTEMP	DS	F		Temporary work area for OS simulation routines
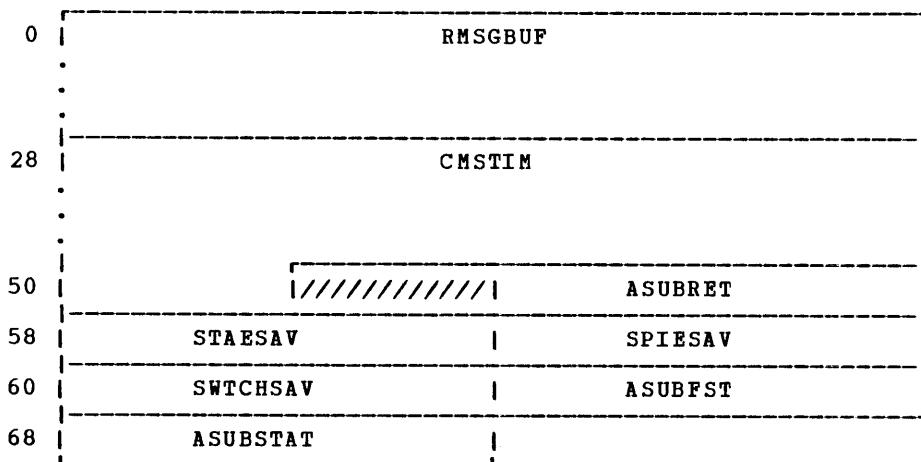
Hexadecimal Displacement	Field Name				Field Description, Contents, Meaning
	DMSKEY Key Stack				
	KEYMAX	EQU	7		Maximum number of keys in stack
94	KEYP	DC	X'00'	A*3	Number of keys on stack
95	KEYS	DS	(KEYMAX)X		Key stack

The following fields are filled in only by DMSOVS, the SVCTRACE subroutine

9C	XGPR0	DS	F	Extra copy of EGPR0
A0	XGPR1	DS	F	Extra copy of EGPR1
A4	XGPR15	DS	F	Extra copy of EGPR15
	XCOUNT	DS	F	Extra copy of SVCOUNT
AC	CHKWRD2	DC	C'EFGH'	Check word 2
	SSAVESZ	EQU	(*-SSAVE+7)/8	Size of system save area

Format of User Save Area

0	USAVE	DSECT		
0		DS	12D	Scratch area passed to user via a pointer in register 13
	USAVESZ	EQU	(*-USAVE+7)/8	Size of user save area

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 DSECT is pointed to by the ASUBSECT field (hex 634) in NUCON.

```
     r-------------------------------------------------------------¬
 0   |                         RMSGBUF                             |
     .                                                             .
     .                                                             .
     .                                                             .
     |-------------------------------------------------------------|
28   |                         CMSTIM                              |
     .                                                             .
     .                                                             .
     |            r------------------------------------------------|
50   |           |///////////|            ASUBRET                  |
     |-----------+-----------------+-------------------------------|
58   |         STAESAV             |            SPIESAV            |
     |-----------------------------+-------------------------------|
60   |         SWTCHSAV            |            ASUBFST             |
     |-----------------------------+-------------------------------J
68   |         ASUBSTAT            |
     L-----------------------------J
```

Hexadecimal Displacement	Field Name			Field Description, Contents, Meaning
	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 CPU 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 DMSLAF (used by SUBSET)
40	STARS	DC	CL8'*'	
48		DC	CL8'*'	
50		DC	CL2'*'	

Subset Address Storage Areas

54		DS	0F	
54	ASUBRET	DS	F	Return address to caller
58	STAESAV	DS	F	STAE 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: SVC INTERRUPT STORAGE

SVCSECT describes the fields used by DMSITS in handling SVC interrupts. SVCSECT is built
dynamically when an SVC is issued. The first SVCSECT is pointed to by the ASVCSECT field
(hex 674) 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                 │      OVBPF       │      OVBTF        │
         ├──────────────┬──────────────┬────────┼──────────────────┴──────────────────┤
  28     │    OVAPF     │    OVATF     │  A*2   │            unused                   │
         ├──────────────┴──────────────┴────────┴─────────────────────────────────────┤
  30     │                            SVCSAVE                                          │
         .                                                                            .
         .                                                                            .
         .                                                                            .
         ├─────────────────────────────────────────────────────────────────────────┤
  78     │                            NRMSAV                                           │
         .                                                                            .
         .                                                                            .
         ├─────────────────────────────────────┬─────────────────────────────────────┤
 158     │              unused                 │              SVCOUNT                │
         ├─────────────────────────────────────┼─────────────────────────────────────┤
 160     │              SVCSTOP                 │              SVLAD                   │
         ├─────────────────────────────────────┼─────────────────────────────────────┤
 168     │              SVLADW                  │              SVLFS                   │
         ├─────────────────────────────────────┴─────────────────────────────────────┤
 170     │                                                                            │
         .                                                                            .
         .                                                                            .
         ├─────────────────────────────────────────────────────────────────────────┤
 1A8     │                            MODLIST                                          │
         ├─────────────────────────────────────────────────────────────────────────┤
 1B0     │                            DUMCOM                                           │
         ├────────┬──────────────────────────────┬─────────────────────────────────────┤
 1B8     │  A*3   │       ZERO3                  │              TRANSRT                │
         ├────────┴──────────────────────────────┼────────┬──────────────────────────────┤
 1C0     │         TRANSRT (cont.)               │  A*4   │           ADTRANS            │
         ├───────────────────────────────────────┴────────┴──────────────────────────────┤
 1C8     │                            TEMP02                                           │
         ├─────────────────────────────────────────────────────────────────────────┤
 1D0     │///////////////////////////////////////////////////////////////////////////│
         ├─────────────────────────────────────────────────────────────────────────┤
 1D8     │///////////////////////////////////////////////////////////////////////////│
         ├─────────────────────────────────────┬─────────────────────────────────────┤
 1E0     │              RGPR0                   │              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	0F	User SVC Table

Keep next four in order

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	JF4	DC	F'4'	Loop increment for BXLE
C	JLAST	DC	A(*-*)	Address of last item in table

Start-up flags — Indicate System Parameter Flag for the Called Routine

10	SFLAG	DC	BL1'0'	A*1	Flag byte

Bits defined in SFLAG

	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
	SFREN	EQU	X'01'	Illegal re-entry 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

Information for SVCTRACE

20	ADMSOVS	DC	A(0)	Address of DMSOVS
24	OVBPF	DC	BL2'0'	'Before print' flags
26	OVBTF	DC	BL2'0'	'Before type' flags
28	OVAPF	DC	BL2'0'	'After print' flags
2A	OVATF	DC	BL2'0'	'After type' flags

The following equate symbols are associated
with the first byte of the flag fields defined above

```
              OVF1ON    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'          PLIST 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 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'          Overrides are on
              OVSPREV   EQU    X'40'          SVCTRACE SAME' is legal
              OVSAFT    EQU    X'20'          After bit, set by DMSITS
              OVSHO     EQU    X'10'          HALT OVERRIDES flag
              OVSSO     EQU    X'08'          SUSPEND OVERRIDES flag

  2D                    DC     XL3'0'         Unused
  30          SVCSAVE   DC     18F'0'         DMSITS work area
  78          NRMSAV    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 reg 14 for DMSLAD
 168          SVLADW    DS     F              Save reg 14 for DMSLDW
 16C          SVLFS     DS     F              Save reg 14 for DMSLFS
 170                    DC     14F'0'         Reserved for IBM use
```

PLIST for Calling DMSLDR
```
 1A8                    DS     0F
 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)         3-byte zero
```

Hexadecimal Displacement	Field Name			Field Description, Contents, Meaning
1BC	TRANSRT	DC	CL8' '	Holds filename of transient routine
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	RGPF5	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	NRMUSAV	DS	12D	NRMSAV user save area

SVEARA: LTA AND PP SAVE AREA DSECT

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

```
      r-------------------------------------------------------------------¬
  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                   |
      L-------------------------------------------------------------------J
```

Hexadecimal Displacement	Field Name			Field Description, Contents, Meaning
				LTA and PP save area DSECT
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 is the CMS simulation of the DOS/VS System Communication Region (SYSCOM).
The ASYSCOM field (hex 4E4) in NUCON points to the SYSCOM block.

```
 0 |          IJBERBLC          |          IJBAREX          |
   |----------------------------------------------------------|
 8 |  IJBERR19  |  IJBERR24     |          IJBPUBRS          |
   |----------------------------------------------------------|
10 |          IJBFETCH          |          IJBINTRT          |
   |----------------------------------------------------------|
18 |          IJBEXTRT          |          IJBLTA            |
   |----------------------------------------------------------|
20 |          IJBPPBEG          |          IJBCHANQ          |
   |----------------------------------------------------------|
28 |  IJBQSIZE  |   IJBQLNG     |   IJBNPART     |//////////|
   |----------------------------------------------------------|
30 |          IJBRSAVE          |          IJBCONSP          |
   |----------------------------------------------------------|
38 |          IJBSAB            |          IJBCHNTB          |
   |----------------------------------------------------------|
40 | A*1 | A*2 | A*3 | A*4 |   IJBSTID   |   IJBEXIT         |
   |----------------------------------------------------------|
48 |          IJBPCADR          |          IJBTKHLD          |
   |----------------------------------------------------------|
50 |          IJBTIMER          |          IJBABTAB          |
   |----------------------------------------------------------|
58 |  IJBLIK   |   IJBTIK       |          IJBPWR           |
   |----------------------------------------------------------|
60 |          IJBTCAVT          |          IJBRFTAB          |
   |----------------------------------------------------------|
68 |          IJBEUECB          |          IJBOLTEP          |
   |----------------------------------------------------------|
70 |          IJBRASLN          |          IJBTRTAB          |
   |----------------------------------------------------------|
78 |          IJBPEOWN          |          IJBJATAB          |
   |----------------------------------------------------------|
80 |          IJBPMGR           |          IJBCCWT           |
   |----------------------------------------------------------|
88 |          IJBSAVSD          |          IJBLNSTB          |
   |----------------------------------------------------------|
90 |          IJBAMCOM          |          IJBAPTA           |
   |----------------------------------------------------------|
98 |          IJBSBLK0          |          IJBSBLKX          |
   |----------------------------------------------------------|
A0 | A*5 | A*6 | A*7 | A*8 | A*9 | A*10 |////////////|
   |----------------------------------------------------------|
A8 |/////////////////////////////////////////////////////////|
   |----------------------------------------------------------|
B0 |///////////////////////////|          IJBMVCAD          |
   |----------------------------------------------------------|
B8 |//////////////////////////////////////////////////////////|
   |----------------------------------------------------------|
C0 |                      IJBMFCER                            |
   |----------------------------------------------------------|
C8 |          | A*11 |   IJBPUBLN   |   IJBAPNO              |
   +----------------------------------------------------------+
```

```
      D0|----------------------------------------------------------------|
        |         IJBSEGT             |              IJBPFT              |
        |----------------------------------------------------------------|
      D8|         IJBPFTX             |              IJBBOX              |
        |----------------------------------------------------------------|
      E0|         IJBDPDTB            |//////////////////////////////////|
        |----------------------------------------------------------------|
      E8|         IJBVIRAD            |              IJBEOR              |
        |----------------------------------------------------------------|
      F0|         IJBFTTAB            |              IJBSVA              |
        |----------------------------------------------------------------|
      F8|         IJBSVIS             |              ARPSL               |
        |----------------------------------------------------------------|
     100|         ARPSR              |              IJBDLAB             |
        |----------------------------------------------------------------|
      10|                      SYS$CODE                                   |
        |                                  r-----------------------------'|
     110|                                  |
        |_____|
```

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'	SYSRES PUB 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	IJBLTA	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	IJBSAB	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 125 RMS
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	IJBLIK	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	IJBTCAVT	DC	A(0)		Space for VTAM address
64	IJBRFTAB	DC	A(0)		Pointer to RF table
68	IJBEUECB	DC	A(0)		Pointer to EU and ECB table
6C	IJBOLTEP	DC	A(0)		Address of OLTEP bucket
70	IJBRASLN	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 comm. area
80	IJBPMGR	DC	A(0)		Base address of PMGR routines
84	IJBCCWT	DC	A(0)		Address of CCW trans work
88	IJBSAVSD	DC	A(0)		Pointer to SDAID comm. area
8C	IJBLNSTB	DC	A(0)		Address of line mode table
90	IJBAMCOM	DC	A(0)		Address of VSAM comm. reg.
94	IJBAPTA	DC	A(0)		Address of PTA
98	IJBSBLKO	DC	A(0)		Pointer to first system task block
9C	IJBSBLKX	DC	F'0'		Pointer to current system task
A0	IJBSYSPT	DC	X'00'	A*5	For alignment
A1	IJBRASPT	DC	AL1(0)	A*6	Pointer to RAS task block
A2	IJBPMRPT	DC	AL1(0)	A*7	Pointer for PMGR task block
A3	IJBSUPPT	DC	AL1(0)	A*8	Pointer to SPVR task block
A4	IJBCRTPT	DC	AL1(0)	A*9	Pointer to CRT task block
A5	IJBERPPT	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	IJBMVCAD	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	IJBMFCER	DS	11X'00'		Information on MFCM and MFCU ERP
CB	IJBNERQ	DC	AL1(0)	A*11	Number of error queue entries
CC	IJBPUBLN	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	IJBPFT	DC	A(0)		Address of page frame table
D8	IJBPFTX	DC	A(0)		Pointer to page frame table extension
DC	IJBBOX	DC	A(0)		Pointer to boundary box
E0	IJBDPDTB	DC	A(0)		Pointer to DPD table
E4		DC	F'0'		Reserved for IBM use
E8	IJBVIRAD	DC	A(0)		Address of VIRTAD routine
EC	IJBEOR	DC	F'0'		End of real storage
F0	IJBFTTAB	DC	A(0)		Address of the fetch table
F4	IJBSVA	DC	A(0)		Address of the SVA start
F8	IJBSVIS	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	IJBDLAB	DC	A(SYS$CODE)		Pointer to system code name
108	SYS$CODE	DC	CL13'CMS/VSAM'		System code name

SYSNAMES: SAVED SYSTEMS NAMES

SYSNAMES defines the names of any saved systems which may be loaded by CMS routines.
ASYSNAMES in NUCON (hex location 5EC) points to the SYSNAMES table.

```
       r------------------------------------------------------------------¬
    0  |                            CMSSEG                                 |
       |------------------------------------------------------------------|
    8  |                            CMSVSAM                                |
       |------------------------------------------------------------------|
   10  |                            CMSAMS                                 |
       |------------------------------------------------------------------|
   18  |                            CMSDOS                                 |
       L------------------------------------------------------------------
```

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	

TSOBLKS: TSO CONTROL BLOCKS

TSOBLKS 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 (hex 5D8) in NUCON points to TSOBLKS.

```
      r-----------------------------------------------------------------------¬
 0    | CPPLOBUF                       |        CPPLUPT                        |
      |-----------------------------------------------------------------------|
 8    | CPPLPSCB                       |        CPPLECT                        |
      |-----------------------------------------------------------------------|
10    |//////////|                    UPTUSER                                 |
      |-----------------------------------------------------------------------|
18    |   UPTUSER (cont.)              | A*1 | A*2 | A*3 |/////////|
      |-----------------------------------------------------------------------|
20    |                   PSCBUSER                          | A*4 |
      |-----------------------------------------------------------------------|
28    |                   PSCBGPNM                                            |
      |-----------------------------------------------------------------------|
30    | A*5 |////////| A*6 |////////|                                         |
      |-----------------------------------------------------J                 |
      .                                                      .
      .                                                      .
      .                                                      .
      |                                 r-------------------------------------|
48    |                                 |        PSCBTCOL                     |
      |-----------------------------------------------------------------------|
50    |       PSCBRLGB                  |        PSCBUPT                       |
      |-----------------------------------------------------------------------|
58    |    PSCBUPTL    |////////////|           PSCBRSZ                       |
      |-----------------------------------------------------------------------|
60    |                         PSCBU                                         |
      |-----------------------------------------------------------------------|
68    | A*7 |     ECTRTCD               |        ECTIOWA                       |
      |-----------------------------------------------------------------------|
70    | A*8 |     ECTSMSG               |        ECTPCMD                       |
      |-----------------------------------------------------------------------|
78    |     ECTPCMD (cont.)            |        ECTSCMD                       |
      |-----------------------------------------------------------------------|
80    |     ECTSCMD (cont.)            | A*9 |   ECTDDNUM                      |
      |-----------------------------------------------------------------------|
88    |          ECTUSER              |//////////////////////////////////////|
      L-----------------------------------------------------------------------
```

Hexadecimal Displacement	Field Name			Field Description, Contents, Meaning
0	CPPL	DS	0F	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

User Profile Table (UPT)
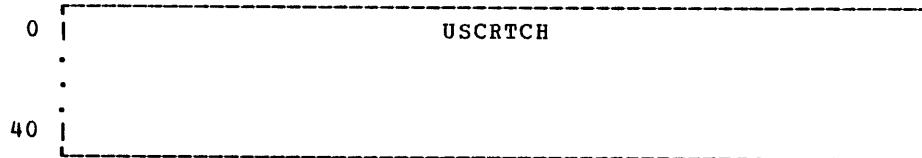
10	UPT	DS	0F	
10		DS	CL2	Reserved for IBM use
12	UPTUSER	DS	CL10	Reserved for installation use

Hexadecimal Displacement	Field Name				Field Description, Contents, Meaning
1C	UPTSWS	DC	X'00'	A*1	User's environment switch
	Bits defined in UPTSWS				
		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
	UPTPAUS	EQU	X'08'		Pause for '?' when in non-interface 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

Protected Step Control Block (PSCB)

Hexadecimal Displacement	Field Name				Field Description, Contents, Meaning
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	15 bit string of user attributes
	PSCBCTRL	EQU	X'80'		Operator command user
	PSCBACCT	EQU	X'40'		Account command user
	PSCBJCL	EQU	X'20'		Submit command user
	*	BITS 3	15		Reserved for IBM use
31		DS	X		Reserved for IBM use
32	PSCBATR2	DS	X	A*6	15 bit string reserved for installation use
33		DS	X		
34		DC	6F'0'		6 fullwords used for TSO accouting; 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 UPT
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

Environment Control Table (ECT)

Hexadecimal Displacement	Field Name				Field Description, Contents, Meaning
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 work area
70	ECTMSGF	DC	X'00'	A*8	High-order bit means delete 2nd level messages (Must be initialized by user.)
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
	Bits defined in ECTSWS				
	ECTNOPD	EQU	X'80'		0 bit= on, no operands exist in command buffer
	ECTATRM	EQU	X'20'		CP terminated by TMP DETACH w/ STAE
	*	EQU	X'40'		Reserved for IBM use
	ECTLOGF	EQU	X'10'		Logon/off requested tmp to log off
	ECTNMAL	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 temporary DDNAMS
88	ECTUSER	DS	A		Reserved for iinstallation
8C		DS	A		Reserved for IBM use

USERSECT: USER WORK AREA

USERSECT defines a general scratch storage area provided for user-defined purposes. It may be redefined to suit installation requirements.

The USERSECT is pointed to by the AUSER field (hex 620) in NUCON.

```
      ┌──────────────────────────────────────────────────────────┐
   0  │                          USCRTCH                         │
      .                                                          .
      .                                                          .
      .                                                          .
  40  │                                                          │
      └──────────────────────────────────────────────────────────┘
```

Hexadecimal Displacement	Field Name			Field Description, Contents, Meaning
0	USCRTCH	DC	18F'0'	User scratch area

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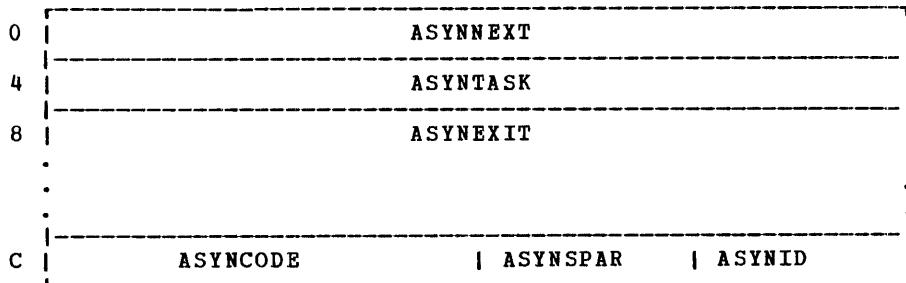
This section of the publication contains only DSECTs. Appendixes B and C contain other control areas used by RSCS.

ASYNE: 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.

```
     r--------------------------------------------------------------1
  0  |                       ASYNNEXT                               |
     |--------------------------------------------------------------|
  4  |                       ASYNTASK                               |
     |--------------------------------------------------------------|
  8  |                       ASYNEXIT                               |
     .                                                              .
     .                                                              .
     .                                                              .
     |----------------------------------------|---------|-----------|
  C  |            ASYNCODE                     | ASYNSPAR | ASYNID   |
     L--------------------------------------------------------------J
```
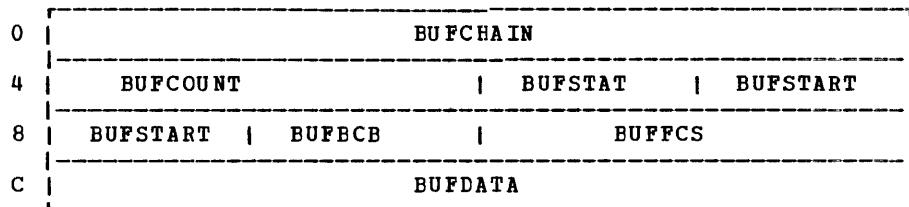
Hexadecimal Displacement	Field Name	Field Description, Contents, Meaning
0	ASYNNEXT DS 1F	Address of the next asynchronous interrupt exit request element
4	ASYNTASK DS 1F	Address of task element describing the task that requested the asynchronous interrupt
8	ASYNEXIT DS 1F	Address of the requested asynchronous exit routine
C	ASYNCODE DS AL2	Address of the device for which asyncronous I/O interrupts are requested or interrupt bit code
E	ASYNSPAR DS 1X	Reserved for IBM use
F	ASYNID DS 1X	1-byte ID 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 SML line driver but are not transmitted.
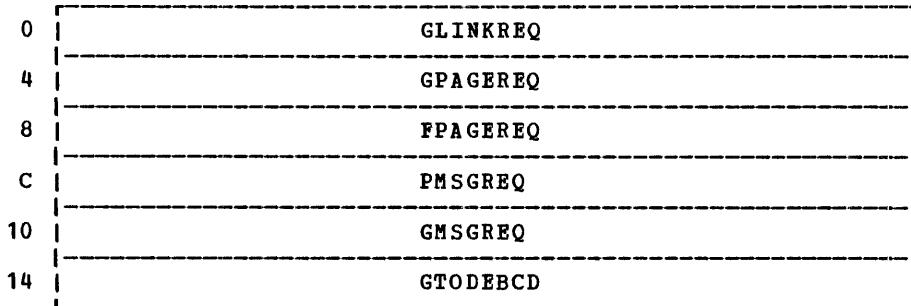
$BUFPOOL in SML points to a queue of available TP buffers; $INBUF in SML points to a queue of TP buffers that have been received from a remote station and are waiting to be deblocked. $OUTBUF in SML points to a queue of TP buffers that are ready for transmission to remote stations.

```
   ┌────────────────────────────────────┬──────────────────────────────┐
0  │                    BUFCHAIN                                        │
   ├──────────────────────────────────┬──────────────┬─────────────────┤
4  │          BUFCOUNT                 │   BUFSTAT    │    BUFSTART     │
   ├──────────────────┬───────────────┴──────┬───────┴─────────────────┤
8  │    BUFSTART      │    BUFBCB            │          BUFFCS          │
   ├──────────────────┴─────────────────────┴──────────────────────────┤
C  │                    BUFDATA                                         │
   └────────────────────────────────────────────────────────────────────┘
```

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
	Bits defined in BUFSTAT		
	BUFFAKE	EQU X'01'	Dummy buffer indicator
	BUFRESP	EQU X'02'	Response only in buffer
	BUFNAK	EQU X'04'	NAK response being sent
	BUFTEXT	EQU X'08'	Buffer contains text information
	BUFUCHEK	EQU X'10'	Unit check expected
7	BUFSTART	DS CL2	Bisynch 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
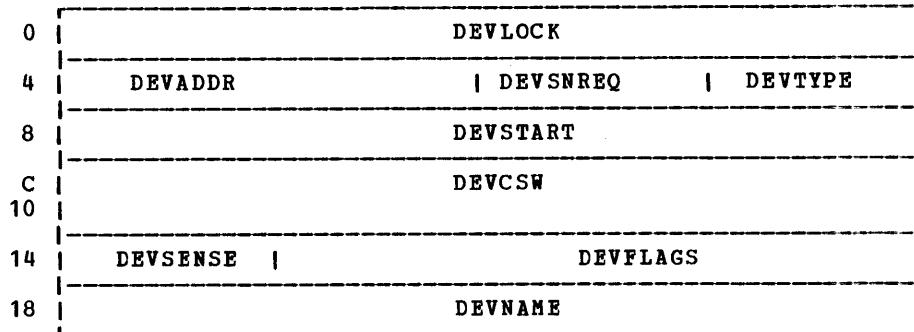
COMDSECT defines address constants used as pointers to subroutines common to all RSCS modules. These subroutines are contained in module DMTCOM. COMDSECT is pointed to by TCOM in SVECTORS.

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

Hexadecimal Displacement	Field Name	Field Description, Contents, Meaning
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.

```
        r------------------------------------------------------------------¬
    0   | DEVLOCK                                                          |
        |-------------------------------------------------------------------|
    4   |     DEVADDR             | DEVSNREQ      |   DEVTYPE               |
        |-------------------------------------------------------------------|
    8   |                         DEVSTART                                  |
        |-------------------------------------------------------------------|
    C   |                         DEVCSW                                    |
   10   |                                                                  |
        |-------------------------------------------------------------------|
   14   |  DEVSENSE  |                 DEVFLAGS                             |
        |-------------------------------------------------------------------|
   18   |                         DEVNAME                                   |
        L------------------------------------------------------------------
```
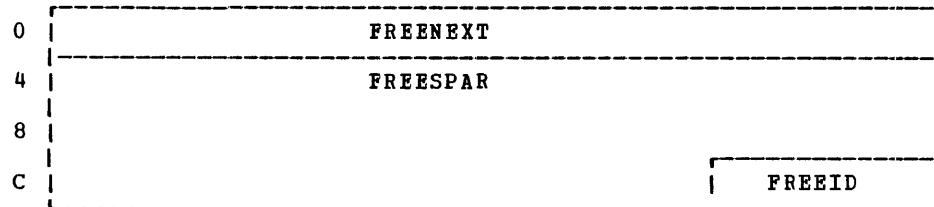
Hexadecimal Displacement	Field Name			Field Description, Contents, Meaning
0	DEVLOCK	DC	F'0'	I/O request synch lock
4	DEVADDR	DC	AL2(0)	CUU address of I/O device
6	DEVSNREQ	DC	AL1(0)	Requested sense bytes
7	DEVTYPE	DC	X'00'	VM/370 device type code
8	DEVSTART	DC	F'0'	Address start channel program for 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: A FREE ELEMENT ON THE SUPERVISOR ELEMENT QUEUE

The FREEE DSECT 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.

```
      r------------------------------------------------------------------1
  0   |                         FREENEXT                                  |
      |------------------------------------------------------------------|
  4   |                         FREESPAR                                  |
      |                                                                  |
  8   |                                                                  |
      |                                          r--------------------|
  C   |                                          |      FREEID         |
      L------------------------------------------------------------------J
```
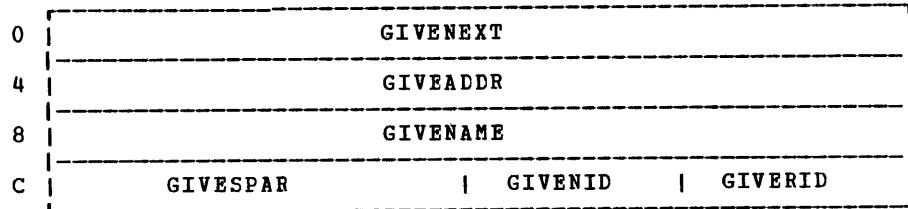
Hexadecimal Displacement	Field Name	Field Description, Contents, Meaning
0	FREENEXT DS 1F	Address of next element in free queue
4	FREESPAR DS CL11	Spare field
F	FREEID DS 1X	Standard taskid offset --ID=X'00'=> free element

GIVEE: A GIVE ELEMENT

The GIVEE DSECT 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	Unused
E	GIVENID	DS	1X	1-byte ID of receiving task after TAKE
F	GIVERID	DS	1X	1-byte ID of sending task

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.

TAKE REQUEST TABLE

The format of a TAKE request table is:
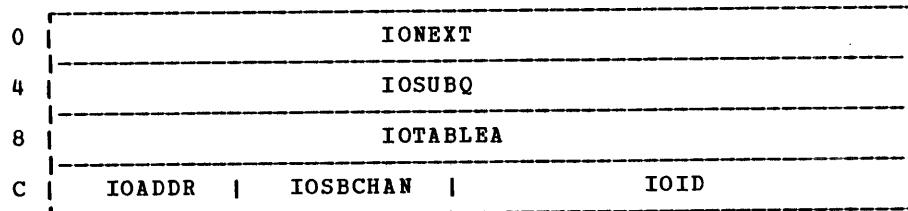
```
     ┌────────────────────────────────────────────────────┐
0    │            Task name of GIVE requestor             │
     ├────────────────────────────────────────────────────┤
4    │              A(TAKE Request Buffer)                │
     ├────────────────────────────────────────────────────┤
8    │              A(TAKE Response Buffer)               │
     └────────────────────────────────────────────────────┘
```

IOE: AN I/O ELEMENT

The IOE DSECT 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/O elements for the multiplexer and selector channels, respectively.

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

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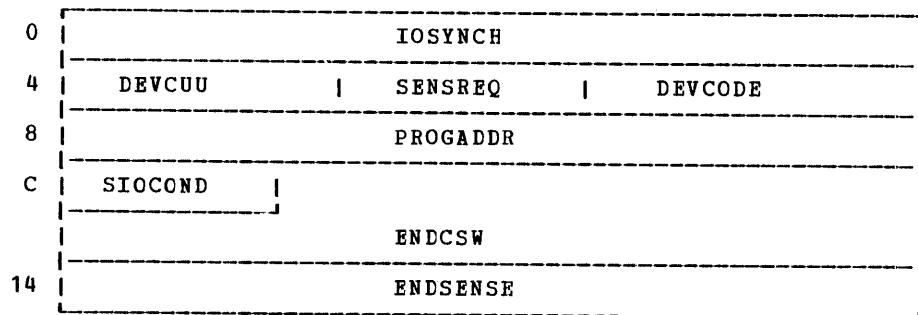
Hexadecimal Displacement	Field Name			Field Description, Contents, Meaning
0	IONEXT	DS	1F	Address of next active I/O element
4	IOSUBQ	DS	1F	Address of next inactive I/O element for a given subchannel
8	IOTABLEA	DS	1F	Address of I/O request table in task storage
	IOSTAT	EQU	*	Status flags for current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

IOTABLE: AN I/O TABLE

The IOTABLE DSECT 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.

```
      ┌─────────────────────────────────────────────────────────┐
 0    │                        IOSYNCH                           │
      ├──────────────────┬──────────────┬───────────────────────┤
 4    │      DEVCUU      │   SENSREQ    │      DEVCODE           │
      ├──────────────────┴──────────────┴───────────────────────┤
 8    │                        PROGADDR                          │
      ├──────────────────┬──────────────────────────────────────┤
 C    │    SIOCOND       │                                       │
      ├──────────────────┘                                       │
      │                        ENDCSW                            │
      ├─────────────────────────────────────────────────────────┤
14    │                        ENDSENSE                          │
      └─────────────────────────────────────────────────────────┘
```

Hexadecimal Displacement	Field Name			Field Description, Contents, Meaning
0	IOSYNCH	DS	1F	Synchronization lock for I/O 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/O operation
	SIOCOND	EQU	*	1-byte SIO condition code return information
C	ENDCSW	DS	2F	Ending CSW with composite status return information
14	ENDSENSE	DS	AL1	Requested return sense information on unit check CSW status
	TYPPUN	EQU	X'80'	VM/370 type code for the punch
	TYPPRT	EQU	X'40'	VM/370 type code for the printer

LINKTABL DSECT

The LINKTABL DSECT 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.

```
       r--------------------------------------------------------------¬
  0  |                         LINKID                                |
     |                                                               |
  4  |                                                               |
     |---------------------------------------------------------------|
  8  |                        LDEFTNME                               |
     |---------------------------------------------------------------|
  C  |                        LACTTNME                               |
     |---------------------------------------------------------------|
 10  |                        LDEFDRVR                               |
     |                                                               |
 14  |                                                               |
     |---------------------------------------------------------------|
 18  |                        LACTDRVR                               |
     |                                                               |
 1C  |                                                               |
     |---------------------------------------------------------------|
 20  |         LDEFLINE          |           LACTLINE               |
     |---------------------------------------------------------------|
 24  |                        LDRVRVAR                               |
     |---------------------------------------------------------------|
 28  |  LACTCLS1   |  LACTCLS2   |   LACTCLS3   |   LACTCLS4        |
     |---------------------------------------------------------------|
 30  |  LTIMEZON   |   LFLAG     |         LRESERVD                 |
     |---------------------------------------------------------------|
 34  |        LPENDING           |          LTAKEN                  |
     |---------------------------------------------------------------|
 38  |                        LPOINTER                              |
     |---------------------------------------------------------------|
 3C  |                        LMSGQ                                 |
     |---------------------------------------------------------------|
 40  |        LTRNSCNT           |          LERRCNT                 |
     |---------------------------------------------------------------|
 44  |        LTOCNT             |          LSPARE                  |
     |---------------------------------------------------------------|
 48  |                        LNKCLOCK                              |
     |                                                               |
 4C  |                                                               |
       L--------------------------------------------------------------
```

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
     r------------------------------------------------------------¬
     | Total links      |max links| current                     |
     |                  |         | links                       |
     L------------------------------------------------------------
```

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.)

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

current links is the number of links active in RSCS at a given time.
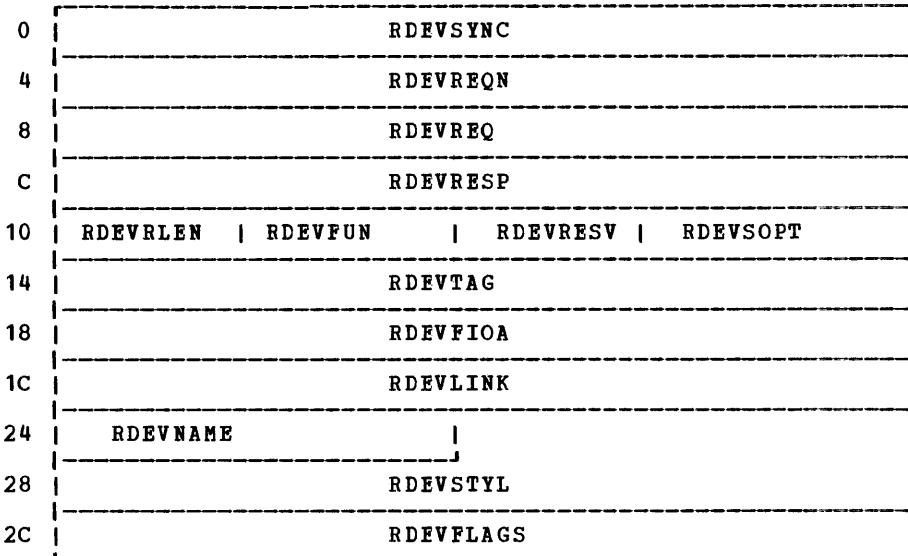
Hexadecimal Displacement	Field Name			Field Description, Contents, Meaning
0	LINKID	DS	CL8	EBCDIC linkid
8	LDEFTNME	DS	CL4	Default task name
C	LACTTNME	DS	CL4	Active task name
10	LDEFDRVR	DS	CL8	Default driver id
18	LACTDRVR	DS	CL8	Active driver id
20	LDEFLINE	DS	2X	Default virtual line address
22	LACTLINE	DS	2X	Active virtual line address
24	LDRVRVAR	DS	1F	Line driver variable information
28	LDEFCLS1	DS	CL1	Default spool file CLS 1
29	LDEFCLS2	DS	CL1	Default spool file CLS 2
2A	LDEFCLS3	DS	CL1	Default spool file CLS 3
2B	LDEFCLS4	DS	CL1	Default spool file CLS 4
2C	LACTCLS1	DS	CL1	Active spool file CLS 1
2D	LACTCLS2	DS	CL1	Active spool file CLS 2
2E	LACTCLS3	DS	CL1	Active spool file CLS 3
2F	LACTCLS4	DS	CL1	Active spool file CLS 4
30	LTIMEZON	DS	1X	2 comp time zone displacement from GMT
31	LFLAG	DS	1X	Link flag byte
	Bits defined in LFLAG			
	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: NPT REQUEST BLOCK

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

The first four fields (hex 0-F) of this DSECT form a GIVE request table, which is
described in "GIVEE: A GIVE Element", in this section.

The next seven fields (hex 10-23) of this DSECT for a GIVE request buffer in the
format of a File Request Element, which is described in "Appendix C: RSCS Request
Elements."

```
      r---------------------------------------------------------------1
   0  |                         RDEVSYNC                              |
      |--------------------------------------------------------------|
   4  |                         RDEVREQN                              |
      |--------------------------------------------------------------|
   8  |                         RDEVREQ                               |
      |--------------------------------------------------------------|
   C  |                         RDEVRESP                              |
      |--------------------------------------------------------------|
  10  | RDEVRLEN   | RDEVFUN      |  RDEVRESV  |   RDEVSOPT           |
      |--------------------------------------------------------------|
  14  |                         RDEVTAG                               |
      |--------------------------------------------------------------|
  18  |                         RDEVFIOA                              |
      |--------------------------------------------------------------|
  1C  |                         RDEVLINK                              |
      |--------------------------------------------------------------|
  24  |    RDEVNAME                |                                  |
      |--------------------------J                                   |
  28  |                         RDEVSTYL                              |
      |--------------------------------------------------------------|
  2C  |                         RDEVFLAGS                             |
      L---------------------------------------------------------------J
```

Hexadecimal Displacement	Field Name			Field Description, Contents, Meaning
	Hex locations 0 through F comprise the NPT GIVE request table			
0	RDEVSYNC	DC	F'0'	Synch 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
	Hex locations 0 through 1F comprise the NPT GIVE request buffer in the format of a file request element (as shown in Appendix C)			
10	RDEVRLEN	DC	AL1(0)	Request length
11	RDEVFUN	DC	AL1(0)	Request function
12	RDEVRESV	DC	AL1(0)	Reserved byte
13	RDEVSOPT	DC	AL1(0)	Sub 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
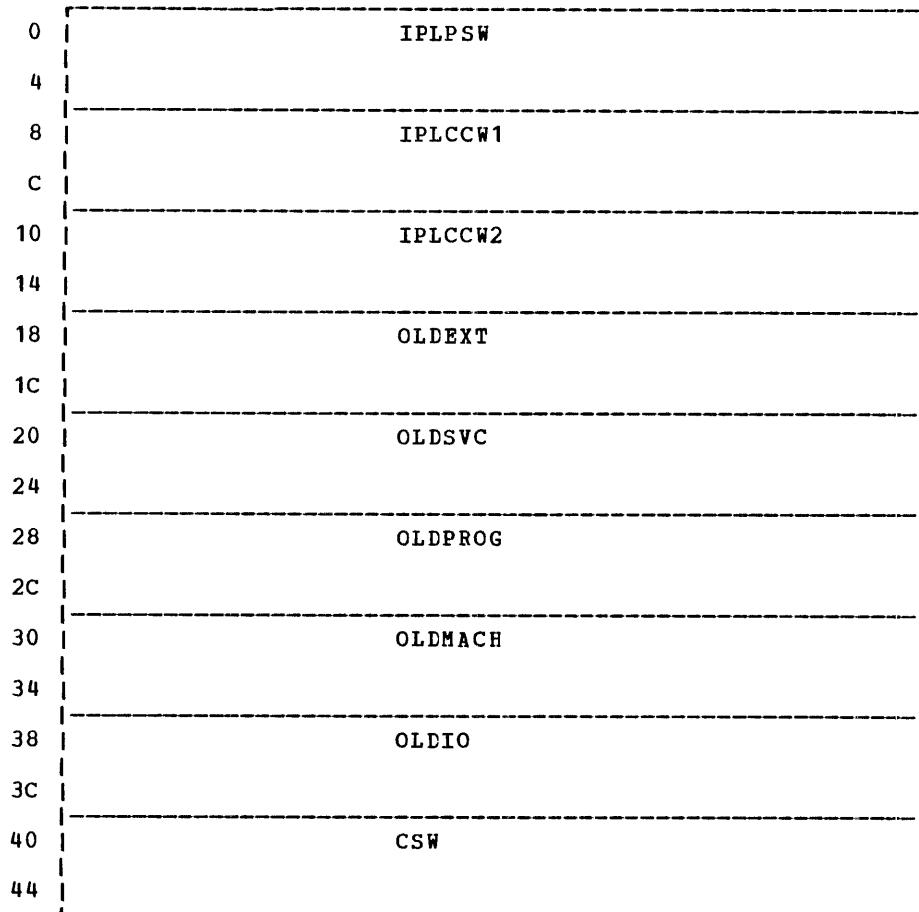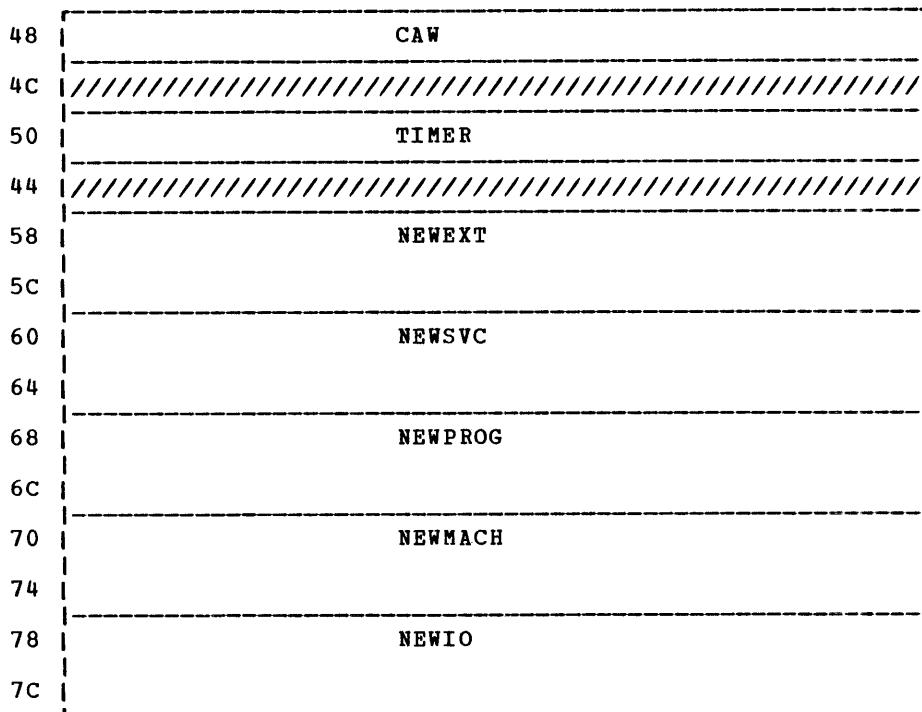
SVECTORS: LOW STORAGE DEFINITIONS

The SVECTORS DSECT 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.

```
        r-------------------------------------------------------------¬
   0    |                        IPLPSW                                |
        |                                                             |
   4    |                                                             |
        |-------------------------------------------------------------|
   8    |                        IPLCCW1                               |
        |                                                             |
   C    |                                                             |
        |-------------------------------------------------------------|
  10    |                        IPLCCW2                               |
        |                                                             |
  14    |                                                             |
        |-------------------------------------------------------------|
  18    |                        OLDEXT                                |
        |                                                             |
  1C    |                                                             |
        |-------------------------------------------------------------|
  20    |                        OLDSVC                                |
        |                                                             |
  24    |                                                             |
        |-------------------------------------------------------------|
  28    |                        OLDPROG                               |
        |                                                             |
  2C    |                                                             |
        |-------------------------------------------------------------|
  30    |                        OLDMACH                               |
        |                                                             |
  34    |                                                             |
        |-------------------------------------------------------------|
  38    |                        OLDIO                                 |
        |                                                             |
  3C    |                                                             |
        |-------------------------------------------------------------|
  40    |                        CSW                                   |
        |                                                             |
  44    |                                                             |
        L-------------------------------------------------------------
```

```
     48 |------------------------------------------------|
        |                       CAW                      |
     4C |------------------------------------------------|
        |////////////////////////////////////////////////|
     50 |------------------------------------------------|
        |                      TIMER                     |
     44 |------------------------------------------------|
        |////////////////////////////////////////////////|
     58 |------------------------------------------------|
        |                                                |
        |                     NEWEXT                      |
     5C |                                                |
        |------------------------------------------------|
     60 |                                                |
        |                     NEWSVC                      |
     64 |                                                |
        |------------------------------------------------|
     68 |                                                |
        |                     NEWPROG                     |
     6C |                                                |
        |------------------------------------------------|
     70 |                                                |
        |                     NEWMACH                     |
     74 |                                                |
        |------------------------------------------------|
     78 |                                                |
        |                      NEWIO                      |
     7C |                                                |
        |------------------------------------------------|
```

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	Unused
50	TIMER	DS	F	4X'FF'
54		DS	F	Unused
58	NEWEXT	DS	D	X'00040000',V(DMTEXT)
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 hex location 200 and is defined specifically for the RSCS virtual machine. It contains pointers to modules that comprise the supervisor, to supervisor control queues, and to queues of requests for supervisor services.

200	NEWPSW
204	NEWPSW
208	SSAVE
20C	SSAVE
210	ACTIVE
214	MAINMAP
218	MAINSIZE
21C	QUEUE
220	QUEUEND
224	FREEQ
228	TASKQ
22C	MPXIOQ
230	SELIOQ
234	IOEXITQ
238	EXTQ
23C	ALERTQ
240	GIVEQ
244	QREQ

248	DISPATCH
24C	WAITREQ
250	POSTREQ
254	IOREQ
258	TASKREQ
25C	MAINREQ
260	ASYNREQ
264	ALERTREQ
268	GIVEREQ
26C	TAKEREQ
270	TVECTOR0
274	TVECTOR1
278	TVECTOR2
27C	TVECTOR3
280	TVECTOR4
284	TVECTOR5
288	TVECTOR6
28C	TVECTOR7
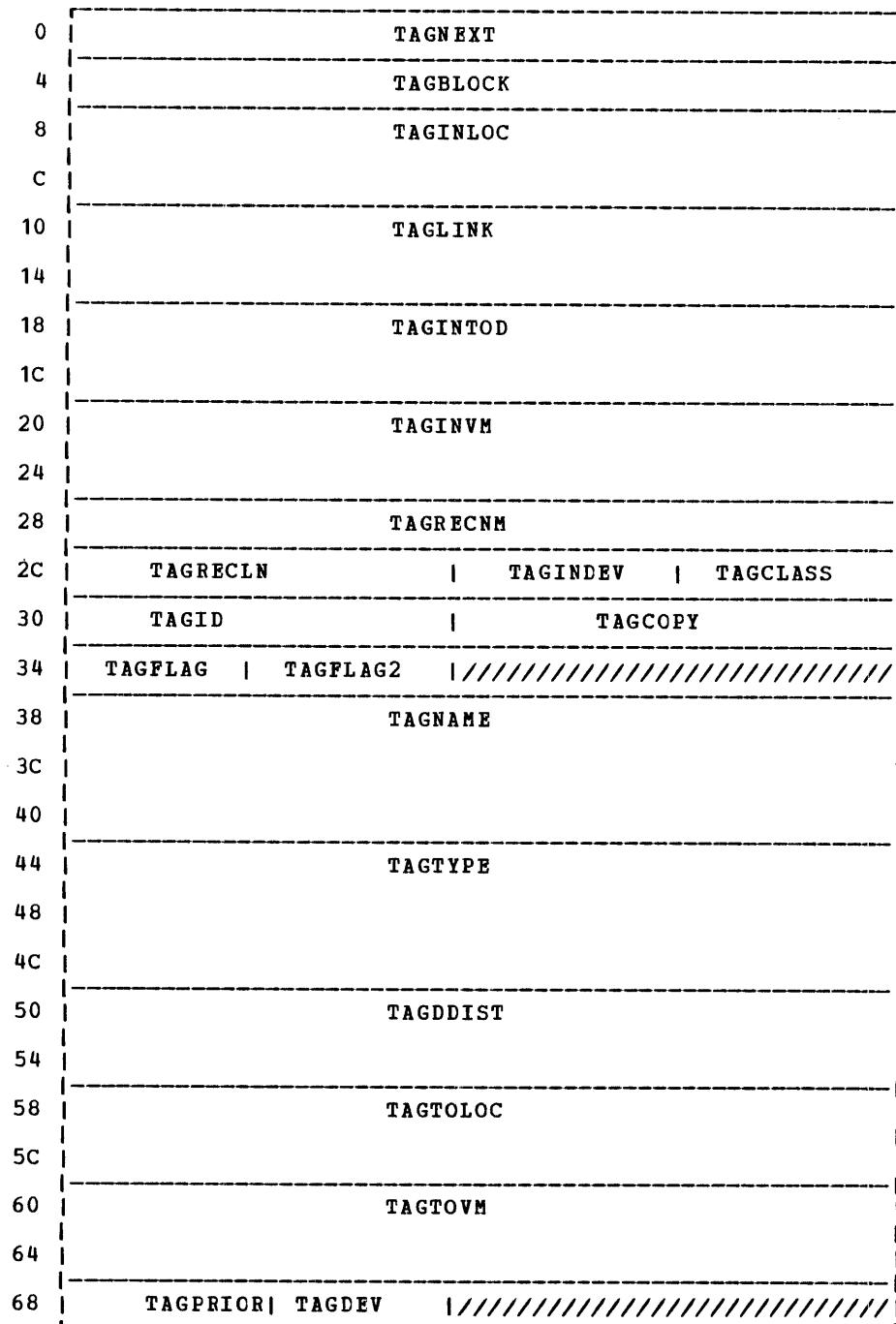
Hexadecimal Displacement	Field Name			Field Description, Contents, Meaning
		ORG	SVECTORS+X'200'	Leave room for machine extensions
200	NEWPSW	DS	D	D'0' Dispatched PSW for last dispatcher
20C	SSAVE	DS	2F	2F'0' General-purpose low storage save area
210	ACTIVE	DS	X	X'00' ID of currently active task
		DS	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

Hexadecimal Displacement	Field Name			Field Description, Contents, Meaning
21C	QUEUE	DS	V(SQUEUE)	Address of start of supervisor queue
220	QUEUEND	DS	V(SQUEUEND)	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 selector 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	LERTREQ	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: THE RSCS FILE DESCRIPTOR

The TAG DSECT 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 spool 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        │////////////////////////// │
     ├──────────────┴───────────────────┴──────────────────────────┤
 38  │                        TAGNAME                              │
     │                                                            │
 3C  │                                                            │
     │                                                            │
 40  │                                                            │
     ├────────────────────────────────────────────────────────────┤
 44  │                        TAGTYPE                              │
     │                                                            │
 48  │                                                            │
     │                                                            │
 4C  │                                                            │
     ├────────────────────────────────────────────────────────────┤
 50  │                        TAGDDIST                             │
     │                                                            │
 54  │                                                            │
     ├────────────────────────────────────────────────────────────┤
 58  │                        TAGTOLOC                             │
     │                                                            │
 5C  │                                                            │
     ├────────────────────────────────────────────────────────────┤
 60  │                        TAGTOVM                              │
     │                                                            │
 64  │                                                            │
     ├──────────────┬───────────────────────────────────────────────┤
 68  │  TAGPRIOR│ TAGDEV    │////////////////////////////////////// │
     └────────────────────────────────────────────────────────────┘
```
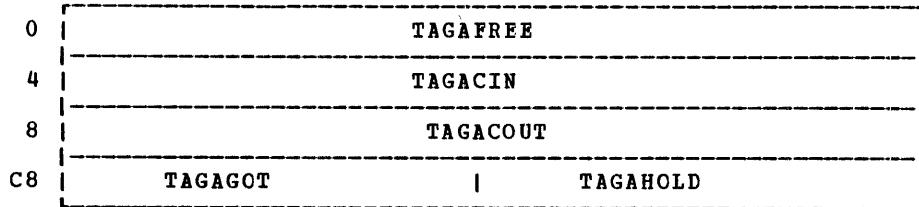
Hexadecimal Displacement	Field Name			Field Description, Contents, Meaning
0	TAGNEXT	DS	1F	Address of next active queue entry
4	TAGBLOCK	DS	1F	Address of associated I/O area
8	TAGINLOC	DS	CL8	Originating location
10	TAGLINK	DS	CL8	Next location for transmission
18	TAGINTOD	DS	CL8	Time of file origin
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 SFBLOK control flags (SFBFLAG)
35	TAGFLAG2	DS	1X	VM/370 SFBLOK control flags (SFBFLAG)
36		DS	1H	Spare halfword
3A	TAGNAME	DS	CL12	Filename
44	TAGTYPE	DS	CL12	Filetype
50	TAGDIST	DS	CL8	File distribution code
58	TAGTCLOC	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 of the file TAG

TAGAREA

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

```
      ┌─────────────────────────────────────────────────────────────┐
   0  │                        TAGAFREE                              │
      ├─────────────────────────────────────────────────────────────┤
   4  │                        TAGACIN                               │
      ├─────────────────────────────────────────────────────────────┤
   8  │                        TAGACOUT                              │
      ├──────────────────────────────────┬──────────────────────────┤
  C8  │          TAGAGOT                  │        TAGAHOLD          │
      └──────────────────────────────────┴──────────────────────────┘
```
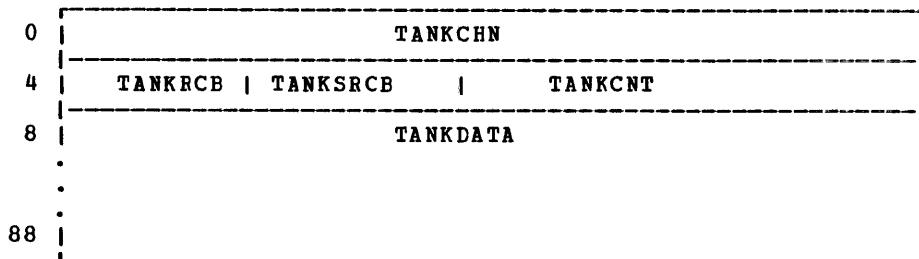
Hexadecimal Displacement	Field Name			Field Description, Contents, Meaning
0	TAGAFREE	DC	A(0)	Address of queue of free TAG slots (or elements)
4	TAGACIN	DC	A(0)	Pointer to queue of active input TAGs
8	TAGACOUT	DC	A(0)	Pointer to queue of active output TAGs
C	TAGAGOT	DC	H'0'	Number free slots left
E	TAGAHOLD	DC	H'0'	Number slots to be held

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 SML points to a queue of available tanks.

```
     r-----------------------------------------------------------------------,
  0  |                          TANKCHN                                      |
     |-----------------------------------------------------------------------|
  4  |   TANKRCB  |  TANKSRCB       |        TANKCNT                         |
     |-----------------------------------------------------------------------|
  8  |                          TANKDATA                                     |
     .                                                                       .
     .                                                                       .
     .                                                                       .
 88  |                                                                       |
     L-----------------------------------------------------------------------J
```
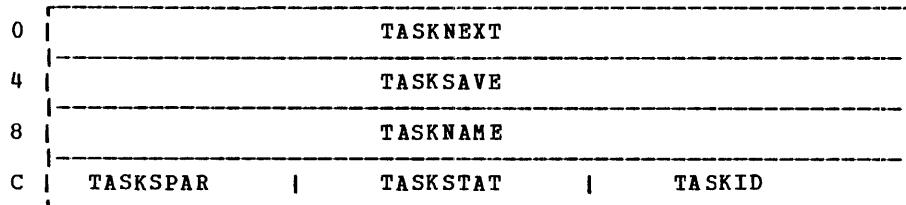
Hexadecimal Displacement	Field Name			Field Description, Contents, Meaning
0	TANKCHN	DC	A(0)	Tank chain field
4	TANKRCB	DC	1C	Tank record control byte
5	TANKSRCB	DS	1C	Tank subrecord control byte
6	TANKCNT	DS	1H	Count of data bytes in tank
8	TANKDATA	DS	CL132	Data area in tank
8C	TANKEND	DS	0F	Force next to word boundary

TASKE: A TASK ELEMENT

The TASKE DSECT 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.

```
    +------------------------------------------------------------------+
  0 |                          TASKNEXT                                |
    |------------------------------------------------------------------|
  4 |                          TASKSAVE                                |
    |------------------------------------------------------------------|
  8 |                          TASKNAME                                |
    |------------------------------------------------------------------|
  C |     TASKSPAR     |     TASKSTAT     |         TASKID             |
    +------------------------------------------------------------------+
```

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
F	TASKID DS 1X	Number ID for the task; 1 byte is assigned by supervisor when task is made active

Bits defined in TASKSTAT

WAITING	EQU X'80'	Flag set to 1 when task is non-dispatchable
LOCKLIST	EQU X'40'	Flag set to 1 while task is waiting for the synch lock list
LIMBO	EQU X'01'	Flag set to 1 when a task is being terminated

TCTDSECT: SML TASK CONTROL TABLE

TCTDSECT defines the format of six tables in SML storage which you can find at labels $CCOM1, $WCOM1, $PCOM1, $RCOM1, $UCOM1, and $JCOM1. Each table corresponds to an SML input/output processor and is used by that processor to perform its I/O function.

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

```
        r------------------------------------------------------------------1
14  |                            TCTSAV1                                  |
    |--------------------------------------------------------------------|
18  |                            TCTNEXT                                  |
    |--------------------------------------------------------------------|
1C  |        TCTFCS                    |   TCTRCBR  |  TCTRCBT            |
    |--------------------------------------------------------------------|
20  |                            TCTCOM                                  |
    |--------------------------------------------------------------------|
24  |                            TDEVSYNC                                |
    |--------------------------------------------------------------------|
28  |                            TDEVREQN                                |
    |--------------------------------------------------------------------|
2C  |                            TDEVREQ                                 |
    |--------------------------------------------------------------------|
30  |                            TDEVRESP                                |
    |--------------------------------------------------------------------|
34  |  TDEVRLEN   |   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   |   TCTBUFCT              |
    L--------------------------------------------------------------------J
```

TCTDSECT

Hexadecimal Displacement	Field Name			Field Description, Contents, Meaning
0	TTCT	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

Bits defined in TCTECB

	TCTBUSY	EQU	X'10'	Device busy bit

11	TCTSTAT	DS	CL1	Status flags

Bits defined in TCTSTAT

	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

Hex locations 24 through 30 comprise the SML give request table

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

Hex locations 34 through 40 comprise the give request buffer in the form of a file request element (as shown in Appendix C)

34	TDEVRLEN	DS	AL1	Request length
35	TDEVFUN	DS	AL1	Request function
36	TDEVRESV	DS	AL1	Reserved byte
37	TDEVSOPT	DS	AL1	Sub 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	VM output destination
54	TCTTANK	DS	1F	Next tank to output
58	TCTBUFER	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

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 |  |                      TPSW                        |
        |--------------------------------------------------|
   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 |  |                     TGREF12                      |
        |--------------------------------------------------|
  3C |  |                     TGREG13                      |
        |--------------------------------------------------|
  40 |  |                     TGREG14                      |
        |--------------------------------------------------|
  44 |  |                     TGREG15                      |
        |--------------------------------------------------|
  48 |  |                    TREQLOCK                      |
        '--------------------------------------------------'
```

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

The following appendixes supplement the information in 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" contains areas containing 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 Blocks References" contains the names of CP, CMS, and RSCS control blocks. This appendix also contains information showing which modules reference the data areas or control blocks and, in some cases, gives information on how the data areas or control blocks are created and released.

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 bisync 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 Ccommunications terminal
TYPUNDEF	EQU	X'1C'	terminal device type is undefined
TYPBSC	EQU	X'80'	Bisync 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
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
TYP3284	EQU	X'02'	3284 printer
TYP3286	EQU	TYP3284	3286 printer
TYP3158	EQU	TYP3277	3158 console
FTROPRDR	EQU	X'80'	Operator ID card reader
CLASURI	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
TYPPUN	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
TYPPRT	EQU	X'40'	Printer type device
TYP1403	EQU	X'41'	1403 printer
TYP3211	EQU	X'42'	3211 printer
TYP1443	EQU	X'44'	1443 printer
TYPTP	EQU	X'20'	Tape punch device
TYP1018	EQU	X'24'	1018 paper tape punch

```
Field
Name                           Field Description, Contents, Meaning
-----                          -------------------------------------------
FTRUCS    EQU   X'01'          UCS feature

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        3201 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 MB data module mounted (3340)
FTR70MB   EQU   X'04'           70 MB data module mounted (3340)
FTRRSRL   EQU   X'02'          Reserve/release are valid CCW OP codes

CLASSPEC  EQU   X'02'          Special device class

TYPCTCA   EQU   X'80'          Channel-to-channel adapter
TYP3704   EQU   X'40'          3704 Programmable comm. CTL unit
TYP3705   EQU   TYP3704        3705 Programmable comm. CTL unit
TYPRSV1   EQU   X'02'          Reserved for IBM use
TYPUNSUP  EQU   X'01'          Device unsupported by VM/370

FTRTYP1   EQU   X'10'          Type 1 channel adapter    (3704/5)
FTRTYP2   EQU   X'20'          Type 2 channel adapter    (3704/5)
```

```
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 IO 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 ind.
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
BUSOUT     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 EXTENDED CONTROL REGISTERS

```
Field
Name                               Field Description, Contents, Meaning
-----                              ----------------------------------------
Bits defined in CREG 0
BYTE 0
BLKMPX    EQU   X'80'              Bit 00 - enable block multiplexing
SSMSUPP   EQU   X'40'              Bit 01 - enable SSM suppression

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
CKCMASK   EQU   X'08'              Bit 20 - mask on change clock comparator int.
CPTMASK   EQU   X'04'              Bit 21 - mask on CPU timer int.

BYTE 3
INTMASK   EQU   X'80'              Bit 24 - mask on interval timer int.
KEYMASK   EQU   X'40'              Bit 25 - mask on operator key int.
SIGMASK   EQU   X'20'              Bit 26 - mask on external signals 2-7

Bits defined in CREG  0
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

Bits defined in CREG 14
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
RECOVRPT  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
ASYNELOG  EQU   X'80'              Bit 08 - asynchronous extended logout ctl
ASYNFLOG  EQU   X'40'              Bit 09 - asynchronous fixed logout ctl
```

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 IO operation
IOERETN	EQU	X'10'	Return I/O errors to caller
SYSTEM	EQU	X'08'	Call to DMKPTRAN for system V.M. space

Equates for parm field for calls to DMKBLDRT/DMKBLDRL

DELSEGS	EQU	X'80'	Release the segment tables
DELPAGES	EQU	X'40'	Release the page/swap tables
NEWPAGES	EQU	X'08'	Build new page/swap table
NEWSEGS	EQU	X'04'	Build new segment table
KEEPSEGS	EQU	X'02'	Retain informatin in old segment table
OLDVMSEG	EQU	X'01'	VMSEG pointer in VMBLOK Vvalid

Bits defined for terminal I/O via DMKQCN

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 & drop line after message
LOGHOLD	EQU	X'40'	Output - logoff & 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 auto 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

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

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

Monitor Class and Code Definitions

MNCLPERF EQU   X'00'         Monitor peform class

MNCOSYS  EQU   X'0000'       Peform class, system peformance
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'       INST 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
```

VM/370 REGISTERS

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

Symbolic register equates

R0	EQU	0	
R1	EQU	1	
R2	EQU	2	
R3	EQU	3	
R4	EQU	4	
R5	EQU	5	
R6	EQU	6	
R7	EQU	7	General register definition
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 definiitions
Y2	EQU	2	
Y4	EQU	4	
Y6	EQU	6	
C0	EQU	0	
C1	EQU	1	
C2	EQU	2	
C3	EQU	3	
C4	EQU	4	
C5	EQU	5	
C6	EQU	6	
C7	EQU	7	Control register definitions
C8	EQU	8	
C9	EQU	9	
C10	EQU	10	
C11	EQU	11	
C12	EQU	12	
C13	EQU	13	
C14	EQU	14	
C15	EQU	15	

This appendix lists the control areas used during task processing. Information such as lists of synch 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.

Field Name			Field Description, Contents, Meaning
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)	File tag arrival synch lock address
	DC	A(REQLOCK)	Request arrival synch lock address
	DC	X'80',AL3(CMDLOCK)	Command synch lock address
ARRLOCK	DC	F'0'	File tag arrival synch lock
CMDLOCK	DC	F'0'	Command synch lock
AXSTAKE	DC	0F'0'	TAKE request table
	DC	CL4' '	Giver's task name
	DC	AL1(L'AXSREQ)AL3(AXSREQ)	Pointer to request buffer (input)
	DC	A(AXSRESP)	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.

Field Name			Field Description, Contents, Meaning
REXREX	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(DMTSYSLK)	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 synch lock address
	DC	X'00',AL3(ATTNLOCK)	Console attention synch lock address
	DC	X'40',AL3(REXCONSL)	Console I/O synch lock address
	DC	X'80',AL3(PROGLOCK)	Program check synch lock address
ATTNLOCK	DC	0F'0',X'00',AL3(0)	Console attention lock
REXCONSL	DC	F'0'	Synch lock for I/O operation
	DC	AL2(CONSADDR)	CUU console device address
	DC	AL1(1)	One byte requested on unit check
	DC	AL1(TYP3252)	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'	Unused
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 ID
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.

Field Name			Field Description, Contents, Meaning
CBUFFER	DC	A(0)	Active communications buffer
CFCSOUT	DC	H'0'	Last FCS transmitted to HASP
CFCSSTD	DC	X'88'	Standard FCS
CTEMP	DC	H'0'	Temporary storage
CMAXDUP	DC	H'3'	Maximum repeated blocks
	DC	AL1(0)	First byte of halfword
CBCBCNTO	.DC	AL1(X'80')	Block check count out
	DC	AL1(0)	Spacer
CBCBCNTI	DC	AL1(X'80')	Block count character expected
	DC	H'0'	
CBUFLAST	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 BCB 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

Bits defined in BUFSYNSW

$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
CUWFAKE	EQU	X'08'	Dummy read on for UE recovery
CACKSW	EQU	X'04'	ACK received

ADAECB	DC	F'0'	Synch lock
ADACUU	DC	X'0000',AL1(1),AL1(TYP2700)	
A	DCCWA 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
ADSAV	DC	8F'0'	$SIO register save area

Control Sequences

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 acknowlegement sequence
XNAKSEQ	DC	AL1(XSYN,XNAK)	Negative acknowlegement sequence

Field
Name   Field Description, Contents, Meaning
                      Channel Command Words

                      Normal Data Write with Return Data Read

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

                      Dummy read to turn off lost data sense

CCWD   CCW 2,0,SILI+SKIP,65000   Nonread a punch

                      DISABLE command
CCWOFF   CCW X'2F',0,SIL,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

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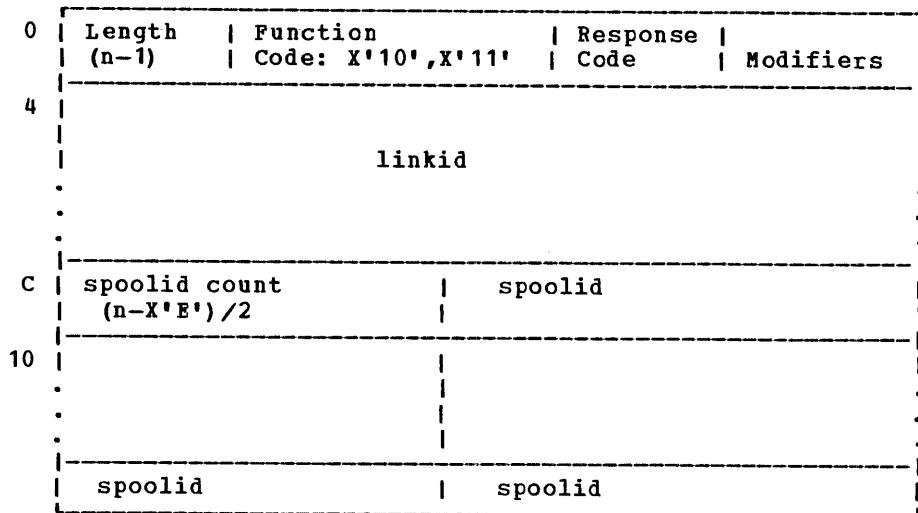
- The name of the module that builds the element.
- The function performed by the element.
- A brief description of the elements usage.
- The format of the element.
- Any operational notes that might be useful in understanding how the element is used.

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.

```
    r---------------------------------------------------------------------------¬
 0  | Length    | Function            | Response |                              |
    | (n-1)     | Code: X'10',X'11'   | Code     | Modifiers                    |
    |---------------------------------------------------------------------------|
 4  |                                                                           |
    |                                                                           |
    |                           linkid                                          |
    |                                                                           |
    .                                                                           .
    .                                                                           .
    .                                                                           .
    |-----------------------------------|---------------------------------------|
 C  | spoolid count                     | spoolid                               |
    | (n-X'E')/2                        |                                       |
    |-----------------------------------|---------------------------------------|
10  |                                   |                                       |
    .                                   |                                       .
    .                                   |                                       .
    .                                   |                                       .
    |-----------------------------------|---------------------------------------|
    | spoolid                           | spoolid                               |
    L---------------------------------------------------------------------------J
```

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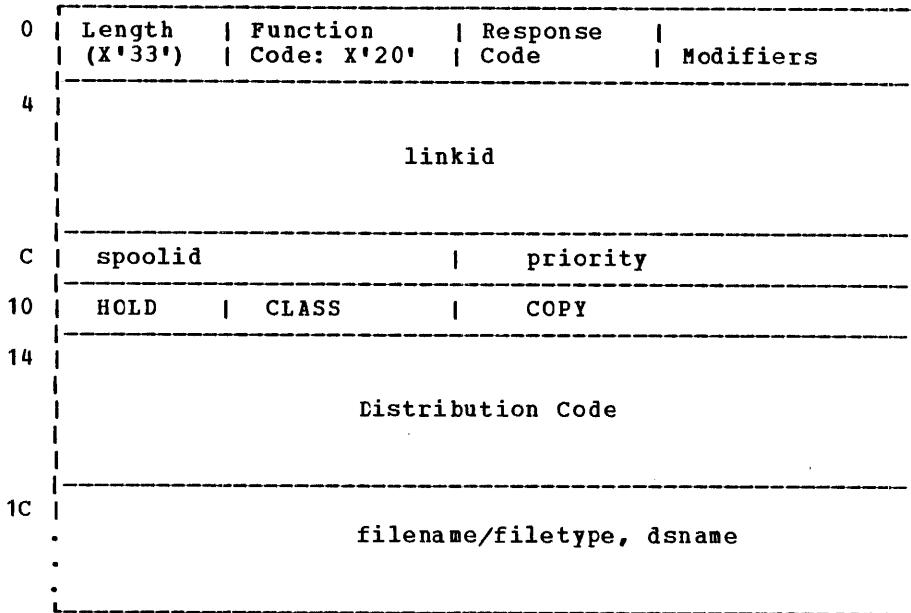
Modifiers:
      X'80' Response messages go to local RSCS operator
      X'40' Purge all files enqueued on the specified link
      X'00' Purge only specified files, response messages go to specified link

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

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 binary priority 0-99
- HOLD X'7F' - set hold status
       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

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
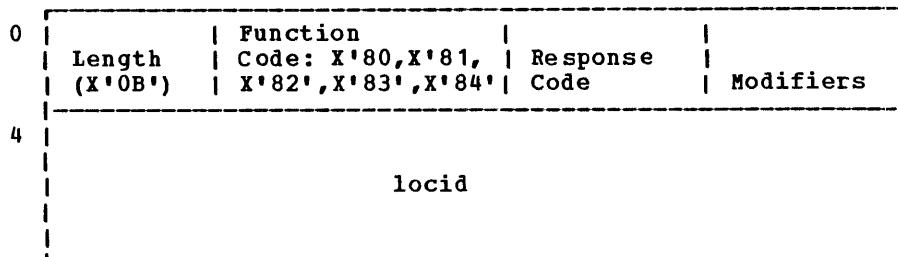
COMMAND ALERT ELEMENT FORMAT LO

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   │             │ Function        │            │              │
    │ Length      │ Code: X'80,X'81,│ Response   │              │
    │ (X'0B')     │ X'82',X'83',X'84'│ Code      │ Modifiers    │
    ├─────────────┴─────────────────┴────────────┴──────────────┤
4   │                                                           │
    │                                                           │
    │                          locid                            │
    │                                                           │
    │                                                           │
    └───────────────────────────────────────────────────────────┘
```

OPERATIONAL NOTES

The locid specifies the location which is to receive response messages. The meaning 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'84'

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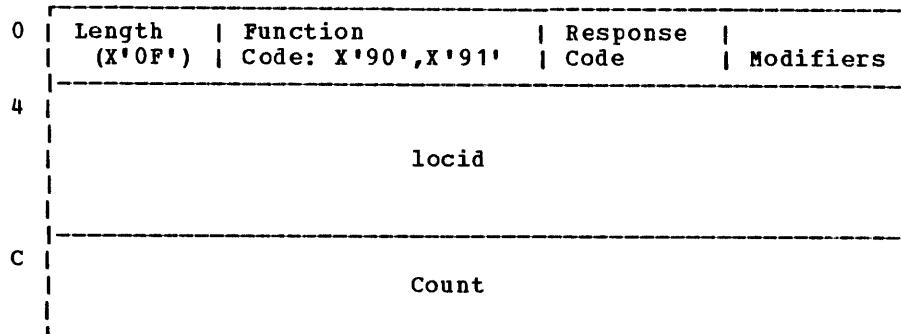
Modifiers:
    X'C0' TRACE end
    X'80' TRACE errors
    X'00' TRACE all

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

OPERATIONAL NOTES

The locid specifies the location which 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
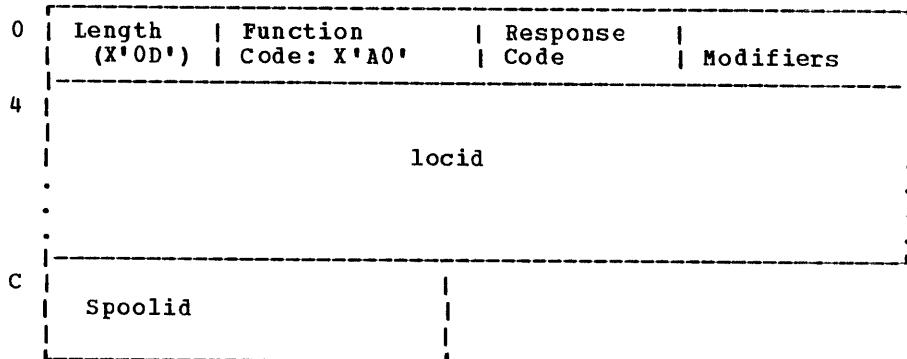
Modifiers: Unused

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 (DMTNPT, DMTSML) to request second-level processing of FLUSH commands.

```
        .---------------------------------------------------------------.
    0   | Length    | Function       | Response    |                     |
        | (X'0D')   | Code: X'A0'    | Code        | Modifiers           |
        |-----------------------------------------------|-------------   |
    4   |                                                               |
        |                                                               |
        |                            locid                              |
        .                                                               .
        .                                                               .
        .                                                               .
        |------------------------------------------------------------'  |
    C   |                                  |                             |
        | Spoolid                          |
        |                                  |
        ._____.
```

OPERATIONAL NOTES

The locid specifies the location which 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 fields follow.

FLUSH Command

Function Code: X'A0'

Response Codes:
    X'00' Element accepted for processing
    X'10' Element rejected, busy
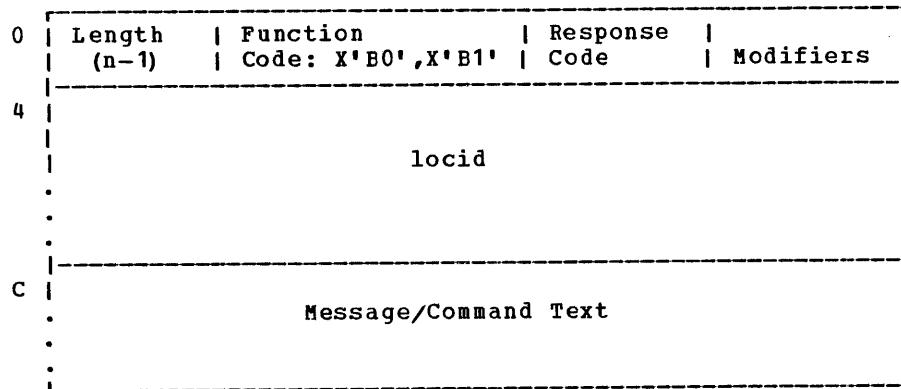
Modifiers:
    X'80' Flush all copies, purge file
    X'40' Flush hold, keep file, do not decrement copy count
    X'00' Flush, decrement copy count, purge file if no copy count remains

COMMAND ALERT ELEMENT FORMAT L3 (ALSO MESSAGE ALERT ELEMENT)

BUILT BY: DMTCMX, 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.

```
  ┌───────────────────────────────────────────────────────────┐
0 │ Length    │ Function          │ Response   │              │
  │ (n-1)     │ Code: X'B0',X'B1' │ Code       │ Modifiers    │
  ├───────────────────────────────────────────────────────────┤
4 │                                                            │
  │                                                            │
  │                        locid                               │
  .                                                            .
  .                                                            .
  .                                                            .
  ├───────────────────────────────────────────────────────────┤
C │                                                            │
  .                   Message/Command Text                     .
  .                                                            .
  .                                                            .
  └───────────────────────────────────────────────────────────┘
```

OPERATIONAL NOTES

The locid specifies the location which is to receive the message or command text. The meanings of the other fields are listed below.

CMD Command

Function Code: X'B0'

Response Codes:
    X'00' Element accepted for processing
    X'10' Element rejected, busy

Modifiers: Unused

MSG Command

Function Code: X'B1'

Response Codes:
    X'00' Element accepted for processing
    X'10' Element rejected, busy
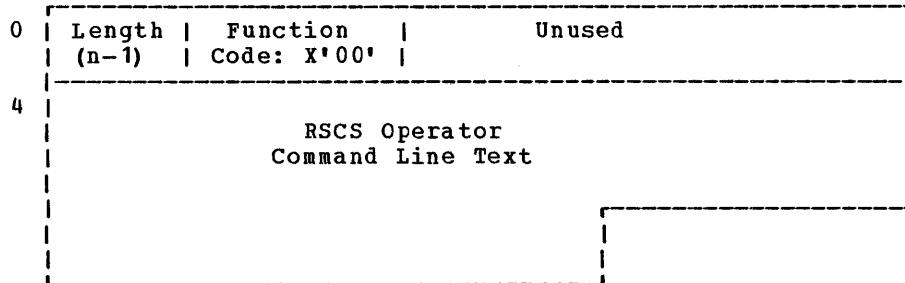
Modifiers: One-byte binary RSCS severity code

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.

```
        r--------------------------------------------------------------¬
    0   | Length  |  Function    |              Unused                 |
        | (n-1)   |  Code: X'00' |                                     |
        |--------------------------------------------------------------|
    4   |                                                              |
        |                     RSCS Operator                            |
        |                   Command Line Text                          |
        |                                                              |
        |                                          r-------------------J
        |                                          |
        |                                          |
        L------------------------------------------J
```

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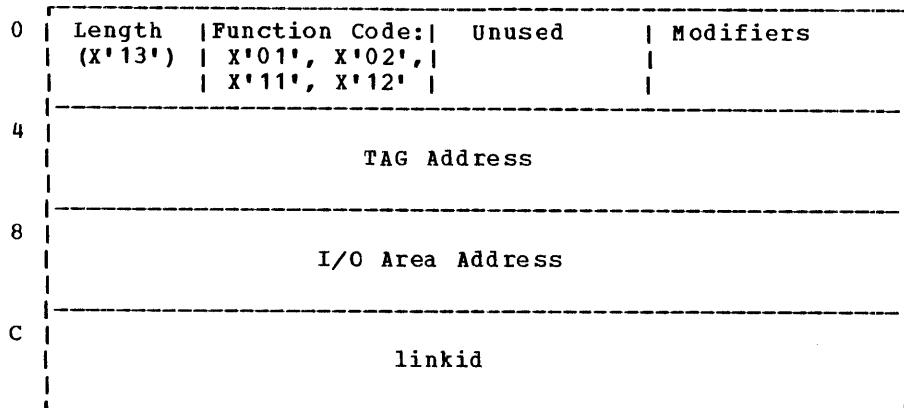
No response text is returned.   Command responses are distributed via DMTMGX.

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.

```
     r--------------------------------------------------------------¬
   0 | Length    |Function Code:|  Unused       | Modifiers        |
     | (X'13')   | X'01', X'02',|               |                  |
     |           | X'11', X'12' |               |                  |
     |-----------------------------------------------------------|
   4 |                                                           |
     |                      TAG Address                          |
     |                                                           |
     |-----------------------------------------------------------|
   8 |                                                           |
     |                    I/O Area Address                       |
     |                                                           |
     |-----------------------------------------------------------|
   C |                                                           |
     |                       linkid                              |
     |                                                           |
     L-----------------------------------------------------------+
```

OPERATIONAL NOTES

The use and meaning of the various field 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.

Open Input

Function Code: X'01'

Modifiers: Unused

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

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

linkid: Request field which 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
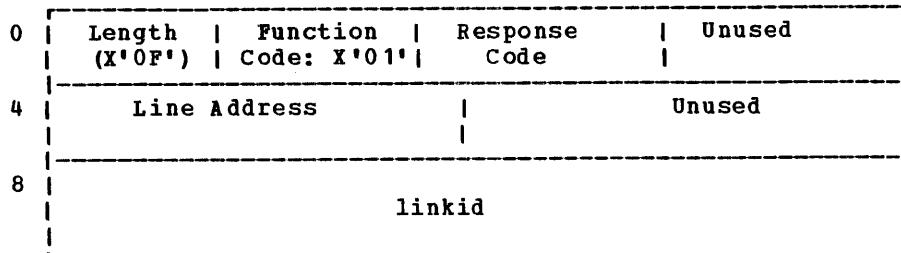
Response Post Codes:
     X'04' I/O area not found, close failed

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      │ Function     │ Response     │ Unused             │
        │ (X'0F')     │ Code: X'01'  │ Code         │                    │
        ├─────────────┴──────────────┼──────────────┴────────────────────┤
    4   │ Line Address               │           Unused                  │
        │                            │                                   │
        ├────────────────────────────┴───────────────────────────────────┤
    8   │                                                                 │
        │                          linkid                                 │
        │                                                                 │
        └─────────────────────────────────────────────────────────────────┘
```

OPERATIONAL NOTES

The use and meaning of the fields are described below. Certain fields are updated during
processing.

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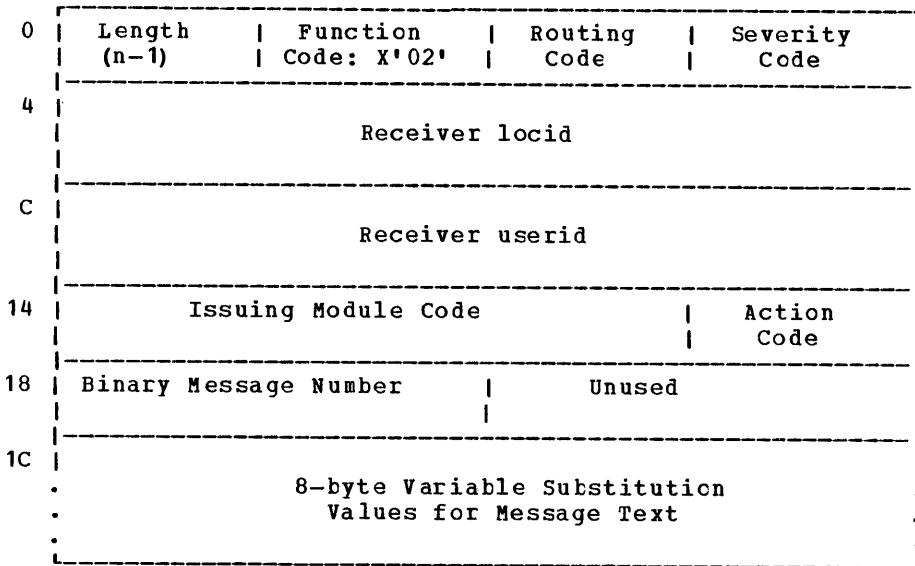
linkid: Response field specifying the ID of the link which has reserved the particular
    requested line address (with response code X'02').

MESSAGE REQUEST ELEMENT

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

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       | Function      | Routing      | Severity            |
     | (n-1)        | Code: X'02'   | Code         | 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 NCTES

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/37C operator
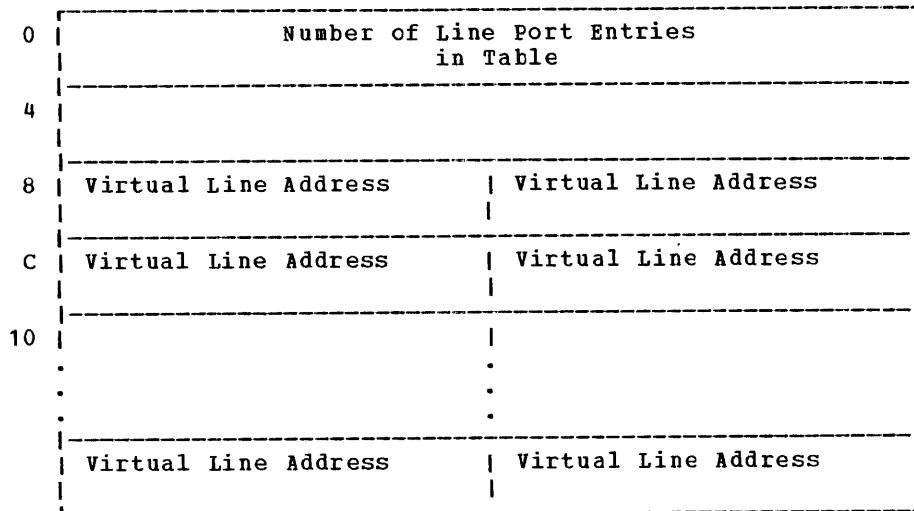
    No response text is returned.

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.

This appendix contains assembler language equate symbols used in CMS to reference data for:

• CMS usage
• CMS registers

```
Field
Name                                     Field Description, Contents, Meaning
-----                                    ----------------------------------------
Bits defined in the program status word (PSW)
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
DOFM       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

Bits defined in the 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-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
```

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

Common channel command codes

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
RDDATA    EQU   X'06'              Read data
SEEK      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
SETSEC    EQU   X'23'              Set csector
SEARCH    EQU   X'31'              Search ID equal


Bits defined in a 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
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

Field Name			Field Description, Contents, Meaning

General purpose registers

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

Floating point registers

F0	EQU	0
F2	EQU	2
F4	EQU	4
F6	EQU	6

Extended control registers

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

This appendix contains lists of CP, CMS, and RSCS control blocks. The lists contain information showing which modules reference the data areas or control blocks and, in some cases, information on how the data areas or control blocks are created and released.

ACCTBLOK

    Built by: DMKHVD

    Released by: DMKHVD, DMKUSO

    Referenced by: DMKHVC, DMKHVD, DMKSPL

ACNTBLOK

    Built by: DMKACO, DMKHVD, DMKWRM

    Released by: DMKACO

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

ALOCBLOK

    Built by: DMKCPI, DMKVDB

    Released by: DMKCPI, DMKVDB

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

ESCBLOK

    Built by: DMKRGB

    Released by: DMKRGA

    Referenced by: DMKBSC, DMKRGF

BUFFER

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

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

    Referenced by: DMKCDB, DMKCFG, DMKCFM, DMKCFP, DMKCFS, DMKCPI, DMKCSO, DMKCSP, DMKCST, DMKCSU, DMKERM, DMKGRF, DMKLNK, DMKLOG, DMKMSG, DMKRGF, DMKRSP, DMKSCN

CCHREC

    Built by: DMKCCH

    Released by: DMKCCH

    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, DMKVIO

CKPELOK

    Built by: DMKRNH

    Released by: DMKRNH

    Referenced by: DMKRNH, DMKWRM

CONTASK

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

    Released by:

    Referenced by: DMKCNS, DMKEDM, DMKGRF, DMKMON, DMKNES, DMKQCN, DMKRGF, DMKRNH

CORTABLE

    Assembled in DMKSYS.

    Released by: N/A

    Referenced by: DMKBLD, DMKCCW, DMKCDS, DMKCFS, DMKCPI, DMKCPV, DMKDGD, DMKDMP, DMKEDM, DMKFRE, DMKMCC, DMKMCH, DMKMON, DMKPAG, DMKPGS, DMKPSA, DMKPTR, DMKRPA, DMKSCH, DMKUDR, DMKUNT, DMKVMA

CPEXBLOK

    Built by: DMKACO, DMKCDS, DMKCFM, DMKCPS, DMKCPV, DMKDIA, DMKGRF, DMKIOE, DMKIOF, DMKIOG, DMKIOS, DMKLOC, DMKMCC, DMKMCH, DMKMON, DMKPGT, DMKPTR, DMKQCN, DMKRGA, DMKRGB, DMKRNH, DMKRPA, DMKRSP, DMKSPL, DMKUSO, DMKVCA, DMKVDB, DMKVMA

    Released by: DMKDSP, DMKIOF, DMKMON, DMKPTR, DMKVDB

    Referenced by: DMKACO, DMKCDS, DMKCFM, DMKCPV, DMKDIA, DMKDSP, DMKGRF, DMKIOE, DMKIOF, DMKIOS, DMKLOC, DMKMCH, DMKMON, DMKPAG, DMKPGS, DMKPGT, DMKPTR, DMKQCN, DMKRGF, DMKRNH, DMKRPA, DMKRSP, DMKSPL, DMKSTK, DMKUSO, DMKVCA, DMKVDB, DMKVMA, DMKVSP

DDRREC

    Built by: DMKVER

    Released by: DMKVER

    Referenced by: DMKVER

DMPBREC

    Built by: DMKDMP

    Released by: DMKDMP

    Referenced by: DMKDMP, DMKEDM

DMPINREC

    Built by: DMKDMP

    Released by: DMKDMP

    Referenced by: DMKDMP, DMKEDM

DMPKYREC

    Built by: DMKDMP

    Released by: DMKDMP

    Referenced by: DMKDMP

ECBLOK

    Built by: DMKBLD

    Released by: DMKCFS, DMKUSO

    Referenced by: DMKBLD, DMKCDB, DMKCDS,
DMKCFG, DMKCFP, DMKCFS, DMKDSP, DMKEDM,
DMKLOG, DMKPRG, DMKPRV, DMKSCH, DMKTMR,
DMKTRC, DMKUSO, DMKVAT

ERRBLOK

    Built by: DMKIOE

    Released by: DMKIOF

    Referenced by: DMKIOF

IOBLOK

    Built by: DMKACO, DMKCCW, DMKCFP,
DMKCNS, DMKCPB, DMKCPI, DMKCPS, DMKCSO,
DMKCSP, DMKCSU, DMKDGD, DMKDIA, DMKGIO,
DMKGRF, DMKHVC, DMKIOS, DMKNLD, DMKRGA,
DMKRGB, DMKSPL,DMKTDK, DMKVCA, DMKVDR,
DMKVIO

    Released by: DMKCFP, DMKCNS, DMKCPB,
DMKCPI, DMKCPS, DMKCSO, DMKDAS, DMKDGD,
DMKDIA, DMKGIO, DMKGRF, DMKHVC, DMKIOS,
DMKMON, DMKNLD, DMKPAG, DMKRGA, DMKRGB,
DMKRNH, DMKRSP, DMKSEP, DMKTDK, DMKVCA,
DMKVIO

    Referenced by: DMKACO, DMKBSC, DMKCCH,
DMKCCW, DMKCFP, DMKCNS, DMKCPB, DMKCPI,
DMKCPV, DMKCSO, DMKCSP, DMKCSU, DMKDAS,
DMKDGD, DMKDIA, DMKDSP, DMKEDM, DMKGIO,
DMKGRF, DMKHVC, DMKIOE, DMKIOS, DMKISM,
DMKLOG, DMKMCC, DMKMON, DMKMSW, DMKNLD,
DMKPAG, DMKPGT, DMKRGF, DMKRNH, DMKRSE,
DMKRSP, DMKSEP, DMKSPL, DMKTAP, DMKTDK,
DMKTRC, DMKUDR, DMKUNT, DMKVCA, DMKVDB,
DMKVDR, DMKVIO

IOBR3211

    Built by: DMKRSE

    Released by: DMKIOE

    Referenced by: DMKIOE, DMKRSE

IOERBLOK

    Built by: DMKBSC, DMKCCH, DMKDAS,
DMKDIA, DMKIOE, DMKIOS, DMKRSE, DMKTAP,
DMKVCA

    Released by: DMKBSC, DMKCCH, DMKCCW,
DMKCFP, DMKCNS, DMKCPS, DMKDAS, DMKDGD,
DMKDIA, DMKGIO, DMKGRF, DMKIOE, DMKIOS,
DMKMON, DMKNLD, DMKRGA, DMKRGB, DMKRNH,
DMKRSE, DMKRSR, DMKTAP, DMKVIO

    Referenced by: DMKBSC, DMKCCH, DMKCCW,
DMKCFP, DMKCNS, DMKCPV, DMKDAS, DMKDGD,
DMKDIA, DMKEDM, DMKEIG, DMKGIO, DMKGRF,
DMKIOE, DMKIOS, DMKMSW, DMKNLD, DMKRGF,
DMKRNH, DMKRSE, DMKRSP, DMKSEV, DMKSIX,
DMKTAP, DMKUNT, DMKVCA, DMKVIO

IRMBLOK

    Built by: DMKCFS

    Released by: DMKCFS, DMKIOE

    Referenced by: DMKCFS, DMKIOE

LOCKBLOK

    Built by: DMKLOC

    Released by: DMKLOC

    Referenced by: DMKLOC

MCHAREA

    Built by: DMKMCH

    Released by: N/A

    Referenced by: DMKCCH, DMKCFS, DMKIOG,
DMKMCH

MCRECORD

    Built by: DMKMCH

    Released by: N/A

    Referenced by: DMKMCH

MDRREC

    Built by: DMKVER

    Released by: DMKVER

    Referenced by: DMKVER

MICBLOK

    Built by: DMKCFS, DMKLOG

    Released by: DMKCFS, DMKLOG, DMKUSO

    Referenced by: DMKBLD, DMKCFS, DMKDSP, DMKLOG

MIHREC

    Built by: DMKVER

    Released by: DMKVER

    Referenced by: DMKVER

MNHDR

    Built by: DMKMON

    Released by: DMKMON

    Referenced by: DMKMON

MN000

    Built by: DMKMON

    Released by: DMKMON

    Referenced by: DMKMON

MN097

    Built by: DMKMON

    Released by: DMKMON

    Referenced by: DMKMON

MN098

    Built by: DMKMON

    Released by: DMKMON

    Referenced by: DMKMON

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

MN600DEV

    Built by: DMKMON

    Released by: DMKMON

    Referenced by: DMKMON

MN600HDR

    Built by: DMKMON

    Released by: DMKMON

    Referenced by: DMKMON

MN700

    Built by: DMKMON

    Released by: DMKMON

    Referenced by: DMKMON

MN802CTR

    Built by: DMKMON

    Released by: DMKMON

    Referenced by: DMKMON

MONCOM

    Built by: DMKMCC

    Released by: DMKMON

    Referenced by: DMKCPV, DMKDMP, DMKMCC,
DMKMON

NCPTBL

    Built by: DMKSNT

    Released by: N/A

    Referenced by: DMKNLD, DMKSNC

NICBLOK

    Built by: DMKNLD

    Released by: DMKNLD

    Referenced by: DMKBLD, DMKCFT, DMKCKP,
DMKCPI, DMKCQR, DMKHVD, DMKDIA, DMKHVD,
DMKLOG, DMKNES, DMKNET, DMKNLD, DMKPSA,
DMKQCN, DMKRGF, DMKRNH, DMKWRM

OBRRECN (Long OBR)

    Built by: DMKIOF

    Released by: DMKIOF

    Referenced by: DMKIOE, DMKIOF, DMKVER

OBRREC (Short OBR)

    Built by: DMKIOF

    Released by: DMKIOF

    Referenced by: DMKIOF

OWNDLIST

    Assembled into DMKSYS

    Referenced by: DMKCKP, DMKCKS, DMKCPI,
DMKDRD, DMKPAG, DMKPGT, DMKPTR, DMKSPL,
DMKUDR, DMKVDB, DMKWRM

PAGTABLE

    Built by: DMKBLD

    Released by: DMKBLD, DMKPGS

    Referenced by: DMKBLD, DMKCFG, DMKCFP,
DMKPGS, DMKVMA

PGBLOK

    Built by: DMKVAT

    Released by: DMKCFP, DMKDSP

    Referenced by: DMKCFP, DMKDSP, DMKVAT

PSA

    Assembled as part of DMKSYS; part of CP
nucleus.

    Referenced by: DMKACO, DMKBLD, DMKBSC,
DMKCCH, DMKCCW, DMKCDB, DMKCDS, DMKCFD,
DMKCFG, DMKCFM, DMKCFP, DMKCFS, DMKCFT,
DMKCKP, DMKCKS, DMKCNS, DMKCPB, DMKCPI,
DMKCPV, DMKCQG, DMKCQP, DMKCSO, DMKCSP,
DMKCST, DMKCSU, DMKCVT, DMKDAS, DMKDEF,
DMKDGD, DMKDIA, DMKDMP, DMKDRD, DMKDSP,
DMKEDM, DMKEIG, DMKERM, DMKFMT, DMKFRE,
DMKGIO, DMKGRF, DMKHVC, DMKIOE, DMKIOF,
DMKIOG, DMKIOS, DMKISM, DMKLNK, DMKLOC,
DMKLOG, DMKMCC, DMKMCH, DMKMID, DMKMON,
DMKMSG, DMKMSW, DMKNES, DMKNET, DMKNLD,
DMKOPR, DMKPAG, DMKPGS, DMKPGT, DMKPRG,
DMKPRV, DMKPTR, DMKQCN, DMKRGF, DMKRNH,
DMKRPA, DMKRSE, DMKRSP, DMKSAV, DMKSCH,
DMKSCN, DMKSEP, DMKSEV, DMKSIX, DMKSNC,
DMKSPL, DMKSSP, DMKSTK, DMKTAP, DMKTDK,
DMKTHI, DMKTMR, DMKTRA, DMKTRC, DMKTRM,
DMKUDR, DMKUNT, DMKUSO, DMKVAT, DMKVCA,
DMKVCH, DMKVCN, DMKVDB, DMKVDR, DMKVDS,
DMKVER, DMKVIO, DMKVMA, DMKVMI, DMKVSP,
DMKWRM

RCHBLOK

    Assembled into CP nucleus module
DMKRIO.

    Released by: DMKRIO

    Referenced by: DMKCCH, DMKCFP, DMKCKP,
DMKCPB, DMKCPI, DMKCPV, DMKCQP, DMKDIA,
DMKEDM, DMKIOG, DMKIOS, DMKMON, DMKNES,
DMKSCN, DMKSSP, DMKVCH

RCUBLOK

    Assembled into CP nucleus module
DMKRIO.

    Released by: DMKRIO

    Referenced by: DMKCCH, DMKCFD, DMKCKP,
DMKCPB, DMKCPI, DMKCPV, DMKCOP, DMKDIA,
DMKEDM, DMKIOF, DMKIOS, DMKMON, DMKNES,
DMKNLD, DMKSCN, DMKSSP, DMKVCH

RCWTASK

    Built by: DMKCCW

    Released by: DMKCCW, DMKUNT

    Referenced by: DMKCCW, DMKHVC, DMKISM,
DMKTRC, DMKUNT

RDEVBLOK

    Built by:

    Released by:

    Referenced by: DMKACO, DMKBLD, DMKBSC, DMKCCH, DMKCCW, DMKCFC, DMKCFD, DMKCFG, DMKCFM, DMKCFP, DMKCFS, DMKCFT, DMKCKP, DMKCKS, DMKCNS, DMKCPB, DMKCPI, DMKCPV, DMKCQG, DMKCQP, DMKCQR, DMKCSO, DMKDAS, DMKDEF, DMKDIA, DMKDMP, DMKDRD, DMKEDM, DMKGRF, DMKHVC, DMKHVD, DMKIOE, DMKIOF, DMKIOG, DMKIOS, DMKLNK, DMKLOG, DMKMCC, DMKMON, DMKMSW, DMKNES, DMKNET, DMKNLD, DMKOPR, DMKPAG, DMKPGT, DMKPSA, DMKPTR, DMKQCN, DMKRGF, DMKRNH, DMKRSE, DMKRSP, DMKSCN, DMKSEP, DMKSNC, DMKSPL, DMKSSP, DMKTAP, DMKTDK, DMKTRM, DMKUNT, DMKUSO, DMKVCH, DMKVCN, DMKVDB, DMKVDR, DMKVDS, DMKVER, DMKVIO, DMKWRM

RECBLOK

    Built by: DMKCKS, DMKCPI, DMKPGT, DMKRSP, DMKVSP, DMKWRM

    Released by: DMKPGT, DMKSPL, DMKUSO

    Referenced by: DMKCKP, DMKCKS, DMKCPI, DMKDMP, DMKEDM, DMKPGT, DMKRSP, DMKSPL, DMKVSP, DMKWRM

RECPAG

    Built by: DMKIOF

    Released by: DMKIOF

    Referenced by: DMKIOF, DMKIOG

RSPLCTL

    Built by: DMKRSP

    Released by: DMKRSP

    Referenced by: DMKCKP, DMKCQP, DMKCSO, DMKEDM, DMKRSP, DMKSPL

SAVEAREA

    Built by: DMKCPI, DMKPSA

    Released by: DMKPSA

    Referenced by: DMKACO, DMKBLD, DMKBSC, DMKCCH, DMKCCW, DMKCDB, DMKCDS, DMKCFC, DMKCFD, DMKCFG, DMKCFM, DMKCFP, DMKCFS, DMKCFT, DMKCKS, DMKCNS, DMKCPB, DMKCPV, DMKCQG, DMKCQP, DMKCQR, DMKCSO, DMKCSP, DMKCST, DMKCSU, DMKDAS, DMKDEF, DMKDGD, DMKDIA, DMKDRD, DMKEIG, DMKERM, DMKGIO, DMKGRF, DMKHVD, DMKIOE, DMKIOF, DMKIOG, DMKIOS, DMKISM, DMKLNK, DMKLOG, DMKMCC, DMKMCH, DMKMID, DMKMON, DMKMSG, DMKMSW,

DMKNEM, DMKNET, DMKNLD, DMKPGS, DMKPSA, DMKPTR, DMKQCN, DMKRGF, DMKRNH, DMKRPA, DMKRSE, DMKRSP, DMKSEP, DMKSEV, DMKSIX, DMKSNC, DMKSPL, DMKSSP, DMKTAP, DMKTDK, DMKTHI, DMKTRA, DMKTRC, DMKTRM, DMKUDR, DMKUNT, DMKUSO, DMKVAT, DMKVCA, DMKVCH, DMKVDB, DMKVDR, DMKVDS, DMKVER, DMKVMA, DMKVSP, DMKWRM

SAVTABLE

    Built by:

    Released by:

    Referenced by: DMKCFG, DMKCFP

SDRBLOK

    Built by: DMKIOF

    Released by: DMKIOE

    Referenced by: DMKIOE, DMKIOF

SEGTABLE

    Built by: DMKBLD

    Released by: DMKBLD

    Referenced by: DMKBLD, DMKCPI, DMKPGS, DMKVMA

SFBLOK

    Built by: DMKCKS, DMKNLD, DMKSPL, DMKWRM

    Released by: DMKCKS, DMKRSP, DMKSPL, DMKUSO

    Referenced by: DMKCKP, DMKCKS, DMKCPI, DMKCQG, DMKCQR, DMKCSO, DMKCSP, DMKCST, DMKCSU, DMKDMP, DMKDRD, DMKEDM, DMKNLD, DMKRSP, DMKSEP, DMKSPL, DMKUSO, DMKVSP, DMKWRM

SHQBLOK

    Built by: DMKCSP, DMKWRM

    Released by: DMKCSP

    Referenced by: DMKCKS, DMKCQR, DMKCSP, DMKSPL, DMKWRM

SHRTABLE

    Built by: DMKCFG

    Released by: DMKPGS, DMKVMA

    Referenced by: DMKCFG, DMKCFP, DMKPGS, DMKVMA

SPLINK

Assembled into CP nucleus module DMKSYS.

Referenced by: DMKCKS, DMKDRD, DMKRSP, DMKSPL, DMKVSP

SWPTABLE

Built by: DMKBLD, DMKVMA

Released by: DMKBLD

Referenced by: DMKBLD, DMKCFG, DMKEDM, DMKPGS, DMKVMA

SYSLOCS

Assembled into CP nucleus module DMKSYS.

Referenced by: DMKACO, DMKBLD, DMKCFS, DMKCFT, DMKCKP, DMKLOC, DMKLOG, DMKUDR, DMKUSO

SYSTBL

Assembled into DMKSNT.

Referenced by: DMKCFG, DMKCFP

TNSREC

Built by: DMKIOF

Released by: DMKIOF

Referenced by: DMKIOF

TREXT

Built by: DMKTRA

Released by: DMKTRA, DMKTRC, DMKUSO

Referenced by: DMKCFG, DMKCFM, DMKCFP, DMKDSP, DMKEDM, DMKPGS, DMKPRG, DMKPRV, DMKPSA, DMKTMR, DMKTRA, DMKTRC, DMKVIO

TRQBLOK

Built by: DMKBLD, DMKCFC, DMKCFS, DMKCPI, DMKMCC, DMKQCN

Released by: DMKCFM, DMKCFS, DMKMCC, DMKMON, DMKQCN, DMKUSO

Referenced by: DMKBLD, DMKCDS, DMKCFC, DMKCFM, DMKCFP, DMKCFS, DMKCPI, DMKGRF, DMKLOG, DMKMCC, DMKMID, DMKMON, DMKPSA, DMKQCN, DMKRGF, DMKSCH, DMKTMR, DMKUSO

UDBFBLOK

Built by: DMKDEF, DMKHVD, DMKSPL

Released by: DMKDEF, DMKHVD, DMKSPL

Referenced by: DMKDEF, DMKHVC, DMKLNK, DMKLOG, DMKSPL, DMKUDR

UDEVBLOK

Built by: DMKCSP, DMKUDR

Released by: DMKCSP, DMKUDR

Referenced by: DMKDEF, DMKDIR, DMKHVD, DMKLNK, DMKLOG, DMKSCN, DMKUDR, DMKVDB, DMKVDS

UDIRBLOK

Built by: DMKCSP

Released by: DMKCSP

Referenced by: DMKCSP, DMKDEF, DMKDIR, DMKHVC, DMKHVD, DMKLNK, DMKLOG, DMKSPL, DMKUDR

UMACBLOK

Built by: DMKDIR

Released by: DMKDIR

Referenced by: DMKDEF, DMKDIR, DMKHVC, DMKHVD, DMKLOG, DMKSPL, DMKUDR

VCHBLOK

Built by: DMKVDS

Released by: DMKUSO

Referenced by: DMKCFM, DMKCFP, DMKCKP, DMKCPB, DMKCPV, DMKCQS, DMKCSP, DMKCSU, DMKDEF, DMKDGD, DMKDIA, DMKDSP, DMKEDM, DMKGIO, DMKLNK, DMKLOG, DMKPRV, DMKSCN, DMKSPL, DMKUSO, DMKVCH, DMKVCN, DMKVDB, DMKVDS, DMKVIO, DMKVSP

VCONCTL

Built by: DMKVDS

Released by: DMKVDR

Referenced by: DMKCFP, DMKGRF, DMKLOG, DMKRGF, DMKVCN, DMKVDR

VCUBLOK

    Built by: DMKVDS

    Released by: DMKUSO

    Referenced by: DMKCFM, DMKCFP, DMKCKP,
    DMKCPB, DMKCPV, DMKCQG, DMKCSP, DMKCSU,
    DMKDEF, DMKDIA, DMKDSP, DMKEDM, DMKLOG,
    DMKNLD, DMKPRJ, DMKSCN, DMKSPL, DMKUSO,
    DMKVCH, DMKVCN, DMKVDB, DMKVDS, DMKVIO,
    DMKVSP

VDEVBLOK

    Built by: DMKLOG, DMKVDS

    Released by: DMKUSO

    Referenced by: DMKACO, DMKCCW, DMKCFG,
    DMKCFM, DMKCFP, DMKCKP, DMKCPB, DMKCPV,
    DMKCQG, DMKCQP, DMKCSO, DMKCSP, DMKCST,
    DMKCSU, DMKDEF, DMKDGD, DMKDIA, DMKDRD,
    DMKDSP, DMKEDM, DMKGIO, DMKGRF, DMKHVC,
    DMKHVD, DMKIOS, DMKLNK, DMKLOG, DMKNLD,
    DMKQCN, DMKRGF, DMKSCN, DMKSPL, DMKTHI,
    DMKTRC, DMKUNT, DMKUSO, DMKVCA, DMKVCH,
    DMKVCN, DMKVDB, DMKVDR, DMKVDS, DMKVER,
    DMKVIO, DMKVSP

VFCBBLOK

    Built by: DMKCSO, DMKCFG

    Released by: DMKVDR

    Referenced by: DMKCSO, DMKVSP

VMABLOK

    Built by: DMKBLD, DMKCFG

    Released by: DMKBLD, DMKPGS, DMKVMA

    Referenced by: DMKCFG, DMKPGS, DMKVMA

VMBLOK

    Built by: DMKBLD

    Released by: DMKBLD, DMKDIA, DMKLOG,
    DMKUSO

    Referenced by: DMKACO, DMKBLD, DMKXCCH
    DMKXCCW DMKCDB, DMKCDS, DMKCFS, DMKCFD,
    DMKCFG, DMKCFM, DMKCFP, DMKCFS, DMKCPT,
    DMKCKP, DMKCKS, DMKCNS, DMKCPB, DMKCPI,
    DMKCPV, DMKCQG, DMKCQP, DMKCQR, DMKCSO,
    DMKCSP, DMKCST, DMKCSU, DMKDEF, DMKDGD,
    DMKDIA, DMKDRD, DMKDSP, DMKEDM, DMKERM,
    DMKGIO, DMKGRF, DMKHVC, DMKHVD, DMKIOE,
    DMKIOG, DMKIOS, DMKISM, DMKLNK, DMKLOG,
    DMKMCC, DMKMCH, DMKMID, DMKMON, DMKMSG,
    DMKMSW, DMKNES, DMKNET, DMKNLD, DMKPAG,
    DMKPER, DMKPGS, DMKPGT, DMKPRG, DMKPRV,
    DMKPSA, DMKPTR, DMKQCN, DMKRGF, DMKRNH,
    DMKRPA, DMKRSE, DMKRSP, DMKSCH, DMKSCN,
    DMKSEP, DMKSNC, DMKSPL, DMKTHI, DMKTMR,
    DMKTRA, DMKTRC, DMKUDR, DMKUNT, DMKUSO,
    DMKVAT, DMKVCA, DMKVCH, DMKVCN, DMKVDB,
    DMKVDR, DMKVDS, DMKVER, DMKVIO, DMKVMA,
    DMKVSP

VSPLCTL

    Built by: DMKDRD, DMKVSP

    Released by: DMKVSP

    Referenced by: DMKCKP, DMKCSP, DMKDRD,
    DMKEDM, DMKSPL, DMKVSP

VSPXBLOK

    Built by: DMKCST

    Released by: DMKCST

    Referenced by: DMKCKP, DMKCQG, DMKCSP,
    DMKCST, DMKSPL, DMKVDR

XINTBLOK

    Built by: DMKCFP, DMKCPB, DMKDSP,
    DMKSGRF, DMKRGA, DMKSCH, DMKTMR

    Released by: DMKCFP, DMKDSP, DMKSCH,
    DMKTRM

    Referenced by: DMKCFP, DMKCPB, DMKDSP,
    DMKSCH, DMKTRM

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, DMSESC, DMSBWR, DMSCMP, DMSCPY, DMSDIO, DMSDLB, DMSDLK, DMSDSK, DMSDSL, DMSERS, DMSEXC, DMSEXT, DMSFOR, DMSHSM, DMSINS, DMSLAD, DMSLAF, DMSLBM, DMSLDS, DMSLFS, DMSLKD, DMSLLU, DMSLST, DMSMVE, DMSPUN, DMSQRY, DMSRNM, DMSROS, DMSSET, DMSSOP, DMSSTT, DMSTQQ, DMSTRK, DMSUPD, DMSXCP

AFTSECT

Assembled as part of DMSNUC; also created and released dynamically by DMSLAF.

Referenced by: DMSBRD, DMSBWR, DMSERS, DMSINT, DMSLAF, DMSPNT, DMSRNM, DMSSOP, DMSSTT, DMSTPE

ANCHSECT

Built by: DMSSTG

Released by: Not released

Referenced by: DMSDOS, DMSSMN

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, DMSBOP, DMSCLS, DMSDLB, DMSDLK, DMSDMP,

DMSDOS, DMSDSV, DMSFCH, DMSFET, DMSINS, DMSITP, DMSLLU, DMSOPL, DMSOPT, DMSPRV, DMSRRV, DMSSET, DMSSMN, DMSSRV, DMSVSR, DMSXCP

CMSTAXE

Built by: DMSSVT

Released by: DMSSVT

Referenced by: DMSCIT, 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, DMSEDX, DMSINI

DIOSECT

Assembled as part of DMSNUC.

Referenced by:

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, 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, DMSSMN, DMSSVN,
DMSSVT

EXTUAREA

　　Assembled as part of DMSNUC.

　　No CMS references.

FCBSECT

　　Built by: DMSFLD

　　Released by: DMSFLD, DMSABN

　　Referenced by: DMSALU, DMSARN, DMSARX,
DMSASM, DMSDSL, DMSFCH, DMSFLD, DMSHSM,
DMSLDS, DMSMVE, DMSQRY, DMSROS, DMSSAB,
DMSSBD, DMSSBS, DMSSCT, DMSSEB, 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

FSTD

　　Built by: N/A

　　Released by: N/A

　　Referenced by: DMSCPY, DMSEDX, DMSEXC,
DMSGND, DMSNCP, DMSTPE

FSTSECT

　　Built by: DMSACF

　　Released by: DMSALU

　　Referenced by: DMSACF, DMSAMS, DMSARN,
DMSARX, DMSASM, DMSBOP, DMSBRD, DMSBSC,
DMSBWR, DMSCPY, DMSDLK, DMSDSK, DMSDSL,
DMSERS, DMSHSM, DMSLAF, DMSLBM, DMSLKD,
DMSMDP, DMSMVE, DMSRNM, DMSSTT, DMSTPE,
DMSUPD, DMSZAP

FVSECT

　　Assembled as part of DMSNUC.

　　Referenced by: DMSABN, DMSACC, DMSACF,
DMSACM, DMSALU, DMSAUD, DMSBTB, DMSBTP,
DMSBWR, DMSCIT, DMSCRD, DMSCWR, DMSCWT,
DMSDIO, DMSDSK, DMSERS, DMSEXC, DMSFNS,
DMSINT, DMSITE, DMSITI, DMSITP, DMSITS,
DMSLFS, DMSMOD, DMSQRY, DMSRNM, DMSTPE,
DMSTQQ

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

LUBTAB

　　Assembled as part of DMSNUC.

　　Referenced by: DMSBOP, DMSCLS, DMSDLB,
DMSFCH, DMSLLU, DMSOPL, DMSPRV, DMSRRV,
DMSSET, DMSSRV, DMSXCP

NICL

Assembled as part of DMSNUC.

Referenced by: DMSBOP, DMSCLS, DMSDLB, DMSCLK, DMSDSV, DMSLLU, DMSPRV, DMSXCP

NUCON

Assembled as part of DMSNUC.

Referenced by: DMSABN, DMSACC, DMSACF, DMSACM, DMSALU, DMSAMS, DMSARE, DMSARN, DMSARX, DMSASM, DMSASN, DMSAUD, DMSBAB, DMSBOP, DMSBRD, DMSBSC, DMSBTB, DMSBTP, DMSBWR, DMSCAT, DMSCIO, DMSCIT, DMSCLS, DMSCMP, DMSCPF, DMSCPY, DMSCRD, DMSCWR, DMSCWT, DMSDBD, DMSDBG, DMSDIO, DMSDLB, DMSCLK, DMSDMP, DMSDOS, DMSDSK, DMSDSL, DMSDSV, DMSEDI, DMSEDX, DMSERR, DMSERS, DMSEXC, DMSEXT, DMSFCH, DMSFET, DMSFLD, DMSFNS, DMSFOR, DMSFRE, DMSGIO, DMSGLB, DMSGND, DMSHDI, DMSHDS, DMSHSM, DMSINA, DMSINI, DMSINM, DMSINS, DMSINT, DMSIOW, DMSITE, DMSITI, DMSITP, DMSITS, DMSLBM, DMSLBT, DMSLDR, DMSLDS, DMSLFS, DMSLGT, DMSLIB, DMSLIO, DMSLKD, DMSLLU, DMSLOA, DMSLSB, DMSLST, DMSLSY, DMSMDP, DMSMOD, DMSMVE, DMSOPL, DMSOPT, DMSOR1, DMSOVR, DMSOVS, DMSPIO, DMSPNT, DMSPRT, DMSPRV, DMSQRY, DMSRDC, DMSRNE, DMSRNM, DMSROS, DMSRRV, DMSSAB, DMSSBS, DMSSCN, DMSSCT, DMSSEB, DMSSET, DMSSLN, DMSSMN, DMSSOP, DMSSQS, DMSSRT, DMSSRV, DMSSSK, DMSSTT, DMSSVN, DMSSVT, DMSSYN, DMSTIO, DMSTPD, DMSTPE, DMSTQQ, DMSTYP, DMSUPD, DMSVIB, DMSVIP, DMSVSR, DMSXCP, DMSZAP

OPSECT

Assembled as part of DMSNUC.

Referenced by: DMSABN, DMSARX, DMSASM, DMSCRD, DMSCWR, DMSCWT, DMSDBG, DMSEXC, DMSEXT, DMSHSM, DMSINS, DMSINT, DMSROS, DMSSBD, DMSSBS, DMSSCT, DMSSEB, DMSSOP, DMSSQS, DMSSVN, DMSSVT

OSFST

Built by: DMSROS

Released by: DMSALU

Referenced by: DMSABN, DMSALU, DMSBOP, DMSDLK, DMSFCH, DMSMVE, DMSROS, DMSRRV, DMSSOP, DMSSRV, DMSSTT

OVSECT

Built by: N/A

Released by: N/A

DMSITS, DMSOVR

PCTAB

Assembled as part of DMSNUC.

Referenced by: DMSBAB, DMSDOS, DMSITP

PDSSECT

Built by: DMSSVT

Released by: DMSSVT

Referenced by: DMSSVT

PGMSECT

Assembled as part of DMSNUC.

Referenced by: DMSITP, DMSSAB, DMSSLN, DMSSMN, DMSSVT

PIBADR

Assembled as part of DMSNUC.

Referenced by: DMSBAB, DMSDOS, DMSITP

PIB2TAB

Assembled as part of DMSNUC.

Referenced by: DMSBAB, DMSDOS, DMSITP

PUBADR

Assembled as part of DMSNUC.

Referenced by: DMSBOP, DMSCLS, DMSDLK, DMSDSV, DMSLLU, DMSPRV, DMSXCP

SSAVE

Built by: DMSITS

Released by: DMSITS

Referenced by: DMSABN, DMSBSC, DMSDBG, DMSDLB, DMSERR, DMSFLD, DMSFRE, DMSITP, DMSITS, DMSLDR, DMSOVS, DMSSMN

SUBSECT

Assembled as part of DMSNUC.

DMSABN, DMSINM, DMSINT

SVCSECT

Assembled as part of DMSNUC.

Referenced by: DMSCIT, DMSFRE, DMSHDS, DMSINT, DMSOVR, DMSOVS, DMSSLN

SVEARA

Assembled as part of DMSNUC.

Referenced by: DMSBAB, DMSDOS, DMSITP

SYSCOM

Assembled as part of DMSNUC.

Referenced by: DMSBAB, DMSBOP, DMSDOS, DMSFET, DMSITP, DMSSMN

SYSNAMES

Assembled as part of DMSNUC.

Referenced by: DMSAMS, DMSBOP, DMSDOS, DMSEDX, DMSEXC, DMSINT, DMSSET, DMSVIB, DMSVSR

TSOBLKS

Assembled as part of DMSNUC.

Referenced by: DMSSET, DMSZAP

USERSECT

Assembled as part of DMSNUC.

No CMS references.

RSCS CONTROL BLOCKS REFERENCES

ASYNE

Built by: DMTASY

Released by: DMTASY, DMTASK

Referenced by: DMTASY, DMTEXT, DMTIOM, DMTSIG

EUFDSECT

Built by: DMTSML

Released by: DMTSML

Referenced by: DMTSML

COMDSECT

Built by: DMTCOM

Released by: N/A

Referenced by: DMTAXS, DMTCMX, DMTMGX, DMTNPT, DMTREX, DMTSML

DEVTABLE

Built by: DMTNPT

Released by: DMTNPT

Referenced by: DMTNPT

FREEE

Built by: DMTQRQ

Released by: DMTQRQ

Referenced by: DMTASK, DMTINI, DMTQRQ

GIVEE

Built by: DMTGIV

Released by: DMTAKE, DMTASK

Referenced by: DMTAKE, DMTASK, DMTGIV

IOE

Built by: DMTIOM

Released by: DMTIOM

Referenced by: DMTASK, DMTIOM, DMTREX

IOTABLE

Built by: DMTIOM, DMTCRE, DMTNPT, DMTREX, DMTSML

Released by: DMTNPT, DMTSML

Referenced by: DMTAXS, DMTCMX, DMTCRE, DMTINI, DMTIOM, DMTREX, DMTSML

LINKTABL

Assembled into DMTSYS at system generation; also built by DMTCMX.

Released by: DMTCMX

Referenced by: DMTAXS, DMTCMX, DMTCOM, DMTCRE, DMTLAX, DMTMGX, DMTNPT, DMTREX, DMTSML

REQBLOCK

Built by: DMTNPT

Released by: DMTNPT

Referenced by: DMTNPT

SVECTORS

Assembled into DMTVEC at system generation; resides in the RSCS nucleus.

Referenced by: DMTAKE, DMTASK, DMTASY, DMTAXS, DMTCMX, DMTCOM, DMTCRE, DMTDSP, DMTEXT, DMTGIV, DMTINI, DMTIOM, DMTLAX, DMTMGX, DMTNPT, DMTQRQ, DMTREX, DMTSIG, DMTSML, DMTSTO, DMTSVC, DMTWAT

TAG

Built by: DMTAXS

Released by: DMTAXS

Referenced by: DMTAXS, DMTCMX, DMTNPT, DMTSML

TAGAREA

Built by: DMTAXS

Released by: n/a

Referenced by: DMTAXS

TANKDSEC

Built by: DMTSML

Released by: DMTSML

Referenced by: DMTSML

TASKE

   Built by: DMTASK

   Released by: DMTASK

   Referenced by: DMTAKE, DMTASK, DMTASY,
   DMTAXS, DMTCOM, DMTDSP, DMTEXT, DMTGIV,
   DMTINI, DMTIOM, DMTNPT, DMTPST, DMTREX,
   DMTSIG, DMTSML, DMTSTO, DMTSVC, DMTWAT

TCTDSECT

   Built by: DMTSML

Released by: DMTSML

Referenced by: 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

Title: IBM Virtual Machine Facility/370:   **Order No.** SY20-0884-0
Data Areas and Control Block Logic

Please check or fill in the items; adding explanations/comments in the space provided.

Which of the following terms best describes your job?

☐ Customer Engineer   ☐ Manager   ☐ Programmer   ☐ Systems Analyst
☐ Engineer   ☐ Mathematician   ☐ Sales Representative   ☐ Systems Engineer
☐ Instructor   ☐ Operator   ☐ Student/Trainee   ☐ Other (explain below)

How did you use this publication?
☐ Introductory text   ☐ Reference manual   ☐ Student/ ☐ Instructor text
☐ 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

Thank you for your cooperation. No postage necessary if mailed in the U.S.A.

SY20-0884-0
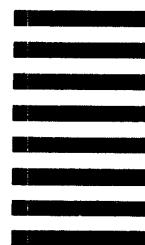
FOLD   FOLD

FIRST CLASS
PERMIT NO. 172
BURLINGTON, MASS.

BUSINESS   REPLY   MAIL
NO POSTAGE STAMP NECESSARY IF MAILED IN U.S.A.

POSTAGE WILL BE PAID BY

IBM CORPORATION
VM/370 PUBLICATIONS
24 NEW ENGLAND EXECUTIVE PARK
BURLINGTON, MASS. 01803

FOLD   FOLD

IBM

International Business Machines Corporation
Data Processing Division
1133 Westchester Avenue, White Plains, New York 10604
(U.S.A. only)

IBM World Trade Corporation
821 United Nations Plaza, New York, New York 10017
(International)

IBM VM/370: Data Areas and Cntrl. Block Logic   Printed in U.S.A.   SY20-0884-0

SY20-0884-0

IBM